A constructive algorithm to maximize the useful life of a mechanical system subjected to ageing, with non-resuppliable spares parts

Francesco Zammori[a*], Massimo Bertolini[a] and Davide Mezzogori[a]

[a]Department of Engineering and Architecture, University of Parma, Parma, Italy

CHRONICLE	ABSTRACT
	In this paper, the focus is on mechanical systems that, like a ship or a submarine, perform risky missions and that must remain operating for the whole mission time. Missions take place far from the operational base and so, in case of failures, although repairs are possible, spares parts cannot be resupplied. Hence, given space constraints, the problem is to define the optimal set of spare parts that should be taken aboard, to maximize the probability to complete the mission. To solve this problem, we propose a constructive algorithm that generates the Pareto Optimal Frontier of all the non-dominated solutions, in terms of the system's reliability and of required space. At first, the algorithm is formulated in a generic way; next, it is contextualized to the common case of Weibull distributed failure times. In this condition, the underlying equations of the model cannot be solved in closed form and an approximated procedure is proposed and validated through extensive numerical simulation. © 2020 by the authors; licensee Growing Science, Canada

1. Introduction

Nowadays, operations' efficiency and costs minimization are imperative issues of survival. In this regard, machines' failures and productions halts represent serious damages for a manufacturing company, both in terms of revenues and of corporate image losses (De Smidt-Destombes et al., 2009; Basten & Ryan, 2019). To counteract these risks, an optimal mix of maintenance strategies is not enough, unless an adequate policy for spare parts management has also been defined (Van Horenbeek et al., 2013; Cai et al., 2017). Indeed, in case of missing spare parts, reparation times amplify and costs grow up, due to production losses and/or to the issuing of urgent orders (Van Jaarsveld & Dekker, 2011; Godoy et al., 2013). However, spare parts are expensive and bulky items, that can even become obsolete, due to the replacement of the machines for which they were originally designed. So, holding high inventories of spare parts is not a viable solution, as it would inevitably lead to unsustainable holding costs. A trade-off, between the system's reliability and costs, must be found, and the optimization of this issue has attracted much scientific interest, as testified by the interesting literature reviews by Kennedy et al. (2002), Ahmed and Sultana (2013), Khanlarzade et al. (2014) and Van der Auweraer et al. (2019).

* Corresponding author Tel.: +39 0521 905887
E-mail: francesco.zammori@unipr.it (F. Zammori)

What emerges from the literature is the fact that standard models for inventory management cannot be applied to spare parts, as they differentiate from raw materials and finished products in many relevant aspects. Briefly:

- spares parts are characterized by intermittent, irregular or lumpy demand, that is very hard to be forecasted (Willemain et al., 2004; Vaughan, 2005; Costantino et al., 2018);
- demand is influenced or even dictated by the adopted mix of maintenance policies (Godoy et. al., 2013; Yang & Kang, 2017) and it could also be met through cannibalization of other parts or units (Rajpal et al., 2006; Sheng & Prescott, 2017; Renna, 2017);
- inventory policies should be dynamically modified, depending on the life cycle stage of the equipment where spare parts are installed (Teunter & Fortuin, 1999, Teunter & Hanevled, 2002).

For instance, when a new mechanical system is installed, reliability data are not available and a major problem concerns the choice of the initial set of spare parts that should be purchased (Do Rego & De Mesquita, 2011). Later on, as reliability data are collected, inventory policies should be redefined, aiming to maximize the overall system's availability (Vand der Auweraer & Boute, 2019).

Despite the relevant number of papers dealing with spare parts management, the case of repairable mechanical systems that must operate for a long time, but that cannot receive supplies of spare parts, has been little studied. Classic examples may be that of a ship, a submarine or a space base (in Earth orbit), that must continuously operate for a long mission time, far away from the operating base and/or in inaccessible places. Certainly, in case of failures, the crew can repair the damage, but only the onboard equipment can be used since additional spare parts cannot be resupplied for the whole mission time (or it would be too costly to do so). Consequently, before a mission starts, there is the need to define the type and number of spare parts that should be taken aboard, to maximize the probability complete the mission in a safe way, without violating the space constraints imposed by the narrow layout of the system.

The above-mentioned problem could be seen as a niche case, yet a similar issue also arises in the end-life management of standard mechanical equipment. Indeed, when a machine is no longer produced and is withdrawn from the market, although the Original Equipment Manufacturer (OEM) must assure spare parts availability (at least for a minimum period imposed by law), their selling price generally rises up a lot. So, when an OEM retires a machine from the market, the buyer should make an opportunistic purchase of spare parts, to anticipate the possible price increase. Obviously, a wise decision, concerning the set of spare parts that should be purchased, should be based both on the available budget and on the residual technological life (i.e., the desired time before disposal) of the old, but still functioning machine. As can be seen, this problem is very similar to the one previously discussed, with the only difference that the available space is replaced by a budget constraint.

Notwithstanding the practical relevance of the above-mentioned problem, as long as the knowledge of the authors, only two optimization methods have been proposed so far. The first one was proposed by Denicoff et al. (1964), who used a generalization of the News-Vendor-Problem to determine the optimal set of spare parts for a Polaris submarine. Later on, Sherbrooke (1968, 2004), faced the same problem and introduced a constructive algorithm to generates the Pareto Optimal Frontier of the non-dominated spare parts inventory configurations, in terms of space and system's reliability. The algorithm can be optimally solved in polynomial time, but unfortunately, its application is restricted to the hypothesis of constant failure rates. This is a major limitation that holds only for electronic devices, but not for mechanical parts subjected to friction and wearing.

The present paper focuses on this topic and generalizes the constructive algorithm by Sherbrooke, making it independent from the probability distribution of the failure rates. Next, the common case of Weibull distributed failure times is considered in more details. In this peculiar condition, the underlying equations of the model cannot be solved in closed form, and so an approximated approach is introduced and numerically validated via Monte Carlo simulation.

The remainder of the paper is organized as follows. Section 2 details the basic constructive algorithm proposed by Sherbrooke, whereas Sections 3 and 4 introduce its generalization and its application to the

specific Weibull case. The quality and the robustness of the algorithm are numerically assessed in Section 5 while concluding remarks and directions for future researches are discussed in Section 6.

2. The basic constructive algorithm

In this section, we will briefly explain the original algorithm proposed by Sherbrook (2004), which will be modified in a novel way in Section 3 and Section 4.

Let us consider a mechanical and reparable system that must operate, continuously, for a period - or mission time - equal or greater than T. In terms of maintenance requirements, the system can be modelled as a set of critical parts operating in series, so that, for the whole system to be operational, all parts must be simultaneously functioning. We also suppose that, in case of failure, the faulty part can be immediately replaced, provided that a spare part is available in the warehouse. However, in the case of stock-out, new spare parts cannot be resupplied. So, the initial spare parts inventory must be sufficient to ensure, with a high confidence level, the survival of the system for the whole mission time T.

In this scenario, the key issue concerns the definition of number and type of spare parts, which should be purchased, at the minimum cost (or with the minimum required space), to maximize the reliability of the system in the mission time T. Without loss of generality, from here on, we will talk in terms of cost minimization, but everything also applies to the minimization of the required warehouse space.

