Critical Systems: Formal Methods and Automated Verification: Joint 21st International Workshop on Formal Methods for Industrial Critical Systems and 16th International Workshop on Automated Verification of Critical Systems, FMICS-AVoCS 2016, Pisa, Italy, pages 81 -96
DOI:10.1007/978-3-319-45943-1_6

Towards the Automated Verification of Weibull Distributions for System Failure Rates

Yu Lu[1,2,**], Alice A. Miller[1], Ruth Hoffmann[1], and Christopher W. Johnson[1]

[1] School of Computing Science, University of Glasgow, Glasgow, UK
[2] School of Aerospace, Transport and Manufacturing, Cranfield University, Cranfield, UK
* y.lu.3@research.gla.ac.uk,
{alice.miller,ruth.hoffmann,christopher.johnson}@glasgow.ac.uk

Abstract. Weibull distributions can be used to accurately model failure behaviours of a wide range of critical systems such as on-orbit satellite subsystems. Markov chains have been used extensively to model reliability and performance of engineering systems or applications. However, the exponentially distributed sojourn time of Continuous-Time Markov Chains (CTMCs) can sometimes be unrealistic for satellite systems that exhibit Weibull failures. In this paper, we develop novel semi-Markov models that characterise failure behaviours, based on Weibull failure modes inferred from realistic data sources. We approximate and encode these new models with CTMCs and use the PRISM probabilistic model checker. The key benefit of this integration is that CTMC-based model checking tools allow us to automatically and efficiently verify reliability properties relevant to industrial critical systems.

Keywords: satellite systems · Weibull distribution · continuous-time Markov chains · semi-Markov chains · probabilistic model checking

1 Introduction

Satellite systems are complex due to the fact that they consist of a large number of interacting subsystems (e.g., gyro / sensor / reaction wheels; control processors (CPs); and telemetry, tracking, and command (TTC)), which ensure redundancy without an unnecessary increase in power or mass requirements. Each subsystem may itself have complex and different failure modes. The failure modes are more complex than for conventional systems because of the limited opportunities for repair except through reconfiguration. A satellite subsystem can suffer whole or partial failures, which may belong to a variety of failure classes. It has been shown that Weibull distributions are able to properly model on-orbit failure behaviours of satellite subsystems [1, 2].

* This research was partially supported by the EC project "ETCS Advanced Testing and Smart Train Positioning System" (FP7-TRANSPORT-314219). The author Yu Lu was funded by the Scottish Informatics and Computer Science Alliance (SICSA).

©2016 Springer. This is the Author Accepted Manuscript.
Please refer to any applicable publisher terms of use.

Failures in satellite subsystems are conveniently modelled using Weibull distributions. Unfortunately such distributions are not amenable to continuous time model checking tools, such PRISM, that mainly support CTMCs with exponentially distributed sojourn time. It has also been shown that it is possible to approximate many common distributions using phase-type distributions such as Erlang distributions and a sum of many exponential distributions (the hyper-exponential distribution), although this has proved computationally difficult [3]. Given the maturity of a CTMC solver such as PRISM, and its focus on minimising state spaces, this difficulty is less of an issue. The aim is to investigate how Weibull distributions can be approximated so that PRISM can be effectively used for model checking based reliability analysis of satellite systems.

Simulation is a commonly used and powerful analysis technique for reliability engineering. It is flexible since it supports arbitrary normal distributions (such as Pareto, Weibull, or Lognormal distributions). However, simulations may take a long time to run as the events (e.g., failure) that we are trying to model may be very rare. In addition, it involves the complex design of valid simulation models and interpretations of simulation results. Probabilistic model checking is a formal method for the specification and verification of complex systems with stochastic behaviours. It allows the additional inclusion of probabilities on transitions, and so gives us the ability to check probabilistic properties, such as, "what is the probability of a failure within 5 years?" The automation of the PRISM is essential for analysing reasonably large and non-trivial Markov models with exponential distributions. CTMC models have been used widely to model reliability and performance of engineering systems or applications. However, the exponentially distributed sojourn time of CTMCs can be unrealistic to model satellite systems that exhibit Weibull failures. PRISM is useful for analysing realistic satellite subsystems, and we can obtain results with high accuracy if good approximations of Weibull distributions can be made without resulting in a state space that is too large to yield to feasibly check.

Model checking of semi-Markov chains is more complicated than that of Markov chains. Techniques for model checking semi-Markov chains have been developed [4, 5], whereas the methods are practically negative or infeasible. In recent years, applying practical probabilistic model checking tools to analyse non-Markov models has attracted a lot of attention. In [6], the authors analyse disk reliability of reasonable sized systems (such as RAID4/5/6) based on non-exponential distributions in PRISM [7]. Approximations of Weibull models are considered in [8], using an M-stage Erlang model, and in [9] where 3-state Hidden Markov Models (HMMs) are used. In both cases, results are contrasted with those obtained via simulation. In [10], a stochastic performance model is constructed and the hyper Erlang distribution of real-world data used in PRISM to analyse a public bus transportation network in Edinburgh. In [11], phase-type distributions are used to analyse a collaborative editing system in PRISM.

Our paper is organised as follows. In Section 2, we define semi-Markov models that specify failures of satellite subsystems based on the Weibull distributions, while in Section 3 we give technical background on CTMCs and PRISM. In Sec-

tion 4, we summarise our technique to approximate the Weibull distributions. In Section 5, approximations of these semi-Markov models as CTMCs are developed in PRISM and their benefits are investigated. Finally, in Section 6 we conclude and outline directions for future research.

2 Multi-state Failure Mode in Satellite Subsystems

We propose an approach to building semi-Markov models for reliability analysis of satellite subsystems using a real-world database. The main data source consists of 1584 Earth-orbiting satellites which were launched between January 1990 and October 2008, and are provided by the SpaceTrak database[1]. The SpaceTrak launch and satellite analytical system and its database are used by most global key launch providers, satellite manufacturers, insurance companies, and satellite operators. It provides a variety of data and important information about satellite on-orbit failures and unexpected behaviour, and also launch attempts from 1957. This has enabled us to predict and analyse failure rates.

