NTNU
Norwegian University of
Science and Technology

Statistical modeling of maintenance on offshore oil and gas installations

Håkon Husby

Master of Science in Physics and Mathematics

Submission date:	June 2011
Supervisor:	Bo Henry Lindqvist, MATH
Co-supervisor:	Atle Stokke, Det Norske Veritas AS
	John Barry, Det Norske Veritas AS

Norwegian University of Science and Technology
Department of Mathematical Sciences

Problem Description

The purpose of this thesis is to study the maintenance effect and how the maintenance evolves over time. Several models will be described and applied to real world data from offshore oil and gas installations to see how these models describe the maintenance.

Assignment given: 04.02.2011
Supervisor: Bo Henry Lindqvist, MATH
Co-Supervisor: Atle Stokke, Det Norske Veritas AS
John Barry, Det Norske Veritas AS

Preface

This master thesis marks my concluding work in my degree in Master in Applied Physics and Mathematics at the Norwegian University of Science and Technology (NTNU). It has been written in collaboration with Det Norske Veritas AS.

Working with this thesis has been a rewarding experience as we have given access to data from still operating oil and gas installations. It has therefore been very interesting to utilize my knowledge within statistics to analyze how maintenance evolves on oil and gas installations.

I would like to thank my supervisor Professor Bo Henry Lindqvist at the Department of Mathematics at NTNU for interesting discussions and valuable guidance during the work. I'm also thankful for the help and guidance from my co-supervisor Atle Stokke and the rest of my colleagues at Det Norske Veritas AS. I'm grateful for the opportunity I got when I moved to Oslo and became a part of the great working environment at Det Norske Veritas AS at Høvik, Oslo when writing this thesis. I would also like to address thanks to Statoil ASA for given us access to the data. Last, but not least, I would like to thank my family, friends and fellow students for great support during my work on this thesis.

Håkon Husby

Høvik, 30.06.2011

Abstract

In this thesis it has been analyzed how the maintenance will evolve over time for different systems on oil and gas installations. Several statistical models have been proposed to analyze this, minimal repair models (NHPP), perfect repair models (HPP/RP) and imperfect repair models (ARA∞/ARA1). The focus has been on the relationships between these models, the state which the system is left in after maintenance and how good each model fit the given data. The maximum likelihood method has been used for all models when finding estimates for the parameters.

The result after fitting the models to the data are also used to simulate how maintenance will evolve during a 30 year period for a specific plant.

Contents

List of Figures

List of Tables

1 Introduction

The main purpose of this thesis is to study the effects of maintenance and how maintenance evolves over time on different systems on ageing offshore oil and gas installations. We are given data from two offshore platforms in the North Sea where we have information about all maintenance actions done on the platforms during a period of almost 8 years.

The offshore installations in the North Sea are getting older, and replacement of components or systems is becoming evident as they may not give satisfactory results with respect to safety, production or other criteria. The study of the maintenance actions may increase the lifetime of the component/system or reduce the number of insignificant maintenance actions.

One interesting aspect is whether or not the number of maintenance actions increases as the installation gets older. In other words we are interested to estimate how the maintenance evolves over time. By doing this we may get a better estimate on how much maintenance is necessary for a given system when the age of the installation gets older. It may be logical to think that more maintenance actions is necessary in order for the system to give satisfactory result when it gets older. If a system inhibits a trend it may be interesting to see if this trend is statistically significant or not, and if the system has a decreasing or increasing trend. A system with a decreasing trend requires less maintenance as the systems gets older while a system with increasing trend requires more maintenance.

In this thesis we propose several different statistical models to analyze how the maintenance evolves over time. We have taken into account perfect, minimal and imperfect repair models. The first two models, (perfect and minimal), are the ones who are usually used, but we have also proposed two imperfect repair models in addition. We will come back to the discussion of these models in section 3 & 4. In section 2 we will give a description of the systems we are considering and the data we have been given access to. We will also use the results from the analysis of the statistical models to simulate how maintenance will evolve on one of the platforms over a 30 year span in section 5 before we give some concluding remarks on the results we have found in this thesis in section 6.

The appendix contains some additional background on the statistical models we have used along with some definitions of technical terms used in the thesis. Throughout the work with this thesis we have used the statistical software *R*, [1], to do most of the programming involving the statistical models, some of the codes are given in section D in the appendix. The simulation is done with the simulation software *ExtendSim* 8, [2].

2 Description

In this thesis data from two platforms in the North Sea are considered. Throughout this thesis they will be denoted as plant A and plant B.

We have been given access to a datadump from a CMMS (Computerized Maintenance Management System) report for these two platforms which gives us information about all the maintenance work they have done on the different systems on the platform. The dataset describes which system the maintenance belongs to, the date the maintenance started, the date they have changed the system, the plant it has been done on, wether it was preventive or corrective maintenance, the priority of the maintenance, what kind of component it is and an ABC indicator. The ABC indicator is a criticality classification, which implies a classification of each component with regard to consequence of failure. This indicator is linked together with the priority since a component with high criticality will need higher priority in order for the system to still perform the intended function.

The data given is from a 8 year period where there has been registered 13844 events. It may be interesting to note that plant B was set into operation only 1 year before they started recording data in comparison to plant A which has operated 10 years before they started to record data. This may lead to a early conclusion that same system on different platforms may exhibit different trends due to ageing because of the period they have operated so far. An interesting aspect is that the total number of maintenance events done on each platform vary considerably. Plant A has 10549 registered events while plant B only has 3295 registered events. This gives also rise to an early conclusion that plant A may exhibit more ageing trends than plant B.

There are in total 49 different systems on the platforms where there has been registered events. The number of events varies considerably from system to system, from only 1 registered event on system 14 up to 1229 registered events on system 53. This leads to a limitation on some systems, since it may be too few registered events in order for us to get interesting results. Therefore the focus has been mostly on systems where there are many registered events in this thesis, since this gives a better foundation on how maintenance will evolve over time. Let us also note that on some systems there are only registered events on one of the plants. This may be because the design of the plant is different than the other and may not contain that specific system, but it also may come from the fact that no maintenance has been registered. This gives us no ability to compare between the plants, but it is still interesting to see how maintenance evolves over time on the specific plant. In this thesis the main focus will be on systems where there are registered events on both plants.

3 Statistical Models

The statistical models proposed in this section will all be used to analyze how maintenance will evolve over time. Data from the analysis from these models will be supplied to *ExtendSim* 8, [2], to do a simulation of how maintenance will evolve during a 30 year span on one of the plants.

All the systems considered are repairable systems which undergoes repair/-maintenance which can restore the system to a state where it can perform its required function satisfactorily. Several statistical models have been proposed in the literature to model systems in order to capture how maintenance will evolve over time. In our analysis the focus has been on counting process models or stochastic point process models.

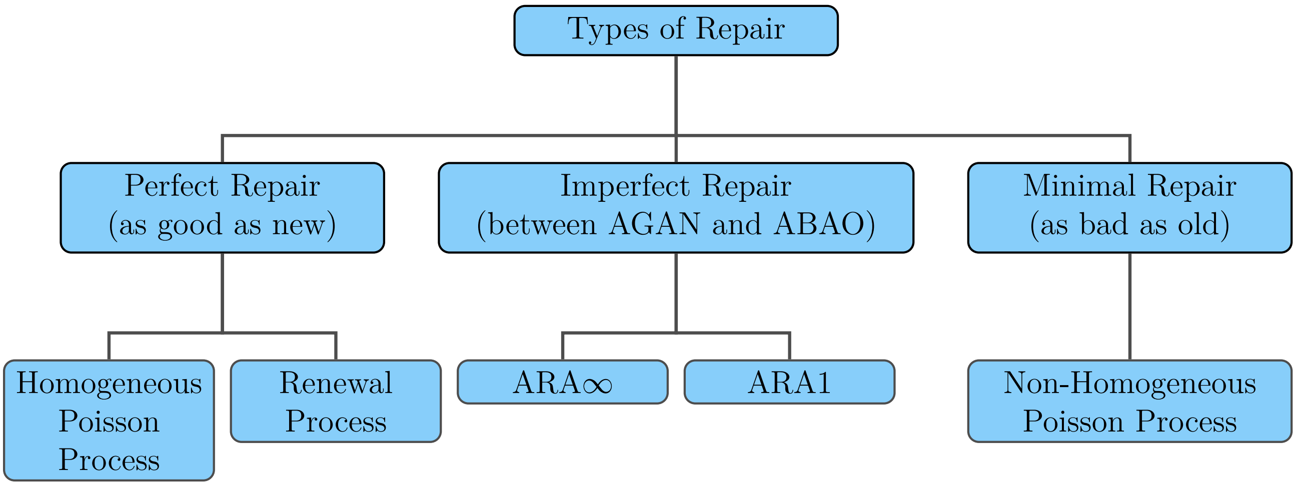


Figure 3.1: Overview of statistical models sorted in categories of repair

In figure 3.1 an overview of the different models used in the analysis of the data are presented. The focus has been on three different categories of repair models; perfect, imperfect and minimal repair. The difference between the categories arises from which state they leave the system in after maintenance is done. A perfect repair leaves the system in a state as good as new, AGAN. Minimal repair leaves the system in a state as bad as old, ABAO, this means that the minimal repair didn't make the system any better than before. And imperfect repair, which is also often denoted normal repair, leaves the system in a state between AGAN and ABAO. One could also argue that an imperfect repair leaves the state of the system in a state worse than old or better than new, but for simplicity and easier interpretation we have made the decision to keep the imperfect repair between AGAN and ABAO. More comments on this will be given in the section for imperfect repair models.

Before continuing to describe these models a few remarks about some definitions that are important for the description of the models. Figure 3.2 show two

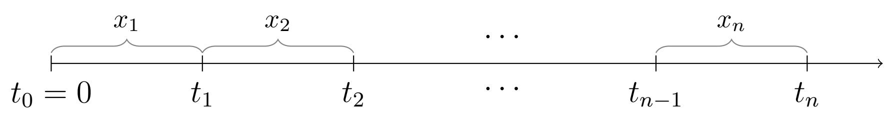


Figure 3.2: Timeline for maintenance events

important definitions of time we need to take into account. t_i describes the time of registered maintenance events and $x_i = t_i - t_{i-1}$ denotes the inter-occurrence time or time between maintenance. Some assumptions of events happening in the x_i-intervals and the relationship between these intervals are similar for all the different models.

- The x_i intervals are disjoint and independent.
- Only one event per day is considered, thus if more than one event happens per day they will be counted as one.
- Each event is equally important, this means that we can't distinguish between criticality of events or how time consuming the events are.
- The repair time is negligible, the interest is only how many events that has happened up to time t.

These assumptions are critical for the modeling of the counting process, having set these in place we will now continue to describe the different models.

3.1 Perfect Repair Models

A perfect repair model leaves the system in a state as good as new, AGAN, after maintenance has been done on the system. This means that after maintenance the system is the same as if we had a new system. In many cases this may be regarded as a optimistic/naive model as a system with many components should not be regarded AGAN if we do maintenance on it, as maintenance is usually done on only a small fraction of the system. Nevertheless, it is interesting to see how a perfect repair model fits the given data to see if maybe this is a good model.

There has been used two different models for this case, homogeneous poisson processes (HPP) and renewal processes (RP). The basic assumption on how these two models work is the difference of the probability distribution for an event occurring in a specific interval. For the HPP model the probability of an event occurring in specific x_i interval follow a exponential distribution with scale parameter α.

$$F_X(x_i) = 1 - \exp\left[-x_i/\alpha\right] \tag{3.1.1}$$

Because the x_i intervals are disjoint and independent, the probability of n number of events happening in the interval $[0, t)$ follow a Poisson distribution with mean $E[N(t)] = t/\alpha$, see section B.3 in the appendix for more details.

Another way to describe the HPP model, see section B.3 in the appendix, is to set the rate of occurrence of maintenance, ROCOM, function to $v(t) = 1/\alpha$. This will give the same model.

The RP model assumes a different probability distribution for an event occurring in a x_i interval. Generally a renewal process can follow any probability distribution in the interval, but in our model we have chosen the Weibull distribution as our basis for the RP model with scale parameter α and shape parameter β.

$$F_X(x_i) = 1 - \exp\left[-\left(\frac{x_i}{\alpha}\right)^{\beta}\right] \tag{3.1.2}$$

A closed form solution for the expected number of maintenance events happening in the interval $[0, t)$ is not attainable unless there are integer values of the shape parameter β, thus the RP model relies on Monte Carlo simulation of the expected number of events. A description of the Monte Carlo simulation will be given later.

Thus the only difference between the models are that the HPP follows an exponential distribution in the x_i intervals while the RP model follow a Weibull distribution. The main difference this leads to is the probability of an event happening in an interval. The exponential distribution has the memoryless property, see equation 3.1.3.

$$\begin{aligned} P(S > s + x | S > x) &= \frac{P(S > s + x)}{P(x)} = \frac{\exp^{-(s+x)/\alpha}}{\exp^{-x/\alpha}} \\ &= \exp^{-s/\alpha} = P(S > s) \end{aligned} \tag{3.1.3}$$

Thus no matter how long the system hasn't experienced maintenance in an interval the probability of an event happening is still the same for the HPP model. For the RP model this is not the case. The scale parameter β in the Weibull distribution makes this not possible for the RP model unless $\beta = 1$. When $\beta = 1$ the Weibull distribution reduces to the exponential distribution and the RP and HPP model may be regarded the same in this case. If $\beta \neq 1$ one may think of local trends in the x_i intervals. For $\beta < 1$ the probability of "surviving" longer without maintenance is greater than in comparison if $\beta > 1$. Thus one may say that it has a decreasing local trend if $\beta < 1$ as the failure rate of the Weibull distribution decreases as time gets larger. While one may say that it has an increasing local trend if $\beta > 1$ since the failure rate increases as time gets larger.

Another concept which is important to note is the effect of global trend. A global trend occur if the expected number of maintenance events doesn't increase linearly with time. For the HPP model we saw that the expected number of maintenance events in an interval $[0, t)$ equals

$$E[N(t)] = t/\alpha \tag{3.1.4}$$

this increases linearly with time, thus the HPP model doesn't have a global trend.

From the elementary renewal theorem, see equation B.4.4, the expected number of maintenance events in an interval $[0, t)$ is approximately

$$E[N(t)] = t/\mu \tag{3.1.5}$$

when t gets large and μ is the mean of the probability distribution for the x_i intervals. Thus $E[N(t)]$ increases linearly also for a RP model. Hence neither the RP nor the HPP model have a global trend.

Comments on how the likelihood function is constructed for the HPP and RP model are seen in section B.7 in the appendix. Differences in the log-likelihood function is readily seen as the log-likelihood for the HPP model is given by

$$l_{HPP}(\alpha|t) = -\,n \log \alpha - \frac{t_n}{\alpha} \tag{3.1.6}$$

while for the RP model the log-likelihood is.

$$l_{RP}(\alpha, \beta|t) = n(\log \beta - \beta \log \alpha) + (\beta - 1) \sum_{i=1}^{n} \log(t_i - t_{i-1}) - \sum_{i=1}^{n} \left(\frac{t_i - t_{i-1}}{\alpha} \right)^{\beta} \tag{3.1.7}$$

The difference between the log-likelihoods is seen to arise from the β parameter. Letting $\beta = 1$ the log-likelihood for the RP model equals the log-likelihood for the HPP model.

$$l_{HPP}(\alpha|t) = l_{RP}(\alpha, \beta = 1|t) \tag{3.1.8}$$

this shows that a RP model with $\beta = 1$ indeed is the same model as the HPP model.

The method of maximum likelihood has been used to find estimates for the parameters in both the HPP and RP model. For the HPP a closed form solution for the maximum likelihood estimator, MLE , of α is easily attainable

$$\hat{\alpha} = \frac{t_n}{n} \tag{3.1.9}$$

For the RP model closed form solutions is not attainable, therefore a numerical maximization procedure has been used to find the MLE's for α and β. In our implementation we have used a quasi-Newton method to solve the maximization problem of the log-likelihood function. A full description of the implementation will not be given here, further details can be seen in section D where the R-codes are given, but a key issue will be highlighted. Both the α and β parameter have a restriction that they have to be greater than zero. Thus before starting the maximization procedure the parameters are transformed such that they may take any value in the range $(-\infty, \infty)$. The transformed parameters are

$$\alpha' = \log(\alpha)$$
$$\beta' = \log(\beta)$$

By doing this and utilizing the invariance property of the MLE's, as described in section B.6 in the appendix, it speeds up the procedure of finding the MLE's.

