Int. J. Production Economics 266 (2023) 109034

Contents lists available at ScienceDirect

International Journal of Production Economics

journal homepage: www.elsevier.com/locate/ijpe

Developing a framework for generating production-dependent failure rate through discrete-event simulation

Leonardo Leoni, Filippo De Carlo *, Mario Tucci

Department of Industrial Engineering (DIEF), University of Florence, Italy

ARTICLE INFO

Keywords:
Discrete-event simulation
Production-dependent failure rate
Maintenance optimization
Failure re-scheduling
Dynamic non-homogenous Poisson process

ABSTRACT

Over the last decades, maintenance has experienced a transition from being a necessary evil to being a pivotal resource to create value for enterprises. Within the process of maintenance planning, distinct decisions could be responsible for different outcomes concerning profit and equipment reliability. Consequently, maintenance optimization has become pivotal to achieving relevant business goals. One of the most popular approaches to conduct maintenance optimization is simulation-based optimization, especially Discrete-Event Simulation (DES). Most works related to DES for maintenance optimization purposes focus on modeling imperfect maintenance or imperfect inspection and prognosis, while failures are often generated through a Weibull distribution. However, failure strongly depends on the production rate or the stress level, defining a Dynamic Non-Homogeneous Poisson Process (DNHPP). To this end, this paper proposes an algorithm for scheduling such DNHPP failure events in a DES framework model and, as a first implementation to apply it, an open-access library capable of generating stress level-dependent failures within the Rockwell ARENA© simulation environment. The developed package, that in the future will be ported to other relevant off-the-shelf simulation environments, provides a more realistic tool for maintenance engineers and researchers to optimize or compare maintenance strategies from an economic perspective.

1. Introduction

The importance of implementing a proper maintenance policy has progressively cleared up, due to the rise in availability requirements (Alabdulkarim et al., 2013). This has led to the development of several Preventive Maintenance (PM) policies, including Condition-Based Maintenance (CBM) (Florian et al., 2021; Havinga and de Jonge, 2020) and Total Productive Maintenance (Hung et al., 2022). CBM involves monitoring one or more degradation parameters, when a given threshold is reached, a maintenance task is scheduled. On the other hand, TPM is a holistic strategy that aims at pursuing the highest operational efficiency by involving the machine operators. This trend is also facilitated by the advances in technology, which are pivotal to improve the maintenance policies (Saihi et al., 2022). Any PM strategy may be described as a series of decisions that influence both operation cost and asset availability. Indeed, an appropriate maintenance strategy is fundamental to assure good working conditions, but it could result in high expenses (Peng et al., 2022). Accordingly, the adoption of distinct maintenance policies could lead to different profits and costs for the companies. As a result, there is an ongoing effort to optimize maintenance plans (Dursun et al., 2022; Mena et al., 2021).

Among the optimization approaches, there are some analytical models that can be adopted, however they conceal some limitations due to their assumptions, which could make the analytical models ineffective for real problems (Omoleye et al., 2019). As a result, Artificial Intelligence (AI) and simulation approaches have gained popularity for optimization purposes (Sharma et al., 2011). Simulation models require fewer assumptions compared to analytical model, allowing the modelling of more complex systems (Marsaro and Cavalcante, 2017). Thus, there is a great deal of research focused on maintenance simulation-based optimization (Davari et al., 2022; Nili et al., 2021).

Different simulation techniques such as agent-based simulation, system dynamics, and numerical approaches to solve an analytical problem could be adopted to model machinery degradation. Among these techniques, Discrete-Event Simulation (DES) is the most common employed (Alrabghi and Tiwari, 2015). Indeed, DES is very popular for modeling manufacturing environments with a general-purpose approach (Alrabghi and Tiwari, 2013). Within the context of DES for

* Corresponding author.
E-mail addresses: leonardo.leoni@unifi.it (L. Leoni), filippo.decarlo@unifi.it (F. De Carlo), mario.tucci@unifi.it (M. Tucci).

https://doi.org/10.1016/j.ijpe.2023.109034
Received 28 November 2022; Received in revised form 23 July 2023; Accepted 5 September 2023
Available online 11 September 2023
0925-5273/© 2023 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

maintenance applications, there are two main simulation tools: General Purpose Tools (GPTs) and Special Purpose Tools (SPTs). GPTs encompass simulation software that is not specifically designed for maintenance purposes, whereas SPTs are exclusively dedicated to maintenance simulation. SPTs have great potential and capabilities in modeling the downtime related to failures and failure generation. However, they usually neglect logistical and production factors. Conversely, GPTs are more flexible and could integrate logistics, production, and resource availability, but they have greater limitations for downtime and failure modeling. Accordingly, GPTs are generally preferable when there is the need to consider production aspects or spare part replenishment.

Considering DES, the machine failures are usually scheduled using an a-priori probability distribution (e.g., exponential or Weibull) based on the machine's operating state at the time of failure scheduling. However, the state of a real system could change over time, possibly leading to a change in the failure parameters. The former failure behavior is often referred to production-dependent or operation-dependent since it is strongly related to the production (or operation) condition (e.g., production rate). Accordingly, when the machine's state changes before a failure occurrence, the failure should be rescheduled. Failure rescheduling is a complex task, which is not generally considered within a DES environment. The inability to properly represent production-dependent failure behavior leads to less realistic models. Moreover, despite its significance, little interest has been devoted to production-dependent failures in a DES environment. Including such dependencies enables the simulation model to capture more realistic information about the occurrence of failures. This, in turns, brings simulation-based inferences about economic and operational performance closer to the actual behavior of the system, allowing to understand its sensitivity to the operating conditions and maintenance philosophies employed. To this end, this paper aims at developing an algorithm capable of re-scheduling the failure as soon as a change of the operation state is observed. Additionally, an open-access library to implement production dependent failure generation within one of the most popular GPT environment. This choice is made to enable the consideration of other aspects during the simulation.

The remainder of this paper is organized as follows. Section 2 presents the literature review with a particular focus on the adopted failure distributions and simulation tools. Section 3 gives an overlook of the concepts behind the model with a discussion on Weibull distribution and production-dependent failure rate. Section 4 describes the algorithm to generate failures and reschedule them based on a stress-level dependent failure rate. In Section 5, two practical applications and a numerical case study are presented. Finally, Section 6 presents the discussion, while Section 7 reports the conclusions along with limitations and possible future developments.

2. Background

This paper contributes to proposing a framework capable of dealing with production-dependent failure generation in a DES environment, while allowing the possibility to include imperfect maintenance. To ease the transition from theory to practice, an open-access library has been developed as well for one of the most common DES tools. In Section 2.1, a brief overview of recent papers on maintenance optimization and planning through DES is presented with a specific focus on the considered maintenance policy and simulation software. In Section 2.2, a summary of the considered failure generation for modelling maintenance activities in a DES environment is presented, along with the research gap and aim of the present paper.

Thanks to its advantages and strengths, DES has attracted significant attention during the past years. Table 1 reports a summary of papers between 2016 and early 2023 found through Scopus and Google Scholar, considering keywords such as "discrete event simulation", "discrete-event simulation", "maintenance", and "optim*", which could stand for optimization, optimization, and optimal. For each considered

Table 1
Considered papers and their related simulation tool and failure generation.

Failure generation	Reference	Year	Journal	Simulation tool
Weibull; General and generalized renewal process; NHPP	Velasquez et al.	2023	Sustainability	ARENA
	Blas et al.	2023	JART	DEVSJAVA
	Savolainen and Urbani	2021	JIM	Python
	Azevedo et al.	2020	AMM	–
	Golbasi and Turan	2020	CAIE	ARENA
	Ugurlu et al.	2020	EJPI	ARENA
	Wakiru et al.	2020	RESS	ARENA
	Turan and Golbasi	2019	ISMPES	ARENA
	Goti et al.	2019	AS	–
	Tan et al.	2019	IOP conference series	WITNESS
	Alqahtani et al.	2019	IJPE	ARENA
	Wang and Djurdjanovic	2018	Machines	–
	de Santana et al.	2018	MAR	–
	Wakiru et al.	2018	Procedia CIRP	ARENA
	Alrabghi et al.	2017	JMS	WITNESS
	Alqahtani and Gupta	2017a	JMSE	ARENA
	Alqahtani and Gupta	2017b	JIEM	ARENA
	Alrabghi and Tiwari	2016	RESS	WITNESS
Exponential	Mwanza et al.	2023	Modelling	Python
	Davari et al.	2022	JS	ARENA
	Cacereño et al.	2021	Mathematics	–
	Orlov et al.	2021	AS	CPN Tools
	Alrabghi	2020	JQME	Simio
	Turan et al.	2020	RESS	Python
	Golbasi and Turan	2020	CAIE	ARENA
	Linnéusson et al.	2020	EJOR	FACTS Analyzer
	Turan and Golbasi	2019	ISMPES	ARENA
	Wakiru et al.	2018	Procedia CIRP	ARENA
	Alrabghi and Tiwari	2016	RESS	WITNESS
Gamma	Alrabghi	2020	JQME	Simio
	Omoleye et al.	2019	JS	ARENA
	Alrabghi et al.	2017	JMS	WITNESS
Degradation model based on number of produced items	Bouslah et al.	2018	IJPE	C++ and ARENA
	Bouslah et al.	2016a	Omega	C++ and ARENA
	Bouslah et al.	2016b	IJPE	C++ and ARENA
Other (e.g., beta, normal, lognormal different mean TTFs, or not clearly specified)	Assid et al.	2023	CAIE	ARENA
	Fauadi et al.	2022	Journal of advanced manufacturing and technology	Anylogic
	Akl et al.	2022	RESS	Python
	Meissner et al.	2021	RESS	PreMade
	Orlov et al.	2021	AS	CPN Tools
	Darmawan and Sheu	2021	PMR	Flexsim
	Triska et al.	2021	IFAC	R – Simmer package
	Aliunir et al.	2020	JNCRS	Simevents

(continued on next page)

Table 1 (*continued*)

Failure generation	Reference	Year	Journal	Simulation tool
	Golbasi and Turan	2020	CAIE	ARENA
	Omoleye et al.	2019	JS	ARENA
	Turan and Golbasi	2019	ISMPES	ARENA
	Alrabghi et al.	2017	JMS	WITNESS
	Lahiani et al.	2016	IFAC	Flexsim

paper, Table 1 lists the journal, the year, the adopted simulation tool, and the Time To Failure (TTF) distribution. Since a study could adopt different failure generations (e.g., Weibull and exponential), the same study could fall in multiple categories. The following journals are identified through acronyms in Table 1: International Journal of Production Economics (IJPE), Reliability Engineering & System Safety (RESS), Journal of Intelligent Manufacturing (JIM), Applied Sciences (AS), Production & Manufacturing Research (PMR), Journal of Quality in Maintenance Engineering (JQME), Applied Mathematical Modelling (AMM), Computers & Industrial Engineering (CAIE), European Journal of Operational Research (EJOR), Engineering Assets and Public Infrastructures in the Age of Digitalization (EJPI), Journal of Simulation (JS), Journal of Manufacturing Systems (JMS), International Symposium on Mine Planning & Equipment Selection (ISMPES), Maintenance & Reliability (MAR), Joint Journal of Novel Carbon Resource Sciences & Green Asia Strategy (JNCRS), Journal of Management Science and Engineering (JMSE), Journal of Industrial Engineering and Management (JIEM), Journal of Applied Research and Technology (JART), Discrete Event Dynamic System (DEDS).

2.1. Maintenance policy and simulation software

First, PM and CBM are the most analyzed maintenance policies. Additionally, opportunistic maintenance, which was considered in many works (Alrabghi, 2020; Alrabghi et al., 2017; Alrabghi and Tiwari, 2016; Golbasi and Turan, 2020; Turan and Golbasi, 2019), has gained attention. As a matter of fact, time-based PM remains one of the most common maintenance strategies due to its simplicity and management with spare part replenishment. Furthermore, CBM has witnessed increase in popularity thanks to the advancements in software and data capabilities. However, the selection of the most suitable maintenance strategy heavily relies on factors such as the kind of failure, the available data, the ability to monitor an equipment, and the organizational needs. Thus, the present paper focuses on a generic algorithm that can be adopted for planning or optimizing any maintenance policy under the assumption that the failure or degradation could be modelled as a Weibull distribution with parameters that depend on the operating condition.

As shown in Table 1, most of the works adopted GPTs, with Rockwell ARENA© being the most commonly employed tool in fifteen papers (Bouslah et al., 2016a, 2016b, 2018; Alqahtani and Gupta, 2017a, 2017b; Omoleye et al., 2019, Alqahtani et al., 2019; Turan and Golbasi, 2019; Golbasi and Turan, 2020; Ugurlu et al., 2020; Wakiru et al., 2018, 2020; Davari et al., 2022; Assid et al., 2023; Velasquez et al., 2023). WITNESS seems to be another common software used for simulation purposes with three studies that employed it (Alrabghi and Tiwari, 2016; Alrabghi et al., 2017; Tan et al., 2019), while Flexsim was used in two papers (Darmawan and Sheu, 2021; Lahiani et al., 2016). Other adopted tools are DEVSJAVA (Blas et al., 2023), FACTS Analyzer (Linnéusson et al., 2020), Python (Mwanza et al., 2023), and R- Simmer package (Triska et al., 2021). This finding is aligned with the one reported by Dias et al. (2016), who identified Rockwell ARENA® as the most popular DES tool. Accordingly, this study selected Rockwell ARENA® as the simulation environment.