One of the problems with stochastic approaches on-orbit is the lack of prior validation given the specialised nature of many designs. Common core components e.g. NOAH and the DoD have a core platform that is then configured but many components and architectures are unique. The database used here is likely to provide a conservative base case but is not tailored to specific missions.

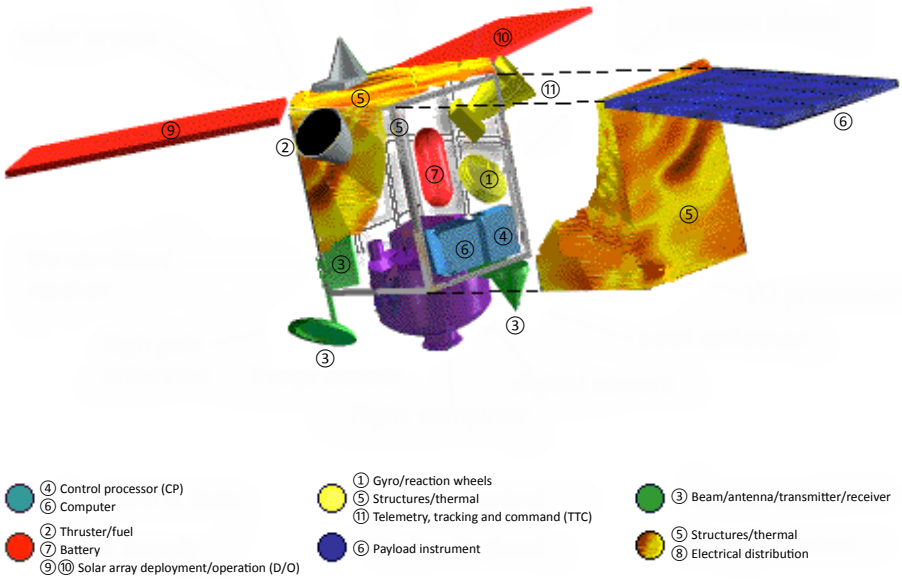


Fig. 1. An overview of key satellite subsystems

[1] http://www.seradata.com/

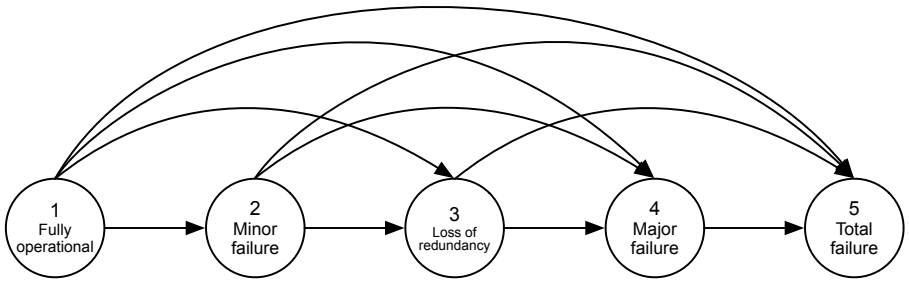


Fig. 2. Multi-state transitions for failure behaviour of satellite subsystems

The database contains several satellite subsystems. In this paper, we only consider 11 subsystems (as shown in Fig. 1). These are: (1) Gyro / sensor / reaction wheel, (2) thruster / fuel, (3) beam / antenna operation / deployment (4) control processor (CP), (5) mechanisms / structures / thermal, (6) payload instrument / amplifier / on-board data / computer / transponder, (7) battery / cell, (8) electrical distribution, (9) solar array deployment (SAD), (10) solar array operating (SAO), (11) telemetry, tracking and command (TTC), and one additional category, which is (12) unknown: when the subsystem causing the failure of the satellite could not be identified.

Unlike traditional binary models of reliability analysis for which satellite subsystems are considered to be either fully operational or suffering a complete failure, additional intermediate states which characterise partial failures are introduced (as shown in Fig. 2). This multi-state modelling approach provides more insights into the failure behaviours of a satellite system and their relationship to total failure through a finer level abstraction. These states are also defined in the SpaceTrak database, and their meanings are summarised as follows:

- State 1: satellite subsystem is fully operational;
- State 2: minor, temporary, or repairable failure that does not cause a substantial and perpetual effect on the operation of the satellite subsystem;
- State 3: major or non-repairable failure that results in loss of redundancy[2] to the operation of the satellite subsystem on a permanent basis;
- State 4: major or non-repairable failure that influences operation of the satellite subsystems on a permanent basis;
- State 5: drastic failure results in satellite retirement, which implies total failure of the satellite.

[2] redundancy: the duplication of critical components or functions of a satellite subsystem.

3 Preliminaries

3.1 Continuous-Time Markov Chains

Satellite failure events occur with a real valued rate. It is therefore natural for us to model our systems as continuous time Markov chains (CTMCs). In a CTMC, time is continuous and state changes can happen at any time. The formal definition of a CTMC is given in Definition 1. This definition is from [12].

Definition 1 *Let AP be a fixed, finite set of atomic propositions. Formally, a continuous-time Markov chain (CTMC) $\mathcal{C}$ is a tuple (S,s_{init},R,L) where:*

- $S = \{s_1, s_2, ..., s_n\}$ *is a finite set of states.*
- $s_{init} \in S$ *is the initial state.*
- $R: S \times S \to \mathbb{R}_{\geq 0}$ *is the transition rate matrix.*
- $L: S \to 2^{AP}$ *is a labelling function which assigns to each state $s_i \in S$ the set $L(s_i)$ of atomic propositions $a \in AP$ that are valid in s_i.*

where $R(s_i, s_j)$ specifies that the probability of moving from s_i to s_j within t time units is $1 - e^{-R(s_i,s_j)\cdot t}$, an exponential distribution with rate $R(s_i, s_j)$. We approximate the semi-Markov chains in Fig. 3 using the underlying semantics of CTMCs. A semi-Markov chain is a model in which state holding times are governed by general distributions, which is a natural extension of CTMCs.