3.2 Minimal Repair Model

A minimal repair model leaves the system in a state as bad as old, ABAO, after maintenance has been done on the system. The implication of this is that after maintenance the system is left in the same state as it was before maintenance. For a system with many components a minimal repair model may be a good approximation due to the fact that maintenance is usually done on only a small fraction of the system, hence the overall state of the system should be unaffected. By saying that maintenance doesn't have an effect on the state of the system, a minimal repair model may be regarded as a pessimistic model.

In the minimal repair model one model has been used to analyze the situation namely a non-homogeneous Poisson process, NHPP, model. The idea behind how this model is developed may arise from two different views. Let us introduce both ideas behind the development of the NHPP model.

The first idea come straight from the definition of a non-homogeneous Poisson process, see section B.5 in the appendix. The NHPP is determined by the time dependent intensity or rate of occurrence of maintenance, ROCOM, function $v(t)$. If we let $v(t)$ follow the power law process where

$$v(t) = \frac{\beta}{\alpha}\left(\frac{t}{\alpha}\right)^{\beta-1}, \alpha > 0, \beta > 0 \tag{3.2.1}$$

and the cumulative rate of the process, $V(t)$, equals

$$V(t) = \int_0^t v(s)ds = \left(\frac{t}{\alpha}\right)^{\beta} \tag{3.2.2}$$

Then the NHPP model follow a Poisson process with time dependent mean, $E[N(t)] = V(t)$. Since the expected number of maintenance events is time dependent it increases if $\beta > 1$ and decreases if $0 < \beta < 1$ as time t gets larger. The probability of n number of maintenance events in the interval $[0, t)$ is then given by

$$P(N(t) = n) = \frac{\left[(t/\alpha)^{\beta}\right]^{n}}{n!} \exp\left[-(t/\alpha)^{\beta}\right] \tag{3.2.3}$$

A special property of the power law process is that the time up to first maintenance is Weibull distributed. This can be shown by considering the probability of no maintenance events in the interval up to first registered event, $[0, t_1)$.

$$P(N(t) = 0) = \exp\left[-(t_1/\alpha)^{\beta}\right] \tag{3.2.4}$$

This is exactly the survivor function for a Weibull distribution, hence the time to first maintenance is Weibull distributed with scale parameter α and shape parameter β.

The second idea makes a different approach to define the NHPP model. It has been taken into consideration mostly because it helps us in constructing the likelihood function, but also to get a clearer picture of how the NHPP model works and how it relates to the other models. Remember from last section where the RP model was defined by a standard Weibull distribution in each x_i interval. In the NHPP model we will also follow a Weibull distribution in the x_i interval, but here this is a conditional Weibull distribution which is conditioned on the time t_i it has operated so far. In figure 3.3 the shaded area represents the conditional probability of experiencing maintenance before t_i given that time is greater than t_{i-1}. Mathematically this can be expressed as

$$\begin{aligned} P(T \leq t_i | T > t_{i-1}) &= \frac{F(t_i) - F(t_{i-1})}{1 - F(t_{i-1})} \\ &= \frac{1 - R(t_i) - 1 + R(t_{i-1})}{R(t_{i-1})} \\ &= 1 - \frac{R(t_i)}{R(t_{i-1})} \end{aligned} \tag{3.2.5}$$

thus by imposing a Weibull distribution for the survivor function $R(\cdot)$ we get

$$\begin{aligned} P(T \leq t_i | T > t_{i-1}) &= 1 - \exp\left[\left(\frac{t_{i-1}}{\alpha}\right)^{\beta} - \left(\frac{t_i}{\alpha}\right)^{\beta}\right] \\ &= 1 - \exp\left[\left(\frac{t_{i-1}}{\alpha}\right)^{\beta} - \left(\frac{t_{i-1} + x_i}{\alpha}\right)^{\beta}\right] \end{aligned} \tag{3.2.6}$$

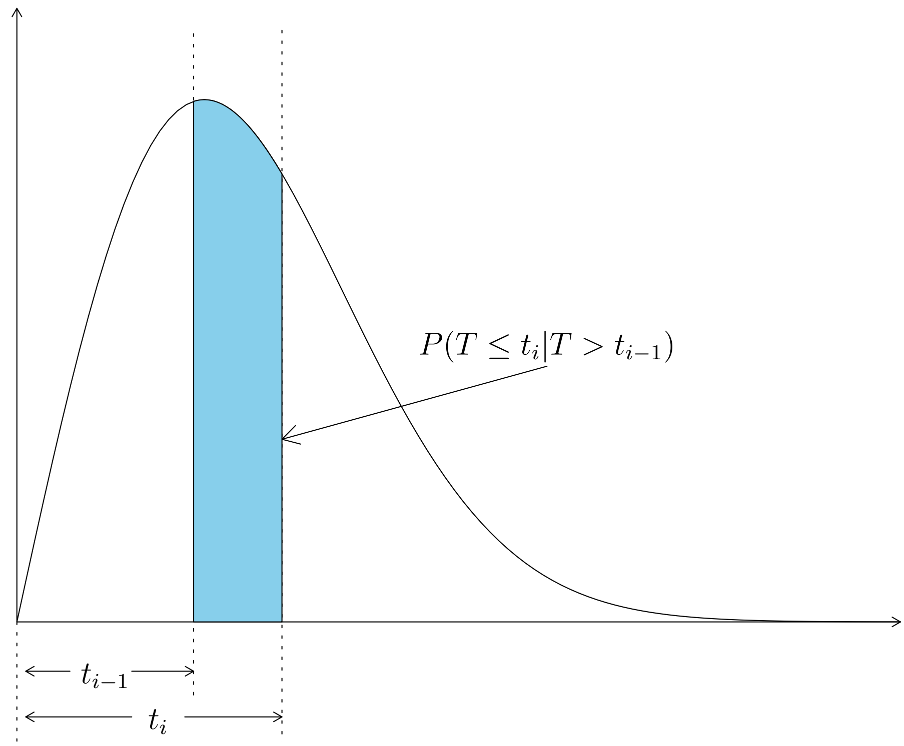


Figure 3.3: The blue area represents the conditional probability of maintenance in the interval $(t_{i-1}, t_i]$ for a NHPP model.

Comparing this to the expression for the probability of maintenance in a x_i interval for the RP model, equation 3.1.2, we clearly see there is a difference in how the probability for maintenance is evaluated for the x_i intervals. The probability of maintenance in the interval up to first maintenance is equal, but this is logical since the condition is on the time $t_0 = 0$. It is also consistent with the discussion from the first idea, since the time up to first maintenance is Weibull distributed for the power law process.

The reason for using the second idea was mainly for helping us constructing the log-likelihood function for the NHPP model, see section B.7 in the appendix for more details on constructing the likelihood. The log-likelihood for the NHPP model is given by

$$l_{NHPP}(\alpha, \beta | t) = n(\log \beta - \beta \log \alpha) + (\beta - 1) \sum_{i=1}^{n} \log(t_i) - \left(\frac{t_n}{\alpha}\right)^{\beta} \tag{3.2.7}$$

As was the case for HPP and RP models the maximum likelihood method has been used to find estimates for the parameters in the NHPP model. For this model closed form solutions are available for the MLE's of the parameters α and β:

$$\hat{\alpha} = \frac{t_n}{n^{1/\hat{\beta}}} \tag{3.2.8}$$

$$\hat{\beta} = \frac{n}{n \log(t_n) - \sum_{i=1}^{n} \log(t_i)} \tag{3.2.9}$$

A NHPP model only is different from the HPP model by having a time dependent ROCOM function, for the HPP model the ROCOM is constant over time. Thus for our NHPP model there exist an important relationship to the HPP model. Letting $\beta = 1$ the ROCOM function for the NHPP model becomes $v(t) = 1/\alpha$, which is exactly the ROCOM function for the HPP model. Therefore by letting $\beta = 1$ the NHPP model also becomes a HPP model as was also the case for the RP model. This can also be seen by letting $\beta = 1$ in the log-likelihood function.

$$l_{HPP}(\alpha|t) = -\, n \log \alpha - t_n/\alpha = l_{NHPP}(\alpha, \beta = 1|t) \tag{3.2.10}$$

Remember from last subsection that the HPP and RP model doesn't have any global trend for the expected number of maintenance events. This is not the case for the NHPP model since the ROCOM function is dependent on time and $E[N(t)] = (t/\alpha)^{\beta}$. If $\beta = 1$ $E[N(t)]$ increases linearly with time, hence no global trend. For $\beta > 1$ there will be an increasing global trend as time gets larger. While for $\beta < 1$ there will be a decreasing global trend as time gets larger. Therefore it is the value of β that indicates if a global trend occur or not.

3.3 Imperfect Repair Models

In the last two sections the minimal and perfect repair models has been described. These may be regarded as extreme cases where the repair is either AGAN or ABAO. Before fitting these models to the data we actually take a decision of the effect of the repair based on expert judgement or maybe a hunch. Systems under normal repair will usually be in a state between AGAN and ABAO after repair, hence models for estimating the repair effect is critical to make better judgement of the effect of repair.

The concept of age of a system is critical for the understanding of these models. Let us define the virtual age at time t_i as A_i. In figure 3.2 the times t_i may be regarded as the real age of the system in consideration. So if there has been a maintenance event at time 100 the real age of the system equals 100, but the virtual age is different dependent on the model we choose. For the perfect repair models which set the state of system back to AGAN the virtual age right after the maintenance event equals 0. While for the minimal repair models which leaves the system in the same state, ABAO, the virtual age equals the real age, so the virtual age is 100. As described earlier the state of the system under normal repair is somewhere between AGAN and ABAO. So for an imperfect repair model the virtual age of the system is somewhere between 0 and 100.

In the imperfect repair models its how you calculate the virtual age of the system which is the issue of finding the effect of the repair. In this thesis two different models for estimating this virtual age are proposed, namely the ARA∞ and the ARA1 model. The names are given analogous to the names given by Doyen and Gaudoin in [10]. These are both arithmetic reduction of age models where a positive function is chosen on how the virtual age is calculated.

Generally for a reduction of age model the virtual age at real age t is a positive function depending on past maintenance events at time t_i.

$$A_t = A(t; N_t, t_1, \dots, t_{N_t}) \tag{3.3.1}$$

Where N_t denotes the number of maintenance events registered up to time t and t_i the corresponding times. The principle of a reduction of age model is that repair rejuvenates the system such that its failure intensity at real age t is equal to the intensity at time A_t. Hence the failure intensity is a function of its virtual age

$$v(t) = v(A_t) \tag{3.3.2}$$

Let us now go into detail on how the two models we propose calculate the virtual age.

For the ARA1 model the maintenance effect at the i'th event only reduces the supplement of age since last maintenance event. The supplement equals the inter-occurrence time x_i at time t_i, hence the the virtual age for the ARA1 model becomes:

$$A_i = A_{i-1} + px_i, \qquad i = 1, 2, \dots \tag{3.3.3}$$

where p is the maintenance effect parameter and $A_0 = 0$. By considering the inter-occurence times x_i and using the fact that $\sum_{j=1}^{i} x_j = t_i$ we get

$$\begin{aligned}
A_1 &= 0 + px_1 \\
A_2 &= A_1 + px_2 = p(x_1 + x_2) \\
&\vdots \\
A_i &= p\sum_{j=1}^{i} x_j = pt_i = t_i - (1-p)t_i
\end{aligned} \tag{3.3.4}$$

In the ARA∞ model the maintenance effect for the i'th event reduces the virtual age of the system based on the virtual age of the system just before the maintenance event. The virtual age of the system just before the event equals the virtual age after last event plus the inter-occurence time x_i since last event, therefore the virtual age for ARA∞ is given by

$$A_i = p\left(A_{i-1} + x_i\right), \qquad i = 1, 2, \ldots \tag{3.3.5}$$

where again p is the maintenance effect parameter and $A_0 = 0$. By considering the inter-occurence times $x_i = t_i - t_{i-1}$ we have

$$\begin{aligned} A_1 &= p\left(0 + x_1\right) \\ A_2 &= p\left(A_1 + x_2\right) = p(px_1 + x_2) = pt_2 - p(1-p)t_1 \\ A_3 &= p\left(A_2 + x_3\right) = pt_3 - p(1-p)t_2 - p^2(1-p)t_1 \\ &\vdots \\ A_i &= pt_i - (1-p)\sum_{j=1}^{i-1} p^j t_{i-j} = t_i - (1-p)\sum_{j=0}^{i-1} p^j t_{i-j} \end{aligned} \tag{3.3.6}$$

There are many similarities between the two models especially the effect parameter p. The effect parameter gives us an estimate on how good the maintenance reduces the age of the system. In other words p gives us an indication on which state the systems is left in after maintenance. As mentioned earlier in section 3.1 a perfect repair model leaves the system in a state AGAN where the virtual age of the system equals 0. By letting $p = 0$ both ARA models give $A_i = 0$. Hence $p = 0$ indicates that the maintenance effect is perfect and we are left in an AGAN state. While for $p = 1$ both ARA models give $A_i = t_i$. For a minimal repair model the system is left in state ABAO where we showed earlier that the virtual age equals the real age. Thus $p = 1$ indicates that the maintenance effect is minimal and leaves the system in an ABAO state. This indicates that if the effect $p \in (0, 1)$ the state of the system will be left in a state between AGAN and ABAO, which indicates normal repair. One could argue that p will also take values that lies outside the interval $[0, 1]$. Where $p < 0$ indicates better than new repair and $p > 1$ indicates worse than old repair, but for simplicity and to get a better understanding of the relationship to the other models we have restricted ourselves to have $p \in [0, 1]$.

Another interesting property of the ARA models arises from how much maintenance history they take into account by estimating the effect parameter. Since the ARA1 model only reduces the age of the system based only on time since last event we say that the model has history of 1. On the other side the ARA∞ model reduces the age based on the total virtual age up to right before the maintenance, therefore the model take into account the entire maintenance history when estimating the effect parameter. Therefore the name ∞ of the ARA∞ model. The

ARA1 and ARA∞ models can therefore be looked upon as two extremes of the class of reduction of age models when it comes to using history when calculating the virtual age. One could also use a ARA model which only takes into account m events in history, but in our analysis these has been restricted to the ARA1 and ARA∞ models.

Constructing the log-likelihood function is an important part of the analysis when finding the estimates of the parameters in the ARA models. The construction is analogous to the second idea in the NHPP model in the last subsection, see section B.7 in the appendix for further details. The idea on how to construct the likelihood function comes from considering the probability of having an event in the interval $(A_{i-1}, A_{i-1}+x_i]$, instead of the the interval $(t_{i-1}, t_{i-1}+x_i]$ as was the case for the NHPP model. This is a conditional probability and the probability of having an event in x_i-interval becomes:

$$\begin{aligned} P(T \leq A_{i-1}+x_i | T > A_{i-1}) &= \frac{F(A_{i-1}+x_i) - F(A_{i-1})}{1 - F(A_{i-1})} \\ &= 1 - \frac{R(A_{i-1}+x_i)}{R(A_{i-1})} \end{aligned} \tag{3.3.7}$$

Assuming a Weibull distribution for the probability distributions and denoting $P(T \leq A_{i-1}+x_i | T > A_{i-1})$ as $F(x_i)$

$$F(x_i) = 1 - \exp\left[\left(\frac{A_{i-1}}{\alpha}\right)^{\beta} - \left(\frac{x_i + A_{i-1}}{\alpha}\right)^{\beta}\right] \tag{3.3.8}$$

The probability density function is found by taking the derivative of $F(x_i)$ with respect to x_i.

$$f(x_i) = \frac{\partial F(x_i)}{\partial x_i} = \frac{\beta}{\alpha}\left(\frac{x_i + A_{i-1}}{\alpha}\right)^{\beta-1} \exp\left[\left(\frac{A_{i-1}}{\alpha}\right)^{\beta} - \left(\frac{x_i + A_{i-1}}{\alpha}\right)^{\beta}\right] \tag{3.3.9}$$

One can now construct the log-likelihood function, but one need to remember a crucial part that the probability in the interval up to first event doesn't follow this conditional probability, it follows a standard Weibull distribution beacuse $A_0 = 0$. Using this the likelihood function becomes:

$$\begin{aligned} l_{ARA}(\alpha, \beta, p|t) =& \log f(x_1) + \sum_{i=2}^{n} \log f(x_i) \\ =& n(\log\beta - \beta\log\alpha) - \left(\frac{x_1}{\alpha}\right)^{\beta} + (\beta - 1)\left(\log(x_1) + \sum_{i=2}^{n}\log(x_i + A_{i-1})\right) \\ &+ \sum_{i=2}^{n}\left[\left(\frac{A_{i-1}}{\alpha}\right)^{\beta} - \left(\frac{x_i + A_{i-1}}{\alpha}\right)^{\beta}\right] \end{aligned} \tag{3.3.10}$$

The log-likelihood for the ARA∞ and ARA1 models will then be given by inserting the respective expressions for the virtual age.