To formulate and solve the problem, the following notation will be used:

- T - The mission time during which the system must operate, but spare parts cannot be resupplied.
- P - The number of different critical parts for which spare parts can be purchased.
- M - The number of locations (of the mechanical system) where critical parts are installed. Since the same critical part could be installed in more than one location, we have that $P \leq M$.
- t_{ij} - The required operating time of part i installed at location j (with $t_{ij} \leq T$).
- N_i - The number of spare parts of type i that are initially available in the warehouse.
- $\boldsymbol{N} = \{N_1, N_2, \dots, N_i \dots, N_P\}$ - A spare parts inventory configuration.
- $R_S(T|\boldsymbol{N}) = R_S(T|N_1, N_2, \dots, N_P)$ - The reliability of the system achieved using a specific spare parts configuration $\boldsymbol{N}$.
- $\delta_i(R_S)$ - The marginal increase of the system's reliability, which can be achieved adding one spare part i to the initial stock level N_i i.e., $\delta_i(R_S) = R_S(T|N_1, N_2, \dots, (N_i + 1), \dots, N_{2_P}) - R_S(T|N_1, N_2, \dots, N_i, \dots, N_{2_P})$.
- $P_i(n \leq N_i)$ - The probability to have, during the mission time $(0, T]$, a number of failures n lower or equal that N_i or, equivalently, the probability of not having a stock-out of spare part i.
- $\delta_i(P_i)$ – The marginal increase of the non-stock-out probability that can be achieved adding one unit of spare part i to the initial stock level N_i i.e., $\delta_i(P_i) = P_i(n \leq N_i + 1) - P_i(n \leq N_i + 1)$.

Also, a spare part configuration $\boldsymbol{N}$ is said to be a Non-Dominated-Solution (NDS) if and only if any other configuration of equal cost achieves a lower level of reliability.

Our goal is to define a step-by-step procedure that permits to generate a sequence of NDS that, at increasing cost levels, maximizes the reliability of the system, at the minimum value of the spare parts inventory.

Let us start by computing $R_S(T|\boldsymbol{N})$. Since all P parts are installed in series, we immediately have that:

$$R_S(T|\boldsymbol{N}) = R_S(T|N_1, N_2, \dots, N_P) = \prod_{i=1}^{P} R_i(T|N_i) \tag{1}$$

Where $R_i(T|N_i)$ can be seen as the share of the system's reliability, which is ascribable to part i, when N_i spare parts of type i are available in the warehouse.

Since spare part i can be substituted N_i times, $R_i(T|N_i)$ coincides with the probability $P_i(n \leq N_i)$ of not having stock out of spare part i. In the over-simplistic case of exponentially distributed lifetimes (i.e., constant failure late λ), since failures are independent, the fault of a generic spare part i coincides with a Poisson random variable, with parameter Λ_i equal to:

$$\Lambda_i = \sum_{j=1}^{M_i} (\lambda_{ij} \cdot t_{ij}) \quad \forall i = 1, 2, \dots, P \tag{2}$$

Where:
- λ_{ij} is the constant failure rate of spare part i installed at location j.
- $t_{ij} \leq T$ is the requested operating time of part i installed at location j.
- $M_i \leq M$ is the number of locations where part i is installed.

In this case, we have:

$$P_i(n = k) = P_i(k) = \frac{\left(\left(\sum_{j=1}^{M_i} (\lambda_{ij} \cdot t_{ij}) \right) \cdot e^{-\sum_{j=1}^{M_i} (\lambda_{ij} t_{ij})} \right)}{k!} = \frac{(\Lambda_i^k \cdot e^{-\Lambda_i})}{k!} \tag{3}$$

Thus, the system's reliability $R_S(T|\mathbf{N})$ can be easily computed as follows:

$$R_S(T|\mathbf{N}) = \prod_{i=1}^{P} R_i(T|N_i) = \prod_{i=1}^{P} \left(\sum_{k=0}^{N_i} \frac{(\Lambda_i^k \cdot e^{-\Lambda_i})}{k!} \right) \tag{4}$$

Lastly, by introducing natural logarithms, eq. (4) can be simplified as follows:

$$\ln(R_S(T|\mathbf{N})) = \ln \left(\prod_{i=1}^{P} R_i(T|N_i) \right) = \sum_{i=1}^{P} \ln(R_i(T|N_i)) \tag{5}$$

With respect to N_i, $R_i(T|N_i)$ is a monotonically increasing function and this property is maintained also by its logarithm $\ln(R_i(T|N_i))$. Also, from eq. (5) it is easy to see that the logarithm of the marginal increase of the system's reliability $\delta_i(R_S(T|\mathbf{N}))$ corresponds, exactly, to the logarithm of the marginal increase of the non-stock-out probability $\delta_i(P_i(n \leq N_i))$. Indeed:

$$\ln(\delta_i(R_S)) = \ln(R_S(T|N_1, \dots, (N_i + 1), \dots, N_P)) - \ln(R_S(T|N_1, \dots, N_i, \dots, N_P)) =$$
$$= \ln(R_i(T|N_i + 1)) - \ln(R_i(T|N_i)) = \ln(\delta_i(R_i)) = \ln(\delta_i(P_i)) \tag{6}$$

This is an important finding because, in terms of the system's reliability, the addition of an extra spare part has an "*addictive*" and "*separable*" effect. Owing to this fundamental property, spare parts can be considered independently and, starting from a null initial inventory (i.e., $N_i = 0 \; \forall \; i$), an optimal algorithm, hereafter referred to as the Spare parts Configuration Constructive Algorithm (SpCCA), can be formulated as follows:

1. Compute the algorithm of the marginal increase of reliability $\ln(\delta_i(R_i))$ for each spare part i. Since the algorithm starts from a null initial inventory (i.e., $N_i = 0 \; \forall \; i$), $\ln(\delta_i(R_i))$, at Step 1 this computation simplifies as in (7):

$$\ln(\delta_i(R_i)) = \ln\left(e^{-\Lambda_i} \cdot (\Lambda_i + 1) \right) - \ln(e^{-\Lambda_i}) = \ln(\Lambda_i + 1) \tag{7}$$

2. Select the spare part that maximizes the Reliability to Cost Ratio, computed as in (8):

$$\frac{\ln(\delta_i(R_S))}{C_i} = \frac{\ln(\delta_i(R_i))}{C_i} \tag{8}$$

where C_i is the purchase cost of spare part i.

3. Let i^* be the selected spare part, then $\boldsymbol{N} = \{N_1, N_2, \ldots, (N_{i^*} + 1) \ldots, N_P\}$ is the new NDS.
4. Update the inventory cost: $C = C_1 \cdot N_1 + \cdots + C_p \cdot N_p$
5. Update the system's reliability: $R_S(T|\boldsymbol{N}) = R_1(T|N_1) \cdot R_2(T|N_2) \cdot \ldots \cdot R_P(T|N_p)$
6. If the reliability $R_S(T|\boldsymbol{N})$ is insufficient and the budget has not been used up yet, use equations (4) to (6) to update the marginal increase of reliability $\ln\big(\delta_{i^*}(R_{i^*})\big)$, for the newly inserted spare parts part i^*. Return to Step 2. Otherwise exit, an optimal NDS has been found.

3. The improved algorithm

The use of a constant failure rate is suitable for electronic devices, but not for mechanical parts subjected to ageing and wearing. Thus, to extend the SpCCA to the case of time-dependent failure rate $\lambda(t)$, substantial modifications are needed.

3.1 Variable failure rate; the single location case

When the failure rate grows over time, faults cannot be modelled as Poisson random variables and Eq. (4) does not hold. To find the general way to compute $R_i(T|N_i)$ we start by observing that, at time $t = 0$ there are exactly $(N_i + X_i)$ spare parts of type i: N_i are in the warehouse and the other X are installed in the system. Since N_i substitution can be made at most, a stock-out for part i will not take place, if and only if the $(N_i + 1)$-th failure will occur after the mission time T. Owing to this observation, and letting $S_i^{(N_i+1)} = \big(T_1 + T_2 + \cdots + T_{N_i} + T_{(N_i+1)}\big)$ be the sum of the $(N_i + 1)$ failure times of spare part i, we have that: $R_i(T|N_i) = P\left(S_i^{(N_i+1)} > T\right)$.