In Fig. 3, not all transitions exist between states for most subsystems as they are not present in the database. For example, no transition from a minor failure (state 2) to a total failure (state 5) of thruster / fuel was ever recorded on orbit for this subsystem in the database. Other transitions also do not occur in the database, so the total number of transitions is reduced. For this reason, they are not subject to formal analysis.

3.2 The PRISM Model Checker

We use the model checker PRISM [7] to obtain CTMC approximations of our multi-state failure models. It supports the analysis of several types of probabilistic models: Discrete-Time Markov Chains (DTMCs), CTMCs [13], Markov Decision Processes (MDPs) [14], and Probabilistic Timed Automata (PTAs) [15], with optional extensions of costs and rewards. PRISM models are expressed using the PRISM modelling language, which is based on the Reactive Modules formalism [16]. A PRISM model consists of the parallel composition of a number of *modules*. Each module is declared in the following way:

module *name* ... **endmodule**

A module consists of a list of variable declarations and a list of commands. At any moment, the *state* associated with a PRISM model is a valuation of all of the variables in the specification. A variable declaration consists of a variable name together with a list of possible values and an initial value. E.g.:

x : [0..4] **init** 0;

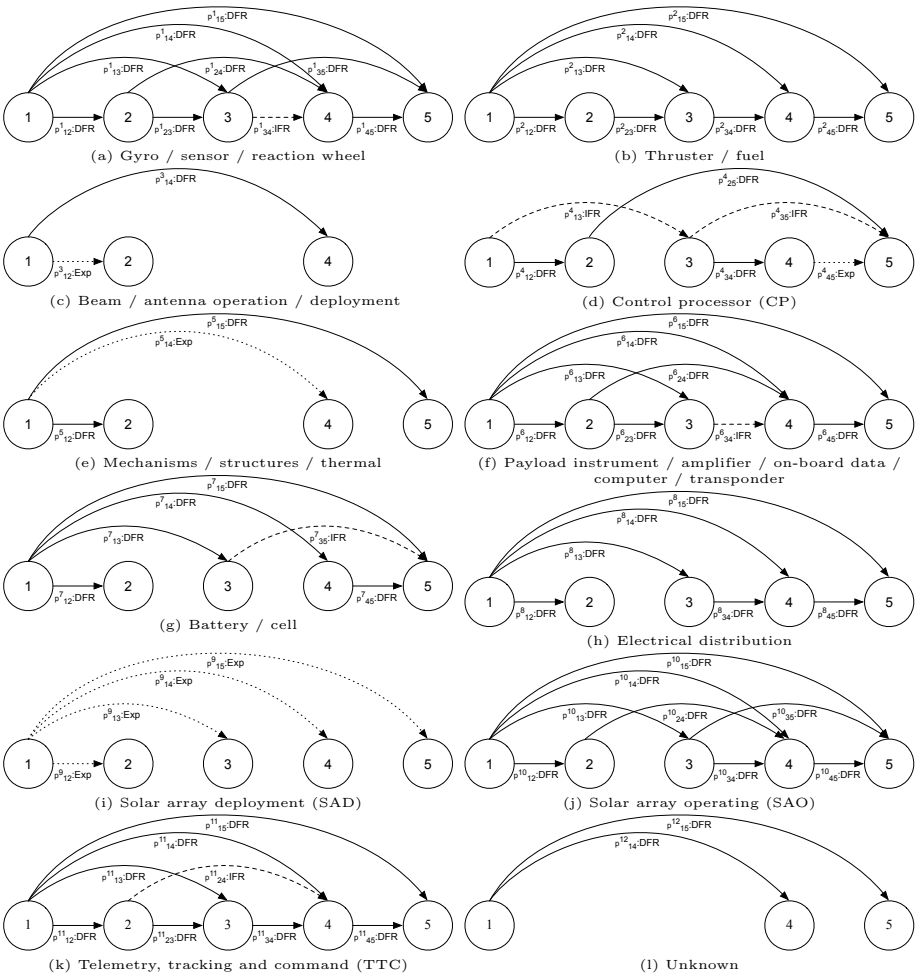


Fig. 3. Semi-Markov chains for multi-state failure mode of satellite subsystems: dotted arrows represent transitions following an exponential distribution (Exp) or Weibull distribution with increasing failure rate (IFR), and solid arrows represent transitions following a Weibull distribution with decreasing failure rate (DFR)

Every command consists of a guard and a non-deterministic choice of updates. Each update has an associated real-value rate. For example:

$$[syncLabel]\ guard\ \rightarrow\ rate_1 : update_1 + rate_2 : update_2 + ...$$

Note that the initial label (*syncLabel* in this example) is optional, and allows for multi-module synchronisation.

3.3 Continuous Stochastic Logic

In this paper, we use Continuous Stochastic Logic (CSL) [17] to specify properties. There are two types of formulae in CSL: state formulae, which are true or false in a specific state, and path formulae, which are true or false along a specific path. One of the most important operators is the **P** operator, which is used to reason about the probability of an event. The **P** operator is applicable to all types of models supported by PRISM. It is often useful to compute the actual probability that some behaviour of a model is observed. Thus, a variation of the **P** operator to be used in PRISM, i.e., $\mathbf{P}_{=?}[pathprop]$, which returns a numerical rather than a Boolean value (i.e., the probability that *pathprop* is true). For example, we might wish to calculate the probability that $j = 1$ is true within the first T time units. This can be specified as $\mathbf{P}_{=?}[\mathbf{F} \leq T\ j = 1]$, where **F** is the "eventually" temporal operator.

