Defining Optimal Maintenance Scope for Multiple k-out-of-n Load-Sharing Production Systems Connected in Series Based on RUL Predictions

Leonardo R. Rodrigues[1]

[1] *Institute of Aeronautics and Space, São José dos Campos, São Paulo, 12228-904, Brazil*
leonardolrr@iae.cta.br

ABSTRACT

This paper presents a method to define the optimal maintenance scope of a production system consisting of multiple k-out-of-n systems connected in series. Maintenance recommendations are based on Remaining Useful Life (RUL) predictions obtained from a Prognostics and Health Management (PHM) system for each production unit within the production system. Defining the techniques applied in order to estimate the degradation level of production units is out of the scope of this paper. It is assumed here that a PHM system is available and provides the degradation level and RUL estimates for each production unit. The goal is to find the maintenance scope that minimizes the expected total cost per cycle until the next maintenance activity. A k-out-of-n load-sharing system is assumed, which means that the failure of a production unit results in a higher load (and consequently a higher degradation rate) on the surviving production units. The total cost comprises the production cost and the maintenance cost. Production cost of each k-out-of-n system is also affected by the number of surviving production units. A preventive maintenance cost is incurred to maintain a degraded but still functional production unit. A corrective maintenance cost is incurred to maintain a failed production unit. An Ant Colony Optimization (ACO) approach is adopted, which allows the proposed method to deal with large instances of the problem. A numerical example is presented to illustrate the application of the proposed method.

1. INTRODUCTION

The manufacturing sector is very competitive and the success or failure of companies in this sector is highly influenced by the operational strategies adopted (Heddy et al., 2015). In order to become more competitive, companies in the manufacturing sector are adopting and implementing the Smart Manufacturing Systems (SMS) concept. Smart Manufacturing Systems are the integration of advanced technologies to enable the implementation of new processes and increase the efficiency of the existing methods (Weiss et al., 2015).

Smart Manufacturing Systems include technologies in a wide range of domains such as automation, decision support, sensing, communication and robotics. The integration of these technologies provides a variety of benefits for companies such as improvements in efficiency and reductions in costs (Weiss et al., 2015), (Bernarden, 2012). Several papers on Smart Manufacturing Systems have been recently published in the literature (Ghonaim, Ghenniwa, & Shen, 2011) (Y. Lu, Morris, & Frechette, 2015) (Brodsky, Krishnamoorthy, Menascé, Shao, & Rachuri, 2014).

One of the topics addressed in the smart manufacturing context is maintenance optimization. Maintenance activities have a direct effect on system availability and are responsible for a relevant part of the operational cost. Defining the maintenance scheduling and the maintenance scope for complex manufacturing systems with redundancy and processes interactions is a challenging task. In order to accomplish this task, smart manufacturing systems require technologies that provide information on the system under consideration and its components. The information required by smart manufacturing systems may include the degradation level and the estimated Remaining Useful Life (RUL) of the system and its components (Jung, Morris, Lyons, Leong, & Cho, 2015). In this context, the use of Prognostics and Health Management (PHM) data emerges as a powerful tool to support the implementation of smart manufacturing systems (Malinowski et al., 2015). PHM technology comprise techniques that enable condition-monitoring, diagnostics and prognostics of components, systems and processes (Vachtsevanos, Lewis, Roemer, Hess, & Wu, 2006).

PHM systems provide the ability to quantify the degradation level and estimate the RUL of a monitored component (Vachtsevanos et al., 2006). These systems collect data from

the monitored component and generate an index to indicate how degraded the component is. The degradation level can be estimated, for instance, based on a comparison between the nominal and the actual performance of the component (Kacprzynski, Roemer, & Hess, 2002).

The term "*prognostics*" refers to the ability of predicting the future health of the monitored components, systems and processes. The main information provided by the prognostic function is the RUL. The term "*health management*" refers to the decision making process to improve maintenance, logistics and operations. This decision making process is based on the information provided by the PHM system and additional available data such as cost models and resources availability. Operational constraints such as facility capacity and budget restrictions are also taken into account in this process (Malinowski et al., 2015), (Barajas & Srinivasa, 2008). The literature on PHM systems acting as a decision support tool is extensive and many papers have been recently published in the literature with solutions for inventory management (Rodrigues & Yoneyama, 2012), task assignment problems (Medeiros, Rodrigues, Shiguemori, Santos, & Nascimento Jr, 2014) and maintenance scheduling (Sandborn & Wilkinson, 2007), (Gebraeel, 2010), (Rodrigues, Medeiros, & Kern, 2015).

In this paper, a method to define the optimal maintenance scope for a production system is presented. A production system consisting of multiple *k*-out-of-*n* systems connected in series is considered. Maintenance scope recommendations are based on RUL predictions of each production unit obtained from a PHM system. A *k*-out-of-*n* load-sharing system is assumed. In a *k*-out-of-*n* load-sharing production system, when a production unit fails and stops working, the remaining production units are subjected to a higher load, resulting in a higher degradation rate (Amari & Bergman, 2008), (Mohammad, Kalam, & Amari, 2013).

The proposed method recommends the maintenance scope based on a cost criterion. Total cost is broken into production cost and maintenance cost. Production cost of each *k*-out-of-*n* production system is influenced by the number of surviving production units. Maintenance cost can be preventive (if maintenance is performed in a degraded but still functioning production unit) or corrective (if maintenance is performed in a failed production unit).

The number of possible solutions (i.e., possible maintenance scopes) grows exponentially with the number of production units within the production system. The Ant Colony Optimization (ACO) algorithm is used in this paper to provide a way to deal with large instances of the problem. The ACO algorithm has been successfully applied in a variety of combinatorial problems, including maintenance scheduling problems (R. Zhou, Lee, & Nee, 2008), (Rodrigues, Gomes, et al., 2015).

The remaining sections of this paper are organized as follows. Section 2 summarizes the related papers in condition-based maintenance (CBM) for multi-component systems. Section 3 describes a *k*-out-of-*n* system and the load-sharing model adopted in this paper. Section 4 presents the formulation of the problem under consideration. Section 5 presents the degradation model used for each production unit and the influences of the load-sharing model in the degradation model parameters. Section 6 introduces the proposed method. Section 7 introduces the ACO algorithm, which is used to find a maintenance scope that reduces the expected total cost per cycle. Section 8 illustrates the application of the proposed method in a numerical example. Section 9 discusses the implementation and some opportunities for further improvements in the proposed method. Concluding remarks are presented in Section 10.

2. Literature Review

Condition-based methods have become a topic of great interest among academic researchers and industry practitioners. In the last few decades, many papers have been published in the literature with component level and system level maintenance CBM methods (Ahmad & Kamaruddin, 2012). Some of the relevant papers are briefly reviewed below.

