Integration of prognostics at a system level: a Petri net approach

Manuel Chiachío[1], Juan Chiachío[2], Shankar Sankararam[3], and John Andrews[4]

[1,2,4] *Resilience Engineering Research Group, Faculty of Engineering, University of Nottingham, NG7 2RD, Nottingham, UK*
manuel.chiachio-ruano@nottingham.ac.uk
juan.chiachioruano@nottingham.ac.uk
john.andrews@nottingham.ac.uk

[3] *NASA Ames Research Center, Intelligent Systems Division, Moffett Field, CA, 94035-1000, USA*
shankar.sankararam@nasa.gov

Abstract

This paper presents a mathematical framework for modeling prognostics at a system level, by combining the prognostics principles with the Plausible Petri nets (PPNs) formalism, first developed in M. Chiachío *et al.* [*Proceedings of the Future Technologies Conference, San Francisco, (2016), pp. 165-172*]. The main feature of the resulting framework resides in its efficiency to jointly consider the dynamics of discrete events, like maintenance actions, together with multiple sources of uncertain information about the system state like the probability distribution of *end-of-life*, information from sensors, and information coming from expert knowledge. In addition, the proposed methodology allows us to rigorously model the flow of information through logic operations, thus making it useful for nonlinear control, Bayesian updating, and decision making. A degradation process of an engineering sub-system is analyzed as an example of application using condition-based monitoring from sensors, predicted states from prognostics algorithms, along with information coming from expert knowledge. The numerical results reveal how the information from sensors and prognostics algorithms can be processed, transferred, stored, and integrated with discrete-event maintenance activities for nonlinear control operations at system level.

1. Introduction

In prognostics, the integration of the predicted information at a system level encompasses two distinct research challenges. First is about predicting the change in system performance and remaining life estimation through an adequate combination of the degradation rates and states of health of individual components. Second, and perhaps most important, to integrate system-level nonlinearities and uncertainties with the predicted information from prognostics. Here, system-level nonlinearities are understood as uncertain environmental elements that affect the system operation irrespectively of the component-wise state of health or degradation, like intervention processes (e.g. maintenance actions), resource availability, *ad hoc* synchronization of components, influence of expert knowledge, etc. In the literature, the majority of prognostics research to date has been focused on individual components, and determining their *end-of-life* (EOL) and *remaining useful life* (RUL), e.g.: (Chiachío, Chiachío, Saxena, & Goebel, 2016; Zio & Peloni, 2011; Myötyri, Pulkkinen, & Simola, 2006; Saha, Celaya, Wysocki, & Goebel, 2009; Daigle & Kulkarni, 2013). Besides, few attempts can be encountered describing system-level prognostics methodologies. Generally, these attempts provide the EOL of a system based on its constituent components and how they interact, like in Gomez, Rodrigues, Galvo, and Yoneyama (2013), where a system-level approach was developed using fault tree analysis from the RUL of individual components. Daigle, Bregon, and Roychoudhury (2014) provided a distributed architecture to model system-level prognostics based on the concept of structural model decomposition whereby the solution of independent local prognostics subproblems were integrated to obtain prognostics at system-level. Liu, Xu, Xie, and Kuo (2014) studied multi-component maintenance models with economic dependence for components that degrade in a continuous manner. More recently, an analytic framework has been provided in (Khorasgani, Biswas, & Shankararaman, 2016) for combining the degradation rate of individual components to predict the variation in system performance over time. Nonetheless, to the authors best known, holistic methodologies for prognostics with integration of system or sub-system level nonlinearities and uncertainties, still remain missing in the literature.

In this paper, a novel holistic framework is proposed for modeling prognostics at a system level by using Petri nets (PNs) (Petri, 1962). In particular, the newly developed *Plausible Petri nets* (PPNs) (Chiachío, Chiachío, Prescott, & Andrews, 2016) are used to integrate uncertain information about the system, like the probability density function (PDF) of EOL for multiple components, information from sensors, expert knowledge, etc., with the dynamics of discrete events like system-level health indicators, maintenance activities, resource availability, to cite but any. Consequently, the approach has the advantage of being able to integrate by first time information from prognostics (e.g. EOL, RUL, etc.), with decision making aspects like go/no-go decisions for maintenance actions. Moreover, the uncertainty is intrinsically accounted for in PPNs, since its formulation stems from combining the information theory principles with the PNs technique, as will be shown further below.

To exemplify some of the problems that can be solved by the proposed methodology, several toy examples are formulated through a set of PPN architectures. Next, the framework is tested using a numerical example to model decision-making over a two-component engineering system under degradation using prognostics information, expert knowledge, and maintenance actions, which is just some of the challenges faced in system-level prognostics applications using PPNs.

The remainder of the paper is organized as follows. Section 2 briefly overviews basic concepts about prognostics and PNs before introducing the fundamentals of PPNs. The PPNs formalism as well as their execution semantics, are succinctly described in §3. A set of examples of PPN architectures for system-level prognostics are provided and discussed in §4. Section 5 illustrates our approach through a numerical example about two-component degrading system. Finally, Section 6 gives concluding remarks.

2. BASIC CONCEPTS

2.1. Foundations of prognostics

Prognostics is concerned with predicting the future health state of engineering systems or components given current degree of wear or damage, and, based on that, estimating the remaining time beyond which the system is expected not to perform its intended function within desired specifications (Chiachío, Chiachío, Sankararaman, Saxena, & Goebel, 2015). In the PHM community, the aforementioned *remaining time* is typically referred as RUL. The potential of prognostics in positively contributing to safety and cost relies in its capacity to provide anticipated information about an anomalous or faulty condition. This information can be used for risk reduction in go/no-go decision, cost optimization through the scheduling of maintenance as-needed, and improved asset availability.

Broadly, formal approaches for prognostics fall into three primary categories (Javed, Gouriveau, & Zerhouni, 2017; Khorasgani et al., 2016): (1) data driven techniques, (2) model based, and (3) hybrid approaches, depending on how the fault propagation process is modeled. Irrespective of the type of modeling approach chosen for prognostics, two main distinct research problems can be devised: (i) the *estimation* problem, which determines the current state of health of the system, and (ii) the *failure prediction* problem, by which the EOL and RUL can be obtained from predictions of the future state of the system ℓ-steps forward in time in absence of new observations. For the estimation problem, the component, sub-system, or system state of health or degradation is typically assumed to be represented using a stochastic variable $\mathbf{x}_k$. This state variable evolves over time k following a specific dynamic equation given in state-space form (Chiachío et al., 2015; Zio & Peloni, 2011), as follows:

$$\mathbf{x}_k = f_k(\mathbf{x}_{k-1}, \mathbf{u}_k, \mathbf{v}_k, \boldsymbol{\theta}) \tag{1a}$$

$$\mathbf{y}_k = h_k(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k, \boldsymbol{\theta}) \tag{1b}$$

where $\mathbf{u}_k \in \mathbb{R}^{n_u}$ is an input vector, $\mathbf{y}_k \in \mathbb{R}^{n_y}$ is a measurement vector, and $\mathbf{v}_k \in \mathbb{R}^{n_v}$ and $\mathbf{w}_k \in \mathbb{R}^{n_w}$ are uncertain variables introduced to account for the model error and measurement error, respectively. The functions f_k and h_k are possibly nonlinear functions for the state transition evolution and observation equation, respectively. This model is sequentially evaluated at every time step k, and produces updated information about the system state $\mathbf{x}_k$ as long as new measurements are available. Next, prognostics algorithms can be employed to project the state predictions into future in absence of new observations. Then, by having defined a failure region $\mathcal{F}$, the EOL can be obtained as the earliest time index $k + \ell, \ell \geqslant 1$ when the event $[\mathbf{x}_{k+\ell} \in \mathcal{F}]$ occurs. It can be computed as:

$$\text{EOL}_k = \inf\big\{k + \ell \in \mathbb{N} : \ell \geqslant 1 \wedge \mathbb{I}_{(\mathcal{F})}(\mathbf{x}_{k+\ell}) = 1\big\} \tag{2}$$

where $\mathbb{I}_{(\mathcal{F})}$ is an indicator function that maps a given point in the $\mathbf{x}$-space to the Boolean domain $\{0, 1\}$, such that $\mathbb{I}_{(\mathcal{F})} = 1$ if $\mathbf{x} \in \mathcal{F}$, and 0 otherwise. The RUL predicted from time k can be straightforwardly obtained from EOL_k as $\text{RUL}_k = \text{EOL}_k - k$. Note that due to the probabilistic nature of the state variable given by $\mathbf{x}_k$, then EOL and RUL are also described as stochastic variables.

