COMPACT RELIABILITY AND MAINTENANCE MODELING OF COMPLEX REPAIRABLE SYSTEMS

A Thesis
Presented to
The Academic Faculty

by

René Valenzuela

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Aerospace Engineering

Georgia Institute of Technology
May 2014

COMPACT RELIABILITY AND MAINTENANCE MODELING OF COMPLEX REPAIRABLE SYSTEMS

Approved by:

Professor Vitali Volovoi,
Committee Chair
School of Aerospace Engineering
Georgia Institute of Technology

Professor Vitali Volovoi, Advisor
School of Aerospace Engineering
Georgia Institute of Technology

Professor Joseph Saleh
School of Aerospace Engineering
Georgia Institute of Technology

Professor David Goldsman
School of Industrial and Systems Engineering
Georgia Institute of Technology

Professor Eric Feron
Schoool of Aerospace Engineering
Georgia Institute of Technology

Dr. Pierre Dersin
RAM Center of Excellence
Alstom TIS

Dr. Fred Villeneuve
Advanced Design Methods
Siemens Energy

Date Approved: March 20th 2013

To Carola, Sofia,

and my parents,

you were my best cheerleaders

ACKNOWLEDGEMENTS

I wish to thank my committee members who were more than generous with their expertise and precious time. A special thanks to my advisor Dr. Vitali Volovoi, for his countless hours of reflecting, reading, encouraging, and most of all patience throughout the entire process. Thank you Dr. Eric Feron, Dr. Joseph Saleh, Dr. David Goldsman, Dr. Pierre Dersin, and Dr. Fred Villeneuve for agreeing to serve on my committee.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

This thesis introduces a framework for modeling maintenance processes of complex engineering systems. The framework explicitly accounts for competing risk failure modes and opportunistic maintenance. A gas turbine example is used to illustrate the methodology presented. The developed framework is intended for a wide range of applications including power plants, aircraft and train systems.

The research fills a gap in the current modeling capabilities by: (1) extending the size of the engineering systems that can be analyzed from the practical perspective and (2) explicitly including dependence effects due to competing risk.

The methodology is based on the following two ideas:

1. A complex system can be decomposed into a set of smaller, less complex systems (components). The maintenance policy of the system must include the simultaneous consideration of the maintenance of all of these components.

2. The timing of the state changes of the rest of the system that have an effect on a given component can be accurately captured by a simple parametric probability distribution.

Based on this, it is postulated that the model of a complex system can be created as the union of component-based models in each of which the timing of the effect of the rest of the system on the component is captured by a simple parametric distribution.

The proposed component-based modeling approach is introduced and validated for both a low and medium complexity model by comparison with a benchmark model of the same system and then it is applied to a maintainability study of a high complexity system (gas turbine engine).

1.1 Motivation

Although reliability has a broad meaning in our daily life, in technical terms it has been given a precise definition. Reliability is a characteristic of an item expressed by the probability that the item performs its intended function under specified conditions for a specified period of time. When the item under consideration can be decomposed in several subsystems, then reliability specifies the probability that no operational interruptions will occur during a stated time interval, i.e. the system can perform its intended functionality even if some of its subsystems might fail. The growing complexity of equipment and systems, as well as the rapidly increasing cost incurred by loss of operation as a consequence of failures, have brought to the forefront the aspects of reliability, maintainability, availability, and safety. The expectation today is that complex equipment and systems are not only free from defects and systematic failures at time $t = 0$ (when they are put into operation), but also perform the required function failure free for a stated time interval and have a fail-safe behavior in case of critical or catastrophic failures [5].

Reliability engineering is the discipline of ensuring that a product will reach a certain predefined level of reliability when operated in a specific manner. In general, three steps are necessary to accomplish this objective: (1) accounting for reliability considerations during the design phase (this will become the product's inherent reliability), (2) minimizing production process variation to assure that the process does not appreciably degrade the inherent reliability, and (3), once the product is deployed, realizing appropriate maintenance actions (according to the product's predefined maintenance policy) to alleviate performance degradation and prolong product life [87]. In the case of a factory which manufactures a product, maintenance is an activity conducted in parallel with production. It can have a great impact on both the capacity of the production and the quality of the products produced. It has been estimated that in a typical factory between 15% to 40% (average 28%) of total production cost is attributed to maintenance activities [58]. Moreover, it is generally true that for complex repairable systems with long life cycles the operating and maintenance cost is a significant portion of its life cycle cost.

In the past, for some manufacturing industries (like the energy generation industry), the first two avenues of improvement were pursued by manufacturers while the third was pursued by operators (who followed guidelines provided by the manufacturers). The last decades, however, have seen a shift in the predominant business model under which these manufacturing industries operate. Instead of a business model oriented toward the "selling of products", the new paradigm is oriented to the "selling of services". In the power generation industry for instance, this shift translates into manufacturers selling a power plant that will be able to operate a certain numbers of hours a year, producing a certain amount of energy for a period of 20 or 25 years. To achieve this, the manufacturer is not only commissioned the task of providing a power plant, but it is also awarded a maintenance contract for the whole life cycle of the plant. In the aerospace industry, an airline customer pays an engine service provider in proportion to the number of aircraft flying hours, which is affected by engine uptime [34]. The effect of supply chain contracting on product reliability have been studied in the operations management literature. In particular, [67] analyze product reliability in the automotive industry.

One of the effects this market shift has had is to increase the need for accurate models that can capture both the failure process of a complex system (like the gas turbine of a power plant) and the effect of the different maintenance policies (with their corresponding maintenance actions) on the remaining useful life of it. We call this models *maintenance models* and their availability would allow the manufacturers to optimize their design and maintenance policy at the same time.

The maintenance modeling problem is the problem of finding a mathematical representation of the ways a technical system under consideration can fail in conjunction with its associated maintenance policy. The usefulness of the resulting model will be based on: (1) its ability to capture all the relevant phenomena under consideration (failures and maintenance actions), (2) its ability to produce accurate enough results in a timely manner, and (3) the easiness with which the representation can be generated and verified.

When developing such models there are several issues that have to be tackled. One of them has to do with the modeling of the inherent dependencies that arise between the different subsystems of a complex systems. In the following sections the characterization of a complex system used in this work is introduced, followed up by an overview of reliability and maintenance process modeling.

1.1.1 On the Characterization of Complex Systems

For the remainder of this work, a complex system will be defined as system which has a structure of multiple units (possibly of several different types) which work together to perform a particular function. Examples of complex systems are railway networks, motor vehicles, and gas turbines.

We define a unit (component) as an entity in a system that is not further subdivided. This does not imply that an element cannot be made of parts; rather, it means that, in a given reliability/maintenance study, it is regarded as a self-contained unit and is not analyzed in terms of the functioning of its constituents.

In the simplest case, a unit can only have two states: operating or failed. In general, one can also include a number (usually finite) of deterioration states. If upon failure the system can be brought back to an operating state the system is called repairable.

The structure of a complex system is based on the relationship between the states of the components and the states of the system. The three most basic structures that are widely found in engineering systems are the series, parallel and k-out-of-n systems. In a series system, failure of any component results in failure of the system. In a parallel system all components of the system must fail for the system to fail. A k-out-of-n system functions when at least k of its components function.

1.1.2 Reliability modeling

Reliability models focus on capturing the failure process of a system. Historically, the reliability and safety modeling of engineering systems has been approached from opposite directions in at least two distinct dimensions. On the one hand, there is white-box vs. black-box dichotomy [6] where the distinction is based on whether the failure process of an entity is modeled with or without the explicit recognition of individual constituents (components) that comprise the entity. Here "component" refers to an elementary building block of a white-box (system) model, which can correspond to a lower-level entity if models are constructed hierarchically, or to the lowest level of the hierarchy, as determined by practical considerations (*e.g.*, individual modules, such as line-replaceable units or LRU). On the other hand, there is non-repairable vs. repairable dichotomy [76] with the former approach dealing with a single failure event of an entity, and the latter addressing repeated failure events (which assumes the possibility of partial or full recovery from failures). While a *non-repairable* entity is characterized by its lifetime distribution (e.g., the cumulative distribution function of its time to failure) random variable, a *repairable entity* behavior is described by a stochastic point process, and so must be characterized differently, e.g., using the rate of occurrence of failures (ROCOF), or the expected number of failures for a given time period. Any permutation of those choices translates into appropriate set of tools: for example, selecting the black-box direction can lead to accelerated testing techniques for non-repairable entities, and to modeling repairable entities by means of stochastic processes.

Furthermore, the white-box approach entails selecting either the repairable or non-repairable option both at the system (output) and component (input) levels.

Boolean algebra methods (*e.g.*, fault trees and reliability block diagrams) assume that both systems and the components comprising those systems are non-repairable. This symmetry between the inputs and outputs characterization (and associated simplicity and clarity) is perhaps one of the reasons for the popularity of those tools. In contrast, the use of superimposed processes [76] implies that both inputs and outputs are repairable entities. In this context, the state-space models that are the subject of this work can be classified as selecting repairable outputs and non-repairable inputs within the white-box (system) approach. This state-space representation is attractive in the context of modeling maintenance processes, as the impact of individual changes to the system (*e.g.*, change of the maintenance interval, or introducing a more reliable module) can be captured directly (See Fig. 1)

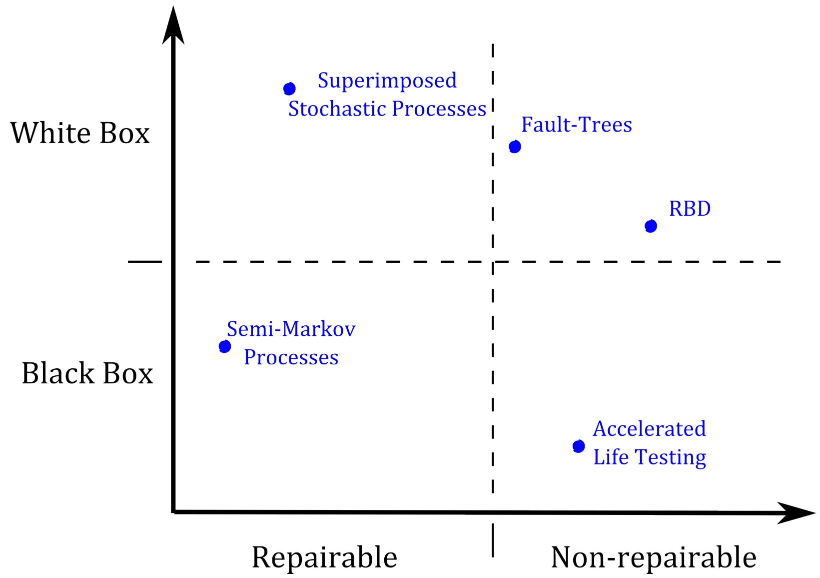


Figure 1: Reliability Modeling Paradigms

1.1.3 Maintenance Modeling

Maintenance is any action that has as an objective to restore or maintain the ability of the system to perform its design function. The high level of sophistication of current systems necessitates appropriate levels of sophistication for the maintenance processes as well. Maintenance not only improves the cost-efficiency of operating a system but it can also significantly reduce the probability of catastrophic failure of the system. The maintenance managers must plan the maintenance actions, so that a balance is achieved between the expected benefits and corresponding expected potential consequences [89]. Numerous models of maintenance process have been developed over the past 30 years; see the review papers by Cho and Parlar [11], Dekker [20, 22], Scarf [69], Nicolai and Dekker [64], and Das and Sarmah [17].

When dealing with complex systems, multi-units maintenance policies must be considered (as opposed to single-unit maintenance policies). The importance of multi-unit maintenance has been well recognized, and extensive literature surveys on the subject [11, 22] make it clear that, as the number of components in the system increases, the level of details captured by the models decreases in order to make modeling overall complexity tractable.

At the same time, there is compelling evidence that in both natural and engineering domains, complex systems are unlikely to be fully coupled, as modular architecture provides clear advantages in developing desirable systems properties. The evolutionary advantage of so-called nearly decomposable systems has been demonstrated for biological systems [72], while similar processes were identified in the history of steam engine development [28]. These concepts are also explicitly employed in the design principles of computer systems [15] (including structured design [74]). It is therefore logical to take full advantage of the modular structure of the systems in modeling system failures.

While in some situations a fully decoupled modeling of each unit is possible, coupling mechanisms can significantly impact the results. Nevertheless, single-unit models are still widely used in practice due to the overwhelming complexity of alternative methods of modeling, leading to sub-optimal selections of maintenance policies.

1.2 Problem Statement

As noted earlier, the maintenance modeling problem consists of developing a mathematical model that captures both the failure process and the maintenance policy during the system's life cycle. The failure process is essentially stochastic and the maintenance policy can have deterministic and stochastic components. For complex systems, the state of the system depends on the state of its components and the state of the components themselves can change due to different failure modes or the state changes of other components. The maintenance policy may stipulate different maintenance actions depending on the states of multiple components. For example failure of some components may trigger maintenance actions on other components.

The goal of this thesis is to define a modeling framework that allows the maintenance modeling in a straightforward, fast, and easily verifiable manner.

1.3 Organization of the thesis

In Chapter 2 we describe the current approaches to reliability and availability modeling. In Chapter 3 we introduce the Stochastic Petri Net with aging tokens modeling framework which is the main tool on which our proposed framework is based. In Chapter 4 we introduce the component based modeling framework and apply it to two problems. In Chapter 5 we show how the methodology is applied to the modeling of the maintenance process of a gas turbine engine. In Chapter 6 we present our conclusions and outline directions for possible future work.

CHAPTER II

BACKGROUND

In this chapter background information on the topics and results known in the literature and which are used throughout this work are concisely presented. The chapter starts with a characterization of the failure and maintenance process of complex system. Then, a discussion on the modeling paradigms used for evaluation of reliability models is given and at the end, the stochastic Petri net framework that will be used as the basis for the modeling is introduced as well. Complementary information presenting a summary of probability descriptions used to capture stochastic events [36, 49, 56] is given in the Appendix.

2.1 Characterization of the Failure and Maintenance Process

Before attempting to define a modeling framework it is essential to understand the characteristics or effects that affect the problem at hand. In this section different characteristics associated to the failure process and the maintenance process are discussed. Dependence effects that contribute to the complexity of the system being modeled and the modeled interactions between the different components include the consequences of failure of a component as well as coupled maintenance policies.

2.1.1 Failure Process

The failure process of a multi-component system is the process that describes how the system goes from an operating state to a failed state (or in the case of multi-state systems to a degraded state and then to a failed state). The process is often the result of forces and stresses generated either during the intended operation of the systems themselves or from external sources. Common failure mechanisms and causes include wear degradation, corrosion, fracture, shock loads, fatigue, etc. The failure process can be characterized by the structure of the system and by the failure models of its components. Random shock modeling has been extensively studied for cases in which devices are exposed to external shock environments, such as sudden and unexpected usage loads and accidental dropping onto hard surfaces. In the literature, there are four categories of random shock models; (i) extreme shock model: failure occurs when the magnitude of any shock exceeds a specied threshold; (ii) cumulative shock model: failure occurs when the cumulative damage from shocks exceeds a critical value; (iii) run shock model: failure occurs when there is a run of k shocks exceeding a critical magnitude; and (iv) δ-shock model: failure occurs when the time lag between two successive shocks is shorter than a threshold δ [50, 61].

When studying the failure process of a complex system a distinction can be made between a failure-based reliability approach and a degradation-based reliability approach. In a failure-based reliability approach, the random variable of interest is the failure time of the component. In general, parametric statistical models are used to capture the stochastic behavior of the lifetime. In contrast, in degradation-based models, the random variable of interest is the remaining useful life (RUL). The RUL is assumed to depend on another random variable that can be monitored. During operation of the system the observable random variables are used to obtain estimates of the RUL (see, for example, Lu and Meeker [52], Singpurwalla [73], and Kharoufeh and Cox [40]).

The structure of a system has a signicant effect on the system availability and reliability. In particular, a system can have a series, parallel, redundant structure or some combination of the above. In a series structure, the failure of any component leads to failure of the system; in a parallel structure all components must fail for the system to fail. For redundant structures, the redundancy can be hot, warm or cold, depending on how the redundant element is being used. In hot redundancy, the redundant element is operating in parallel to the main element. In warm redundancy the redundant element is in stand-by while the main element operates. Only when the main element fails the redundant element starts operating. In cold redundancy, the redundant element is switched off and is only put into operation if and when the main element fails. In complex systems with several hierarchical levels redundancy can be implemented in any of the hierarchical levels. Finding the specific optimal configuration of a specific system is addressed by the reliability allocation problem (Chern [10], Coit [13], Kuo [44], [45]). The *structure function* is used to map the state of the components to the state of the system. One of the basic characteristic that is required for a system in this sense, is that it is coherent. A system is coherent if all of its components are *relevant* and its structure function is monotone. In this work we focus on complex systems that are coherent.

At the lowest level of the hierarchy a unit can have different failure modes. Considering the modes separately might be of importance as either the consequences of the failures might be different or the maintenance actions that each failure mode triggers might be different. In general, the failure of any single component can be considered in a competing risks framework where every failure mode is competing against the others to make the component fail in that mode.

The proposed modeling framework is able to deal with all these characteristics of the failure processes associated with a complex systems. The examples shown however, only deal with a failure-based reliability approach.

2.1.2 Maintenance Process

The classification of maintenance processes described in this section follows [53]. Maintenance includes all technical and administrative actions intended to maintain a system in or restore it to state in which it can perform at least part of its intended action [20]. Maintenance can be classified according to its type and its degree.

Maintenance type can be classified into two main categories: corrective maintenance (CM) and preventive maintenance (PM). Corrective maintenance are all maintenance actions performed after a system has failed with the objective of restoring its functionality. Preventive maintenance refers to planned maintenance actions performed while the system is operational with the objective maintaining the system over a desired operational time-horizon by preventing or delaying failures. Condition-based maintenance is a type of PM which triggers the maintenance action when the measurement of a condition or state of the system reaches a threshold that reflects some degradation and/or loss of performance (but not yet failure). Opportunistic maintenance includes both CM and PM and has as an objective to make use of the economic dependencies that are present in the system being studied.

According to its degree maintenance can be classified in five types [85]: a) *perfect repair* or *perfect maintenance* are actions which restore a system operating condition to "as good as new" (generally, replacement of a failed system by a new one is a perfect repair); b) *minimal repair* are maintenance actions that restore the system to the same failure rate that it had when it failed; and c) *imperfect repair* are maintenance actions that make a system not "as good as new" but younger (better) than it was before the repair; d) *worse repair* are actions which unintentionally make the system failure rate (or actual age) increase but the system does not break down; and e) *worst repair* are actions which unintentionally make the system fail or break down.

Maintenance policies are a set of rules that describe the types of maintenance actions that are considered in response to what types of events [53]. For instance, in an age replacement maintenance policy a unit is replaced at failure (CM) or time T (PM), where T is constant [85]. Preventive maintenance policies are perhaps one of the most studied maintenance policies in the literature [17, 84].

2.1.3 Dependence

In a complex system, there exist different types of interactions between the units. These interactions arise from the design of the system and its intended function. Three major types can be identified, the first two are stochastic dependence and structural dependence [75] and the third is economic dependence [11].

Stochastic dependence occurs if the condition of components influences the lifetime distribution of other components. This kind of dependence defines a relationship between components upon failure of a component. For example, it may be the case that the failure of one component induces the failure of other components or causes a shock to other components.

Structural dependence applies if components structurally form a part, so that maintenance of a failed component implies maintenance of working components, or at least dismantling them. This restricts the maintenance manager in his decision on the grouping of maintenance activities.

In a multi-unit repairable system, economic dependence between components of the system is said to occur if the cost of performing maintenance on the group of components is different than the cost of performing the same type of maintenance individually [11]. [64] refines the concept by introducing the concepts of positive economic dependence (if the cost of joint maintenance is lower than the cost of individual maintenance) and negative economic dependence (if the cost of joint maintenance is higher than the cost of individual maintenance).

Positive economic dependency can arise, for instance, from economies of scale (set-up costs). This economies of scale result when the maintenance cost per component decreases with the number of maintained components. Another form of positive economic dependency is downtime opportunity (component failure is treated as an opportunity for PM of non-failed components) like in the case of most continuous operating systems. For this type of system, the cost of system unavailability may be much higher than component maintenance costs [85].

Negative economic dependency can arise, for instance, from manpower restrictions. Grouping maintenance actions results in a peak in manpower needs. Manpower restrictions may even be violated and additional labor needs to be hired. Another form of negative economic dependence arises from safety requirements, for instance, use of equipment may hamper use of other equipment and cause unsafe operation [64]

There are several general sources of dependent state transitions [11, 47]:

- Common-cause failures, either due to common environment (*e.g.*, cold temperatures during the Challenger launch that impacted both O-rings), or a common defect (*e.g.*, if components were obtained from the same batch made by the same manufacturer). They can be considered an example of stochastic dependence. In either case it is possible to explicitly model the common-cause failure by providing appropriate state transitions that impact several components simultaneously, while considering the remaining causes of failures for the components to be independent.

- Shared-load configuration (another example of stochastic dependence), where the failure of a component can cause redistribution of the load for the components that remain in operation, and therefore change their failure distributions. From the white-box modeling perspective, this implies that several component states must be distinguished, each having distinct failure distribution associated with the transition to the failed state (in this context care needs be taken to account for aging processes in each state [80]). In addition, the consequences of the failure of the modeled component can be different from the system's perspective: redundant configurations can be considered a special case of shared load, if a failure of another component does not alter the "load" for the component, and therefore its failure distribution.

- Opportunistic maintenance: if one of the components fails and needs a replacement or a repair, other components of the system can be serviced at the same time (an opportunity is created for conducting maintenance actions on those parts). For a variety of reasons (*e.g.,* the system has been taken off line or partially disassembled, technicians are available, etc.), servicing several components simultaneously is more efficient economically than servicing each component in isolation (hence the name "economic coupling").

- Induced (cascading) failures, when the failure of a component causes other failures (for example, a liberated blade in a gas turbine can break neighboring vanes).

- Logistics coupling usually refer to the logistics constraints where repair of a component is dependent on what happen to other components that might be competing for the same resources (spare parts, labor, or facilities). However, there is also the possibility of a positive coupling, where failure of another similar system provides an additional repair resource (*e.g.*, cannibalization of parts).

2.2 Modeling Paradigms

Before describing the modeling paradigm on which this work is based, a discussion of the different paradigms usually encountered in the literature is provided. Fig. 2 shows a general classification of modeling paradigms. A possible classification is provided by [88]. The classification is purposely done at a very high-level which does not show some approaches combining more than one paradigm.

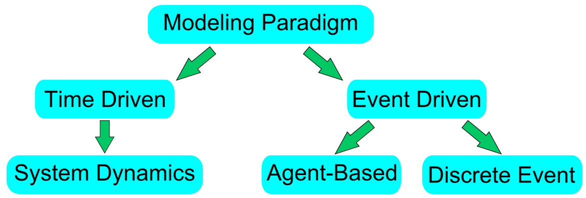


Figure 2: Modeling Paradigms

A model built using the *system dynamics* paradigm involves the iterative evaluation of a system of ordinary differential equations. In such models, the state of the system varies continuously in time. In contrast, in event-driven modeling, the state of the system only changes when an event (from a set of possible events) occurs. Although it is always possible to create a model of a given problem based on any of the paradigms, some types of problems are more easily captured using a specific paradigm. System dynamics is useful where the number of individual entities is large and all the concepts in the real system can be thought of as continuous quantities interconnected in loops of information feedback. On the other hand, since event driven modeling has as a main focus the occurrence of an event and describes the evolution the system as a sequence of events, it is appropriate where each event is important enough to register individually. In the agent-based approach the models simulate the simultaneous operations and interactions of multiple agents (who are presumed to be acting on their perceived self-interest) with the goal of recreating and/or predicting the appearance of complex phenomena. With respect to the discrete-event paradigm there are two main differences: agent-based models can have continuous states and the agents can use more sophisticated decision rules including for instance, adaptation of those rules (i.e. learning).

The modeling of maintenance processes involves in general a large number of items with stochastic events associated to each of them. Their operational lives are often much larger than the time needed to inspect, repair or replace them. Although each item has associated failure behavior and maintenance rules and hence could be considered an agent, they are in passive in the sense that events occur *to* them. Based on this, the most appropriate paradigm to capture their behavior is an event-driven modeling paradigm. Different frameworks exist for the modeling of event-driven processes. One of them, Stochastic Petri nets, possesses enough flexibility to capture the dynamics of the maintenance process and is introduced next.

2.3 Petri Nets

Petri Nets are a modeling framework introduced by Carl Adam Petri in the second half of the 20^{th} century [66]. It offers a graphical notation for event-driven processes that include choice, iteration, and concurrent execution. Petri nets have an exact mathematical definition of their execution semantics. Since their inception, Petri nets and in particular their Stochastic Petri Net generalization have been used extensively to model discrete event-based processes. Although several books [18, 31, 35, 37, 54, 70] extensively describe the formalism in the following sections a brief review of it is given.

A Petri net consists of places, transitions, and arcs. Arcs can connect a place to a transition or vice versa, but never two places or two transitions (mathematically, a Petri net is a bipartite directed graph with one set of nodes being the places and the other set of nodes being the transitions). Graphically, places are represented by empty circles, transitions are represented by filled rectangles and arcs are represented by arrows. Fig. 3 shows a simple Petri net with all of its elements appropriately labeled.

The places from which an arc runs to a transition are called the input places of the transition; the places to which arcs run from a transition are called the output places of the transition. In Fig. 3, Place 1 is an input place of Transition 2 and an output place of Transition 1.

Graphically, places in a Petri net may contain a discrete number of marks called tokens. Any distribution of tokens over the places will represent a configuration of the net called a marking. In Fig. 3, there is one token inside Place 1, the marking of the Petri net is then 1 token in Place 1 and 0 tokens in Place 2.

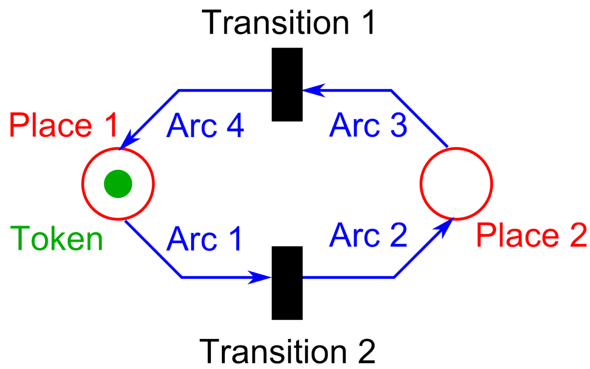


Figure 3: Two-state Petri Net. Marking 1

2.3.1 Execution Semantics

Given a marking, the execution semantics of the Petri net are as follows:

- When there are sufficient tokens in each of the inputs places of a transition, this transition becomes enabled and it can fire.
- When a transition fires it consumes the required input tokens, and creates tokens in its output places. A firing is atomic, i.e., a single non-interruptible step.

In Fig. 3, Transition 2 is enabled because there is one token in its only input place. Immediately after becoming enabled, Transition 2 fires consuming the token in Place 1 and creating a token in Place 2. The marking on the net after the firing of Transition 2 is shown in Fig. 4.

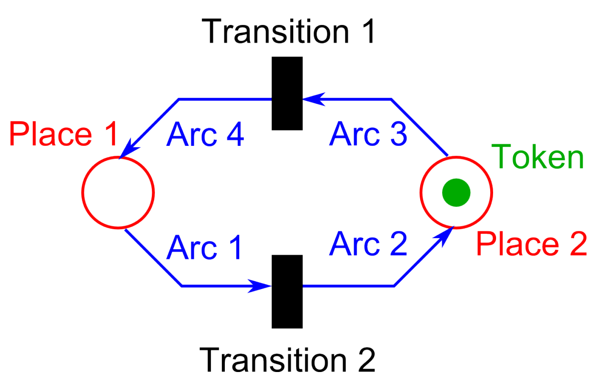


Figure 4: Two-state Petri Net. Marking 2

Unless an execution policy is defined, the execution of Petri nets is nondeterministic: when multiple transitions are enabled at the same time, any one of them may fire. Since firing is nondeterministic, and multiple tokens may be present anywhere in the net (even in the same place), Petri nets are well suited for modeling the concurrent behavior of distributed systems.