4 Approximation of Weibull Failure Models

4.1 Weibull Distributions

In systems engineering, the Weibull distribution [18] is one of the most extensively used lifetime distributions for reliability analysis. It includes two parameters: (1) the shape parameter γ and (2) the scale parameter α, together with key formulas such as cumulative density function (CDF) and probability density function (PDF). A Weibull PDF is expressed as:

$$f(t;\gamma,\alpha) = \frac{\gamma}{\alpha}(\frac{t}{\alpha})^{\gamma-1}e^{-(\frac{t}{\alpha})^{\gamma}}, t \geq 0, \gamma, \alpha > 0 \tag{1}$$

and a Weibull CDF as:

$$F(t;\gamma,\alpha) = 1 - e^{-(\frac{t}{\alpha})^{\gamma}} \tag{2}$$

We abbreviate f(t) and F(t) as the PDF and CDF of the Weibull distribution respectively, then the instantaneous failure rate is $\frac{f(t)}{1-F(t)}$. The failure rate is proportional to a power of time t. The shape parameter, γ, is equal to this power plus one.

The semantics of the Weibull distributions (also known as the bathtub curve) with different γ can be shown in Fig. 4 and explained as follows: (1) $\gamma < 1$ means that the failure rate decreases over time (decreasing failure rates). This occurs whenever a clear infant mortality[3] exists, and the failure rate decreases over time as the failure is discovered and the subsystem removed; (2) $\gamma = 1$ means that the failure rate is constant at any time. This is the useful life of the satellite ; (3) $\gamma > 1$ means that the failure rate increases with time (increasing failure rates). It occurs whenever a wear out exists, or a subsystem failure becomes more likely over time.

[3] infant mortality: a subsystem fails early due to defects designed into or built into it.

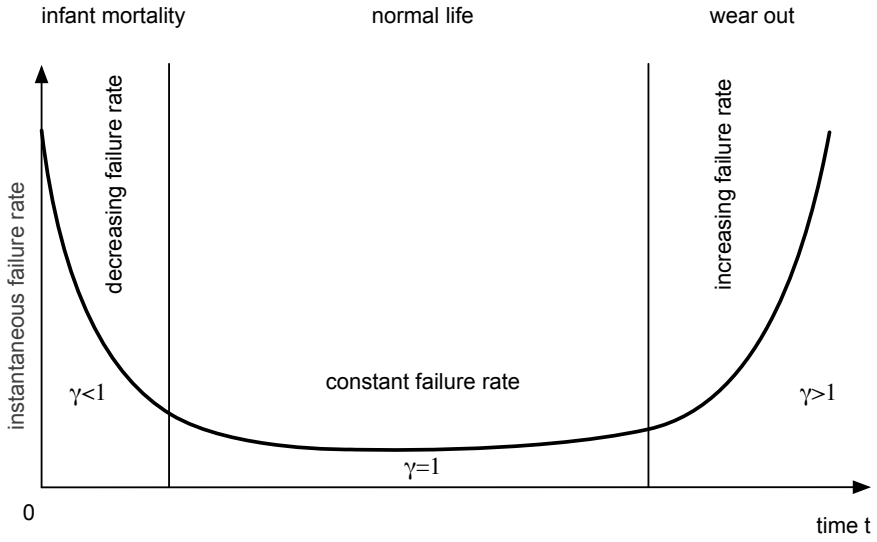


Fig. 4. Semantics of the Weibull distribution (the bathtub curve)

Generally, the ways to approximate the Weibull distributions is non-trivial. The simple technique of phase-type distributions is useful in some cases. Thus, we follow this line of work that Weibull IFR approximated by a M-stage Erlang distribution and Weibull DFR by a hyper-exponential distribution since there are intuitive and strong justifications for the model [3, 8]. Further, these general distributions provide simple mathematical structures such that the their underlying semi-Markov chains can be included in the Markov model framework.

4.2 Increasing Failure Rates (IFR)

A simple technique for the realisation of approximations to the Weibull distribution models is *matching moments*, where the mean is the first moment and the variance the second. We first consider the approximation of a Weibull distribution modelling increasing failure rates (IFR) using an M-stage Erlang distribution [19], which belongs to the class of phase-type distributions. The M-stage Erlang PDF can be expressed as:

$$f(t; M, \lambda) = \frac{\lambda^M}{\Gamma(M)} x^{M-1} e^{-\lambda x}, t \geq 0, \lambda > 0 \tag{3}$$

The Erlang CDF can be expressed as:

$$F(t; M, \lambda) = 1 - e^{-\lambda x} \sum_{n=0}^{M-1} \frac{(\lambda t)^n}{n!} \tag{4}$$

Table 1. Difference between the Weibull distribution with IFR and its approximation as an Erlang distribution: i is the index of the semi-Markov chain for the corresponding satellite subsystem, and xy is the transition from state x to state y

P^i_{xy}	Weibull distribution with IFR		Erlang distribution	
	γ	α	k	λ
P^1_{34}	1.1593	17	2	0.1239
P^4_{13}	1.1229	664	2	0.0031
P^4_{35}	1.0366	15	2	0.1353
P^6_{34}	1.2452	16	5	0.3352
P^7_{35}	28.6487	9	20	2.2652
P^{11}_{24}	2.8232	23	3	0.1464

According to [8], we have the first two moments of the M-Erlang:

$$m_1 = \frac{M}{\lambda},\ m_2 = \frac{M(M+1)}{\lambda^2} \tag{5}$$

As a result, we have:

$$M = \frac{m_1^2}{m_2 - m_1^2},\ \lambda = \frac{m_1}{m_2 - m_1^2} \tag{6}$$

where m_1 and m_2 are equal to the first two moments of the Weibull distribution with IFR, and are given as follows:

$$m_1 = \alpha\Gamma(\frac{\gamma+1}{\gamma}),\ m_2 = \alpha^2\Gamma(\frac{\gamma+2}{\gamma}) \tag{7}$$

The value of M is rounded to the nearest integer and the value of λ recalculated depending on this rounded value, so that the mean is matched.