Finally, it is important to remark that in this work, the focus is not on predicting the EOL or RUL of a system as a probability, but on methodology that integrates EOL and RUL within an asset management framework at a system level, as will be described next.

2.2. Basis of Petri nets

Petri Nets (PNs) are bipartite directed graphs (digraph) used for modeling the dynamics of systems. The underlying graph of a PN consists of two kinds of nodes, *transitions* and *places*, where *arcs* are either from a place to transition or vice versa.

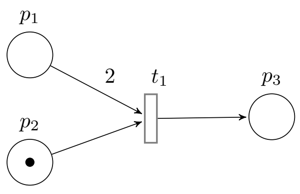


Figure 1. Example of Petri net of three places and one transition.

A place represents a particular state of the system or activity being modeled (e.g. considering health management modeling, places can be used to indicate the current state of a component or sub-system, if the performance of this component or sub-system is currently being tested by a maintenance team to determine its state, or if any maintenance activity is currently in progress). Places are temporarily visited by *tokens*, the abstract moving units of a PN. The distribution of tokens over the PN at a specific time of execution is referred to as *marking*, which is expressed as a vector indicative of the state of the PN. The transitions are responsible of the dynamic behavior of the PN, and enable the system to move from one state to another. For example, a component wear process is one of such processes which can be reflected using a transitions (Andrews, Prescott, & De Rozières, 2014). In graphical representation, places are typically expressed using circles while transitions are drawn as bars or boxes. Arcs are labeled with their corresponding *weights*, non-negative integer values indicating the amount of parallel arcs (1 by default). Figure 1 is provided to serve as illustrative example of a PN of three places (p_1, p_2, p_3), and one transition (t_1).

From a mathematical point of view, a PN can be defined as an ordered 6-tuple $\mathfrak{N}$ as follows (Murata, 1989):

$$\mathfrak{P} \triangleq \left\langle \mathbf{P}, \mathbf{T}, \mathbf{E}, \mathbf{M}_0, \mathbf{D}, \mathbf{W} \right\rangle \tag{3}$$

where $\mathbf{P} \in \mathbb{N}^{n_p}$ and $\mathbf{T} \in \mathbb{N}^{n_t}$ denote the set of n_p places and n_t transitions of the PN respectively, $\mathbf{M}_0 \in \mathbb{N}^{n_p}$ is the initial marking, and $\mathbf{D} \in \mathbb{R}^{n_t}$ is the non-negative vector of switching delays of the transitions (0 by default). The set $\mathbf{E} \subset \mathbb{N}^{n_p} \times \mathbb{N}^{n_t}$ represents the edges (also referred to as *arcs*), which are expressed through ordered pairs of nodes to indicate the connections between places and transitions, i.e. $\mathbf{E} \subseteq (\mathbf{P} \times \mathbf{T}) \cup (\mathbf{T} \times \mathbf{P})$. As mentioned above, each edge has assigned a weight (1 by default) within the set of weights $\mathbf{W}$.

At a certain state k, the PN dynamics can be described through an algebraic equation defined by:

$$\mathbf{M}_{k+1} = \mathbf{M}_k + \mathbf{A}^T \mathbf{u}_k \tag{4}$$

where $\mathbf{u}_k$ is the *firing vector* at k, a n_t-dimensional vector of Boolean values whose elements are obtained according to the *firing rule*. $\mathbf{A} \in \mathbb{N}^{n_t \times n_p}$ is the *incidence matrix* of the graph, whose elements are the result of subtracting the *forward* $(\mathbf{A}^+)$ and *backward* $(\mathbf{A}^-)$ *incidence matrices* respectively, as follows:

$$\mathbf{A} = \mathbf{A}^+ - \mathbf{A}^- \tag{5}$$

where

$$\mathbf{A}^+ = \begin{pmatrix} a^+_{11} & a^+_{12} & \cdots & a^+_{1n_p} \\ a^+_{12} & a^+_{22} & \cdots & a^+_{2n_p} \\ \vdots & & \ddots & \\ a^+_{n_t 1} & a^+_{n_t 2} & \cdots & a^+_{n_t n_p} \end{pmatrix} \quad \mathbf{A}^- = \begin{pmatrix} a^-_{11} & a^-_{12} & \cdots & a^-_{1n_p} \\ a^-_{12} & a^-_{22} & \cdots & a^-_{2n_p} \\ \vdots & & \ddots & \\ a^-_{n_t 1} & a^-_{n_t 2} & \cdots & a^-_{n_t n_p} \end{pmatrix} \tag{6}$$

The element a^+_{ij} from Eq. (6) represents the weight of the arc from transition t_i to output place p_j, whereas a^-_{ij} is the weight of the arc to transition t_i from input place p_j, $i = 1, \ldots, n_t$, $j = 1, \ldots, n_p$. If transition t_i is activated at state k, then $u_{i,k} \in \mathbf{u}_k$ is modified according to the *firing rule*, which can be expressed as follows:

$$u_{i,k} = \begin{cases} 1, & \text{if} \quad M(j) \geqslant a^-_{ij} \ \ \forall p_j \in {}^{\bullet}\mathbf{P}_{t_i} \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

where $M(j) \in \mathbb{N}$ is the marking for place p_j, and ${}^{\bullet}\mathbf{P}_{t_i}$ denotes the set of places that belong to the preset of transition $t_i, i = 1, \ldots, n_t$. For example in the PN from Fig. 1, ${}^{\bullet}\mathbf{P}_{t_1} = \{p_1, p_2\}$.

Note that by means of PNs and their marking, the behavior of complex engineering systems can be described in terms of discrete system states and their changes over time. The following rules summarize the algebra of PNs as explained above:

1. Transitions always consume from all the input arcs at the same time and produce from all out-coming arcs the same amount;
2. Transition t_i is enabled if every input place p_j from its preset ${}^{\bullet}\mathbf{P}_{t_i}$ is marked with at least a^-_{ij} tokens;
3. An enabled transition t_i will fire once the delay time $\tau_i \in D$ has passed;
4. After firing, transition t_i removes a^-_{ij} tokens from p_j, and adds a^+_{ij} tokens to each j-th output place of t_i.

Similarly to other formalisms for system modeling like Bayesian networks, Artificial Neural Networks, etc., PNs suffer from the well-known “state explosion” problem, which is particularly common in real-life engineering applications, where the number of states increases exponentially with the size of the system. Furthermore, it should be noted that PNs are not well-suited to deal with uncertainty since their output are

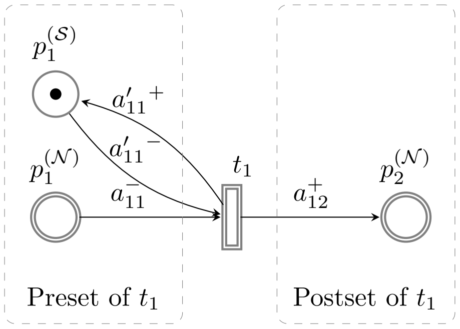


Figure 2. Illustration of a sample PPN with two numerical places ($p_1^{(\mathcal{N})}$, $p_2^{(\mathcal{N})}$), one symbolic place ($p_1^{(\mathcal{S})}$), and one transition (t_1).

based on sequences of Boolean operations. Thus, over the last decades, researchers have enhanced the original PNs to handle (1) uncertainty (Konar & Mandal, 1996; Looney, 1988; Cao & Sanderson, 1993; Bugarin & Barro, 1994; Zhou & Zain, 2016; Cardoso, Valette, & Dubois, 1999), and (2) continuous variables to efficiently reproduce the hybrid aspect of systems without the need of employing complex graphs (David, 1997; Antsaklis, 2000; Júlvez, Di Cairano, Bemporad, & Mahulea, 2014; Vazquez & Silva, 2015; Silva, 2016). Plausible Petri Nets (PPNs) (Chiachío, Chiachío, Prescott, & Andrews, 2016) can be classified as one of these variants of the PNs developed to better reproduce the nature of engineering systems, hence giving partial response to the referred drawbacks, as will be shown next.