To compute $P\left(S_i^{(N_i+1)} > T\right)$, we assume that spare parts cannot be repaired, but they can only be replaced with new ones. In other words, letting $\lambda_i(0)$ and $\lambda_i(t^+)$ be, respectively, the failure rate of spare part i at the beginning of its useful life and immediately after its replacement, we have that $\lambda_i(t^+) = \lambda_i(0)$. Also, only for the moment, we consider the simplified, but common case of $P = M$, that is we assume that each part can be installed in a single location of the system. Under these hypotheses, the replacement process can be modelled as a "*renewal process*", because failure times (of spare parts of the same type) are i.i.d. random variables, with probability distribution function $f(t)$ that, from reliability theory, is known to be as in (9):

$$f(t) = \lambda_i(t) \cdot exp\left\{-\int_0^t \lambda_i(\tau)d\tau\right\}. \tag{9}$$

So, $R_i(T|N_i)$ can be computed as in Eq. (10), where the probability distribution function of $S_i^{(N_i+1)}$ is obtained taking the N_{i+1}-fold convolution $c_i^{(N_i+1)}(t)$ of $f(t)$:

$$R_i(T|N_i) = P\left(S_i^{(N_i+1)} > T\right) = 1 - \int_0^T c_i^{(N_i+1)}(t)dt = 1 - C_i^{(N_i+1)}(T). \tag{10}$$

Similarly, the probability $P_i(n = N_i)$ to have exactly N_i failures in the interval $(0, T]$ is given by Eq. (11):

$$P_i(n = N_i) = P\left(S_i^{(N_i+1)} > T\right) - P\left(S_i^{(N_i)} > T\right)$$
$$= \left(1 - \int_0^T c_i^{(N_i+1)}(t)dt\right) - \left(1 - \int_0^T c_i^{(N_i)}(t)dt\right) = C_i^{(N_i)}(T) - C_i^{(N_i+1)}(T). \tag{11}$$

Noting that $C_i^{(1)}(T) \equiv F_i(T)$ and that $C_i^{(0)}(T) = 1$, we also have that:

$$R_i(T|0) = 1 - C_i^{(N+1)}(T) = 1 - F_i(T) = R_i(T), \tag{12}$$

$$P_i(n = 0) = C_i^{(0)}(T) - C_i^{(1)}(T) = 1 - F_i(T) = R_i(T), \tag{13}$$

where $R_i(T) = \left(1 - F_i(T)\right)$ is the reliability of part i.

Using this set of equations to compute the value of the reliability $R_i(T|N_i)$, the SpCCA described in Section 2 remains optimal, also for parts with time-dependent failure rate.

3.2 Variable failure rate; the multiple locations case

If a part i can be installed in M_i different locations, although independent, failure times may not be identically distributed. This is because each location j could have a specific failure rate $\lambda_{ij}(t)$ and/or a different required operation time $t_{ij} \leq T$. The equality $R_i(T|N_i) = \mathrm{P}\left(S_i^{(N_i+1)} > T\right)$ is still valid, but $S_i^{(N+1)}$ cannot be computed as the N_{i+1}-fold convolution $c_i^{(N_i+1)}(t)$ of the probability distribution function $f(t)$. For this reason, a combinatorial approach is needed.

Let us consider a generic part i and a series of failures $\boldsymbol{F}_i \equiv \{f_{i,1}, f_{i,2}, \dots, f_{i,j}, \dots f_{i,M_i}\}$, where $f_{i,j}$ is the number of failures occurred at location j, within the mission time T. At the end of the mission, relatively to part i, the system will be up if and only if the cumulative number of failures, occurred at each location j, is lower or equal than the available number of spare parts N_i. In this regard, we can indicate as "positive-series" $\boldsymbol{F}_{k,N_i}$ a series of failures that leaves the system in an operating state (i.e., such that $\sum_{j=1}^{M_i} f_{i,j} \leq N_i$), and we can indicate with $\boldsymbol{\mathcal{F}}_{N_i} \equiv \{\boldsymbol{F}_{1,N_i}, \dots, \boldsymbol{F}_{k,N_i}, \dots, \boldsymbol{F}_{K_i,N_i}\}$ the set of all the possible K_i positive series of part i. $\boldsymbol{\mathcal{F}}_i$ can be organized in a $K \times M_i$ combinatorial table $\boldsymbol{CT}_i$, and sorted in terms of a total number of failures. For instance, considering a part i that can be installed in three different locations ($M_i = 3$), and for which two spare parts are available ($N_i = 2$), there is a total of ten positive series, such that $\sum_{j=1}^{3} f_{i,j} \leq 2$. This is shown in Table 1, where positive series are highlighted in bold.

Table 1
The combinatorial table for $M_i = 3$ and $N_i = 2$

Failures at Location #1	Failures at Location #2	Failures at Location #3	Total Failures
0	**0**	**0**	**0**
1	0	0	1
0	**1**	**0**	**1**
0	0	1	1
2	0	0	2
0	2	0	2
0	**0**	**2**	**2**
1	1	0	2
1	**0**	**1**	**2**
0	1	1	2

Due to the independence of failure times, the occurrence probability of the k-th positive sequence $\boldsymbol{F}_{k,N_i}$ equals the product of the probability of each one of its elements $f_{i,j}$. Also, since each sequence in $\boldsymbol{\mathcal{F}}_{N_i}$ corresponds to an operating state of the system, $R_i(T|N_i)$ can be obtained summing the occurrence probability of all sequences in $\boldsymbol{\mathcal{F}}_i$. In short:

$$R_i(T|N_i) = \sum_{k=1}^{K_i} P\left(\boldsymbol{F}_{k,N_i}\right) = \sum_{k=1}^{K_i} \left(\prod_{j=1}^{M_i} P\left(n_{i,j} = f_{i,j}^{(k)}\right)\right) =$$
$$= \sum_{k=1}^{K_i} \left(\prod_{j=1}^{M_i} P\left(n_{i,j} = \boldsymbol{CT}_i[k,j]\right)\right), \tag{14}$$

where:

- $P\left(n_{i,j} = f_{i,j}^{(k)}\right)$ is the probability that part i breaks down exactly $n_{i,j} = f_{i,j}$ times at location j and the superscript k indicate the reference sequence.
- $CT_i[k,j]$ is the value (i.e., number of failures) of the cell in position (k, j).

Also, since failure times of part i installed at location j are i.i.d., the probability $P\left(n_{i,j} = CT_i[k,j]\right)$ can be computed as in (11). For instance, considering the example of Table 1 we have:

$$R_i(T|N_i) = P\left(n_{i,1} = 0\right) \cdot P\left(n_{i,2} = 0\right) \cdot P\left(n_{i,3} = 0\right) + \cdots + P\left(n_{i,1} = 0\right) \cdot P\left(n_{i,2} = 1\right) \cdot P\left(n_{i,3} = 1\right) =$$

$$= \left(C_{i,1}^{(0)}(T) - C_{i,1}^{(1)}(T)\right) \cdot \left(C_{i,2}^{(0)}(T) - C_{i,2}^{(1)}(T)\right) \cdot \left(C_{i,3}^{(0)}(T) - C_{i,3}^{(1)}(T)\right) + \cdots + \left(C_{i,1}^{(0)}(T) - C_{i,1}^{(1)}(T)\right) \cdot$$
$$\cdot \left(C_{i,2}^{(1)}(T) - C_{i,2}^{(2)}(T)\right) \cdot \left(C_{i,3}^{(1)}(T) - C_{i,3}^{(2)}(T)\right)$$

Please note that we added the subscript j both to $n_{i,j}$ and to the convolution $C_{i,j}^{(0)}(T)$, because the distribution function of the failure rate of part i also depends on the location where it is installed.