Finding the MLE's for the parameters α, β and p requires us to use numerical maximization procedures in this case as no closed form solutions are available. A quasi-Newton has also been used in this case to solve the numerical maximization problem. Since there are restrictions on all three parameters, the parameters are transformed in order for us to find the MLE's. For the α and β parameters the same transformation is used as in the RP model while for p another transformation is used since p is restricted to the range $[0, 1]$.

$$\begin{aligned}\alpha' &= \log(\alpha)\\ \beta' &= \log(\beta)\\ p' &= \log\left(\frac{p}{1-p}\right)\end{aligned}$$

By doing this transformation all transformed parameters are in the range $(-\infty, \infty)$. This speeds up the procedure of finding the MLE's and by utilizing the invariance property of the MLE's, as described in section B.6 in the appendix, the MLE's for our parameters will be found.

Some important relationships to the other models are evident for the ARA model as well. As noted earlier in this section when $p = 0$ and $p = 1$ we have respectively a RP model and a NHPP model. This can be shown by considering the log-likelihood function. By letting $p = 0$ we saw that $A_i = 0$ hence the likelihood function becomes

$$\begin{aligned}l_{ARA}(\alpha, \beta, p = 0|t) &= n(\log\beta - \beta\log\alpha) + (\beta - 1)\sum_{i=1}^{n}\log(t_i - t_{i-1}) - \sum_{i=1}^{n}\left(\frac{t_i - t_{i-1}}{\alpha}\right)^{\beta}\\ &= l_{RP}(\alpha, \beta|t) \end{aligned} \tag{3.3.11}$$

In the case where $p = 1$ we have that $A_i = t_i$ and by using that $x_i = t_i - t_{i-1}$ we get

$$\begin{aligned}l_{ARA}(\alpha, \beta, p = 1|t) &= n(\log\beta - \beta\log\alpha) - \left(\frac{t_1}{\alpha}\right)^{\beta} + (\beta - 1)\left(\log(t_1) + \sum_{i=2}^{n}\log(t_i)\right)\\ &\quad + \sum_{i=2}^{n}\left[\left(\frac{t_{i-1}}{\alpha}\right)^{\beta} - \left(\frac{t_i}{\alpha}\right)^{\beta}\right]\\ &= n(\log\beta - \beta\log\alpha) + (\beta - 1)\sum_{i=1}^{n}\log(t_i) - \left(\frac{t_n}{\alpha}\right)^{\beta}\\ &= l_{NHPP}(\alpha, \beta|t) \end{aligned} \tag{3.3.12}$$

This shows that at the two extremes $p = 0$ and $p = 1$ the ARA models indeed becomes a RP model and NHPP model respectively. But, as was the case for the RP model and the NHPP model, we have one more case, namely the case where $\beta = 1$. By letting $\beta = 1$ we get:

$$l_{ARA}(\alpha, \beta = 1, p|t) = -n \log \alpha - \frac{t_n}{\alpha} = l_{HPP}(\alpha|t) \tag{3.3.13}$$

Hence the ARA models also becomes a HPP model if $\beta = 1$. One may also note that the dependence of the p parameter cease to exist if $\beta = 1$. This is not surprising as the conditional probability in the ARA models, $F(x_i)$, doesn't depend on the A_i function if $\beta = 1$.

3.4 Characteristics and Relationships Between Models

Five different models has been constructed which all follow equation 3.3.7. Thus we have a broad class of models which only differ in what probability distribution they assign to $R(\cdot)$ and how they calculate the virtual age A_{i-1}. In table 1 a summary

Model	**Virtual Age, A_i**	**Effect, p**	**State after Maintenance**	$R(\cdot)$
HPP	0	0	AGAN	exponential(α)
RP	0	0	AGAN	Weibull(α, β)
NHPP	t_i	1	ABAO	Weibull(α, β)
ARA1	$t_i - (1-p)t_i$	$\in [0, 1]$	between AGAN and ABAO	Weibull(α, β)
ARA∞	$t_i - (1-p)\sum_{j=0}^{i-1} p^j t_{i-j}$	$\in [0, 1]$	between AGAN and ABAO	Weibull(α, β)

Table 1: Main characteristics of the statistical models.

of the main characteristics of the statistical models are given. We see that the models offer us the ability to model all the different states after maintenance has been done. By having these characteristics of the models it will be interesting to see what fits the data best.

But before going into the analysis we will emphasize the relationships between the models. In figure 3.4 a relationship diagram of how the models are related is presented. We see that if $\beta = 1$ all models reduce to the HPP model. This is not surprising as this means that a exponential distribution is fitted to the x_i intervals. For the ARA models they reduce to the RP model if $p = 0$ as this sets the virtual age to zero. While for $p = 1$ the ARA models reduce to the NHPP model as the virtual age becomes t_i.

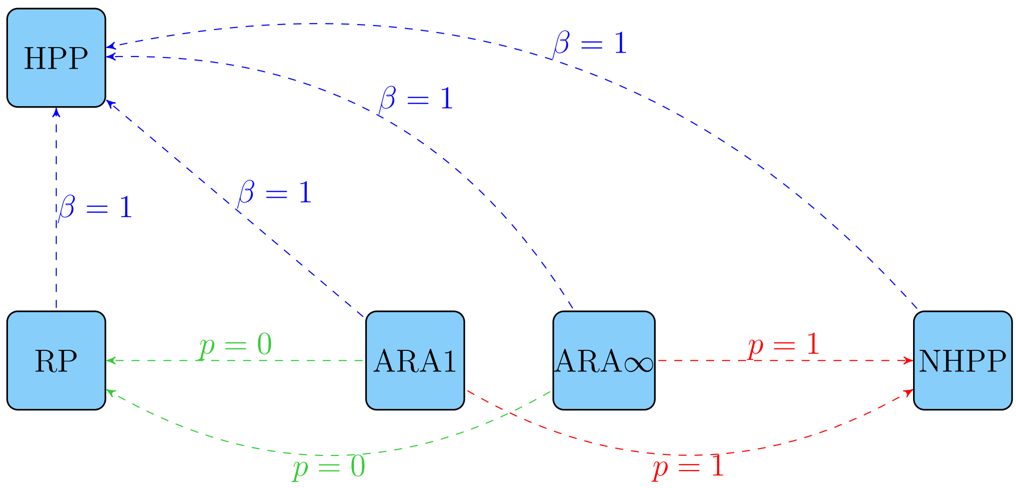


Figure 3.4: Overview of the relationship between the statistical models. From the left perfect repair models, imperfect repair models and last the minimal repair model.

3.5 Expected number of maintenance events

In the last subsections we saw that the expected number of maintenance events at time t are only given explicitly for the HPP and NHPP model. For these two it is given as

$$E_{HPP}[N(t)] = t/\alpha \tag{3.5.1}$$

$$E_{NHPP}[N(t)] = \left(\frac{t}{\alpha}\right)^{\beta} \tag{3.5.2}$$

Consequently it is easy to find the expected number of maintenance events for these models. For the RP and ARA models this is not the case, if we don't know the times maintenance happened. Two methods to find the expected number of maintenance events for these models will therefore be proposed. The first approach take into account that we know the times maintenance occurred whereas in the other we don't have this information.

3.5.1 Integrating the Failure Rate Function

The first method arises from integrating the conditional failure rate function for each x_i-interval from $t = 0$ to $t = t_n$. Given a probability density function $f(x_i)$

and survivor function $R(x_i)$ for the x_i-interval, the failure rate function is

$$z(x_i) = \frac{f(x_i)}{R(x_i)} \tag{3.5.3}$$

which gives us the probability of a failure occuring in the interval $[x, x+\Delta x)$. In our case we may think of this as the maintenance rate function since the probability of maintenance occurring in the interval is considered. Integrating the maintenance rate function from 0 to x_i the cumulative maintenance rate function, $Z(x_i)$ is found. This will represent the total number of maintenance actions in the interval $[0, x_i)$. Thus by integrating $z(x_i)$ for a specific x_i-interval the expected number of maintenance events for this interval is found.

Utilizing that each x_i-interval is disjoint and independent one can find the total number of events in the interval $t = 0$ to $t = t_n$ by integrating each maintenance rate function and summarizing.

$$\begin{aligned}
Z(t_n) &= \int_0^{t_n} \tilde{z}(s)ds \\
&= \int_0^{x_1} z_1(x_1)dx_1 + \int_0^{x_2} z_2(x_2)dx_2 + \cdots + \int_0^{x_n} z_n(x_n)dx_n \\
&= Z_1(x_1) + Z_2(x_2) + \cdots + Z_n(x_n) = \sum_{i=1}^{n} Z_i(x_i)
\end{aligned} \tag{3.5.4}$$

Where $z_i(x_i)$ denotes the conditional maintenance rate function in the i'th x_i-interval. Thus $Z(t_n) = \sum_{i=1}^{n} Z_i(x_i)$ represents the expected the number of maintenance events for a given dataset with maintenance occuring at t_i.

In general for all our models which follow a Weibull distribution the maintenance rate function is on the form:

$$z_i(x_i) = \frac{f(x_i)}{R(x_i)} = \frac{\beta}{\alpha}\left(\frac{x_i + A_{i-1}}{\alpha}\right)^{\beta-1} \tag{3.5.5}$$

where the respective calculation of the A_{i-1} function gives us the different models. Hence $Z(t_n)$ for all the respective models are found on the general form:

$$\begin{aligned}
Z(t_n) &= \int_0^{t_n} \tilde{z}(s)ds = \sum_{i=1}^{n} \int_0^{x_i} z_i(x_i)dx_i \\
&= \sum_{i=1}^{n} \int_0^{x_i} \frac{\beta}{\alpha}\left(\frac{x_i + A_{i-1}}{\alpha}\right)^{\beta-1} dx_i = \sum_{i=1}^{n} \int_{A_{i-1}}^{x_i + A_{i-1}} \frac{\beta}{\alpha}\left(\frac{u}{\alpha}\right)^{\beta-1} du \\
&= \sum_{i=1}^{n} \left[\left(\frac{x_i + A_{i-1}}{\alpha}\right)^{\beta} - \left(\frac{A_{i-1}}{\alpha}\right)^{\beta}\right] = E[N(t_n)]
\end{aligned} \tag{3.5.6}$$

	HPP	RP	NHPP	ARA1/ARA∞
$E[N(t_n)]$	$\frac{t_n}{\alpha}$	$\sum_{i=1}^{n}\left(\frac{x_i}{\alpha}\right)^\beta$	$\left(\frac{t_n}{\alpha}\right)^\beta$	$\sum_{i=1}^{n}\left[\left(\frac{x_i+A_{i-1}}{\alpha}\right)^\beta - \left(\frac{A_{i-1}}{\alpha}\right)^\beta\right]$

Table 2: Expected number of maintenance events for the different models given maintenance occurring at times t_i

By inserting the respective expressions for A_{i-1} we can now find the expected number of maintenance events for each model given that we have a dataset where we know where the maintenance times occurred.

In table 2 an overview over how the expected number of maintenance event for the different models are given. Note that there are only the RP and ARA models which need to know times of maintenance history in order to estimate $E[N(t)]$.

3.5.2 Monte Carlo Simulation

A problem with the method in the last subsection is that the times where maintenance occurred in order for us to calculate $E[N(t)]$ for the RP and ARA models need to be given. If we want to estimate the expected number of maintenance events for systems where the times when maintenance occurred is not given we therefore encounter problems. In this situation we have implemented a Monte Carlo simulation procedure to find the expected number of maintenance. Let us emphasize that this is only necessary for the RP and ARA models as explicit formulas for $E[N(t)]$ are available for NHPP and HPP. For a given model the idea behind the simulation is as follows:

- There is a period of interest $(0, t_{max})$ where one want to find the expected number of maintenance events. For the given model solve the cumulative distribution function, $F(x_i)$, for x_i in order to get a function for x_i on the form $x_i = g(F(x_i), \theta)$, where θ is the parameters in the model.
- Let m be a vector which stores the number of maintenance events generated
- Generate random numbers from the uniform distribution in the range $(0, 1)$ for $F(x_i)$ and calculate x_i
- Add the random value for x_i to the sum of the past x_i's and set $m = m + 1$.
- if $\sum x_i > t_{max}$ stop generating numbers
- if $\sum x_i < t_{max}$ generate another random number until $\sum x_i > t_{max}$

- Repeat this procedure n times. Then the expected number of events is given as m/n.

A flow chart for the simulation is given in figure 3.5.

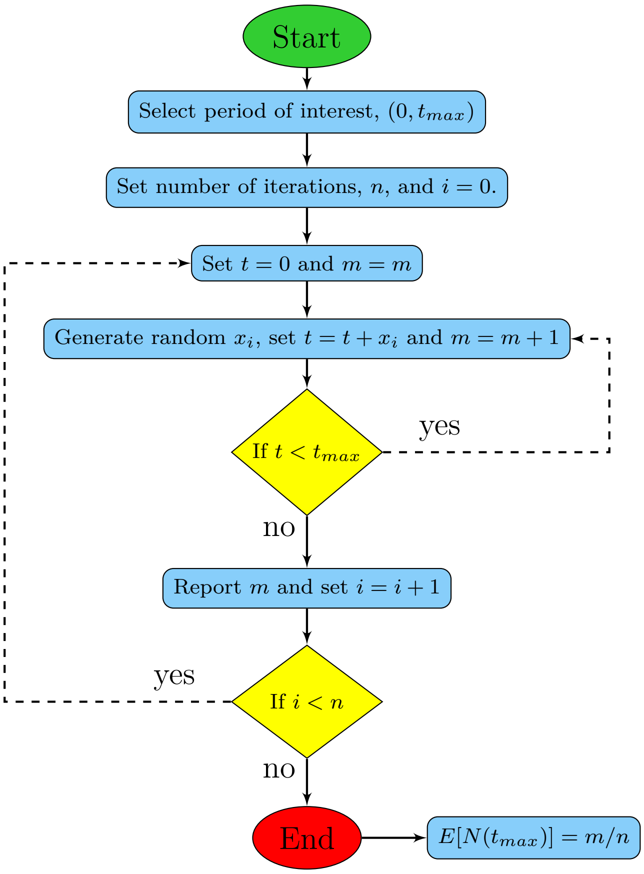


Figure 3.5: Flow chart for simulating expected number of maintenance events in the RP and ARA models.

Finding $x_i = g(F(x_i), \theta)$ is the most important part of the simulation. For both the RP and ARA models this is pretty straight forward, but it is important to note that the time up to first maintenance for the ARA models follow a standard cumulative Weibull distribution. In table 3, $x_i = g(F(x_i), \theta)$ is listed for the RP and ARA models.

Having the Monte Carlo simulation and integrating the failure rate function methods, we know have the ability to find the expected number of maintenance events for all the models in the cases where maintenance times are unknown or known.