3. PLAUSIBLE PETRI NETS (PPNS)

Plausible Petri nets (PPNs) are a class of PNs recently developed by the authors, which are based on a combination of discrete and continuous numerical processes whose values may be uncertain (*plausible*). Two interacting subnets form the PPN graph: 1) a *symbolic subnet*, where the tokens are objects in the sense of integer moving units, as in classical PNs (Petri, 1962), 2) a *numerical subnet*, where tokens are *states of information* [1]. The resulting framework is a hybrid variant of PNs where the sets of nodes $\{\mathbf{P}, \mathbf{T}\}$ are partitioned into two disjoint subsets, namely numerical and symbolic, which are denoted using superscripts $(\mathcal{N})$ and $(\mathcal{S})$, respectively. In particular, the set of places $\mathbf{P}$ are partitioned into subset $\mathbf{P}^{(\mathcal{N})} \in \mathbb{N}^{n_p}$ and $\mathbf{P}^{(\mathcal{S})} \in \mathbb{N}^{n'_p}$, such that $\mathbf{P}^{(\mathcal{N})} \cup \mathbf{P}^{(\mathcal{S})} = \mathbf{P}$, and $\mathbf{P}^{(\mathcal{N})} \cap \mathbf{P}^{(\mathcal{S})} = \emptyset$. Superscripts n_p, n'_p represent the number of numerical and symbolic places, respectively. Analogously, transitions $\mathbf{T}$ are partitioned into numerical transitions $\mathbf{T}^{(\mathcal{N})} \in \mathbb{N}^{n_t}$ and symbolic transitions $\mathbf{T}^{(\mathcal{S})} \in \mathbb{N}^{n'_t}$, where $\mathbf{T}^{(\mathcal{N})} \cup \mathbf{T}^{(\mathcal{S})} = \mathbf{T}$, and $\mathbf{T}^{(\mathcal{N})} \cap \mathbf{T}^{(\mathcal{S})} \neq \emptyset$. In this case, n_t, n'_t denote the number of numerical and symbolic transitions, respectively. Observe that those transitions that belong to $\mathbf{T}^{(\mathcal{N})} \cap \mathbf{T}^{(\mathcal{S})}$ are referred to as *mixed transitions* (Chiachío, Chiachío, Prescott, & Andrews, 2016).

In PPNs, the referred states of information about a system state variable $\mathbf{x}_k \in \mathcal{X}$ are denoted by $f^p(\mathbf{x}_k)$ and $f^t(\mathbf{x}_k)$ for numerical places and transitions, respectively. In practical terms, these states of information can be understood as probability density functions (PDFs) except for a normalizing constant. Thus, the marking $\mathbf{M}_k$ of a PPN at a certain time k consists in a combined vector $\mathbf{M}_k = \left(\mathbf{M}_k^{(\mathcal{N})}, \mathbf{M}_k^{(\mathcal{S})}\right)$, where $\mathbf{M}_k^{(\mathcal{N})}$ and $\mathbf{M}_k^{(\mathcal{S})}$ are column vectors of normalized PDFs and integer values, respectively. Moreover, as in classical PNs, there exist arc weights for the symbolic places, denoted by $a_{ij}^{'+}, a_{ij}^{'-} \in \mathbf{W}^{(\mathcal{S})} \subset \mathbb{N}$, whereby the incidence matrix $\mathbf{A}^{(\mathcal{S})}$ can be obtained by Eq. (5). The arc weights for the numerical places are denoted by $a_{ij}^{+}, a_{ij}^{-} \in \mathbf{W}^{(\mathcal{N})} \subset \mathbb{R}^{+}$, such that $\mathbf{A}^{(\mathcal{N})} = \left[a_{ij}^{+}\right] - \left[a_{ij}^{-}\right]$, and $i = 1, \ldots, n_t$, $j = 1, \ldots, n_p$, where n_t, n_p represent the amount of numerical transitions and numerical places of the PPN, respectively. Note that the arc weights from the symbolic subnet, e.g. $(a_{11}'^{-})$ are differentiated from the numerical ones using an accent ($'$). In graphical representation, numerical nodes are represented using double line, whereas single line is used for the rest. A PPN model is shown in Fig. 4 for illustration purposes. The dashed rectangles shown in Fig. 4 are to highlight the preset and postset of t_1.

3.1. Execution semantics

As stated in the last section, the marking at time k of PPNs consists of both types of information given by $\mathbf{M}_k^{(\mathcal{N})}$ for the numerical places, and $\mathbf{M}_k^{(\mathcal{S})}$ for the case of symbolic places. The marking evolution of the symbolic subnet corresponds to the state equation of a PN (Murata, 1989) (recall Eq. [4]). However, the evolution of $\mathbf{M}_k^{(\mathcal{N})}$ relies on an *ad hoc* information flow dynamics based on two basic operations (Chiachío, Chiachío, Prescott, & Andrews, 2016): the *conjunction* and *disjunction* of states of information (Tarantola & Valette, 1982; Rus et al., 2016). Both markings evolve in a synchronized manner, as will be described in next section. In these operations, the first principles from Boolean logic, in particular the logic operators AND ($\wedge$) and OR ($\vee$), are invoked to allow the continuos information from the numerical subnet to be exchanged into the PPN. More specifically, they enable the combination and aggregation of states of information across the numerical subnet of a PPN. To avoid repeating literature but conferring a sufficient conceptual framework, the conjunction and disjunction of states of information have been briefly explained and illustrated in Fig. 3. The interested reader is referred to (Tarantola & Valette, 1982) for further

[1] A state of information can be described by a set of numerical values about a state variable, along with a mapping over them that assigns each numerical value with its *relative plausibility* (Tarantola & Valette, 1982; Rus, Chiachío, & Chiachío, 2016).

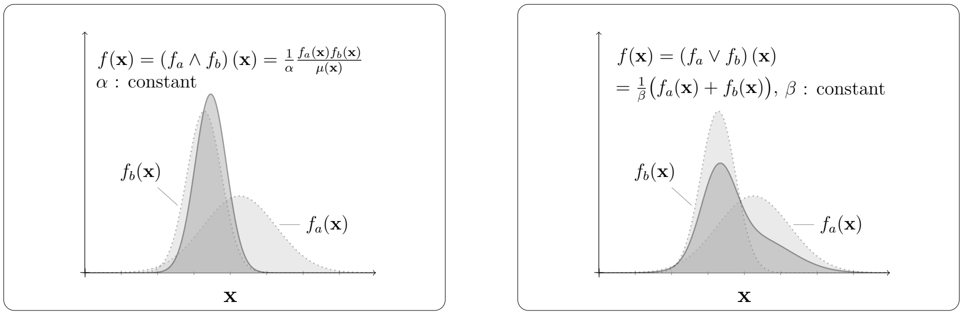


Figure 3. Illustration of the conjunction (left) and disjunction (right) of two arbitrary states of information, namely $f_a(\mathbf{x})$ and $f_b(\mathbf{x})$.

details.

From this standpoint, the dynamics of PPNs is formulated under the adoption of the following rules (Chiachío, Chiachío, Prescott, & Andrews, 2016):

1. An input arc from place $p_j^{(\mathcal{N})}$ to transition $t_i \in \mathbf{T}^{(\mathcal{N})}$ conveys a state of information given by $a_{ij}^{-}\big(f^{p_j} \wedge f^{t_i}\big)(\mathbf{x}_k)$, which remains in $p_j^{(\mathcal{N})}$ after transition t_i has fired;
2. Transition $t_i \in \mathbf{T}^{(\mathcal{N})}$ produces to an output arc a state of information given by $a_{ij}^{+}\big(f^{t_i} \wedge f^{\bullet\mathbf{P}_{t_i}}\big)(\mathbf{x}_k)$, where $f^{\bullet\mathbf{P}_{t_i}}(\mathbf{x}_k)$ denotes the resulting density from the disjunction of the states of information of the preset of t_i. As stated in Fig. (3), the normalized version of $f^{\bullet\mathbf{P}_{t_i}}(\mathbf{x}_k)$ can be obtained as:

$$f^{\bullet\mathbf{P}_{t_i}}(\mathbf{x}_k) = \frac{1}{\beta}\big(f^{p_1} + f^{p_2} + \cdots + f^{p_m}\big)(\mathbf{x}_k) \qquad (8)$$

   where β is a constant, and $p_1, \ldots, p_m \in {}^{\bullet}\mathbf{P}_{t_i} \subset \mathbf{P}^{(\mathcal{N})}$;
3. After firing numerical transition t_i, the state of information resulting in place $p_j^{(\mathcal{N})}$ from the postset of t_i, is the disjunction of the state of information $f^{p_j}(\mathbf{x}_k)$ (the previous state of information), and $a_{ij}^{+}\big(f^{t_i} \wedge f^{\bullet\mathbf{P}_{t_i}}\big)(\mathbf{x}_k)$ (the information produced after firing transition t_i). In mathematical terms:

$$f^{p_j}(\mathbf{x}_{k+1}) = \Big(f^{p_j} \vee a_{ij}^{+}\big(f^{t_i} \wedge f^{\bullet\mathbf{P}_{t_i}}\big)\Big)(\mathbf{x}_k) \qquad (9)$$