Lastly, considering that SpCCA proceeds by marginal increments in the number of spare parts, to compute $R_i(T|N_i + 1)$ there is no need to consider all rows of CT_i, but only the ones that correspond, exactly, to a number $(N_i + 1)$ of failures. More formally, letting $K_i^{(N)}$ and $K_i^{(N+1)}$ be, respectively, the number of positive series when N_i and $(N_i + 1)$ spare parts are available, the following recursive formula can be finally obtained:

$$R_i(T|N_i + 1) = R_i(T|N_i) + \sum_{K_i^{(N)}+1}^{K_i^{(N+1)}} \left(\prod_{j=1}^{M_i} P(n_i = CT_i[i,j])\right). \tag{15}$$

For instance, relatively to the example of Table 1, if spares parts were incremented from two to three units, eleven additional rows (corresponding to three failures) should be added to CT_i and only these newly inserted rows would be needed to compute $R_i(T|N_i + 1)$. This is shown in Table 2, where new rows are highlighted in bold. Lastly, it is worth noting that, thanks to the recurrent structure of TM_i, the combinatorial table can be easily generated in an automatic way. A straightforward algorithm, based on integer partitions and on multiset permutations, is detailed in Appendix A.

Table 2
The combinatorial table for $M_i = 3$ and $N_i = 3$

Sequence	Failures at Location #1	Failures at Location #2	Failures at Location #3	Total Failures
1	0	0	0	0
2	1	0	0	1
…	…	…	…	…
9	1	0	1	2
10	0	1	1	2
11	**0**	**1**	**1**	**3**
12	**3**	**0**	**0**	**3**
13	**0**	**3**	**0**	**3**
14	**0**	**0**	**3**	**3**
15	**2**	**1**	**0**	**3**
16	**2**	**0**	**1**	**3**
17	**1**	**2**	**0**	**3**
18	**1**	**0**	**2**	**3**
19	**0**	**2**	**1**	**3**
20	**0**	**1**	**2**	**3**
21	**1**	**1**	**1**	**3**

4. The Weibull case

We now focus on a specific form of the failure rate $\lambda(t)$ that is frequently used to model the failure rate of mechanical parts:

$$\lambda(t) = \eta^{-\beta} \cdot \beta \cdot t^{\beta-1}. \tag{16}$$

In (16) both parameters η and β are non-null positive values: η is a shape parameter, whereas β is a scale parameter that determines the trend of the failure rate over time. More specifically, $\beta > 1$ is used to model mechanical parts subjected to wearing and/or degradation because, in this case, $\lambda(t)$ increases over time. Conversely, with $\beta = 1$ the failure rate is constant, typical behaviour for most electronics component. A special case is that of $0 < \beta < 1$, which corresponds to a decreasing failure rate, a particular behaviour that can sometimes be found during the "*infant mortality stage*" of new parts.
We also note that the failure rate modelled as in (16) corresponds to a two parameters Weibull probability function, with cumulative distribution function $F(t)$ given by:

$$F(t) = 1 - exp\left\{-\left(\frac{t}{\eta}\right)^{\beta}\right\}. \tag{17}$$

4.1 A Gamma Normal series truncation approximation

Unfortunately, for a Weibull distribution model, it is not possible to analytically obtain the function $C_i^{(N_i+1)}(T)$. Although this value could be numerically computed, the easiest way could be that to apply the Gamma-Normal Series Truncation approximation, proposed by Jiang (2008), to approximate a Weibull with the mixture of a Gamma $g\left(t; k = (\mu/\sigma)^2; \theta = (\sigma^2/\mu)\right)$ and of a Normal $\phi(\mu, \sigma)$ distribution, both having the same first two moments of the original Weibull distribution.

As known, if a random variable X_i is $g(t; k; \theta)$ then $\sum_{i=1}^{n} X_i \sim g(t; nk; \theta)$ provided all X_i are i.i.d. Similarly, if Z_i is $\phi(t; \mu, \sigma)$ then $\sum_{i=1}^{n} Z_i \sim \phi\left(t; n\mu; \sigma\sqrt{N_i}\right)$ provided all Z_i are i.i.d. Since the replacement process, we are considering is a Renewal Process, the i.i.d hypothesis holds and we can take advantage of the above-mentioned properties of the Gamma and Normal functions, to approximate the cumulative distribution of the n-fold convolution $C^n(t)$, of a Weibull random variable. This can be made as shown in Eq. (18):

$$C^n(t) = p \cdot G(t; nk, \theta) + q \cdot \Phi\left(t; n\mu; \sigma\sqrt{n}\right), \tag{18}$$

$$p = 1 + 0.4115 \cdot (1 - \beta), \tag{19}$$

$$q = 1 - p, \tag{20}$$

where G and Φ are, respectively, the cumulative distribution functions of the Gamma and of the Normal distribution, and β is the scale parameter of the original Weibull distribution.

Specifically, as shown by Jiang (2008), using as coefficients of the mixture distribution the values of p and q obtained through Eq. (19) and Eq. (20), the approximation error E, defined as in Eq. (21) is always less than 1%:

$$E = \max_{j}\left|\widehat{M}_{num}(t_j) - \widehat{M}_{app}(t_j)\right|, \tag{21}$$

where:
- $M(t_j)$ is the expected number of replacements in (0, t_j] as defined in Eq. (22).
- $\widehat{M}_{num}(t_j)$ is the estimation of $M(t_j)$, obtained via numerical computation of $C^{(j)}(t_j)$.
- $\widehat{M}_{app}(t_j)$ is the estimation of $M(t_j)$ obtained with the Gamma-Normal Series Truncation approximation of $C^{(j)}(t_j)$.

$$M(t_j) = \sum_{j=0}^{\infty} \left(j \cdot P(n = j, t_j) \right) = \sum_{j=0}^{\infty} \left(j \cdot \left(C^{(j)}(t_j) - C^{(j+1)}(t_j) \right) \right) = \sum_{j=1}^{\infty} C^{(j)}(t_j). \qquad (22)$$

We conclude this Subsection noting that using equation (11) and/or (14) to compute $R_i(T|N_i)$ and using Eq. (18) to compute the n-fold convolution $C^n(t)$, the standard SpCCA remains optimal also for mechanical parts characterized by a Weibull distributed failure rate. Consequently, hereafter this enhanced approach will be referred to as the Weibull-SpCCA.

4.2 A lower bound approximation of $R_i(T|N_i)$

When the n-fold convolution $C^n(t)$ are approximated, also $R_i(T|N_i)$ is affected by an approximation error. So, it may be useful to evaluate an exact lower bound of $R_i(T|N_i)$. To this aim, the Minimal Repair Model (MRM) can be used. According to the MRM, reparations/substitutions bring the system back to the operating state in which it was, immediately before the fault. Formally, the MRM assumes that $\lambda_i(t^-) = \lambda_i(t^+) \neq \lambda_i(0)$ where $\lambda_i(t^-)$ is the failure rate just before the failure occurred. Obviously, since we supposed that spare parts can be substituted but never repaired, the MRM underestimates the real reliability of the system and, as such, it gives a lower bound of the overall system's reliability.

For our purpose, the main advantage of the MRM model is the absence of discontinuity points of the function $\lambda_i(t)$ i.e, at time t when a failure occurs $\lambda_i(t^-) = \lambda_i(t^+)$. In virtue of this fact, the distribution of the number of failures in Δt follows a Non-Homogeneous Poisson process, similar to an ordinary Poisson process, except that the average rate of arrivals may vary over time.
The important point is that, a Non-Homogeneous Poisson, the probability $P_i(n = k)$ can be simply computed as in (23):

$$P_i(n = k) = \frac{\left((H_i(T))^k \cdot e^{-H_i} \right)}{k!}, \qquad (23)$$

where:

$$H_i(T) = \int_0^T \lambda_i(\tau) d\tau. \qquad (24)$$

Thus $R_i(T|N_i)$ can be exactly evaluated as in Eq. (25):

$$R_i(T|N_i) = \sum_{k=0}^{N_i} \frac{\left(\int_0^T \lambda_i(\tau) d\tau \right)^k \cdot e^{-\int_0^T \lambda_i(\tau) d\tau}}{k!} = \sum_{k=0}^{N_i} \frac{(H_i(T))^k \cdot e^{-H_i(T)}}{k!}. \qquad (25)$$

Also, if part i is installed in M_i locations, the overall failure rate is given by the sum of all $H_{i,j}(T)$:

$$H_i(T) = \int_0^{t_{ij}} \sum_{j=1}^{M_i} \lambda_{ij}(\tau) d\tau = \sum_{j=1}^{M_i} \int_0^{t_{ij}} \lambda_{ij}(\tau) d\tau = \sum_{j=1}^{M_i} H_{ij}(t_{ij}). \qquad (26)$$

Lastly, we observe that, for a Weibull distribution, Eq. (26) simplifies as follows:

$$H_i(T) = \sum_{j=1}^{M_i} \int_0^{t_{ij}} \left((\eta_{ij})^{-\beta_{ij}} \cdot \beta_{ij} \cdot (t_{ij})^{(\beta_{ij}-1)} \right) d\tau = \sum_{j=1}^{M_i} \left(\frac{t_{ij}}{\eta_{ij}} \right)^{\beta_{ij}}. \qquad (27)$$

Hereafter, the use of this lower bound approach for the computation of $R_i(T|N_i)$ will be referred to as the Weibull-MRM-SpCCA.