	RP	ARA
$t \leq t_1$	$x_i = \alpha \left[-\log\left(1 - F(x_i)\right)\right]^{1/\beta}$	$x_i = \alpha \left[-\log\left(1 - F(x_i)\right)\right]^{1/\beta}$
$t > t_1$	$x_i = \alpha \left[-\log\left(1 - F(x_i)\right)\right]^{1/\beta}$	$x_i = \alpha \left[\left(\frac{A_{i-1}}{\alpha}\right)^{\beta} - \log\left(1 - F(x_i)\right)\right]^{1/\beta} - A_{i-1}$

Table 3: Inverse of the cumulative distribution function $F(x_i)$ for the RP and ARA models

This is an important part of the analysis of the data as a plot with $E[N(t)]$ against time is very informative in order for us to see how good the models fit the dataset.

Let us note that in this thesis the integrating the failure rate function method has been used for the cases where the maintenance times are available. This is actually true for all the cases in our case. The reason for proposing the Monte Carlo simulation method is mostly to emphasize the importance of on how the RP and ARA models calculate $E[N(t)]$, but also because of its importance for the cases where we don't have the maintenance times.

4 Implementation and Analysis

In this section an analysis of how the statistical models fit to the data will be given. The focus will be only on some of the systems, but nevertheless all the models could be fitted to all systems. Before starting the analysis there are some key elements we need to describe.

As mentioned in the last section the method of maximum likelihood has been used to find estimates of the parameters we wish to estimate. A short summary of the method used to maximize the log-likelihood will be given here, see in the appendix section B.6 and D, and in [6] for more details. For the HPP and NHPP models closed form solutions of the MLE's exist and therefore the estimates are easily found along with their standard deviations. For the RP and ARA models this was not the case and a quasi-Newton method has been used to find the MLE's. The quasi-Newton method depends on how you update the hessian and in our implementation we have chosen the BFGS method to update the hessian. After finding the MLE's a finite difference method to find the hessian of the log-likelihood has been used and the hessian is inverted to find the information matrix and accordingly the standard deviations.

4.1 Test Statistics

4.1.1 Likelihood Ratio Test

An important part of the analysis relies on test statistics where we are interested to see if one model is better than the other and thus is preferable. Usually we are interested in fitting a model with as few parameters as possible. As this "simple" model is usually more easy to work with. In section 3.4 it is shown that there exist many relationships between the models. Often one model is a special case of the other as was the case for $p = 0$, $p = 1$ and $\beta = 1$. When analyzing this the likelihood ratio test has been used as described in section B.6.4 in the appendix. The likelihood ratio test takes into account the difference of the log-likelihood of the "smaller" and the "full" model and relates this to a chi-square distribution. Thus by doing this we can check if one model is statistically significant different from the other.

If we consider figure 3.4 we are interested to see how the models on the bottom is statistically significant different than the other models: Let us sum up the relevant cases we have to test for the RP and ARA models first

- For the RP model we want to check if the model is significantly different than the HPP model. Thus we check if β is significantly different from 1.

- For the ARA models we have three cases to consider, $p = 0$, $p = 1$ and $\beta = 1$

- $p = 0$: We want to see if the ARA model is significantly different from the RP model
- $p = 1$: We want to see if the ARA model is significantly different from the NHPP model
- $\beta = 1$: Is the ARA model different from the HPP model. We therefore check if β is significantly different from 1.

For the RP and ARA models we therefore use the likelihood ratio test to see if these models are different than the other. For the NHPP model the case is different, we use a different test statistic. Here we are interested to check if β has to be included or not. To check this the test statistic often denoted the military handbook test or the Vuong test as described in [3, 4] has been used. The test statistic is given as:

$$W = 2\sum_{i=1}^{n} log(t_n/t_i) \tag{4.1.1}$$

Asymptotically W has a χ^2 distribution with $2n$ degrees of freedom under the null hypothesis, where the null hypothesis is

$$H_0 : \text{No trend, } \beta = 1 \tag{4.1.2}$$

Against the alternative hypothesis

$$H_1 : \text{There is a trend, either increasing or decreasing, } \beta \neq 1 \tag{4.1.3}$$

We reject H_0 for *small* or *large* values of W, where *small* values of W are indicative of a increasing trend, $\beta > 1$, while *large* values of W correspond to a decreasing trend, $\beta < 1$.

The reason why we use the military handbook test for the NHPP model is because it can be shown that this is the optimal test in this case.

4.1.2 Kolmogorov-Smirnov Test

When fitting the models to the data we are not only interested in seeing if one model could be reduced to the other one, but we are also interested to analyze the goodness of fit for each model to the data. To analyze this the Kolmogorov-Smirnov test has been used. The K-S test is a nonparametric test which can be used to test equality between two samples or comparison between a sample and a probability distribution. In our case it has been used as a test of equality between two samples. Where our two samples are the given data and the statistical model we fit. The reason we consider them as two samples is that the distribution of $P(N(t) = N)$ is not easily given for the RP and ARA models.

The null hypothesis for the K-S test is:

$$H_0 : \text{The dataset is drawn from the statistical model} \tag{4.1.4}$$

versus the alternative hypothesis

$$H_1 : \text{The dataset is not drawn from the statistical model} \tag{4.1.5}$$

Thus the K-S test can be used as a method of checking if the model is a good fit to the data. The test statistic for a two sample Kolmogorov-Smirnov test is given as.

$$D_n = \sup_t |F_{data,n}(t) - F_{model,n}(t)| \tag{4.1.6}$$

$F_{data,n}(t)$ represents the empirical cumulative distribution function for the dataset and $F_{model,n}(t)$ represent the cumulative distribution function for the statistical model. It can be shown that we reject the null hypothesis at level a if

$$\sqrt{n/2}D_n > K_a \tag{4.1.7}$$

Where K_a can be determined by $P(K \leq K_a) = 1-a$ and K follow the Kolmogorov distribution which has cumulative distribution on the form:

$$\begin{aligned} P(K \leq s) =& 1 - 2\sum_{i=1}^{\infty}(-1)^{i-1}\exp\left(-2i^2s^2\right) \\ =& \frac{\sqrt{2\pi}}{s}\sum_{i=1}^{\infty}\exp\left[-(2i-1)^2\pi^2/(8s^2)\right] \end{aligned} \tag{4.1.8}$$

Due to the complexity of the cumulative distribution of K we have used a built in method in R, $ks.test()$, to find the appropriate test statistics and consequently p-values for the Kolmogorov-Smirnov test. Therefore after finding these p-values the null hypothesis will be rejected if the p-value is less than the given significance level a. One will therefore conclude that the statistical model is a good fit if the p-value is greater than a.

4.2 Results

Since the dataset consists of 49 different systems the focus will be on only some of the systems. In table 4 the names of the systems in consideration are presented. The names are given according to a NORSOK standard, see [11], which systems on offshore oil and gas installations on the norwegian continental shelf follow. Throughout the rest of this thesis the systems will only be referred through the system number.

System	Name
21	Crude Handling
42	Chemical Injection
52	Ballast Water
58	Chemical Injection
73	Material Handling

Table 4: Name of the systems we are considering.

As said earlier in section 2 the dataset give us access to if the event was preventive maintenance or corrective maintenance. When analyzing the data we will see how the maintenance evolve when we consider both at the same time and when distinguishing between these two types of maintenance. It will be interesting to see which model fits the data best as preventive maintenance is usually conducted at predetermined interval or at some prescribed criteria, hence we should maybe expect a perfect repair model to be the best. While on the other hand the corrective maintenance is conducted after a failure has occurred, which would maybe imply that they follow a minimal or imperfect repair model.

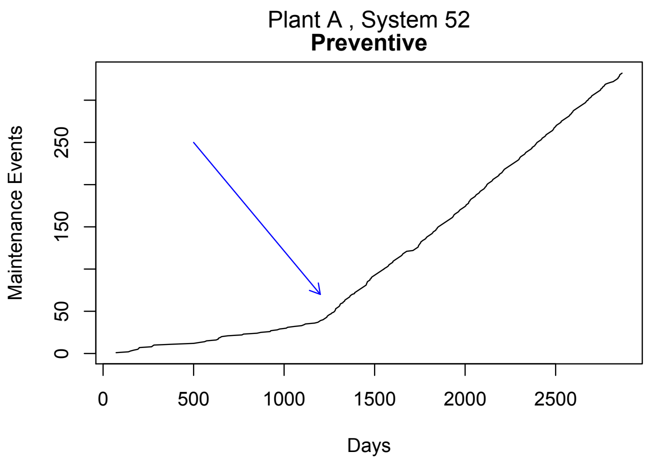


Figure 4.1: Plot of maintenance number against time for system 52

Let us look at system 52 first. Figure 4.1 presents a plot of the accumulated preventive maintenance events against time for system 52. One thing that we can easily see from this plot is that it seems like there is a change in maintenance routines after approximately $t = 1200$. After this change the slope of the graph increases, it almost looks like there is a change in how often they do preventive maintenance. This change is evident on several systems in our dataset and it seems pretty consistent that this change happened in the time period from day 1000 to 1400. This change will corrupt our results when we try to fit our statistical models as the models will not adequately manage to fit the data. Or they will give a ”wrong” impression of how the number of maintenance events will evolve over time.

After discussions with personell from Statoil ASA, who gave us access to the dataset, we understood a bit more why this change is evident. The reason behind this change is that there was a change in how they reported the maintenance they did on the systems. Roughly speaking you can say that jobs they also did before wasn't reported, but due to the change in routines they now record these jobs as well. Another factor that leads us to believe that this is the reason is that the change in routines happened in the time period we mentioned earlier. This may certainly be the reason behind this change in slope and after analyzing several systems we came to the conclusion to consider the data after the change in this slope.

Another interesting fact is that this change in slope is only evident on Plant A. For plant B we couldn't find any change in slope, so for plant B we consider the entire maintenance history we have been given access to.

The time periods we are considering will now vary from system to system, but for the systems we will analyze here the time periods are given in table 5.

	Time Period	
System	**Plant A**	**Plant B**
21	$[0, \rightarrow)$	$[0, \rightarrow)$
42	$[0, \rightarrow)$	$[0, \rightarrow)$
52	$[1168, \rightarrow)$	$[0, \rightarrow)$
58	$[985, \rightarrow)$	$[0, \rightarrow)$
73	$[0, \rightarrow)$	$[0, \rightarrow)$

Table 5: Time periods for the systems we are considering

We see that not all systems we consider have this change in slope since we consider them in the time interval $t \in [0, \rightarrow)$, but nevertheless we need to take into account the change for those systems where it is apparent. Of course optimally we are most interested in considering systems where we have the largest time interval and consequently more data. But as we saw from system 52 not many events had occurred prior to the change in slope, this is usually the case for all systems, thus we consider the time period after the change where we have most events.

In table 6 an overview over the number of events for the different categories of maintenance are given for plant A and plant B. We have chosen these systems

	Plant A			Plant B		
System	Prev	Corr	Total	Prev	Corr	Total
21	312	166	458	73	63	133
42	429	225	603	28	80	108
52	297	52	342	162	74	229
58	103	33	130	8	24	32
73	171	39	208	12	15	26

Table 6: Number of events for the systems we are considering.

mainly because there are many registered events on plant A. The reason for this is that in the next section we are going to do a simulation on how maintenance will evolve for a 30 year period on plant A. We see that preventive plus corrective doesn't always equal total events, this is due to the fact that we only consider one event per day as mentioned earlier. If for instance a preventive and corrective event happens on the same day we consider them as one event, therefore the total numbers of events may differ from preventive plus corrective.

When presenting the results from fitting the models we have chosen only to present the β parameter for the RP and NHPP models, the reasoning behind this is that this parameter determines if we have a HPP or not. For the ARA models we present both the p and β parameter as these parameters determine if we have a RP, NHPP or HPP model. For the HPP model we present the α parameter which is the only parameter in this model. Along with these parameters we also present the results from the Kolmogorov-Smirnov test for each model which describes if the dataset could be described as a sample from the respective model.

The results from the tests is described by the colored circles.

- Reject the null hypothesis H_0 on a 5% significance level
- Reject the null hypothesis H_0 on a 10% significance level
- Accept the null hypothesis H_0

The appropriate null hypothesis need to be taken into account in order to understand the colors of the circles. For the β parameters the null hypothesis is:

$$H_0 : \text{HPP is the appropriate model, } \beta = 1$$

Therefore a green or yellow circle would indicate that β is statistically significant different from 1 and a Weibull distribution of the inter-occurrence times x_i is better than a exponential distribution. Hence we shouldn't reduce the model to a HPP model.

For the K-S test the null hypothesis is:

$$H_0 : \text{The dataset is drawn from the statistical model}$$

Thus a green or yellow circle would indicate that we reject the null hypothesis and conclude that the model is a poor fit to the dataset.

The case for the p parameter is somewhat different as the circle is a colored semicircle. The left half of the circle indicates the null hypothesis:

$$H_0 : \text{State after maintenance equals AGAN, } p = 0$$

Thus a green or yellow color indicates that the state after maintenance doesn't equal AGAN. While for the right half of the circle we have the null hypothesis:

$$H_0 : \text{State after maintenance equals ABAO, } p = 1$$

Hence a green or yellow color indicates that the state after maintenance doesn't equal ABAO.

Plant A: Preventive and Corrective Maintenance

MLE's and Goodness of Fit

	HPP		**RP**		**NHPP**	
System	α	$K-S$	β	$K-S$	β	$K-S$
21	6.266	0.001	0.981	0.001	1.148	0.009
42	4.799	0.051	0.946	0.070	1.083	0.182
52	4.982	0.994	1.722	0.848	1.018	0.966
58	14.155	0.001	0.948	0.003	0.713	0.352
73	13.837	0.006	1.176	0.001	1.533	0.990

	ARA∞			**ARA1**		
System	p	β	$K-S$	p	β	$K-S$
21	0.838	0.410	0.061	0.249	1.160	0.009
42	0.518	0.616	0.364	0.001	0.836	0.023
52	0.000	1.722	0.848	0.000	1.722	0.848
58	1.000	0.713	0.352	0.229	0.655	0.534
73	1.000	1.533	0.990	0.055	1.918	0.970

Table 7: MLE's and goodness of fit for the statistical models on plant A. Here we have taken into account both preventive and corrective maintenance.

4.2.1 Plant A

In table 7, 8 and 9 the results for plant A is presented when we consider total number of, only preventive and only corrective maintenance respectively. The corresponding plots for the systems is given in section C in the appendix.

Let us consider system 21 first. In figure C.1 a plot of the expected number of maintenance events is given for system 21. From table 7 we see that it is only the ARA∞ model which gives a slightly good enough fit to the dataset as $K-S$ equals 0.061 when we consider the total number of maintenance actions. Therefore we have reason to believe that it is not a sample on a 10% significance level. Hence all the models is fairly poor fit in this case. If we consider the plot given in figure C.1 we see that the ARA∞ model is the closest to the dataset, but it overestimates the expected number of failures throughout the entire time period we are considering. Therefore we may say that none of the models are a good fit for the system 21 when we consider the total number of maintenance actions, but ARA∞ is the best out of the models. The reason for this may be due to the shape of the dataset curve since it has a sharp increase in maintenance in the middle.

When we only consider preventive maintenance, table 8, we again see that the models have problems to be a good enough fit as none of the models give a good fit. The shape of the dataset indicates here that the we had a sharp increase in preventive jobs in the middle before it flattens out again. Therefore the models will give a bad fit to the dataset. Nevertheless it is the ARA∞ model which give the best fit, even though it is not significant.

For the corrective events in table 9 the case is different. Here we see that the NHPP and ARA∞ models give a good enough fit. The ARA1 model also gives a good enough fit, but the MLE of p is at the endpoint $p = 1$ hence it equals the NHPP model. We also see that the ARA∞ model doesn't have a p-value which is significant different from 1. We will therefore conclude that the NHPP model is the best fit in this case as we are interested in fitting a model which is as simple as possible. It is interesting to note that the significant β value for the NHPP model equals 0.766 which indicates that the expected number of corrective maintenance events is decreasing.