The rules given above are sufficient to explain the information flow dynamics of PPNs. Notwithstanding, observe that they are mostly based on conjunction of states of information which requires the evaluation of normalizing constants involving an intractable integral. Also, note that there are situations where the conjunction is conducted using density functions which are not completely known, perhaps because they are defined trough samples. Particle methods (Arumlampalam, Maskell, Gordon, & Clapp, 2002; Doucet, De Freitas, & Gordon, 2001) can be used in these cases to circumvent the evaluation of the normalizing constant with a feasible computational cost. In particle methods, a set of N samples $\big\{\mathbf{x}^{(n)}\big\}_{n=1}^{N}$ with associated weights $\big\{\omega^{(n)}\big\}_{n=1}^{N}$ are used to obtain an approximation for the required density function [e.g. $\big(f_a \wedge f_b\big)(\mathbf{x})$], as follows:

$$\big(f_a \wedge f_b\big)(\mathbf{x}) \approx \sum_{n=1}^{N} \omega^{(n)}\delta\big(\mathbf{x} - \mathbf{x}^{(n)}\big) \qquad (10)$$

where δ is the Dirac delta and $\mathbf{x}^{(n)} \sim \big(f_a \wedge f_b\big)(\mathbf{x})$. The particle weight $\omega^{(n)}$ represents the likelihood value of $\mathbf{x}^{(n)}$, and is representative of the plausibility of $\mathbf{x}^{(n)}$ when it is distributed according to $\big(f_a \wedge f_b\big)(\mathbf{x})$. It can be evaluated for the case of $\mathcal{X}$ being a linear space as follows (Tarantola & Mosegaard, 2007):

$$\omega^{(n)} = \frac{f_a(\mathbf{x}^{(n)})f_b(\mathbf{x}^{(n)})}{\sum_{n=1}^{N} f_a(\mathbf{x}^{(n)})f_b(\mathbf{x}^{(n)})} \qquad (11)$$

A pseudocode implementation to obtain a particle approximation from the conjunction $\big(f_a \wedge f_b\big)(\mathbf{x})$, is provided in the Appendix as Algorithm 1.

3.1.1. Firing rule

In PPNs, any transition $t_i \in \mathbf{T}$ is fired at time k if the delay time has passed and:

1. Every symbolic place from the preset of t_i has enough tokens according to their input arc weight, as in classical PNs;
2. Each of the conjunction of states of information between f^{t_i} and $f_k^{p_j}$ is possible, where $p_j^{(\mathcal{N})}$ belongs to the preset of t_i;
3. Conditions (a) and (b) are both satisfied when t_i is a mixed transition, i.e. $t_i \in (\mathbf{T}^{(\mathcal{S})} \cap \mathbf{T}^{(\mathcal{N})})$.

Note from Condition (b) that a conjunction, e.g. $\big(f_k^{p_j} \wedge f_k^{t_i}\big)(\mathbf{x}_k)$, is possible if $\big(f_k^{p_j} \wedge f_k^{t_i}\big)(\mathbf{x}_k) \neq \emptyset$ (Tarantola & Valette, 1982). Note also that when any of the states of information

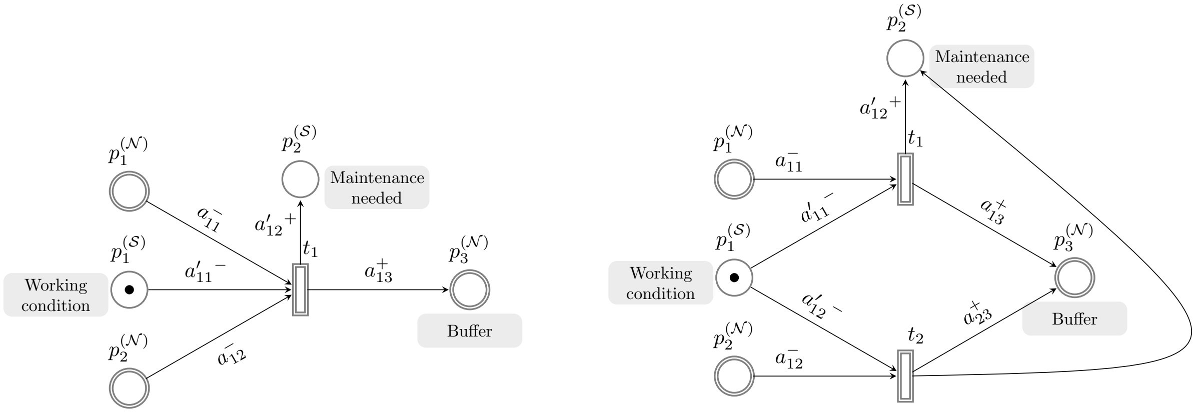


Figure 4. Plausible Petri nets of the examples given in §4.1.

involved in a conjunction is the *homogenous density* (also referred to as "non-informative density") $\mu(\mathbf{x}_k)$ of the state space of consideration $\mathcal{X}$, then the conjunction is always possible (Tarantola & Valette, 1982; Chiachío, Chiachío, Prescott, & Andrews, 2016), thus Condition (b) is automatically fulfilled. This argument is important in terms of using PPNs in practical applications, as will be demonstrated in next section.

4. System level PHM by PPNs

In this section, a set of PPN sample architectures are provided to illustrate how our PPNs can be used for decision making at a system level in applications where information from prognostics along with other sources of information (like expert knowledge, sensors, etc.) may coexist, as usual in practice. The examples have been kept as simple as possible since they are mostly intended to serve as guideline to built more complex PPNs. Moreover, the examples provide representative architectures whereby to conceptualize the proposed PNN methodology in a prognostics and health management context.

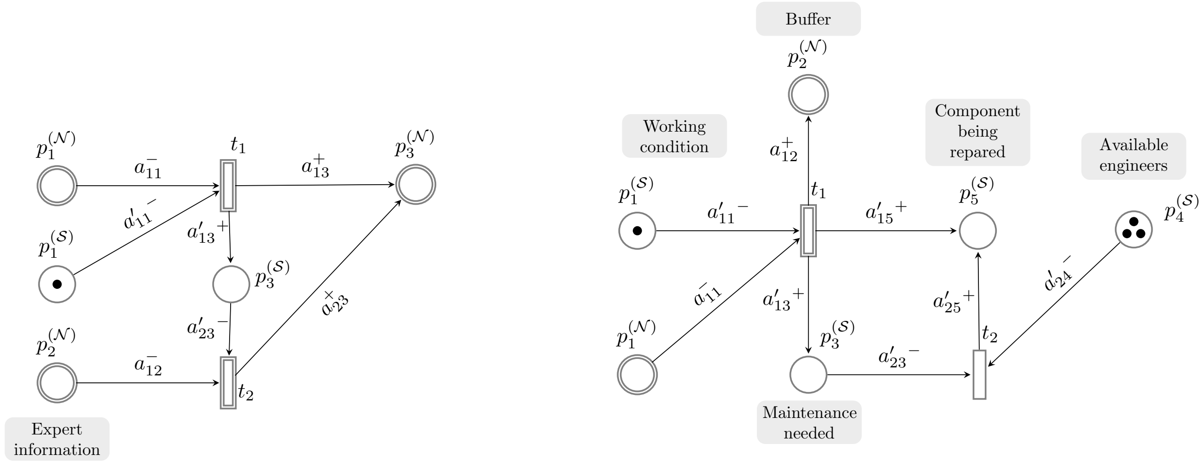
4.1. Integrative decision making for multi-component prognostics