For example, we consider Weibull parameters for the control processor. The Weibull parameters for the reliability of this subsystem are given by: $\gamma = 1.4560$, $\alpha = 408$ (years). Then, according to equations (6)-(8), $M = 2$ and $\lambda = 0.0054$ for the M-Erlang distribution. Using the Erlang distribution, the approximation result of the Weibull distribution with increasing failure rate for the relevant satellite subsystems is given in Table 1.

4.3 Decreasing Failure Rates (DFR)

The procedure for approximating the Weibull distribution with decreasing failure rates (DFR) by hyper-exponential distributions [20] can be summarised as follows, for details see [3].

First, we choose the number k of exponential components and k arguments: $m_1 > ... > m_i > m_{i+1} > ... > m_k$, for which the ratios $\frac{m_i}{m_{i+1}}$ have to be sufficiently small (e.g., $\frac{m_i}{m_{i+1}} \geq 10$).

Second, we choose the number n such that for all i, $1 < n < \frac{m_i}{m_{i+1}}$.

Then, for the Weibull distribution CDF (see equation (3)), we have a complementary CDF (CCDF) given by:

$$F^c(t;\gamma,\alpha) = 1 - F(t;\gamma,\alpha) = e^{-(\frac{t}{\alpha})^\gamma} \tag{8}$$

and we choose λ and p_1 to match the CCDF $F^c(t;\gamma,\alpha)$ (we abbreviate $F^c(t;\gamma,\alpha)$ as $F^c(t)$) at the arguments m_1 and nm_1, so we solve the following equation:

$$p_1 e^{-\lambda_1 m_1} = F^c(m_1),\ p_1 e^{-\lambda_1 n m_1} = F^c(nm_1) \tag{9}$$

for p_1 and λ_1. As a result, we obtain:

$$\lambda_1 = \frac{1}{(n-1)m_1} ln\left(\frac{F^c(m_1)}{F^c(nm_1)}\right),\ p_1 = F^c(m_1)e^{\lambda_1 m_1} \tag{10}$$

Then, for $2 \le i \le k$, we have:

$$F_i^c(m_i) = F^c(m_i) - \sum_{j=1}^{i-1} p_j e^{-\lambda_j m_i},\ F_i^c(nm_i;) = F^c(nm_i) - \sum_{j=1}^{i-1} p_j e^{-\lambda_j n m_i} \tag{11}$$

and similarly, we solve the further equation:

$$p_i e^{-\lambda_i m_i} = F_i^c(m_i),\ p_i e^{-\lambda_i n m_i} = F_i^c(nm_i) \tag{12}$$

for p_i and λ_i when $2 \le i \le k-1$. As a result, we obtain:

$$\lambda_i = \frac{1}{(n-1)m_i} ln\left(\frac{F_i^c(m_i)}{F_i^c(nm_i)}\right),\ p_i = F_i^c(m_i)e^{\lambda_i m_i} \tag{13}$$

Finally, for $i = k$, we can have:

$$p_k = 1 - \sum_{j=1}^{k-1} p_j,\ p_k e^{-\lambda_k m_k} = F_k^c(m_k),\ \lambda_k = \frac{1}{m_k} ln\left(\frac{p_k}{F_k^c(m_k)}\right) \tag{14}$$

Using the hyper-exponential distribution, the approximation result of the Weibull distribution with decreasing failure rate for the relevant satellite subsystems is given in Table 2. For clarity, we only give the distribution for the subsystem (1), which is Gyro/sensor/reaction wheel.

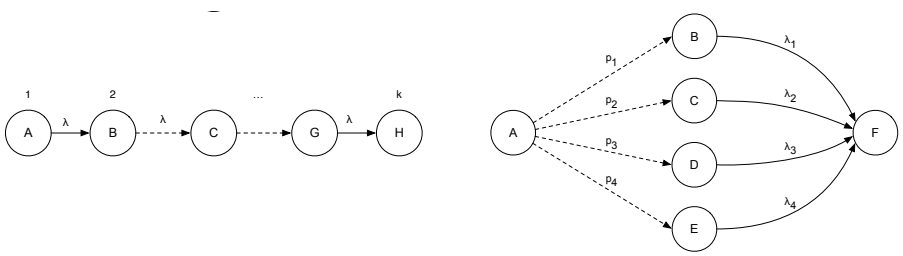
5 Encoding the Weibull Models with CTMCs in PRISM

5.1 Encoding the Weibull distribution with IFR

The approximation of the non-exponential sojourn time distributions can be realised via the insertion of one or more intermediate states between any existing deterioration transition. We approximate a Weibull IFR with an Erlang distribution. In Fig. 5(a), $\frac{k}{\lambda}$ is the time taken for transition from state A to state H.

Table 2. Difference between the Weibull distribution with DFR and its approximation as a hyper-exponential distribution: i is the index of the semi-Markov chain for the corresponding satellite subsystem, and xy is the transition from state x to state y

P^i_{xy}	Weibull distribution with DFR		Hyper-exponential distribution							
	γ	α	p_1	λ_1	p_2	λ_2	p_3	λ_3	p_4	λ_4
P^1_{12}	0.4482	12,526	0.8149	0.000117	0.1258	0.0038	0.0384	0.0433	0.0210	0.8802
P^1_{13}	0.4334	80,050	0.9074	0.000052	0.0630	0.0037	0.0189	0.0434	0.0108	0.9015
P^1_{14}	0.3815	210,126	0.9133	0.000039	0.0548	0.0038	0.0188	0.0444	0.0131	0.9903
P^1_{15}	0.5635	65,647	0.9518	0.000045	0.0377	0.0034	0.0077	0.0408	0.0028	0.7348
P^1_{23}	0.8229	59	0.0933	0.007895	0.6383	0.0132	0.2326	0.0458	0.0359	0.5320
P^1_{24}	0.5600	4,003	0.7852	0.000218	0.1631	0.0037	0.0378	0.0411	0.0139	0.7382
P^1_{35}	0.7115	221	0.3461	0.001866	0.5000	0.0058	0.1258	0.0404	0.0281	0.6022
P^1_{45}	0.4703	135	0.2068	0.000988	0.4133	0.0058	0.2396	0.0466	0.1404	0.8653