It is worth noting that many papers address relevant concepts such as imperfect maintenance (Azevedo et al., 2020; Wakiru et al., 2020; Goti et al., 2019; Alqahtani et al., 2019; Wang and Djurdjanovic, 2018; de Santana et al., 2018; Wakiru et al., 2018; Alqahtani and Gupta, 2017a; Alqahtani and Gupta, 2017b; Bouslah et al., 2016a) or joint optimization of spare part management (Turan et al., 2020; Aliunir et al., 2020) or workforce (Akl et al., 2022). Other papers are also incorporating agent-based (Fauadi et al., 2022), continuous simulation (Assid et al., 2023), or system dynamics (Savolainen and Urbani, 2021). These topics are really relevant for enhancing DES modelling by incorporating more realistic aspects. Indeed, maintenance or repair activities could not be able to completely restore the life of a given equipment. In this case, the maintenance is addressed as imperfect. Moreover, maintenance optimization could be particularly effective just for the management of maintenance activities. However, all the real industrial environments are complex systems, thus, the optimization of one process could negatively influence the others. Accordingly, including other factors in the optimization such as spare part management could be useful to obtain overall more profitable operations. According to the previous considerations, the possibility of specifying imperfect maintenance is considered in the developed algorithm. Moreover, since the algorithm is developed in a DES environment, it provides high flexibility for incorporating maintenance planning with other industrial aspects.

2.2. Failure generation

Given the relevance of the topic, researchers have focused on developing frameworks that can account for operation-dependent or production-dependent failures outside the field of GPT for DES (Colledani and Tolio, 2011; Colledani and Tolio, 2012; Martinelli and Piedimonte, 2008; Ouaret et al., 2017; Zied et al., 2011; Celen and Djurdjanovic, 2012; uit het Broek et al., 2021). This includes approaches like Markov modeling. However, as indicated by Table 1, limited attention has been given to production-dependent failures within the context of DES. Most of works consider classic failure distributions and particularly exponential, Weibull, and NHPP or Generalized Renewal Process. These distributions are common and appropriate for modelling various failure behaviors. Specifically, the exponential distribution is very common for electric devices characterized by random failures. However, it could also be used for other devices like pumps such as in Cacereño et al., 2021. On the other hand, the Weibull distribution could model wear-in failures of mechanical components, which are characterized by an increasing failure rate during the last part of their lives. In the work by Orlov et al. (2021), the authors also adopted a normal distribution. Indeed, even though normal distribution is usually considered for maintenance or repair activities, it could also be a proper distribution to model complex system whose failure depends upon many components. Finally, the gamma distribution is usually adopted to model the degradation within CBM framework (Alrabghi, 2020; Omoleye et al., 2019). It is worth mentioning that Goti et al. (2019) adopted the concept of a Weibull distribution that changes when a certain deterioration level is reached. This aligns with equipment that experiences a rapid increase in the failure rate in the last part of its life. Similarly, Wakiru et al. (2020) employed an impact rate to reduce the TTFs of different units as soon as a given deterioration of the lubricant is reached, denoting a state-rate interaction. The concept of production-dependent failure or stress-dependent failure or operation-dependent failure is usually associated with equipment whose deterioration process is influenced by the production rate or the type of operation that the equipment is performing. In real life, the majority of devices exhibit operation-dependent failure behavior. However, modelling such failures is much harder compared to classic time-dependent failures (Ait-El-Cadi et al., 2021). Accordingly, including production-dependent failure rate is a major challenge, and neglecting them is a limits the realism of the models.

A popular option to address the production-dependent condition in a

DES environment is linking the age of the machine to the number of parts produced (see sixth column of Table 1). Examples are Bouslah et al. (2016a, 2016b), and Bouslah et al. (2018). Specifically, the authors of the previous works developed through ARENA a machine's degradation based on the cumulative number of items produced. In other words, a higher productivity leads to more frequent failures. However, only the age of the machine varies based on productivity or the cumulative number of items produced, while the parameters characterizing the failure generation (e.g., shape and scale of a Weibull distribution) remain unchanged over time based on the production rate. In Bouslah et al. (2018), the concept of quality-dependent failure has also been introduced. In another interesting work by Meissner et al. (2021), a prescriptive maintenance for aircraft tires is proposed. The degradation of the tires is modelled through a degradation module which considers the assumed ambient conditions to define the health increment. Accordingly, the degradation of a given tire gradually increases. As a matter of fact, the degradation models allow to consider future conditions and schedule maintenance in case a given threshold is reached. Nevertheless, a reliability model would allow to extend the analysis to system or equipment which are not characterized by degradation parameters.

Based on the highlighted gaps and previous considerations, the main objective of this paper is to develop an algorithm capable of generating production-dependent failure in a DES environment, considering a reliability modelling. Moreover, the failure generation is characterized by parameters that vary according to the production state (e.g., productivity, worked items). Considering such a generation will allow to model a failure behavior that depends on both the time and the history of the operational state (i.e., the consumed life up to the state change). Thus, at a given time, the failure rate and the probability of failure are influenced by both time and operation-degradation history. To address such failures, the developed algorithm schedules and reschedules failure events as soon as a change in the production or operating condition occurs. Additionally, this work aims to build an open-access software library for DES commercial tools, capable of generating the aforementioned failure distributions. The developed library can also include imperfect maintenance condition. Based on the previous considerations, it is possible to state that the developed algorithm and library could be used to perform more realistic sensitivity or optimization analysis, enabling improved economic or availability-related decision-making process. It is worth pointing out that production-dependent or operation-dependent or stress-dependent failures will also be addressed as Dynamic Non-Homogenous Poisson Process (DNHPP), which is extensively described in Section 3.2. Although the algorithm is of general applicability in DES modeling, the software library was specifically developed for the Rockwell ARENA © environment, which is the most popular DES tool used in the industry (Dias et al., 2016). Rockwell ARENA © can generate failures based on a binary failure rate (i.e., an operating state and a stand-by state characterized by the absence of failures), however, there is no possibility to define multiple Weibull distributions related to distinct stress levels. At best, life-consuming can be tied to a resource state. Nevertheless, when failure occurs, the downtime can only be modelled as an independent probability distribution and is not easily bound to a logistic process, producing delays and waiting times due systems' state, the resource availability (both active resources like maintenance operators and spare parts) or external factors. To the best of the authors' knowledge, no one has attempted to create such a library for public use.

3. Conceptualization of the model

3.1. Stress level dependent failure rate

Any asset's behavior can be characterized by a specific lifetime distribution, which depends on several factors such as aging or environmental agents. In reliability analysis, the Weibull distribution is widely used for modelling lifetime data (T. Zhang and Xie, 2007). Indeed, in many reliability applications, the failure rate is considered following the so-named bath-tub curve (Jiang, 2013), and the Weibull distribution has proven itself as really flexible in modeling all its three sections (Xie and Lai, 1996).

The failure behavior of a given asset is strongly influenced by the production parameters, including throughput, working cycle, speed, and power usage. As a matter of fact, the reliability of certain machines could depend on how they are used (Francie et al., 2014). In other words, changes in production parameters lead to changes in failure parameters. Accordingly, assuming constant failure parameters oversimplifies the problem. Let $u = (1, \dots, m)$ be a vector defining a discretization of production rates or states (i.e., stress levels). Accordingly, the Weibull Probability Density Function (PDF) for a given state k is given by Eq. (1). This equation is a practical extension of the typical three-parameters Weibull distribution, expressed as $f(t) = \frac{\beta}{\eta}\left(\frac{max\{t-\gamma,0\}}{\eta}\right)^{\beta-1} e^{-\left(\frac{max\{t-\gamma,0\}}{\eta}\right)^{\beta}}$ (Wais, 2017).

$$f(t,k) = \frac{\beta_k}{\eta_k}\left(\frac{max\{t-\gamma_k,0\}}{\eta_k}\right)^{\beta_k-1} e^{-\left(\frac{max\{t-\gamma_k,0\}}{\eta_k}\right)^{\beta_k}} \quad 1 \leq k \leq m \tag{1}$$

Where β_k, η_k, and γ_k denote respectively the shape, the scale, and the location parameters related to the k-th stress level. Considering a single stress level, the reliability distribution can be expressed as shown in Eq. (2).

$$R(t,k) = e^{-\left(\frac{tmax\{t-\gamma_k,0\}}{\eta_k}\right)^{\beta_k}} \tag{2}$$

However, when the stress level varies in time, the reliability could not be calculated based on Eq. (2), but it should take into account the different visited states and the period of time spent in each state.

3.2. Dynamic Non-Homogenous Poisson Process

The failure process is a stochastic point process since its realizations are point events in time (i.e., the failures). When times between two consecutive failures are independent and identically distributed, the process is known as Homogenous Poisson Process (HPP). Under these circumstances, the time between two failures, also known as Time To Failure (TTF), follows an exponential distribution. Accordingly, failures are characterized by a constant arrival rate, which could be referred as rate of occurrence of failure (ROF). On the other side, if the TTFs are neither independent nor identically distributed, the stochastic point process is referred as Non-Homogenous Poisson Process (NHPP). The NHPP identifies a failure behavior with a variable ROF, which can be expressed through the Power Law Process (PLP), shown in Eq. (3) (BahooToroody et al., 2019).

$$ROF(t) = \left(\frac{\beta}{\eta}\right)\left(\frac{t}{\eta}\right)^{\beta-1} \tag{3}$$

where β and η are the shape and scale parameters respectively. It is worth mentioning that NHPP and the related PLP refers to repairable items, and the first TTF is Weibull distributed with parameters β and η (Kelly and Smith, 2009). β and η are usually considered constant, however, when failures are influenced by production factors, their values should vary based on the production state, or the operation state, or the stress level. In this paper, we propose a novel concept called DNHPP, which refers to an NHPP with failure parameters (i.e., β and η) that depend on a variable stress level arising from distinct production parameters.

3.3. Problem description, main assumptions, and notation

In this study, a machine characterized by a production-dependent failure behavior is considered. In other words, the failure parameters depend on the production state (e.g., production rate, working speed, worked material, worked item), while the failure rate depends on the actual production state and the history of the machine (i.e., the state that the machine has been in and the period the machine has spent in each state). Accordingly, the age of the machine is significantly affected by both the actual and past production states. In this context, the maintenance engineers should decide which is the best maintenance strategy to implement for the machine. Different maintenance policies can be considered and compared such as CM, time-based PM, and CBM. Moreover, for the same maintenance policy different parameters could be selected (e.g., different maintenance interval for a time-based PM). The main objective is to find out the best maintenance policy from economic or operational perspectives by adopting a GPT for DES.

Based on the former considerations, users of the developed algorithm and library should specify as inputs which are the possible production states and their corresponding Weibull failure parameters (i.e., shape, scale, and location). The user should also specify as input how the production state changes over time or which are the events or causes that lead to a change in the production state. It is worth mentioning that the changes of state could be related to a stochastic process. In other words, the production strategy should be specified by the users. Finally, users should specify which are the maintenance policies to be considered and their respective parameters. The main output of the described problem could be one or more economic or operational indicators for each maintenance policy, facilitating the determination of the most suitable one. To evaluate distinct maintenance strategies, it could also be possible to consider production and logistic aspects, along with their associated costs. Indeed, GPT for DES can include the previous aspects in the analysis. If production and logistic aspects are included, the economic indicators will include information related to the overall management of maintenance, production, and logistic activities, providing a more comprehensive point of view and higher flexibility.

The main assumptions of the present problem are as follows.

- The failure behavior of the machine is known, and it is characterized based on the stress level or production state.
- The possible stress levels or production states are known.
- The production parameters could either be controlled or not. In the first case, there is the potential for joint optimization of maintenance and production. Otherwise, in case the production parameters are not controllable, or their sequence is fixed over time, only maintenance activities can be optimized (potentially with the inclusion of logistic activities).

To sum up this section, Table 2 reports the notations of the former variables and parameters, along with additional parameters described in the following sections.

Table 2
Decision variables and parameters characterizing the problem.

Decision Variable	
M_k	k-th maintenance policy
$VarM_k$	Variables of the k-th maintenance policy
Parameter	
P_j	j-th production strategy
SL_i	i-th stress level, or production state, or operating state
β_i	Shape parameter of the i-th stress level
η_i	Scale parameter of the i-th stress level
γ_i	Location parameter of the i-th stress level
$\lambda_i(\cdot)$	Failure rate of the i-th stress level at a time t
h_t	History of the machine till t
$T_{o,i}$	Virtual origin of the i-th NHPP interval
T_{p_i}	Instant of failure of the i-th NHPP interval
T_{c_i}	Final instant of the i-th NHPP
$HPE1$	Homogeneous Poisson Extraction with mean equal to 1
$HPE1_{c_i}$	Consumed life up to T_{c_i}
$HPE1_e$	Consumed life at the beginning of the DNHPP
$HPE1_f$	Consumed life at the end of the DNHPP
$\Lambda(\cdot)$	Cumulative distribution of the failure rate
$F(\cdot)$	Cumulative distribution of TTF
$KPI_{k,j}$	Economic or operational indicator associated with the k-th maintenance under the j-th production strategy

4. Problem formulation: modelling DNHPP by means of discrete-event simulation

DES consists of modelling a system that evolves through events; that is in certain moments the system state changes due to an instantaneous event which can happen because of two reasons.

a) a state change in the system which enables or produces the event.
b) the end of a time period (deterministic or stochastic), which usually represents a process or a task duration.

Between two consecutive events, the system's state remains unchanged.