When we consider system 42 the results in table 7 indicates that the ARA∞ and NHPP models is a good fit in this case when we consider total number of maintenance actions. The plot for system 42 in figure C.2 also indicates this as the lines for ARA∞ and NHPP are the closest to the dataset. Here the p-value of 0.518 is also significant different from both AGAN and ABAO therefore we conclude that the ARA∞ model is the best fit as it has significant parameters and highest $K-S$ value.

For the preventive maintenance in table 8 we see that the ARA∞ model is the only model which gives us a good fit.

For the corrective maintenance, table 9, the case is different as all models give a good fit. We also see that the p and β parameters in the RP, NHPP, ARA∞ and ARA1 models is not significant or only slightly significant, we therefore conclude that the HPP model is the best fit in this case as it is the simplest model and

Plant A: Preventive Maintenance

MLE's and Goodness of Fit

	HPP		**RP**		**NHPP**	
System	α	$K-S$	β	$K-S$	β	$K-S$
21	9.199	0.001	0.802	0.001	1.535	0.001
42	6.636	0.001	0.806	0.027	1.194	0.033
52	5.740	0.996	1.951	0.051	1.075	0.999
58	17.902	0.058	0.960	0.083	0.768	0.822
73	16.830	0.152	1.350	0.002	1.332	0.998

	ARA∞			**ARA1**		
System	p	β	$K-S$	p	β	$K-S$
21	0.738	0.327	0.019	0.001	0.670	0.001
42	0.316	0.543	0.184	0.001	0.668	0.005
52	0.000	1.951	0.051	0.000	1.951	0.051
58	1.000	0.768	0.822	0.345	0.742	0.822
73	0.000	1.350	0.002	0.000	1.350	0.002

Table 8: MLE's and goodness of fit for the statistical models on plant A. Here we have taken into account both preventive and corrective maintenance.

because the other models doesn't give any significant additional information.

For system 52 in table 7 when we consider the total number of maintenance actions we have a interesting case. Both the ARA models indicates a RP model thus we have only three models which fits the data, HPP, RP and NHPP. We see that all models give a good fit, figure C.3 also indicates this. Since the NHPP model doesn't have a significant β parameter we conclude that it is perfect repair models, the HPP or the RP model which gives the best fit. Interesting to note that the β parameter in the RP model is significant thus we conclude that the RP model is the best fit event though the HPP model also could have been used.

For the preventive maintenance in table 8 we again see that the ARA models indicates a RP model, but the RP model give only a slightly good fit. Both the HPP and NHPP model give a good fit and since the β parameter in the NHPP model is not significant we conclude that the HPP model is the best fit.

When we consider only corrective maintenance, table 9, all the models give a good fit. Due to the fact that we are most interested in fitting a simple model we conclude that the NHPP model is the best fit since the p parameter of the ARA1 model is not significant different from $p = 1$ and because the $K-S$ value of the NHPP model is higher than the HPP and RP model. We also see that the $\beta = 0.813$ parameter in the NHPP model indicates a decreasing trend.

When we look at system 58 considering the total number of maintenance actions, table 7, we see that both the perfect repair models doesn't give a good fit.

Plant A: Corrective Maintenance

MLE's and Goodness of Fit

	HPP		**RP**		**NHPP**	
System	α	$K-S$	β	$K-S$	β	$K-S$
21	17.229	0.034	1.002	0.034	0.766	0.424
42	12.862	0.699	0.970	0.776	0.922	0.951
52	30.750	0.125	0.958	0.196	0.813	0.574
58	50.848	0.005	0.743	0.051	0.595	0.287
73	73.359	0.001	0.578	0.155	5.497	0.752

	ARA∞			**ARA1**		
System	p	β	$K-S$	p	β	$K-S$
21	0.872	0.574	0.507	1.000	0.766	0.424
42	0.522	0.820	0.906	0.006	0.841	0.979
52	1.000	0.813	0.574	0.117	0.728	0.574
58	0.374	0.476	0.453	0.070	0.460	0.654
73	1.000	5.497	0.752	1.000	5.497	0.752

Table 9: MLE's and goodness of fit for the statistical models on plant A. Here we have taken into account both preventive and corrective maintenance.

The ARA∞ model indicates a NHPP model and the ARA1 model is not significant different from the NHPP model. Thus we conclude that the NHPP model is the best fit and $\beta = 0.713$ indicates that we have a decreasing trend.

Considering only preventive maintenance gives the same result as the perfect repair models doesn't give a good fit and we choose the NHPP model as the best fit. $\beta = 0.768$ indicates again that we have a decreasing trend.

For the corrective maintenance the ARA models give the best fit as the $K-S$ values is highest here. We also see that the p parameter is significant for both the ARA models, thus we conclude that one of the ARA models give the best fit. Looking at figure C.4 also indicates this as the ARA and NHPP models are a fairly good fit while the HPP and RP models clearly gives a poor fit.

The last system we are considering is system 73. Considering the total number of maintenance actions, table 7, we see that the NHPP and the ARA models give the best fit. Since the p parameter is significant we conclude that the ARA1 model is the best fit. Also interesting to to note that $\beta = 1.533$ in the NHPP model which indicates a clear increasing trend. Looking at figure C.5 we see that both the ARA1 and NHPP model give a good fit, but the perfect repair models clearly give a poor fit.

For preventive maintenance, table 8, we see that both the ARA models indicate a RP model, but this gives a poor fit. The HPP and NHPP model indicates a good fit, but since the $K - S$ value of the NHPP is much higher and since $\beta = 1.332$ is significant we conclude that the NHPP model is the best fit. We also see that the β parameter in the NHPP model indicates a increasing trend.

When we consider only corrective maintenance it is first interesting to look at figure C.5. We see that there are very few corrective jobs before day 1700 approximately. Both the ARA models indicate a NHPP model and since the RP model has a small $K - S$ value in comparison to the NHPP model we conclude that the NHPP is the best fit. This is also easily seen from the figure. A very high β-value, 5.497, indicates a clear increasing trend.

Plant A: Summary

System	Total	Preventive	Corrective
21	-/(ARA∞)	-	ARA∞/ARA1/ NHPP
42	ARA∞	ARA∞	ARA∞/ARA1/ HPP
52	ARA∞/ARA1/HPP/RP	HPP	ARA∞/ARA1/ NHPP
58	ARA∞/NHPP	ARA∞/NHPP	ARA∞/ARA1
73	ARA∞/ARA1/NHPP	NHPP	ARA∞/ARA1/ NHPP

Table 10: Summary of best models plant A

In table 10 a summary of which model is the best fit for the different cases and systems are given. It is interesting to note that the ARA models fits the data better in most of the cases as the RP/NHPP model is a special case of the ARA models. This is especially true for the ARA∞ model. Therefore it seems by adding the p-parameter we get a better fit to the dataset in most of the cases.

If we look at the different models for the cases where we only considered preventive or corrective maintenance. We can see that there is a difference in how the trend in maintenance increase or decrease as time goes by. By considering the β parameter in the NHPP model for corrective maintenance we see that this parameter is less than 1 for all systems except system 73. Thus we may say that for these systems the expected number of maintenance events has a decreasing trend. While for system 73 it has a strong increasing trend. Comparing this values of β for the preventive case we can see that we have a slight increasing trend for system 21, 42, and 73, no trend for system 52 and a decreasing trend for system 58. Hence in general we may say that while the preventive has a slight increasing trend we have decreasing trend for corrective on system 21 and 42. For system 52 the preventive doesn't have a trend while still the corrective has a decreasing trend. System 58 has decreasing trends on both preventive and corrective where the decrease in corrective is greater than for preventive. On the other hand we have for system 73 that both preventive and corrective has a increasing trend where the corrective has a much higher increasing trend. The implication of this is that the trends on the plant vary considerably from system to system. One would maybe expect that the preventive maintenance shouldn't have any trend as it is usually carried out at predetermined intervals, but maybe due to change in how often corrective events happen it seems like there has been a change in routines on how often they do preventive maintenance at the same time.

One interesting aspect is what state the system is left in after maintenance has been done. Let us recall that a NHPP leaves the system in an ABAO state, RP/HPP in an AGAN state and ARA somewhere in between. By considering the model which give the best fit from table 10 we see that for corrective maintenance all systems except system 42 the system is left in a ABAO state or somewhere between ABAO and AGAN. Thus an imperfect repair model or minimal model seems to be the best model. This may be consistent with how corrective jobs are usually carried out as they often are minimal/imperfect repair jobs due to the fact that the system has many components and maintenance are carried out only on a small part of the system.

For preventive maintenance it is also interesting to note that NHPP and ARA models are the best models for most of the systems except for system 52. Thus it seems in general that a NHPP or ARA model is the adequate model to use on plant A. This has some interesting implications which will become more evident after we have analyzed the results from plant B.

4.2.2 Plant B

Let us now look at the same systems on plant B. In table 11, 12 and 13 the results for plant B is given when we consider the total number of maintenance actions, only preventive and only corrective maintenance respectively. In section C in the appendix the respective plots are given.

When considering the total number maintenance actions for the different systems on plant B it is easier to interpret the results than in comparison to plant A. In table 11 we see that all the different models give a good fit to the data for all systems. For system 21, 42 and 58 we see that all the circled dots are in fact red for all the models. This will indicate that a HPP model would adequately fit the given datasets. Hence the most simple model we have is good enough to fit the data. For system 73 the HPP model is a good fit, but we have a slight tendency that the RP model is a better fit since the β-parameter is significantly different from 1 on a 10% significance level. One could therefore argue that the RP model fits the dataset better. System 52 also has the same trend as the other as all models is a good fit, but here we could argue that a NHPP better as the β parameter is again significant on a 10% significance level. Hence we should maybe fit the data using a NHPP model. Summarizing we could say that a HPP model

Plant B: Preventive and Corrective Maintenance

MLE's and Goodness of Fit

	HPP		**RP**		**NHPP**	
System	α	$K-S$	β	$K-S$	β	$K-S$
21	20.932	0.921	1.028	0.846	1.057	0.753
42	26.435	0.996	0.920	0.999	1.060	0.977
52	12.493	0.162	1.004	0.162	1.098	0.347
58	89.219	0.964	0.909	0.968	0.964	0.838
73	107.192	0.918	0.763	0.996	0.895	0.926
	ARA∞			**ARA1**		
System	p	β	$K-S$	p	β	$K-S$
21	1.000	1.057	0.753	1.000	1.057	0.753
42	0.114	0.864	0.999	0.000	0.920	0.999
52	1.000	1.098	0.347	0.041	1.163	0.410
58	0.001	0.907	0.968	0.001	0.889	0.968
73	0.002	0.750	0.996	0.001	0.750	0.996

Table 11: MLE's and goodness of fit for the statistical models on plant B. Here we have taken into account both preventive and corrective maintenance.

fit all the data, but in the case for system 52 and 73 we may instead have used a NHPP and RP model respectively.

For the preventive maintenance it as again easy to verify that the HPP model fits very good for all systems except system 52. For system 52 we see that the ARA1 model has the highest $K-S$ value and both the p and β parameter are significant. Thus we may conclude that ARA1 model is the best fit. But from figure C.8 it is hard to distinguish between the NHPP and ARA1 model thus we may say that the both models give a good fit. For the other systems we see that the HPP is a good fit in all cases, while we may argue that a RP model is better for system 42. But due to the small amount of registered events we can conclude that the HPP model is a good fit.

When we only consider the corrective maintenance the results are very easy to interpret. Here we can see in table 13 that all the circles are red. Thus no p or β parameters are significant and all the models give a good fit. Hence we conclude that the HPP model is the appropriate model to use. This is a very interesting fact as this also says that expected number of corrective maintenance events doesn't have a trend as time goes by.

In table 14 we have given a summary on which model give the best fit to the data from plant B. Following the discussion of which model is the best fit in the last paragraphs we see that the HPP is the best fit in most of the cases.

Plant B: Preventive Maintenance

MLE's and Goodness of Fit

	HPP		**RP**		**NHPP**	
System	α	$K-S$	β	$K-S$	β	$K-S$
21	38.137	0.966	0.971	0.996	1.050	0.999
42	101.964	0.541	0.565	0.944	1.444	0.763
52	17.630	0.076	1.017	0.076	1.196	0.334
58	356.875	1.000	23.079	0.980	1.262	1.000
73	232.25	0.996	1.165	0.998	1.607	0.998

	ARA∞			**ARA1**		
System	p	β	$K-S$	p	β	$K-S$
21	1.000	1.050	0.999	0.046	1.193	0.999
42	0.007	0.504	0.944	0.001	0.492	0.944
52	0.560	0.683	0.270	0.019	1.626	0.581
58	0.000	23.079	0.980	0.000	23.079	0.980
73	1.000	1.607	0.998	1.000	1.607	0.998

Table 12: MLE's and goodness of fit for the statistical models on plant B. Here we have taken into account both preventive and corrective maintenance.

This is interesting as this implies that expected number of maintenance events doesn't have a trend. Therefore we can conclude that on plant B an exponential distribution on the inter-occurrence times between maintenance events may be a good approximation. But we need to take into account that we had fewer registered events on plant B than in comparison to plant A.

Recall from the discussion of plant A we saw that a NHPP or ARA model is usually the model which gave the fit. This gave an implication on the state the system is left in after maintenance has been done. One interesting aspect on plant B is that most systems is left in a AGAN state as a HPP model is the best model in almost all cases. This also has some implications on the difference in trends for preventive and corrective maintenance events. We see here that for plant B there seems to be no trend for neither corrective nor preventive.

The difference between plant A and plant B is readily seen as the best model is usually different, but why is this the case? It seems like the effect on how old the plants are play a role. From table 6 we saw that the number of maintenance events vary considerably from plant A to plant B. Plant A has a several more registered events, while plant B has fewer. The reason for this may arise from different factors. For preventive maintenance it may arise from difference in maintenance routines or technical components which are newer on plant B and therefore require less frequent maintenance. For corrective maintenance it may also arise from the same argument as newer equipment have better design or better material which

Plant B: Corrective Maintenance

MLE's and Goodness of Fit

	HPP		RP		NHPP	
System	α	$K-S$	β	$K-S$	β	$K-S$
21	38.794	0.836	1.075	0.836	1.116	0.940
42	34.600	0.819	0.943	0.822	1.001	0.819
52	38.662	0.780	0.909	0.898	0.917	0.898
58	100.042	0.441	0.802	0.686	1.058	0.449
73	152.600	0.660	0.969	0.678	0.797	0.999

	ARA∞			ARA1		
System	p	β	$K-S$	p	β	$K-S$
21	0.958	1.184	0.990	0.056	1.196	0.940
42	0.795	1.303	0.562	0.001	0.935	0.922
52	0.001	0.907	0.898	0.004	0.836	0.970
58	0.001	0.801	0.686	0.001	0.775	0.686
73	1.000	0.797	0.999	0.428	0.758	0.999

Table 13: MLE's and goodness of fit for the statistical models on plant B. Here we have taken into account both preventive and corrective maintenance.

doesn't give that many failures. Several other aspects and reasons may also be evident, but it seems in general that the age of the plant plays a role in what kind of model fits the best and consequently what state the system is left in after maintenance.

To summarize we may say that the age of the plant plays an important role when it comes to considering what kind of model we should use when we fit it to the given data. There also seems to be evident that plant A has more ageing characteristics as a minimal or imperfect repair model is usually the best in comparison to plant B where a perfect repair model seems to be the best. An interesting aspect is that the ARA∞ model seems to be a consistent better fit to

Plant B: Summary

System	Total	Preventive	Corrective
21	HPP	HPP	HPP
42	HPP	HPP/RP	HPP
52	NHPP/HPP	NHPP/ARA1	HPP
58	HPP	HPP	HPP
73	RP/HPP	HPP	HPP

Table 14: Summary of best models plant B

the data than the other models. Therefore by using an ARA model we are more flexible when we need to fit the model and it is easier for the ”user” when fitting the model to use an ARA model as it doesn’t require any decision on which state the system is left in after maintenance. On the other hand there seem to be some cases where the ARA model give a good fit, but at the same time another models give better fit. Thus it is crucial to check all the special cases of the ARA models at the same time when fitting an ARA model. It also seems consistent that the ARA∞ model is better than the ARA1 model, which is interesting as the ARA∞ model take into account more data history.