2.4 Stochastic Petri Nets

Stochastic Petri nets are one of the many extensions to the Petri Net framework. In Stochastic Petri nets, the original framework is extended as follows,

- First, a measure (concept) of time is introduced. The traditional method of doing this is by allowing a period of time to elapse between enabling of the transition and firing of the transition. This period of time will be called the transition firing time. With this extension, the framework is usually named Timed Petri Nets.
- Second, a probability distribution is associated to the firing time of all the transitions. Molloy [60] introduced this extension by considering exponentially distributed transition firing times and presenting an isomorphism between the behavior of the resulting framework and Markov processes. The execution semantics are modified as follows: When a transition becomes enabled, its associated probability distribution is sampled once. The sampled time is the next firing time of the transition. With this extension, the framework is usually named Stochastic Petri Nets. It must be noted that within the same framework transitions with a deterministic firing time and with immediate firing time can be captured. Conceptually they are characterized by an appropriately shifted Dirac distribution. The latter extension is due to Marsand [55].

2.4.1 Execution Semantics

Given an initial marking and an initial time (which is usually equal to 0), the execution semantics of the net are as follows:

- The set of all enabled transitions is calculated.
- For each enabled transition, the transition's firing time is calculated according to its associated firing rule (probabilistic, deterministic, or immediate).

- The minimum value of the set of firing times is calculated. Let's call this time $\Delta_{\min}$.
- The clock is advanced $\Delta_{\min}$ (i.e. the new time is $t = t + \Delta_{\min}$). The transition associated with $\Delta_{\min}$ is fired. The firing process is the same as the one for regular Petri Nets.

This process repeats itself until either:

- The set of enabled transitions becomes empty, or
- The clock reaches or surpasses the (predefined) final time

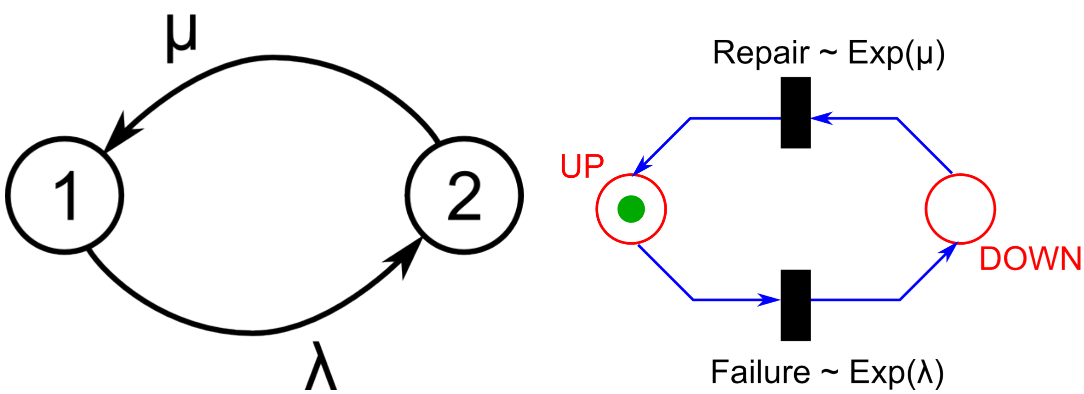
With the semantics defined like this, it can be proven that the modeling power of the framework is equivalent to the modeling power of the Semi-Markov process [35].

2.5 Reliability Modeling with Stochastic Petri nets

One advantage of the Stochastic Petri Net paradigm when used to model maintenance processes is its ability to elegantly deal with PM maintenance actions. The modeling of PM actions, in particular the quantifying of the effect of performing PM actions at different intervals is one of the most difficult maintenance activities to model when using stochastic point process as the modeling framework [12, 42, 65].

In this section, a simple example of a Petri Net model is constructed. Consider an item with a single failure mode. The time to failure is modeled using an exponential distribution with failure rate λ. Once the item has failed, it is repaired. The repair process is modeled using an exponential distribution with repair rate μ. Fig. 5a shows a Markov model representing the failure-repair process of the item and Fig. 5b shows the same process using the Stochastic Petri Net framework.

In the SPN model the marking of the net capture the state of the item and the transitions capture the failure and repair processes. Looking at both models, a valid question is why introduce the SPN framework if, in this case, it looks almost exactly the same as the Markov model? To answer that question, consider the following variation of the previous problem: the failure process is better modeled by a Weibull distribution instead of an exponential distribution. To model this new problem with Markov we either:



(a) Markov Model

(b) Stochastic Petri Net Model

Figure 5: Single Item Model

- have to use an average failure rate that approximates the failure rate of the new distribution in the period of interest or,
- we have to increase the state-space with rates between the new transitions capturing the behavior of the Weibull distribution

Both alternatives involve an approximation of the "reality" of the problem. Alternatively, to model the new version of the problem with SPNs we simply need to modify the probability distribution associated to the failure transition from an exponential distribution to a Weibull distribution. Clearly, for the new problem, the SPN framework allows for a simpler representation of the failure repair process of the item.

Summarizing, the SPN framework has a greater modeling power than the Markov process framework. This greater modeling power is the main reason motivating the introduction of the SPN framework for the modeling of reliability problems. It must be noted as well that this greater modeling power is not free; it comes at the expense of the number of analytical tools that can be used to evaluate the model.

2.6 Stochastic Petri Net Model Evaluation

So far the discussion has been centered on the framework specification and reliability model creation. In this section, we focus on how to use an available model for the evaluation of performance indicators of interest.

The main point that must be realized is that, unlike the case of homogeneous Markov

Chains models, it is very difficult to obtain performance parameters analytically. Hence, to evaluate the model we must resort to simulation of the model. Furthermore, since the model contains probabilistic elements the appropriate simulation framework is the "Monte Carlo Simulation" framework. Monte Carlo methods are a class of algorithms that obtain numerical results by performing repeated random sampling. [68, 78] present the general method, [90] presents applications of the method in the reliability evaluation setting.

2.7 Overview of Simulation Approach

In general, *simulation* is the calculation of some metric of interest based on a model representing a real system and not the real system itself. When the simulation uses random numbers it is called a *Monte Carlo simulation.*

There are two types of simulation model analysis:

static Models the behavior of the model of a system at a given instant of time. It requires knowledge of the full-state of the system at that instant of time.

dynamic Models the time evolution of the model of a system

In the context of "Monte Carlo Simulation" the research question is stated in such a way that its answer is the expected value μ of a random variable X. In this framework, a Monte Carlo simulation will calculate n realizations of X and then estimate its expected value with the sample mean,

$$\bar{X}_n = \frac{1}{n}\sum_{i=0}^{n} X_i \tag{1}$$

The number n of realizations that must be calculated is based on the desired error of the approximation. The central limit theorem (CLT) states that for a sequence of i.i.d random variables with finite mean μ and finite variance σ^2, the random variables $\sqrt{n}(S_n - \mu)$ converge in distribution to a normal $N(0, \sigma^2)$ as n goes to infinity.

Using the CLT it can be shown that,

$$\mathbb{P}\left(\bar{X}_n - c_\alpha \frac{\sigma}{\sqrt{n}} \leq \mu \leq \bar{X}_n + c_\alpha \frac{\sigma}{\sqrt{n}}\right) \approx 1 - \alpha \tag{2}$$

where c_α is the α-quantile. Eq. 2 relates the error of the approximation (width of the interval) with the number of replications.

The principal strength of simulation is its flexibility. There are few restrictions on the behavior that can be simulated, so a system can be represented at an arbitrary level of detail. At the abstract end of this spectrum is the use of simulation to evaluate Markov models or Stochastic Petri Nets. At the concrete extreme, running a benchmark experiment is in some sense using the system as a detailed simulation model of itself.

The principal weakness of simulation modeling is its relative expense. Simulation models generally are expensive:

- to define, because this involves writing and debugging a complex computer program.
- to parameterize, because a highly detailed model implies a large number of parameters.
- to evaluate, because running a simulation requires substantial computational resources, especially if narrow confidence intervals are desired. It must be noted, however, that there exist techniques that allow the computation of estimators with a reduced variance (with respect to the variance of the typical estimator). This reduces the number of replications that need to be performed to obtain a desired level of confidence. Two such techniques are importance sampling and splitting [78].

CHAPTER III

COMPONENT-BASED MODELING OF MAINTENANCE PROCESSES WITH ECONOMIC DEPENDENCE

A component-based modeling approach to maintenance process that feature economic dependence is proposed. The approach, its advantages and required inputs are characterized.

3.1 Motivation

The most signicant challenge in building practical system-level dependability modeling tools has been dealing with largeness and complexity in the systems that are modeled [14]. Design space explosion, i.e. the exponential increase of the number of design configurations that the modeler intends to evaluate as the number of parameter or parameter values used to vary the model configuration increases [29].

Furthermore, when using SPN's to capture the behavior of systems of practical interest, very rapidly the graphical model itself often *looks* complicated and difficult to understand. Usually it comprises of numerous parallel or crossing arcs which give an idea of complexity even if, in reality, the behavior is rather simple [71].

To overcome the design space explosion, a modeling based on component-level models is proposed. The approach eases model creation by allowing the modeler to only explicitly capture design configurations that are explicitly related to the component being modeled. The interactions between different components are captured through a probability model whose parameters are found iteratively.

3.2 System-level vs component-level models

To illustrate these options, let us consider a system consisting of n identical components that exhibit an increasing hazard rate (as specified below) and are subject to age-based replacement [3]: all parts are replaced at failure or after a specified interval s. The components have a single failure mode and a failed component causes a total renewal of the

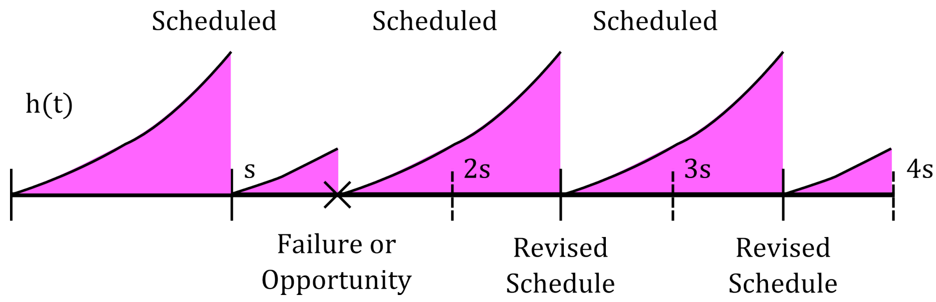


Figure 6: Age-based replacement maintenance schematics

system: all components (not only the failed one) are replaced with brand new ones; furthermore, we assume that the replacements occur instantaneously. Note that failures can shift the original schedule of replacements (see Figure 6). Here the outputs of interest are the expected numbers of scheduled replacements and failures. For a given component, an opportunistic maintenance action is performed if the component has been replaced due to the failure of another component. The number of opportunistic maintenance actions for a given component is simply the sum of the failures for all other components.

A system three-state diagram is depicted in Figure 7A: either scheduled maintenance takes place and all n components are replaced, or one of the components fails (and is replaced), while the rest (*i.e.,* $n-1$ components) are replaced as a result of opportunistic maintenance. Out of four transitions depicted, only one, τ_{UF}, is random: transitions τ_{FU} and τ_{SU} are instantaneous, while τ_{US} has a fixed delay or holding time. For a single component system ($n = 1$) no opportunistic maintenance is possible, and the transition τ_{UF} occurs when the component fails. The timing of transition τ_{UF} is fully defined by the cumulative distribution function $F(t)$ for the failure of the component: once a new part is put in service t_1, the failure time is independent identically distributed in accordance with $F(t-t_1)$, as the replacement component has the same properties as the original one. The resulting model corresponds to a semi-Markov process where the time distribution of a transition to a new state depends on the current state together with the time the system spent in the current state (so-called holding or sojourn time), but not on the previous history. The failure intensity, or hazard rate, is evaluated as $h(t) = f(t-t_1)/(1-F(t-t_1))$, where $f(t)$ is the probability density function corresponding to $F(t)$. An increasing hazard rate implies $h(t_3) > h(t_2), \forall\, t_3 > t_2 > t_1$.

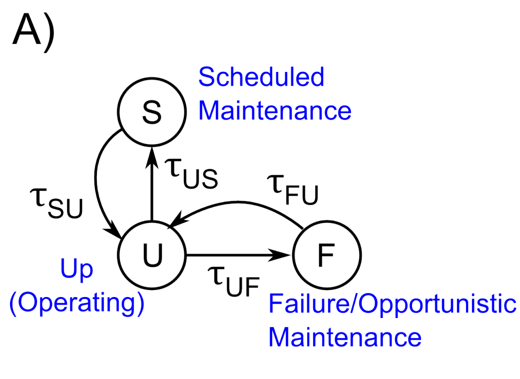


Figure 7: State-space representation of a simple system with identical components: A) Global (system) B) local (component)

As the number n of components in the system increases, the transition distribution τ_{UF} needs to be adjusted to represent failure of any of n components instead of one, but the structure of the model (Figure 7A) remains unchanged. However, this is rather an unusual situation, as real-life systems are likely to consist of distinct types of components with nonsymmetric interrelationships, which will cause the size of the system model to grow dramatically, as the system states must be explicitly represented (while the state of each component can be inferred from the system state). We will refer to such models as "global" (Markov chains also fall under this category), as opposed to "local" models that describe explicitly the states of components, so that the state of the system can be inferred. Stochastic Petri Nets (SPN) represent an example of local modeling that can provide more compact models with large number of components [24, 35], and a version of SPN will be utilized in the next section to create such system-level models. However, even for local models, a description of the fully coupled behavior with hundreds (or thousands) of components is challenging, if not impossible.

3.3 Introduction to Component Modeling

To this end, constructing decoupled (component) models with accurate representation of the coupling among components provides an intriguing alternative (*cf.* [51]). In the context of state-space modeling this implies specifying any appropriate additional states of the component, which is relatively easy to do if the coupling mechanisms are well understood.

However, compact representation of the transitions between the component states due to coupling effects can be significantly more challenging, and addressing this challenge is the main focus of this chapter. It is reasonable to assume that this representation can depend on the type of coupling, so a brief overview of the possible types of coupling is described next.

First, we note that in the context of state-space reliability modeling, coupling is closely related to dependency. It can be argued that one of the purposes of white-box modeling is explicit modeling of dependency mechanisms, so that the assumption of independence can be reasonably applied to individual state transitions, even if, from the black-box perspective, the corresponding system-level transitions appear to be statistically dependent.

From the modeling perspective, these sources can divided into two scenarios: in the first (relatively simple) scenario the occurrence of an additional condition is time-independent (as it either happens or does not). Effectively, the relevant event occurs prior to the modeled time period. For example, components can belong to a higher-risk subpopulation, and a single value of the corresponding probability is sufficient (combined with the definitions of the lifetime distributions for both higher-risk subpopulation and for the rest of the component population). In contrast, the timing of another event is important in the second scenario so, strictly speaking, the full description of the timing of that event needs to be specified. It can be observed that the majority of the listed above coupling types follow this second scenario, where effectively there is a "race" between the internal and external events in the component model. We will refer to those scenarios as competing risks as this term is commonly used [4], but in the context of maintenance modeling the competition is not limited to risk-related transitions (*i.e.* some type of failure), but instead refers to any possible transitions for a given state (*e.g.*, timing of the first available spot among several repair queues).

Figure 7 B depicts a single component view of the described process. Here, parameters of a single component model are used for the failure transition ν_{UF}, while a new state corresponding to the opportunity is introduced along with the corresponding transition ν_{UO} associated with the failure of the other $n-1$ components. Parametric distributions are preferred in state-space modeling from the compactness perspective, assuming that their accuracy is assured. Significant further simplification can be achieved if constant transition rates are used: first, each transition is fully characterized by a single parameter, its constant rate λ; and second, if all state transitions have constant rates, the holding times are not affecting the chances of transition to the next state, and the resulting process is Markov (i.e., the chances of transitioning to a new state are fully determined by the current state). The analysis for Markov processes is significantly easier than for semi-Markov processes. A transition with constant rate λ follows an exponential distribution, whose cumulative form is given by $F_e(t) = 1 - e^{-\lambda t}$.

In the case of failure transitions, λ is the reciprocal of the mean time to failure. Fixing the choice of the type of distributions for transitions for Markov models also facilitates hierarchical model construction and aggregation of states and transitions [7, 77].

Steady-state results often depend only on the mean parameters of the distributions associated with the state transitions, justifying the use of exponential distribution even if the underlying distributions are different (see, for example, the extensions of the Palm-Khinchin theorem, especially in the context of logistics [9]).

An additional argument for using exponential distribution is based on what can be characterized as the central limit theorem for repairable systems: in a complex repairable system with multiple components, failures form a homogeneous Poisson process [3] at the system level. The latter argument, however, assumes that there is no coordination among component failures. In practice, for many systems with clear aging or degradation patterns (*e.g.*, gas turbine engines), major inspections and overhauls impose an overall structure, and within each maintenance cycle the failure rate is generally increasing. Opportunistic maintenance has been extensively studied [46], [16] and [25], and in all of those studies the opportunities were assumed to follow exponential distribution [19], [21]. Opportunities can be caused by extraneous events that can be considered random, in which case exponential distribution is quite appropriate. However, the opportunities can be also caused by failure of components, which provides an impetus for investigating the performance of exponential distributions for such cases, and establishing the requirements for a distribution to adequately represent the combined effect of multiple components. Specifically, if the exponential distribution is inadequate, we seek to find an alternative compact representation of a given combined distribution. To this end, the natural step is approximate the targeted distribution with two-parametric distributions, and ensure that both first two moments (mean and standard deviation, respectively) of the targeted distribution are matched. Similarly, the targeted distribution can be sampled and the maximum-likelihood estimate (MLE) can be used to obtain the most appropriate distribution parameters [57].

The hypothesis explored is that an alternative selection of parameters can be more advantageous in the specific context of modeling for coupled maintenance scenarios that involve competing risk type of coupling. The goal is not only to predict the expected number of maintenance events for multiple components, but also to represent the effect of multiple components for component simulation in a compact fashion. The resulting representation can be used as modeling blocks for larger models (for example to evaluate the durations of outages and other relevant system-level effects).

We explore whether in the case of non-identical distributions and when the number of distributions is finite (and even quite small) a Weibull distribution can provide a good approximation for the described combined effect. The justification for the developed method stems from the asymptotic considerations with respect to the small parameter $s/\theta \ll 1$, where s is the replacement interval and θ is scale of the failure distribution. Asymptotic considerations have been successfully used [8, 30, 32] to make approximations for reliable systems where the first terms of the Taylor series have been used. In contrast to that previous work, the present approach relies on the second order approximation.

3.4 Computational Methods for System Analysis

There are no closed-form solution existing for finite time horizons, but two numerical options exist: numerical integration of the corresponding renewal integral equations, or discrete-event simulation. Both options are briefly discussed next.

3.4.1 Renewal equations

For a single component with no scheduled maintenance (when the part is replaced only upon failure) the corresponding renewal process is well studied. For our purposes the following form of the corresponding integral equation is useful:

$$m(t) = f(t) + \int_0^t m(\tau)f(t-\tau)d\tau \tag{3}$$

Here the renewal density $m(t) = \frac{dM(t)}{dt}$, $M(t)$ is the expected number of renewals or renewal function, and $f(t)$ is probability density function [3, 62] (here we assume that derivative of renewal function exists). Efficient methods for the numerical solution of renewal equations exist using, for example, finite differences [86, 23]. The form of Eq. 3 provides a natural interpretation that is amenable to generalization: renewal at time t can occur either due to the first failure at that time with the probability density $f(t)$, or due to the repeated failure, where the previous renewal took place at time τ with the corresponding renewal density $m(\tau)$, and the chances of the failure $f(t-\tau)$ for the renewal time τ.

Let us now introduce scheduled replacements at interval s, so the renewal can be caused either by a failure or by scheduled replacement (if no failures occurred during interval s). Therefore, we can separate renewal density $m(t)$ into the two distinct parts: $m(t) = u(t) + w(t)$, where $u(t)$ and $w(t)$ represent renewal density due to failures and scheduled replacements, respectively. For the first cycle $0 < t < s$ Eq. 3 remains unchanged with $m(t) = u(t)$ and the renewal density due to scheduled replacements does not contribute ($w(t) = 0$). Noting that $R(s) = 1 - F(s)$ represents the chances that no failures will occur throughout the segment s, we conclude that in the vicinity of the first scheduled replacement $t = s$, the renewal density due to scheduled replacement can be described using Dirac delta function $w(t) = R(s)\delta(t-s)$. This is equivalent to stating that the expected number of renewals due to scheduled maintenance is zero for $t < s$ and is equal to $R(s)$ for $t = s$. For $s \leq t$, the renewal density $m(t) = u(t) + w(t)$ can be obtained from the following system of equations:

$$w(t) = R(s)m(t-s) = R(s)\left[u(t-s) + w(t-s)\right] \tag{4}$$

$$u(t) = \int_{t-s}^{t} m(\tau)f(t-\tau)d\tau = \int_{t-s}^{t} \left[u(\tau) + w(\tau)\right] f(t-\tau)d\tau \tag{5}$$

Here the Equation 4 states that in order for the scheduled maintenance to occur at time $t > s$ two conditions must be met: there was a renewal at time $t - s$ and there were no failures during interval s. Similarly, Equation 5 states that in order for the failure to occur at time t previous renewal must occur at some time $t - s < \tau < t$. In general, for n distinct components that can all cause renewal, we can introduce separate renewal densities to to the failure of each type of the component $u_i(t)$, $i \dots n$, and the corresponding equations have the following form:

$$w(t) = R(s)m(t-s) = R(s)\left[w(t-s) + \sum_{i=1}^{n} u_i(t-s)\right] \tag{6}$$

$$u_i(t) = \int_{t-s}^{t} m(\tau)f_i(t-\tau)\tilde{R}_i(t-\tau)d\tau = \int_{t-s}^{t} \left[\sum_{i=1}^{n} u_i(\tau) + w(\tau)\right] f_i(t-\tau)\tilde{R}_i(t-\tau)d\tau \tag{7}$$

Here $\tilde{R}_i(t-\tau)$ is the reliability of all other components, provided by the following expression:

$$\tilde{R}_i(t) = \prod_{i \neq j}^{n} R_j(t)$$

Effectively, in the presence of multiple components we use disjoint sets of possible outcomes, which allows us to sum the probability densities. As a result, we can interpret Eq. 7 as stating that renewal due to failure of component i occurs if the following three conditions are met:

1. Previous renewal took place at time τ, hence the term $m(\tau)$
2. Component i has failed at time t, while it has not failed in the interval $]t-\tau, t[$, hence the term $f_i(t-\tau)$
3. No other components failed during interval $]t-\tau, t[$, hence the term $\tilde{R}_i(t-\tau)$

Solving Eqs. 6,7 using finite differences can lead to highly accurate results as long as the selected time step is small enough. In this study, s is discretized into 1000 segments, which leads to the results with the relative error usually not exceeding 10^{-5}.

3.4.2 Simulation

For a small system, a custom Monte Carlo simulation model can be easily developed, but using standardized graphical representation provides advantages for creating larger models, in particular for verification purposes. To this end, local or component-based representation of the state space is more convenient: instead of each state representing the system as a whole (as in Markov chains), states of individual components are described along with their interactions, so that the system state can be inferred from its component states. This is the essence of Stochastic Petri Nets (SPNs) [63],[59]. In the specific version of SPNs used in this thesis [80], each token can change states independently of the others, unless their behavior is explicitly coupled by means of inhibitors or enablers (denoted as an arc terminated at a transition with a hollow or solid circle, respectively). A transition is disabled by an inhibitor if there are enough tokens in the place where the inhibitor originates, while an enabler acts in the opposite way (the corresponding transition is disabled *unless* there are enough tokens in the corresponding place, see [83]).

Figure 8 depicts an SPN model for $n = 3$ components. On the left, the operating configuration is shown, while on the right is the situation when the failure of one of the components occurs: the corresponding token is moved from the place "Operating" to "Failure," thus enabling the transition of the rest of the tokens to the place "Opportunity." Due to the fact that the state transitions are local, SPNs are not restricted conceptually to constant transition rates, although in practical terms this implies that the solutions are obtained using Monte Carlo simulations. Following a common SPN convention, immediate transitions (without time delay) are depicted with narrow rectangles. Note that both the global state representation, Figure 7, and the local one, Figure 8, depict the same process: Figure 8, A represents the operating state of the system, while B corresponds to failure/opportunistic maintenance in Figure 7. The compactness of the model shown in Figure 7 is somewhat deceptive: generally global models do not scale well with the number of components, and local models are more compact [80] (but still very large for a system with a large number of components).

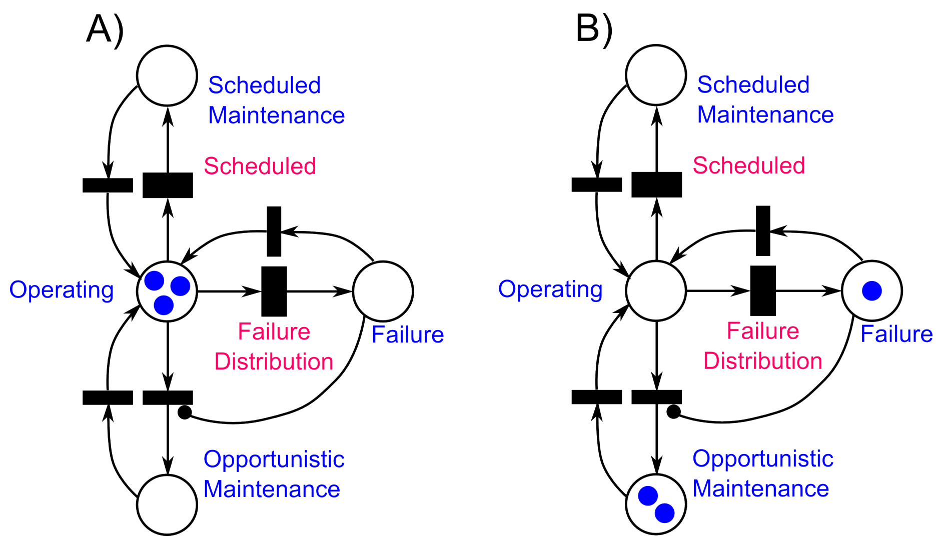


Figure 8: SPN coupled model with three components: A) system is in operating state B) system is in a state when one of the components failed and the rest are undergoing opportunistic replacements

CHAPTER IV

PARAMETRIC REPRESENTATION OF EXOGENOUS EFFECTS

Parametric models are useful because they provide the user with prognosis capabilities and a sense of objectivity, the study of their parameters can lead to profound insights, and permit the automation of statistical analysis.

The selection of an appropriate model is often based on how well it fits the given data. However, what is accepted for one application may not be applicable for another. This concern was captured by [41], who noted that "Although it is often possible to justify the use of a distribution empirically, simply because it appears to fit the data, it is more satisfactory if the structure of the distribution reflects plausible features of the underlying mechanism."

4.1 Decoupled modeling

Further simplification can be achieved by creating separate models for each component (see Figure 9 for an SPN model and compare for a generic state-space representation Figure 7 B) and then combining the results. Again, the simplification becomes apparent only as the complexity of the problem increases. Indeed, in the considered simple case, the coupled model (Figures 7, 8) has three and four total states, respectively, as opposed to three separate models (one per component) that each have four states (Figure 9). It is important to note, however, that the situation is reverse in practical situations where components are distinct (and so the failures of each component need to be counted separately, each requiring a separate state). In general, coupled representation implies explicit modeling of n^2-n pair-wise coupling for n components (coupling in maintenance does not have to be symmetric). In contrast, there will be only n separate component models, with only the combined effect of all other components being represented. This is an attractive strategy, provided that the coupling among the components behavior is either negligible, or adequately captured. In the case of identical components due to symmetry only one such model is required, and

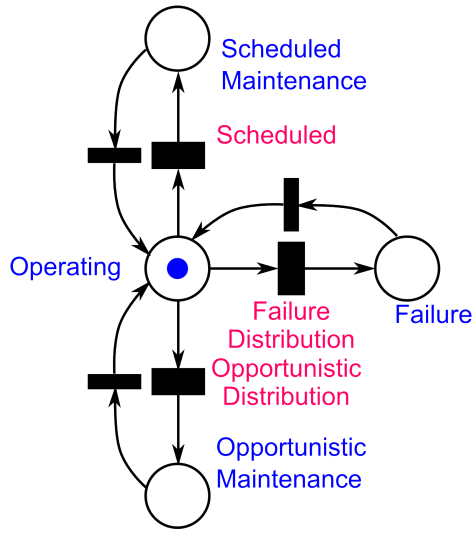


Figure 9: SPN Model for a single component with opportunistic maintenance

the main question is how to represent the opportunity distribution (see the corresponding transition in Figure 9).