Such a property of the system model allows for the implementation of a timing mechanism, typically based on a time-sorted calendar of events, where all the events of type b) are scheduled using an appropriate probability distribution. That means that the stochasticity of events is usually modelled using a-priori probability distributions or specific probability distributions based on the past or present state of the system. Thus, the times of future events could not be easily updated in case the state of the system changes. For instance, given a certain stress level at which the system is operating, the simulator schedules a failure time, which may not be modified when the stress level changes. This is the common behavior in simulation software, which usually has no straight functionality to automatically reschedule events in response to system changes. This constitutes a limitation because the representation of the system in the simulation context is far different from its real behavior. This issue encouraged the authors to develop a computational model to cope with this need of rescheduling the events. Then, the model can be implemented in the desired simulation language or software tool.

Let $HPE1$ be a variable belonging to an HPP with a mean equal to 1, i.e., belonging to an exponential distribution, following Eq. (4):

$$HPE1 = -\ln(U) \quad U\epsilon(0,1) \tag{4}$$

As stated by Leemis and Schmeiser (1985), a time arising from an NHPP could be generated by applying the cumulative inverse function of the failure rate to an HPP variable characterized by a mean value of 1, as shown by Eq. (5).

$$t = \Lambda^{-1}(HPE1) \tag{5}$$

where t is a time extracted from an NHPP, while Λ^{-1} denotes the inverse cumulative function of the failure rate, i.e., $\Lambda(t) = \int_0^t \lambda(\tau)d\tau$. For a Weibull distribution, the cumulative function of the failure rate is expressed as shown in Eq. (6) (Razali et al., 2009).

$$\begin{cases} \Lambda(t) = \left(\dfrac{t-\gamma}{\eta}\right)^{\beta} & t \geq \gamma \\ \Lambda(t) = 0 & 0 \leq t < \gamma \end{cases} \tag{6}$$

Consequently, the inverse cumulative function of the failure rate is obtained through Eq. (7).

$$\begin{cases} \Lambda^{-1}(y) = \eta y^{\frac{1}{\beta}} + \gamma = \mathrm{t} & y > 0 \\ \mathrm{non-disclosed} & y = 0 \end{cases} \tag{7}$$

4.1. Generating a Dynamic Non-Homogeneous Poisson Process

GPTs for DES could include a binary DNHPP characterized by a working state (1) and a standby, or non-operating, state (2). In this model, a failure (denoted by 0) could occur only if a given asset is in the working state, as shown in Fig. 1.

As revealed in Fig. 1 there is only one failure function responsible for the transition from a working state to a failure state. Thus, only one Weibull distribution is considered. On the other side, the inverse process depends on the Time To Repair (TTR) distribution. However, this is insufficient to fully represent a production-dependent (or stress level-dependent) failure rate. Indeed, there could be several operating conditions (or working states), each of which is characterized by distinct failure parameters (i.e., distinct scale, shape, and location parameters). Therefore, the actual capabilities of a GPT should be extended to include multiple production states or stress levels, as illustrated in Fig. 2.

The notation $\lambda_i(t, \beta_i, \eta_i, \gamma_i, h_t)$ means that the failure rate of the i-th change of stress level (SL_i) depends on the Weibull parameters related to the i-th stress level (β_i, η_i and γ_i), the time (t) and the history (h_t). The history considers the stress level changes up to t. Indeed, based on the visited operating states and the residence time in each visited operating state, the consumed life changes. For a more comprehensive failure generation, the framework presented in Fig. 2 should also include a stand-by or non-operating state, as visible in Fig. 3.

To incorporate a DNHPP into a GPT for DES, including also a stand-by or non-operating state, the flowchart in Fig. 4 should be implemented.

A brief description of the algorithm is reported below.

Step 1. Set the origin of the NHPP process equal to 0 (see Eq. (8)), and extract HPE1, which is maintained constant for all the DNHPP.

$$T_0 = T_{c_i} = T_{o,i} = 0 \; ; \quad i = 0 \tag{8}$$

Step 2. Given a stress level, characterized by a Weibull distribution with β_i, η_i and γ_i as parameters, generate a TTF based on the inverse cumulative distribution of the failure rate as shown in Eq. (9). Indeed, as previously mentioned, this process is required to randomly generate a failure time. In case the new state is a stand-by one (i.e., $T_{o,i} = -\infty$), $T_{p_{i+1}}$ is set to infinity (i.e., $T_{p_{i+1}} = \infty$). The only assumption is that the location parameter (i.e., γ_i) must be a positive value. This is a fair assumption since the location parameter denotes the failure free life and it is unlikely that a component could fail before the start of the operation.

$$T_{p_{i+1}} = \Lambda_i^{-1}(HPE1) + T_{o,i} = \eta_i \cdot HPE1^{\frac{1}{\beta_i}} + \gamma_i + T_{o,i} \tag{9}$$

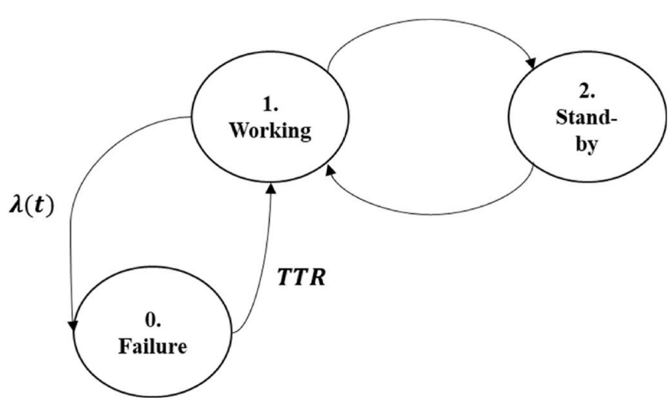

Fig. 1. Distinct states that could usually be represented by GPTs for DES.

Step 3. Increment the counter i by 1. Let T_{c_i} be the time when a stress level transition occurs. There could be two distinct scenarios.

Step 3.1. ($T_{c_i} \geq T_{p_i}$): A failure occurs in T_{p_i} and the process restarts from Step 1.

Step 3.2. ($T_{c_i} < T_{p_i}$ *AND* $T_{p_i} = \infty$): The state changes in T_{c_i}. This condition is required to check if the previous state was stand-by. Particularly, if the previous phase was a non-operating one, then consider the consumed life estimated up to the previous cycle, as shown in Eq. (10). Indeed, if a non-operating state lasts, the failure generation is stopped, and no useful life is consumed. Thus, as soon as a transition to an operating state is observed, the consumed life is set equal to the consumed life of the previous cycle. Next, proceed with Step 4.2.

$$HPE1_{c_i} = HPE1_{c_{i-1}} \tag{10}$$

Step 3.3. ($T_{c_i} < T_{p_i}$ *AND* $T_{p_i}! = \infty$): The state changes in T_{c_i}, moreover, the previous state was an operating one. Thus, estimate the consumed life up to T_{c_i} , as in Eq. (11). Then, proceed with Step 4.

$$HPE1_{c_i} = \Lambda_{i-1}(T_{c_i} - T_{o,i-1}) = \left(\frac{T_{c_i} - T_{o,i-1} - \gamma_{i-1}}{\eta_{i-1}}\right)^{\beta_{i-1}} \tag{11}$$

$HPE1_{c_i}$ is called "Poisson Life consumed" and it could be used to specify imperfect maintenance conditions, among which "as bad as old" (ABAO) maintenance through the introduction of a recovery factor. This ability is better explained in Section 4.3. The consumed life is the equipment's life that has been eroded before the state change. It is required to store this variable to re-schedule the failure considering the previous history.

Step 4. In T_{c_i} the new state could be either a stand-by or an operating state.

Step 4.1. *(Stand-by state)*: Set the new virtual origin equal to minus infinity as shown in Eq. (12). Then iterate from Step 2.

$$T_{o,i} = -\infty \tag{12}$$

Step 4.2. *(Operating state):* Estimate the virtual origin of the new Weibull, based on the consumed life, as expressed in Eq. (13). The new virtual origin takes into account the consumed life up to the state change. Accordingly, the new virtual origin is influenced by what has previously happened regarding state changes.

$$T_{o,i} = T_{c_i} - \Lambda_i^{-1}(HPE1_{c_i}) = T_{c_i} - \eta_i \cdot HPE1_{c_i}^{\frac{1}{\beta_i}} - \gamma_i \tag{13}$$

where $\beta_i, \eta_i, \gamma_i$ denote the parameters related to the Weibull distribution characterizing the new stress level. Then, iterate from Step 2 which allows to re-schedule the failure.

If Step 4.1 is carried out, the new TTF ($T_{p_{i+1}}$) estimated through the subsequent step 2 is equal to ∞ since $T_{o,i}$ is equal to $-\infty$, denoting a stand-by state. Accordingly, during Step 3, $T_{p_i} = \infty$ denotes that the previous state was a stand-by one that consumed no useful life.

4.2. Example of DNHPP

The generation of a DNHPP heavily relies on the capability of re-scheduling failures as soon as a change of the stress level occurs. To better clarify the process of failure re-scheduling through the algorithm of Fig. 4 a simple example is presented in this section, representing on graphs the unreliability vs time, which is easier to interpret compared to the failure rate.

Let $T_0 = 0$ be the start of the operations (Step 1) and SL_0 be the first stress level characterized by three Weibull parameters denoted as β_0, η_0

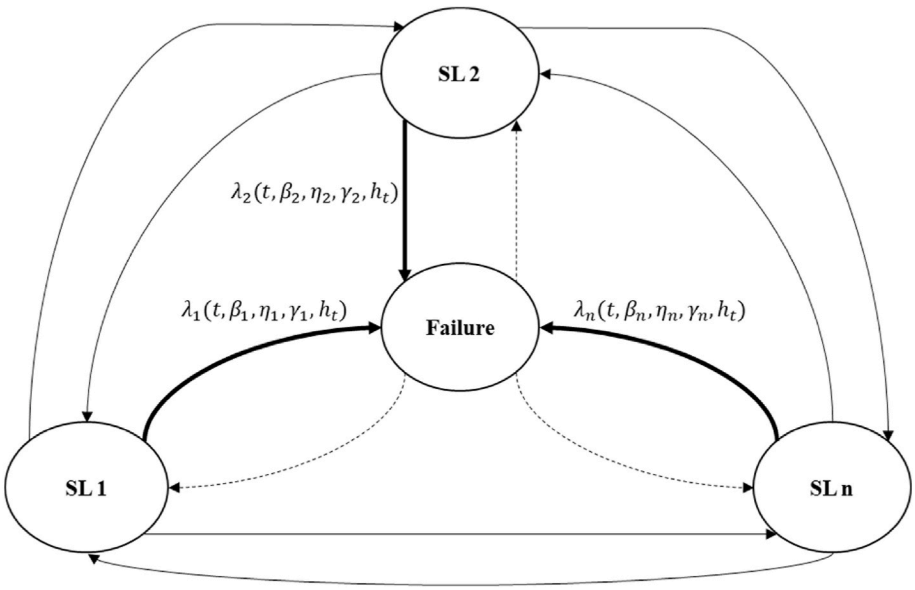

Fig. 2. Multiple production rates or stress levels. The rate associated with the thin dashed black arrow is the MTTR. The failure rates are associated with the thick black arrow. SL stands for Stress Level.

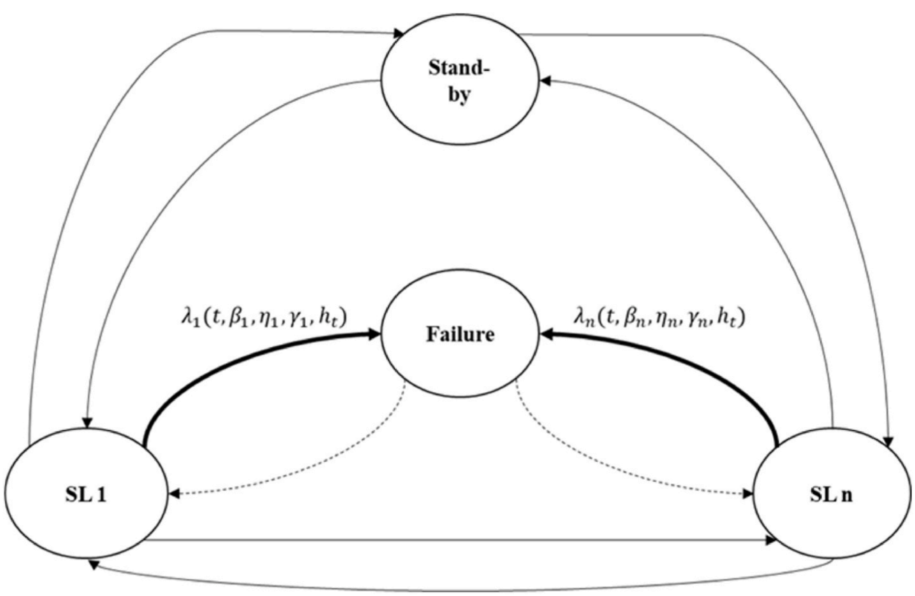

Fig. 3. Practical extension of Fig.2, considering the presence of a stand-by state which does not consume life, thus it characterized by the absence of failures.

and γ_0. Through Eq. (9), a failure is scheduled in T_{p_1} (Step 2), as shown in Fig. 5 on the curve of unreliability, which is expressed as $F_0 = 1 - e^{-\Lambda_0(t)}$.