Camci (2009) discusses the importance of considering the dependencies among components in a CBM optimization method for multiple component systems. Three different types of dependencies are studied, namely, functional dependence, economic dependence and stochastic dependence. A CBM optimization method using Genetic Algorithm is proposed. Resources availability and maintenance conflicts are included as constraints in the proposed method. The paper shows the need of optimizing cost and availability for the whole system rather than applying failure thresholds for individual components in order to fully benefit from CBM.

Y. Zhou, Zhang, Lín, and Ma (2013) present a maintenance optimization method for series-parallel systems. The proposed method considers economic dependencies among components and discrete inspection intervals based on the state of the components in order to maximize the average revenue per time unit. The authors argue that, for many practical systems, continuous monitoring of components may be impractical due to economical or technical issues. Therefore, their model assume that system health condition is updated only during inspections which are carried out at discrete times. The interval between two consecutive inspections is also optimized.

Rodrigues, Gomes, et al. (2015) use a fault tree representation of the system in order to compute a sistem-level RUL based on RUL predictions of individual components. Fault trees are widely adopted in many industry sectors and can be used to represent complex functional dependencies among com-

ponents. The authors present an optimization maintenance method to select a subset of components to be repaired in order to assure a minimum system safety level until the next scheduled maintenance intervention with minimum cost. In the proposed model, failures do not affect the degradation rate of surviving components. The authors show that, for complex systems, it is crucial to assess the impact of each component in the whole system degradation level. The ACO algorithm was used with very satisfactory results. The authors use simulation-based experiments to demonstrate that the proposed method provides better results than simply maintaining the most degraded components without taking into account their influence in the whole system.

Huynh, Barros, and Bérenguer (2015) develop a multi-level CBM decision making method which combines maintenance decisions at both system level and component level. The authors use a *k*-out-of-*n* system to investigate the performance of the proposed method. Economic dependencies among components are represented by opportunistic maintenance strategies. The authors compare the proposed multi-level CBM method with a method that relies only on the RUL of the components. They conclude that the multi-level decision making approach is more efficient in avoiding inopportune maintenance interventions and taking into account the interactions among components.

Rodrigues, Medeiros, and Kern (2015) present a CBM optimization method for multiple identical components that considers the economical benefits of maintaining multiple components at the same time instead of scheduling maintenance interventions for different components at different time instants, based on individual optimization recommendations. The objective function to be minimized is the expected maintenance cost per time unit. A gamma process is used to model the degradation process of components. Components health condition are updated during inspections that are equally distributed in time. However, the authors show that the proposed method can be easily adapted to consider inspections not equally distributed in time.

This paper aims to contribute to the literature on CBM methods for multi-component systems. A method to define maintenance scopes based on the system health condition is proposed. The proposed method uses RUL predictions obtained from a PHM system. Economic and structural dependencies among production units are considered. A load-sharing approach is used to model the influence of a failure event in the surviving production units. Improvements could be obtained by allowing uncertainties in the time required for each maintenance.

3. *k*-OUT-OF-*n* SYSTEMS

Redundancy is a widely used technique to improve system availability. The *k*-out-of-*n*:G system is a popular type of redundancy applied in fault-tolerant systems which can be found in many industrial applications (Moghaddass, Zuo, & Wang, 2011), (Khatab, Nahas, & Nourelfath, 2009). For example, some commercial aircraft have five displays in the cockpit. The aircraft can be dispatched if at least three displays are operating. Thus, the multidisplay system may be represented by a 3-out-of-5:G system. Another common example of a *k*-out-of-*n*:G system can be observed in the automotive industry: Vehicles are usually equipped with one spare tire. The vehicle can be driven if at least four tires are in good condition. Thus, the vehicle may be represented by a 4-out-of-5:G system (Kuo & Zuo, 2003).

An *n*-component system that works if and only if at least *k* of the total *n* components work is called a *k*-out-of-*n*:G system. An *n*-component system that fails if and only if at least *k* of the total *n* components fail is called a *k*-out-of-*n*:F system. Based on these two definitions, a *k*-out-of-*n*:G system is equivalent to an (*n-k*+1)-out-of-*n*:F system (Kuo & Zuo, 2003). In this paper, the term *k*-out-of-*n* system will be used to indicate a *k*-out-of-*n*:G system.

Both series and parallel systems are special cases of a *k*-out-of-*n* system. A series system can be represented by an *n*-out-of-*n* system. Similarly, a parallel system can be represented by a 1-out-of-*n* system (L. Lu & Lewis, 2008).

3.1. *k*-out-of-*n* Load-Sharing Systems

Most methods applied to multiple component systems adopt the assumption of i.i.d. components. However, in many real world applications, a load is shared by several components or systems. Examples of load-sharing systems include electric generators in a power plant and pumps in a hydraulic system. In these applications, when a component fails, the load is distributed by the remaining components, resulting in a higher degradation rate to the surviving components (Yinghui & Jing, 2008).

A *k*-out-of-*n* load-sharing system can be defined as a *k*-out-of-*n* system in which the following properties hold:

- The surviving components share the load.
- The degradation rate of a component is affected by the magnitude of the load it shares.
- The higher the number of failed components, the higher the degradation rate of each surviving component.

The Capacity Flow Model proposes an expression for the increase in the degradation rate of surviving components as a function of the number of failed components (Pozsgai, Neher, & Bertsche, 2003). Let $L_D(y)$ be the load-sharing degradation factor, where y is the number of failed production units in the *k*-out-of-*n* system, with $y = \{0, \ldots, n - k\}$. It is assumed that the load is equally distributed among all surviving components. The expression to compute $L_D(y)$ is presented

in Eq. (1).

$$L_D(y) = \left[\frac{n}{n-y}\right]^{\beta_D} \quad (1)$$

where β_D is the load-sharing degradation exponent that models the additional load shared by the surviving components, with $0 \le \beta_D \le 1$.

When $y = 0$ (i.e., all components are functioning), each component is subjected to its nominal load and $L_D(0) = 1$. If $\beta_D = 1$, the total load is constant. If $\beta_D = 0$, then $L_D(y) = 1, \forall y$ and the fail of a component does not affect the surviving components.

4. PROBLEM STATEMENT

Consider a production system composed by N_S k-out-of-n systems connected in series. Each k-out-of-n system i, with $i = \{1, \ldots, N_S\}$ has n_i production units and works if and only if at least k_i production units are functioning. Figure 1 shows the production system under consideration.