5 Simulation of Maintenance on plant A

In the last section we analyzed some of the systems after fitting the models to the given data. In this section the results from plant A will be utilized to do a simulation on how often maintenance occur for a given system over a 30 year period for this plant. To do this simulation we have used the simulation software *ExtendSim*, 8 see [2].

The reason why we are interested in doing this simulation is to see how maintenance will evolve over a 30 year period on the plant and to use it as a input when conducting an analysis of lifetime extension on the plant. Analysis on how the maintenance will evolve over a time period may be done from several points of view. One may for instance do it as a qualitative analysis which rely on expert judgements on how maintenance usually evolve on similar systems on other plants. The purpose of this simulation is to use a more quantitative approach when analyzing how maintenance will evolve over time taking history into account. This can then be used together with the qualitative analysis to find a better picture on how maintenance will evolve.

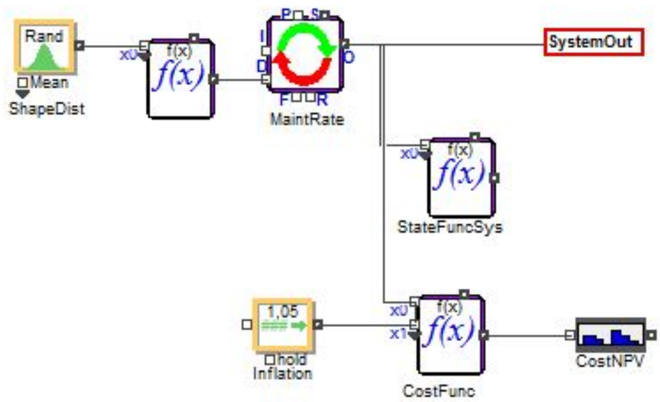


Figure 5.1: Outline of each system block in the Extend simulation

Before going more into the analysis of the simulation let us give a description on how we have done the simulation. As said earlier the simulation is done with the simulation software *ExtendSim*. The idea behind the simulation is that we take into account both preventive and corrective maintenance events, as seen from the results in table 7, and simulate how the maintenance will evolve for each system.

In figure 5.1 a picture on how we have built up each system block is given. The main block is the $MaintRate$ block. This block needs the respective values of α, β and p in order for us to calculate the time to next maintenance event. A special part of the block is that it calls a value of the β parameter based on a random number from a normal distribution with mean $\hat{\beta}$ and standard deviation $se(\hat{\beta})$. This random number is given from the $ShapeDist$ block which is only drawn in the start of each simulation. Having this we will now get a signal out from the $MaintRate$ block which indicates if maintenance has occurred or not. This signal is then used to record total number of maintenance events per year in $StateFuncSys$ and to add a cost to each event in $CostFunc$. The cost added in the $CostFunc$ block is based on a random number from a lognormal distribution with values set for the mean and standard deviation based on the system we are considering. We also adjust the cost based on the inflation rate set in $Inflation$. This cost is then sent to $CostNPV$ which records the present value of the cost per year based on a discount rate.

Let us also note that the ARA models wasn't supported in the original $MaintRate$ block, we have therefore added these features in the block in order for us to use the ARA models. This was done by recoding some of the source code in the block.

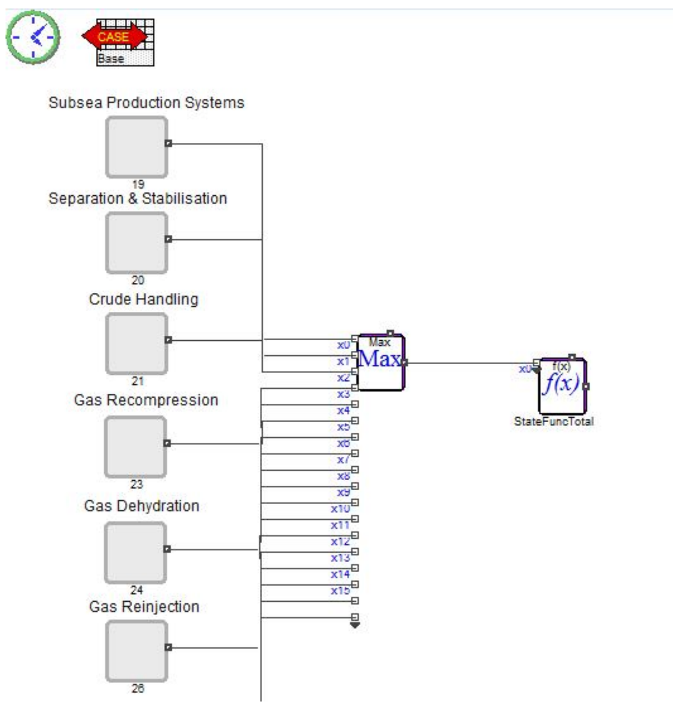


Figure 5.2: Outline of the plant in the Extend simulation

In figure 5.2 we have a description on how we have simulated the entire plant. Each system is represented by the system block on the left hand side. The signal we get from each system is sent to a Max block to check if maintenance has occurred or not. The signal from this Max block is then sent to the $StateFuncTotal$ block which record maintenance per year for the entire plant. Having set these notions on how the simulation works we now just need to run the simulation in order for us to get the results. Since we are working with days in our dataset we do the simulation for $30 \cdot 365 = 10950$ days and we do the simulation 500 times in order for us to get more accurate results.

The only thing that is left now is to find out which systems to include on the plant. As described earlier in the description of the dataset the number of registered maintenance events vary considerably from system to system. We have therefore focused on systems where we have more than 100 registered events. This gives us a total of 18 systems which we do the simulation on. The reason for this is that we have better estimates of α, β and p for these systems. We can see that these systems are represented on the left hand side of figure 5.2. We only see the top 6 here, but the rest of the systems are further down on the simulation screen.

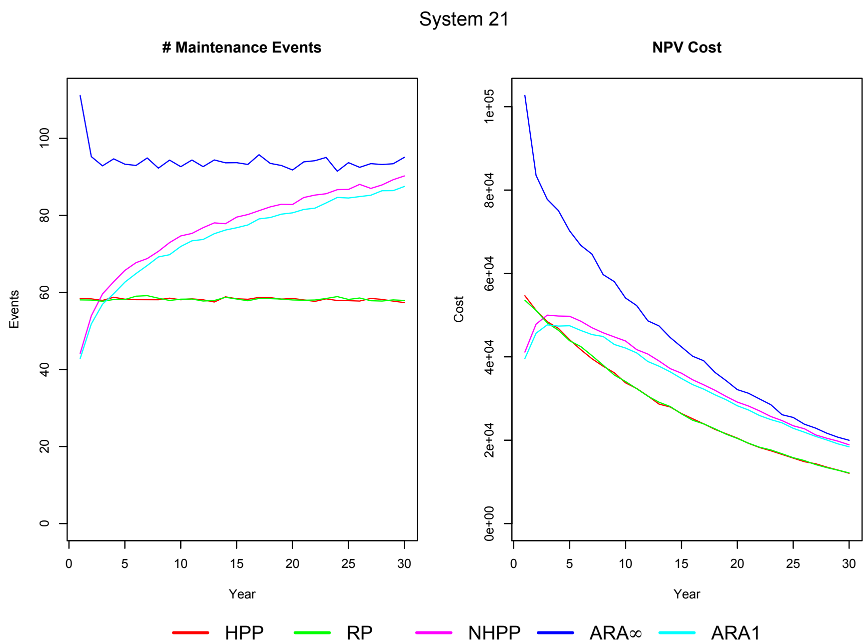


Figure 5.3: Results from the simulation for system 21

5.1 Results From the Simulation

The simulation is done for each of the models; HPP, RP, NHPP, ARA∞ and ARA1. When presenting the results the same systems considered in section 4 will be used.

Before presenting the results there are two important concepts we need to describe. The first is that when we run the simulation for each model we do that only for the given model. Thus for a HPP simulation all the systems follow the HPP model, and similarly for the RP, NHPP, ARA∞ and ARA1 model. The second concept arises from how we add the cost for each maintenance event in the $CostFunc$ block. This is based on a random number from a lognormal distribution with given mean and standard deviation. In the implementation the focus hasn't been on how to analyze the cost for each system it is only set to draw the number based on a lognormal distribution with mean 1000 and standard deviation 500. The inflation rate is set to 5% and the discount rate in the NPV block is set to 10% for each system. This gives a very weak estimate of the associated cost to each system, but the analysis has focused mostly on how the number of maintenance events evolve over time.

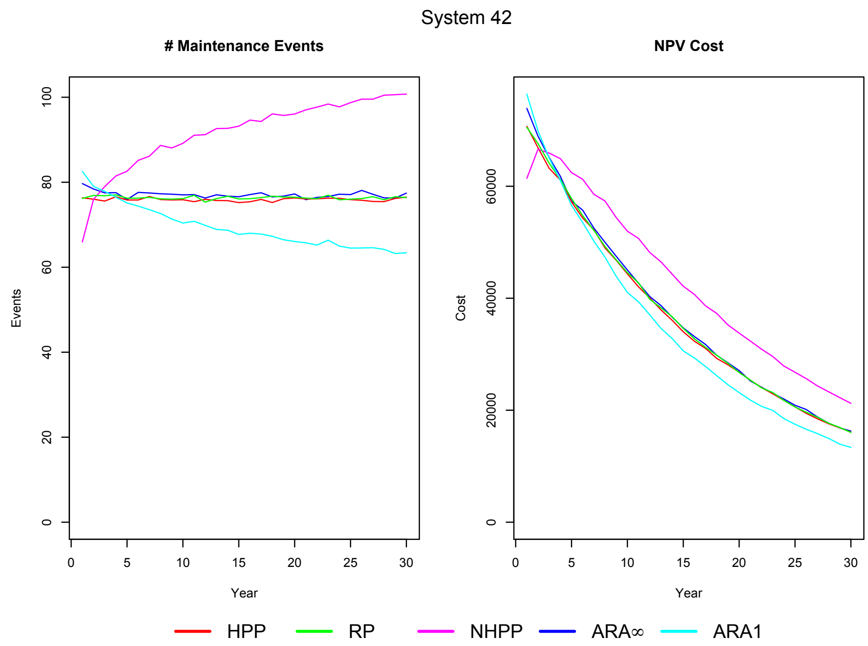


Figure 5.4: Results from the simulation for system 42

If we first consider system 21, the results are given in figure 5.3. We see that there are great differences between the models as the ARA∞ model has the highest number of maintenance events per year, while the HPP and RP has the lowest. For the NHPP an ARA1 models they seem to have a increasing trend up to the line for the ARA∞ model. The cost plot also indicates this as the cost for ARA∞ lies higher than the others.

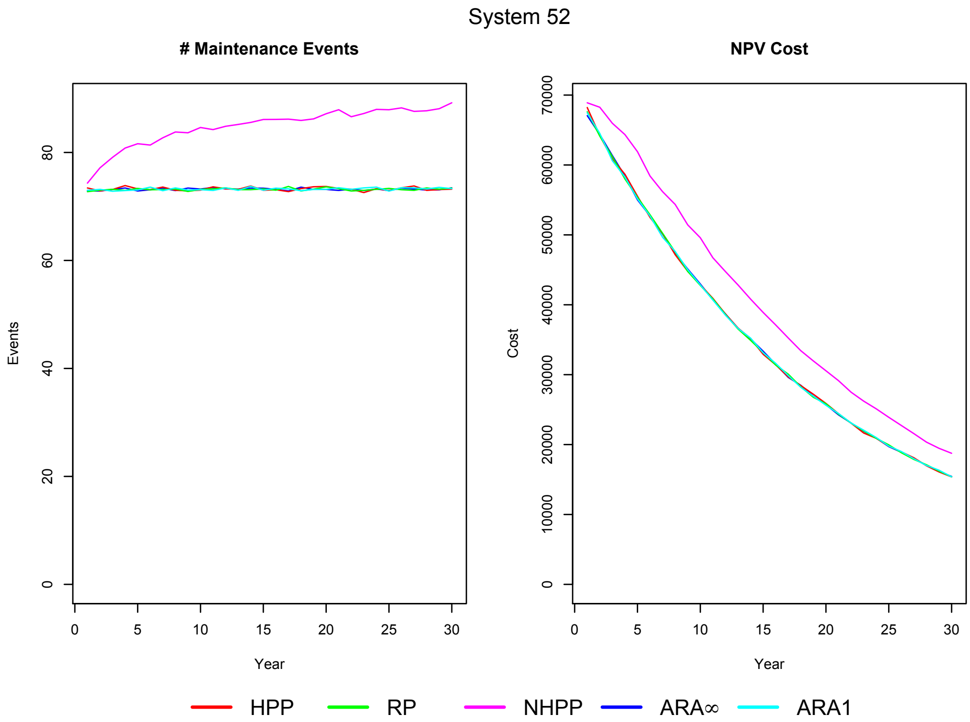


Figure 5.5: Results from the simulation for system 52

In figure 5.4 the results from the simulation on system 42 is given. We clearly see here that the NHPP model has the highest number of maintenance events per year while the ARA1 model has the lowest. The HPP, RP and ARA∞ model is fairly similar as they give almost the same results. A look at the NPV cost also indicates this as the NHPP model lies higher than the other models.

For system 52 in figure 5.5 the situation is somewhat the same. The NHPP model has the highest number of maintenance events per year, while the other models has less events. The corresponding cost plot also indicates this as the NHPP lies higher than the others.

The results for system 58 in figure 5.6 is different as the NHPP, ARA∞ and ARA1 models clearly has a decreasing tendency as time goes by. The RP and

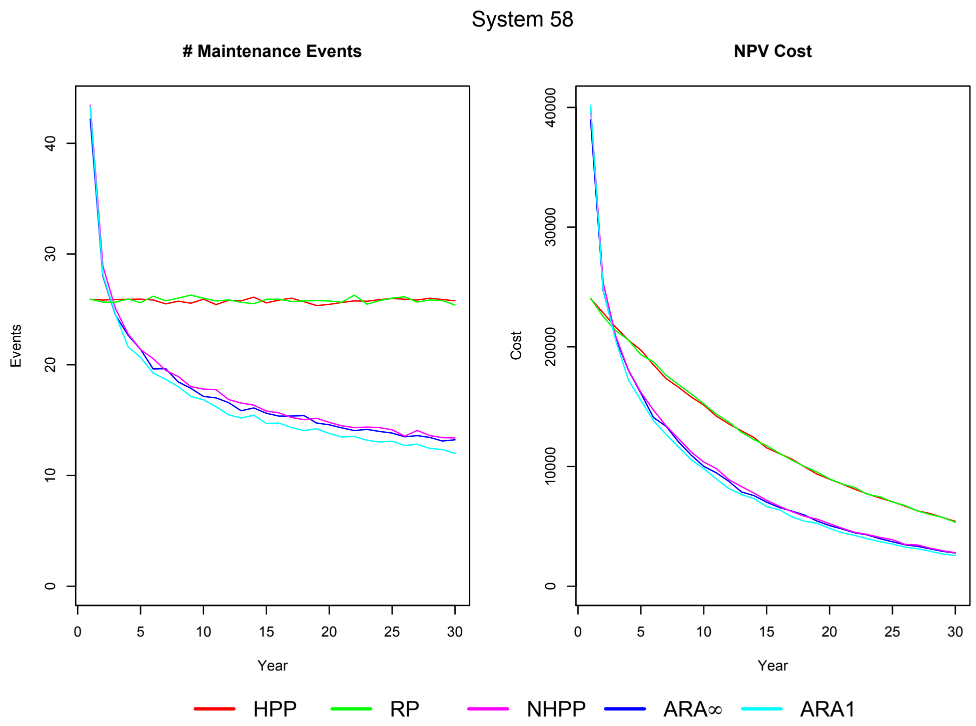


Figure 5.6: Results from the simulation for system 58

HPP give the somewhat same result which is higher than the estimate from the other models. Again the cost function indicates the same as the RP and HPP curve lies higher than the others.

System 73 in figure 5.7 gives some very different results than the others. The ARA1 model clearly has an increasing tendency as the slope of the line of maintenance events per year clearly increases. While for the ARA∞ and NHPP model they have a smaller increase, but still much higher than the RP and HPP models. The cost plot again indicates the same as the plot for maintenance events.