Assuming that in T_{c_1} there is a transition into a non-operating state (SL_1) and that $T_{c_1} < T_{p_1}$, the stress level changes in T_{c_1}, while the consumed life up to T_{c_1} ($HPE1_{c_1}$) is calculated following Eq. (11) (Step 3.3). Being SL_1 a non-operating state the new virtual origin is set equal to minus infinity as illustrated in Eq. (12) (Step 4.1), then the failure generation is suspended following Step 2. Accordingly, $T_{p_2} = \infty$ as revealed in Fig. 6.

In T_{c_2} the stress level changes into SL_2 which is characterized by the following failure parameters: β_2, η_2 and γ_2. Being that $T_{p_2} = \infty$, T_{c_2} results lower than T_{p_2}, thus a stress level change occurs in T_{c_2}. Furthermore, since SL_1 was a non-operating state, no life has been consumed between T_{c_1} and T_{c_2}, consequently $HPE1_{c_2} = HPE1_{c_1}$ as depicted in Eq. (10) (Step 3.2). Considering the new virtual origin obtained through Eq. (13) (Step 4.2), the failure is re-scheduled in T_{p_3} according to Eq. (9) (Step 2). The third stress level change is represented in Fig. 7.

A third stress level change is seen at T_{c_3}. Given that $T_{c_3} < T_{p_3}$, the stress level successfully changes into SL_3. The consumed life up to T_{c_3} is estimated through Eq. (11) (Step 3.3), along with a new virtual origin (Step 4.2). Subsequently, the failure is re-scheduled based on Eq. (9) (Step 2). The new TTF is denoted as T_{p_4}. Finally, the last stress level change is foreseen at T_{c_4}, however, assuming that $T_{c_4} > T_{p_4}$, a failure occurs in T_{p_4} (Step 3.1). The last stress level transition is shown in Fig. 8.

4.3. *Considering imperfect maintenance within a dynamic Non-Homogenous Poisson Process*

The developed algorithm and library allow to consider imperfect maintenance tasks. When a failure occurs, the entity is associated with

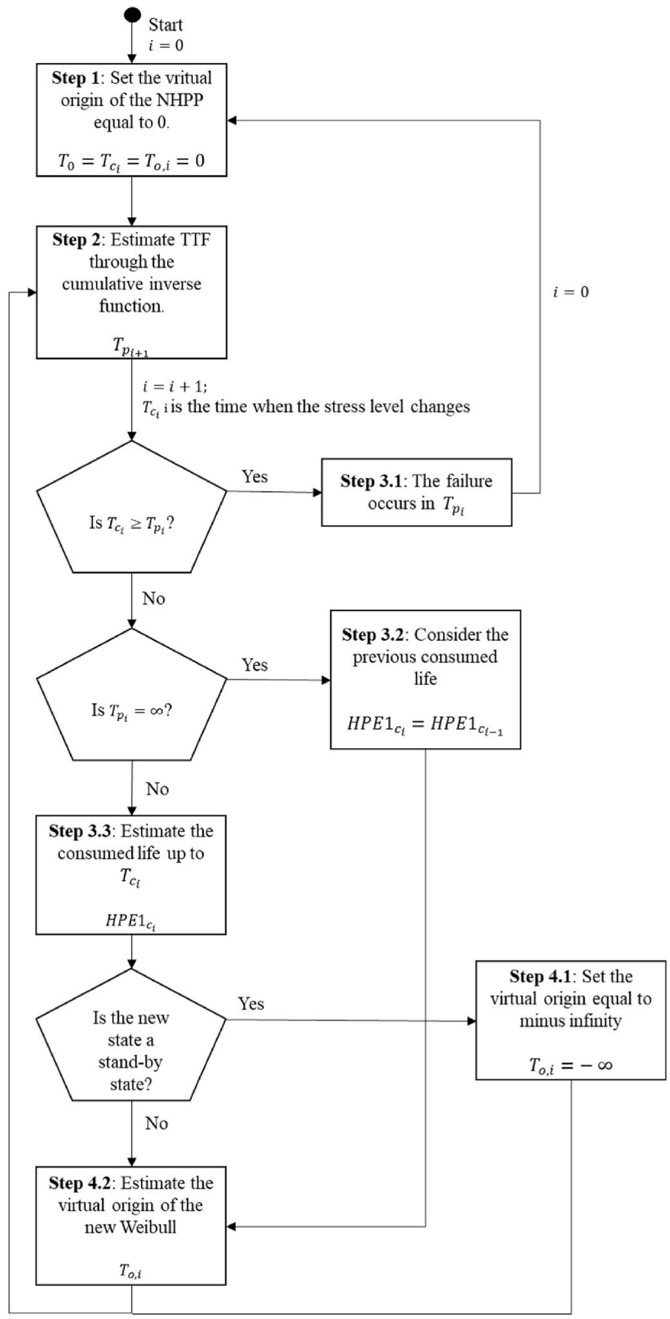

Fig. 4. Required steps to define a DNHPP with a stand-by state through a GPT.

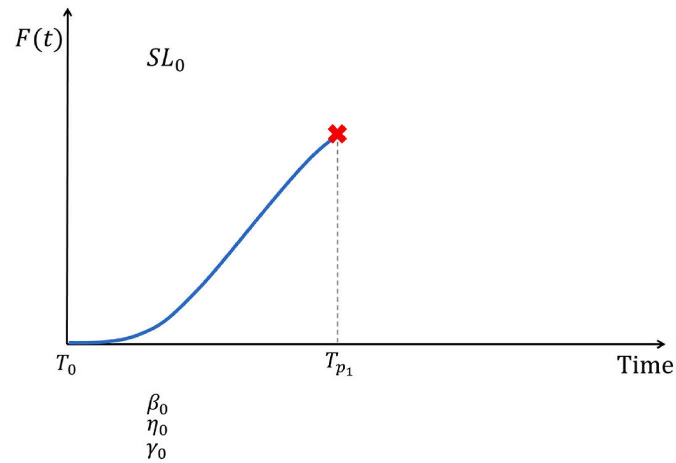

Fig. 5. Unreliability of the first observed stress level and first scheduled failure.

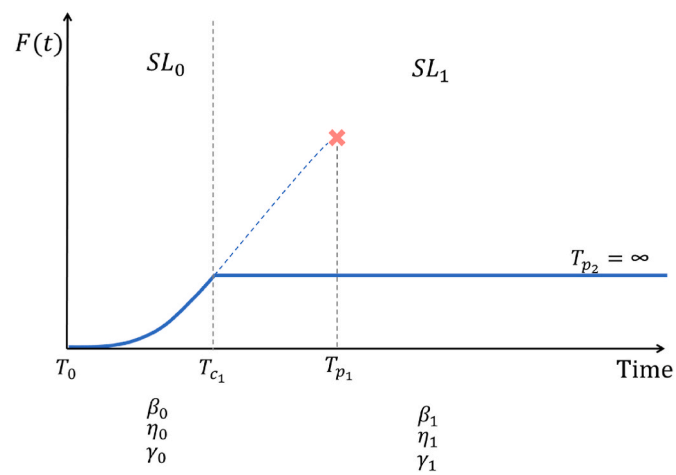

Fig. 6. Unreliability after the transition into a non-operating state and suspended failure.

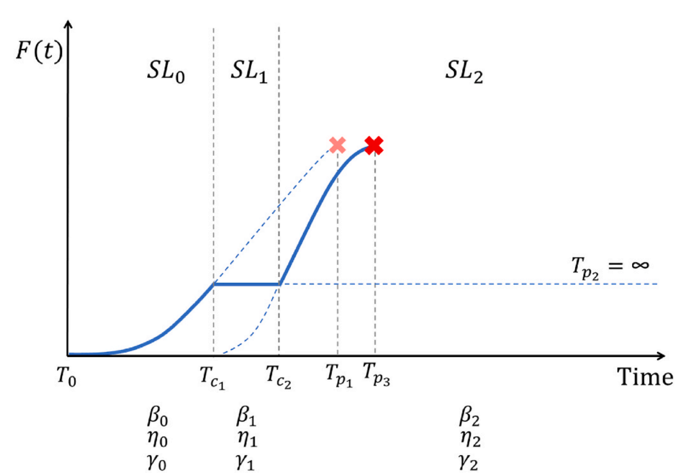

Fig. 7. Unreliability up to the third re-scheduling of the failure.

an attribute that reports its “Poisson Life consumed” up to the failure. Specifically, the “Poisson Life consumed” is estimated through Eq. (11) considering the instant of failure. Furthermore, the library allows for the specification of the initial “Poisson Life consumed” of any entity into a DNHPP process. Accordingly, at first a recovery factor could be defined. The former parameter denotes the portion of life that is restored through maintenance. Next, before the entity enters the DNHPP process, the initial “Poisson Life consumed” could be set equal to the product of the one’s complement of the recovery factor and the “Poisson Life consumed” when the failure occurred. This choice is made following previous works that similarly adopt a factor that restores part of the equipment life (Diallo et al., 2018; Khatab et al., 2018).

Let $HPE1_e$ be the “Poisson Life consumed” when an entity enters a DNHPP process, while $HPE1_f$ be the “Poisson Life consumed” at the moment of failure, an imperfect maintenance task could be introduced according to Eq. (14) or Eq. (15).

$$HPE1_e = HPE1_f * (1 - recovery\ factor) \tag{14}$$

$$HPE1_e = \left(HPE1_f - HPE1_{f-1}\right) * (1 - recovery\ factor) \tag{15}$$

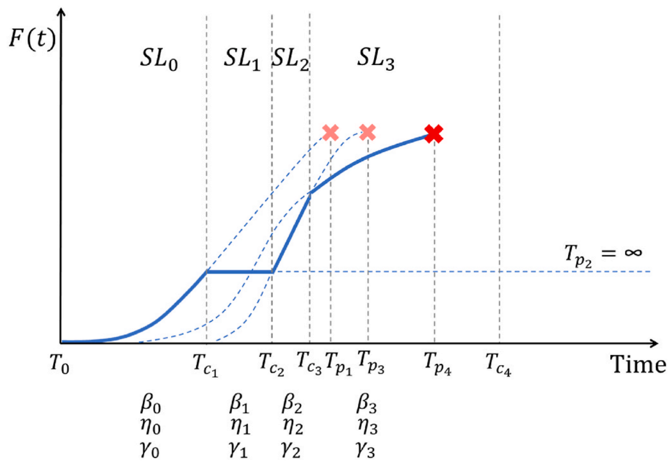

Fig. 8. Unreliability up to the failure occurrence.

where $HPE1_{f-1}$ is the life consumed up to the previous failure. Thus, in case Eq. (15) is adopted a the restore life is just a portion of the time consumed between two subsequent failures. Through Eq. (14), a perfect or "as good as new" (AGAN) maintenance could be specified by setting the recovery factor equal to 1. By contrary, a minimal or ABAO maintenance is defined by adopting a recovery factor equal to 0. Finally, an imperfect maintenance task is obtained through the adoption of a recovery factor that is higher than 0, but lower than 1.

Based on the previous statements, the flowchart of Fig. 4 could be modified to incorporate imperfect maintenance condition as shown in Fig. 9.

According to Fig. 9, when a failure occurs, in case of perfect maintenance the process is iterated from Step 1, and nothing changes compared to the previous case. Otherwise, after the occurrence of a failure, the following steps are required.

Step 5. Estimate the consumed life up to the failure ($HPE1_f$) through Eq. 11

Step 6. Determine the initial consumed life ($HPE1_e$) after the introduction of a recovery factor through Eq. (14) or Eq. (15).

Step 4.2 (Operation restart): When the maintenance ends and the operation is restarted, the virtual origin of the new Weibull is estimated through Eq. (13) based on the initial consumed life ($HPE1_e$). Then iterate from Step 2 and re-schedule the failure after the extraction of a new HPE1, which is rejected in case it is lower than $HPE1_e$.

5. Application of the developed library

A library capable of generating failures based on a DNHPP was created in a Rockwell ARENA© environment. The library is described in this section and shared through GitHub.

To demonstrate the utility and strengths of the developed tool, two simple applications are presented in this section. The first application focuses on a single asset that undergoes a PM cycle, along with the possibility of imperfect maintenance. The second application considers an asset characterized by two failure modes. For clarity, both the applications consider just a single maintenance operator, who is always available. The changes in stress level are fed into the model through an external block representing the production process (or the operations). Thus, the production block is regarded as a black box that gives as an output the current stress level. Finally, the symbols from the Business Process Modelling Notation (BPMN) are used to visualize the two applications. Indeed, the BPMN is a common standard for graphical representation, which is often adopted as a precursor of DES.

Furthermore, this section includes a case study presented as a practical numerical example, illustrating the benefits of the developed library. Specifically, the case study shows how the developed library could be adopted to conduct a sensitivity analysis for optimizing the maintenance plan.

5.1. First application: preventive and imperfect maintenance

The BPMN diagram of the first application is illustrated in Fig. 10.

At first, the asset is created and sent into the DNHPP block for the failure scheduling. Along with the asset, a PM task and a maintenance operator are created. A maintenance interval is defined, while the maintenance operator waits for communication of a PM or CM task. Indeed, if the DNHPP generates a failure before the end of the maintenance interval, then a CM task is required. On the opposite side, if the DNHPP schedules a failure after the maintenance interval, preventive maintenance is performed. Whenever a stress level change occurs before the scheduled failure or the PM task, a "*Stress level variable*" is changed accordingly and the failure is re-scheduled as described in Section 4. Finally, when a CM or PM task is performed, an imperfect of perfect maintenance is specified by introducing a "*Recovery Factor*".