Each k-out-of-n system within the production system is a load-sharing system. In a k-out-of-n load-sharing production system, when the number of failed production units increases, the load per surviving production unit also increases. This process continues until the number of failed production units reaches $n - k + 1$. At this moment, the k-out-of-n system stops working. The whole production system fails as soon as one of the k-out-of-n system fails.

A load-sharing degradation factor $L_D(y)$, where y is the number of failed production units in the k-out-of-n system, is used in order to modify the degradation process of the surviving production units. A detailed description of the degradation process of each production unit and the effects of the load-sharing degradation factor in the degradation model are presented in Section 5.

Let $S = [s_1, \ldots, s_{N_P}]$ be the maintenance scope vector, where
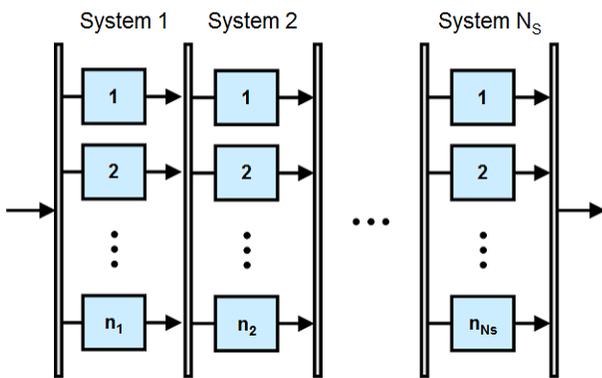
Figure 1. Multiple k-out-of-n systems connected in series.

N_P is the total number of production units in the production system, obtained by Eq. (2). Each element s_i of the maintenance scope vector, with $i = \{1, \ldots, N_P\}$, is a binary variable that assumes value 1 if production unit i is maintained and zero otherwise.

$$N_P = \sum_{i=1}^{N_S} n_i \quad (2)$$

A fixed maintenance cost C_F is incurred if at least one production unit is maintained. A preventive maintenance cost $C_P^{(i)}$ is incurred if maintenance is performed in production unit i while it is still functioning. A corrective maintenance cost $C_C^{(i)}$ is incurred if maintenance is performed in production unit i when it is already failed.

The following assumptions are also made:

- The production system under consideration operates in production cycles. Maintenance activities can be carried out at the end of each cycle.

- The amount of time required for performing a maintenance activity is small and may be neglected.

- Maintenance activities always bring the production unit to an "as good as new" condition (i.e., the degradation level returns to zero).

- A production unit fails and stops working as soon as its degradation level is equal to or greater than a failure threshold level FT.

- A maintenance intervention is required if the probability of the production system finishimg the next production cycle without failing is lower than a safety level SL.

The problem consists in finding the maintenance scope S which minimizes the expected total cost per cycle, $TC(S)$, until the next maintenance activity is performed. The total cost comprises the production cost and the maintenance cost. The steps to compute $TC(S)$ are presented in Section 6.

5. DEGRADATION PROCESS

Let $z_i(t)$ be the expected degradation level of production unit i at the end of cycle t. It is assumed that the increment in the degradation level z_i of each production unit in each production cycle is a random variable that follows a gamma distribution. The degradation evolution of a production unit is illustrated in Fig. (2). The PDF (probability density function) of the gamma distribution is presented in Eq. (3).

$$f(x|w, \theta) = \frac{x^{w-1} \cdot exp^{\frac{-x}{\theta}}}{\Gamma(w) \cdot \theta^w}; x, w, \theta > 0 \quad (3)$$

where w is the shape parameter of the gamma distribution, θ is the scale parameter of the gamma distribution and $f(x|w,\theta)$ is the probability density function evaluated at x given w and θ. $\Gamma(w)$ is the Gamma function evaluated at w, as presented in Eq. (4).

$$\Gamma(w) = \int_{x=0}^{\infty} x^{w-1} \cdot exp^{-x} dx \tag{4}$$

The mean μ and the variance σ^2 of a gamma distribution with shape parameter w and scale parameter θ are given by Eqs. (5) and (6), respectively (van Noortwijk, 2009).

$$\mu = w \cdot \theta \tag{5}$$
$$\sigma^2 = w \cdot \theta^2 \tag{6}$$

It is assumed that, when a production unit fails, the increment in the degradation level of each surviving production unit in the same k-out-of-n system for the next cycles is still a random variable that follows a gamma distribution. The parameters of the resulting gamma distribution are affected by the load-sharing degradation factor according to the following assumptions:

- **Assumption 1:** The mean of the resulting gamma distribution is multiplied by the load-sharing degradation factor $L_D(y)$.

- **Assumption 2:** The variance of the resulting gamma distribution is not affected.

Let μ_y and σ_y^2 be, respectively, the mean and the variance of the gamma distribution that represents the increment in the degradation level of a production unit in one production cycle when y production units in the k-out-of-n system are failed. Also, let w_0 and θ_0 be, respectively, the shape and the scale
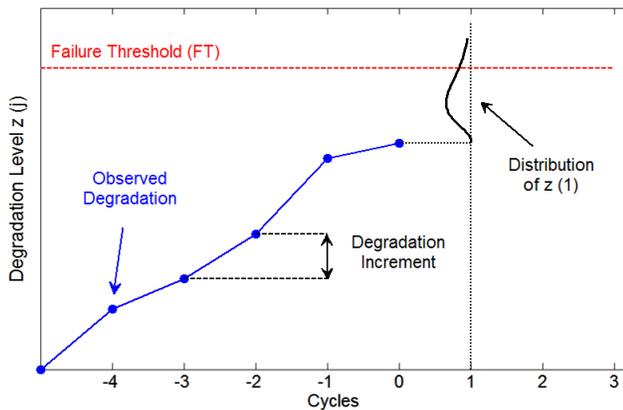
parameter of the gamma distribution that represents the increment in the degradation level of a production unit in one production cycle when $y = 0$ (i.e., all production units in the k-out-of-n system are functioning). It is easy to show that, in order for Assumptions 1 and 2 be simultaneously satisfied, the shape parameter w and the scale parameter θ of the gamma distribution have to vary as a function of the umber of failed production units y according to Eqs. (7) and (8), respectively. Remember that $L_D(0) = 1$.

$$w_y = w_0 \cdot [L_D(y)]^2 \tag{7}$$
$$\theta_y = \frac{\theta_0}{L_D(y)} \tag{8}$$

6. PROPOSED METHOD

As mentioned earlier, the goal is to find the maintenance scope S to be carried out now which minimizes the expected total cost per cycle until the next maintenance intervention, $TC(S)$. The next maintenance intervention is assumed to be performed when the probability of the whole production system failing reaches the value $1 - SL$. The expected total cost per cycle $TC(S)$ can be obtained using Eq. (9).

$$TC(S) = \frac{M(S) + \sum_{j=1}^{L(S)} P_j(S)}{L(S)} \tag{9}$$

where $L(S)$ is the expected number of cycles that the production system will operate until it reaches the safety level SL if a maintenance activity with scope S is carried out, $M(S)$ is the total maintenance cost associated with the maintenance activity with scope S and $P_j(S)$ is the expected production cost in the j-th cycle if a maintenance activity with scope S is carried out now.