A couple of PPN architectures are exemplified here for modeling decision making aspects in presence of multiple information about the EOL from different components of an engineering system. Figure 4 shows two PPNs of three numerical $\{p_1^{(\mathcal{N})}, p_2^{(\mathcal{N})}, p_3^{(\mathcal{N})}\}$ and two symbolic $\{p_1^{(\mathcal{S})}, p_2^{(\mathcal{S})}\}$ places, along with one mixed transition t_1 (two transitions for Fig. 4b). This example assumes that numerical places $p_1^{(\mathcal{N})}$ and $p_2^{(\mathcal{N})}$ enclose uncertain information about the EOL of two components or sub-systems, which is expressed through a PDF denoted by f^{EOL_1} and f^{EOL_2}, respectively. Transition t_1 is defined based on condition (Chiachío, Chiachío, Prescott, & Andrews, 2016) using an indicator function for the state space that assigns a value of 1 when the expected value of EOL reaches a specific threshold denoted by $\epsilon \in \mathbb{R}$, and 0 otherwise, as indicated in Table 1. Provided that place $p_1^{(\mathcal{S})}$ has one token (assumed that $a'_{11} = 1$ in this example), then, according to the firing rule given in §3.1.1, transition t_1 is activated once the expected EOL from both components or subsystems has reached the threshold value ϵ (not necessarily at the same time), whereupon the system turns to a "maintenance needed" state. The resulting information is collected in place $p_3^{(\mathcal{N})}$ which acts as a buffer of information. This buffer can be used for diagnostics purposes since it collects a weighted distribution of plausible EOL values from the two-component system, conditioned to $\mathbb{E}(EOL_k) > \epsilon$. As stated by the execution semantics of PPNs (refer to §3.1), the resulting PDF in $p_3^{(\mathcal{N})}$ can be described as:

$$f^{p_3} = \frac{a_{13}^+}{a_{11}^- + a_{12}^-}\Big(a_{11}^- f^{\text{EOL}_1} + a_{12}^- f^{\text{EOL}_2}\Big) \tag{12}$$

Note from the last equation that when $a_{11}^- = a_{12}^- = a_{13}^+ = 1$, then $f^{p_3} = \frac{1}{2}\Big(f^{\text{EOL}_1} + f^{\text{EOL}_2}\Big)$, an averaged sum of both PDFs of EOL.

In Fig. 4b, an analogous PPN architecture is illustrated for the two-component system. However in this case, the graph represents a two-component system acting in series ($t_2 \equiv t_1$), thus it turns to the "maintenance needed" state once the expected value of EOL from any of the components reaches the threshold value ϵ. Note that, both PPN architectures can be straightforwardly extended to the case of multiple components or sub-systems provided that the PDF of EOL from those components are known. Finally observe that through these simple graph architectures, go/no-go decisions can be made using uncertain information from multiple components acting in series, parallel, or a combination from them.



(a) (b)

Figure 5. Illustration of the exemplified PPN architecture explained in §4.2 (left) and §4.3 (right).

4.2. Combining expert knowledge and prognostics measures

In this section, an example of PPN is provided which includes information about the PDF of EOL of a component or subsystem, along with information coming from expert knowledge. Figure 5a illustrates the PPN consisting of three numerical places, two symbolic places, and two transitions. Let us now assume that the numerical place $p_1^{(\mathcal{N})}$ comprises the PDF of EOL of a component given by f^{EOL}, whereas $f^{p_3} = \emptyset$, initially. Next, the proposed PPN architecture also encompasses information from one expert in place $p_2^{(\mathcal{N})}$, which is represented by a PDF given by f^{Expert} (e.g. f^{Expert} can be a uniform PDF of EOL representing an interval of possible values of EOL). As in the last example, transition t_1 is defined based on condition, such that it is fired if the uncertainty[2] of the PDF f^{EOL} reaches or exceeds a specific threshold value. If t_1 is fired, then place $p_2^{(\mathcal{S})}$ receives a token that enables the information from the expert to be transferred to place $p_3^{(\mathcal{N})}$. Note that through this exemplified PPN, a decision making process can be modeled about the adoption of information from an expert if the uncertainty about the EOL is higher than a certain value ξ. For example, this can be as a consequence of a faulty sensor, or a perturbed prognostic estimation. The resulting information in $p_3^{(\mathcal{N})}$ includes the PDF f^{EOL} coming from $p_1^{(\mathcal{N})}$ and that from the expert, as follows:

$$f^{p_3} = \frac{a_{13}^+}{a_{11}^- + a_{12}^-}\left(a_{11}^- f^{\mathrm{EOL}_1} + a_{12}^- f^{\mathrm{Expert}}\right) \tag{13}$$

If required, higher relevance can be conferred to the expert information by increasing the weight a_{12}^- with respect to a_{11}^-. Finally, observe that the PDF f^{p_3} can be further used in an operational context (e.g., for diagnostics), as was described in last section.

[2] For example by calculating the differential entropy of the PDF of EOL (Chiachío, Beck, Chiachío, & Rus, 2014; Chiachío, Chiachío, Prescott, & Andrews, 2016)

Table 1. Description of the transitions of the PPN shown in Figs. 4a, 4b, 5a, and 5b.

Reference	Transition	Type	Condition	State of information
Fig. 4	t_1	Mixed	$\mathcal{C}_1 = \left\{x_k \in \mathcal{X} : \mathbb{E}_{f^{\mathrm{EOL}_1}}(x_k) \leqslant \epsilon\right\}$	$f^{t_1} \sim \mathbb{I}_{\mathcal{C}_1}(x_k)$
	t_2 (Fig.4b)	Mixed	$\mathcal{C}_2 = \left\{x_k \in \mathcal{X} : \mathbb{E}_{f^{\mathrm{EOL}_2}}(x_k) \leqslant \epsilon\right\}$	$f^{t_2} \sim \mathbb{I}_{\mathcal{C}_2}(x_k)$
Fig. 5a	t_1	Mixed	$\mathcal{C}_1 = \left\{x_k \in \mathcal{X} : \mathbb{E}_{f^{\mathrm{EOL}}}(x_k) \leqslant \epsilon\right\}$	$f^{t_1} \sim \mathbb{I}_{\mathcal{C}_1}(x_k)$
	t_2	Mixed	None (homogeneus)	$f^{t_2} \sim \mu(x_k)$
Fig. 5b	t_1	Mixed	$\mathcal{C}_1 = \left\{x_k \in \mathcal{X} : \mathbb{E}_{f^{\mathrm{EOL}}}(x_k) \leqslant \epsilon\right\}$	$f^{t_1} \sim \mathbb{I}_{\mathcal{C}_1}(x_k)$
	t_2	Symbolic	τ (delay)	Not applicable

4.3. Integration of prognostics and resource availability

This example illustrates the case of a decision making process, like activation of a maintenance action, taken after the PDF of EOL has reached a certain value ϵ as in §4.1, except that in this case it occurs contingent upon availability of engineers. Fig. 5b shows the idealized system using a PPN of two numerical places, four symbolic places, one mixed transition, and one symbolic transition. Numerical place $p_1^{(\mathcal{N})}$ is assumed to represent the PDF f^{EOL} of a component. As in the example shown in Fig. 4a, t_1 is fired once the expectation of f^{EOL} has reached the threshold ϵ provided that $p_1^{(\mathcal{S})}$ has enough amount of tokens according to ${a'_{11}}^{-}$, which for the sake of illustration, is assumed to be $a'_{11} = 1$. Next, let us also assume that symbolic transition t_2 represents the activation of the maintenance activity (an activation delay τ can be assigned to t_2 (Andrews et al., 2014), although this information is irrelevant here). Observe that when t_1 is fired, then t_2 will be activated if the number of available engineers is higher or equal to ${a'_{24}}^{-}$. In such case, $p_5^{(\mathcal{S})}$ is marked, then the system turns to "component being repaired" state.