5.2. Second application: multiple failure modes

There could be competing failure modes on the same subsystem, subject to the same level of maintenance (e.g., AGAN), thus a failure could be provoked by multiple causes or deterioration processes. Fig. 11 shows the BPMN representation related to an application of the DNHPP library in case an asset is characterized by two competing failure modes.

Initially, the DNHPP schedules failures for both failure modes. Denoting by TTF_1 and TTF_2 the scheduled failure time for the first and the second failure mode respectively, the TTF is given as $\{TTF_1, TTF_2\}$. Assuming that TTF_1 is lower than TTF_2, the failure will be caused by the first failure mode and will occur after a timespan equal to TTF_1. Once the failure occurs, a "Stress level variable" is exploited to simulate a stand-by operation and suspend the generation of the second failure mode (i.e., TTF_2 is rescheduled and set equal to infinity). Next, after maintenance has been performed to restore the asset to the initial condition and the operation has been restarted, both the failure modes are restored and two new TTFs are generated. It is worth mentioning that TTF_1 represents a complete TTF for the first failure mode and a censored TTF for the second failure mode. Finally, whenever a stress level change occurs, the TTFs of both the failure modes are re-scheduled accordingly. It is also worth noting that the two failure modes could be related to two independent stress levels.

In case the occurrence of a given failure mode does not prevent the other failure mode from occurring (e.g., the production is not stopped, or the failure mode has a dependency on time), the generation of the second failure mode should not be blocked. Finally, just consider that it is possible to restore a single failure mode when it occurs, leaving unchanged the consumed life associated with the other failure mode. The aforementioned scenarios are illustrated in Fig. 12.

5.3. Application of the library to a case study

The case study under consideration is similar to the one depicted in Fig. 10, considering the following assumptions.

- AGAN restoration is considered when a preventive task is performed.
- ABAO restoration is considered when a corrective action is performed after a failure.
- The case study includes a single machine able to process two distinct items or components, each with its own failure behavior.
- The first item's failure behavior follows a Weibull distribution with shape and a scale parameter equal to 4 and 200 h respectively. The first item represents the first stress level (SL_1).

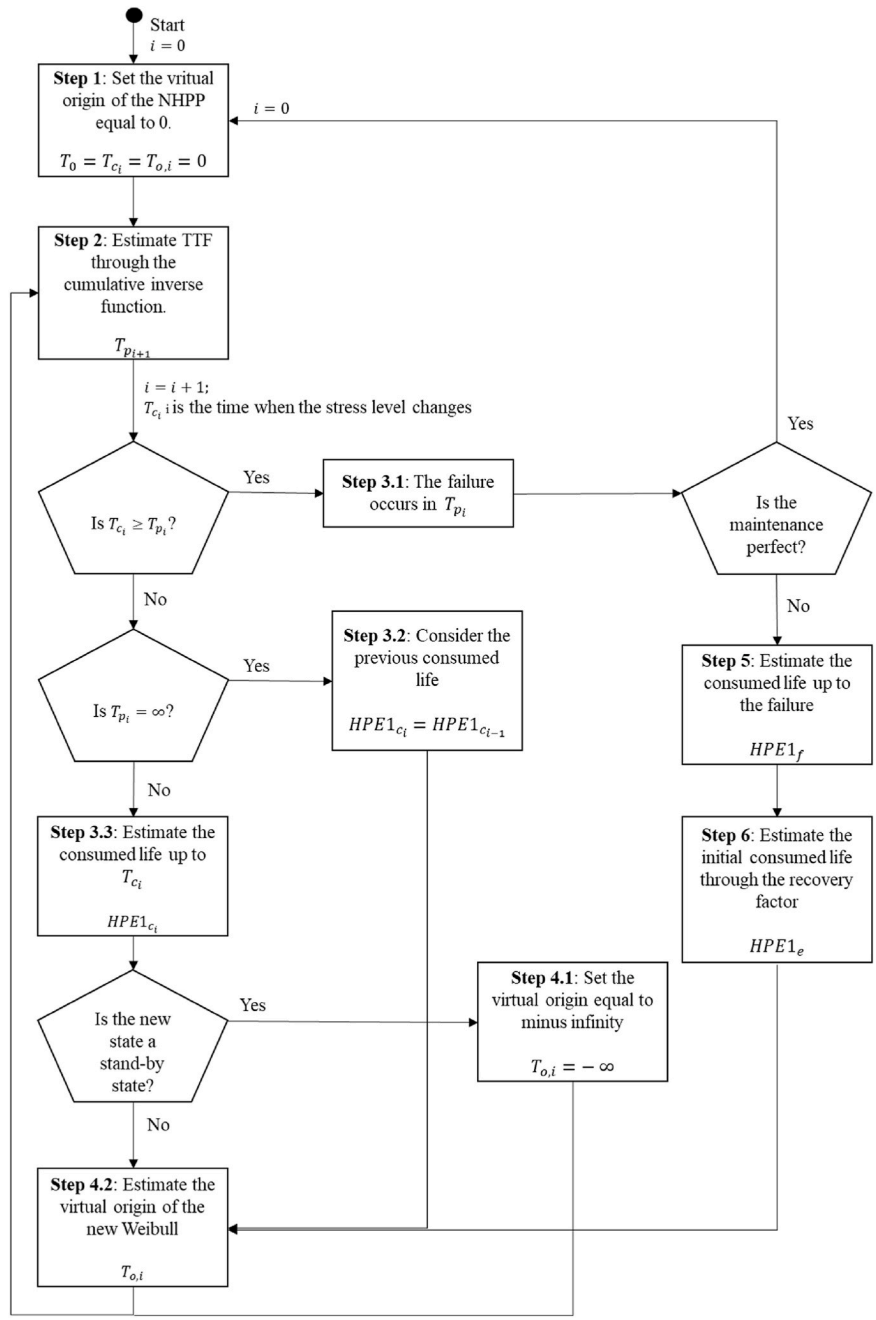

Fig. 9. Required steps to define an DNHPP with a stand-by state and imperfect maintenance through a GPT.

- The second item's failure behavior follows a Weibull distribution with shape and scale parameter equal to 4 and 1000 h respectively. The second item represents the second stress level (SL_2).
- The machine operates 80% of the time on the first item, resulting in a production mix of 80-20.
- The machine is assumed to worked continuously 24/7 all the year.
- There is no setup between one item and the subsequent one, thus, the only downtime is related to preventive maintenance tasks or failures.
- The machine is served by a maintenance squad that is always available.
- The machine undergoes a preventive maintenance task every 400 h. The preventive maintenance tasks are normally distributed with mean equal to 2.5 h and standard deviation of 0.2 h.
- The corrective actions are normally distributed with a mean of and a standard deviation equal to 1.5 and 0.6 h and respectively.
- The former maintenance durations account for possible logistic delay, set-up time, and effective maintenance time.
- The working time of each item is normally distributed with mean equal to 0.5 and standard deviation equal to 0.03 h respectively.
- In case the machine is working an item when the preventive maintenance should be performed, the maintenance squad waits till the end of the working phase.
- Any preventive maintenance task is performed at the end of the maintenance interval, regardless of the number of failures occurred during the interval.
- The occurrence of a failure stops the production, which is restored after the ABAO has been carried out. In this scenario, the machine works the item for the remaining working time at the time of failure. In other words, the time that the machine has worked till the failure

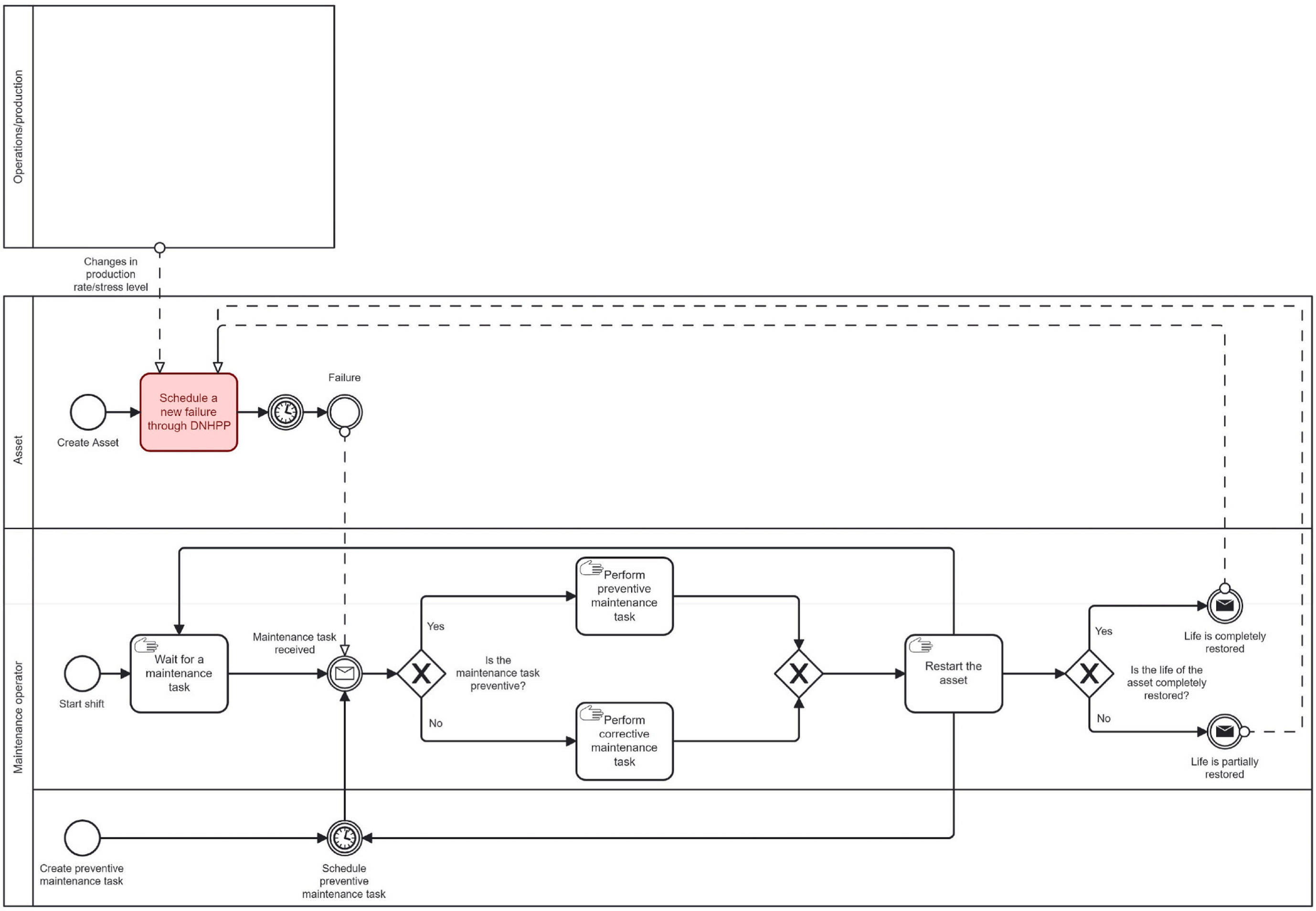

Fig. 10. BPMN representation associated with the first application: preventive and imperfect maintenance. The red block represents the failure scheduling and rescheduling through the DNHPP tool.

is considered as valid, thus, it is not required to start working from scratch.

Accordingly, the machine is subjected to two stress levels (one for each item), with the first stress level being more wearisome for the machine. A practical example of equipment that could represent the case study is a drilling machine that has to work two items made of different materials, where the first material is characterized by higher mechanical properties (e.g., higher hardness).

Based on the context presented above, the investigation is focused on identifying the best combination of preventive maintenance interval and production mix through a DES analysis. The selected parameter to be optimized is the operational availability defined as the ratio between the actual working hours and the total available time. The only source of loss is related to the failures, thus net utilization and operational availability coincide. In other words, the operational availability could be denoted as $KPI_{k,j}$. Due to the complex nature of real applications, it is often challenging to represent the problem through equations or functions. However, the adoption of DES could overcome this limitation, allowing to model the logical-mathematical interactions among events. When this is the case, no function is used for the optimization, but a sensitivity analysis is carried out. Accordingly, a sensitivity analysis is conducted to define the combination of production mix and preventive maintenance interval associated with the highest operational availability. In this context, nine preventive maintenance intervals and seven production mixes are considered. Specifically, the minimum maintenance interval is set at 100 h, while the longest maintenance interval is 900 h. Based on the previous considerations, the considered maintenance policies (M_k) are time-based PMs, each of which is characterized by a single user-defined variable ($VarM_k$), which is the time between two consecutive PMs. On the other hand, the production mix ranges from 80_1-20_2 to 20_1-80_2, where the first number represents the percentage associated with the first item (i.e., the more stressful for the machine), while the second one refers to the produced percentage of the second item. To avoid any confusion, the subscript 1 refers to the percentage associated with the first item, while the subscript 2 is related to the percentage of the second item. The different production mixes represent distinct production strategies (P_j). The simulation is conducted with both the DNHPP library, considering two stress levels (one for each item), and without the DNHPP library (i.e., with a single stress level). For the second scenario, three different scale parameters are considered to model the failure generation: 235, 250, and 280 h. Indeed, since at the beginning the first item is produced more than the second one, the scale parameter in the single stress scenario should be much closer to the scale parameter associated with the first item.