6.1. Computation of $M(S)$

The total maintenance cost associate with a maintenance activity with scope S, $M(S)$, can be calculated using Eq. (10).

$$M(S) = C_F + \sum_{i=1}^{N_P} s_i \cdot \left[\delta_i \cdot C_C^{(i)} + (1 - \delta_i) \cdot C_P^{(i)} \right] \tag{10}$$

where δ_i is a binary variable that assumes value 1 if production unit i is failed and zero otherwise.

6.2. Computation of $L(S)$

In order to compute the expected number of cycles that the production system will operate until it reaches the safety level

Figure 2. Degradation evolution of each production unit.

if a maintenance activity with scope S is carried out, $L(S)$, the first step is to update the degradation level of maintained production units. The assumption that maintenance activities always bring the production unit to an "as good as new" condition is used, and the degradation level of maintained production units are set to zero. The degradation level of the remaining production units are not affected.

Then, the next step is to compute the RUL distribution for each production unit. Let $z_i(0)$ be the current degradation level of production unit i. Let $p_i(j)$ be the probability that production unit i will fail at the end of the j-th cycle from now. Once the current degradation level of production unit i and the distribution of the increment in the degradation level in each cycle are known, the RUL distribution of production unit i is given by Eq. (11) (Rodrigues, Medeiros, & Kern, 2015).

$$p_i(j+1) = p(F(x|w,\theta) \geq FT - z_i(j)|z_i(j) < FT) \quad (11)$$

where $F(x|w,\theta)$ is the CDF (cumulative distribution function) of the gamma distribution, $p_i(j)$ is the probability that a failure in production unit i will be detected at the end of the j-th cycle from now, $z_i(j)$ is the expected degradation level of production unit i at the end of the j-th cycle from now and FT is the failure threshold.

If a production unit i is failed, then $p_i(0) = 1$ and $p_i(j) = 0$ $\forall j > 0$. If a production unit i is functioning, then $p_i(0) = 0$ and $p_i(j)$ $\forall j > 0$ can be recursively computed using Eq. (12).

$$p_i(j+1) = \left[1 - \sum_{v=0}^{j} p_i(v) \right] \cdot \Omega \quad (12)$$

where

$$\Omega = p\left[F(x|j \cdot w_y, \theta_y) \geq FT - z_i(j) \right] \quad (13)$$

where y is the number of failed production units in the k-out-of-n system containing production unit i. The parameters w_y and θ_y are computed according to Eqs. (7) and (8), respectively.

The next step is to compute the RUL distribution for each k-out-of-n system. Let $p_k(j)$ be the probability that k-out-of-n system k, with $k = \{1, \ldots, N_S\}$, will fail at the end of the j-th cycle from now. Let v be a combination of n-k+1 production units of k-out-of-n system k and V be the set of all possible combinations v. The k-out-of-n system k will fail when all production units belonging to a combination $v \in V$ fail. Once the failure probability $p_i(j)$ of all production units in the k-out-of-n system k are known, the failure probability

of the k-out-of-n system k can be obtained using Eq. (14).

$$p_k(j) = \sum_{v \in V} \left[\prod_{i \in v} p_i(j) \right] \quad (14)$$

Now, it is necessary to calculate the RUL distribution for the production system as a whole. Let $p_s(j)$ be the probability that the production system will fail at the end of the j-th cycle from now. Once the failure probability of each k-out-of-n system k is known, the RUL distribution of the production system can be obtained using Eq. (15).

$$p_s(j) = 1 - \prod_{k=1}^{N_S} \left[1 - p_k(j) \right] \quad (15)$$

Finally, the expected number of cycles that the production system will operate until it reaches the safety level SL if a maintenance activity with scope S is carried out can be obtained according to Eq. (16).

$$L(S) = \min j | \left[p_s(j) > 1 - SL \right] - 1 \quad (16)$$

6.3. Computation of $P_j(S)$

In the proposed method, it is assumed that the failure of a production unit in a k-out-of-n system increases the degradation rate of the surviving units. It is also assumed that the overload of the surviving production units results in an increase of the production cost of the corresponding k-out-of-n system.

Let $L_C(y)$ be the load-sharing cost factor, where y is the number of failed production units in the k-out-of-n system, with $y = \{0, \ldots, n - k\}$. The expression to compute $L_D(y)$ is presented in Eq. (17).

$$L_C(y) = \left[\frac{n}{n - y} \right]^{\beta_C} \quad (17)$$

where β_C is the load-sharing cost exponent that models the additional production cost of the k-out-of-n system, with $0 \leq \beta_C \leq 1$. When $y = 0$, $L_C(0) = 1$ and the system operates with nominal cost.

In order to compute the expected production cost for the whole system in the j-th cycle if a maintenance activity with scope S is carried out, $P_j(S)$, it is necessary to compute the expected production cost for each k-out-of-n system k in each cycle j, $j = \{1, \ldots, L(S)\}$.

Let $\mathbf{p} = \{p_j^{k,0}(S), \ldots, p_j^{k,n-k}(S)\}$ be a probability vector, where each element $p_j^{k,y}(S)$ is the probability that y production units of k-out-of-n system k will fail at the end of the j-th cycle, assuming that a maintenance activity with scope S is

carried out. The steps to calculate vector **p** are as follows.

Set **p** $= 1$ (initialization). Then, for each production unit i in k-out-of-n system k, use the probability that production unit i will fail at the end of the j-th cycle, $p_i(j)$, to update vector **p** according to Eq. (18).

$$\mathbf{p} = \left[\begin{array}{c} \mathbf{p} \\ 0 \end{array} \right] \cdot (1 - p_i(j)) + \left[\begin{array}{c} 0 \\ \mathbf{p} \end{array} \right] \cdot p_i(j) \qquad (18)$$

After using all probabilities $p_i(j)$ to update vector **p**, discard all elements $p_j^{k,y}(S)$ with $y > n$-k from vector **p**. Then, normalize vector **p** such that the sum of all its elements equals one. Then, the expected production cost of k-out-of-n system k in the j-th cycle, $P_j^k(S)$, can be obtained using Eq. (19).

$$P_j^k(S) = \sum_{y=0}^{n-k} p_j^{k,y}(S) \cdot L_C(y) \cdot P_k \qquad (19)$$

where P_k is the nominal production cost of k-out-of-n system k (production cost incurred when $y=0$).