(a) Modelling the Weibull distribution with IFR (b) Modelling the Weibull distribution with DFR

Fig. 5. Modelling the Weibull distribution with CTMCs

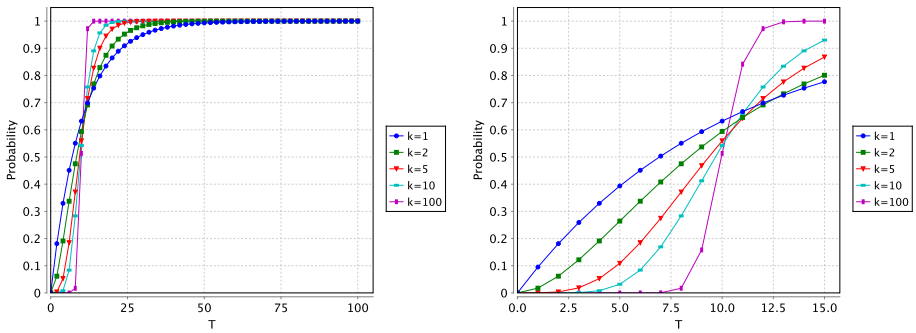
```
ctmc
const int k;
const double mu = 10/k;
module erlang
        i : [1..k+1];
        [] i < k -> 1/mu : (i' = i + 1);
        [sync] i = k -> 1/mu : (i' = i + 1);
endmodule
module weibull_ifr
        j : [0..1];
        [sync] j = 0 -> (j' = 1);
endmodule
```

Fig. 6. Encoding the Weibull distribution with IFR in PRISM

Thus, in order to approximate the interval, the total number of existing deterioration transitions is $k-1=7$. The transition rate is proportional to k, ensuring a constant total transition time.

Consider the PRISM model in Fig. 6. Labelled action *sync* occurs with an Erlang distribution with scale μ and shape k. For the purpose of the analysis, the CSL formula used is: $\mathbf{P}_{=?}[\mathbf{F} \leq T\ j = 1]$, expressing the probability that a

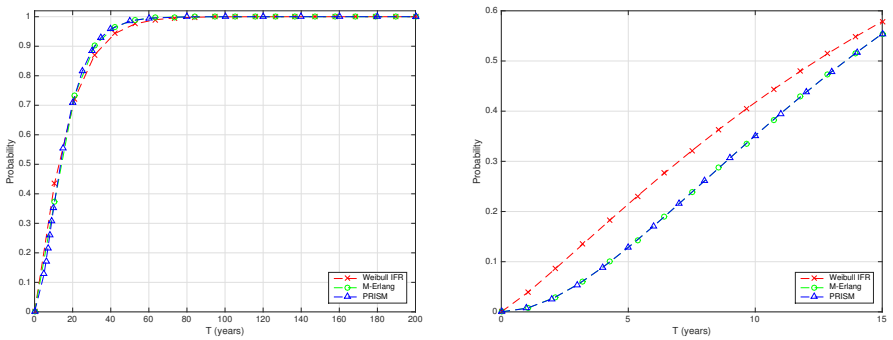
satellite subsystem will fail in T years. In Fig. 7, we show the probability curve of the sojourn time for various values of k, where $k = 1, 2, 5, 10, 100$.



(a) T=100 (b) T=15

Fig. 7. Results of encoding the Weibull distribution with IFR in PRISM

Fig. 7 shows the results of using PRISM (on our CTMC model) to approximate the probability distribution with a constant sojourn time (i.e. of $\mathbf{P}_{=?}[\mathbf{F} \leq T\ j = 1]$ for various values of k, where $k = 1, 2, 5, 10, 100$) for both 100 years and 15 years. This is useful for modelling failure rates with multiple states, while guaranteeing the Markov property. In addition, a significant trade-off exists between the accuracy and the underlying expansion in the state space of the model. For example, when $k = 100$, we can see from Fig. 7(a), that the approximation is very close to the actual distribution. However, increasing k by a factor of 100 increases the size of the underlying model by 100.



(a) T=200 (b) T=15

Fig. 8. Comparison between the Weibull distribution with IFR, its approximation, and PRISM encoding

To understand the differences better, we compare the CDF of the original Weibull IFR distribution with its approximation as an Erlang distribution and its implementation as a CTMC model in PRISM. As shown in Fig. 8(a), the difference between Weibull and the other two curves apparently tends to zero, indicating the approximation and implementation both to be accurate for right long tail probabilities. In Fig. 8(b), we see that the difference is at most 0.05, this is due to the fact that we lose a little accuracy in order to reduce the size of the state space associated with our PRISM model.