From the discussion of these plots we clearly see that there are big differences in how the number of maintenance events per year evolve from system to system. If recall from the last section where we discussed the model which had the best fit, see table 10. We saw that the ARA∞ model gave the best fit for all the different systems in most of the cases. Therefore it is interesting to note that the blue line should be the best based on the dataset we were given. Hence it shows that the ARA∞ model maybe should be the model to use since it gives the best results. Especially is this interesting to note for system 42 as the ARA∞ gave a clearly

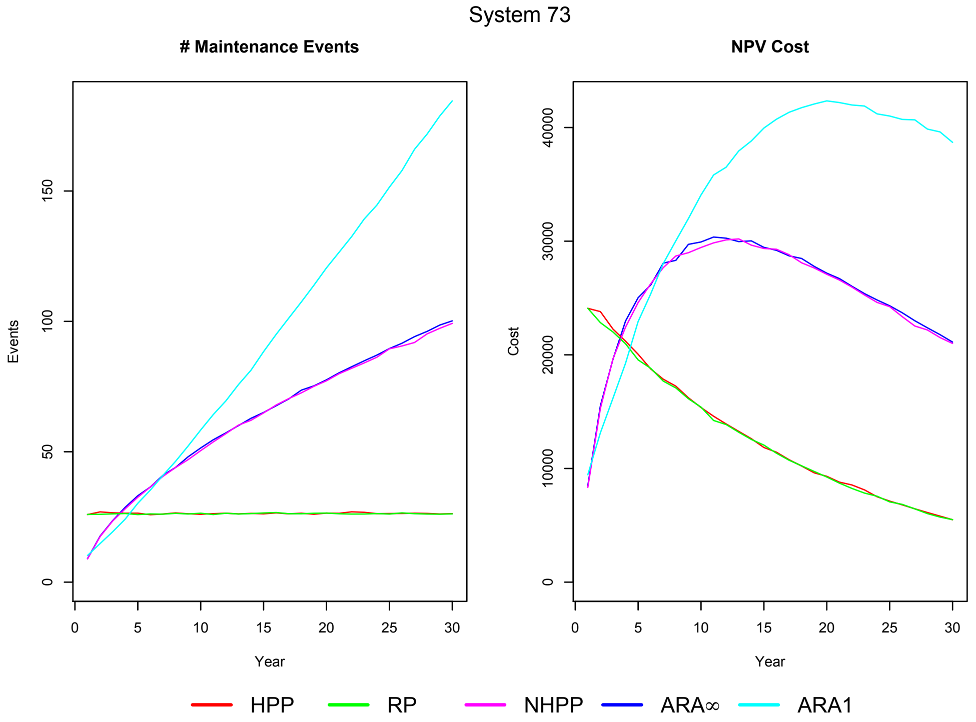


Figure 5.7: Results from the simulation for system 73

much better fit than the other models.

In figure 5.8 we have plotted the total number of maintenance events for the entire plant based on the results from the different models. Note that there is a big difference between the NHPP and ARA1 model to the ARA∞, RP and HPP models. We clearly see that the ARA∞ model gives an estimate which lies between the other models. It also seems like that the ARA∞ model maybe gives us the best out of both the minimal and perfect repair models. Hence by imposing a ARA∞ model we maybe get a better model which can describe more situations.

An interesting part of this analysis is also that the cost function follow the same tendency as the maintenance function. It is not surprising, but this gives us a better understanding that if we have good models to analyze how maintenance will evolve over time it will give us a better picture on the corresponding costs. Of course you should give a more thorough and comprehensive report of how the cost will evolve based on different scenarios, but this cost function will on the bottom line be highly influenced by how the maintenance will evolve over time.

A problem when simulating the number of events over a 30 year period in this case is that we have used a limited amount of data to calculate how the maintenance will evolve based on our dataset. Therefore these estimates of how maintenance will evolve come with a rather high degree of uncertainty, but by using these results along with experts judgements we will get a better picture on how the maintenance will evolve.

Of course there are some serious limitations of this simulation as well. If for instance we know that 10 year ahead we need to replace an entire system there will of course be a lot more registered events at this time. This simulation would not be able to take into account this ”peak”, therefore we will emphasize that this analysis should be used as a decision support together with a qualitative analysis in order for us to get the total picture.

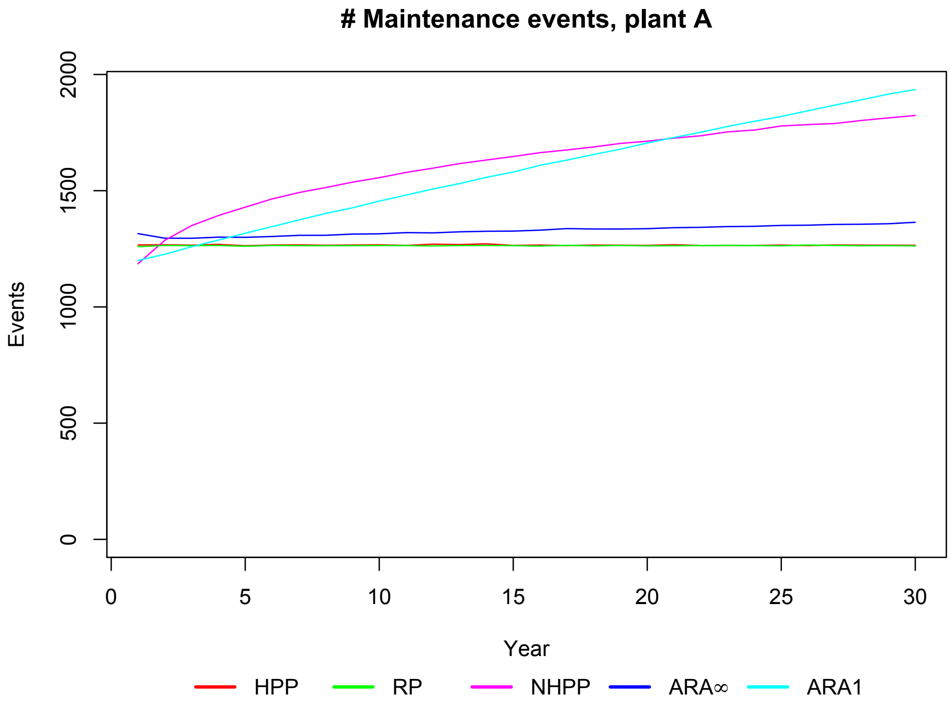


Figure 5.8: Total number of maintenance events from the simulation.

6 Conclusion

The analysis of how good different models fit to the given data are of crucial importance as the result of how the maintenance will evolve highly rely on what kind of model one use. In this thesis a general class of statistical models has been used which all follow the following expression for the probability of an event occuring in a x_i interval.

$$F(x_i) = 1 - \frac{R(A_{i-1} + x_i)}{R(A_{i-1})} \tag{6.0.1}$$

The differences between the models arises from what probability distribution one assign to $R(\cdot)$ and how the virtual age A_i defined/calculated. We have focused on exponential and Weibull distribution of the probability distributions which has some well known properties and corresponding statistical models if we consider perfect and minimal repair (HPP,RP and NHPP). And we have also analyzed some more interesting cases where more sophisticated techniques to describe the virtual age has been used, namely ARA∞ and ARA1. The results in section 4 and 5 has shown that especially the ARA∞ has proven itself to be a good model to use in most of the cases as it has given a good fit for the given dataset.

We would like to emphasize that the ARA∞ model gives the "user" the ability to not take any decision about what state the system is left in after maintenance before the analysis and therefore becomes a more flexible model than the minimal and perfect repair models. From the results of the ARA∞ model one would also get a better picture on what state the system is left in after maintenance and consequently what kind of model should be used for the data. In other words one can say with more certainty that the ARA∞ model will give the correct result of how the maintenance will evolve.

Of course the results from this analysis must also be taken with a degree of uncertainty as we are not be able to model how the maintenance on the systems will evolve perfectly. All the results are based on data which again come with some uncertainty as there are people who do the maintenance that are responsible for giving correct input of when they did the maintenance. Also some maintenance jobs are of much greater importance than others and this hasn't been taken into account as we have only focused on the number of maintenance events. This can be a serious limitation of how good the model fit the reality.

But from this analysis we can conclude that there seem to exist great differences with respect to ageing and how good the maintenance is between plant A and B. In general it seems like the maintenance on plant B follow a perfect repair model, while on plant A an imperfect or minimal repair model seems to be more accurate.

The results from the simulation of how the maintenance evolve over a 30 year period has also given us a better fundament when describing how the maintenance will evolve. We would like to emphasize that this quantitative analysis together with a qualitative analysis will give us a better foundation when describing how the maintenance will evolve.

The relationship between how the maintenance evolve and the corresponding cost gives us a better understanding that good models for how the maintenance evolve is crucial in order to get better estimates for the cost. As noted earlier this thesis hasn't focused on the cost, but the results from this thesis can be used as a fundament when implementing better and more accurate results for the cost. This can for instance be interesting to use when we want to analyze a lifetime extension of a plant.

At last we would like to highlight that in this analysis we have focused on the exponential and Weibull distribution, but there is no restriction to use any other probability distribution. Therefore equation 6.0.1 together with simulation of the expected number of events, section 3.5, gives a very general class of models which can be utilized in many sitiutations.

References

[1] *R: A language and environment for statistical computing*
R Development Core Team (2011), R Foundation for Statistical Computing, Vienna, Austria, http://www.R-project.org/

[2] *ExtendSim Professional Simulation Tools, Version 8*
Imagine That, Inc., San Jose, California, 2010, http://www.extendsim.com/

[3] *System reliability theory: models, statistical methods, and applications*
M. Rausand and A. Høyland. John Wiley & Sons, Inc., New Jersey, 2004

[4] *Repairable systems*
M.J. Crowder, A.C. Kimber, R.L. Smith and T.J. Sweeting. Chapter 8 in Statistical analysis of reliability data, Chapman & Hall/CRC, 1991

[5] *Statistical Inference*
G. Casella and R.L. Berger, Duxbury, 2004

[6] *Numerical Optimization*
J. Nocedal and S.J. Wright, Springer series in optimization, 2006

[7] *Introduction to Mathematical Statistics*
R.V. Hogg, J.W. McKean and A.T. Craig, Pearson Education, Inc., 2005

[8] *Introduction to Probability Models*
S.M. Ross, Elsevier Inc., 2007

[9] *Generalized renewal process for analysis of repairable system with limited failure experience*
M. Yaï£ · es, F. Joglar and M. Modarres, Reliability engineering & system safety vol 77 p. 167-180, Elsevier Inc., 2002

[10] *Classes of imperfect repair models based on reduction of failure intensity or virtual age*
L. Doyen, O. Gaudoin, Reliability engineering & system safety vol 84 p. 45-56, Elsevier Inc., 2004

[11] *NORSOK standard Z-DP-002 coding system (Rev.3, Oct. 1996)*
http://www.standard.no/no/Fagomrader/Petroleum/NORSOK-Standard-Categories/Z-Technical-Info/

[12] *The image on the front page is used with approval from Det Norske Veritas AS*

A Definitions

Maintenance: The combinations of all technical and corresponding administrative actions, including supervision actions, intended to retain an entity in, or restore it to, a state in which it can perform its required function [IEC 50]

Preventive maintenance: The maintenance carried out at predetermined intervals or corresponding to prescribed cirteria and intended to reduce the probability of failure or the performance degradation of an item [BS 4778]

Corrective maintenance: The maintenance carried out after a failure has occurred and intended to restore an item to a state which it can perform its required function [BS 4778]

Failure: The termination of its ability to perform a required function [BS 4778]

System: Set of elements which interact according to a design, where an element of a system can be another system, called a subsystem, which may be a controlling system or a controlled system and may include hardware, software and human interaction [IEC 61508, Part 4]

B Theory

In this section some of the theory used in the report are described in more detail. If otherwise not stated the theory in this section is based on [3, 4, 5, 7, 8, 9].

B.1 Exponential Distribution

The exponential distribution is a continuous probability distribution with probability density function,

$$f(x) = \frac{1}{\alpha}\exp\left[-x/\alpha\right], x > 0 \tag{B.1.1}$$

survival function

$$R(x) = 1 - F(x) = \exp\left[-x/\alpha\right], x > 0 \tag{B.1.2}$$

and failure rate function

$$z(x) = \frac{1}{\alpha} \tag{B.1.3}$$

Where $\alpha > 0$ is a scale parameter. The mean of the exponential distribution is $\mathrm{E}(x) = \alpha$ and the variance equals $\mathrm{Var}(x) = \alpha^2$.

B.2 Weibull distribution

The Weibull distribution is a continuous probability distribution with probability density function,

$$f(x) = \frac{\beta}{\alpha}\left(\frac{x}{\alpha}\right)^{\beta-1} \exp\left[-(x/\alpha)^{\beta}\right], x > 0 \tag{B.2.1}$$

survival function

$$R(x) = 1 - F(x) = \exp\left[-(x/\alpha)^{\beta}\right], x > 0 \tag{B.2.2}$$

and failure rate function

$$z(x) = \frac{\beta}{\alpha}\left(\frac{x}{\alpha}\right)^{\beta-1} \tag{B.2.3}$$

Where $\beta > 0$ is the shape parameter and $\alpha > 0$ is the scale parameter of the distribution. The mean of the Weibull distribution is $\mathrm{E}(x) = \alpha\Gamma(1 + 1/\beta)$ and the variance equals $\mathrm{Var}(x) = \alpha^2\Gamma(1 + 1/\beta) - \mathrm{E}^2(x)$. Where $\Gamma(y)$ depicts the gamma function which is given as, $\Gamma(y) = \int_0^{\infty} t^{y-1} \exp^{-t} dt$

Some interesting properties of the Weibull distribution follows by considering diferent values of the shape parameter. If $\beta = 1$ the failure rate is not dependent on time and the probability distribution is the same as the exponential distribution. If $\beta < 1$ the failure rate decreases over time and if $\beta > 1$ the failure rate increases over time.

B.3 Homogeneous Poisson Process

A counting process $(N(t), t \geq 0)$ is homogeneous Poisson process, HPP, with rate $1/\alpha$, for $\alpha \geq 0$, if $N(0) = 0$, and the inter-occurrence times $x_1, x_2, \cdots$ are independent and exponentially distributed with scale parameter α.

Because of the subsequent definition of a HPP some properties follows:

1. The rate of occurrence of maintenance, ROCOM, of the HPP is constant and independent of time,

$$v(t) = 1/\alpha, \text{ for all } t \geq 0$$

2. The number of maintenance actions in the interval $(t, t + x]$ is Poisson distributed with mean x/α,

$$P(N(t + x) - N(t) = n) = \tfrac{(x/\alpha)^n}{n!} \exp\left[-x/\alpha\right], \text{ for all } t \geq 0, x \geq 0$$

3. The expected number of maintenance actions and variance at time t equals

$$\mathrm{E}[N(t)] = \mathrm{Var}[N(t)] = t/\alpha$$

4. The inter-occurrence times $x_1, x_2, \cdots$ are independent and identically distributed exponential random variables having mean α

B.4 Renewal Process

A renewal process, RP, is closely linked to the HPP. In the HPP model the inter-occurrence times $x_1, x_2, \cdots$ are exponentially distributed with scale parameter α. While for the RP the inter-occurrence times are independent and identically distributed with distribution function

$$F_X(x) = P(X_i \leq x) \text{ for } x \geq 0, i = 1, 2, \cdots \tag{B.4.1}$$

Thus a renewal process may be thought of as a generalization of the HPP where the underlying distribution of the inter-occurrence times may take any probability distribution other than exponential.

Expected number of maintenance actions in a time interval $[0, t]$ is given by the fundamental renewal equation

$$E[N(t)] = V(t) = F_X(t) + \int_0^t V(t-s) dF_X(s) \tag{B.4.2}$$

The corresponding renewal density, $v(t)$, is found by taking the derivative of the renewal equation on both sides with respect to t

$$v(t) = f_X(t) + \int_0^t v(t-s) f_X(s) ds \tag{B.4.3}$$

Finding closed form solutions of the renewal density may be hard or even impossible to find for most probability distributions. Thus estimation of the expected number of maintenance actions for a given time t rely on simulation of the process.