Finally, the expected production cost for the entire production system in the j-th cycle if a maintenance activity with scope S is carried out now, is obtained using Eq. (20).

$$P_j(S) = \sum_{k=1}^{N_S} P_j^k(S) \qquad (20)$$

7. ANT COLONY OPTIMIZATION

The ACO algorithm will be used to find good maintenance scopes in order to minimize the expected total cost per cycle $TC(S)$.

The ACO algorithm was chosen because it suits the structure of the problem under consideration, which can easily be represented in the form of a weighted graph (Cordon, Herrera, & Stützle, 2002). The nodes of the graph represent production units and the edges of the graph represent the decision of selecting a production unit to be maintained or not.

Although many papers have been published showing combinatorial problem applications in which the ACO outperforms other optimization algorithms such as Simulated Annealing (Chmait & Challita, 2013), Bacterial Foraging Optimization Algorithm (Afac & Saini, 2011) and Genetic Algorithm (Adbuljabbar, Khalefa, & Jabar, 2013), it should be noted that other metaheuristics could be equally employed, with no conceptual alterations in the proposed method.

7.1. Algorithm Description

The basic principle of the ACO algorithm is to have a population of m artificial ants that find possible solutions to a com-

binatorial problem. In the ant system, the problem is represented by a graph, in which ants walk through branches from the starting node to the final node and build paths that represent possible solutions to the problem. Ants choose the next node to be visited based on a state transition rule (Colorni, Dorigo, & Maniezzo, 1991).

In the proposed framework, possible solutions are codified as binary vectors of length N_C, where N_C is the number of production units in the whole production system. Each element s_i of the maintenance scope vector S can be filled with "1" when production unit i is maintained, or "0" when the production unit i is not maintained. The solutions are evaluated according to their expected total cost per cycle.

7.2. Pheromone Update Rule

At the end of each iteration of the ACO algorithm, when all ants have walked through the branches, the pheromone update rule is applied. This rule consists in removing a fraction of pheromone from all edges (emulating natural evaporation) and then adding an increment of pheromone to those edges visited by the ant that has found the best-so-far solution. In the next iteration of the algorithm, those edges belonging to the best-so-far solution will have higher probability to be visited by other ants. The increment of pheromone can be fixed for all branches belonging to the best-so-far solution or can be proportional to the heuristic distance of the branch. A fixed increment is used in this paper.

In order to reproduce the behavior of real ants more closely, an increment of pheromone should be given to all edges visited by any ant in the current iteration, directly proportional to the number of ants walking by the edges and inversely proportional to the edge heuristic distance. However, simulations have shown that an increment in only the edges belonging to the best-so-far solution results in a better performance of the algorithm (van der Zwaan & Marques, 1999). Therefore, the best-so-far updating strategy will be used in this paper.

7.3. Parameters Tuning

In order to implement the ACO algorithm, a set of parameters must be defined. The ACO algorithm parameters are the number of ants in the colony, m, and the pheromone evaporation rate, ρ. The choice of parameters is problem-specific and depends on the required accuracy of the solution (Gaertner & Clark, 2005). In the absence of a mathematical model to determine the optimal values for the algorithm parameters, they have to be determined empirically (van der Zwaan & Marques, 1999).

7.4. Stop Criterion

At the end of each iteration of the ACO algorithm, the stop criterion must be checked. Commonly used stop criteria de-

pend not only on the total number of iterations, but also on the number of consecutive iterations without any improvement in the best solution.

The stop criterion adopted in this paper is based on the probabilities associated to the branches belonging to the best solution. In order to illustrate this criterion, let us consider an example with four production units. Assume that, at the end of an iteration, the probability associated to each branch P_i is as indicated in Table 1, where $P_i^{(0)}$ is the probability that an ant will generate a solution in which production unit i is not maintained. Similarly, $P_i^{(1)}$ is the probability that an ant will generate a solution in which production unit i is maintained. Note that these probabilities are associated to the amount of pheromone deposited in each branch. Also, let the best solution found so far be to maintain production units 1 and 4. In this situation, the path that leads to the best solution is $P_1^{(1)}$, $P_2^{(0)}$, $P_3^{(0)}$ and $P_4^{(1)}$ (Rodrigues, Gomes, et al., 2015).

Table 1. Stop criterion example.

Prod. Unit i	1	2	3	4
$P_i^{(0)}$	40%	**55%**	**75%**	30%
$P_i^{(1)}$	**60%**	45%	25%	**70%**

In this example, the mean probability of the path corresponding to best solution is 65%. This value is expected to increase as the number of iterations increases. The stop criterion is met when the mean of the probabilities belonging to the best solution is higher than a stop coefficient λ.

8. NUMERICAL EXAMPLE

This section presents a numerical example to illustrate the definition of the maintenance scope in order to minimize the expected total cost per cycle. In this example, a production system comprising four k-out-of-n systems connected in series is considered.

The k-out-of-n structure can be found in many production systems. The production system considered in this example could represent, for instance, an electrical power system. Each k-out-of-n system could represent one stage of the electrical power system (generation, transmission, conversion, distribution, etc.). Each production unit in a k-out-of-n system could represent one device in the respective stage (generating units, transmission lines, transformers in a substation, distribution lines, etc.). The minimum number of functioning production units k in each k-out-of-n system could represent the minimum number of production units needed to meet the demand.

It is assumed that all production units in each k-out-of-n system are identical. The increment in the degradation level z_i of each production unit i during each cycle is a random variable

that follows a gamma distribution. Table 2 shows the configuration (k, n), the nominal parameters of the gamma distribution w_0 and θ_0, the nominal production cost P_k and the preventive maintenance cost C_P of each k-out-of-n system. Figure 3 shows the block diagram of the production system under consideration.

A fixed maintenance cost C_F of 40 is used. The corrective maintenance cost C_C is assumed to be 2.5 times more expensive than the corresponding preventive maintenance cost C_P. The failure threshold level FT and the safety level SL are 100 and 0.95, respectively. Also, the degradation load-sharing exponent β_D and the cost load-sharing exponent β_C are assumed to be 1.0 and 0.5, respectively.

Table 3 shows the current degradation level of each production unit. It can be seen that three production units (B, D and J) are failed. However, the production system as a whole is still functioning since each k-out-of-n system has at least the required k functioning production units. System status is illustrated in Fig. 4.

8.1. Current Expected Total Cost per Cycle

The first step is to calculate the expected total cost per cycle assuming that no maintenance activity will be carried out, which will be denoted by scope S_0. By definition, $M(S_0) = 0$. To calculate $TC(S_0)$, Eq. (9) is reduced to the following expression:

Table 2. System Data.