5. NUMERICAL EXAMPLE

The PPN framework presented above is exemplified here using a numerical example to illustrate some of the advantages of using PPNs for integration of prognostics at a system level. In this example, a self-managed two-component system is modeled through a PPN which comprises uncertain information about the EOL of two degrading components acting in series, along with expert knowledge about the whole system's EOL. Figure 6 illustrates the idealized system model through a PPN of four numerical places, four symbolic places, and five transitions. Noisy measurements about the components' state of degradation are assumed to be available, whereby an estimation of the EOL can be obtained, provided that there exists a specified degradation threshold, and an appropriate prognostics algorithm to make predictions (see for example (Chiachío, Chiachío, Shankararaman, & Andrews, 2017)). Let us denote by $\text{EOL}_k^{(j)} \in \mathcal{X} \subset \mathbb{R}^+$ a stochastic variable corresponding to the EOL of the j-th component, $j = 1, 2$, which evolves over time $k \in \mathbb{N}$ following a dynamic equation $\text{EOL}_k^{(j)} = h_k(x_{k-1}^{\text{EOL}_j}, \theta_j)$ given by:

$$\text{EOL}_k^{(j)} = e^{-\theta_j k}\text{EOL}_{k-1}^{(j)} + v_k \tag{14}$$

where θ_j is an uncertain decay parameter whose values are modeled as a Gaussian centered at 0,006 and 0,008 for $j = 1, 2$, respectively, and 100% of coefficient of variation in both cases. The term v_k represents a measurement error which is assumed to be modeled as a zero-mean Gaussian density function with standard deviation given by $\sigma_v = 5$. In the PPN, the mixed transitions t_i, $i = \{1, 2, 4\}$ are defined by condition, thus their states of information can be expressed using Diract Delta density functions (Tarantola, 2005), i.e., $f^{t_i}(\text{EOL}_k) \sim \mathbb{I}_{\mathcal{C}_i}(\text{EOL}_k)$. Henceforth, their activation is prescribed for the stochastic variable $\text{EOL}_k^{(j)}$ on fulfilling the condition $\text{EOL}_k^{(j)} \in \mathcal{C}_i$, such that:

$$\mathcal{C}_1 = \left\{\text{EOL}_k \in \mathcal{X} : \mathbb{E}_{f^{p_1}}(\text{EOL}_k) \leqslant \epsilon\right\} \tag{15a}$$

$$\mathcal{C}_2 = \left\{\text{EOL}_k \in \mathcal{X} : \mathbb{E}_{f^{p_2}}(\text{EOL}_k) \leqslant \epsilon\right\} \tag{15b}$$

$$\mathcal{C}_3 = \left\{\text{EOL}_k \in \mathcal{X} : H(\text{EOL}_k) > \xi\right\} \tag{15c}$$

where $\epsilon = 20$ and $\xi = 5$. In (15c), H denotes the differential entropy of EOL_k, that can be obtained by evaluating

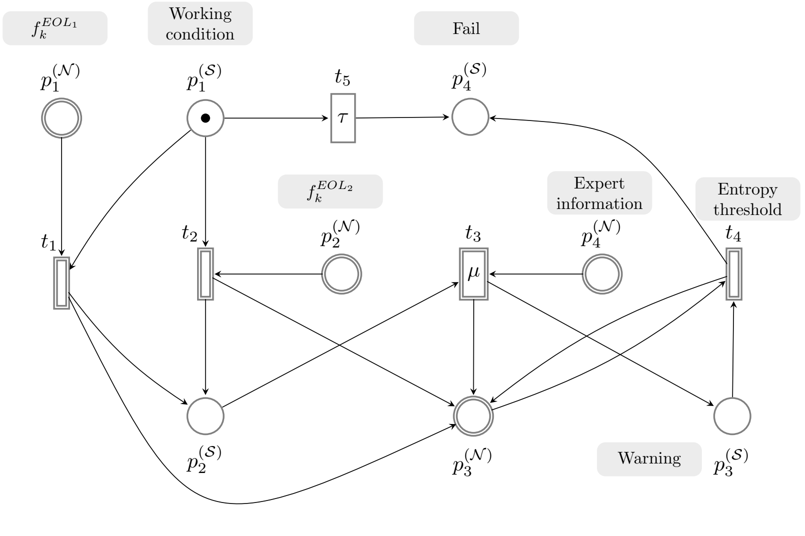
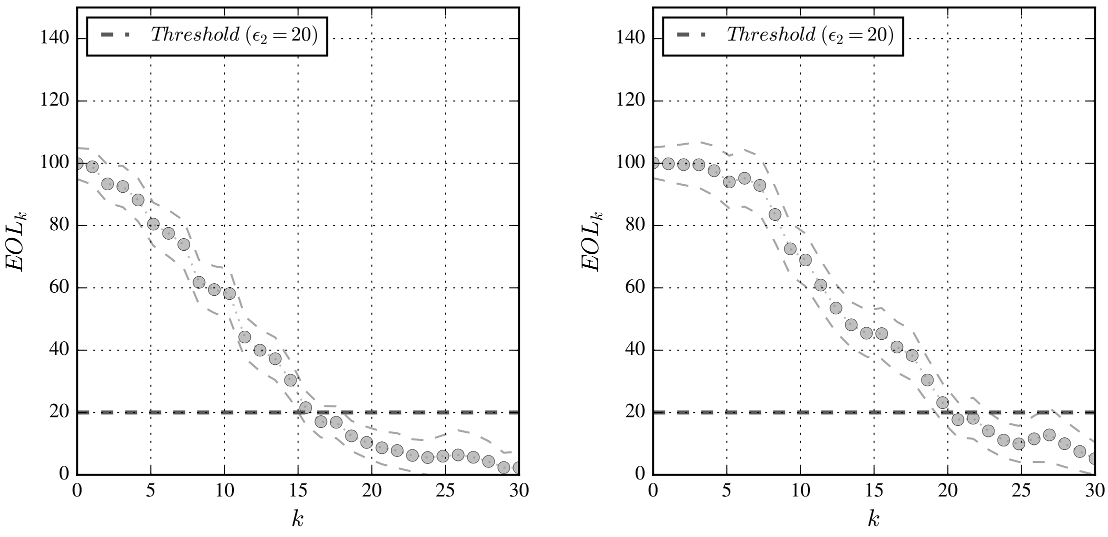


Figure 6. Illustration of the PPN from the numerical example given in Section 5.



(a)

(b)

Figure 7. Plots of the evolution over time of the states of information about the stochastic variable $\mathrm{EOL}_k^{(j)}$ for components $j = 1, 2$ (panels [a] and [b], respectively) from the Plausible Petri net of the example given in §5. The expectation of the states of information is represented using gray circles. The dashed lines represent the 5th-95th probability band.

$1/2 \ln\left[(2\pi e) var(\mathrm{EOL}_k)\right]$ as a measure quantifying the uncertainty of EOL_k. The initial marking of the numerical places is given by $f_0^{p_1} \sim \mathcal{N}(100, 5)$, $f_0^{p_2} \sim \mathcal{N}(100, 5)$, $f_0^{p_3} = \emptyset$, and $f^{p_4} = \mathcal{U}[18, 22]$. The latter represents expert knowledge about EOL, which is given by a uniform PDF defined over the interval $[18, 22]$.

Initially at $k = 0$, the system starts in the "working condition" state represented by one token at $p_1^{(\mathcal{S})}$, thus $\mathbf{M}_0^{(\mathcal{S})} = (1, 0, 0, 0)^T$. Once any of the expected values of the predicted EOL of components 1 and 2 (represented in place $p_1^{(\mathcal{N})}$ and $p_2^{(\mathcal{N})}$, respectively) has reached the threshold value ϵ, then transitions t_1, t_2 (not necessarily both nor simultaneously) produce one token to $p_2^{(\mathcal{S})}$ which enables transition t_3 to be fired, whereupon the expert information about system's EOL enters into play and is transferred to $p_3^{(\mathcal{N})}$. Next, the system turns to a "warning" state, and a decision is made conditioned upon: 1) the quality of the information given by $f_k^{p_3}$, and 2) the total time spent by the system under no failure states. The differential entropy (DE) is used in this example as a quality indicator of the information in place $p_3^{(\mathcal{N})}$, so that the transition t_4 is activated if the DE of $f_k^{p_3}$ is higher than the threshold $\xi = 5$. In such case, the system turns to "inspection needed" state, otherwise the system remains in "warning" mode. In this example, the time spent by the system in performing transition t_5 (which corresponds to an activation time given by τ) is assumed to represent a scheduled periodic maintenance activity such that if $k > \tau$, then the system directly turns to the "maintenance needed" condition irrespective of the component's degradation state nor their predicted EOL.

For the numerical evaluation of the PPN in Fig. 6, the execution semantics rules (recall §3.1) along with Eq. (4), are applied in confluence with the firing rule for the system state evolution described through the marking $\mathbf{M}_k$. The algorithm for particle approximation of conjunction of states of information described in the Appendix, has been applied using $N = 1000$. Note that the disjunction of states of information

Table 2. Summary of the discrete events taking place when running the PPN shown in Fig. 6. The second and third column show the symbolic marking and firing vector, respectively.