5. Numerical analysis

To assess the robustness of the Weibull-SpCCA, a software application, developed in Delphi®, was built. Given a specific instance of the spare parts configuration problem, the application returns the Pareto optimal frontier; its first point corresponds to an empty warehouse (in this case the solution cost is zero and the system's reliability equals the product of the reliability of all its parts), whereas the last one, is the most reliable spare parts configuration that can be generated with the available budget. The problem instance passed as input can be randomly generated or manually defined by the user. Anyway, the following parameters are considered: mission time, required reliability level, available budget (or available warehouse space), number of locations, number of parts, their cost (or occupied space) and failure rate.

For each spare parts configuration lying on the Pareto optimal frontier, the software also returns a numerical estimation of the corresponding reliability level. This estimation is obtained performing a Monte Carlo simulation, which takes advantage of the inverse function technique, to generate failure times accordingly to the chosen failure rate (Law, 2014). The interested reader is referred to Appendix B for further details.

5.1. Robustness of the Gamma Normal series truncation approximation

At first, we used the software to assess the accuracy of the Gamma Normal approximation in estimating $R_i(T|N_i)$. To this aim, we considered a mission time T of 1000 units of time and an unreliable part (with $\beta = 1.2$, $\eta = 800$, Mean Time Between Failure MTBF = 752.5 units of time), and we computed $R_i(T|N_i)$ both via simulation (averaged over 100,000 repetitions) and using the set of equations given in Section 3 and Section 4. Results, for different levels of N_i in the range [0; 11] and of M_i in the range [1;4], are shown in Fig. 1.
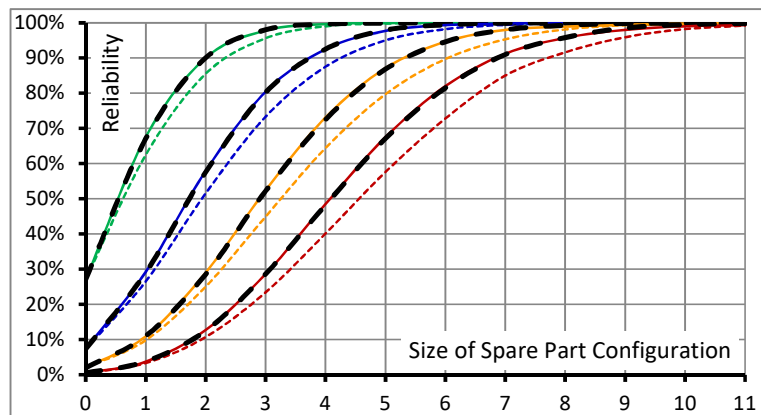

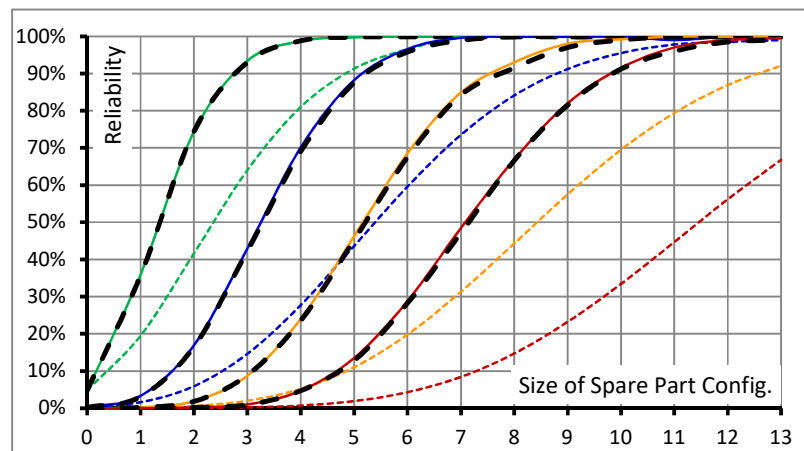**Fig. 1.** Comparison of different estimates of $R_i(T|N_i)$ for an unreliable part.

In Fig. 1, continuous, dotted and dashed lines represent, respectively, the values of $R_i(T|N_i)$ obtained with the Gamma Normal approximation, with the MRM approach and via Monte Carlo simulation. The different colours refer to the number of locations: green corresponds to a single location, while red corresponds to four locations.

As can be seen, the Gamma Normal approximation is robust and accurate, as the continuous and the dashed line almost coincide. Also, the MRM approach works quite well; being a lower bound its value is always below that of the other curves and, as expected, the estimation error amplifies when the number of locations increases. Anyhow, the average percentage error, relative to the value of $R_i(T|N_i)$ obtained through simulation, is rather low, as clearly shown by Table 3.

Table 3
Percentage Error in the estimation of $R_i(T|N_i)$ for an unreliable part

Number of Locations	MRM – Lower Bound			Gamma Normal Approximation		
	Average % Error	Max % Error	Min % Error	Average % Error	Max % Error	Min % Error
1	3.3%	7.4%	0.1%	0.2%	0.7%	0.0%
2	5.1%	11.7%	0.1%	0.3%	0.7%	0.0%
3	7.4%	16.3%	0.4%	0.5%	1.8%	0.0%
4	10.8%	22.2%	0.6%	0.7%	2.6%	0.0%

As clearly shown by Fig. 2 and by Table 4, the Gamma Normal approximation remains robust also for a very unreliable part with $\beta = 1.6$, $\eta = 500$ and MTBF = 448.3 units of time. Conversely, the lower bound provided by the MRM approach becomes almost useless. In this extreme condition, the probability to observe many failures is high and, consequently, the minimal repair hypothesis becomes too restrictive and the real value of $R_i(T|N_i)$ is strongly underestimated.

Fig. 2. Comparison of different estimates of $R_i(T|N_i)$ for a very unreliable part.

Table 4
Percentage Error in the estimation of $R_i(T|N_i)$ for a very unreliable part

Number of Locations	MRM – Lower Bound			Gamma Normal Approximation		
	Average % Error	Max % Error	Min % Error	Average % Error	Max % Error	Min % Error
1	34.6%	83.0%	0.5%	0.3%	0.6%	0.1%
2	65.8%	189.7%	0.9%	1.2%	1.8%	0.7%
3	153.3%	352.0%	8.5%	1.6%	2.7%	0.8%
4	321.6%	584.2%	48.2%	2.1%	3.9%	0.8%

5.2 Performance of the WCA

Next, we used the software application to investigate the quality of the spare part configurations generated by the Weibull SpCCA. As mentioned above, apart from the small approximation error in the estimation of the *n*-fold convolutions, the solution generated by the Weibull SpCCA is optimal. What we want to investigate is how much better this solution is, relatively to the ones generated by the MRM and by the Exponential SpCCA. To this end, we compared the non-dominated spare parts configurations generated by the three constructive algorithms, in eight alternative scenarios, generated as a full factorial combination of the following parameters:

- parts can be either Unreliable (U) or Very Unreliable (VU). In the case of U parts, β is uniformly distributed on [1, 1.2] and η is uniformly distributed on [0.8T, 1.2T], with an average MTBF of 0.965·T units of time. In the case of VU parts, β is uniformly distributed on [1.2, 1.6] and η is uniformly distributed on [0.5T, 0.9T], with an average MTBF of 0.638·T units of time.
- The number of critical parts P can be either 5 or 10.
- The number of locations M_i for each critical part i, can be randomly generated either from the set {1, 2} or from the set {2, 3, 4}.