4.1.1 Components with identical shape Weibull failure times

The main objective is developing a procedure for a compact representation of a cumulative distribution representing the probability distribution of the opportunity for renewal due to failure of multiple parts.

First, let us consider a situation where all n components $X_1 \dots X_n$ of a given system follow Weibull distributions with the same shape parameter $\beta_i = \beta$, while scale parameters are allowed to be different: θ_i.

Then the opportunity for the first component $i = 1$ stems from the failure of components $i = 2 \dots n$, and the corresponding cumulative distribution function for the opportunity can be calculated as

$$\tilde{F}_1(t) = 1 - \tilde{R}_1(t) = 1 - \prod_{i=2}^{n} R_i(t) =$$

$$= 1 - \exp\left[-t^{\beta} \sum_{i=2}^{n} \frac{1}{\theta_i^{\beta}}\right] \qquad (8)$$

Here $R(t) = 1 - F(t)$ denotes respective reliability functions. As a result, the combined effect of the opportunity is represented by a Weibull distribution with the following scale and shape parameters:

$$\tilde{\beta}_1 = \beta; \qquad \tilde{\theta}_1 = \frac{1}{\left[\sum_{i=2}^{n} \frac{1}{\theta_i^{\beta}}\right]^{\frac{1}{\beta}}} \tag{9}$$

So, if all components have the same shape, then the combined failure distribution has the same shape, and the scale is uniquely defined analytically.

4.1.2 Example 1: Three identical components

The simplest case where the opportunity representation is non-trivial is for systems that consist of three components, so the combined effect of failure of two components needs to be represented. Let us consider a system that operates for time $T = 4$. Each component follows a Weibull distribution with the same shape $\beta = 3$ and scale $\theta = 1$. Based on Eq. 9 we can conclude that the opportunity for each component will follow a Weibull distribution with the same shape parameter $\beta = 3$ and scale $\theta = 1/\sqrt[3]{2}$. We consider a full range of fixed replacement intervals s that span the values from very conservative s that are significantly smaller than the mean time to failure, to the large values of s where the effect of scheduled maintenance is negligible, as the failure will always occur first. As shown in [3], the values of s that are integer fractions of time T correspond to the discrete jumps of the total cost function that combines the cost of failures and scheduled replacements (the jumps are associated with an extra scheduled replacement right before the end of the system life T), while for all other values the cost function is a smooth function of s. As a result, those values provide good characterization of the overall cost function, and so they will be utilized in this and the following examples. There is no replacement taking place at time T, as the system has reached the end of its life. This last point is usually irrelevant for systems that operate for long periods of time, *i.e.,* when the scheduled replacement interval s is much less than T, but it can be significant when the time horizon T is of the same order as s.

As expected, if the opportunity is provided by the calculated Weibull function, solving both coupled and component-based models by discretizing Eqs. 6, 7 leads to identical results within the considered accuracy of 10^{-5} (here and below, the results from finite-difference solutions are based on 1000 time steps for a replacement interval s). Next, we use exponential distribution to represent the opportunity instead of using the Weibull distribution with the shape parameter $\beta = 3$.

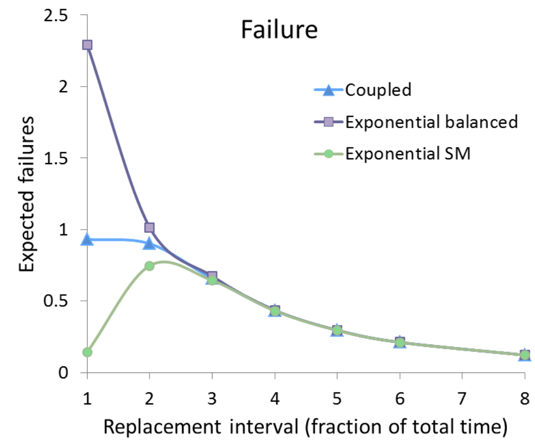
Figure 10 shows the results for expected failures, opportunities, and scheduled replacements for various numbers of scheduled maintenance segments s.

Two methods are compared for estimating the scale of the opportunity:

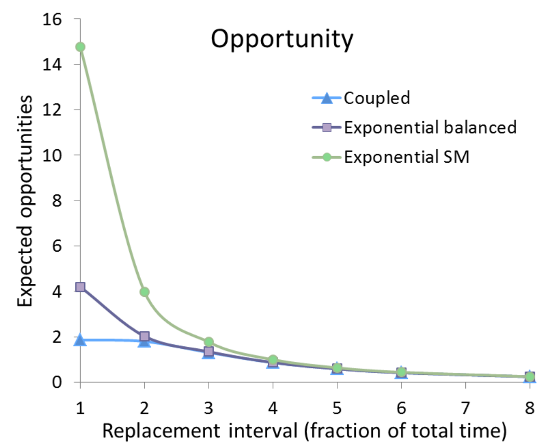
1. Balanced: scale parameters of the opportunity distribution are not pre-calculated, but rather obtained by means of iteration, to ensure that the number of opportunities matches the corresponding number of failures (since each failure entails two opportunities). The convergence of this procedure is assured due to the fact that the involved mapping is monotonically decreasing. Indeed, the more opportunities, the more frequent the maintenance actions, and the fewer failures occur for components with an increasing failure rate. As a result of this convergence, an optimum scale parameter is obtained with respect to failures and opportunities: any improvement in the prediction of opportunities will incur the deterioration of the prediction of failures, and *vice versa*. Note that the while it is relatively easy to obtain this solution for identical components (as this a one-dimensional problem), the scalability of this approach is far from being straightforward, as the system of nonlinear equations needs to be solved.

2. Scale-matched (SM): A more computationally efficient and scalable approach consists of evaluating the combined chance of opportunity considering failures of other components for the relevant replacement interval (independent of the failures of the component itself) and then finding the opportunity scale so that combined number of failures during the replacement interval is matched.

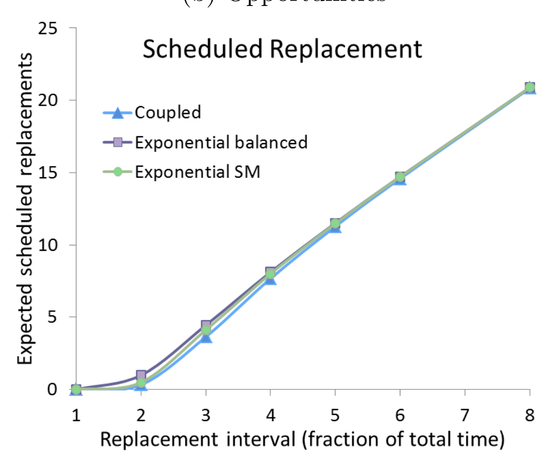
Figure 11 shows the relative errors associated with those approximations. One can observe that as long as the failures are relatively infrequent in comparison to the maintenance intervals, both exponential approximations of the opportunity work reasonably well. When replacements are less frequent, the balanced approach generally over-predicts both failures and opportunities, while the scale-matched method actually under-predicts failures while



(a) Failures



(b) Opportunities



(c) Scheduled Replacement

Figure 10: Expected failures (top), opportunities (middle), and scheduled replacements (bottom) for three identical components following Weibull with $\beta = 3$ and $\theta = 1$ for time $T = 4$ as functions of the number of scheduled replacements. The coupled model is compared to the sum of three component models with opportunities approximated by exponential distributions. The scale is found either by balancing total opportunities and failures ("balanced"), or by matching the scale of the distribution with the total number of expected failures during the full replacement interval ("SM.")

over-predicting opportunities. As a result, there is no assurance that the prediction of the scheduled replacements by the balanced method is optimal: one can observe in Fig. 11 that scale-matching actually provides a slightly smaller error. Next, we explore the possibility of selecting Weibull shape parameters for modeling opportunities in order to further improve the accuracy of the approximation, effectively trying to generalize Eq. 9 for the components with different shapes as well as other distributions.

4.2 Proposed approach: Winning Race ratio

We can still calculate the opportunity for the first component (that is, the combined chances that one of other components fails) using the general formula for the cumulative distribution $\tilde{F}_1(t)$:

$$\tilde{F}_1(t) = 1 - \prod_{i=2}^{n} R_i(t) \tag{10}$$

Taking a derivative of Eq. 10 we can obtain the corresponding probability density function $\tilde{f}_1(t)$ and evaluate the chances of the opportunity as

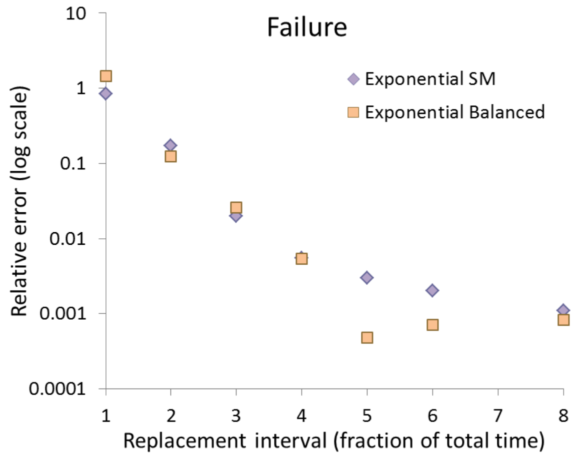
$$O_1(s) = \int_0^s R_1(t)\tilde{f}_1(t)dt \tag{11}$$

A direct use of this formula can be complicated by the fact that for a large number of components the integration can be somewhat involved. However, the main difficulty is that the formulae provide only the odds of the first action. As renewals take place, the schedule of replacements can change (see Figure 6), so to calculate the mean number of failures or opportunistic maintenance for some interval T_w (say, the warranty period), either the renewal Eqs. 6, 7 need to be solved, or a simulation needs to be used since the number of maintenance intervals that would "fit" into T_w is unknown *a priori.*

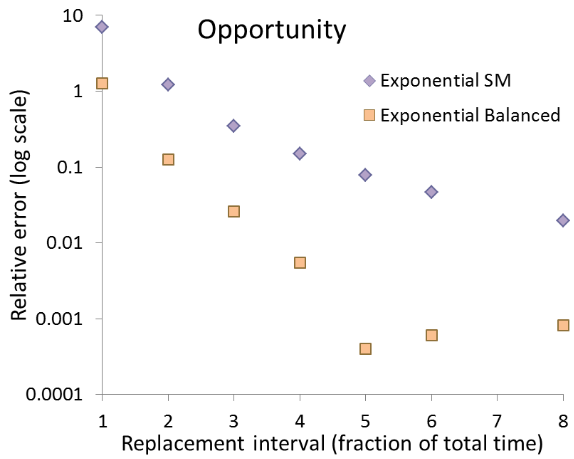
There are two distinct possibilities for an opportunity to occur for a given interval s (see Fig. 12):

A: The component would not fail on its own during the interval s, while some other component does fail during this interval. This probability is easily obtained using Boolean operations if failures are independent and individual distributions are known.

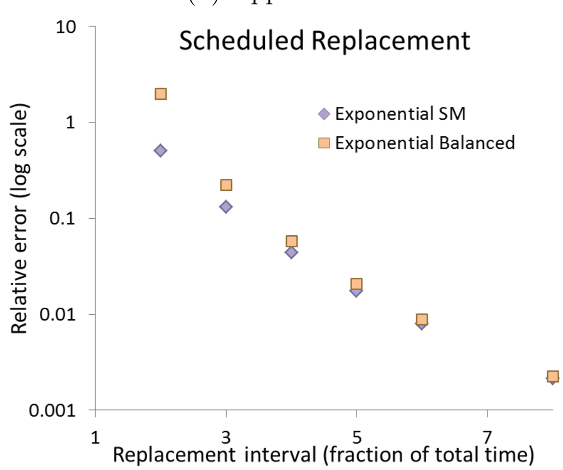
$$A_1(s) = R_1(s)\tilde{F}_1(s) \tag{12}$$



(a) Failures



(b) Opportunities



(c) Scheduled Replacement

Figure 11: Absolute values of relative errors due to approximation of opportunities using exponential distribution. Logarithmic scale is used for the errors due to large variability. Failures (top), opportunities (center), and scheduled replacement (bottom) are evaluated for both balanced and scale-matched (SM) approximations.

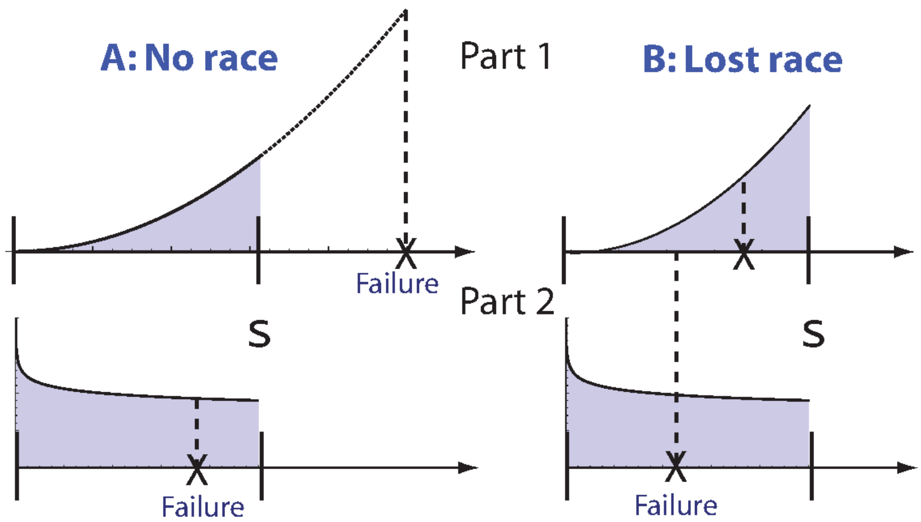


Figure 12: Two cases for part 1 losing a "race". A - Part 1 does not enter the race, B - Part 1 enters the race but loses it to some other part (Part 2)

B: The component would fail on its own during interval s, but some other component also fails earlier during this interval. In other words, the component loses the race to failure to another component (in the first scenario the component does not enter the race at all). If we denote the odds of this component's "losing" this race to the other component with γ_1, the following relationship can be written:

$$B_1(s) = O_1(s) - A_1(s) = \tilde{F}_1(s)F_1(s)\gamma_1(s) \tag{13}$$

Substituting and rearranging, the expression for the "odds of the component losing the race" is

$$\gamma_1 = \frac{\int_0^s R_1(t)\tilde{f}_1(t)dt - R_1(s)\tilde{F}_1(s)}{\tilde{F}_1(s)F_1(s)} \tag{14}$$

The second scenario is less likely to occur for the practical range of values, but its probability is non-negligible. If general rules regarding the chances of losing or winning the race are developed, then a good approximation of those chances will facilitate finding an equivalent function that represents the action of many components simultaneously. Since the second scenario is relatively less likely, and the odds of first scenario are easy to compute, the overall accuracy is expected to be quite good.

4.2.1 Odds of "winning" for Weibull functions

One can observe that for Weibull distributions, the smaller the shape function, the more likely the race will be "won." This dependency can be captured parametrically: Figure 13 shows γ, the chances of "winning" the race, for the distribution where the first component has the shape parameter $\beta_1 = 1$, while the shape of the opportunity varies (the scale parameter is fixed).

Effectively, if competing failures both occur during a given interval, the more distribution is skewed toward the beginning of the interval, the larger the chances that this distribution will win the race.

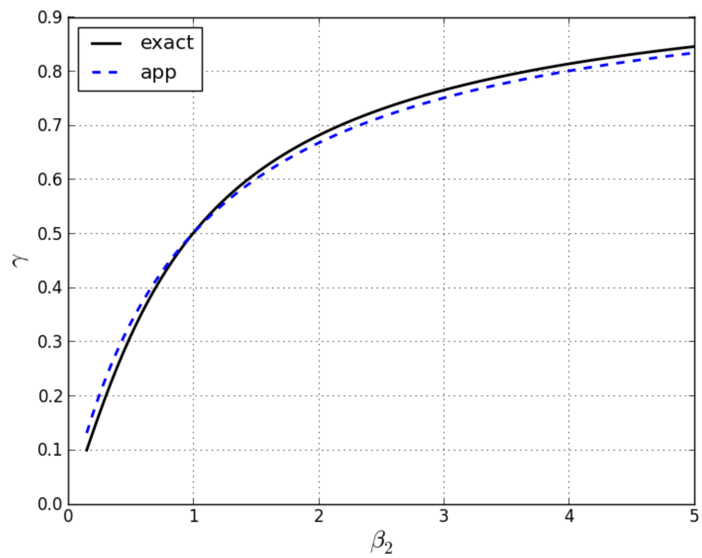


Figure 13: Shape sensitivity of γ_{12} for one of the functions with $\beta_1 = 1$ and varying β_2 of the other function. The scale $\theta = 5$ for both functions. Asymptotic approximation (Eq. 19) is shown by the solid line.

What is more remarkable is that, for the practical range of scale parameters (which is unlikely to be less than $2.5s$, as this would lead to too many failures), γ is quite insensitive to the scale parameter, and can be well approximated by a constant value that is a function only of the respective shape parameters. In particular, this implies that in that range γ, as defined in Eq. 13, can be considered independent of s. In fact, this scale independence for Weibull distributions can be confirmed using asymptotic considerations, as shown next.

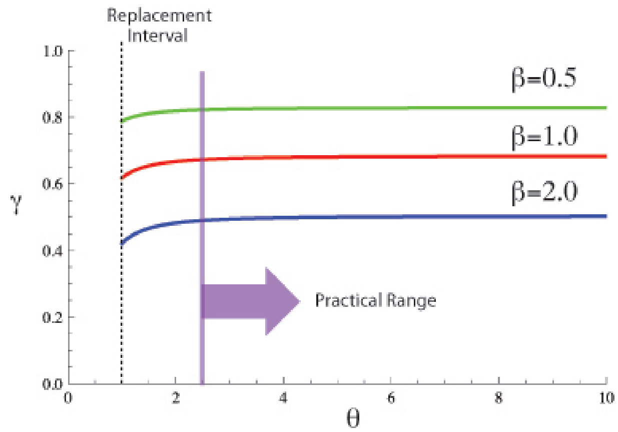


Figure 14: Scale sensitivity for race ratio γ when the first component follows Weibull distribution with shape $\beta = 2$ and scale $\theta = 5$ for different shape values. The replacement interval is unity.

4.2.2 Asymptotic Considerations

Let us consider s to be small as compared to the the failure scales θ_i, so we can introduce small parameters $\epsilon_i = \left(\frac{t}{\theta_i}\right)^{\beta_i} \ll 1$ for $0 \leq t \leq s$ and evaluate γ:

$$\gamma = \frac{-\int_0^s R_1(t)\frac{dR_2(t)}{dt}dt - R_1(s)(1-R_2(s))}{(1-R_1(s))(1-R_2(s))} \tag{15}$$

The first-order terms were studied for highly reliable systems [32, 30, 8], but they are not sufficient to evaluate γ. Let us include the second-order terms with respect to the introduced small parameters and evaluate the ratio γ. First, we can observe that $R_i = e^{-\epsilon_i} \approx 1 - \epsilon_i + \frac{\epsilon_i^2}{2} \ldots$. The integral in the nominator can be evaluated with accuracy up to the second order terms:

$$\begin{aligned} -\int_0^s R_1(t)\frac{dR_2(t)}{dt}dt &\approx \int_0^s \left(1 - \epsilon_1 + \frac{\epsilon_1^2}{2}\right)\frac{\beta_2}{t}\left(\epsilon_2 - \epsilon_2^2\right)dt \approx \\ \approx \int_0^s \frac{\beta_2}{t}\left(\epsilon_2 - \epsilon_2^2 - \epsilon_1\epsilon_2\right)dt &= \int_0^s \frac{\beta_2}{t}\left(\frac{t^{\beta_2}}{\theta_2^{\beta_2}} - \frac{t^{2\beta_2}}{\theta_2^{2\beta_2}} - \frac{t^{\beta_1+\beta_2}}{\theta_1^{\beta_1}\theta_2^{\beta_2}}\right)dt = \\ = \frac{s^{\beta_2}}{\theta_2^{\beta_2}} - \frac{s^{2\beta_2}}{2\theta_2^{2\beta_2}} - \frac{\beta_2}{\beta_1+\beta_2}\frac{s^{\beta_1+\beta_2}}{\theta_1^{\beta_1}\theta_2^{\beta_2}} &= \epsilon_2 - \frac{\epsilon_2^2}{2} - \frac{\beta_2}{\beta_1+\beta_2}\epsilon_1\epsilon_2 \end{aligned} \tag{16}$$

Similarly, neglecting the terms that are higher than the second order in the remaining terms in Eq. 15:

$$R_1(s)(1 - R_2(s)) \approx \epsilon_2 - \epsilon_1\epsilon_2 - \frac{\epsilon_2^2}{2} \tag{17}$$

$$(1 - R_1(s))(1 - R_2(s)) \approx \epsilon_1\epsilon_2 \tag{18}$$

Substituting these expressions into Eq. 15, and simplifying, we obtain the following asymptotic expression:

$$\gamma \approx \frac{\beta_1}{\beta_1 + \beta_2} \tag{19}$$

Figure 13 shows the numerical accuracy of this approximation.

This observation leads to the following strategy for deriving an equivalent Weibull distribution that represents the opportunity:

1. Evaluate the chances of "winning" the race for one of the other components:

$$\hat{F}_1(s) = 1 - \prod_{i=2}^{n} (1 - \gamma_{1i} F_i(s)) \tag{20}$$

   Here pair-wise winning odds, γ_{1i}, are calculated based on some reference scale parameters, and so can be pre-calculated (if only Weibull functions are involved, then Eq. 19 can be used).

2. Divide the chances obtained in the previous step by $\tilde{F}_1(s)$ (see Eq. 10) to yield an estimate of the combined winning ratio $\hat{\gamma}_1(s) = \frac{\hat{F}_1(s)}{\tilde{F}_1(s)}$.

3. Find the appropriate shape parameter $\hat{\beta}_1(s)$ given $\hat{\gamma}_1(s)$ (for Weibull distributions Eq. 19 can be used).

4. Determine the scale parameter by matching the chances of failure for the total interval $\tilde{F}_1(s)$ given the shape $\hat{\beta}_1(s)$.

The key consideration is that if renewal of the system occurred, then the remaining interval is even smaller as compared to the scales of the failure functions, so the asymptotic assumptions hold.

4.3 Results

4.3.1 Testing the procedure: Example 2

We can note that the smallest non-trivial number of components is three, as for two-component systems opportunity is provided by another component, and there is no need to combine the distributions. As described in example 1, if all three components are identical then the combination is trivial. Let us therefore consider a system where two components are identical, but the third component is distinct: $\beta_1 = \beta_2 = 4$, $\theta_1 = \theta_2 = 3$; $\beta_3 = 2$, $\theta_3 = 5$. Then the opportunity for the third component follows Weibull distribution with $\beta_{1o} = 4$ and calculated using Eq. 9 $\theta_{1o} \approx 2.5227$. For the other two components we use the developed method, and by then compare the results for the first component by evaluating the coupled model, subtracting the results for the third component, and dividing the results by two (since the first two components are identical). For the total time $T = 5$, the relative errors are shown in Fig. 15, demonstrating that for a broad range of replacement intervals, $s < T/2$ are not exceeding 1%, which for most of the application is sufficient. The natural question arises regarding the relative importance of the shape selection, as one can envision the possibility that for any shape parameter, the accuracy will be reasonable as long as the scale-matching is performed, and perhaps other shape parameters might provide even better accuracy. The results shown in Fig. 16 directly address this question by varying the shape parameter parametrically for a fixed replacement interval $s = T/5 = 1$, and matching the scale for every shape. One can observe that the shape selected based on the proposed procedure $\beta \approx 4.2$ is indeed quite close to optimal, and the sensitivity with respect to the shape parameter is not trivial. We note that the perfect match would require that all three curves in Fig. 16 intersect in a single point with the zero ordinate. It is also interesting to observe a non-linear dependence of the failure prediction errors (as exponential distribution can provide a smaller error that $\beta = 1.5$.)

4.3.2 Example 3: Weibull distributions with different shapes

Let us consider a three-component system where all three components have different shapes, and check that the developed procedure provides reasonable accuracy. The first component

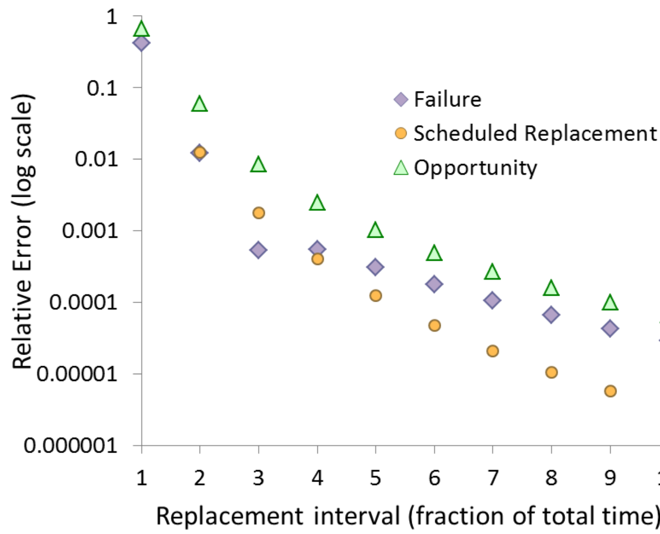


Figure 15: Relative errors for Case 2 as a function of the number of replacement intervals per time $T = 5$

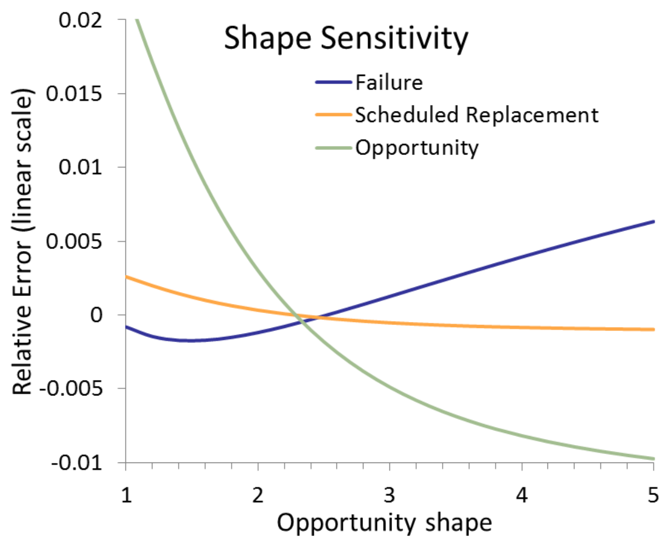


Figure 16: Sensitivity of the errors for $s = T/5 = 1$ for Case 2 as a function of shape parameter β for opportunity. Dashed lines represent the shape selected in accordance with the matched γ ratio.

follows Weibull distribution, with $\beta_1 = 3$ and $\theta_1 = 4$, while the maintenance interval $s = 1$. The opportunity is provided by the failures of two other components that have the following Weibull parameters: $\beta_2 = 2$, $\theta_2 = 5$, $\beta_3 = 4$, $\theta_3 = 3$.

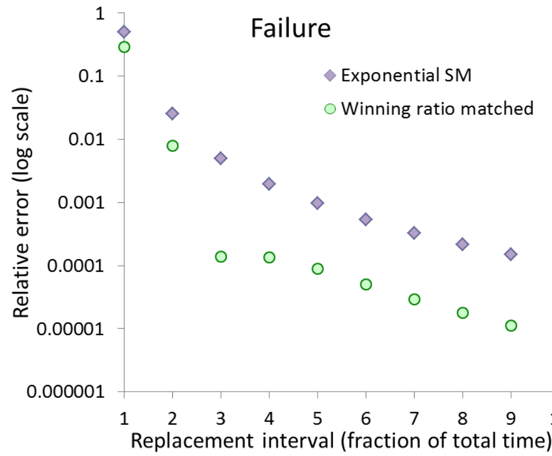
We can test the procedure by estimating the total number of events of interest for the time interval $T = 5$. The SPN models described in Figures 8, 9 have been used to independently verify the procedure, but the results presented here are obtained using the finite-difference method due to their superior accuracy.

One can observe (see Figure 17) that up to $s = T/3$ the accuracy of the method is quite good, and provides about an order of magnitude of improvement in terms of error with respect to the use of exponential distribution, so for smaller replacement intervals s exponential distribution can be sufficient as well.