	System 1	System 2	System 3	System 4
k	3	1	2	2
n	6	2	4	3
w_0	3.5	3.0	2.5	2.5
θ_0	2.5	2.0	2.0	1.5
C_P	30	40	50	60
P_k	80	120	150	50
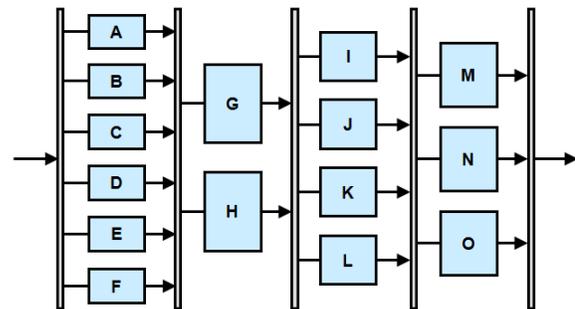
Figure 3. Production system block diagram.

Table 3. Current degradation level.

Production Unit	Current Degradation	Production Unit	Current Degradation
A	9	I	15
B	100	J	100
C	23	K	33
D	100	L	51
E	41	M	13
F	56	N	22
G	25	O	37
H	55		

$$TC(S_0) = \frac{\sum_{j=1}^{L(S_0)} P_j(S_0)}{L(S_0)} \qquad (21)$$

Table 4 shows the load-sharing degradation factor L_D and the load-sharing cost factor L_C for this example, based on the number of failed production units y in each k-out-of-n system.

Table 4. Load-sharing factors.

System	n	y	L_D	L_C
1	6	2	1.50	1.22
2	2	0	1.00	1.00
3	4	1	1.33	1.15
4	3	0	1.00	1.00

Using Eqs. (12) to (16), it is possible to conclude that, if no maintenance activity is carried out, the system will operate for more 3 cycles before reaching the safety level SL. Thus, $L(S_0) = 3$. The expected production cost $P_j(S_0)$ for cycles $j = [1, \ldots, L(S_0)]$ are obtained as described in Section 6.3. The values are shown in Table 5. Finally, using Eq. (21), it is possible to obtain $TC(S_0) = 442.7$.
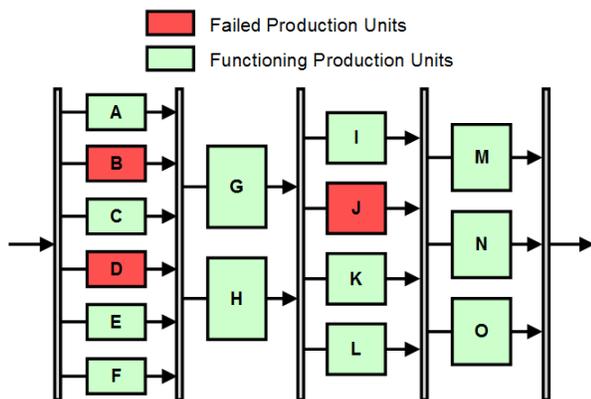
Figure 4. System current status.

Table 5. Production cost if no maintenance is carried out.

Cycle j	$P_j(S_0)$
1	441.2
2	441.3
3	445.5

8.2. Reducing the Expected Total Cost per Cycle

Now, the objective is to find a maintenance scope S that will reduce the expected average total maintenance cost. The ACO algorithm is used to find the maintenance scope S that minimizes the expected total cost per cycle $TC(S)$.

A colony with 20 ants ($m = 20$), a pheromone evaporation rate ρ of 10% and a stop coefficient λ of 0.9 were used. These values were obtained after a series of simulations. As mentioned earlier, these parameters must be defined empirically since a mathematical model to determine the optimal values for them does not exist. A total of 50 simulations were performed using the ACO algorithm with the chosen parameters. The results are presented in Table 6.

Table 6. ACO algorithm result.

Maintained Prod. Units	$TC(S)$	Occurrences
{B, D, F, H, J}	431.55	30
{B, D, J}	431.69	17
{B, D, E, F, H, J}	432.47	2
{D, J}	436.04	1

In this example, the maintenance scope with the highest number of occurrences (maintain components B, D, F, H and J) is the optimal solution for this specific problem. It was validated by running an exhaustive search. The second ranked maintenance scope (maintain components B, D and J) is the solution in which only the failed production units are maintained. The fourth ranked maintenance scope (maintain components D and J) is the only solution found during the simulations that does not maintain all failed production units (production unit B is not maintained in this scope).

In the simulations, the average number of iterations until the stop criterion is reached was 32. In each iteration, one solution (maintenance scope) for each of the 20 ants is evaluated. So, the average number of maintenance scopes evaluated in each run of the ACO algorithm is $32 \times 20 = 640$. The total number of possible maintenance scopes for a production system with 15 production units is $2^{15} = 32,768$ (including scope S_0). It means that the ACO algorithm evaluated only 1.95% of the search space to find close to optimal solutions (the optimal solution was found in 60% of the times). Figure 5 shows the expected total cost per cycle for the optimal solution. For a comparison purpose, the expected total cost per cycle for maintenance scope S_0 is also shown in Fig 5.
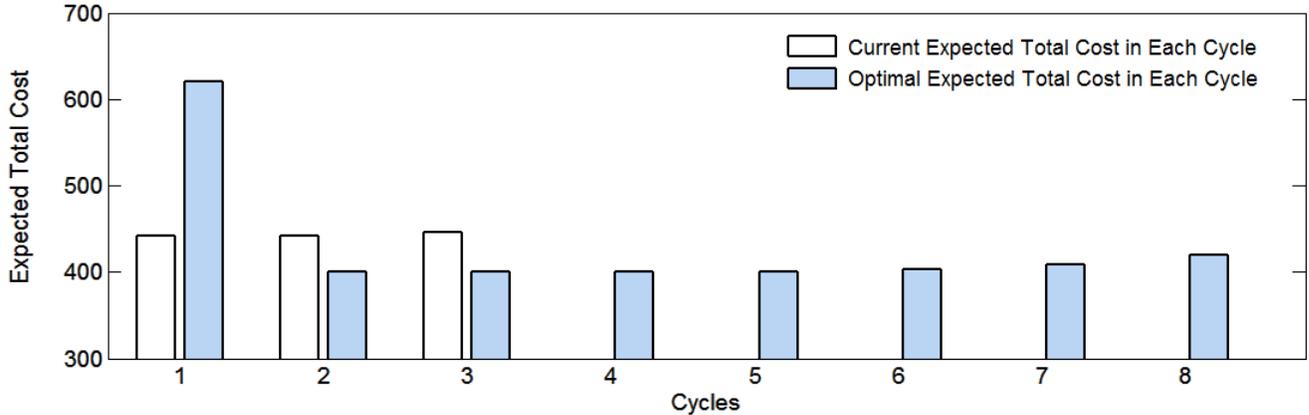
Figure 5. Comparison of the expected total cost per cycle.