In each simulated scenario, the mission time T equals 1000 units of time, the budget equals 100, the price of each component is randomly generated in the range [0.1; 1.5] and the target level of reliability is set to 90%.

For the sake of completeness, we also considered two additional critical scenarios, characterized by 10 critical parts, four of type U, four of type VU and two Halfway (H) between U and VU, with an average MTBF of 0.8·T units of time. Each part is installed in 2 and in 4 locations, in the first and in the second scenario, respectively.

For each one of the 10 alternative scenarios, 100 instances were randomly generated and solved with the three algorithms. We note that to ensure comparability, the constant failure rate λ, needed by the exponential approach, was defined as in Eq. (28):

$$\lambda = \left(\eta \cdot \Gamma \left(1 + \frac{1}{\beta} \right) \right)^{-1} \tag{28}$$

where $\Gamma(x)$ is the Gamma Function. In this way, the MTBF of the parts is the same, both in case of Weibull and of the Exponentially distributed failure times.

An example of the obtained Pareto Optimal Solution is shown in Fig. 3, relative to the case of 4 parts of type U, 4 of type VU and 2 of type (H), each one installed in 4 different locations of the mechanical system.
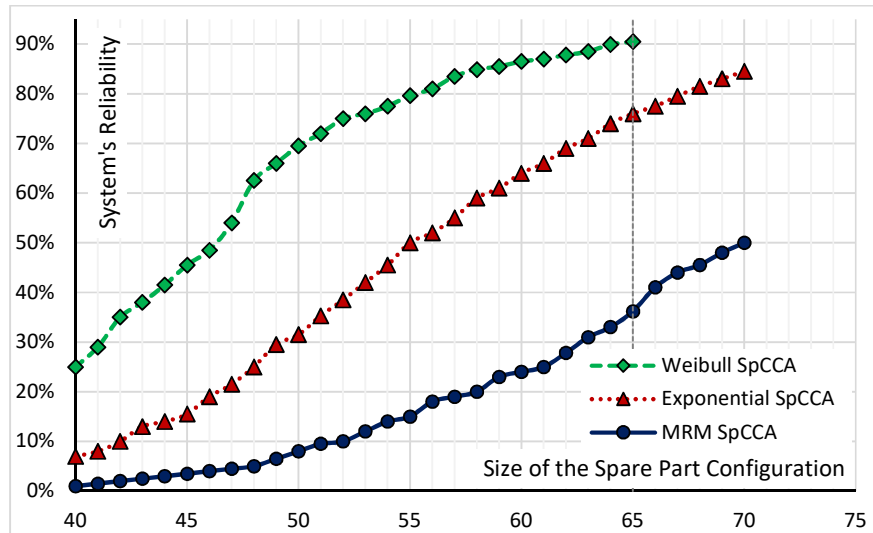

Fig. 3. An example of the Pareto Optimal Frontier generated by the three constructive algorithms

The Pareto optimal frontier generated by the SpCCA is the only one that stops before the budget has been fully utilized. As soon as the target reliability of 90% is reached, with the purchase of 65 spare parts, the algorithm stops. Conversely, due to the underestimation of the true value of the system's reliability, neither the Exponential (in red) nor the MRM (in blue) approach reaches the target and so, new spare parts are continuously added, until the budget runs out.

This concept is clearly exemplified in Table 5, which show the size, the cost and the true reliability of the last non dominated solution generated by each approach. The optimal solution proposed by the Weibull SpCCA has a size of 65, uses 91.4% of the budget and perfectly match the reliability target. Conversely, the exponential model proposes a solution of size 71 that consumes all of the budgets, with an underestimated reliability of 84.5%. This solution is definitively oversized, as its true reliability is 97% much higher than the target. A similar and even worse solution is proposed by the MRM SpCCA.

Table 5
Last generated configuration

Algorithm	Last NDS Spare part configuration	Size	% of Budget	Computed value of $R_S(T|N)$	True value of $R_S(T|N)$
MRM	{9, 10, 10, 9, 5, 5, 5, 5, 6, 7}	71	100%	0.510	0.970
Exponential	{10, 10, 10, 10, 5, 4, 4, 4, 6, 7}	70	100%	0.845	0.930
Weibull	{7, 9, 9, 8, 5, 5, 6, 5, 6, 6,}	65	91.4%	0.900	0.905

For this reason, in case of Weibull SpCCA, the last point of the Pareto frontier always coincides with the spare parts configuration that reaches the target at the minimum cost, or, in case of insufficient budget, it is the configuration ensuring the maximum reliability that can be achieved with the available budget. Conversely, the last point of the Pareto frontier of the other two approaches is an oversized configuration, with a number of spare parts higher than the required ones. Owing to this consideration, the alternative algorithms were compared relative to the percentage error E, computed as in Eq. (29):

$$E = \frac{R(SPC_W) - R(SPC_{O,N})}{R(SPC_W)},\qquad(29)$$

where:
- $R(SPC_W)$ is the true reliability of the spare part configuration SPC_W corresponding to the last point of the Pareto optimal frontier generated by the Weibull SpCCA.
- N is the number of spare parts (i.e., the size) of the spare parts configuration SPC_W.
- $R(SPC_{O,N})$ is the true reliability of the spare part configuration of size N lying on the Pareto optimal frontier generated by the other two approaches.

Table 6
Obtained performance

SIMULATED SCENARIO		EXPONENTIAL SpCCA			MRM SpCCA		
Number and type of parts	Number of Positions	Average % Error	Max % Error	Min % Error	Average % Error	Max % Error	Min % Error
5 U	{1, 2}	1.40%	5.75%	0.00%	0.90%	4.95%	0.00%
5 VU	{1, 2}	1.40%	5.20%	0.00%	1.60%	5.40%	0.00%
5 U	{2, 3, 4}	1.40%	5.00%	0.00%	1.50%	4.50%	0.00%
5 VU	{2, 3, 4}	1.51%	5.80%	0.00%	6.40%	13.90%	0.00%
10 U	{1, 2}	3.50%	10.30%	0.00%	3.40%	14.30%	0.00%
10 VU	{1, 2}	4.80%	15.60%	0.00%	11.64%	32.50%	1.50%
10 U	{2, 3, 4}	5.00%	13.80%	0.00%	6.60%	12.50%	0.00%
10 VU	{2, 3, 4}	5.45%	20.00%	1.20%	13.60%	28.00%	4.60%
4U 4VU 2H	{2}	1.80%	4.35%	0.00%	16.40%	26.80%	8.60%
4UP 4VUP 2H	{4}	2.10%	4.90%	0.00%	39.40%	50.00%	28.50%

Obtained results are shown in Table 6. As expected, the Weibull SpCCA is always the best approach, whereas the performance of the MRM SpCCA rapidly deteriorates, as soon as parts' reliability decrease and/or the number of locations increases. Instead, the basic Exponential SpCCA ensures stable and rather good performance, as indicated by an average error that is higher than 5% only in case of extreme scenarios. Hence, one could argue that the Exponential SpCCA could be used also for mechanical parts subjected to ageing and wearing. Actually, the generated solution would be good, if not even optimal, but the real value of the obtained reliability would remain unknown since the exponential model underestimate the value of $R_i(T|N_i)$ and, consequently, that of $R_S(T|\boldsymbol{N})$. Also, and perhaps more important, as we mentioned before, due to the underestimation of $R_S(T|\boldsymbol{N})$, the Exponential SpCCA tends to build an oversized configuration with more spare parts than required, yet the target reliability level could be achieved with a much cheaper configuration.