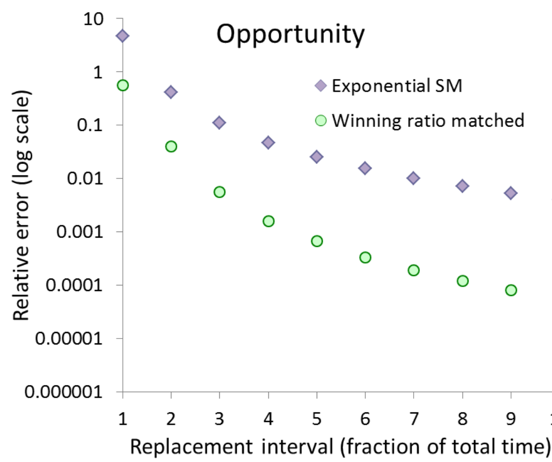
Let us now compare the proposed model with more traditional fits, such as matching the first two moments of the target distribution. Left sides of Figures 18, 19, and 20 show the corresponding probability density functions, and one can observe that the proposed distribution *does not* match the target distribution over the whole range of values as well as the more traditional fits. However, the situation is reversed when we focus our attention on the portion of the distribution that is most relevant to maintenance (see the right sides of Figures 18, 19, and 20). In other words, one can conclude that the described procedure is effectively a tail-fitting one, and some potential for similarities with the second theorem in extreme value theory [2] can be further explored. The practical implications of the difference are not negligible. Indeed, for the interval $s = 1$, while using the distribution with the parameters obtained by matching the moments provides a reasonable prediction of scheduled maintenance and failures, it leads to a 4.89% underestimation of the number of opportunities for $T = 100$.

4.3.3 Example 4: Lognormal distributions

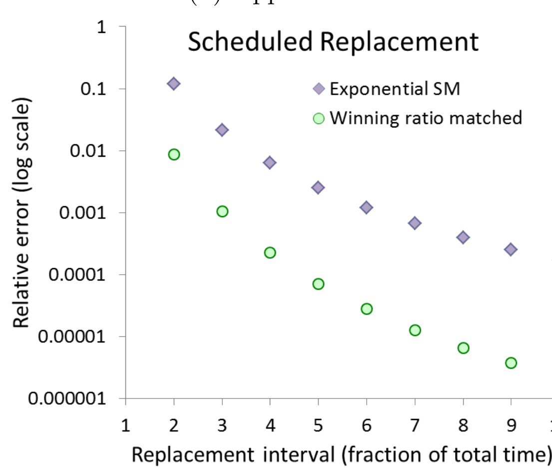
Finally, let us consider a situation where three components follow lognormal distribution with the following parameters: $\mu_1 = \log 2.5$, $\mu_2 = \log 4$, and $\mu_3 = \log 5$; $\sigma_1 = 0.4$, $\sigma_2 = 0.6$, and $\sigma_3 = 0.8$. As in the previous examples we considered the importance of estimating the



(a) Failures

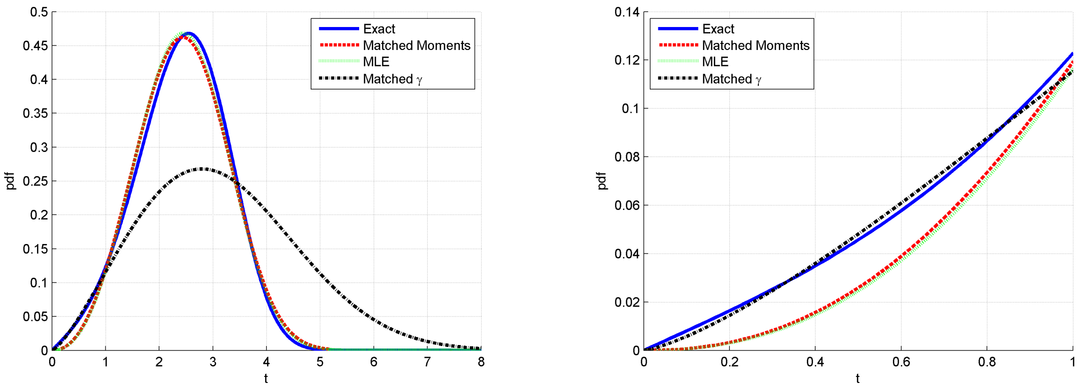


(b) Opportunities



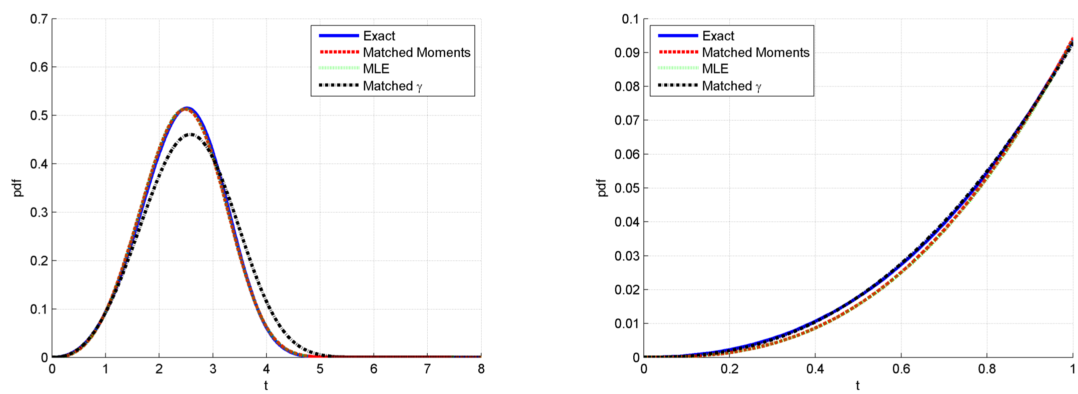
(c) Scheduled Replacement

Figure 17: Example 3: failures follow Weibull distributions (all three components are different). Comparison of relative errors (in logarithmic scale) obtained using component models with opportunities represented using Weibull distribution with matched γ and exponential distribution with the matched scale. Total time $T = 5$.



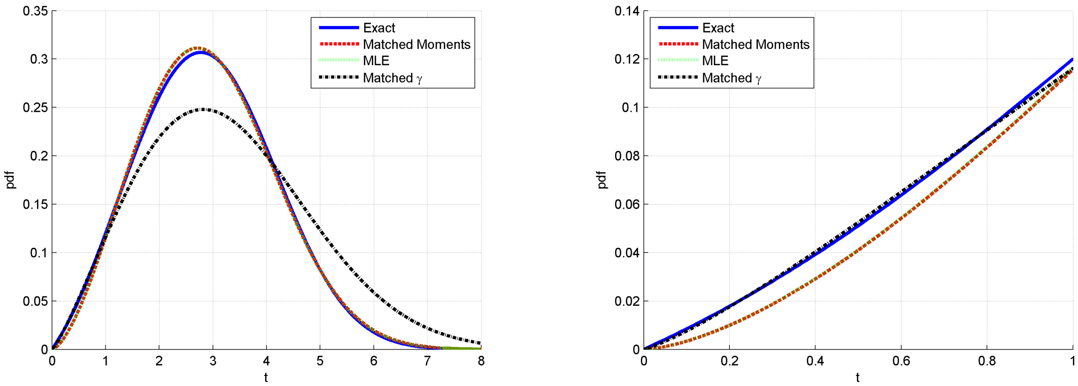
(a) Full (b) Left Tail

Figure 18: Probability density function (PDF) of opportunity for the first component, and its approximations using matching the first two moments, maximum likelihood estimate, and the current procedure that matches weighted γ. Full PDF (left), the left tail (right) $0 \leq t \leq 1.1$



(a) Full (b) Left Tail

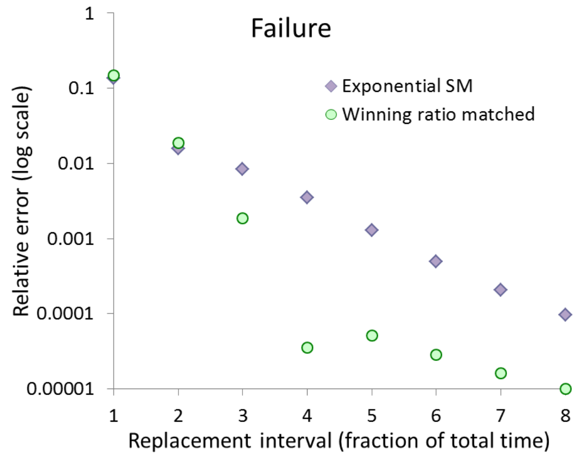
Figure 19: Probability density function (PDF) of opportunity for the second component, and its approximations using matching the first two moments, maximum likelihood estimate, and the current procedure that matches weighted γ. Full PDF (left), the left tail (right) $0 \leq t \leq 1.1$



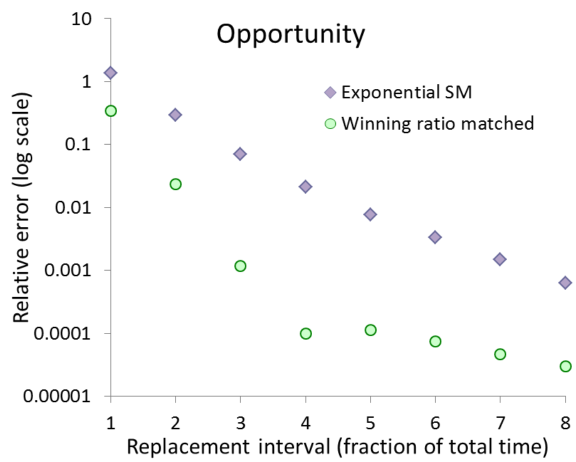
(a) Full (b) Left Tail

Figure 20: Probability density function (PDF) of opportunity for the first component, and its approximations using matching the first two moments, maximum likelihood estimate, and the current procedure that matches weighted γ. Full PDF (left), the left tail (right) $0 \leq t \leq 1.1$

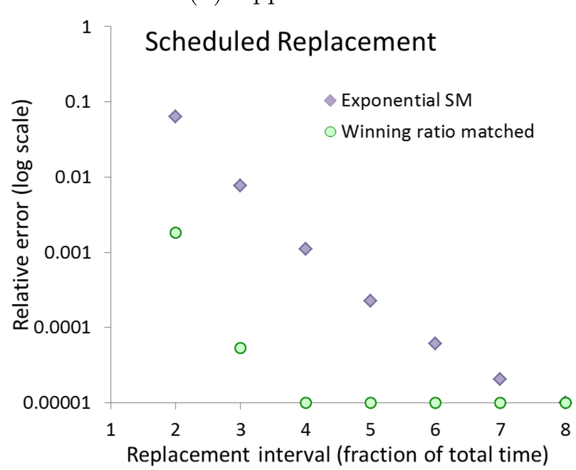
Weibull shape, and compared the developed procedure with the exponential representation of the opportunity obtained by scale-matching (similar to the case with Weibull distribution, about an order of magnitude improvement is obtained by matching the winning ratio). The total time horizon interval $T = 4$ and use finite-difference solutions for Eqs. 6, 7. The summary of the results is shown in Figure 21, demonstrating the relative importance of the shape selection. Note that the shape was selected based on $s = 1$ which is equivalent to four replacement intervals for the lifetime T of the system, and explains why the accuracy is relatively better for that particular segment. One could adjust the corresponding Weibull shape for each segment s, but that is clearly not necessary as the accuracy is quite good already for a wide range of replacement intervals. As expected, for small replacement intervals, exponential representation of the opportunity suffices.



(a) Failures



(b) Opportunities



(c) Scheduled Replacement

Figure 21: Example 4: failures follow log-normal distributions. Comparison of relative errors (in logarithmic scale) obtained using component models with opportunities represented using Weibull distribution with matched γ and exponential distribution with the matched scale. Total time $T = 5$.

CHAPTER V

VERIFICATION OF THE COMPONENT BASED MODELING METHODOLOGY

5.1 Mid Size Model: 2-Component Type Model

To assess the ability of the component-based methodology to appropriately model a system with the level of complexity comparable to that of the gas turbine engine maintenance process a mid size model that presents the same phenomena but on a smaller scale is introduced. The advantage of studying this mid size model is that while complex enough to present the phenomena of interest is still small enough to allow for a global model of the system to be created. This global model can then be used as a benchmark against which the results of the component-based model simulation can be compared.

The system consists of four types of components. There are five units of each component type. Each component type has a single failure mode modeled by a Weibull distribution. The maintenance policy replaces a component at failure (corrective maintenance) and includes two maintenance levels (preventive maintenance): "minor" and "major", the maintenance schedule is such that every other maintenance is major. Type-1 and type-2 components are replaced during every minor and major maintenance, while type-3 and type-4 components are replaced only during the major maintenance. Furthermore, failure of a component of the first or second type leads to an opportunistic minor maintenance (and therefore shifting forward the overall schedule) while failure of a component of the third or fourth type component only leads to the major maintenance. Table 1 summarizes the parameters of the system. The objective of the modeling is to obtain the expected number of events of interest, namely: failure replacements, preventive replacements and opportunistic replacements. The time horizon for the system is $L_s = 4$[time-units]. Figure 22 shows the global coupled model and Fig. 23 shows the component model for a component of type 2.

To reduce the size of the global model (shown in Fig. 22) the usage of colors is introduced

type	N_{C_i}	scale	shape	T_{PM}
1	5	8	2	1
2	5	2.5	4	1
3	5	12	2	2
4	5	4	4	2

Table 1: Summary of parameters of mid-size model

here. Components of each type can have different failure distributions (denoted by different colors). The model has three branches, on the left side two identical branches are used to represent the state of the components and on the right portion of the model a branch modeling the scheduler of the maintenance is used.

The component-state branch uses one token to represent a component (hence each branch has five tokens representing the component of one type (either one or three) and five components (of a different color) representing the components of the other type (either two or four)). Each token can leave the operating state due to either a failure or a scheduled maintenance action. The maintenance action can be triggered by two different actions: either the time of preventive maintenance has arrived or failure of another component of the appropriate type has provided an opportunity.

The coupling mechanism between branches is modeled through the use of enablers between places and transitions on different branches. Enablers and inhibitors are depicted by an arc with a small empty circle on the end of the arc next to the transition. When the arc is an enabler a negative weight is given to it.

For instance, the triggering of an opportunistic replacement of the components of type-1 and type-2 due to failure of a single component of type-1 goes as follows: At the time of failure, the token representing the failed component moves from the operating place to the failure place and it remains there for a time interval small enough to trigger the actions described next. When the token arrives to the failure place the upper transition in the scheduler branch is enabled and the token on the scheduler branch moves from the scheduler place to the minor place. This in turn enables the scheduled transition of the component-state branch moving all the tokens remaining in the operating place to the scheduled place

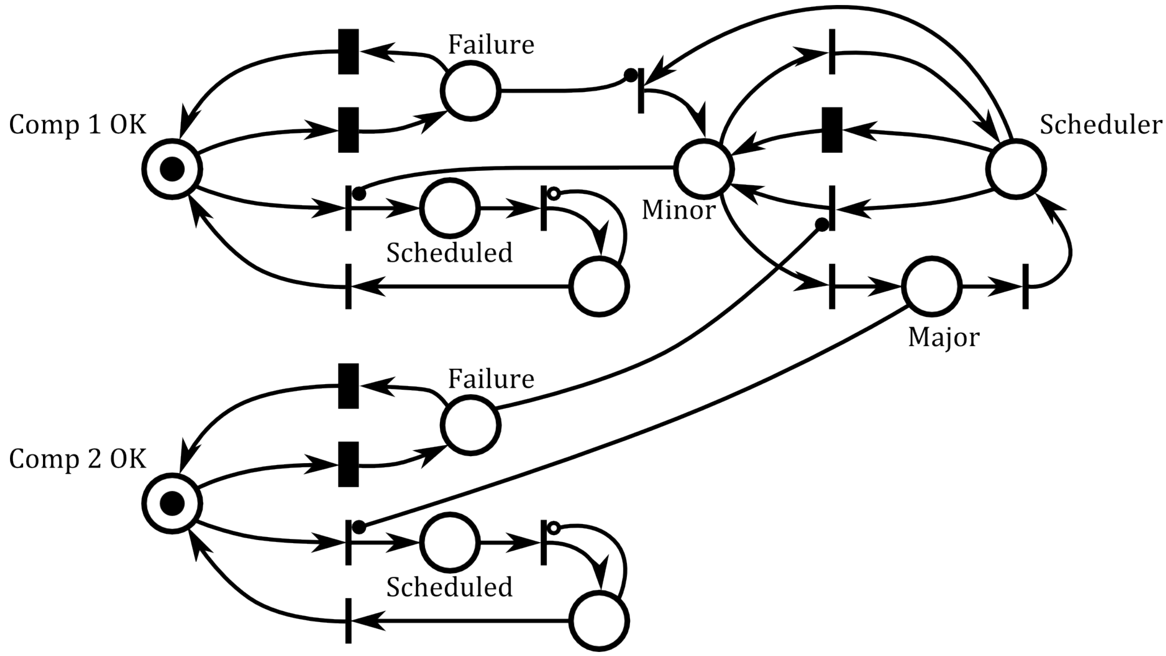


Figure 22: Midsize coupled model

(triggering an opportunistic replacement). Furthermore, the token in the scheduler branch has changes color appropriately moving the clock ahead in time for the major inspection as well.

The scheduler branch has only one token (which represents the maintenance-clock). This token can have two colors 0 - meaning it is in the first cycle (before the major maintenance) and 1 during the second cycle (before the major maintenance). When the first cycle ends, and the scheduler token moves to place "minor" it cannot move further down to place "major" as the corresponding transition does not have a matching policy for color 0, so the token returns to the scheduler position with the color changed to 1, so that during the next cycle it can move to the major place.

We note that here the enablers and immediate transitions have to be timed appropriately to allow enough time to trigger appropriate actions.

The instantiation of the component-based model template for the components of type-1 is shown in Fig. 23. When compared with the global model it can be seen that the scheduler branch now includes two extra transitions that take the scheduler token from the scheduler place to the minor place. One of this transitions models the opportunities of a

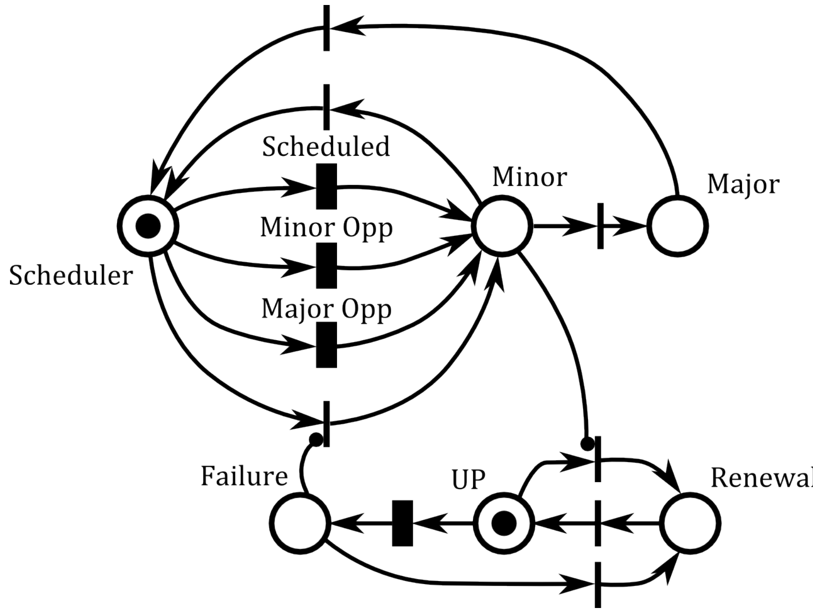


Figure 23: Midsize uncoupled model

minor inspection that the rest of the system causes on a component of type-1 and the other represents the opportunities of a major inspection provided by the rest of the system on a component of type-1. The component-state branch has the same states but it only needs a single token now because this is a model of any component of type-1.

5.2 Simulation results

A Monte Carlo simulation of both the coupled model and the component-level models (for each component type) is performed and the number of failures and scheduled replacements is recorded (the number of replacements due to opportunity can be inferred from the number of failures). The modeling of the time-to-next-opportunity distribution is made based on the three different methods outlined in Ref. [81]. The results of the simulation are summarized in the table shown in Fig 24.

It is first concluded that in this case uncoupled models can overestimate failures and furthermore, this overestimation can be significant (approximately 10% in this example). The presence of this effect will occur for any system where an age-based replacement policy is used, the components have failure modes with an increasing failure rate and there is economic dependence among them. This is because in the real system the components

	Failure 1	Failure 2	Failure 3	failure 4	Sch 1+2	sched 3+4
Coupled (system) model	0.283	0.426	0.236	0.444	40.744	19.810
Component Uncoupled	0.311	0.505	0.263	0.555	39.187	19.206
% error	9.97%	18.64%	11.75%	25.23%	-3.82%	-3.05%
Inferred opportunity	0.205	0.235	0.135	0.133	88.351	54.411
% error	-27.65%	-44.67%	-42.91%	-69.93%	116.85%	174.67%
Scaled matched opp	0.267	0.383	0.202	0.345	50.014	28.057
% error	-5.58%	-10.03%	-14.19%	-22.32%	22.75%	41.63%
Winning ratio matched	0.284	0.426	0.233	0.441	40.781	19.896
% error	0.47%	0.10%	-0.95%	-0.56%	0.09%	0.44%

Figure 24: Mid-size model summary of Results

are replaced more often due to opportunities provided by the failure of the other relevant components.

Second, the importance of running several iteration of the simulation (until parameter convergence is achieved) is clearly seen by noting the large errors obtained using the inferred opportunities from the uncoupled model (which would correspond to the first iteration).

Finally, the matching shape procedure introduced in Ref. [82, 81] provides about an order of magnitude improvements in accuracy when compared to the other methods.

CHAPTER VI

EVALUATION OF MAINTENANCE COST FOR A GAS TURBINE ENGINE

6.1 Introduction

The modeling of the maintenance process of a gas turbine engine is of both great relevance and complexity. Relevance, because as the power generation market becomes more competitive, plant owners try to maintain their profit level by reducing the cost of ownership. Maintenance cost account for a large part of the cost of ownership [48]. Furthermore, the industry is currently undergoing a fundamental shift where manufacturers go from selling products to providing services [34, 88]. To minimize maintenance costs throughout the product life cycle engineers must achieve a better understanding of the maintenance process and one of the avenues to do this is through the construction of more accurate models.

A gas turbine engine can be considered a macro-system composed by several subsystems which in turn are composed by several components. To estimate the reliability of the engine it is necessary to consider the reliability of the several subcomponents. Depending on the desired level of resolution of the study to be performed the number of components that needs to be considered can range in the thousands.

The failures for a gas turbine engine that can produce forced outage of the gas turbine are considered to be of particular importance. An *outage* can be either: the shutdown of a generating unit for emergency reasons, or the condition in which the equipment is unavailable as a result of an unanticipated breakdown. An outage is *forced* if it could not reasonably be delayed beyond some fixed amount of time (usually 48 hours) after identification of the problem. To simplify the modeling process this work will focus only on the set of failures that leads to a forced outage.

While this problem has not been dealt under the current assumptions or with the same objectives in the existing literature an overview of related work is provided next. Ref.

[39] uses a Monte Carlo simulation approach to draw a correlation between a modified TRL (Technology Readiness Level) and downtime of a gas turbine engine. With such a model the authors can quantify the risk between different gas turbine engines. In that work the failures follow a Weibull probability distribution model and the time-to-repair follow lognormal distribution. The maintenance model makes the following assumptions: the failures of each component are independent, the components are connected in series (from the point of view of system reliability), and scheduled downtime is not considered. The system is considered to be composed of 10 subcomponents. Ref. [1] fits a generalized proportional intensities model to gas generator failure and maintenance data and then use the resulting model to estimate an optimal preventive maintenance interval. The methodology used lumps together the failures due to different components and so it cannot be easily modified to study a different maintenance policy (for instance, one where different components can be subject to different inspection intervals). In Ref. [88] a discrete-event model is compared with an agent-based model of a fleet of turbofan gas turbines within the life cycle period extending from just after entry into fleet operation until just before retirement and disposal. The simulated model consisted of a fleet of 310 engines but since the focus of the work was on comparing the discrete-event modeling paradigm with the agent-based model paradigm not enough information is provided as to the level of detail in the modeling of each component. One of the conclusions of their work is that the discrete-event simulation approach scales badly with an increasing number of components. They attribute this to the need of keeping a central event list which becomes the choke-point of the simulation.

The following sections are organized as follows, first a description of the problem statement and the available data is presented. Then, the solution approach is described including the modeling assumptions. Finally the models constructed and the results of their simulation is included.

6.2 Problem Statement

A gas turbine engine manufacturer has provided failure and maintenance data for a set of components and associated failure modes for a gas turbine engine. This portion of the work has been conducted in collaboration with Siemens Gas Turbines [79]. Failure and maintenance data has been obfuscated for proprietary reasons. However, great care was taken to preserve the representative complexity and general trends of the problem.

Maintainability studies have, in general, the following objectives [43],

1. To guide and direct design decisions by considering all the maintenance tasks, with a prime objective of reducing the total maintenance tasks/time

2. To predict quantitative maintainability characteristics of a system

3. To identify changes required to a system's design to meet operational requirements by reducing maintenance/inspection times

In this study our main objective will be to predict the quantitative maintainability characteristics of the given system and maintenance policy.

Specifically, a maintenance model is created such that the number of corrective maintenance and preventive maintenance actions (throughout the life of the system) for each relevant component can be estimated. This information is then used in conjunction with the costs associated to each action to estimate the life-cycle maintenance cost of the system studied. Fig. 25 depicts the followed procedure.

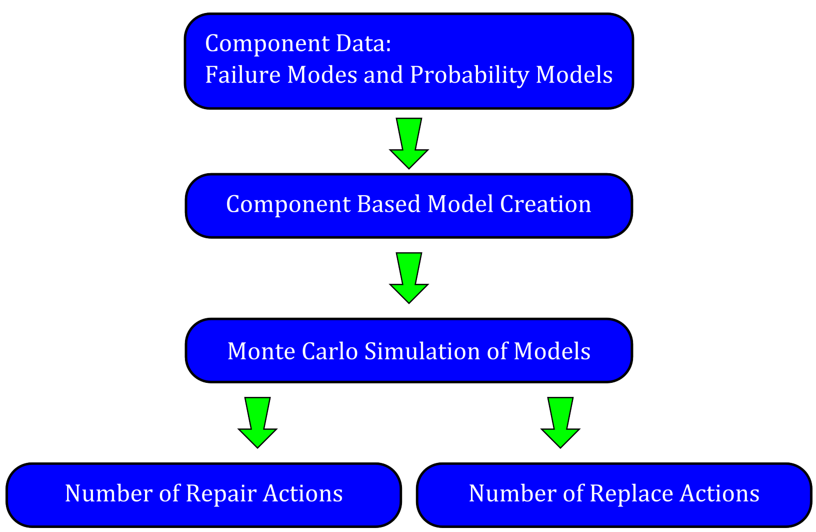


Figure 25: Solution Algorithm

6.3 System Description

The original data provided from the manufacturer corresponds to a system comprised of a set of 18 component types. Each component type is characterized by: a hard-life L_i, up to three failure modes, and a coupling mechanism between failures where failure of one component induces failure of other, a time-to-scrap model, an inspection interval and a coupling mechanism that triggers an inspection before the inspection time if relevant components of other types have failed (opportunistic inspection/maintenance). Table 2 shows the normalized replacement and repair cost per part, inspection interval and part life.

Table 2: Component type maintenance policy data

Comp. Type	Repl. cost per part	Repair cost	Inspection Interval	Part Life	Number in Set
1	0.4490	0.1049	1	3	16
2	0.3245	0.1512	1	6	16
3	0.8740	0.3745	1	6	16
4	0.0802	0.0000	4	4	16
5	0.7444	0.2545	1	3	16
6	2.8154	0.0000	1	1	1
7	0.1203	0.0595	2	4	72
8	0.1052	0.0233	2	4	66
9	0.0828	0.0250	2	6	112
10	0.0887	0.0139	4	4	84
11	0.3124	0.1782	2	4	32
12	0.3297	0.2451	1	4	24
13	0.2674	0.1396	6	6	16
14	0.4218	0.1428	4	8	14
15	0.0273	0.0125	1	4	48
16	0.0287	0.0000	4	4	48
17	0.0269	0.0000	4	4	32
18	0.0288	0.0000	6	6	28

There are three possible inspection intervals corresponding to three "levels" of inspection. If the level-0 inspection is performed every T units then the level-1 inspection is performed every $2T$ units and the level-2 inspection is performed every $4T$ units. It is assumed that higher levels of inspection trigger lower levels of inspection. Hence, the scheduled inspections would go (as time progresses) as follows: level-0, level-1, level-0, level-2, etc (See Fig. 26). A normalization has been performed so that 1 unit corresponds to the time interval of one level-0 inspection. The planning horizon is one system lifetime which corresponds to $8T$ normalized time units .