Time	$\mathbf{M}_k^{(\mathcal{S})}$	$\mathbf{u}_k$	Event
$k = 0$	$(1\,0\,0\,0)$	$(0\,0\,0\,0\,0)$	–
$k = 1$	$(1\,0\,0\,0)$	$(0\,0\,0\,0\,0)$	–
$\vdots$		$\ddots$	
$k = 16$	$(1\,0\,0\,0)$	$(1\,0\,0\,0\,0)$	t_1 fired
$k = 17$	$(0\,1\,0\,0)$	$(0\,0\,0\,0\,0)$	t_3 enabled
$k = 18$	$(0\,1\,0\,0)$	$(0\,0\,1\,0\,0)$	t_3 fired
$k = 19$	$(0\,0\,1\,0)$	$(0\,0\,0\,0\,0)$	t_4 enabled
$k = 20$	$(0\,0\,1\,0)$	$(0\,0\,0\,1\,0)$	t_4 fired
$k = 21$	$(0\,0\,0\,1)$	$(0\,0\,0\,0\,0)$	–

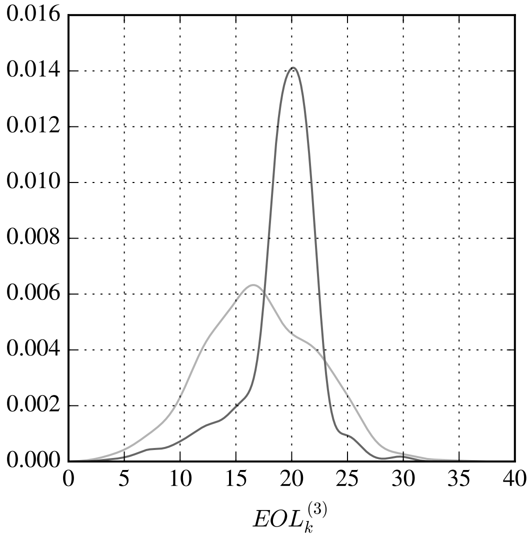


Figure 8. Kernel density estimates of the state of information about EOL in place $p_3^{(\mathcal{N})}$ for $k = 17$ and $k \geqslant 18$ (darker gray color).

can be straightforwardly evaluated using samples by just joining the samples from the component-wise density functions, and affecting their particle weights using an appropriate normalizing constant so as to obtain a bone fide density. Figure 7 shows the results for the two-components' EOL (from places $p_1^{(\mathcal{N})}$ and $p_2^{(\mathcal{N})}$) for time indexes $k = 0$ to $k = 30$. Note from Fig. 7 that component 1 first reaches the threshold, i.e., the expectation of $\text{EOL}_k^{(1)}$ first reaches the value $\epsilon = 20$ at $k = 16$. Next, transition t_3 is enabled and the information coming from the expert is aggregated to $p_3^{(\mathcal{N})}$ at $k = 18$. Figure 8 shows the resulting PDF in $p_3^{(\mathcal{N})}$ before and after the information from the expert was incorporated. Observe the influence of the expert knowledge in terms of gain of information, which makes the distribution of plausible EOL values being more concentrated around the recommended values from the expert, namely [18,22]. Notwithstanding, the DE of the resulting state of information given by f^{p_3} at $k \geqslant 18$ is higher than $\xi = 5$, then the system finally turns to the "maintenance needed" state. A summary of the results for the symbolic subnet of the PPN model is provided in Table 2.

This example illustrates that uncertain information about the EOL of different components along with information from experts, can be integrated into a system level model for prognostics and decision making. The numerical results confirm that system non linearities, apart from that attributable to the stochastic variable EOL_k, can be taken into account. Examples of the referred non linearities are: *ad hoc* synchronies between components, user-defined time to failure, resources availability, etc., which makes our PPNs useful for prognostics and health management at a system level.

6. Conclusions

This paper presented a novel prognostics methodology to integrate information from prognostics with decision making aspects at a system level using Plausible Petri nets. The approach has the advantage of addressing the prognostics problem as a unified system-level approach where multiple sources of uncertain information can be integrated with discrete-events, conferring high versatility to better reproduce real-world problems of prognostics without the need of using extremely complex nets, as usual when adopting other formalisms. Further research is needed to investigate suitable PPN architectures to better integrate the influence of intervention activities within the predicted state of health at a system level.

Acknowledgment

This work was supported by the Lloyd's Register Foundation, a charitable foundation in the UK helping to protect the life and property by supporting engineering-related education, public engagement, and the application of research.

References

Andrews, J., Prescott, D., & De Rozières, F. (2014). A stochastic model for railway track asset management. *Reliability Engineering & System Safety*, *130*, 76–84.

Antsaklis, P. J. (2000). Special issue on hybrid systems: theory and applications a brief introduction to the theory and applications of hybrid systems. *Proceedings of the IEEE*, *88*(7), 879-887.

Arumlampalam, M., Maskell, S., Gordon, N., & Clapp, T. (2002). A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, *50*(2), 174–188.

Bugarin, A. J., & Barro, S. (1994). Fuzzy reasoning supported by Petri nets. *IEEE Transactions on Fuzzy Systems*, *2*(2), 135–150.

Cao, T., & Sanderson, A. C. (1993). Variable reasoning and analysis about uncertainty with fuzzy Petri nets. In *International conference on application and theory of Petri nets* (p. 126-145).

Cardoso, J., Valette, R., & Dubois, D. (1999). Possibilistic Petri nets. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, *29*(5), 573-582.

Chiachío, J., Chiachío, M., Sankararaman, S., Saxena, A., & Goebel, K. (2015). Prognostics design for structural health management. In *Emerging design solutions in structural health monitoring systems* (pp. 234–273). IGI Global.

Chiachío, M., Beck, J. L., Chiachío, J., & Rus, G. (2014). Approximate Bayesian computation by Subset Simulation. *SIAM Journal on Scientific Computing*, *36*(3), A1339-A1358.

Chiachío, M., Chiachío, J., Prescott, D., & Andrews, J.

(2016). An information theoretic approach for knowledge representation using Petri nets. In *Proceedings of Future Technologies Conference, 6-7 December 2016, San Francisco* (pp. 165–172).

Chiachío, M., Chiachío, J., Saxena, A., & Goebel, K. (2016). An energy-based prognostic framework to predict evolution of damage in composite materials. In *Structural health monitoring (shm) in aerospace structures* (p. 447-477). Woodhead Publishing-Elsevier.

Chiachío, M., Chiachío, J., Shankararaman, S., & Andrews, J. (2017). A new algorithm for prognostics using Subset Simulation. *Reliability Engineering & System Safety*, in press.

Daigle, M., Bregon, A., & Roychoudhury, I. (2014). Distributed prognostics based on structural model decomposition. *IEEE Transactions on Reliability*, *63*(2), 495-510.

Daigle, M., & Kulkarni, S. (2013). Electrochemistry-based battery modeling for prognostics. In *Proceedings of the annual conference of the prognostics and health management society, 2013* (Vol. 1, p. 249-261).

David, R. (1997). Modeling of hybrid systems using continuous and hybrid Petri nets. In *Proceedings of the seventh international workshop on Petri nets and performance models (PNPM'97), 1997.* (pp. 47–58).

Doucet, A., De Freitas, N., & Gordon, N. (2001). An introduction to sequential Monte Carlo methods. In A. Doucet, N. De Freitas, & N. Gordon (Eds.), *Sequential Monte Carlo methods in practice* (pp. 3–14). Springer.

Gomez, J., Rodrigues, L., Galvo, R., & Yoneyama, T. (2013). System level rul estimation for multiple-component systems. In *Proceedings of Annual Conference of the Prognostics and Health Management Society* (p. 74-83).

Javed, K., Gouriveau, R., & Zerhouni, N. (2017). State of the art and taxonomy of prognostics approaches, trends of prognostics applications and open issues towards maturity at different technology readiness levels. *Mechanical Systems and Signal Processing*, *94*, 214–236.

Júlvez, J., Di Cairano, S., Bemporad, A., & Mahulea, C. (2014). Event-driven model predictive control of timed hybrid Petri nets. *International Journal of Robust and Nonlinear Control*, *24*(12), 1724-1742.