As mentioned earlier, if no maintenance intervention is carried out, the production system is expected to reach the safety level SL in three cycles, with an average total cost per cycle of 442.7. In the optimal solution, the production system is expected to reach the safety level SL in eight cycles, with an average total cost per cycle of 431.6.

In the optimal solution, the expected cost in cycle 1 is higher because the maintenance cost is incurred. However, for the next cycles, the expected cost becomes lower and compensates the amount of money spent in order to perform the maintenance intervention.

9. DISCUSSION

In this paper, the use of RUL predictions in a framework to define maintenance scopes in order to minimize the expected total maintenance cost was investigated. The numerical experiments showed that the maintenance scope definition process can be performed more efficiently when PHM information is taken into account.

The implementation of the proposed method in a manufacturing facility shall consider that maintenance interventions must be carried out with minimum impact in production activities. Maintenance interventions can be scheduled, for instance, to be carried out overnight or at weekends. In this scenario, the proposed optimization routine would be run at the end of each work day, which would be considered as a production cycle. It is worth pointing that it only makes sense to run the optimization model if new information on the current degradation level of at least one production unit is available.

In order to compute the probabilities $p_i(j)$ in Eq. (11), it was considered that no failures will occur between the current cycle and the next maintenance intervention. It may lead to an optimistic estimation of $L(S)$ since the occurrence of a failure would increase the degradation rate of some production units. However, the decision on performing a maintenance intervention is made at the end of each cycle, and it is assumed

that production unit failures are self-announcing. Then, when a production unit fails, the degradation rates of the surviving production units belonging to the same k-out-of-n system are updated at the end of the cycle.

In this paper, the uncertainty in the estimation of the current degradation level z of each production unit is not considered. Further improvements of the proposed method may include considering the uncertainties in the current degradation level of each production unit, as well as the uncertainties in the parameters of the gamma distribution of each production unit.

The mathematical formulation considers that the production cost of a production unit is affected only by the number of failed production units in the k-out-of-n system. Another opportunity for improvements in the proposed method is to consider other factors that may affect production cost such as raw material quality, human factors and production workload.

10. CONCLUSIONS

A method to define the optimal maintenance scope for a production system consisting of multiple k-out-of-n load-sharing systems connected in series is presented. In the proposed method, RUL predictions obtained from a PHM system are used to recommend maintenance scopes according to a cost criterion. Total cost is broken into production cost and maintenance cost.

The ACO algorithm was applied in order to provide the capability of solving large instances of the problem. The ACO algorithm has been successfully used in many combinatorial problems, and the results observed in the proposed method were very satisfactory. Close to optimal solutions were obtained by evaluating a very small fraction of the search space. In the numerical example presented to illustrate the application of the proposed method, although less than 2% of the possible solutions were evaluated, the proposed method found the optimal solution in most simulations.

Planning and scheduling maintenance activities is a crucial task in manufacturing systems. In the competitive manufacturing sector, high availability and low production cost are required for production units. As the production system complexity increases, managing the maintenance processes such as planning, scheduling and scope definition becomes a very challenging task.

PHM systems can be a powerful tool to enable the implementation of smart manufacturing systems. By estimating the degradation level of production units, PHM systems provide valuable information to allow the maintenance processes to be executed in an automated framework.

Future research may extend the scope of this paper by investigating the application of different metaheuristics for this problem. The application of hybrid methods combining features of different metaheuristics could also be investigated. Additionally, an investigation on the benefits provided by incorporating PHM information in CBM methods which consider different combinations of the most commonly studied assumptions (economic, functional and stochastic dependency, load-sharing models, system-level health condition prediction, discrete or continuous health monitoring, etc.) could also be conducted.

REFERENCES

Adbuljabbar, Z. A., Khalefa, M. S., & Jabar, M. A. (2013). Comparison between ant colony and genetic algorithm using traveling salesman problem. *International Journal of Soft Computing*, 8(3).

Afac, H., & Saini, S. (2011). On the solutions to the travelling salesman problem using nature inspired computing techniques. *International Journal of Computer Science Issues*, 8(2), 326–334.

Ahmad, R., & Kamaruddin, S. (2012). A review of condition-based maintenance decision-making. *European Journal of Industrial Engineering*, 6(5), 519–541.

Amari, S. V., & Bergman, R. (2008, January). Reliability analysis of *k*-out-of-*n* load-sharing systems. In *Proceedings of the 2008 annual reliability and maintainability symposium (RAMS)* (pp. 440–445). Las Vegas.

Barajas, L. G., & Srinivasa, N. (2008, October). Real-time diagnostics, prognostics and health management for large-scale manufacturing maintenance systems. In *Proceedings of the ASME 2008 international manufacturing science and engineering conference* (pp. 85–94). Evanston.

Bernarden, J. (2012). *Indirect jobs: A direct way to talk about why we need smart manufacturing* (Tech. Rep.). Rockwell Automation.

Brodsky, A., Krishnamoorthy, M., Menascé, D. A., Shao, G., & Rachuri, S. (2014, June). Toward smart manufacturing using decision analytics. In *Proceedings of the 2014 ieee international conference on big data* (pp. 967–977). Anchorage.

Camci, F. (2009). System maintenance scheduling with prognostics information using genetic algorithm. *IEEE Transactions on Reliability*, 58(3), 539–552.

Chmait, N., & Challita, K. (2013). Using simulated annealing and ant-colony optimization algorithms to solve the scheduling problem. *Computer Science and Information Technology*, 1(3).

Colorni, A., Dorigo, M., & Maniezzo, V. (1991, December). Distributed optimization by ant colonies. In *Proceedings of the european conference on artificial life* (pp. 134–142). Paris.

Cordon, O., Herrera, F., & Stützle, T. (2002). A review on the ant colony optimization metaheuristic: Basis, models and new trends. *Mathware & Soft Computing*, 9, 141–175.

Gaertner, D., & Clark, K. (2005, July). On optimal parameters for ant colony optimization algorithms. In *Proceedings of the international conference of artificial intelligence* (pp. 83–89). Pittsburgh.

Gebraeel, N. (2010). Prognostics-based identification of the top k units in a fleet. *IEEE Transactions on Automation and Science Engineering*, 7, 37–48.

Ghonaim, W., Ghenniwa, H., & Shen, W. (2011, June). Towards an agent oriented smart manufacturing system. In *Proceedings of the 2011 15th international conference on computer supported cooperative work in design* (pp. 636–642). Lausanne.