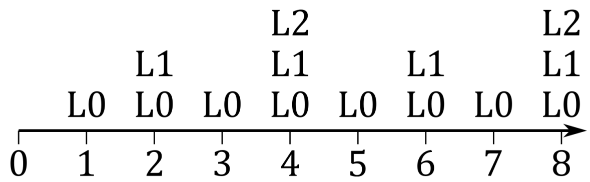


Figure 26: Scheduled inspections

Table 3 shows the parameters characterizing the failure modes for each of the components. The manufacturer has found that Weibull distributions with scale parameter η and shape parameter β adequately represent the failure behaviors being studied. It is important to note that failure data is not provided for all of the components (but it is sufficient to capture the forced outages).

Table 3: Forced Outage Distribution by component and failure mode type

Comp. Type	Failure 1		Failure 2		Failure 3	
	η	β	η	β	η	β
1	9.224005	2.2	–	–	–	–
2	27.22308	1.9	15.181506	1.2	–	–
3	–	–	–	–	–	–
4	–	–	–	–	–	–
5	–	–	–	–	–	–
6	–	–	–	–	–	–
7	46.52168	2.5	2493.104	1	83.50587	2.2
8	30.34957	2.5	95.9943	1.6	1741.0011	1.2
9	54.5105	2.5	–	–	–	–
10	86.0502	2.5	–	–	–	–
11	–	–	–	–	–	–
12	22.39089	2.4	1224.8666	1	–	–
13	29.40841	2.3	–	–	–	–
14	–	–	–	–	–	–
15	–	–	–	–	–	–
16	–	–	–	–	–	–
17	–	–	–	–	–	–
18	–	–	–	–	–	–

Once a component has reached its life it must be replaced. For some components, the replacement item comes from a pool of heterogeneous spare parts. Some of those spare parts are new and other have been used and then repaired. Due to this the manufacturer may choose to scrap a part even though it has not failed nor has reached its stated part life. To account for this a Weibull probability distribution model is used. Table 4 shows the parameters of the probability model (scale and shape), the mean of the distribution and the value of the cumulative distribution function at the time equal to the part life.

Table 4: Scrap Weibull probability model by component type

Comp. Type	η	β	$E[X]$	$P(X \leq \text{Part Life})$
1	24.00686	1.6	21.5239	0.03524
2	5.517648	2.1	4.8869	0.6965
3	0.904386	1.8	0.8043	1.0000
4	–	–	–	–
5	15.30319	2.3	13.5573	0.02329
6	–	–	–	–
7	1.6595	2.1	1.4698	0.9982
8	2.1559	1	2.156	0.8436
9	4.32276	1.5	3.9023	0.8051
10	5.15763	0.9	5.4268	0.5486
11	2.0949	1.9	1.8589	0.9672
12	2.2523	1.8	2.0030	0.9399
13	3.7717	2.4	3.3435	0.9525
14	1482.17	0.9	1.5595	0.009
15	–	–	–	–
16	–	–	–	–
17	–	–	–	–
18	–	–	–	–

With regards to the probability of the distribution being less than the part life we have two extreme behaviors. For some component types (1, 5, 14) the probability of it being scrapped before its life is very low and for others (3, 7–13) the probability of it being scrapped before its life is due is almost certain. An explanation for this behavior is not readily available but a possible alternative is that in the former case the ratio of new components to used/repaired components in the spare parts pool is very high and in the latter case the ratio is low and the repair process is unable to bring the components to a "like-new" condition. This behavior can be visualized in Fig. 27 where the probability density functions are plotted against the time going up to the life of the component type.

Component types 1, 5, and 14 are plotted in red; 3, 7–13 in blue and component type 2 in green.

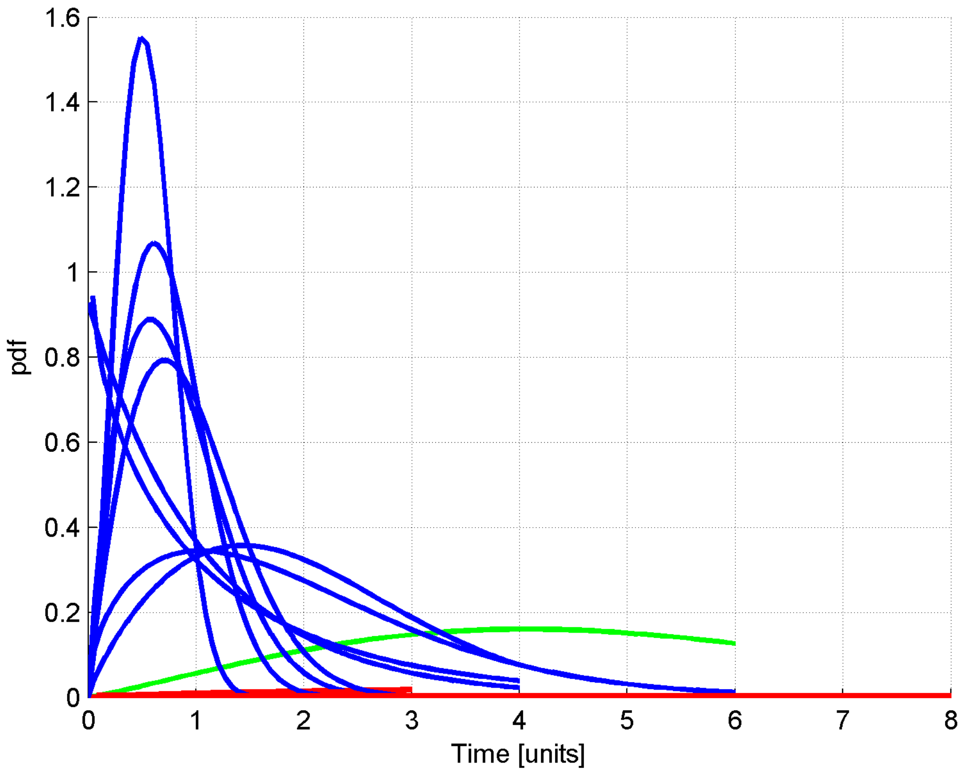


Figure 27: Scrap Weibull probability density functions

Another characteristic of the maintenance policy being modeled is that it considers opportunistic maintenance of some components when a failure in a specific mode of a given component occurs. Table 5 shows the failure modes and opportunistic actions they trigger. The table must be interpreted as follows: a failure of component type i in mode j triggers an opportunistic inspection of level k where k can be read from the table. For instance, according to the table, failure of component type 7 in mode 2 triggers an inspection of the type level-2.

Finally, there is a coupling behavior in some of the failure modes between some of the components. In particular this coupling implies that when a component of a certain type fails in a specific failure mode it induces a failure in some other components. The

Table 5: Opportunistic Maintenance Induced inspection by failure mode and component type

Comp. Type	Fail. Mode 1	Fail. Mode 2	Fail. Mode 3
1	0	–	–
2	0	1	–
3	–	–	–
4	–	–	–
5	–	–	–
6	–	–	–
7	–	2	2
8	–	2	2
9	–	–	–
10	–	–	–
11	–	–	–
12	2	2	–
13	–	–	–
14	–	–	–
15	–	–	–
16	–	–	–
17	–	–	–
18	–	–	–

appropriate maintenance action is then selected according to the maintenance policy of the component. This behavior is summarized in Table 6.

Table 6: Induced Failures

Failure of Comp. type	in Fail. Mode	induces failure in
7	3	9,12,13,14,16,17,18
8	3	9,13,14,17,18

6.4 Solution Approach

As stated previously, the objective of the model is to estimate the number of corrective and preventive maintenance events under two different maintenance policies. These estimates are used to evaluate the total maintenance cost ($C_{\text{m,policy}}$) of the policy which is calculated as:

$$C_{\text{m,policy}} = \sum_{i=1}^{n} N_i \left(c_{rpr,i} N_{rpr,i} + c_{rpl,i} N_{rpl,i} \right) \tag{21}$$

where N_i is the number of components of type i, $N_{rpr,i}$ and $N_{rpl,i}$ are the number of repair and replacement events of component i, and $c_{rpr,i}$ and $c_{rpl,i}$ are the repair and replacement cost per component for component i.

Before creating the models a normalization step was carried to obscure the origins of the data. After analyzing the data provided by the manufacturer the decision was made to focus on the following component types: 1-3,5 and 7-14. The remaining components were excluded from the analysis. As a result, the model considers 484 components.

Two component template models were created. One representing the perfect repair maintenance policy and one representing the minimal repair maintenance policy.

An experiment consist of the following steps:

1. For each component type,
   (a) Instantiate the component template with the data for the component type
   (b) Perform a Monte Carlo simulation of the instantiated component template without considering the opportunities (coupling due to exogenous effects). The number of replications is $N_{\mathrm{MC}} = 1E7$
   (c) Collect the estimated number of corrective maintenance and preventive maintenance effects.
2. With the collected data calculate the parameters of the probability model representing the coupling effects (opportunities)
3. Repeat step 1 but now including the coupling effects
4. Repeat steps 2 and 3 until the desired convergence is achieved.

6.5 Perfect Repair Modeling

6.5.1 Component Template

Figure 28 shows the general component model file for the perfect-repair simulation. This model is composed of three groups of nodes (branches) which interact among them by inhibitors/enablers arcs. The groups are called the *component state* group (at the top of the model), the *scrap-clock or maintenance* group (at the lower-left portion of the model) and the *scheduler* group (at the lower-right portion of the model). Each group has a token, so there are three tokens in this SPN model.

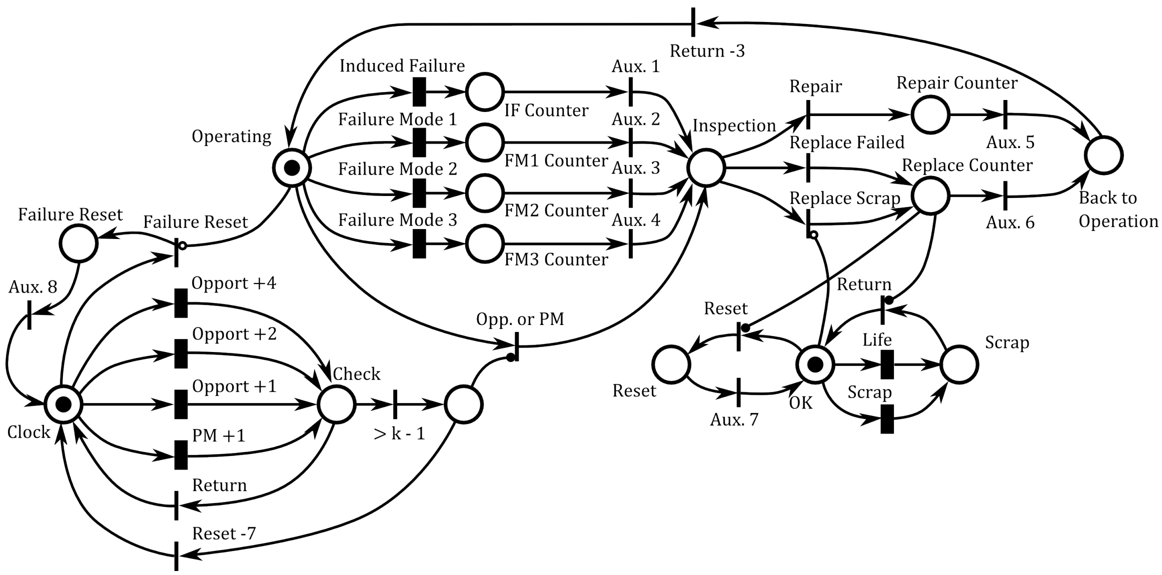


Figure 28: Perfect repair template

The "component-state" group captures all the possible states that the component can be at any given time (including Operating, Failure modes, etc). Recall that the token in the "component-state" network represents the current state of the component. There are two main states in this group: The *Operating* state and the *Inspection* state. From the operating state, the component can transition to the inspection state through five different paths: three possible failure modes, an opportunistic maintenance or inspection due transition, and an "induced failure" transition which corresponds to another component failing and hence, inducing a failure on the component being modeled. If the particular component being modeled has less than three failure modes, or it does not fails due to the failure of other component then the corresponding transitions are disabled (by setting them to type "None").

To show how the component-state network works consider the following cases:

Component fails before preventive maintenance/opportunity Assume the component fails according to failure mode two before the time of preventive maintenance

/ opportunity arrives. Then, the token moves from the *operating* place to the *FM2 Counter* place and its color is increased by two. The counter count is increased by one. After a small time interval has elapsed (ϵ) the token moves to the *Inspection* place. Consider that the token in the scrap-group is on the *OK* place. Then, only the *Repair* and *Replace Failed* transitions are enabled (due to the inhibitor on the *Replace Scrap* transition) and since the first one is fired after 2ϵ and the second one is fired after ϵ then the *Replace Failed* transition is fired and the token moves to the *Replace Counter* place. The counter is increased by one and after ϵ it moves to the *Back to Operation* place. Finally, after ϵ the token moves back to the *Operating* place, its color is reset to the color zero and the process starts again.

Time for inspection occurs before failure The scheduler-group token enables the *Opp or PM* transition which fires after a time of ϵ has elapsed. Due to this, the token in the *Operating* place moves to the *Inspection* place. Since the color of the token is still zero, the only transitions that can possibly be enabled are the *Repair* and *Replace Scrap* transition. If the scrap-clock group token is in the *OK* place then the *Replace Scrap* transition is inhibited and the *Repair* transition fires moving the token through the *Repair Counter* place (where the counter increases by one) to the *Back to operation* place. After ϵ the token moves back to the *Operating* place, its color is reset to the color zero and the process starts again.

The "scrap group" models the timing for the scrapping of the component. A component is always scrapped if it has reached its hard life. However, the maintenance policy also states that some of the components will be scrapped before reaching its hard life. This maintenance action is modeled using a Weibull model. The objective of the scrap-clock group is to model when a component that has not failed has to be scrapped. There are three states associated with this group: OK, Scrap, and Reset. This clock works as following: at initialization the token is in the *OK* place. At this point the *Life* and *Scrap* transitions are enabled and three possible paths are possible:

Life transition fires The component has reached its hard life and needs to be scrapped.

The token moves to the *Scrap* stopping the inhibition of the *Replace Scrap* transition on the component-state group. The token stays there until the token of the component-state group enables the *Return* transition by going through the *Replace Counter* place. The token then moves to the *OK* place and the process starts again.

Scrap transition fires The component has not failed but it will still be replaced. The token moves to the *Scrap* stopping the inhibition of the *Replace Scrap* transition on the component-state group. The same process as when the *Life* transition fires occurs.

Reset transition fires The component has failed and it needs to be replaced. The token in the component-state network has reached the *Replace Counter* place through the *Replace Failed* transition. Due to this the *Reset* transition is enabled and fired (after a time ϵ has elapsed) moving the token in the scrap-clock group from the *OK* place to the *Reset* place. The token remains there an interval of ϵ and then moves to the *OK* place and starts the process again.

The "inspection group" (scheduler) models the timing of the next inspection for the component. This group includes the following places: Clock, Check, and Inspection Due. To show how the scheduler network achieves its objective lets consider a component that has an inspection period of $4T$. The *PM +1* transition is set to T which is the base inspection period. A firing rule for the $> k-1$ transition is set only for color $k=4$. Consider now, that the *PM +1* transition fires. Then, the token is moved from the *Clock* to the *Check* place and its color is increased by one making the token color equal to two. Since the $> k-1$ has a single firing rule which is only set for color $k=4$ it is not enabled and the token stays in the place until the *Return* transition fires returning the token to the *Clock* place but now with color $k=1$. If this process repeats three more times then the token will reach the *Check* place with color $k=4$. At that point the $> k-1$ transition is enabled and immediately fired moving the token to the *Inspection due* place which enables the *PM or Opp* transition in the component-state group triggering an inspection of the component. The token on the scheduler network then moves to the *Clock* place through the *Reset -7* transition which changes its color back to zero and resets the scheduler.

Consider now the case where the token is in the *Clock* place with color $k = 1$. This models the situation where one base inspection has already occurred. Next, an opportunity occurs due to failure of component which is under a T hours inspection period. This opportunity does not trigger an opportunity of the component being analyzed but it does change the schedule of all the subsequent inspections. The scheduler network captures this by increasing the color of the token and resetting the clock on the transitions due to preventive maintenance or opportunities.

The perfect repair template collects results in 8 sensors associated with the following places: IF Counter, FM1 Counter, FM2 Counter, FM3 Counter, Inspection Due, Inspection, Repair Counter and Replace Counter.

The maintenance policy implemented replaces or repairs the component every time an inspection on the component is performed. The component is replaced if it has reached its life or if it needs to be scrapped and otherwise it is repaired. From the token point of view, a replacement is identical to a repair. The token returns to the operating place with a zero life (like a new component).

6.5.2 Verification of the template

Initial verification of the model is achieved by a step-by-step simulation run. The purpose is to check that the behavior of the model is as expected. This is important because SPN@does not directly implement priorities and care needs to be taken to ensure that the desired transition will fire first when two or more transitions are scheduled to fire at the same time. Instead, relative timing is used throughout the model to assure proper behavior.

For instance, consider the behavior of the "scrap group" (see Fig. 29). When the token arrives to the "Replace Counter" place and the token in the "scrap group" is in the "Ok" place the expected behavior is for the "Reset" transition to fire and then the "Aux. 6" in that order. If the order inverted then after "Aux. 6" transition fires the "Reset" transition will be disabled and the "Life" and "Scrap" transitions will not be disabled-enabled as they should. To ensure this, "Aux. 6" is set to fire after 2ε and "Reset" is set to fire after ε. The verification step confirms that this and other similar situations in the rest of the model are

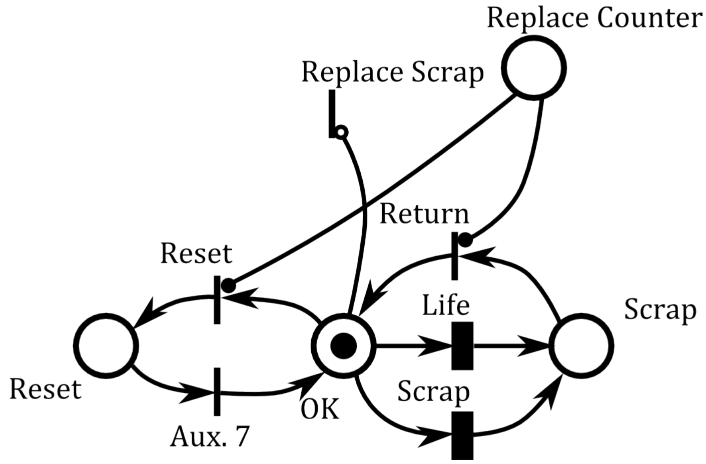


Figure 29: Screenshot scrap-clock group SPN@

being properly handled.

6.5.3 Simulation of the template: Perfect Repair Experiment

6.5.3.1 Uncoupled Simulation

In the uncoupled simulation everything but the opportunities caused by failures of other components are enabled. The results from the uncoupled simulation provide the starting point for the estimation of the parameters of the opportunities probability model.

The result for one of the experiments ran in the uncoupled configuration are shown in Table 7. Since the induced failures are not enabled the column corresponding to the induced failures has only zeros.

Table 7: Results of Uncoupled Experiment, per component. 1E7 replications

	FM1	FM2	FM3	Repair	Replace	Insp.	Ind. Fail.
C1	0.06016	0.0	0.0	5.90278	2.09801	7.94064	0.0
C2	0.01495	0.30760	0.0	6.48543	1.55598	7.71886	0.0
C3	0.0	0.0	0.0	1.97599	6.02401	8.0	0.0
C5	0.0	0.0	0.0	5.97251	2.027489	8.0	0.0
C7	0.008658	0.0064593	0.0049837	0.007006	3.99346	3.9804	0.0
C8	0.02509	0.024729	0.002738	0.55261	3.44812	3.94818	0.0
C9	0.005818	0.0	0.0	1.38843	2.61157	3.99419	0.0
C10	0.02964	0.0	0.0	1.0E-7	2.0001	1.9704	0.0
C11	0.0	0.0	0.0	0.12804	3.87196	4.0	0.0
C12	0.004588	0.006652	0.0	4.90276	3.09802	7.98954	0.0
C13	1.3234	0.0	0.0	4.0E-7	1.55433	0.2309	0.0
C14	0.0	0.0	0.0	0.9997	1.0003	2.0	0.0

The uncoupled simulation shows that the perfect repair policy is effective in maintaining the number of unscheduled outages of the system for all the component types except component type 13 which has a large likelihood of failing at least once throughout the system life.

Also, there is an imbalance in the number of events. Consider for instance, components type 1, 2 and 3. This component are under the same inspection schedule however the number of "inspection" events for each of them differs slightly (7.94064, 7.71886 and 8.0). This is due to the decoupling of each of the component simulation from the rest of the system in conjunction with the disabling of the transitions representing this coupling.

6.5.3.2 Uncoupled + Induced Failures Simulation

The results from the uncoupled simulation are used to estimate a model for the induced failure probability distribution. The induced failures are assumed to follow an exponential distribution. From the system architecture the induced failure behave as a series system. Hence, the rate parameter of the induced failure probability distribution is calculated as,

$$\begin{aligned}
\lambda_{9,\mathrm{if}} &= N_7\lambda_{7,\mathrm{FM3}} + N_8\lambda_{8,\mathrm{FM3}} \\
\lambda_{12,\mathrm{if}} &= N_7\lambda_{7,\mathrm{FM3}} \\
\lambda_{13,\mathrm{if}} &= N_7\lambda_{7,\mathrm{FM3}} + N_8\lambda_{8,\mathrm{FM3}} \\
\lambda_{14,\mathrm{if}} &= N_7\lambda_{7,\mathrm{FM3}} + N_8\lambda_{8,\mathrm{FM3}}
\end{aligned}$$

where,

$$\begin{aligned}
\lambda_{7,\mathrm{FM3}} &= \frac{N_{7,\mathrm{FM3}}}{T_{LC}} \\
\lambda_{9,\mathrm{FM3}} &= \frac{N_{8,\mathrm{FM3}}}{T_{LC}}
\end{aligned}$$

and T_{LC} is the system's life cycle.

In this experiments, the template is only instantiated with the data for C9, C12, C13 and C14.

6.5.3.3 *Coupled Iterations Experiments*

The results from the uncoupled including induced failures are used to estimate the parameters of the opportunistic maintenance probability distributions. The same procedure applied to the calculation of the parameters of the induced failure distribution are used. The equations for the level-0 opportunities are given by,

$$\lambda_{1,L0} = (N_1 - 1)\lambda_{1,\text{FM1}} + N_2\lambda_{2,\text{FM1}}$$

$$\lambda_{2,L0} = N_1\lambda_{1,\text{FM1}} + (N_2 - 1)\lambda_{2,\text{FM1}}$$

$$\lambda_{i,L0} = N_1\lambda_{1,\text{FM1}} + N_2\lambda_{2,\text{FM1}}$$

for $i \in 3, 5, 7, 8, 9, 10, 11, 12, 13, 14$.

The equations for the level-1 opportunities are given by,

$$\lambda_{2,L1} = (N_2 - 1)\lambda_{2,\text{FM2}}$$

$$\lambda_{j,L1} = N_2\lambda_{2,\text{FM2}}$$

for $j \in 1, 3, 5, 7, 8, 9, 10, 11, 12, 13, 14$.

The equations for the level-2 opportunities are given by,

$$\lambda_{7,L2} = (N_7 - 1)\left(\lambda_{7,\text{FM2}} + \lambda_{7,\text{FM3}}\right) + N_8\left(\lambda_{8,\text{FM2}} + \lambda_{8,\text{FM3}}\right) + N_{12}\left(\lambda_{12,\text{FM1}} + \lambda_{12,\text{FM2}}\right)$$

$$\lambda_{8,L2} = N_7\left(\lambda_{7,\text{FM2}} + \lambda_{7,\text{FM3}}\right) + (N_8 - 1)\left(\lambda_{8,\text{FM2}} + \lambda_{8,\text{FM3}}\right) + N_{12}\left(\lambda_{12,\text{FM1}} + \lambda_{12,\text{FM2}}\right)$$

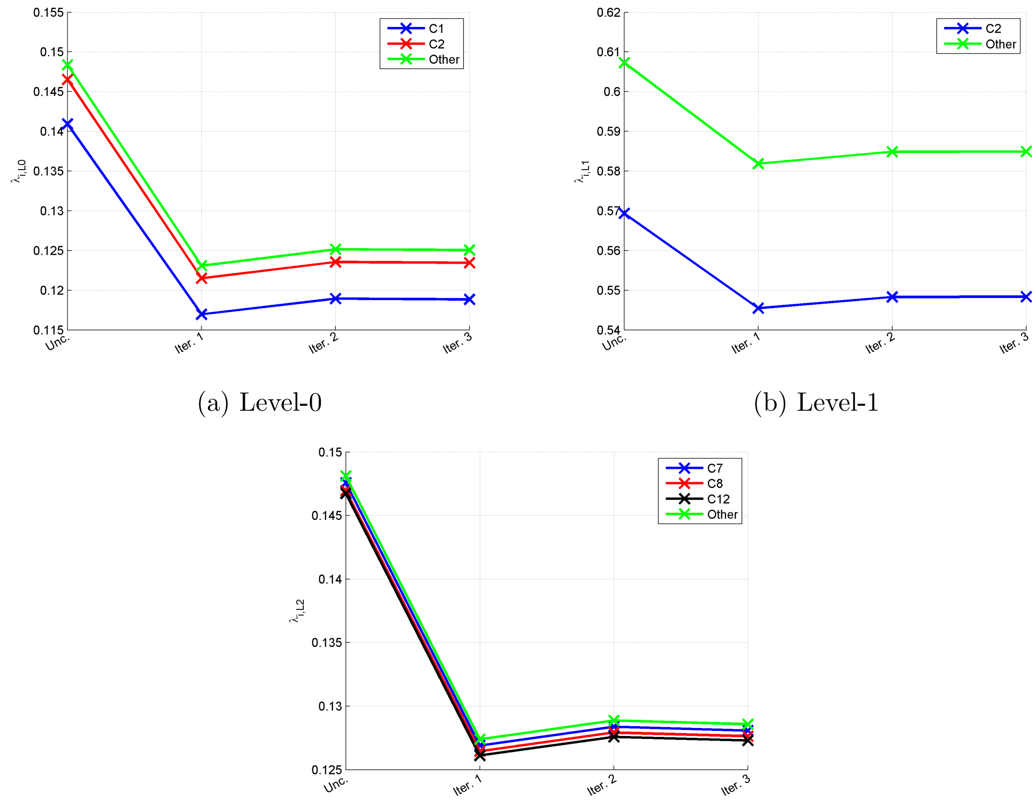
$$\lambda_{12,L2} = N_7\left(\lambda_{7,\text{FM2}} + \lambda_{7,\text{FM3}}\right) + N_8\left(\lambda_{8,\text{FM2}} + \lambda_{8,\text{FM3}}\right) + (N_{12} - 1)\left(\lambda_{12,\text{FM1}} + \lambda_{12,\text{FM2}}\right)$$

$$\lambda_{k,L2} = N_7\left(\lambda_{7,\text{FM2}} + \lambda_{7,\text{FM3}}\right) + N_8\left(\lambda_{8,\text{FM2}} + \lambda_{8,\text{FM3}}\right) + N_{12}\left(\lambda_{12,\text{FM1}} + \lambda_{12,\text{FM2}}\right)$$

for $k \in 1, 2, 3, 5, 9, 10, 11, 13, 14$.

One aspect of the calculation of opportunities is that when the parameter for the opportunistic distribution model are calculated for component types that themselves provide opportunities then when a component of that type fails it does not provide an opportunity for itself (and hence we subtract one from the total number of items being considered as providing opportunities).

Three iterations of the coupled experiment are run, each using the results of the previous iteration to update the estimate of the induced failures and the opportunities. The results of the experiments are shown next.