Considering the initial production mix and PM interval, the Welch's method is employed to identify the warm-up period, which was cautiously set at 3000 h. The simulation run is chosen to be more than ten times longer than the warm-up period. Ten simulation runs and four years of simulation (i.e., 35,040 h) for each combination of production mix and maintenance interval are conducted. The verification was conducted by using the animation and debug feature of ARENA. As an additional validation step, one stress level at a time was considered to compare the number of failures arising from the simulation with those

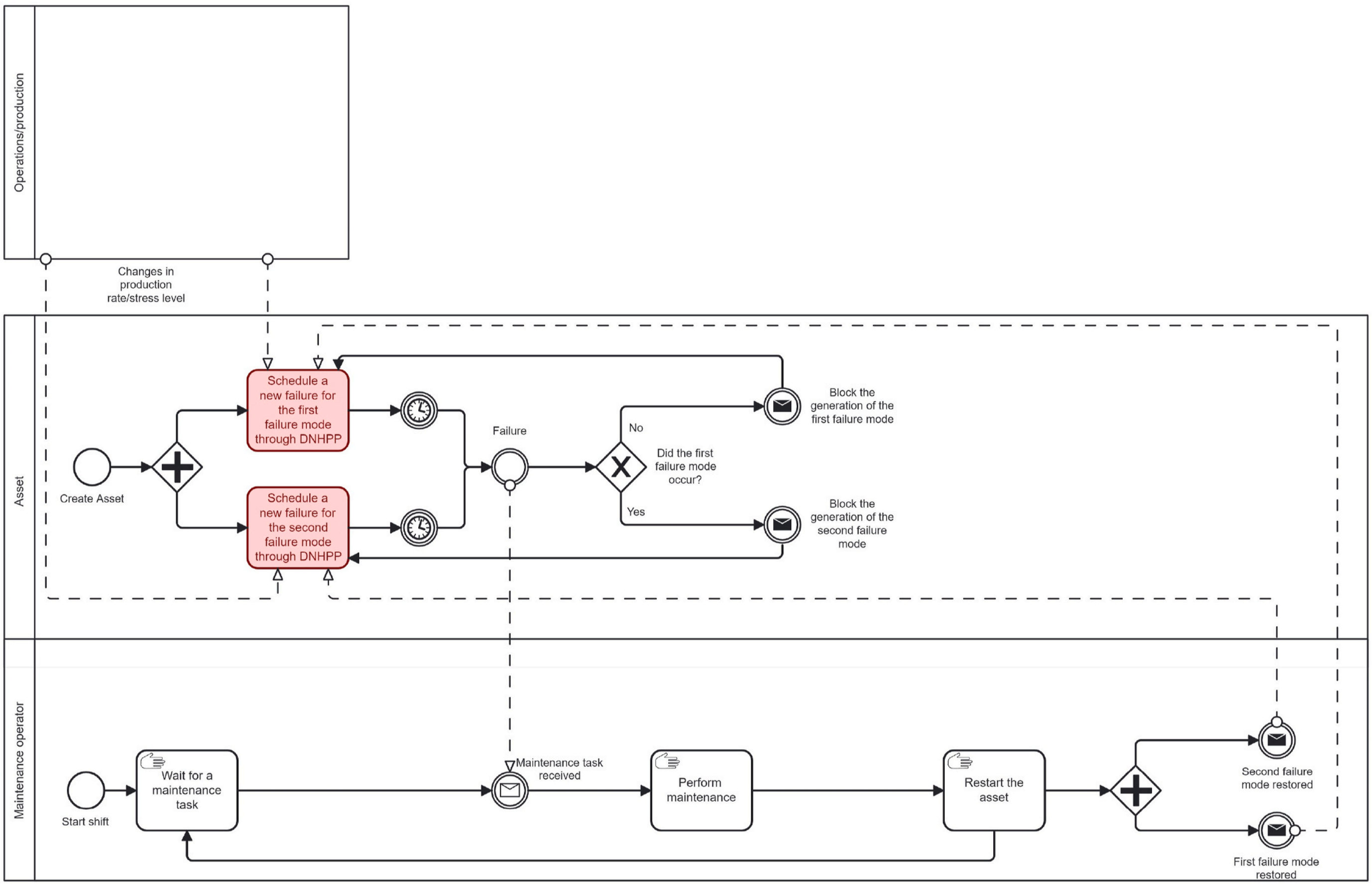

Fig. 11. BPMN diagram of the second application: asset characterized by two failure modes. The red block identifies the failure scheduling and rescheduling through the DNHPP tool.

arising from NHPP under minimal repair assumption. The mean operational availability arising from both the single stress simulation and the double stress modelled through the DNHPP are shown in Table 3 and Fig. 13. The acronym SS stands for single stress, while the number associated with the SS acronym is the scale parameter used for the scenario. In Table 3, the highlighted green cells with bold text represent the optimal preventive maintenance interval for each production mix. To test the differences in operational availability arising from the adoption of distinct PM intervals, a one-way ANOVA with Tukey post-hoc test was conducted for each production mix. Accordingly, nine samples (one for each PM interval) of ten observations (one for each run) were compared for each production mix. The ANOVA depicted that different PM intervals could result in non-statistically different operational availabilities. This condition could also be present for the optimal PM interval. To underline this finding, a given production mix could have more than one green cell representing the optimal PM interval. It follows that, in case more than one green cell is shown for a given production mix, all the related PM intervals could be considered as optimum since no significant statistical difference is found.

As shown in Table 3 and Fig. 13, the results arising from the single stress level with scale parameter of either 235 or 250 h resemble the results arising from the DNHPP application considering the initial production mix (i.e., 80_1-20_2). This fact denotes that the choices related to the scale parameter of the single stress level scenarios could be appropriate. In other words, it is possible to state that the modelling with a single stress level characterized by a scale parameter between 235 and 250 h could be proper to model the initial production mix. On the other hand, adopting a single stress level with a scale parameter of 280 h generates slightly different results compared to the 80_1-20_2 production mix modelled through DNHPP. The maximum operational availability is achieved with a production mix of 20_1-80_2, adopting a maintenance interval between 400 and 600 h. Furthermore, when considering a double streel level through the DNHPP library, the optimum preventive maintenance interval varies based on the production mix. Specifically, the preventive maintenance should be scheduled less frequently for production mixes characterized by higher percentage of the second item (i.e., the maintenance interval should be longer). However, when considering a single stress level, the model is not able to distinguish between the two items. Consequently, the optimum maintenance interval remains equal to 200 h for all the tested single stress scenarios independently from the production mix, denoting a great limitation of the modelling without the double stress level.

6. Discussion

The algorithm presented in this paper enables the generation of production-dependent failures within a DES environment, assuming that the failure behavior under different stress levels follows a three-parameter or two-parameter Weibull distribution. This peculiar feature provides the possibility to conduct simultaneous optimization of production and maintenance planning, instead of optimizing production and subsequently the maintenance plan (Zied et al., 2011). The DES environment offers greater flexibility compared to analytic models, which often rely on strong assumptions, such as deterministic processing or working times (Matta and Simone, 2016). Moreover, other assumptions such as perfect maintenance (uit het Broek et al., 2021) and deterministic deterioration (Martinelli and Piedimonte, 2008) are relaxed.

Previous works introduced the possibility to link the failure rate or the probability of failure to the production rate (Francie et al., 2014; uit

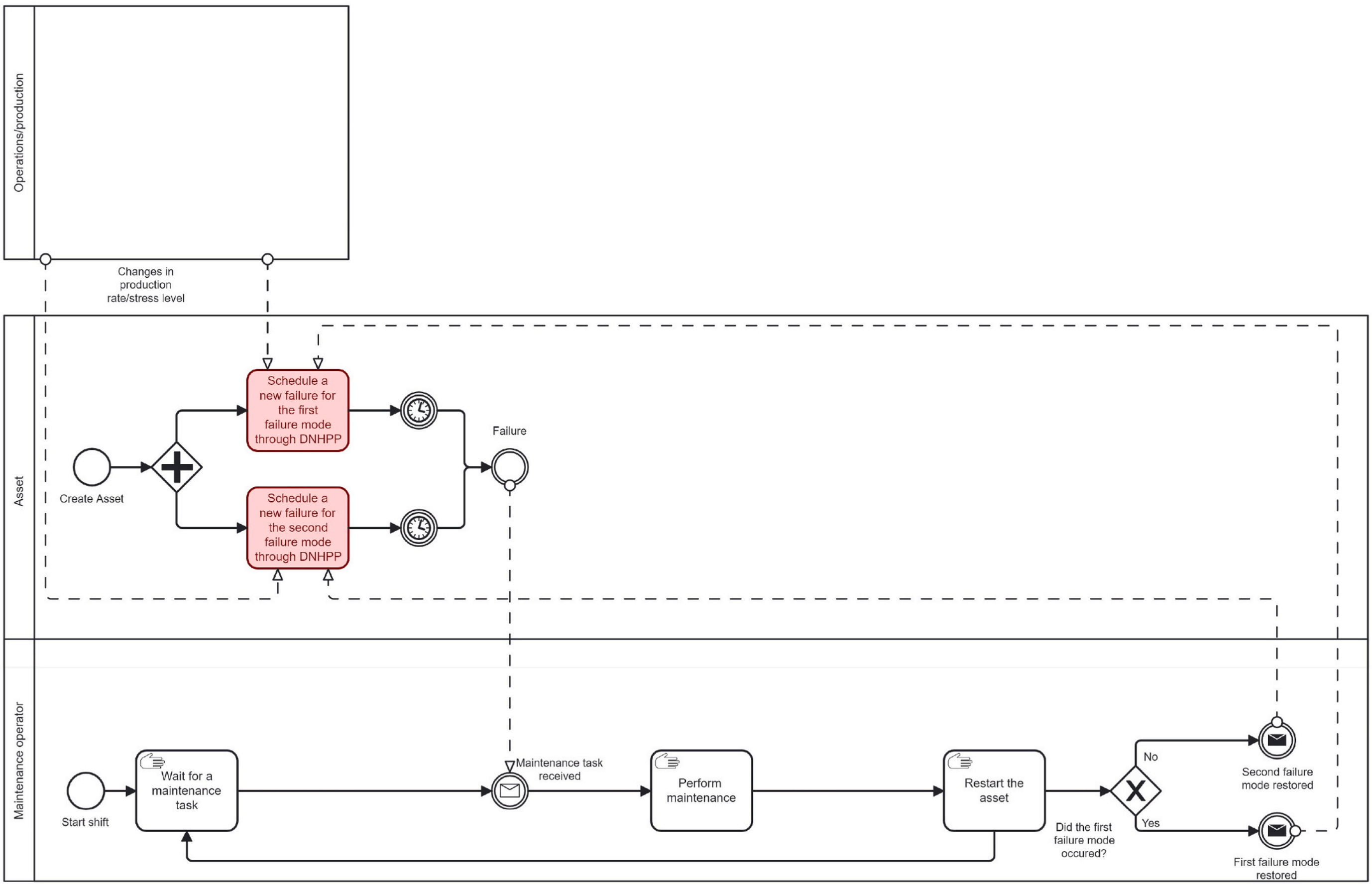

Fig. 12. BPMN diagram of an asset characterized by two failure modes. The occurrence of a failure mode does not prevent the other from occurring, moreover, it is only restored the failure mode that has occurred.

Table 3
Mean operational availability for each combination of preventive maintenance interval and production mix.

		Mean operational availability								
PM Interval (hours)		100	200	300	400	500	600	700	800	900
Production mix	**80_1–20_2**	97.47%	**98.41%**	98.01%	96.72%	94.76%	92.40%	89.39%	86.37%	83.33%
	70_1–30_2	97.48%	**98.50%**	**98.39%**	97.57%	96.15%	94.29%	92.00%	89.38%	86.80%
	60_1–40_2	97.48%	**98.58%**	**98.67%**	98.17%	97.31%	95.98%	94.37%	92.24%	90.12%
	50_1–50_2	97.48%	98.65%	**98.84%**	98.63%	98.07%	97.28%	96.12%	94.72%	93.05%
	40_1–60_2	97.48%	98.70%	**98.99%**	**98.98%**	98.68%	98.21%	97.53%	96.60%	95.59%
	30_1–70_2	97.49%	98.71%	**99.08%**	**99.13%**	**99.09%**	98.84%	98.50%	98.02%	97.40%
	20_1–80_2	97.49%	98.73%	99.13%	**99.28%**	**99.31%**	**99.24%**	99.12%	98.89%	98.60%
	SS-235	97.46%	**98.37%**	97.93%	96.59%	94.68%	92.02%	89.06%	85.96%	82.77%
	SS-250	97.46%	**98.45%**	98.20%	97.20%	95.53%	93.38%	90.71%	88.02%	85.05%
	SS-280	97.48%	**98.54%**	**98.58%**	97.93%	96.78%	95.22%	93.29%	91.04%	88.68%

het Broek et al., 2020) or the cumulative number of items produced (Bouslah et al., 2016, 2018). The proposed model could generate the former kinds of failures in case different levels of production rate are specified (the cumulative number of items produced could be related to the production rate), as they could be seen as different stress levels or production states. Additionally, the developed algorithm could generate failures that depend on other operating conditions, such as the worked item or degradation state. Accordingly, the stress level could also be defined as a combination of production rate and worked item or degradation state, offering a higher level of flexibility. However, for each combination of production rate and worked item or degradation state, proper failure parameters should be specified. Furthermore, in past studies where production or operating dependent failures are considered, the failure generation is usually related to a degradation threshold (uit het Broek et al., 2020) or distinct failure rates are associated with distinct operating conditions or degradation states (Colledani and Tolio, 2012). On the other hand, the proposed model reschedules failures when a stress level change occurs, considering the consumed life up to the stress level change. Finally, the model could handle the generation of competing failure modes as described in Section 5.2.