Khorasgani, H., Biswas, G., & Shankararaman, S. (2016). Methodologies for system-level remaining useful life prediction. *Reliability Engineering and System Safety*, *154*, 8-18.

Konar, A., & Mandal, A. K. (1996). Uncertainty management in expert systems using fuzzy Petri nets. *IEEE Transactions on Knowledge and Data Engineering*, *8*(1), 96-105.

Liu, B., Xu, Z., Xie, M., & Kuo, W. (2014). A value-based preventive maintenance policy for multi-component system with continuously degrading components. *Reliability Engineering & System Safety*, *132*, 83-89.

Looney, C. G. (1988). Fuzzy Petri nets for rule-based decision making. *IEEE Transactions on Systems, Man, and Cybernetics*, *18*(1), 178–183.

Murata, T. (1989). Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, *77*(4), 541-580.

Myötyri, E., Pulkkinen, U., & Simola, K. (2006). Application of stochastic filtering for lifetime prediction. *Reliability Engineering and System Safety*, *91*(2), 200–208.

Petri, C. A. (1962). *Kommunikation mit automaten* (Unpublished doctoral dissertation). Institut fr Instrumentelle Mathematik an der Universitt Bonn.

Rus, G., Chiachío, J., & Chiachío, M. (2016). Logical inference for inverse problems. *Inverse Problems in Science and Engineering*, *24*(3), 448-464.

Saha, B., Celaya, J. R., Wysocki, P. F., & Goebel, K. F. (2009). Towards prognostics for electronics components. In *Aerospace conference, 2009 ieee* (pp. 1–7).

Silva, M. (2016). Individuals, populations and fluid approximations: A Petri net based perspective. *Nonlinear Analysis: Hybrid Systems*, *22*, 72–97.

Tarantola, A. (2005). *Inverse problem theory and methods for model parameters estimation*. SIAM.

Tarantola, A., & Mosegaard, K. (2007). *Mapping of probabilities, theory for the interpretation of uncertain physical measurements*. Cambridge University Press.

Tarantola, A., & Valette, B. (1982). Inverse problems = quest for information. *Journal of Geophysics*, *50*(3), 159-170.

Vazquez, C. R., & Silva, M. (2015). Stochastic hybrid approximations of Markovian Petri nets. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, *45*(9), 1231–1244.

Zhou, K.-Q., & Zain, A. M. (2016). Fuzzy Petri nets and industrial applications: a review. *Artificial Intelligence Review*, *45*(4), 405-446.

Zio, E., & Peloni, G. (2011). Particle filtering prognostic estimation of the remaining useful life of nonlinear components. *Reliability Engineering and System Safety*, *96*(3), 403–409.

BIOGRAPHIES

Manuel Chiachío is Postdoctoral Research Fellow at the Resilience Engineering Research Group, University of Nottingham, UK. He holds a PhD in Structural Mechanics (*international mention*) awarded by the University of Granada, (Spain), and a MSc in Civil Engineering (2007), and also a MSc in Structural Engineering (2011), by the same University. His research focus on uncertainty quantification methods and algorithms, risk and reliability analysis, and artificial intelligence methods in application to a variety of engineering areas, which range from structural and mechanical engineering to bioengineering applications. During the course of his PhD work, Manuel worked as guest scientist at world-class universities and institutions, like Hamburg University of Technology (Germany), California Institute of Technology (Caltech), and NASA Ames Research Center (USA). This research has led to several publications in highly ranked journals. It has been awarded by the National Council of Education of Spain through one of the prestigious FPU fellowships, by the Andalusian Society of promotion of the Talent, by the European Council of Civil Engineers (ECCM) with the Silver Medal prize in the 1st European Contest of Structural Design (2008), and also by the Prognostics and Health Management Society with a Best Paper Award in 2014. Prior to joining the University of Granada in 2011, Manuel worked as a consultancy engineer for four years in top engineering companies in Spain.

Juan Chiachío is a Research Fellow in Infrastructure Asset Management in the Resilience Engineering Research Group at the University of Nottingham (UK). He received his PhD in Structural Engineering (*international mention*) in 2014 by the University of Granada, Spain. In addition, he holds a MSc in Structural Engineering (2011) and a MSc in Civil Engineering (2007), both by the University of Granada. His research is focused on translating reliability and prognostics methods into the life-cycle analysis of structural and infrastructural systems subjected in-service degradation. This research has led to several publications in highly ranked journals, a best-paper award and nominations in major conferences, and it has been awarded by the Spanish National Council of Education through one of the FPU annual fellowships, by the Andalusian Society of Promotion of Talent, by the Prognostics and Health Management Society with a Best Paper Award, and by the European Council of Civil Engineers. In addition, his work has attracted the interest of world-class institutions for collaborative research, like the Prognostics Center of Excellence of NASA, the California Institute of Technology, and the Hamburg University of Technology (Germany). His current research at the University of Nottingham deals with the development of a Bayesian prognostics framework for infrastructure asset management, under EPSRC project titled ”Whole-life cost assessment of novel material railway drainage systems (EP/M023028/1)”.

Shankar Sankararaman received his B.S. degree in Civil Engineering from the Indian Institute of Technology in Madras (2007) and later, obtained his Ph.D. in Civil Engineering from Vanderbilt University, Nashville, Tennessee, U.S.A. in 2012. His research focuses on the various aspects of uncertainty quantification, integration, and management in different types of aerospace, mechanical, and civil engineering systems. His research interests include probabilistic methods, risk and reliability analysis, Bayesian networks, system health monitoring, diagnosis and prognosis, decision-making under uncertainty, treatment of epistemic uncertainty, and multidisciplinary analysis. He is a member of the Non-Deterministic Approaches (NDA) technical committee at the American Institute of Aeronautics, the Probabilistic Methods Technical Committee (PMC) at the American Society of Civil Engineers (ASCE), and the Prognostics and Health Management (PHM) Society. Currently, Shankar is a researcher at NASA Ames Research Center, Moffett Field, CA, where he develops algorithms for uncertainty assessment and management in the context of system health monitoring, prognostics, and decision-making. He has been recently named as virtual member of the Resilience Engineering Research Group, University of Nottingham (UK).

John Andrews is Head of the Resilience Engineering Research Group at the University of Nottingham where he holds a Royal Academy of Engineering Research Chair in Infrastructure Asset Management. Prior to this he worked for 20 years at Loughborough University where his final post was Professor of Systems Risk and Reliability. The prime focus of his research has been on methods for evaluating the system resilience, unavailability, unreliability and risk. Much of this work has concentrated on the Fault Tree analysis technique and the use of Binary Decision Diagrams (BDDs) as an efficient and accurate solution method. Recently attention has turned more the degradation modeling and the effects of maintenance, inspection and renewal on asset performance. In this context the modeling he has carried out has extended the Petri net and Bayesian Network capabilities. In 2005, John founded the Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability of which was the Editor-in-chief for 10 years. He is also a member of the Editorial Boards for 6 other international journals in this field.

APPENDIX

Pseudocode implementation to obtain particles from the conjunction of two arbitrary states of information $f_a(\mathbf{x})$ and $f_b(\mathbf{x})$.

Algorithm 1 Particle approximation of conjunction of states of information

```
Inputs: $N, f_a(\mathbf{x}), f_b(\mathbf{x})$ {number of particles and states of information}
Outputs: $\left\{\mathbf{x}^{(n)}, \omega^{(n)}\right\}_{n=1}^{N}$, where $\mathbf{x}^{(n)} \sim (f_a \wedge f_b)(\mathbf{x})$
  Begin
  Sample $\left\{ \left(\tilde{\mathbf{x}}_a^{(n)}, \tilde{\omega}_a^{(n)}\right) \right\}_{n=1}^{N}$ from $f_a(\mathbf{x})$
  Set $\mathbf{x}^{(n)} \leftarrow \tilde{\mathbf{x}}_a^{(n)}$, $n = 1, \ldots, N$
  Set $\hat{\omega}^{(n)} \leftarrow f_b\big(\mathbf{x}^{(n)}\big)$, $n = 1, \ldots, N$ {unnormalized weights}
  Normalize weights $\omega^{(n)} \leftarrow \frac{\hat{\omega}^{(n)}}{\sum_{n=1}^{N} \hat{\omega}^{(n)}}$, $n = 1, \ldots, N$
```