(a) Level-0

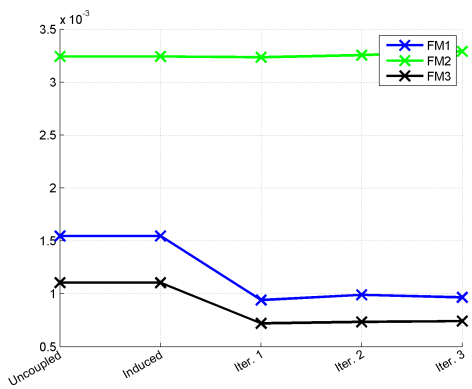
(b) Level-1

(c) Level-2

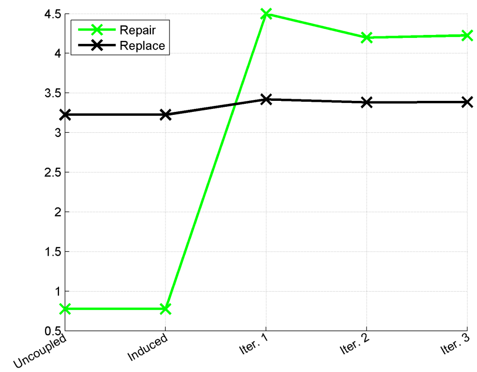
Figure 30: Rate Parameters of opportunities probability model

Fig. 30 shows the estimation of the scale parameter for all of the opportunity probability models. First, it is noted that the estimation has reached an approximately stationary value after the third iteration. This provides the basis for a stopping criteria.

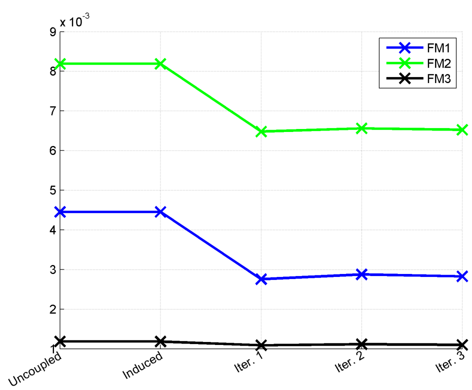
Second, the estimated scale parameters based on the results from the uncoupled simulation over predicts the stationary values by 5% to 25%. This is consistent with the maintenance policy which replaces the components when an opportunity for replacement presents itself. In the uncoupled simulation failures of other components do not provide opportunities for replacement (they are not enabled) and hence the components are replaced less often which in turns results in having more failures throughout the life-cycle. Since the number of failures is used for estimation of the scale parameter of the opportunities, a larger number of predicted failures will result in a larger estimation of the scale parameter. In the same vein, if comparing the estimated values when using the information from the first iteration and when using the information from the second iteration an alternating trend is observed. This is once again consistent with the maintenance policy. Due to the over prediction of the number of opportunities when using the data from the uncoupled simulation, the resulting number of failures in the first iteration of the coupled simulation is under predicted which in turn results in an under prediction of the scale parameter of the opportunities.



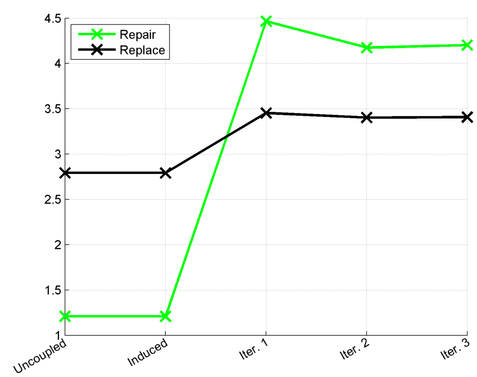
(a) C7, Number of Failures



(b) C7, Number of Maint. Actions



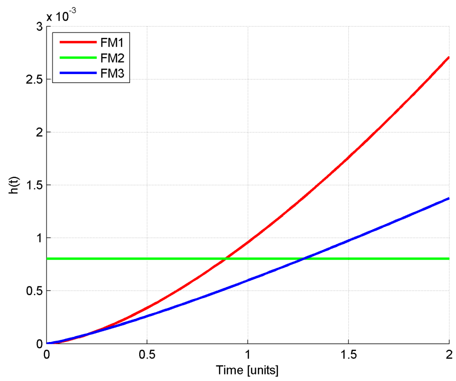
(c) C8, Number of Failures



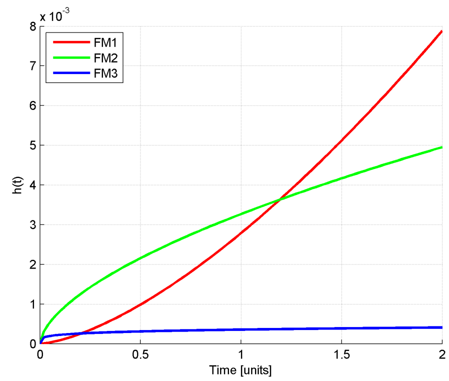
(d) C8, Number of Maint. Actions

Figure 31: Sample of number of events collected from experiments

Fig. 31 shows a sample of the number of events collected from the experiment for component types 7 and 8. Similar figures for the remaining of the components can be seen in Sec. D.1. Fig. 31a and 31c show the overall effect of the opportunistic maintenance policy on the number of failures due to the different failure modes. The opportunistic maintenance policy clearly reduces the overall number of failures. At the same time, it is observed that for component type 7 the number of failures in mode 2 and for component type 8 the number of failures in mode 3 is not affected by the opportunistic maintenance policy. An explanation of this trend can be gleaned from Fig. 32: the hazard rate function in the time interval up to their inspection is for both modes either constant (for C7-FM2), or almost constant (for C8-FM3).



(a) C7



(b) C8

Figure 32: Hazard rate function, failure modes

Fig. 31a and 31c show the overall effect of the opportunistic maintenance policy on the number and type of maintenance actions performed on the component types. It is clear that the reduction in the overall number of failures comes at the expense of an increased number of maintenance action. Whether this is acceptable or not will depend on all the costs associated with both scheduled and unscheduled maintenance actions.

Finally in Fig. 33 the number of inspections per component type as they evolve from uncoupled to coupled iterations is shown. The plot is color coded by inspection level. If the model constructed was a system level model with all the couplings explicitly stated then the number of inspections of different levels would clearly have a definite value. It is interesting to note that even though there is no explicit rule enforcing that the number of inspections for the different component types corresponding to the same inspection level must be the same (in the proposed component-wise model) it turns out that approximately the same value is found from the component-level models.

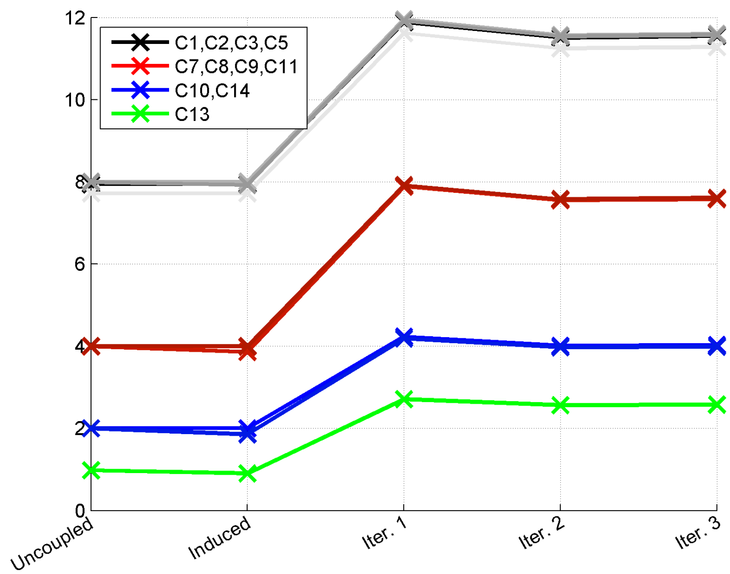


Figure 33: Number of Inspection Events

6.6 Minimal Repair Modeling

6.6.1 Component Template

The minimal-repair component template is shown in Fig. 34. The key idea to model the minimal-repair maintenance action is to make the scheduler clock (the token in the scheduler branch) independent of the component-state branch for the case in which a scheduled maintenance arrives and a repair action is decided by the scrap-clock group.

To achieve this the model is modified so that the token from the component-state branch only leaves the operating state when the component needs to be replaced. Since the number of maintenance events is still recorded the number of repairs can be calculated as follows (in a post-processing stage after the simulation). Let c_{maint}, c_{fail} and c_{rpl} be the number of times the token of scheduler branch arrives at the maintenance place, the token of the component branch arrives to the failure place and the number of times the token of the component-state branch arrives to the replace place respectively. Recall first that (from the maintenance policy) an inspection always results in either a replacement or a repair and

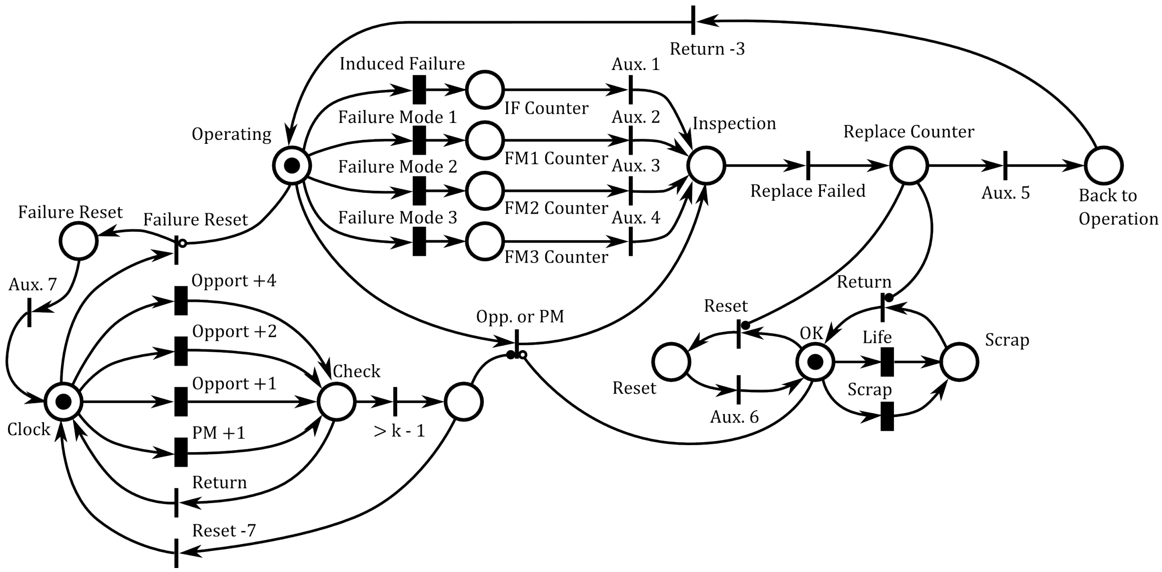


Figure 34: Screenshot of minimal repair template in SPN@

that a failure always results in a replacement. Then, we have

$$n_{rpr} + n_{rpl,OM,PM} = c_{\text{maint}} \tag{22}$$

$$c_{\text{fail}} + n_{rpl,OM,PM} = c_{\text{rpl}} \tag{23}$$

where n_{rpr} is the number of repairs and $n_{rpl,OM,PM}$ is the number of replacements due to either opportunistic maintenance or preventive maintenance. From this, the number of repairs is given by

$$n_{rpr} = c_{\text{maint}} + c_{\text{fail}} - c_{\text{rpl}} \tag{24}$$

6.6.2 Verification of the template

The verification process of the template is made by several step-by-step simulations with modification of some parameters to ensure that the logic of the model follows the specifications. The main difference in the logic behavior of the model when compared to the perfect repair template is the inclusion of an inhibitor arc going from the “Ok” place in the scrap-clog group to the “Sched. Repl.” transition. It is this inhibitor arc that indirectly enables the behavior expected for the minimal repair maintenance policy. The absence of this inhibitor arc reduces the minimal-repair template to the perfect-repair template. As a further verification step of the consistency between the two templates an experiment was run for the minimal-repair template modified so that it does not include the inhibitor arc. The results were the same as the ones obtained by an equivalent experiment using the prefect-repair template as a basis.

6.6.3 Simulation of the template: Minimal Repair Experiment

6.6.3.1 Uncoupled Simulation

The result for one of the experiments ran in the uncoupled configuration are shown in Table 8. Since the induced failures are not enabled the column corresponding to the induced failures has only zeros.

Table 8: Results of Uncoupled Experiment (Minimal repair), per component. 1E7 replications

	FM1	FM2	FM3	Repair	Replace	Insp.	Ind. Fail.
C1	0.20147	0	0	5.8555	2.1537	7.8077	0
C2	0.05350	0.40026	0	6.43654	1.61105	7.59383	0
C3	0	0	0	1.97553	6.02447	8.0	0
C5	0	0	0	5.97250	2.02749	8.0	0
C7	0.00869	0.006426	0.005005	0.006963	3.99313	3.97997	0
C8	0.03458	0.027310	0.002797	0.5526347	3.44801	3.9359	0
C9	0.01242	0	0	1.3880027	2.61204	3.98762	0
C10	0.02965	0	0	2E-7	2.000061	1.97040	0
C11	0	0	0	0.12802	3.87198	4.0	0
C12	0.01846	0.006539	0	4.89967	3.10064	7.97532	0
C13	1.30614	0	0	4E-7	1.53449	0.22834	0
C14	0	0	0	0.99971	1.000284	2.0	0

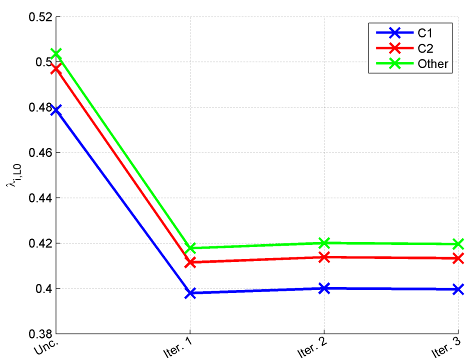
Table 9 shows the total number of failures per component type for each of the maintenance policies when they do not include opportunistic maintenance (uncoupled simulation). The general trend is that the total number of failures throughout the life cycle of the system is larger when a minimal repair policy is used instead of a perfect repair policy (without including opportunistic maintenance).

The scale parameter for the opportunity probability model is calculated in the same way as outlined in the perfect repair section. Their evolution as iterations are carried out is shown in Fig. 35.

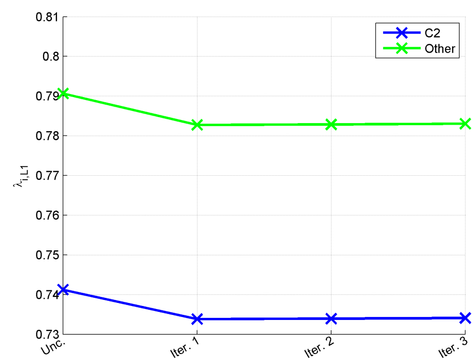
The same effect noted for the simulation of the system under a perfect repair policy is

Table 9: Total number of failures of perfect repair and minimal repair policy without opportunistic actions

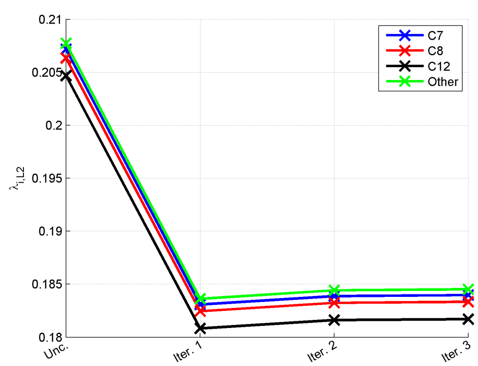
	Perfect Repair	Minimal Repair	Rel. % Diff.
C1	0.0601617	0.2014748	234%
C2	0.3225539	0.4537627	40.7%
C3	0	0	–
C5	0	0	–
C7	0.0201015	0.0201251	0.12%
C8	0.0525571	0.0646931	23%
C9	0.0058183	0.0124197	113%
C10	0.0296444	0.0296553	0.0368%
C11	0	0	–
C12	0.0112405	0.0249995	122.4056
C13	1.3234265	1.3061459	-1.3057
C14	0	0	–



(a) Level-0



(b) Level-1

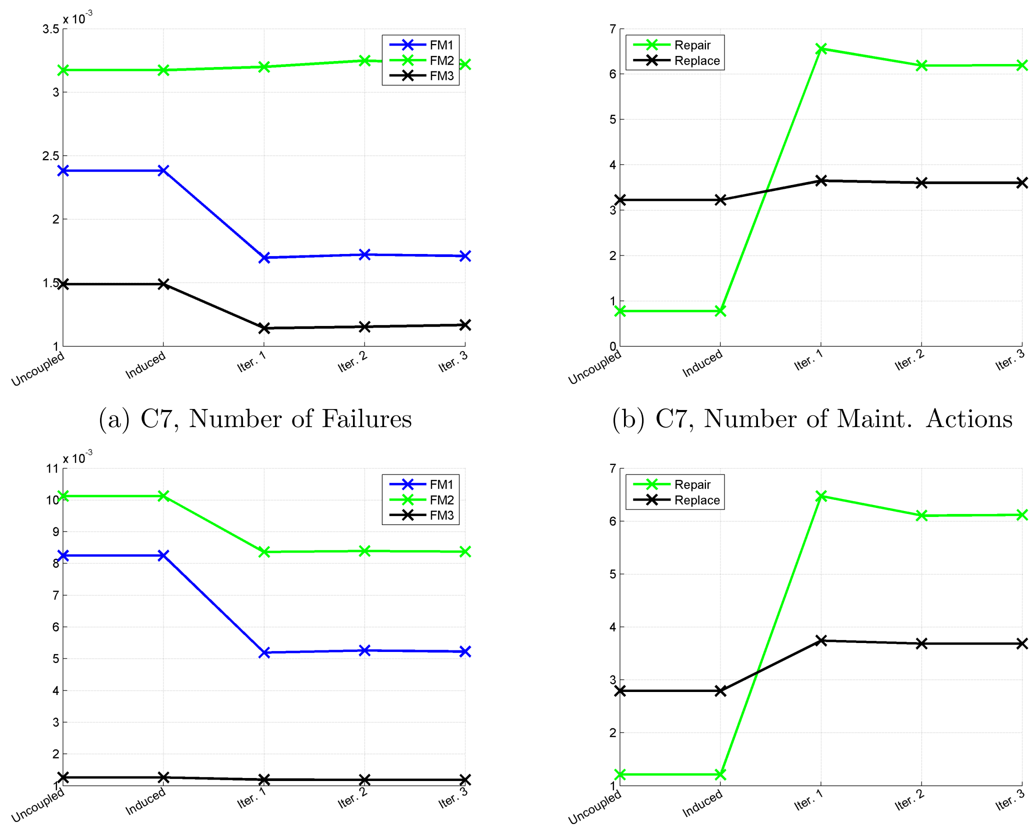


(c) Level-2

Figure 35: Scale Parameters of opportunities probability model, minimal repair

present in the simulation of the system under a minimal repair policy. Namely, using the data from the uncoupled model simulation results to estimate the scale parameter of the opportunities results in an over prediction of its value.

Fig. 36 shows a sample of the number of events collected from the experiment for component types 7 and 8. Similar figures for the remaining of the components can be seen in Sec. D.2. Fig. 36a and 36c show the overall effect of the opportunistic maintenance policy on the number of failures due to the different failure modes. The opportunistic maintenance policy clearly reduces the overall number of failures at the cost of increasing the number of maintenance actions.



(a) C7, Number of Failures

(b) C7, Number of Maint. Actions

(c) C8, Number of Failures

(d) C8, Number of Maint. Actions

Figure 36: Sample of number of events collected from experiments

Table 10 compares the total number of maintenance actions for the perfect repair and the minimal repair policy. The minimal repair policy consistently increases the total number of maintenance actions for each component type. Focusing on type of maintenance action one can observe that the minimal repair policy increases in a larger percentage the number of (minimal) repairs.

Table 10: Total number of maintenance actions of perfect repair and minimal repair policies

	Repair		Replace			
	PR	MR	Rel. % Diff.	PR	MR	Rel. % Diff.
C1	9.4831	12.5635	32.5	2.9484	3.0108	2.1
C2	10.8478	13.9249	28.4	1.5775	1.6551	4.9
C3	6.3178	9.2713	46.7	6.1228	6.3107	3.1
C5	9.5181	12.6202	32.6	2.9216	2.9601	1.3
C7	3.6535	5.4469	49.1	4.9412	5.5165	11.6
C8	4.1703	6.0928	46.1	4.4286	4.8815	10.2
C9	5.9054	8.1526	38.1	3.0002	3.1704	5.7
C10	0.0017	0.0026	52.9	4.9543	6.3191	27.5
C11	4.1753	6.1259	46.7	4.416	4.8321	9.4
C12	9.2951	12.4167	33.6	3.3	3.3606	1.8
C13	0.0001	0.0001	0.0	3.9479	4.8457	22.7
C14	3.9339	5.2919	34.5	1.3073	1.3498	3.3

CHAPTER VII

CONCLUSIONS AND FUTURE WORK

An analytical method for the compact characterization of competing-risk coupling of a component with the rest of the system has been developed and tested on small scale state-space models. While the models considered in the thesis were specific to modeling opportunistic maintenance combined with the age-based replacement policies, the described phenomena is pertinent to a broad range of coupling scenarios that involve several competing transitions from the same state. It is well known (*e.g.*, Ref. [26]) that the use of exponential distributions in such situations can sometimes lead to significant errors (if the underlying distribution is non-exponential, including deterministic delays). However, it was not clear as to which (compact) properties of a distribution (in addition to the mean) impact the system-level results. While the standard deviation seems to be a natural candidate, the present work argues that the use of so-called "winning ratio" γ, provides superior accuracy.

Specifically, it was shown that for Weibull distributions matching the winning ratio leads to more precise results than the use of a distribution that matches the two first moments of the targeted distributions or an approximation obtained using MLE. For Weibull distributions, the developed procedure is also justified on asymptotic considerations that demonstrate that, as the inspection interval gets significantly smaller than the scale of the distribution, the winning ratio tends to a simple ratio determined by the shape parameters of Weibull distributions. Instead of providing a good global match for the whole range of the distribution, the resulting approximation targets the left tail of the distribution, which is the most relevant for realistic maintenance scenarios. It was further shown that a combination of lognormal distributions can be well approximated by a Weibull distribution with matched winning ratio as well. It is hoped that the developed procedure will facilitates the compact representation of maintenance policies for complex systems by enabling the application of component-wise representation of maintenance processes that accurately represent inter-component couplings.

One possible avenue of exploration for future work is to study different iteration procedures used to calculate the parameters of the approximating distribution. Research questions on this instance would include if all the iterating procedures converge to the same value or the speed of convergence. Another possible extension of this work is to the modeling of dependent failure modes. The proposed methodology is based on the idea that the exogenous effect on the item under consideration can be modeled using competing risks ideas. Since competing risk analysis has been used in reliability modeling to capture dependent failure modes one possible research question would be to study the suitability of the approach for the reduction of the item model complexity. A positive answer would lead to a larger field of application of the methodology and to the possibility of dealing with more complex problems.

APPENDIX A

PARAMETRIC STATISTICAL DISTRIBUTIONS IN RELIABILITY MODELING

Since in this work we focus on reliability modeling we will only provide a description of probability distributions of nonnegative random variables. These random variables are the ones encountered when studying waiting times. While for nonnegative random variables with standard deviations that are small compared to their means the normal distribution can offer an appropriate approximation, for the remainder of the cases this is not so. For nonnegative random variables and in particular when dealing with reliability modeling, the exponential distribution is undoubtedly the most pervasive distribution in use [56].

One of the central concepts used to describe phenomena which are of a stochastic in nature is the concept of a random variable.

Definition 1 (Random Variable)**.** Consider a random experiment with a sample space $\mathcal{C}$. A function X, which assigns to each element $c \in \mathcal{C}$ one and only one number $X(c) = x$, is called a *random variable.* The *space* or *range* of X is the set of real numbers $\mathcal{D} = \{x : x = X(c), c \in \mathcal{C}\}$.

If $\mathcal{D}$ is a countable set then the associated random variable is called a *discrete random variable*; otherwise, if $\mathcal{D}$ is an interval of real numbers it is called a *continuous random variable.*

Various alternative functions can be used to fully represent a random variable. These functions include the distribution function, the reliability function, probability density function, hazard function, cumulative hazard function, mean residual life function, and total time-on-test transform. These representations apply to continuous and discrete random variables and when one of them is given the rest can be uniquely obtained. All of them are useful because for a given distribution one of them may have a very simple form and also because certain aspects of the distribution may be more readily seen by looking at a specific functions (e.g. the mode of a distribution is easily appreciated in the graph of its probability density function but harder to find in the graph of its cumulative distribution function).

In the following definitions we will let X be a random variable.

Definition 2 ((Cumulative) Distribution Function)**.** The function $F : (-\infty, \infty) \to \mathbb{R}$ defined by,

$$F(x) = P\left(X \leq x\right) = P\left(\{c \in \mathcal{C} : X(c) \leq x\}\right) \tag{25}$$

is called the *(cumulative) distribution function* of X

Distribution functions are sometimes called "cumulative distribution functions" to emphasize the fact that $F(x)$ accumulates the probabilities less than or equal to x. The distribution function is non-decreasing and right-continuous. Moreover, it can be shown that $\lim_{z \to -\infty} F(z) = 0$ and $\lim_{z \to \infty} F(z) = 1$.

Definition 3 (Reliability Function)**.** The function $R : (-\infty, \infty) \to \mathbb{R}$ defined by,

$$R(x) = P\left(X \geq x\right) \tag{26}$$

is called the *reliability function* of X.

Reliability functions are sometimes called "survival functions" [56] or "survivor functions" [49]. Moreover, sometimes they are denoted by $\bar{F}(x)$ [3, 56] or $S(x)$ [49]. Clearly, $R(x) = 1 - F(x)$. When the random variable T represents the lifetime of an item, then the value of the reliability function at t is interpreted as the probability that the item is still functioning at time t.

Definition 4 (Probability Density)**.** If f is a nonnegative function for which,

$$F(x) = \int_{-\infty}^{x} f(z)\,\mathrm{d}z, \text{ for all real } x \tag{27}$$

then f is called a *probability density* of X (or of F).

Definition 5 (Hazard rate)**.** If F is an absolutely continuous distribution function with density f, then the function h defined on $(-\infty, \infty)$ by

$$h(x) = \begin{cases} \frac{f(x)}{R(x)} & \text{if } R(x) > 0 \\ \infty & \text{if } R(x) = 0 \end{cases} \tag{28}$$

is called the *hazard rate* of X, (or of F).

Very often the hazard rate is called the *hazard function* or the *failure rate*. It is maybe the most popular representation when dealing with random variables modeling lifetime due to its interpretation as the amount of risk associated with an item at time t. It can be shown [49] that, for small Δt, the hazard rate function is,

$$h(t)\Delta t = P(t \leq T \leq t + \Delta t \mid T \geq t)$$

which in words says that the probability that an item will fail within the next Δt time units when it is known that it has already survived up to time t is proportional to the failure rate.

A.0.4 Exponential Distribution

It is the most important of one parameter family of life distributions. They are quite simple to describe and exceptionally amenable to statistical analysis. Several of the most commonly used families of life distributions are two- or three parameter extensions of the exponential distribution. The exponential distribution, with its constant hazard rate forms a baseline for evaluating other families.

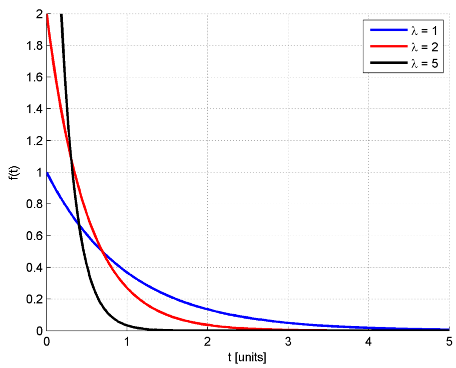
The reliability function, probability density function and hazard rate function are given by,

$$R(t) = e^{-\lambda t} \tag{29}$$

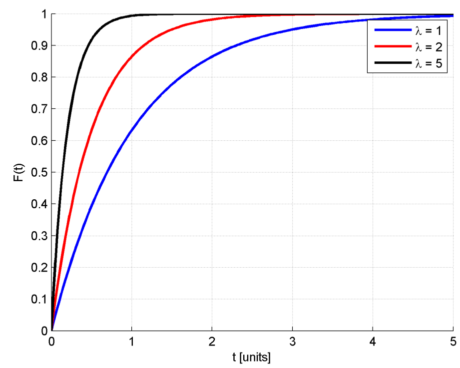
$$f(t) = \lambda e^{-\lambda t} \tag{30}$$

$$h(t) = \lambda \tag{31}$$

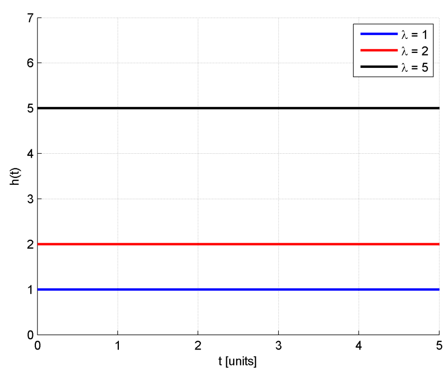
where λ is a scale parameter often called the *failure rate*. Fig. 37 shows the density, distribution and hazard function for some values of the failure rate.