6.1. Operational availability and downtime cost

Considering the results presented in Section 5.3, it is possible to state that the double stress level modelling introduced through the DNHPP library provides more flexibility compared to the modelling based on a single stress level. Indeed, the DNHPP failure generation is sensitive to variations in production mix. By contrary, the single stress level modelling disregards the changes in production mix. This could possibly

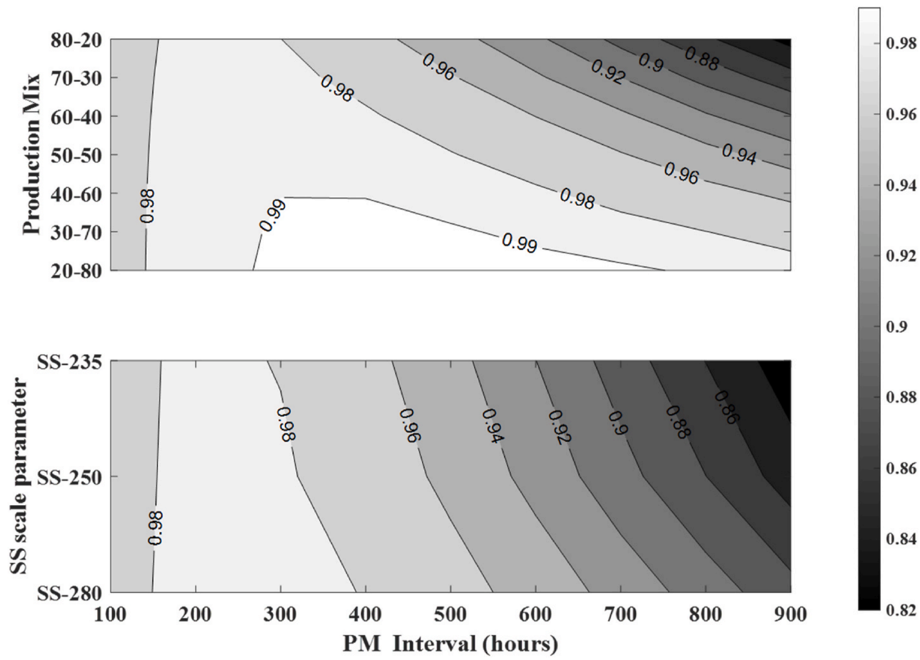

Fig. 13. Contour plot of the mean operational availability for each combination of preventive maintenance interval and production mix.

generate misleading results, especially when the production mix shifts towards a higher percentage of the second item.

To investigate the results even further, it is worth mentioning that the 80_1-20_2, 70_1-30_2, and 60_1-40_2 production mixes are characterized by a maximum operational availability when the preventive maintenance interval is set at 200 h (or also 300 h for 70_1-30_2, and 60_1-40_2 production mixes). Similar outcomes are obtained in case a single stress level modelling is considered. Accordingly, even though the production mix changes, it is possible to state that the variation is not sufficient to influence the optimal maintenance interval. Thus, the single stress level modelling is still effective to evaluate the maintenance strategy to adopt for a production mix in the range of 80_1-20_2 and 60_1-40_2. By contrary, the simulation of the 50_1-50_2 production mix modelled through the DNHPP depicts the optimum value in correspondence to a maintenance interval equal to 300 h. In fact, exploiting a maintenance interval of 200 h (i.e., the optimum value arising from the single stress level) would generate an operational availability which is close to 0.2% lower compared to the optimal operational availability (see Table 3). Furthermore, the difference between the optimal operational availability arising from the single stress level and the double stress level modelling increases for the remaining production. Indeed, the 40_1-60_2 production mix is characterized by an optimum condition when the preventive maintenance is scheduled every 300 or 400 h, which results in an operational availability of 98.98%. Planning preventive maintenance activities every 200 h would generate almost a 0.3% reduction in operational availability for the 40_1-60_2 production mix. Finally, the 30_1-70_2 production mix experiences optimum operational availability in case the preventive maintenance is carried out every 300, 400, or 500 h, while the optimum of the 20_1-80_2 production mix is obtained when the machine is subjected to a maintenance interval between 400 and 600 h. The 30_1-70_2 and 20_1-80_2 production mixes would observe a decrease in the operational availability respectively close to or higher than 0.4% in case 200 h is defined as the maintenance interval.

All the aforementioned considerations can be translated into economic terms by introducing a reference unit downtime cost. Given the unit downtime cost (UDC), the total downtime cost (TDC) during four-year of operations can be obtained as shown in Eq. (16).

$$TDC = 35040 * UDC * (1 - Operational\ Availability) \tag{16}$$

Based on Eq. (16), it is possible to evaluate the TDC when the optimum maintenance interval is adopted. However, as previously mentioned, the optimum maintenance interval is not affected by the production mix when the single stress modelling is considered. Thus, there is a difference between the optimum TDC in case the system is modelled with or without the DNHPP library. Considering 1000 €/hours and 5000 €/hours as unit downtime costs, the differences between the TDCs associated with the optimal maintenance policy arising from the single stress and double stress modellings are listed in Table 4. As a reminder, the optimal maintenance interval for the single stress modelling is equal to 200 h, whereas it ranges between 300 and 600 h for the double stress modelled through the DNHPP library.

It emerged that, in case the UDC is equal to 1000 €/h, the TDC difference ranges between 67,224 € and 178,893 €. The first value is associated with the 50_1-50_2 production mix, while the second one refers to the 20_1-80_2 production mix. Accordingly, modelling the 20_1-80_2 production mix with a single stress level would lead to obtain a TDC higher than one thousand euros compared to the double stress modelling. The situation is even worse when the unit downtime cost is higher, i.e., 5000 €/h. Indeed, scheduling a preventive maintenance every 200 h (single stress optimum) would generate a TDC almost nine hundred thousand euros higher compared to the TDC arising from the adoption of a preventive task every 600 h (one of the double stress optimums). Based on the previous considerations, it is possible to state that considering a single stress model deeply affects the sensitivity analysis, which points out the same optimum independently from the production mix. Thus, the real optimum maintenance interval is not detected, possibly leading to higher downtime costs, which, in turn, could undermine the profits arising from the production.

6.2. Operating outside the optimum

It is worth mentioning that a company could decide to adopt maintenance interval different from the optimum value arising from the simulation. Indeed, the preventive maintenance interval could be regulated by national regulations, or a company could decide to adopt opportunistic maintenance strategies. Considering the results arising from the single stress modelling, it can be seen that moving outside of the optimum would generate quite lower operational availability. As a

Table 4
Differential downtime cost between single stress and double stress optimization for different production mixes.

Production mix	Optimum operational availability		Downtime cost difference	
	Double stress	Single stress	1000 €/h	5000 €/h
50_1–50_2	98.84%	98.65%	67,224	336,121
40_1–60_2	98.98%	98.70%	97,874	489,369
30_1–70_2	99.08%	98.71%	128,341	641,705
20_1–80_2	99.24%	98.73%	178,893	894,466

matter of fact, considering the 80_1-20_2 production mix, the single stress simulation underlines that scheduling a preventive action every 400 h instead of 200 h would lead to a reduction in operational availability close to 2%. Accordingly, the adoption of opportunistic maintenance could be discouraged or hindered by this finding. Moreover, the finding is independent from the production mix. On the other hand, from the DNHPP double stress-based simulation, it emerged that moving outside the optimum interval is not very impactful for production mix shifted towards the second item (e.g., 30_1-70_2). Indeed, as depicted by Fig. 13, the areas associated with an operational availability higher than 99% and 98% are much broader for the aforementioned production mixes. For instance, considering the 20_1-80_2 production mix, an operational availability lower than 99% is obtained with a maintenance interval lower than 300 h or higher than 700 h. Furthermore, the differences between the operational availability associated with the optimal maintenance interval and the operational availabilities outside the optimum are much lower compared to the ones obtained with the single stress modelling. Consequently, these findings could be helpful to integrate the maintenance activities more effectively with the other needs of companies.

7. Conclusions

In the process of simulating maintenance activities through a GPT, developing an appropriate model for generating failures could be regarded as a significant challenge. To this end, this paper presents an algorithm capable of generating failures based on an NHPP whose parameters vary according to the stress level (e.g., production rate), i.e., a DNHPP. The aforementioned model schedules failure events by extracting a random TTF through the inverse of the cumulative distribution of the failure rate. Moreover, when the stress level changes, the failure is re-scheduled or suspended based on the new stress level. To ease the transition from theory to practice, this study also introduces an Arena library for generating a DNHPP.

Based on the previous considerations, the main theoretical outcome of the present paper is the development of an algorithm capable of modeling production-dependent failures in a DES environment. As a matter of fact, considering production-dependent failure rate is a popular assumption outside the field of DES. Being able to incorporate this failure behavior into DES simulations allows to relax the usual strong assumptions of mathematical models, while including multiple factors typical of real operating environments (e.g., multiple resources, multiple competing or independent failure modes, and imperfect maintenance). From a managerial perspective, the proposed algorithm and associated library allow to define more realistic simulation models, serving as essential tools for decision-making processes related to maintenance activities. For instance, the library could be used to conduct comparison or optimization analyses using a general-purpose DES. Users are provided with a wide range of options to model different equipment and failure behaviors, without the need to implement any complex modelling or coding. Furthermore, the GPTs allow users to model other aspects related to maintenance such as production and logistic activities. Thus, managers could adopt the library for a simulation of the entire production environment and integrate maintenance and production decisions (e.g., production mix) to optimize a performance indicator (e.g., operational availability).

Furthermore, a practical case study is presented to demonstrate the advantages and benefits arising from the adoption of the DNHPP library compared to the traditional single stress modelling. Indeed, the single stress modelling may indicate optimum maintenance intervals different from the ones that are obtain from the simulation of multiple stresses. Accordingly, the sensitivity or optimization analysis could result in operating outside the real optimum, generating higher costs and lower profits. In other words, the developed algorithm and tool could assist managers during the decision-making process related to maintenance and production activities, possibly leading to cost savings or higher operational efficiency.

On the other hand, even though the library is designed to be user-friendly and is accompanied by instructions, there may be a learning curve involved in fully understand its functionality. However, the library comes with a manual, which we plan to progressively extend with useful example of applications. Moreover, it could be difficult to determine the failure parameters related to distinct production rates or operating conditions. Accordingly, a robust reliability analysis is required to determine the parameters of the simulation. In this regard, a possible future development could involve the definition of advanced approaches to estimate failure parameters under variable working conditions. In addition, even though the library is capable of modelling the 3-parameter Weibull distribution, the effect of the location parameters on the failure rescheduling is still under investigation. Thus, it is recommended to consider a 2-parameter Weibull distribution. Furthermore, the current version of the only compatible with Rockwell ARENA©. Thus, we are planning to develop the library also for other open-source DES environments such as AnyLogic or WITNESS. Developing the library for other open-source DES tools could be helpful to extend its usage since there could be organizations where only one tool is available for DES purposes. Finally, it is crucial to define other libraries that allow to create more realistic simulation models within the context of maintenance optimization and scheduling, especially for the most popular maintenance policies (e.g., CBM). Therefore, another future development could be the development of an open-source library that allows to model the degradation parameters based on the working condition and the working history. In other words, a future development could be the implementation of a production-dependent condition monitoring. The former library could be useful to optimize CBM policies.

Data availability

The ARENA library is shared through GitHub

Appendix

Nomenclature	
Acronym	*Extended name*
ABAO	As bad as old
AGAN	As good as new
AI	Artificial intelligence

(*continued on next page*)

(*continued*)

BPMN	Business process modelling and notation
CBM	Condition-based maintenance
CM	Corrective maintenance
DES	Discrete-event simulation
DNHPP	Dynamic Non-Homogeneous Poisson Process
GPT	General purpose tool
HPP	Homogeneous Poisson Process
NHPP	Non-Homogeneous Poisson Process
PDF	Probability Density Function
PLP	Power law process
PM	Preventive maintenance
ROF	Rate of occurrence of failure
SPT	Special purpose tool
TTF	Time to failure
TTR	Time to repair

References

Ait-El-Cadi, A., Gharbi, A., Dhouib, K., Artiba, A., 2021. Integrated production, maintenance and quality control policy for unreliable manufacturing systems under dynamic inspection. Int. J. Prod. Econ. 236, 108140.

Akl, A.M., El Sawah, S., Chakrabortty, R.K., Turan, H.H., 2022. A joint optimization of strategic workforce planning and preventive maintenance scheduling: a simulation–optimization approach. Reliab. Eng. Syst. Saf. 219, 108175.

Alabdulkarim, A.A., Ball, P.D., Tiwari, A., 2013. Applications of simulation in maintenance research. World J. Model. Simulat. 9 (1), 14–37.

Aliunir, F., Zagloel, T.Y.M., Ardi, R., 2020. Discrete-event simulation and optimization of spare parts inventory and preventive maintenance integration model considering cooling down and machine dismantling time factor. Joint Journal of Novel Carbon Resource Sciences & Green Asia Strategy 7 (1), 79–85.

Alqahtani, A.Y., Gupta, S.M., 2017a. Warranty and preventive maintenance analysis for sustainable reverse supply chains. Journal of Management Science and Engineering 2 (1), 69–94.