5.2 Encoding the Weibull distribution with DFR

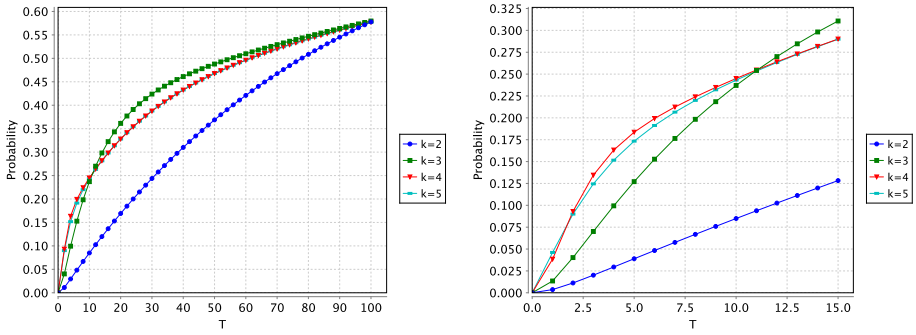
We approximate a Weibull DFR with an hyper-exponential distribution, which is a mixture of exponential distributions. The hyper-Erlang distribution is also a generalisation of the hyper-exponential distribution. So, the hyper-exponential distribution also belongs to the class of phase-type distributions. In general, it can be represented with respect to the time until absorption in a CTMC. For instance, a hyper-exponential distribution having four branches ((p_1,λ_1), (p_2,λ_2), (p_3,λ_3),(p_4,λ_4)) can be represented by a CTMC model as shown in Fig. 5(b). Dotted arrows indicate instantaneous probabilistic transitions, and solid arrows transitions with exponentially distributed durations.

```
ctmc
const double p1, p2, p3, p4, lambda1, lambda2, lambda3, lambda4;
module weibull_dfr
        s : [0..5] init 0;
        [] s = 0 -> p1 : (s' = 1) + p2 : (s' = 2) + p3 : (s' = 3) +
                    p4 : (s' = 4);
        [] s = 1 -> lambda1 : (s' = 5);
        [] s = 2 -> lambda2 : (s' = 5);
        [] s = 3 -> lambda3 : (s' = 5);
        [] s = 4 -> lambda4 : (s' = 5);
endmodule
```

Fig. 9. Encoding the Weibull distribution with DFR in PRISM

In Fig. 9, we encode the behaviour of the CTMC in Fig. 5(b) using PRISM. For CTMC, updates in commands are labelled with positive-valued rates, rather than probabilities. Since there are four transitions leaving state 0 which are all instantaneous, if we make the probabilistic choice between them, the states with instantaneous transitions can be removed to construct the underlying CTMC.

Fig. 10 shows the results of using PRISM (on our CTMC model – see Fig. 9) to approximate the probability distribution of a constant sojourn time (i.e. of $\mathbf{P}_{=?}[\mathbf{F} \leq T\ s = 5]$ for $k = 2, 3, 4, 5$ for both 100 years and 15 years). Although there is trade-off between the accuracy and the size of the resulting state space between $k = 2$ and $k = 4$, the difference is not so obvious between $k = 4$ and



(a) T=100 (b) T=15

Fig. 10. Results of encoding the Weibull distribution with DFR in PRISM

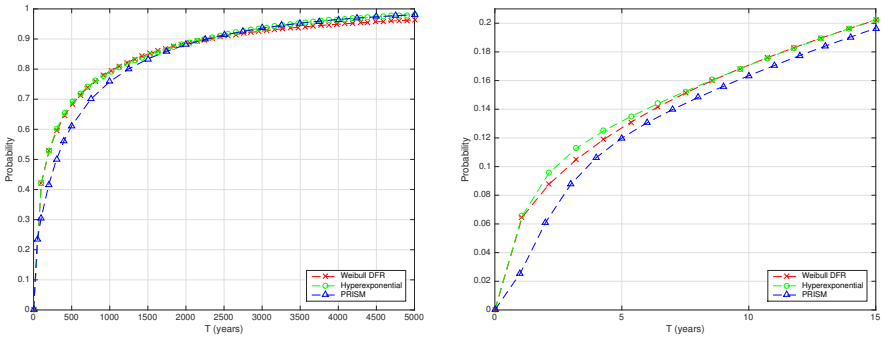
$k = 5$. Therefore, we consider $k = 4$ to be a good approximation parameter for the implementation of Weibull DFR in PRISM.

For the same purpose, we compare the CDF of the original Weibull DFR distribution with its approximation in a hyper-exponential distribution and its implementation with a CTMC in PRISM. As shown in Figures 11(a) and 11(b), for a time scale ($\alpha = 5000$ years), the difference between the Weibull DFR and the other two curves in the left short head is at most 0.01, and in the right long tails apparently becomes zero, indicating the approximation and implementation both to be accurate for a short scale for both left short head and right long tail probabilities. Though for a large scale ($\alpha = 50000$ years) in Fig. 11(c), we can see that the difference can be very large in the right long tails. However, in Fig. 11(d), for $T \leq 15$ years, the approximation and implementation both appear to be accurate for large scale and left short head probabilities.

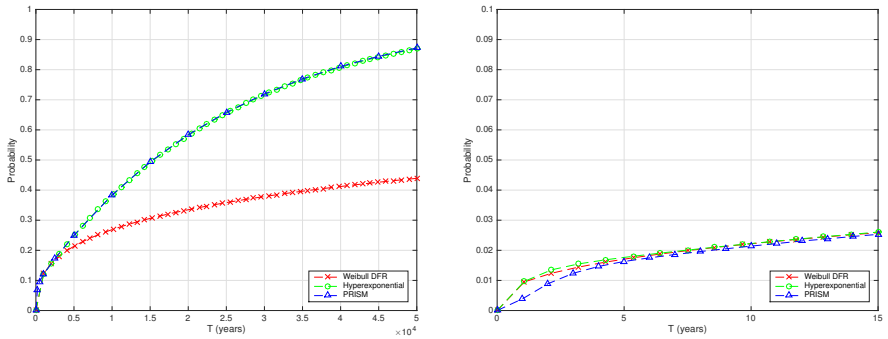
6 Conclusion and Future Work

We have shown that difficulties in modelling the Weibull distribution for satellite failures can be handled if appropriate approximations and modelling methods are considered. We have also proposed novel non-exponential models that characterise failure behaviours, based on Weibull failure modes (both increasing failure rates and decreasing failure rates) inferred from real-world datasets. We have approximated and encoded these new models with CTMCs in PRISM, and shown their approximation is accurate in matching a Weibull distribution in isolation.