6.   Conclusions and future works

The paper focused on a specific case of spare parts inventory optimization, where a mechanical system that must work, continuously, for a mission time *T* during which, although it can be repaired, spare parts cannot be resupplied. To solve the problem, an optimal constructive algorithm was developed. The algorithm proceeded by steps and generated the Pareto optimal frontier of the non-dominated spare parts inventory configurations maximizing the overall system's reliability at increasing levels of cost. The algorithm was introduced in general terms and could be used for any probability distributions of the failure times of the critical parts of the system. A specific version of the algorithm, called the Weibull SpCCA, was also formulated to treat the very common case of Weibull distributed failure times. In this peculiar condition, some approximations were made, especially to compute the *n*-fold convolutions of Weibull random variables, needed for the estimation of the system's reliability. Due to the introduced approximations, a software application was also developed to assess the accuracy and quality of the algorithm. Obtained results, compared with exact lower bounds and with solutions obtained via Monte Carlo simulation, are more than satisfactory and certified the quality of the Weibull SpCCA and its practical usability.

At present, the algorithm is based on the fundamental hypothesis that all the critical parts of the system operate in series and so they must all be working for the system to be operational. Future researches are intended to relax this constraint, to be able to consider more complex arrangements of the system's critical parts. We also note that the Pareto optimal frontier may not be unique. In fact, there could be different spare parts configurations with the same cost and the same reliability or, in any case, with very similar values. Thus, it could be interesting to enlarge the Pareto optimal frontier including all the non-dominated solutions (for each pair of cost and reliability), so as to give the user a wider range of options.

References

Ahmed, I. & Sultana, I. (2013). A literature review on inventory modelling with reliability considerations. *International Journal of Industrial Engineering Computation, 5*(1), 169 – 178.

Basten, R.J.I. & Ryan, J.K. (2019). The value of maintenance delay flexibility for improved spare parts inventory management. *European Journal of Operational Research, 278*(2), 646 – 657.

Cai, J., Li, X. & Chen, X. (2017). Joint optimization of maintenance inspection and spare provisioning for aircraft deteriorating parts. *Journal of Systems Engineering and Electronics, 28*(6), 1133-1140.

Costantino, F., Di Gravio, G., Patriarca, R. & Petrella, L. (2018). Spare parts management for irregular demand items. *Omega, 81*, 57-66.

De Smidt-Destombes, K.S., Van Der Heijden,M.C. & Van Harten, A. (2009). Joint optimisation of spare part inventory, maintenance frequency and repair capacity for k-out-of-N systems. *International Journal of Production Economics, 118*, 260 - 268.

Denicoff, M., Fennell, J., Haber, S.E., Marlow, W.H. & Solomon, H. (1964). A Polaris logistic model, *Naval Research Logistics Quarterly*, *11*(4), 259–272.

Do Rego, J.R & De Mesquita, M.A. (2011). Spare parts inventory control: a literature review. *Producão*, *21*(4), 656-666.

Godoy, D.R., Pascual, R. & Knights, P. (2013). Critical spare parts ordering decisions using conditional reliability and stochastic lead time. *Reliability Engineering & system Safety, 119*, 199 – 206.

Hu, T.C. & Tien, B.N. (1976). Generating permutations with non-distinct items. *American Mathematical Monthly, 83*, 193-196.

Jiang, R. (2008). A Gamma–normal series truncation approximation for computing the Weibull renewal function. *Reliability Engineering & System Safety, 93*, 616-626.

Johnson, S.M. (1963). Generation of permutations by adjacent transpositions. *Mathematics of Computation, 17*, 282-285.

Khanlarzade, N., Yegane, B.Y., Kamalabadi, I.N. & Farughi, H. (2014). Inventory control with deteriorating items: a state of the art literature review. *International Journal of Industrial Engineering Computation, 5*(2), 179 – 198.

Kennedy, W.J., Patterson, J.W., & Fredendall, L.D. (2002). An overview of recent literature on spare parts inventories. *International Journal of Production Economics*, *76*, 201–215.

Korsh, J. & Lipschutz, S. (1997). Generating Multiset Permutations in Constant Time. *Journal of Algorithms, 25*, 321-335.

Law, A.M. (2015). *Simulation modelling and analysis*. 5th Edn., McGraw-Hill Education, USA., ISBN-13: 978-0073401324.

Lehmer, D.H. (1964). *The machine tools of combinatorics*. In Applied Combinatorial Mathematics, edited by Beckenbach, E.F., pp. 5-31, Wiley, New York.

Rajpal, P.S., Shishodia K.S. & Sekhon, G.S. (2006). An artificial neural network for modeling reliability, availability and maintainability of a repairable system. *Reliability Engineering & system Safety,* 91, 809-819.

Renna, P. (2017). Flexibility configurations and preventive maintenance impact on job-shop manufacturing systems. *International Journal of Industrial Engineering Computation, 8*(4), 481-492.

Sheng, J. & Prescott, D. (2017). A hierarchical coloured Petri net model of fleet maintenance with cannibalisation. *Reliability Engineering & System Safety, 168*, 290-305

Sherbrooke, C.C. (1968). Metric: A multi-echelon technique for recoverable item control. *Operations Research, 16*(1), 122-141.

Sherbrooke, C.C. (2004). *Optimal Inventory Modeling of Systems: Multi-Echelon Technique*. 2nd Edn., Kluwer Academic Publishers, USA., ISBN-13: 978-1402078491.

Teunter R.H. & Fortuin L. (1999). End-of-life service. *International Journal of Production Economics*, *59*, 487–497.

Teunter R.H. & Haneveld, W.K.K. (2002). Inventory control of service parts in the final phase. *European Journal of Operational Research, 137*, 497-511.

Van der Auweraer, S., Boute, R.N. & Syntetos, A.A. (2019). Forecasting spare part demand with installed base information: a review. *International Journal of Forecasting, 35*(1), 181-196.

Van der Auweraer, S. & Boute, R. (2019). Forecasting spare part demands using service maintenance information. *International Journal of Production Economics, 213*, 38-149.

Van Horenbeek. A., Scarf. P.A., Cavalcante, C.A.V. & Pintelon. L. (2013). The Effect of Maintenance Quality on Spare Parts Inventory for a Fleet of Assets. *IEEE Transactions on Reliability, 62*(3), 596 – 607.

Van Jaarsveld, W. & Dekker, R. (2011). Spare parts stock control for redundant systems using reliability centered maintenance data. *Reliability Engineering & System Safety, 96*(11), 1576 -1586.

Vaughan T.S. (2005). Failure replacement and preventive maintenance spare parts ordering policy. *European Journal of Operational Research*, *161*, 183–190.

Yang, R. & Kang. J. (2017). A joint optimal policy of block preventive replacement and spare part inventory. *International Journal of Systems Assurance Engineering and Management*, *8*(4), 740-746.

Willemain T.R., Smart C.N. & Schwarz H.F. (2004). A new approach to forecasting intermittent demand for service parts inventories. *International Journal of Forecasting, 20*, 375–387.

Appendix A. Automatic generation of the combinatorial table

To explain how the combinatorial table CT_i could be automatically built, let us consider again the numerical example of Section 3.2, and suppose that a total of two failures have occurred. This event corresponds to all the positive sequences listed from Row 5 to Row 10 of Table 1. Since parts are installed in three different locations, to get a total of two failures, either they both occurred in the same location or there was a single fault in two different locations. This subdivision corresponds, exactly, to the integer partitions of the number two. We remind that, in number theory, an integer partition of a positive integer n, is a way of writing n as a sum of positive integers. For $n = 2$ there are only two possible partitions: $\langle 2 \rangle$ and $\langle 1; 1 \rangle$. Also, since $M_i = 3$, to fully describe our event, we must rewrite the previous partitions, as the sum of three addendums. In other words, we need to insert an addendum equal to zero for each location where no faults occurred i.e., $\langle 2 \rangle \rightarrow \langle 2; 0; 0 \rangle$ and $\langle 1; 1 \rangle \rightarrow \langle 1; 1; 0 \rangle$.