Alqahtani, A., Gupta, S.M., 2017b. Optimizing two-dimensional renewable warranty policies for sensor embedded remanufactured products. J. Ind. Eng. Manag. 10 (2), 145–187.

Alqahtani, A.Y., Gupta, S.M., Nakashima, K., 2019. Warranty and maintenance analysis of sensor embedded products using internet of things in industry 4.0. Int. J. Prod. Econ. 208, 483–499.

Alrabghi, A., 2020. Modelling condition-based maintenance for nonidentical two-component systems considering four types of dependencies. J. Qual. Mainten. Eng. 27 (1), 169–185.

Alrabghi, A., Tiwari, A., 2013. A Review of Simulation-Based Optimisation in Maintenance Operations. Paper presented at the 2013 UKSim 15th international conference on computer modelling and simulation.

Alrabghi, A., Tiwari, A., 2015. State of the art in simulation-based optimisation for maintenance systems. Comput. Ind. Eng. 82, 167–182.

Alrabghi, A., Tiwari, A., 2016. A novel approach for modelling complex maintenance systems using discrete event simulation. Reliab. Eng. Syst. Saf. 154, 160–170.

Alrabghi, A., Tiwari, A., Savill, M., 2017. Simulation-based optimisation of maintenance systems: industrial case studies. J. Manuf. Syst. 44, 191–206.

Assid, M., Gharbi, A., Hajji, A., 2023. Integrated control policies of production, returns' replenishment and inspection for unreliable hybrid manufacturing-remanufacturing systems with a quality constraint. Comput. Ind. Eng., 109000

Azevedo, R.V., das Chagas Moura, M., Lins, I.D., Droguett, E.L., 2020. A multi-objective approach for solving a replacement policy problem for equipment subject to imperfect repairs. Appl. Math. Model. 86, 1–19.

BahooToroody, A., Abaei, M.M., BahooToroody, F., De Carlo, F., Abbassi, R., Khalaj, S., 2019. A condition monitoring based signal filtering approach for dynamic time dependent safety assessment of natural gas distribution process. Process Saf. Environ. Protect. 123, 335–343.

Blas, M., Alvarez, G., Sarli, J., 2023. Improving the optimization of electric power systems through a discrete event based simulation model. J. Appl. Res. Technol. 21, 17–35.

Bouslah, B., Gharbi, A., Pellerin, R., 2016a. Integrated production, sampling quality control and maintenance of deteriorating production systems with AOQL constraint. Omega 61, 110–126.

Bouslah, B., Gharbi, A., Pellerin, R., 2016b. Joint economic design of production, continuous sampling inspection and preventive maintenance of a deteriorating production system. Int. J. Prod. Econ. 173, 184–198.

Bouslah, B., Gharbi, A., Pellerin, R., 2018. Joint production, quality and maintenance control of a two-machine line subject to operation-dependent and quality-dependent failures. Int. J. Prod. Econ. 195, 210–226.

Cacereño, A., Greiner, D., Galván, B.J., 2021. Multi-objective optimum design and maintenance of safety systems: an in-depth comparison study including encoding and scheduling aspects with NSGA-II. Mathematics 9 (15), 1751.

Celen, M., Djurdjanovic, D., 2012. Operation-dependent maintenance scheduling in flexible manufacturing systems. CIRP Journal of Manufacturing Science and Technology 5 (4), 296–308.

Colledani, M., Tolio, T., 2011. Joint design of quality and production control in manufacturing systems. CIRP Journal of Manufacturing Science and Technology 4 (3), 281–289.

Colledani, M., Tolio, T., 2012. Integrated quality, production logistics and maintenance analysis of multi-stage asynchronous manufacturing systems with degrading machines. CIRP annals 61 (1), 455–458.

Davari, A., Ganji, M., Sajadi, S.M., 2022. An integrated simulation-fuzzy model for preventive maintenance optimisation in multi-product production firms. J. Simulat. 16 (4), 374–391.

Darmawan, A., Sheu, D.D., 2021. Preventive maintenance scheduling: a simulation-optimization approach. Production & Manufacturing Research 9 (1), 281–298.

de Santana, J.M.M., Santiago, R.L.V., Moura, M.d.C., Lins, I.D., 2018. Extended warranty of medical equipment subject to imperfect repairs: an approach based on generalized renewal process and Stackelberg game. Eksploatacja i Niezawodność 20 (4), 567–578.

Diallo, C., Venkatadri, U., Khatab, A., Liu, Z., 2018. Optimal selective maintenance decisions for large serial k-out-of-n: G systems under imperfect maintenance. Reliab. Eng. Syst. Saf. 175, 234–245.

Dias, L.M., Vieira, A.A., Pereira, G.A., Oliveira, J.A., 2016. Discrete Simulation Software Ranking—A Top List of the Worldwide Most Popular and Used Tools. Paper presented at the 2016 Winter Simulation Conference (WSC).

Dursun, İ., Akçay, A., van Houtum, G.-J., 2022. Data pooling for multiple single-component systems under population heterogeneity. Int. J. Prod. Econ. 250, 108665.

Fauadi, M.M., Damanhuri, A.M., Yahaya, S.H., Noor, A.M., Hao, X., 2022. AGENT-BASED discrete event simulation–system dynamics approach to optimize manufacturing system with maintenance activities. J. Adv. Manuf. Technol. 16.

Florian, E., Sgarbossa, F., Zennaro, I., 2021. Machine learning-based predictive maintenance: a cost-oriented model for implementation. Int. J. Prod. Econ. 236, 108114.

Francie, K.A., Jean-Pierre, K., Pierre, D., Victor, S., Vladimir, P., 2014. Stochastic optimal control of manufacturing systems under production-dependent failure rates. Int. J. Prod. Econ. 150, 174–187.

Golbasi, O., Turan, M.O., 2020. A discrete-event simulation algorithm for the optimization of multi-scenario maintenance policies. Comput. Ind. Eng. 145, 106514.

Goti, A., Oyarbide-Zubillaga, A., Sanchez, A., Akyazi, T., Alberdi, E., 2019. Multi equipment condition based maintenance optimization using multi-objective evolutionary algorithms. Appl. Sci. 9 (22), 4849.

Havinga, M.J., de Jonge, B., 2020. Condition-based maintenance in the cyclic patrolling repairman problem. Int. J. Prod. Econ. 222, 107497.

Hung, Y.-H., Li, L.Y., Cheng, T.C.E., 2022. Uncovering hidden capacity in overall equipment effectiveness management. Int. J. Prod. Econ. 248, 108494.

Jiang, R., 2013. A new bathtub curve model with a finite support. Reliab. Eng. Syst. Saf. 119, 44–51.

Kelly, D.L., Smith, C.L., 2009. Bayesian inference in probabilistic risk assessment—the current state of the art. Reliab. Eng. Syst. Saf. 94, 628–643.

Khatab, A., Diallo, C., Venkatadri, U., Liu, Z., Aghezzaf, E.-H., 2018. Optimization of the joint selective maintenance and repairperson assignment problem under imperfect maintenance. Comput. Ind. Eng. 125, 413–422.

Lahiani, N., El Mhamedi, A., Hani, Y., Triki, A., 2016. A novel improving method of industrial performance based on human resources management. IFAC-PapersOnLine 49 (28), 262–267.

Leemis, L., Schmeiser, B., 1985. Random variate generation for Monte Carlo experiments. IEEE Trans. Reliab. 34 (1), 81–85.

Linnéusson, G., Ng, A.H., Aslam, T., 2020. A hybrid simulation-based optimization framework supporting strategic maintenance development to improve production performance. Eur. J. Oper. Res. 281 (2), 402–414.

Marsaro, M.F., Cavalcante, C.A.V., 2017. Random preventive maintenance policy based on inspection for a multicomponent system using simulation. Eksploatacja i Niezawodność 19 (4), 552–559.

Martinelli, F., Piedimonte, F., 2008. Optimal cycle production of a manufacturing system subject to deterioration. Automatica 44 (9), 2388–2391.

Matta, A., Simone, F., 2016. Analysis of two-machine lines with finite buffer, operation-dependent and time-dependent failure modes. Int. J. Prod. Res. 54, 1850–1862.

Meissner, R., Rahn, A., Wicke, K., 2021. Developing prescriptive maintenance strategies in the aviation industry based on a discrete-event simulation framework for post-prognostics decision making. Reliab. Eng. Syst. Saf. 214, 107812.
Mena, R., Viveros, P., Zio, E., Campos, S., 2021. An optimization framework for opportunistic planning of preventive maintenance activities. Reliab. Eng. Syst. Saf. 215, 107801.
Mwanza, J., Telukdarie, A., Igusa, T., 2023. Optimising maintenance workflows in healthcare facilities: a multi-scenario discrete event simulation and simulation annealing approach. Modelling 4, 224–250.
Nili, M.H., Taghaddos, H., Zahraie, B., 2021. Integrating discrete event simulation and genetic algorithm optimization for bridge maintenance planning. Autom. ConStruct. 122, 103513.
Omoleye, T.J., Alabdulkarim, A.A., Tsui, K.L., 2019. Impact of resources and monitoring effectiveness on prognostics enabled condition based maintenance policy. J. Simulat. 13 (4), 254–271.
Orlov, S.P., Susarev, S.V., Uchaikin, R.A., 2021. Application of hierarchical colored petri nets for technological facilities' maintenance process evaluation. Appl. Sci. 11 (11), 5100.
Ouaret, S., Kenné, J.-P., Gharbi, A., Polotski, V., 2017. Age-dependent production and replacement strategies in failure-prone manufacturing systems. Proc. IME B J. Eng. Manufact. 231 (3), 540–554.
Peng, S., Jiang, W., Wei, L., Wang, X.-L., 2022. A new cost-sharing preventive maintenance program under two-dimensional warranty. Int. J. Prod. Econ. 254, 108580.
Razali, A.M., Salih, A.A., Mahdi, A.A., 2009. Estimation accuracy of Weibull distribution parameters. J. Appl. Sci. Res. 5, 790–795.
Savolainen, J., Urbani, M., 2021. Maintenance optimization for a multi-unit system with digital twin simulation. J. Intell. Manuf. 32 (7), 1953–1973.
Sharma, A., Yadava, G., Deshmukh, S., 2011. A literature review and future perspectives on maintenance optimization. J. Qual. Mainten. Eng. 17 (1), 5–25.
Tan, S., Hwang, J., Ab-Samat, H., 2019. WITNESS Simulation of Preventive and Corrective Maintenance for Surface Mounted Technology (SMT) Line. IOP Conference Series: Materials Science and Engineering. Paper presented at the.
Triska, Y., Agostino, Í.R.S., Penna, P.M., Braghirolli, L.F., Frazzon, E.M., 2021. Integrated production and maintenance planning method with simulation-based optimization. IFAC-PapersOnLine 54 (1), 349–354.
Turan, H.H., Atmis, M., Kosanoglu, F., Elsawah, S., Ryan, M.J., 2020. A Risk-Averse Simulation-Based Approach for a Joint Optimization of Workforce Capacity, Spare Part Stocks and Scheduling Priorities in Maintenance Planning, vol. 204. Reliability Engineering & System Safety, 107199.
Turan, M.O., Golbasi, O., 2019. Discrete-Event Simulation of a Maintenance Policy with Multiple Scenarios. International Symposium on Mine Planning & Equipment Selection. Paper presented at the.
Ugurlu, O.F., Ozdemir, B., Kumral, M., 2020. Drilling Performance Management through Reliability-Based Optimization Engineering Assets and Public Infrastructures in the Age of Digitalization. Springer, pp. 23–29.
uit het Broek, M.A.J., Teunter, R.H., de Jonge, B., Veldman, J., Van Foreest, N.D., 2020. Condition-based production planning: adjusting production rates to balance output and failure risk. Manuf. Serv. Oper. Manag. 22 (4), 792–811.
uit het Broek, M.A.J., Teunter, R.H., De Jonge, B., Veldman, J., 2021. Joint condition-based maintenance and condition-based production optimization. Reliab. Eng. Syst. Saf. 214, 107743.
Velasquez, N., Anani, A., Munoz-Gama, J., Pascual, R., 2023. Towards the application of process mining in the mining industry—an LHD maintenance process optimization case study. Sustainability 15, 7974.
Wais, P., 2017. Two and three-parameter Weibull distribution in available wind power analysis. Renew. Energy 103, 15–29.
Wakiru, J., Pintelon, L., Muchiri, P.N., Chemweno, P., 2018. Maintenance optimization: application of remanufacturing and repair strategies. Procedia CIRP 69, 899–904.
Wakiru, J., Pintelon, L., Muchiri, P.N., Chemweno, P.K., Mburu, S., 2020. Towards an Innovative Lubricant Condition Monitoring Strategy for Maintenance of Ageing Multi-Unit Systems, vol. 204. Reliability Engineering & System Safety, 107200.
Wang, K., Djurdjanovic, D., 2018. Joint optimization of preventive maintenance, spare parts inventory and transportation options for systems of geographically distributed assets. Machines 6 (4), 55.
Xie, M., Lai, C.D., 1996. Reliability analysis using an additive Weibull model with bathtub-shaped failure rate function. Reliab. Eng. Syst. Saf. 52 (1), 87–93.
Zhang, T., Xie, M., 2007. Failure data analysis with extended Weibull distribution. Commun. Stat. Simulat. Comput. 36 (3), 579–592.
Zied, H., Sofiene, D., Nidhal, R., 2011. Optimal integrated maintenance/production policy for randomly failing systems with variable failure rate. Int. J. Prod. Res. 49 (19), 5695–5712.