The key contribution of this work is that the CTMCs-based formalisms come equipped with mature model checking tools, such as PRISM and so allow a wide range of analyses relevant to industrial critical systems to be performed automatically and efficiently. In future work, it would be interesting to see how their approximation matches the true distribution when multiple distributions are combined, e.g. when constructing a model for an entire satellite or a subset



(a) Weibull DFR with small scale and T has value of 5000 (b) Weibull DFR with small scale and T has value of 15



(c) Weibull DFR with large scale and T has value of 50000 (d) Weibull DFR with large scale and T has value of 15

Fig. 11. Comparison between the Weibull distribution with DFR, its approximation, and PRISM encoding

of subsystems. Another interesting direction is to use various techniques such as symmetry reduction [21, 22] for reducing the state space of the approximation.

References

1. Castet, J.F., Saleh, J.H.: Satellite and satellite subsystems reliability: Statistical data analysis and modeling. Reliability Engineering & System Safety **94**(11) (2009) 1718–1728
2. Castet, J.F., Saleh, J.H.: Beyond reliability, multi-state failure analysis of satellite subsystems: A statistical approach. Reliability Engineering & System Safety **95**(4) (April 2010) 311–322
3. Feldmann, A., Whitt, W.: Fitting mixtures of exponentials to long-tail distributions to analyze network performance models. Performance Evaluation **31**(3-4) (1998) 245–279
4. López, G.G.I., Hermanns, H., Katoen, J.P.: Beyond Memoryless Distributions: Model Checking Semi-Markov Chains. In: Process Algebra and Probabilistic Methods. Performance Modelling and Verification. Volume 2165 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2001) 57–70

5. Kwiatkowska, M., Norman, G., Segala, R., Sproston, J.: Verifying Quantitative Properties of Continuous Probabilistic Timed Automata. In: CONCUR 2000 — Concurrency Theory. Volume 1877 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2000) 123–137
6. Gopinath, K., Elerath, J., Long, D.: Reliability Modelling of Disk Subsystems with Probabilistic Model Checking. Technical Report UCSC-SSRC-09-05, University of California, Santa Cruz (2009)
7. Kwiatkowska, M., Norman, G., Parker, D.: Probabilistic symbolic model checking with PRISM: a hybrid approach. International Journal on Software Tools for Technology Transfer **6**(2) (2004) 128–142
8. Malhotra, M., Reibman, A.: Selecting and implementing phase approximations for semi-Markov models. Communications in Statistics. Stochastic Models **9**(4) (1993) 473–506
9. Xin, Q., Thomas J. E. Schwarz, S.J., Miller, E.L.: Disk infant mortality in large storage systems. In: Proceedings of the 13th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS 2005), IEEE (2005) 125–134
10. Reijsbergen, D., Gilmore, S., Hillston, J.: Patch-based Modelling of City-centre Bus Movement with Phase-type Distributions. Electronic Notes in Theoretical Computer Science **310** (2015) 157–177
11. Ciobanu, G., Rotaru, A.: Phase-Type Approximations for Non-Markovian Systems: A Case Study. In: Software Engineering and Formal Methods. Volume 8938 of Lecture Notes in Computer Science. Springer International Publishing (2015) 323–334
12. Baier, C., Katoen, J.P.: Principles of Model Checking. The MIT Press (2008)
13. Peng, Z., Lu, Y., Miller, A.A., Johnson, C.W., Zhao, T.: Formal Specification and Quantitative Analysis of a Constellation of Navigation Satellites. Quality and Reliability Engineering International **32**(2) (2014) 345–361
14. Lu, Y., Peng, Z., Miller, A., Zhao, T., Johnson, C.: How reliable is satellite navigation for aviation? checking availability properties with probabilistic verification. Reliability Engineering & System Safety **144** (2015) 95–116
15. Peng, Z., Lu, Y., Miller, A.: Uncertainty Analysis of Phased Mission Systems with Probabilistic Timed Automata. In: Proceedings of the 7th IEEE International Conference on Prognostics and Health Management (PHM 2016), IEEE (2016)
16. Alur, R., Henzinger, T.A.: Reactive Modules. Formal Methods in System Design **15**(1) (1999) 7–48
17. Aziz, A., Sanwal, K., Singhal, V., Brayton, R.: Model-Checking Continuous-Time Markov Chains. ACM Transactions on Computational Logic **1**(1) (2000) 162–170
18. Weibull, W.: A statistical distribution function of wide applicability. Journal of Applied Mechanics **18** (1951) 293–297
19. Evans, M., Hastings, N., Peacock, B.: Erlang Distribution. In: Statistical Distributions. 3rd edn. Wiley, New York (2000) 71–73
20. Bolch, G., Greiner, S., de Meer, H., Trivedi, K.S.: Introduction. In: Queueing Networks and Markov Chains: Modeling and Performance Evaluation with Computer Science Applications. Wiley, New York (1998)
21. Miller, A., Donaldson, A., Calder, M.: Symmetry in temporal logic model checking. ACM Computing Surveys **38**(3) (2006)
22. Kwiatkowska, M., Norman, G., Parker, D.: Symmetry Reduction for Probabilistic Model Checking. In: Computer Aided Verification. Volume 4144 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2006) 234–248

https://dspace.org/

Towards the automated verification of Weibull distributions for system failure rates

Lu, Yu; Miller, A. A.; Hoffmann, R; Johnson, C. W.

2016-09-13
Springer
https://dspace.lib.cranfield.ac.uk/handle/1826/11020
Lu Y, Miller AA, Hoffmann R, Johnson CW. (2016) Towards the automated verification of Weibull distributions for system failure rates. Critical Systems: Formal Methods and Automated Verification: Joint 21st International Workshop on Formal Methods for Industrial Critical Systems and 16th International Workshop on Automated Verification of Critical Systems, FMICS-AVoCS 2016, Pisa, Italy, pp. 81-96

Downloaded from a DSpace repository. https://dspace.org/