(a) Density function



(b) Distribution function



(c) Hazard function

Figure 37: Exponential function

The exponential distribution is the only continuous distribution for which the hazard rate is a constant. It greatly simplifies calculations with it because of the following properties [5],

- *Memoryless Property* Knowing that the item is functioning at the present time, its behavior in the future *will not depend on how long it has already been operating.*

- *Constant hazard rate at system level* If a system without redundancy consists of n elements and the failure-free times of these elements are independent and exponentially distributed then, the system failure rate is also constant and equal to the sum of the failure rates of its elements.

A.0.5 Weibull Distribution

A generalization of the exponential distribution with a hazard rate that can be decreasing, constant or increasing. It was introduced by W. Weibull in 1951, related to investigations on fatigue in metals.

The reliability function, probability density function and hazard rate function are given by,

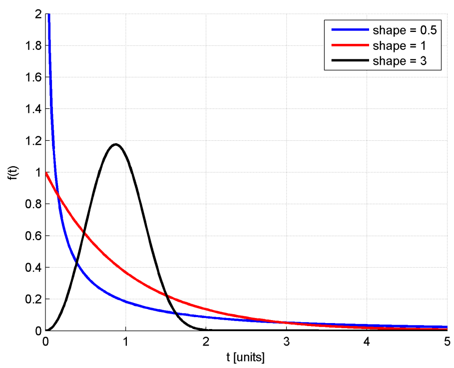
$$R(t) = e^{-(\eta t)^\beta} \tag{32}$$

$$f(t) = \beta \eta^\beta t^{\beta-1} e^{-(\eta t)^\beta} \tag{33}$$

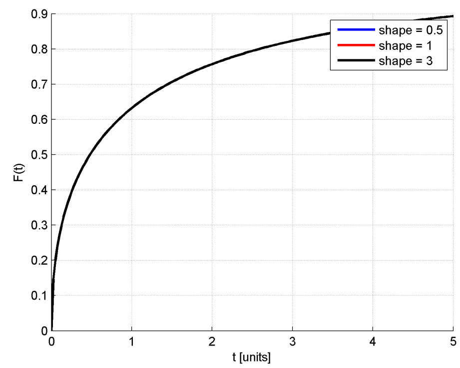
$$h(t) = \beta \eta^\beta t^{\beta-1} \tag{34}$$

where η and β are the scale and shape parameter respectively. Since no parametrization of the Weibull distribution has been standardized one must always check the specific parametrization being used by any given author. Fig. 38 shows the density, distribution and hazard function for some values of the shape parameter.

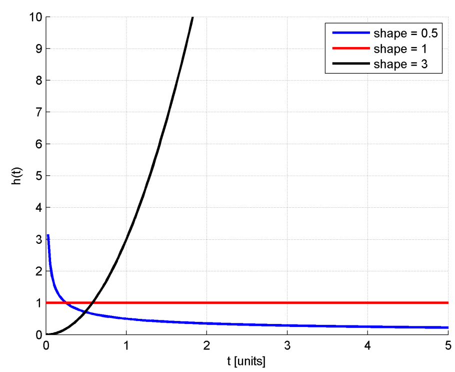
In the context of reliability, Weibull distributions are often used due to their flexibility of representing hazard rate functions that can be either increasing or decreasing with time. The former correspond to the shape parameter $\beta > 1$ (*e.g.,* failures in deteriorating systems), while the latter correspond to the shape parameter $\beta < 1$. Conveniently, for $\beta = 1$, an exponential distribution is recovered, with the scale parameter η. There are additional



(a) Density function



(b) Distribution function



(c) Hazard function

Figure 38: Weibull distribution

reasons for using Weibull distribution in system reliability, including ease of integration for finding moments, and the relationship to the "weakest link" mode of failure. The latter property is particularly pertinent: the Fisher-Tippett-Gnedenko theorem [27, 33] states that for a large number of identically distributed functions, the competing risk (i.e., the minimum of failure times) will converge to one of the three families of extreme value distributions (Weibull, Gumbel, or Fréchet).

A.0.6 Lognormal Distribution

If X is a random variable with a normal distribution, then $Y = \exp(X)$ has a lognormal distribution. Because the normal distribution arises as a limiting distribution for sums, the lognormal distribution arises as a limiting distribution for products.

In some circumstances, multiplicative models are appropriate. It is for these types of models that the lognormal distribution is used. In the physics of failure field, arguments have been made concerning the failure mechanism of solid state devices that lead to the lognormal distribution as the distribution of life length [38]. The lognormal distribution is often used in situations where the right-hand tail is "fatter" than the left one, that is, the distribution is skewed to the right.

The presence of right-hand tail (positive) skewness is sometimes used as a basis for modeling time to restoration of items. This is particularly applicable when logistics aspects of the system are not being explicitly modeled (for instance, availability of spares). The assumption in this case is that if the restoration does not occur in a relatively short time span then values several times larger are possible because the likely reason for not being able to carry out the restorations in the short time span is that something severe has occurred (i.e. there is no available spare on site).

A.0.7 Uniform Distribution

The uniform distribution is a simple two-parameter model. The main application of the uniform distribution is to approximate lifetime distributions over relatively small intervals. The uniform distribution can also be used to model time-to-restoration of items where the restoration action under consideration is a very simple set of instructions. For instance, the time to replace a failed line replaceable unit when the maintainer is already on site with all the required tools and spares available and on hand can be modeled using a uniform distribution.

The uniform distribution in the interval (0,1) plays an important role in simulation problems where is the basis for one of the methods to generate random variables from different parametric distributions.

APPENDIX B

BATCH FILE CONFIGURATION

The proposed component based modeling approach relies on having the means to simulate the component models in a straightforward manner. To facilitate the simulation procedure an extra feature was added to the simulation software SPN@. This extra feature allows performing batch simulations on a given model. These batch simulations allow to

1. Perform a predefined number of Monte Carlo experiments on the same model with the same parameters of all the transitions but using a different seed value for the pseudo random number generator.

2. Perform a predefined number of Monte Carlo experiments on the same model with different parameters for a set of all the transitions.

To perform a batch simulation you need the model on which you will perform the Monte Carlo experiments and a configuration file that specifies the values the parameters of each of the transitions assumes for a given Monte Carlo run. The results of the Monte Carlo experiment are written to a text file.

In this work the batch simulation feature allows the user to set up a batch simulation on a general component template file so that a Monte Carlo experiment is performed on the model using the transition parameters corresponding to the different component types.

In the following sections a description of the limitations and usage of the batch simulation feature is provided.

B.1 Current limitations of the Batch Simulation feature

- It can only vary parameters of existing transitions in the model open in the GUI at the time the simulation is launched. In other words, a batch simulation that first performs a Monte Carlo experiment on a given model and then performs a Monte Carlo experiment on another model is at the present time not possible.

- The number of replications of each Monte Carlo experiment cannot be specified in the configuration file.

- The initial state of the net cannot be fully specified in the configuration file alone.

B.2 Usage

It must be noted that in the current implementation of SPN@ the numbering of the transitions could be modified by the software when a change in the model is performed. This is important because the transitions that will be modified are accessed by this automatically assigned index. Hence, as a precaution, before launching a batch simulation an inspection of the transition indexes must be performed (to check that the transitions being modified are the ones that need to be modified).

A second note of caution arises from the fact that the initial state of the net cannot be fully specified in the configuration file alone. In the current set-up the batch engine reads the configuration file row by row and modifies only the specified transitions to the specified values. This implies that the value of all the other transitions will be left in the same state that it was before and hence steps must be taken to ensure that the user is aware of the initial state of the model at the beginning of the simulation run. Two approaches that work fairly well are,

1. Set the firing law in all the transitions of the template to None and then bring the configuration file to the desired initial state by specifying the values of all the transitions in the model.

2. To define a template with the desired initial state for all the transitions and then creating a copy of this template on which the batch simulation is then run.

The steps that are to be followed to perform a batch simulation then are,

1. Open the model on which the simulation will be performed.

2. Click the Go button to open the simulation menu window.

3. Click the Batch Set-Up button to open the Batch Simulation Properties window.

4. Input the name of the configuration file and the name of the file where results will be written in the appropriate fields.

5. Read the configuration file by clicking the OK button in the Batch Simulation Properties window.

6. Run the batch simulation by clicking the Run Batch button.

B.3 Configuration File format

The configuration file is a comma separated value file where each row identifies the parameter that is being accessed. Hence, if only one parameter is modified between runs then the configuration file will consist of a single line. The first three columns are used to identify the parameter and the following columns specify the values that need to be used for each run. Hence, if a row in the configuration file has 10 columns it means that the batch simulation consists of 7 runs (the first three columns specify the parameter being modified). A typical row of a configuration file will look like

Table 11: Batch Configuration File example

tranIdx, policyColorIdx, typeIdx parIdx, run1Val, . . . , runNVal

Table 2 specifies typeIdx and what parameters are modified by parIdx for the available firing distributions.

Table 12: key-value pairs for batch configuration file

Parameter	Description
tranIdx	Index of the transition
policyColorIdx	Index of the policy color
typeIdx	Firing Type of the given transition
parIdx	1 if we are accessing parameter 1 and 2 if we are accessing parameter 2
run1Val	Value of run 1
runNVal	Value of run N

Distribution	typeIdx	parIdx	
		1	2
None	-1	–	–
Fixed	0	Fixed Delay	–
Weibull	1	Weibull Scale	Weibull Shape
Lognormal	2	Lognormal Mean	Lognormal Variance
Uniform	3	Uniform Delay	–
Inverse Fixed	4	Inverse Fixed Delay	–
Exponential	5	Failure Rate	–
Age Dependent Fixed	6	Age Dependent Delay	–

APPENDIX C

GAS TURBINE MODEL SET-UP

C.1 Perfect Repair

C.1.1 Model Logic

Tables 13 and 14 illustrate the setting of each of the transitions in the perfect repair model template and the description of what each sensor is counting.

Table 13: Transitions that are not modified but intrinsic to the model logic

Transition	Index	Color	Color Change Factor	Distribution	Parameter
PM (+1)	25	0	1	Fixed	1
Repair	12	0	0	Fixed	2E-7
Replace Failed	13	1	0	None	–
		2	0	Fixed	1E-7
Replace Scrap	11	0	0	Fixed	1E-7
Back to Op.	0	0	-3	Fixed	1E-6
Opp. or PM	1	0	0	Fixed	1E-6
Return	15	0	0	Fixed	1E-7
Reset	16	0	0	Fixed	1E-7
Reset -7	22	0	-7	Fixed	2E-6
Failure Reset	6	0	-3	Fixed	2E-6
Clock Return	23	0	0	Fixed	2E-6
Aux. 1	10	0	0	Fixed	1E-6
Aux. 2	7	0	0	Fixed	1E-6
Aux. 3	8	0	0	Fixed	1E-6
Aux. 4	9	0	0	Fixed	1E-6
Aux. 5	17	0	0	Fixed	1E-7
Aux. 6	14	0	0	Fixed	2E-7
Aux. 7	18	0	0	Fixed	2E-7
Aux. 8	24	0	0	Fixed	1E-6

C.1.2 Component type specific transitions

In this section the parameters for each of the transitions in the perfect repair component model template are listed.

Table 14: Sensors description

Sensor ID	Description
0	Failure Mode 1
1	Failure Mode 2
2	Failure Mode 3
3	Inspection
4	Repair Counter
5	Replace Counter
6	Inspection Due Counter
7	Induced Failure

Table 15: Transitions that are modified during batch simulation

Transition	Index	Color	Color Change Factor	Distribution	Parameter
Induced Failure	5	0	2	None	–
Failure Mode 1	2	0	2	None	–
Failure Mode 2	3	0	2	None	–
Failure Mode 3	4	0	2	None	–
Life	20	0	0	None	–
Scrap	19	0	0	None	–
Opport. +1	26	0	1	None	–
		1			
		2			
		3			
Opport. +2	27	0	2	None	–
		1			
		2			
		3			
Opport. +4	28	0	4	None	–
		1			
		2			
		3			
$>$ k - 1	21	0	0	None	–
		1			
		2			
		3			
		4			
		5			
		6			

C.2 Minimal Repair

C.2.1 Model Logic

Tables 16 and 17 illustrate the setting of each of the transitions in the minimal repair model template and the description of what each sensor is counting.

Table 16: Transitions that are not modified but intrinsic to the model logic

Transition	Index	Color	Color Change Factor	Distribution	Parameter
PM (+1)	22	0	1	Fixed	1
Replace	11	0	0	Fixed	1E-7
Back to Op.	0	0	-3	Fixed	1E-6
Sched. Repl.	1	0	0	Fixed	1E-6
Return	13	0	0	Fixed	1E-7
Reset	14	0	0	Fixed	1E-7
Reset -7	19	0	-7	Fixed	2E-6
Failure Reset	6	0	-3	Fixed	2E-6
Clock Return	20	0	0	Fixed	2E-6
Aux. 1	10	0	0	Fixed	1E-6
Aux. 2	7	0	0	Fixed	1E-6
Aux. 3	8	0	0	Fixed	1E-6
Aux. 4	9	0	0	Fixed	1E-6
Aux. 5	12	0	0	Fixed	2E-7
Aux. 6	15	0	0	Fixed	2E-7
Aux. 7	21	0	0	Fixed	1E-6

Table 17: Sensors description

Sensor ID	Description
0	Failure Mode 1
1	Failure Mode 2
2	Failure Mode 3
3	Inspection
4	Repair Counter
5	Replace Counter
6	Inspection Due Counter
7	Induced Failure

C.2.2 Component type specific transitions

In this section the parameters for each of the transitions in the minimal repair component model template are listed.

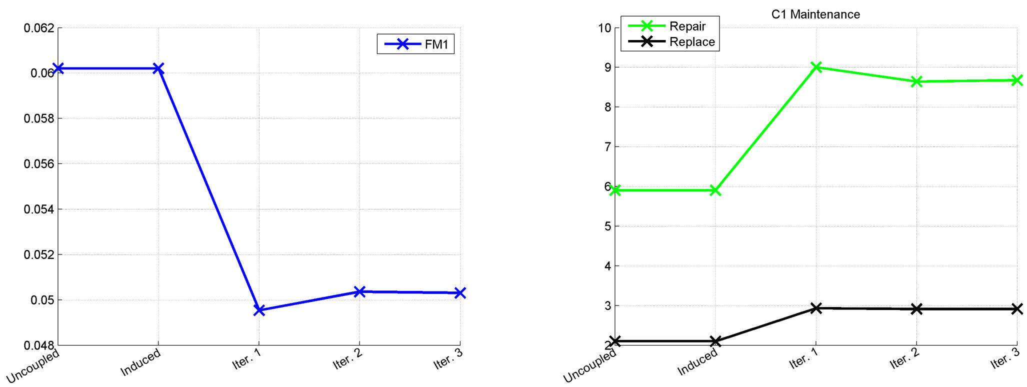
Table 18: Transitions that are modified during batch simulation

Transition	Index	Color	Color Change Factor	Distribution	Parameter
Induced Failure	5	0	2	None	–
Failure Mode 1	2	0	2	None	–
Failure Mode 2	3	0	2	None	–
Failure Mode 3	4	0	2	None	–
Life	17	0	0	None	–
Scrap	16	0	0	None	–
Opport. +1	23	0 1 2 3	1	None	–
Opport. +2	24	0 1 2 3	2	None	–
Opport. +4	25	0 1 2 3	4	None	–
$>$ k - 1	18	0 1 2 3 4 5 6	0	None	–

APPENDIX D

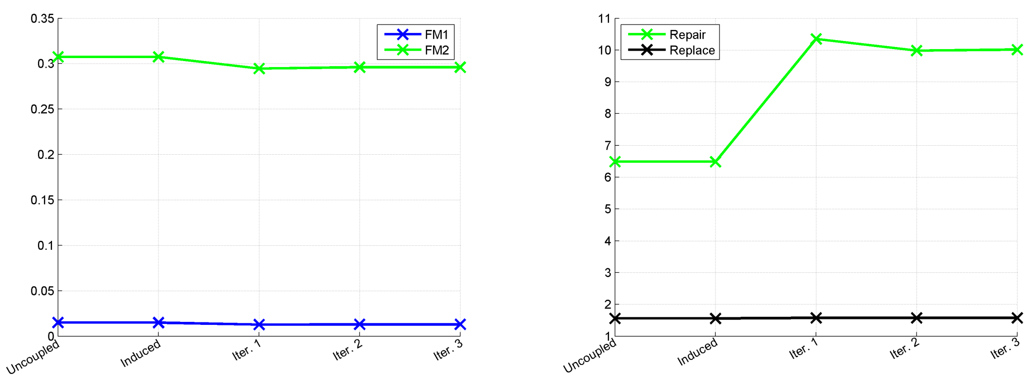
EXPERIMENT RESULTS

D.1 Perfect Repair



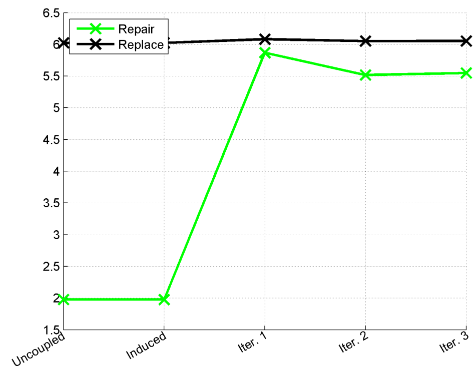
(a) Number of Failures (b) Number of Maint. Actions

Figure 39: Component type 1



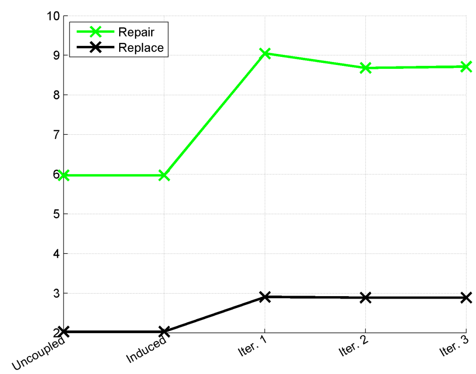
(a) Number of Failures (b) Number of Maint. Actions

Figure 40: Component type 2



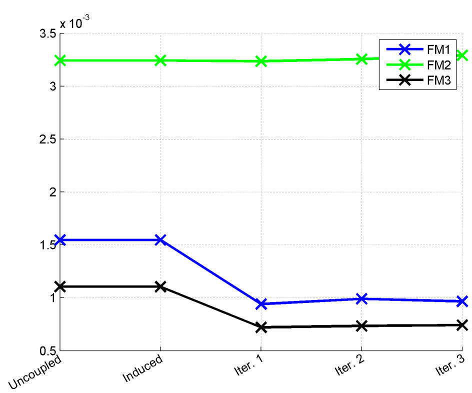
(a) Number of Maint. Actions

Figure 41: Component type 3

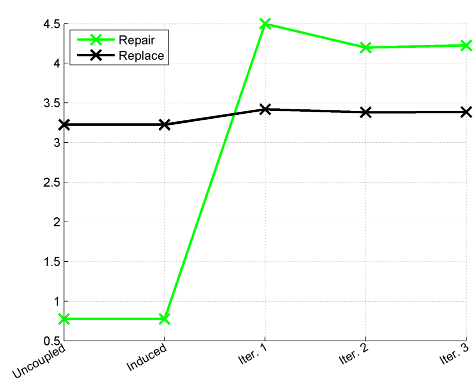


(a) Number of Maint. Actions

Figure 42: Component type 5

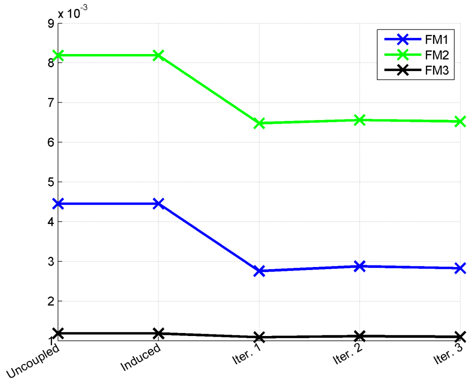


(a) Number of Failures

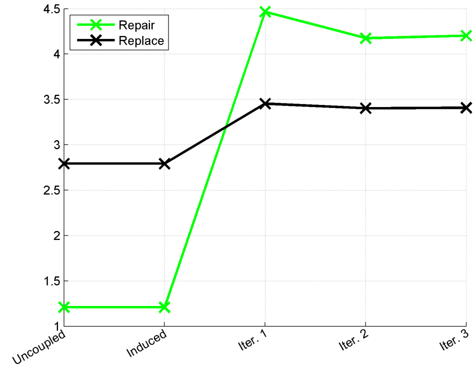


(b) Number of Maint. Actions

Figure 43: Component type 7

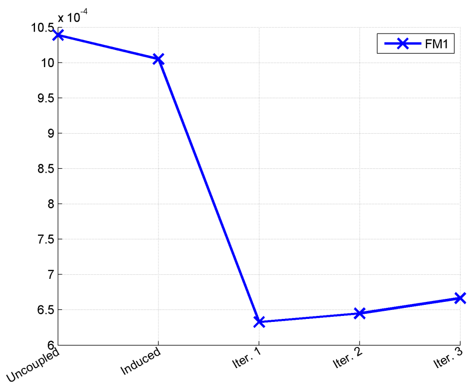


(a) C8, Number of Failures

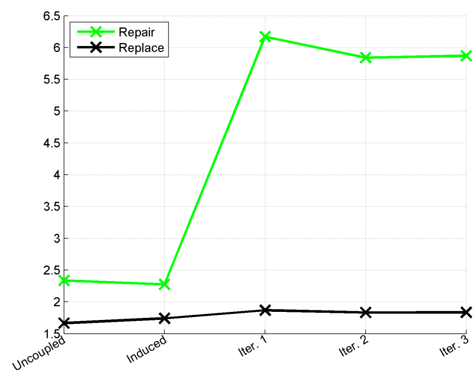


(b) C8, Number of Maint. Actions

Figure 44: Component type 8

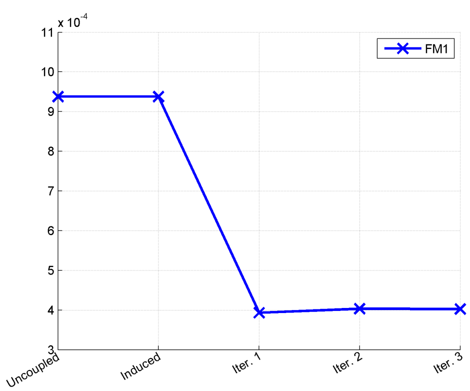


(a) Number of Failures

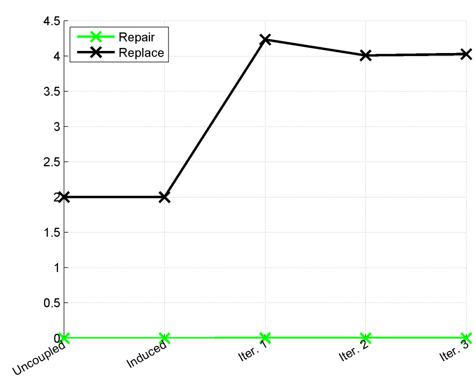


(b) Number of Maint. Actions

Figure 45: Component type 9

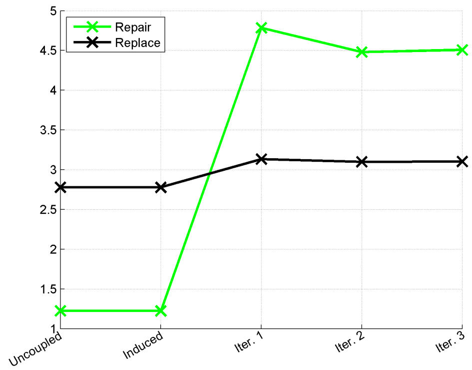


(a) Number of Failures



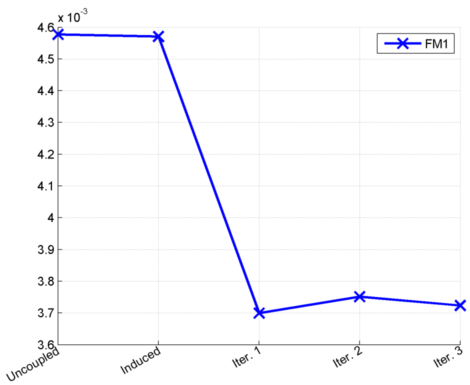
(b) Number of Maint. Actions

Figure 46: Component type 10

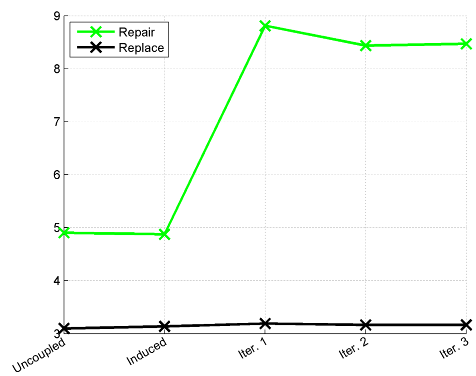


(a) Number of Maint. Actions

Figure 47: Component type 11

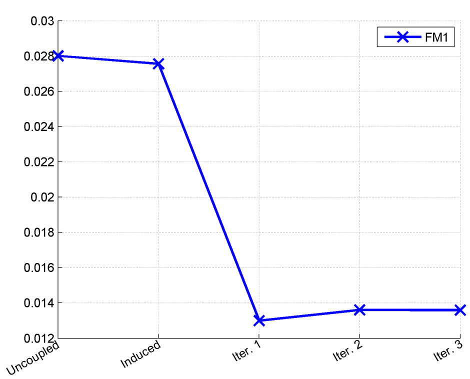


(a) Number of Failures

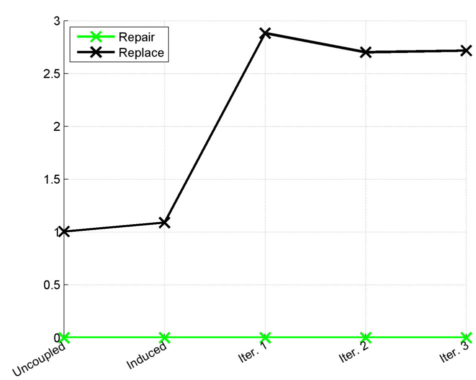


(b) Number of Maint. Actions

Figure 48: Component type 12

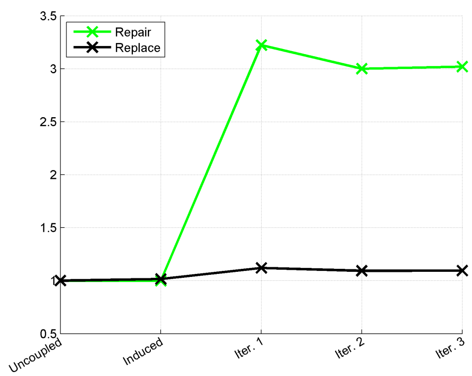


(a) Number of Failures



(b) Number of Maint. Actions

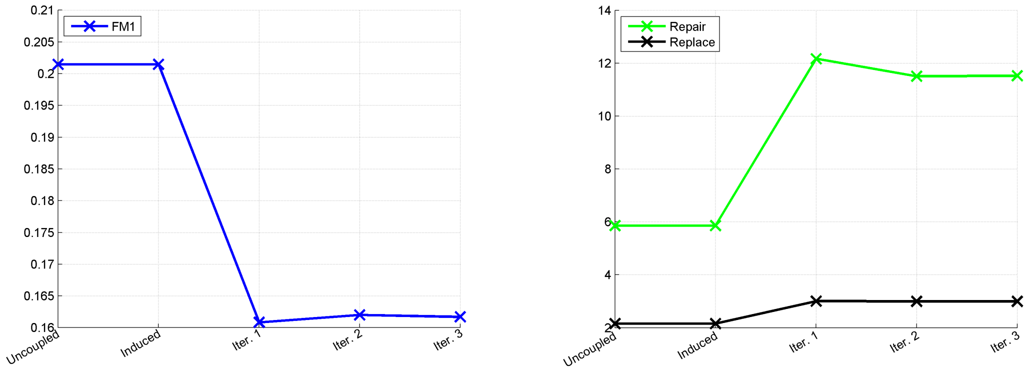
Figure 49: Component type 13



(a) Number of Maint. Actions

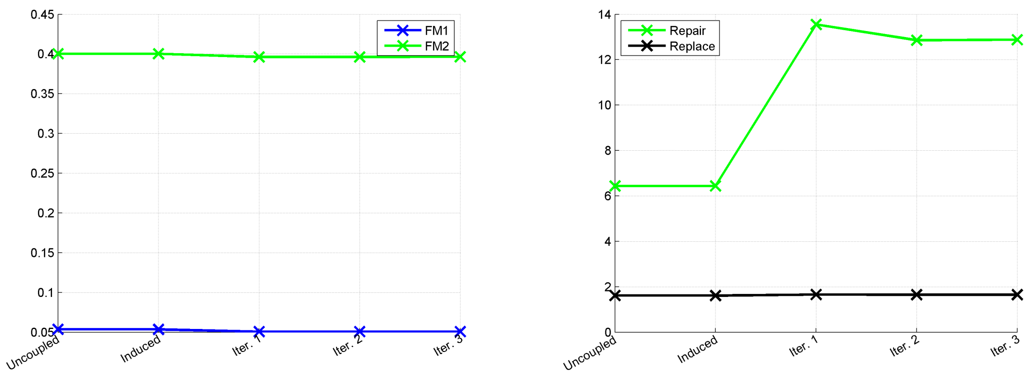
Figure 50: Component type 14

D.2 Minimal Repair



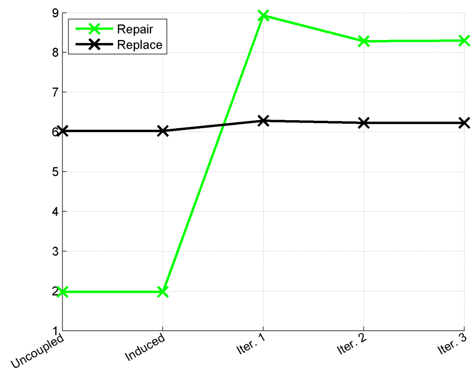
(a) Number of Failures (b) Number of Maint. Actions