Lastly, in order to come up with all possible events corresponding to two failures, we need to evaluate in how many ways these failures can be rearranged among M_i locations. That is, we need to evaluate all the $(M_i!/(m_1! \, m_2! \ldots m_k!))$ multiset permutations (i.e., permutations with repetitions) of n failures in M_i locations, when $m_1, m_2, \ldots, m_k$ failures occur in the same location. For instance, considering partition $\langle 2; 0; 0 \rangle$ we have: $M_i = 3$, $m_1 = 2, m_2 = 0, m_3 = 0$. So, there are $(3_i!/(2! \, 0! \, 0!)) = 3$ multiset permutations to be considered. These are $\{2, 0, 0\}, \{0, 2, 0\}, \{0, 0, 2\})$, which correspond to Row 5 to Row 7 of Table 1.

It follows that the rows of CT_i corresponds to:
- the integer partitions (with a number of addendum lower or equal to the available positions M_i) of all the natural numbers equal or lower than N_i;
- the multiset of each integer partition.

Generating all the integer partitions of the number n in lexicographical order (i.e., from the first trivial partition $\langle n \rangle$ to the last one made of n repetitions of the number 1 $\langle 1_1, 1_2, \ldots, 1_n \rangle$) is straightforward. Specifically, given the partition $I\langle \pi_1; \ldots; \pi_s \rangle$ obtained at a certain step, the next one, in lexicographical order, is generated by means of the following steps:

- Identify the last component of partition I greater than one. Let this component be π_r.
- Compute $S = \pi_{(r-1)} + \cdots + \pi_s$.
- Create a new partition as follows:
  - Copy all elements $\pi_1; \ldots; \pi_{(r-1)}$ i.e., the part of the sequence to the left of π_r is preserved.
  - The element π_r must be decreased by one ($\pi_r = \pi_r - 1$) and followed by the first integer partition of the number S with all addendums equal or lower than $(\pi_r - 1)$.
  - This partition is made of $\lfloor (S/(\pi_r - 1)) \rfloor$ repetitions of $(\pi_r - 1)$ and $[S \bmod (\pi_r - 1)]$ repetitions of 1, where **mod** is the modulus operator returning the reminder after division.

Generating permutations is, instead, a more complex problem. Running through this topic goes beyond the purpose of this paper, also because, in case of spare parts optimization, the number of different locations (and thus the number of elements that need to be permuted) is rather low. Thus, any exhaustive algorithm could be used without dramatically worsening the computation time. However, the software that we implemented took advantage of the very performing HT algorithm proposed by Hu and Tien (1976). Briefly, this algorithm integrates the algorithm by Johnson (1963) that lists, using only adjacent transpositions, all permutation of n distinct numbers, and the algorithm by Lehmer (1964) that lists all

the permutation of a n-bit vectors (i.e., a 0 - 1 binary vectors of length n). For more details the interested reader is referred to the paper by Korsh and Lipschutz (1995) where a C++ implementation of the permutation algorithm can be found.

Appendix B. The Monte Carlo simulation pseudo-code

Let $F(t)$ be the cumulative distribution function of a random variable X. If $F(t)$ is continuous and one-to-one, then $F(U)$ has the same probability distribution of X, provided that U is a random variable uniformly distributed on [0, 1]. Thus, a Weibull distributed failure time f_{ij} (of component i installed at location j) can be generated as in (a.1):

$$f_{ij} = \eta_{ij} \cdot \ln\left(\frac{1}{1-U}\right)^{\frac{1}{\beta_{ij}}}, \tag{A.1}$$

where: $\eta_{i,j}$ and $\beta_{i,j}$ are the Weibull's parameters.

Given a spare parts inventory configuration $\boldsymbol{N} = \{N_1, N_2, \dots, N_i \dots, N_P\}$, for each part $i \in \{1, \dots, P\}$ and for each one of its possible locations $j \in \{1, \dots, M_i\}$, a sequence of failure times $\boldsymbol{f_{ij}} = \{f_{1ij}, f_{2ij}, \dots, f_{kij}, \dots, f_{Kij}, f_{(K+1)ij}\}$ is randomly generated using (a.1), until the sum of the failure times does not exceed the required operational time $t_{ij} \leq T$, i.e., $\sum_{k=1}^{K} f_{kij} \leq t_{ij}$ and $\sum_{k=1}^{K+1} f_{kij} > t_{ij}$. In other words, K_{ij} is the number of the randomly generated failures of part i occurred at location j, during the mission time T. Since component i is installed in M_i locations, for a mission to be successfully completed, the sum of the number of failures K_{ij} on all locations M_i, and for all part i must be lower that the available stock N_i, i.e., if $\sum_{j=1}^{M_i} K_{ij} \leq N_i \ \forall i$. By operating in this way, the system's reliability can be estimated as the frequency of the successful missions, relatively to the total number of simulated missions. This is clearly detailed in the Delphi code of figure a.1.

```
Interface
Type
Tspare = Record
    M: Integer; // Number of locations
    N: Integer; // Number of spare parts available in the warehouse
    Tmission: Array of Integer;
    Eta:  Array of real; // Weibull □
    Beta: Array of real; // Weibull □
    Tfailure: Array of real; // Failure time at each location
    Reliability: Real;
End;
Function LifeSim(Spare: Tspare): Boolean;  // Simulates the life of a spare and returns true if it survives
Function CompR (Var Cpn : Array of Tspar; SimNum: Integer = 100000): Real; // Computes the system's reliability
Function FTime (Eta, Beta: Real): Real; // Generates Weibull distributed failure times

Implementation

Function FTime (Eta, Beta: Real): Real;
// Failure times t are generating as follows: t = F⁻¹(y), where y is uniformly distributed in [0, 1]
Var Base, Exponent, y: Real;
begin
  y := Random ();
  Base := Ln(1/(1-y));
  Exponent := (1/Beta);
  FTime := Eta * Power (Base, Exponent);
end;

Function LifeSim(Spare: Tspare): Boolean;
Var Survive, Stop: Boolean;
Var  j, N: Integer;
```

```
Begin
  Survive := True;
  N := Spare.N;
  For j:= 0 To (Spare.M - 1) Do
  Begin
    If Survive Then
    Begin
      Spare.Tfailure[j] := 0; // for each (new) location j the lifetime restarts from zero
      Stop := False;
      Repeat
          Spare.Tfailure[j]:= Spare.Tfailure[j] + Ftime(Spare.Eta[j], Spare.Beta[j])
          If  Spare.Tfailure[j] >= Spare.Tmission[j] Then Stop := true // spare has survived
          Else Dec(N); // spare has not survived and must be replaced
          If N < 0 Then // stock out
           Begin
              Stop := True;
              Survive := False;
           End;
      Until Stop;
    End
  End;
  LifeSim := Survive;
End;

Function CompR (Var Spares: Array of Tspare; SimNum: Integer = 100000): Real;
Var R: Real;
Var Survive: Boolean;
Var Successes, i, j: Integer;
Begin
  R := 1;
  For i := 0 to High(Spares) Do // Rᵢ(T|Nᵢ) is computed (via simulation) for each spare part
  Begin
    Successes := 0;
    For j := 1 To SimNum Do
    Begin
      Survive := SimLifea(Spares[i]);
      If Survive Then Inc(Successes);
    End;
    If Successes > 0 Then Spares[i].R := (Success/SimNum)
      Else Spares[i].R := 0;
   R:= R*Cpn[i].Aff; //Overall reliability
  End;
  ComputeR := R;
End;
End.
```

Fig. A.1. Reliability's simulation pseudo code