Heddy, G., Huzaifa, U., Beling, P., Haimes, Y., Marvel, J., Weiss, B., & LaViers, A. (2015, October). Linear temporal logic (LTL) based monitoring of smart manufacturing systems. In *Proceedings of the 2015 annual conference of the prognostics and health management society* (pp. 640–649). San Diego.

Huynh, K. T., Barros, A., & Bérenguer, C. (2015). Multi-level decision-making for the predictive maintenance of *k*-out-of-*n*:F deteriorating systems. *IEEE Transactions on Reliability*, 64(1), 94–117.

Jung, K., Morris, K. C., Lyons, K. W., Leong, S., & Cho, H. (2015). Mapping strategic goals and operational performance metrics for smart manufacturing systems. *Procedia Computer Science*, 44, 184–193.

Kacprzynski, G. J., Roemer, M. J., & Hess, A. J. (2002, March). Health management system design: Development, simulation and cost/benefit optimization. In *Proceedings of the 2002 IEEE aerospace conference* (pp. 3065–3072). Big Sky.

Khatab, A., Nahas, N., & Nourelfath, M. (2009). Availability of *k*-out-of-*n*:G systems with non-identical components subject to repair priorities. *Reliability Engineering and System Safety*, 94, 142–151.

Kuo, W., & Zuo, M. J. (2003). *Optimal reliability modeling - principles and applications* (1st ed.). New Jersey:

Wiley.

Lu, L., & Lewis, G. (2008). Configuration determination for *k*-out-of-*n* partially redundant systems. *Reliability Engineering and System Safety*, *93*(11), 1594–1604.

Lu, Y., Morris, K. C., & Frechette, S. (2015, August). Standards landscape and directions for smart manufacturing systems. In *Proceedings of the 2011 15th international conference on computer supported cooperative work in design* (pp. 998–1005). Gothenburg.

Malinowski, M. L., Beling, P. A., Haimes, Y., LaViers, A. E., Marvel, J. A., & Weiss, B. A. (2015, October). System interdependency modeling in the design of prognostic and health management systems in smart manufacturing. In *Proceedings of the 2015 annual conference of the prognostics and health management society* (pp. 210–222). San Diego.

Medeiros, I. P., Rodrigues, L. R., Shiguemori, E. H., Santos, R., & Nascimento Jr, C. L. (2014, April). PHM-based multi-UAV task assignment. In *Proceedings of the 2014 IEEE systems conference* (pp. 42–49). Ottawa.

Moghaddass, R., Zuo, M. J., & Wang, W. (2011). Availability of a general *k*-out-of-*n*:G system with non-identical components considering shut-off rules using quasi-birthdeath process. *Reliability Engineering and System Safety*, *96*, 489–496.

Mohammad, R., Kalam, A., & Amari, S. V. (2013, January). Reliability of load-sharing systems subject to proportional hazards model. In *Proceedings of the 2013 annual reliability and maintainability symposium (RAMS)* (pp. 1–5). Orlando.

Pozsgai, P., Neher, W., & Bertsche, B. (2003, January). Models to consider load-sharing in reliability calculation and simulation of systems consisting of mechanical components. In *Proceedings of the 2003 annual reliability and maintainability symposium* (pp. 493–499). Tampa.

Rodrigues, L. R., Gomes, J. P. P., Ferri, F. A. S., Medeiros, I. P., Galvao, R. K. H., & Nascimento Jr, C. L. (2015). Use of PHM information and system architecture for optimized aircraft maintenance planning. *IEEE Systems Journal*, *9*(4), 1197–1207.

Rodrigues, L. R., Medeiros, I. P., & Kern, C. S. (2015, April). Maintenance cost optimization for multiple components using a condition based method. In *Proceedings of the 2015 IEEE Systems Conference* (pp. 164–169). Vancouver.

Rodrigues, L. R., & Yoneyama, T. (2012, September). Spare parts inventory control for non-repairable items based on prognostics and health monitoring information. In *Proceedings of the 2012 annual conference of the prognostics and health management society* (pp. 53–62). Minneapolis: PHM society.

Sandborn, P. A., & Wilkinson, C. A. (2007). A mainte-nance planning and business case development model for the application of prognostics and health management (PHM) to electronic systems. *Microelectronics Reliability*, *47*, 1889–1901.

Vachtsevanos, G., Lewis, F. L., Roemer, M., Hess, A., & Wu, B. (2006). *Intelligent fault diagnosis and prognosis for engineering systems* (1st ed.). New Jersey: John Wiley & sons.

van der Zwaan, S., & Marques, C. (1999). Ant colony optimization for job shop scheduling. In *Proceedings of third workshop on genetic algorithms and artificial life.*

van Noortwijk, J. M. (2009). A survey of the application of gamma processes in maintenance. *Reliability Engineering and System Safety*, *94*, 2–21.

Weiss, B. A., Vogl, G., Helu, M., Qiao, G., Pellegrino, J., Justiniano, M., & Raghunathan, A. (2015, October). Measurement science for prognostics and health management for smart manufacturing systems: Key findings from a roadmapping workshop. In *Proceedings of the 2015 annual conference of the prognostics and health management society* (pp. 232–333). San Diego.

Yinghui, T., & Jing, Z. (2008). New model for load-sharing *k*-out-of-*n*:G system with different components. *Journal of Systems Engineering and Electronics*, *19*(4), 748–751.

Zhou, R., Lee, H. P., & Nee, A. Y. C. (2008). Applying ant colony optimization (ACO) algorithm to dynamic job shop scheduling problems. *International Journal of Manufacturing Research*, *3*(3), 301–320.

Zhou, Y., Zhang, Z., Lín, T. R., & Ma, L. (2013). Maintenance optimisation of a multi-state series-parallel system considering dependence and state-dependence inspection intervals. *Reliability Engineering and System Safety*, *11*, 248–259.

BIOGRAPHY

Leonardo Ramos Rodrigues holds a bachelor's degree on Electrical Engineering from Universidade Federal do Espírito Santo (UFES, 2003) Brazil, a Master's Degree (2008) and a D.Sc. Degree (2013) on Aeronautical Engineering from Instituto Tecnológico de Aeronáutica (ITA), Brazil. He is currently a technologist with Instituto de Aeronáutica e Espaço (IAE) in São José dos Campos - Brazil, where he performs research in the field of aerospace technologies. He was formerly with Embraer S.A. in São José dos Campos - Brazil from 2006 to 2015, working with the PHM research group performing research in the fields of Prognostics and Health Management and Data Analytics solutions for the aviation industry.