Figure 51: Component type 1



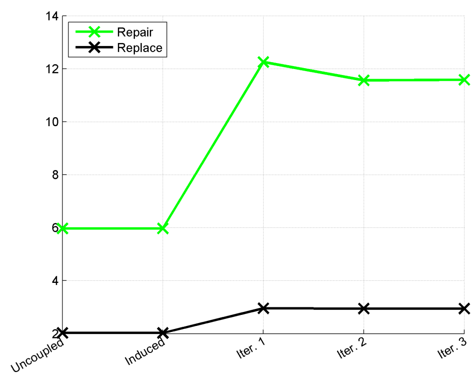
(a) Number of Failures (b) Number of Maint. Actions

Figure 52: Component type 2



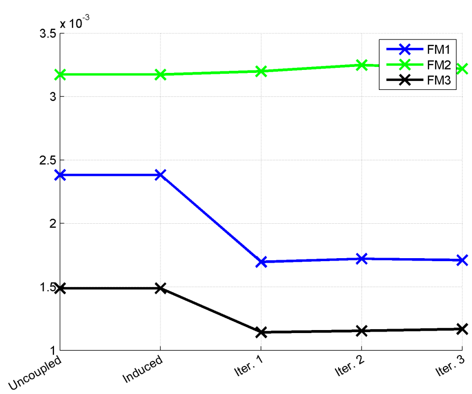
(a) Number of Maint. Actions

Figure 53: Component type 3

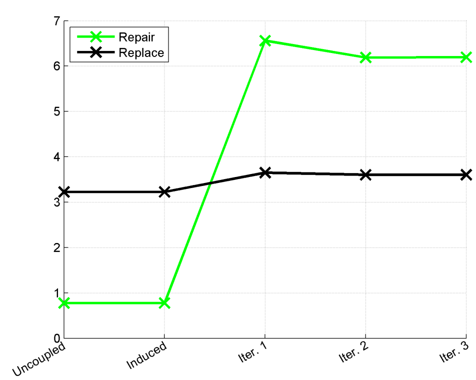


(a) Number of Maint. Actions

Figure 54: Component type 5

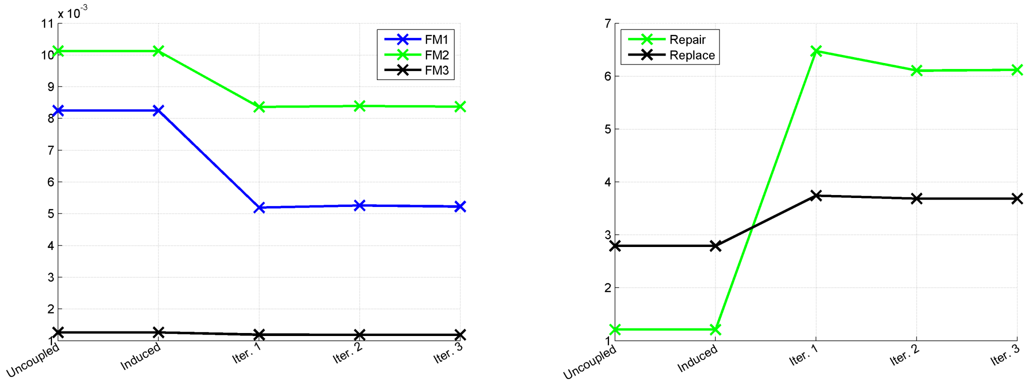


(a) Number of Failures



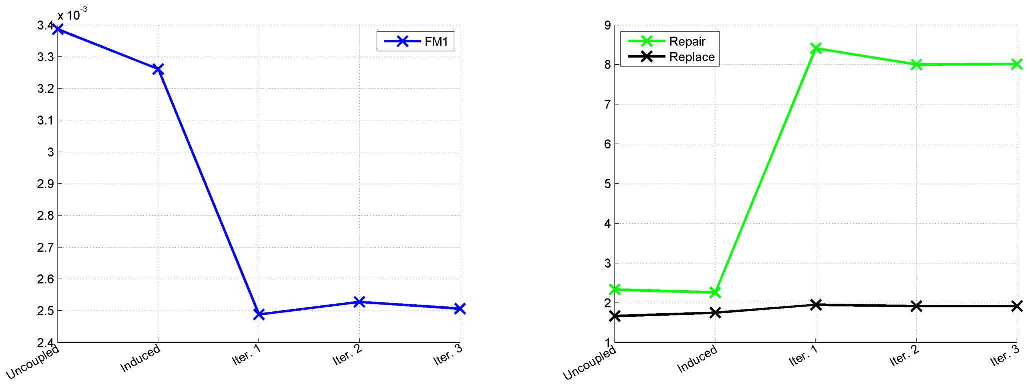
(b) Number of Maint. Actions

Figure 55: Component type 7



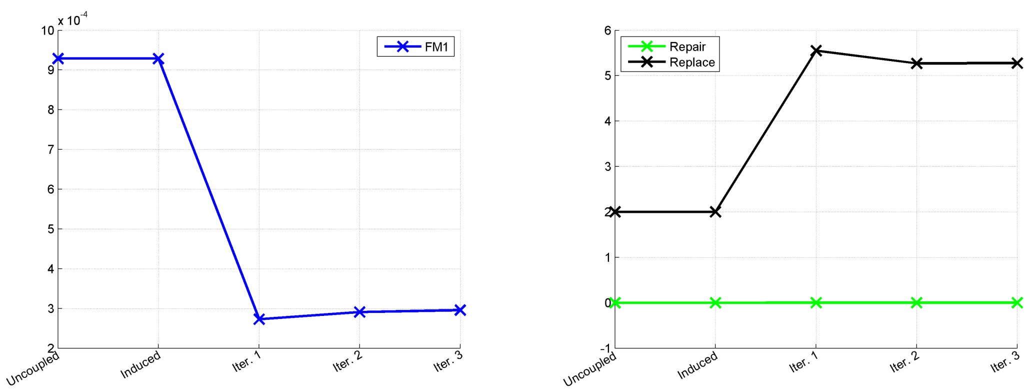
(a) C8, Number of Failures (b) C8, Number of Maint. Actions

Figure 56: Component type 8



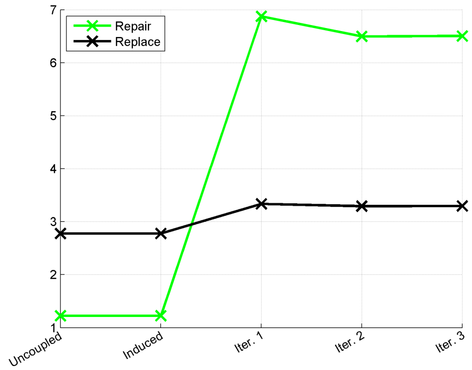
(a) Number of Failures (b) Number of Maint. Actions

Figure 57: Component type 9



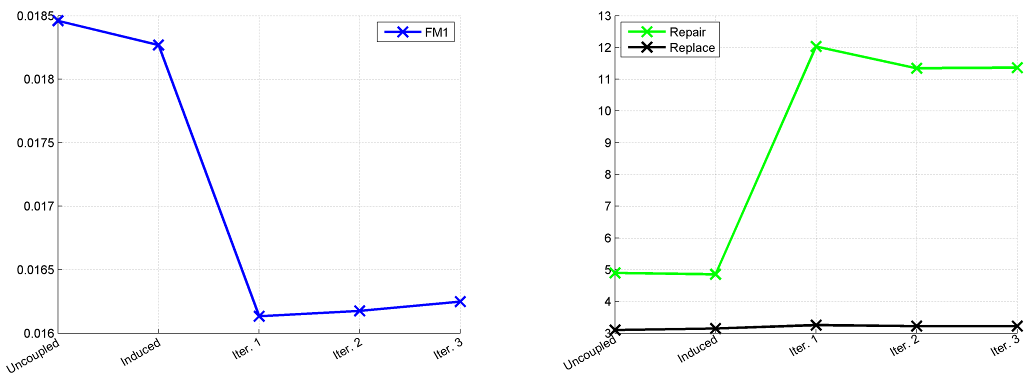
(a) Number of Failures (b) Number of Maint. Actions

Figure 58: Component type 10



(a) Number of Maint. Actions

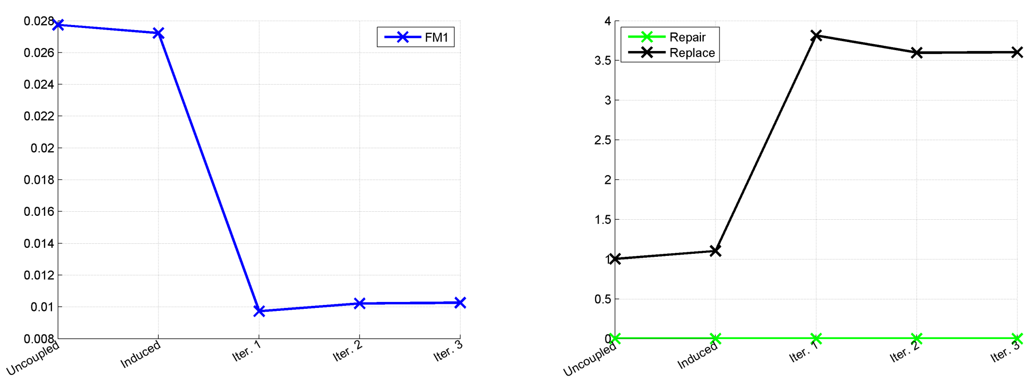
Figure 59: Component type 11



(a) Number of Failures

(b) Number of Maint. Actions

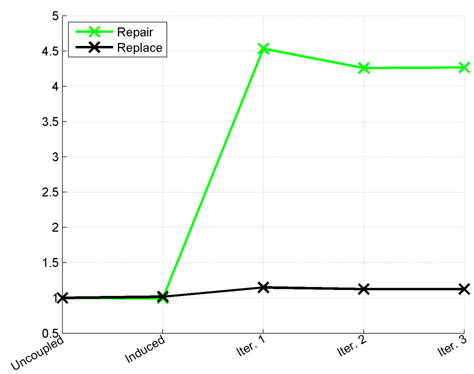
Figure 60: Component type 12



(a) Number of Failures

(b) Number of Maint. Actions

Figure 61: Component type 13



(a) Number of Maint. Actions

Figure 62: Component type 14

REFERENCES

[1] Alkali, B., "Evaluation of generalised proportional intensities models with application to the maintenance of gas turbines," *Quality Reliability Engineering International*, vol. 28, pp. 577–584, 2012.

[2] Balkema, A. and de Haan, L., "Residual life time at great age," *Annals of Probability*, vol. 2, pp. 792—804, 1974.

[3] Barlow, R. E. and Proschan, F., *Mathematical theory of reliability.* John Wiley, New York, 1965.

[4] Bedford, T. and Alkali, B. M., "Competing risks and opportunistic informative maintenance," *Proceedings of the Institution of Mechanical Engineers Part O-Journal of Risk and Reliability*, vol. 223, pp. 363–372, Dec 2009.

[5] Birolini, A., *Reliability Engineering: Theory and Practice.* Springer, 6th ed., 2010.

[6] Blischke, W. R. and Murthy, D. N. P., *Reliability. Modeling, Prediction, and Optimization.* John Wiley and Sons, 2000.

[7] Bobbio, A. and Trivedi, K. S., "An aggregation technique for the transient analysis of stiff Markov chains," *IEEE Transactions on Computers*, vol. 35, no. 9, pp. 803–814, 1986.

[8] Burtin, Y. and Pittel, B., "Asymptotic estimates of the reliability of a complex system," *Engineering Cybernetics*, vol. 10, no. 3, pp. 445–451, 1972.

[9] Carrillo, M. J., "Extensions of Palm's theorem: A review.," *Management Science*, vol. 37, no. 6, pp. 739 – 744, 1991.

[10] Chern, M.-S., "On the computational complexity of reliability redundancy allocation in a series system," *Operations Research Letters*, vol. 11, no. 5, pp. 309 – 315, 1992.

[11] Cho, D. I. and Parlar, M., "A survey of maintenance models for multi-unit systems," *European Journal of Operations Research*, vol. 51, pp. 1–23, 1991.

[12] Christer, A. H. and Waller, W. M., "An operational research approach to planned maintenance: Modelling P.M. for a vehicle fleet," *Journal of the Operational Research Society*, vol. 34, pp. 1041–1052, 1984.

[13] Coit, D. and Smith, A., "Reliability optimization of series-parallel systems using a genetic algorithm," *Reliability, IEEE Transactions on*, vol. 45, pp. 254–260, 266, Jun 1996.

[14] Courtney, T., Gaonkar, S., Keefe, K., Rozier, E., and Sanders, W., "M ́obius 2.3: An extensible tool for dependability, security, and performance evaluation of large and complex system models," in *Dependable Systems Networks, 2009. DSN '09. IEEE/IFIP International Conference on*, pp. 353–358, June 2009.

[15] Courtois, P. J., *Decomposability: Queueing and computer system applications.* New York: Academic Press, 1977.

[16] Dagpunar, J., "Maintenance model with opportunities and interrupt replacement options," *Journal of the Operational Research Society*, vol. 47, no. 11, pp. 1406 – 1409, 1996.

[17] Das, A. N. and Sarmah, S. P., "Preventive replacement models: an overview and their application in process industries," *European Journal of Industrial Engineering*, vol. 4, no. 3, pp. 280–307, 2010.

[18] David, R. and Alla, H., *Discrete, continuous, and hybrid Petri nets.* Springer, 2nd ed., 2010.

[19] Dekker, R. and Dijkstra, M., "Opportunity-based age replacement: exponentially distributed times between opportunities," *Naval Research Logistics*, vol. 39, no. 2, pp. 175 – 90, 1992.

[20] Dekker, R., "Applications of maintenance optimization models: a review and analysis," *Reliability Engineering and System Safety*, vol. 51, pp. 229–240, 1996.

[21] Dekker, R. and Smeitink, E., "Opportunity-based block replacement," *European Journal of Operational Research*, vol. 53, no. 1, pp. 46 – 63, 1991.

[22] Dekker, R., Wildeman, R. E., and van der Duyn Schouten, F. A., "A review of multi-component maintenance models with economic dependence," *Mathematical Methods of Operations Research*, vol. 45, pp. 411–435, 1997.

[23] Dohi, T., Kaio, N., and Osaki, S., "Renewal processes and their computational aspects," in *Stochastic Models in Reliability and Maintenance* (Osaki, S., ed.), ch. 1, Heidelberg, Germany: Springer-Verlag, 2002.

[24] Dutuit, Y., Châtelet, E., Signoret, J.-P., and Thomas, P., "Dependability modelling and evaluation by using stochastic Petri nets: Application to two test cases," *Reliability Engineering and System Safety*, vol. 55, pp. 117–124, 1997.

[25] Fard, N. and Zheng, X., "Approximate method for non-repairable systems based on opportunistic replacement policy," *Reliability Engineering and System Safety*, vol. 33, no. 2, pp. 277–288, 1991.

[26] Faria, J. A. and Matos, Manuel, A., "An analytical methodology for the dependability evaluation of non-Markovian systems with multiple components," *Reliability Engineering and System Safety*, vol. 74, no. 2, pp. 193–210, 2001.

[27] Fisher, R. and Tippett, L. H. C., "Limiting forms of the frequency distribution of the largest and smallest member of a sample," *Proc. Cambridge Phil. Soc.*, vol. 24, pp. 180–190, 1928.

[28] Frenken, K. and Nuvolari, A., "The early development of the steam engine: an evolutionary interpretation using complexity theory," *Industrial and Corporate Change*, vol. 13, no. 2, pp. 419—450, 2004.

[29] Gaonkar, S. and Sanders, W., "Simultaneous simulation of alternative system configurations," in *Dependable Computing, 2005. Proceedings. 11th Pacific Rim International Symposium on*, pp. 8 pp.–, Dec 2005.

[30] Gertsbakh, I., *Reliability Theory with applications to Preventive Maintenance.* Berlin, Germany: Springer, 2000.

[31] Girault, C. and Valk, R., *Petri nets for systems engineering.* Springer, 2003.

[32] Gnedenko, B. and Ushakov, I., *Probabilistic Reliability Engineering.* New York: John Wiley and Sons, 1995.

[33] Gnedenko, B., "Sur la distribution limite du terme maximum d'une série aléatoire," *Annals of Mathematics*, vol. 44, no. 3, pp. 423–453, 1943.

[34] Guajardo, J. A., Cohen, M. A., Kim, S.-H., and Netessine, S., "Impact of performance-based contracting on product reliability: An empirical analysis," *Management Science*, vol. 58, pp. 961–979, 2012.

[35] Haas, P. J., *Stochastic Petri Nets: Modelling, Stability, Simulation.* Springer Series in Operations Research, Springer, 2002.

[36] Hogg, R. V., McKean, J. W., and Craig, A. T., *Introduction to Mathematical Statistics.* Pearson Prentice-Hall, 6^{th} ed., 2005.

[37] Jensen, K., *Coloured Petri nets: basic concepts, analysis, methods and practical use.* Springer, 1997.

[38] Joyce, W. and Anthony, P., "Failure rate of a cold- or hot-spared component with a lognormal lifetime," *IEEE Transactions on Reliability*, vol. 37, pp. 299–307, 1988.

[39] Khan, R. S. R., Lagana, M. C., Ogaji, S. O. T., Pilidis, P., and Bennett, I., "Risk analysis of gas turbines for natural gas liquefaction," *Journal of Engineering for Gas Turbines and Power*, vol. 133, 2011.

[40] Kharoufeh, J. P. and Cox, S. M., "Stochastic models for degradation-based reliability," *IIE Transactions*, vol. 37, no. 6, pp. 533–542, 2005.

[41] Kingman, J., "Discussion of "the inverse gaussion distribution and its statistical application–a review", by J.L. Folks and R.S. Chhikara," *Journal of the Royal Statistical Society Series B*, vol. 40, pp. 263–289, 1978.

[42] Kobbacy, K. A. H., Fawzi, B. B., Percy, D. F., and E., A. H., "A full history proportional hazards model for PM scheduling," *Quality and Reliability Engineering International*, vol. 13, pp. 187–198, 1997.

[43] Kumar, U. D. and Crocker, J., *Case Studies in Reliability and Maintenance*, ch. Maintainability and Maintenance—A Case Study on Mission Critical Aircraft and Engine Components, pp. 377–398. Wiley, 2003.

[44] Kuo, W. and Prasad, V., "An annotated overview of system-reliability optimization," *Reliability, IEEE Transactions on*, vol. 49, pp. 176–187, Jun 2000.

[45] Kuo, W. and Wan, R., "Recent advances in optimal reliability allocation," *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, vol. 37, pp. 143–156, March 2007.

[46] Laggoune, R., Chateauneuf, A., and Aissani, D., "Opportunistic policy for optimal preventive maintenance of a multi-component system in continuous operating units," *Computers and Chemical Engineering*, vol. 33, pp. 1499–1510, 2009.

[47] Lai, C.-D. and Xie, M., *Stochastic Ageing and Dependence for Reliability*. New York: Springer, 2006.

[48] Lee, Y. K., Mavris, D. N., Volovoi, V. V., Yuan, M., and Fisher, T., "A fault diagnosis method for industrial gas turbines using Bayesian data analysis," *Journal of Engineering for Gas Turbines and Power*, vol. 132, 2010.

[49] Leemis, L. M., *Reliability: probabilistic models and statistical methods*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1995.

[50] Liu, Y., Huang, H.-Z., and Pham, H., "Reliability evaluation of systems with degradation and random shocks," in *Proceedings of the Reliability and Maintainability Symposium*, 2008.

[51] Lollini, P. and Bondavalli, A., "A decomposition-based modeling framework for complex systems," *IEEE Transactions on Reliability*, vol. 58, no. 1, 2009.

[52] Lu, C. and Meeker, W., "Using degradation measures to estimate a time-to-failure distribution.," *Technometrics*, vol. 35, pp. 161–174, 1993.

[53] Marais, K. B. and Saleh, J. H., "Beyond its cost, the value of maintenance; an analytical framework for capturing its net present value," *Reliability Engineering and System Safety*, vol. 94, pp. 644–657, 2009.

[54] Marsan, M., Balbo, G., Conte, G., Donatelli, S., and Franceshinis, G., *Modelling with generalized stochastic Petri nets*. Wiley series in parallel computing, John Wiley and Sons, 1995.

[55] Marsand, M. A., Conte, G., and Balbo, G., "A class of generalized stochastic Petri nets for the performance evaluation of multiprocessor systems," *ACM Transactions on Computer Systems*, vol. 2, pp. 93–122, 1984.

[56] Marshall, A. W. and Olkin, I., *Life Distributions. Structure of Nonparametric, Semiparametric and Parametric Families*. Springer Series in Statistics, Springer, 2007.

[57] Meeker, W. Q. and Escobar, L. A., *Statistical Methods for Reliability Data*. Wiley Series in Probability and Statistics, Wiley and Sons, 1998.

[58] Mobley, R. K., *An Introduction to Preventive Maintenance*. Van Nostrand Reinhold, 1990.

[59] Molloy, M. K., *On the Integration of Delay and Throughput Measures in Distributed Processing Models*. PhD thesis, Department of Computer Science, University of California, Los Angelos, CA, USA, 1981.

[60] Molloy, M. K., "Performance analysis using stochastic Petri nets," *IEEE Transactions on Computers*, vol. 31, pp. 913–917, 1982.

[61] Nakagawa, T., *Shock and Damage Models in Reliability Theory.* Springer, 2007.

[62] Nakagawa, T., *Maintenance Theory of Reliability.* Reliability Engineering, Springer, 2005.

[63] Natkin, S., *Les réseaux de Petri Stochastiques et leur Application a l'Evaluation des Systemes Informatiques.* PhD thesis, Conservatoure National des Arts et Metier, Paris, France, 1980.

[64] Nicolai, R. P. and Dekker, R., *Complex System Maintenance Handbook*, ch. 11, pp. 263–286. Springer, 2008.

[65] Percy, D. F., Kobbacy, K. A. H., and Ascher, H., "Using proportional-intensities models to schedule preventive maintenance intervals," *IMA Journal of Mathematical Applied in Business and Industry*, vol. 9, pp. 289–302, 1998.

[66] Petri, C. A., *Kommunikation mit Automaten.* PhD thesis, University of Bonn, 1962.

[67] Ramdas, K. and Randall, T., "Does component sharing help or hurt reliability? an empirical study in the automotive industry.," *Management Science*, vol. 54, pp. 922–938, 2008.

[68] Ross, S., *Simulation.* Academic Press, 2012.

[69] Scarf, P. A. and Deara, M., "Block replacement policies for a two-component system with failure dependence," *Naval Research Logistics*, vol. 50, pp. 70–87, February 2003.

[70] Schneeweiss, W., *Petri nets for reliability modelling.* LiLOle, 1999.

[71] Signoret, J.-P., Dutuit, Y., Cacheux, P.-J., Folleau, C., Collas, S., and Thomas, P., "Make your Petri nets understandable: Reliability block diagrams driven petri nets," *Reliability Engineering & System Safety*, vol. 113, no. 0, pp. 61 – 75, 2013.

[72] Simon, H. A., "Near decomposability and the speed of evolution," *Industrial and Corporate Change*, vol. 11, no. 3, pp. 587—599, 2002.

[73] Singpurwalla, N., "Survival in dynamic environments," *Statistical Science*, vol. 10, pp. 86–103, 1995.

[74] Stevens, W., Myers, G., and Constantine, L., "Structured design," *IBM Systems Journal*, vol. 13, no. 2, pp. 115–139, 1974.

[75] Thomas, L., "A survey of maintenance and replacement models for maintainability and reliability of multi-item systems," *Reliability Engineering*, vol. 16, no. 4, pp. 297–309, 1986.

[76] Thompson, W. A., J., "On the foundations of reliability," *Technometrics*, vol. 23, no. 1, pp. 1–13, 1981.

[77] Trivedi, K. S. and Geist, R. M., "Decomposition in reliability analysis of fault-tolerant systems," *IEEE Transactions on Reliability*, vol. R-32, no. 5, pp. 463–468, 1983.

[78] Tuffin, B., *La Simulation de Monte Carlo.* Lavoisier, 2010.

[79] Volovoi, V., Valenzuela, R., Bhatnagar, H., Nordlund, S., Davis, F., Villeneuve, F., Guo, Y., and C.Patel, "Component-based approach to maintenance modeling of gas turbines," in *Proceedings of ASME Turbo Expo 2012*, June 2012.

[80] Volovoi, V. V., "Modeling of system reliability using Petri nets with aging tokens," *Reliability Engineering and System Safety*, vol. 84, no. 2, pp. 149–161, 2004.

[81] Volovoi, V. and Valenzuela-Vega, R., "Combined representation of coupling effects in maintenance processes of complex engineering systems," in *European Safety and Reliability Conference, ESREL 2011, 18-22 September 2011, Troyes-France*, 2011.

[82] Volovoi, V. and Valenzuela-Vega, R., "On compact modeling of coupling effects in maintenance processes of complex systems," *International Journal of of Engineering Science*, vol. 59, pp. 193–210, 2012.

[83] Volovoi, V. V., "Stochastic Petri net modeling using spn@," in *Proceeding of RAMS-2006 Symposium*, (Newport Beach, CA), January 2006.

[84] Wang, H., "A survey of maintenance policies of deteriorating systems," *European Journal of Operational Research*, vol. 139, pp. 249–289, 2002.

[85] Wang, H. and Pham, H., *Reliability and Optimal Maintenance.* Springer, 2006.

[86] Xie, M., "On the solution of renewal-type integral equations," *Communications in Statistics - Simulation and Computation*, vol. 18, no. 1, pp. 281–293, 1989.

[87] Yang, G., *Life Cycle Reliability Engineering.* John Wiley and Sons, 2007.

[88] Yu, T.-T., Scanlan, J. P., Crowder, R. M., and Wills, G. B., "Approaches to modeling the gas-turbine maintenance process," *Journal of Computing and Information Science in Engineering*, vol. 12, 2012.

[89] Zio, E., "Reliability engineering: Old problems and new challenges," *Reliability Engineering and System Safety*, vol. 94, pp. 125–141, 2009.

[90] Zio, E., *The Monte Carlo Simulation Method for System Reliability and Risk Analysis.* Springer, 2013.