Due to the problem of finding the exact expression for $E[N(t)]$ some approximations formulas come in to play. The elementary renewal theorem states

$$\lim_{t \to \infty} \frac{E[N(t)]}{t} = \frac{1}{\mu} \tag{B.4.4}$$

where μ equals the average length of each renewal interval or simply the expectation of $F_X(t)$ in the renewal intervals.

B.5 Non-Homogeneous Poisson Process

A counting process $(N(t), t \geq 0)$ is a non-homogeneous Poisson process, NHPP, with time dependent intensity $v(t)$ for $t \geq 0$ if

1. $N(0) = 0$
2. Non-overlapping increments are independent
3. $P(N(t+\Delta t) - N(t) = 1) = v(t)\Delta t + o(t)$
4. $P(N(t+\Delta t) - N(t) \geq 2) = o(t)$

for all t and where $\frac{o(t)}{\Delta t} \to 0$ as $\Delta t \to 0$. The main parameter in a NHPP model is the rate parameter $v(t)$ which is in our case denoted as the rate of occurence of maintenance, ROCOM. The cumulative rate of the process is

$$V(t) = \int_0^t v(s)ds \tag{B.5.1}$$

And thus the number of failures in an interval $(0, t]$ is Poisson distributed

$$P(N(t) = n) = \frac{[V(t)]^n}{n!} \exp\left[-V(t)\right] \tag{B.5.2}$$

with mean and variance

$$E\left[N(t)\right] = Var\left[N(t)\right] = V(t) \tag{B.5.3}$$

The NHPP can be used to model if a system has increasing or decreasing inter-occurrence times, where the rate $v(t)$ will respectively be increasing or decreasing. Because of the assumption of independent increments, the number of events in the interval $(t_1, t_2]$, will be independent of the number of events and inter-occurrence times before t_1. Hence the ROCOM given the history up to time t, $\mathcal{H}_t$, will be unaffected of the history and only depend on the time interval we are considering.

$$v(t|\mathcal{H}_t) = v(t) \tag{B.5.4}$$

Because of this assumption a NHPP has often been termed a minimal repair model since the ROCOM doesn't depend on the history. In our case where we are considering a system with many components, the maintenance events described is usually done on only a small fraction of the components in the system and thus the system will almost be in the same state as before the event. Minimal repair is therefore usually a realistic approximation for systems with many components.

B.6 Maximum likelihood

Given data at times $t_1, t_2, \cdots, t_n$ and one want to fit a probability distribution $f(t|\theta)$ to the data. The method of maximum likelihood is one of the most popular methods for deriving estimators for the unknown parameters in probability distribution when fitting it to known data. The basic idea behind the maximum likelihood method is that one are interested in finding estimators of the unknown parameters θ that maximize the likelihood function. This can be thought of a way of finding the estimators of the parameters that have the highest probability to fit the data. Its widespread use is because it has some desirable asymptotic properties which in most cases lead to good estimators of the unknown parameters. These estimators are referred to as maximum likelihood estimators, MLE's. Some of these asymptotic properties will be described here and then later go on to construct the likelihood functions for the statistical models used in this thesis in the next section.

The likelihood function for a probability distribution $f(t|\theta)$ where θ is the parameters one want to estimate is given as

$$L(\theta|t) = \prod_{i=1}^{n} f(t_i|\theta) \tag{B.6.1}$$

Using the corresponding log-likelihood may be as it is easier to work with in many situations. The log-likelihood is given as

$$l(\theta|t) = \sum_{i=1}^{n} \log f(t_i|\theta) \tag{B.6.2}$$

The MLE's is found by maximizing the likelihood or log-likelihood with respect to the parameters θ.

Finding the MLE's may in many cases be a tedious task, as closed form solution of the MLE's may not be possible to find. Working with the log-likelihood, possible candidates for the MLE of $\theta_1, \theta_2, \cdots, \theta_k$ are the ones who solve

$$\frac{\partial}{\partial \theta_i} l(\theta|t) = 0, \qquad i = 1, 2, \cdots, k \tag{B.6.3}$$

If this leads to a closed form solution for the MLE's possible candidates are found. Often a closed form solution is not attainable and thus the maximum likelihood method relies highly on numerical maximization procedures in those cases. In our work we have implemented a quasi-Newton method, as described in [6], which maximizes the likelihood in those cases where a closed form solution is not attainable. Other numerical maximization procedures may also be used, but due to desirable convergence properties of the quasi-Newton method this is often chosen to solve the problem.

An important property of maximum likelihood estimates is the invariance property

> If $\hat{\theta}$ is the MLE of θ, then for any function $g(\theta)$ the MLE of $g(\theta)$ is $g(\hat{\theta})$

A proof of the invariance property can be found in [5]. This property if often very useful when finding the MLE's in the situations where a numerical maximization procedure is needed to solve the problem.

B.6.1 Conditions

Some conditions come in to play when finding the maximum of the log-likelihood function which need to be fulfilled in order to make sure that the MLE's give a local maximum.

- The first order partial derivatives are zero, equation B.6.3.
- At least one second order partial derivative is negative
- The determinant of the second order partial derivatives are positive

If all these assumptions are satisfied a local maximum has been found, but in order to make sure that this is the global maximum the endpoints of the parameters we wish to estimate also has to be checked.

B.6.2 Asymptotic Properties

As mentioned earlier the maximum likelihood method has some desirable properties. Denoting the MLE's as $\hat{\theta}$ and the true value of the parameters as θ_0 for the given dataset with n observed times, it can be shown that as n goes to infinity the MLE's converges in probability to its true value

$$\text{as } n \to \infty \text{ then } \hat{\theta} \xrightarrow{p} \theta_0 \tag{B.6.4}$$

This consistency property is very desirable when a large enough dataset is available as the MLE's will give a very good approximation of the true value.

Another desirable property which is very useful when it comes to analyzing the MLE's is that is asymptotically has a normal distribution when n goes to infinity

$$\sqrt{n}(\hat{\theta} - \theta_0) \xrightarrow{d} N(0, I^{-1}) \tag{B.6.5}$$

Where I^{-1} is the inverse of the information matrix and is often denoted the Cramér-Rao lower bound for an unbiased estimator. Thus for the MLE's the variance will attain the Cramér-Rao lower bound if n is large enough. Hence the Cramér-Rao lower bound can be used as an estimate for the variance of the MLE's.

B.6.3 Information Matrix and Variances

The information matrix plays an important role when analyzing the MLE's as it can be used to find the variance and hence the standard deviation of the MLE's. In the case where k parameters needs to be estimated the information matrix becomes

$$I(\theta) = -\begin{bmatrix} \frac{\partial^2 l(\theta|t)}{\partial\theta_1^2} & \frac{\partial^2 l(\theta|t)}{\partial\theta_1\partial\theta_2} & \cdots & \frac{\partial^2 l(\theta|t)}{\partial\theta_1\partial\theta_k} \\ \frac{\partial^2 l(\theta|t)}{\partial\theta_2\partial\theta_1} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ \frac{\partial^2 l(\theta|t)}{\partial\theta_k\partial\theta_1} & \cdots & \cdots & \frac{\partial^2 l(\theta|t)}{\partial\theta_k^2} \end{bmatrix} \tag{B.6.6}$$

As described earlier the Cramér-Rao lower bound is given as the inverse of the information matrix. Thus by inverting the information matrix the variance and the standard deviation is found where the standard deviation is given as

$$se(\hat{\theta}) = \sqrt{diag(I^{-1})} \tag{B.6.7}$$

Where the square root is taken on the diagonal elements in the inverted information matrix.

B.6.4 Likelihood Ratio Test

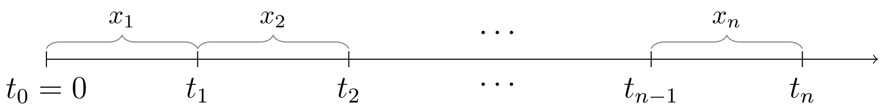
An important test when analyzing nested models is the likelihood ratio test. Assume two nested models Y and Z with k and r parameters to estimate respectively, has been fitted and where $r > k$. Then the likelihood ratio test is given as

$$\begin{aligned} V &= -2\left[\log(L_Y(\theta|t) - \log(L_Z(\theta|t))\right] \\ &= -2\left[l_Y(\theta|t) - l_Z(\theta|t)\right] \end{aligned} \tag{B.6.8}$$

It can be shown that this approximately has a chi-square distribution with $r - k$ degrees of freedom. Thus given a significance level a the model with fewest parameters, Y, will be rejected if $V > \chi^2_{a,r-k}$

B.7 Construction of the Likelihood Function for the Statistical Models

For all the different statistical models used in this thesis the maximum likelihood method has been used to estimate the unknown parameters θ in the models. One crucial part of the analysis is the construction of the likelihood function. Given data with events registered at times $t_1, t_2, \cdots, t_n$ and corresponding inter-occurence times $x_i = t_i - t_{i-1}$ one has a timeline as illustrated below.



Let us derive the likelihood function for the different models one at a time.

B.7.1 Homogeneous Poisson Process

In the HPP model an exponential distribution, exponential(α), is assumed for the inter-occurrence times. Thus the log-likelihood is constructed as follows

$$\begin{aligned} l(\alpha|x) =& \sum_{i=1}^{n} \log f(x_i) = -n \log \alpha - \sum_{i=1}^{n} \frac{x_i}{\alpha} \\ =& -n \log \alpha - \frac{t_n}{\alpha} \end{aligned} \tag{B.7.1}$$

Where $\sum_{i=1}^{n} x_i = t_n$ has been used. Because of the simplicity of the log-likelihood function the MLE of α, $\hat{\alpha}$, can be easily derived.

$$\begin{aligned} \frac{\partial l(\alpha|t)}{\partial \alpha} =& 0 \\ \Rightarrow \hat{\alpha} =& \frac{t_n}{n} \end{aligned} \tag{B.7.2}$$

the variance and standard deviation of the MLE is in this case easily found by using the Cramér-Rao lower bound.

$$\begin{aligned} \mathrm{Var}(\hat{\alpha}) =& 1/\frac{\partial^2 l(\alpha|t)}{\partial \alpha^2} = \frac{t_n^2}{n^3} \\ \Rightarrow se(\hat{\alpha}) =& \frac{t_n}{n^{3/2}} \end{aligned} \tag{B.7.3}$$

B.7.2 Renewal Process

For the RP model a Weibull distribution, Weibull(α, β), is assumed for the inter-occurence times. Thus the log-likelihood becomes.

$$\begin{aligned} l(\alpha, \beta|x) =& \sum_{i=1}^{n} \log f(x_i) \\ =& n(\log \beta - \beta \log \alpha) + (\beta - 1) \sum_{i=1}^{n} \log(x_i) - \sum_{i=1}^{n} \left(\frac{x_i}{\alpha}\right)^{\beta} \end{aligned} \tag{B.7.4}$$

A closed form solution is not available thus finding estimates of the MLE's of α and β relies on numerical maximization procedures.

B.7.3 Non-Homogeneous Poisson Process

In the NHPP model a different approach to address the problem of finding the likelihood function has been used. Let us consider the interval $(t_{i-1}, t_i]$ and look at the conditional probability of an event occuring in this interval. In figure B.1 the shaded area represents the conditional probability we are interested in.

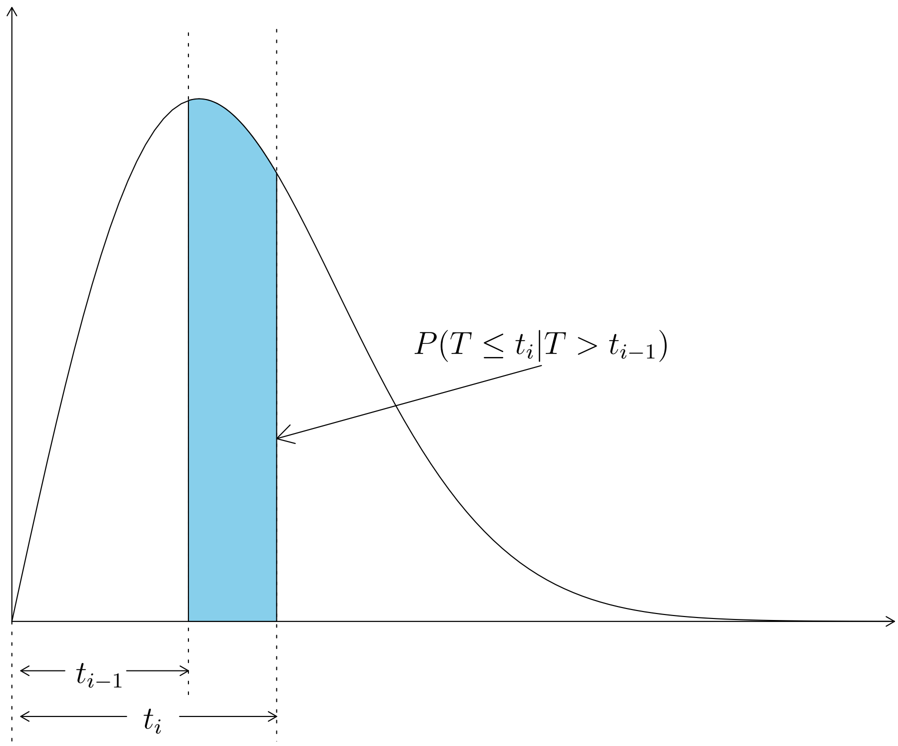


Figure B.1: The blue area represents the conditional probability of maintenance in the interval $(t_{i-1}, t_i]$

$$\begin{aligned} P(T \leq t_i | T > t_{i-1}) &= \frac{F(t_i) - F(t_{i-1})}{1 - F(t_{i-1})} \\ &= \frac{1 - R(t_i) - 1 + R(t_{i-1})}{R(t_{i-1})} \\ &= 1 - \frac{R(t_i)}{R(t_{i-1})} \end{aligned} \tag{B.7.5}$$

Let $F(\cdot)$ and $R(\cdot)$ denote the probability of maintenance and "survival" of maintenance at the respective times. By assuming a Weibull distribution of the $R(\cdot)$

and denoting $P(T \leq t_i | T > t_{i-1})$ as $F(t_i)$ we get

$$F(t_i) = 1 - \exp\left[\left(\frac{t_{i-1}}{\alpha}\right)^{\beta} - \left(\frac{t_i}{\alpha}\right)^{\beta}\right] \tag{B.7.6}$$

By taking the the derivative of the conditional distribution function with respect to t_i the corresponding conditional Weibull density function is found.

$$f(t_i) = \frac{\partial F(t_i)}{\partial t_i} = \frac{\beta}{\alpha}\left(\frac{t_i}{\alpha}\right)^{\beta-1} \exp\left[\left(\frac{t_{i-1}}{\alpha}\right)^{\beta} - \left(\frac{t_i}{\alpha}\right)^{\beta}\right] \tag{B.7.7}$$

Before continuing to derive the likelihood function it is important to note that the time up to first maintenance action doesn't obey the previous conditionality. The probability density function for the time up to the first maintenance action is determined by the regular density function for a Weibull distribution, see equation B.2.1. This can also be seen by setting $t_0 = 0$. Taking into account this difference the likelihood function becomes.

$$\begin{aligned} L(\alpha, \beta | t) &= \prod_{i=1}^{n} f(t_i) = f(t_1) \prod_{i=2}^{n} f(t_i) \\ &= \left(\frac{\beta}{\alpha^{\beta}}\right)^{n} t_1^{\beta-1} \exp\left[-\left(\frac{t_1}{\alpha}\right)^{\beta}\right] \prod_{i=2}^{n} t_i^{\beta-1} \exp\left[\left(\frac{t_{i-1}}{\alpha}\right)^{\beta} - \left(\frac{t_i}{\alpha}\right)^{\beta}\right] \qquad \text{(B.7.8)} \\ &\Downarrow \\ l(\alpha, \beta | t) &= n(\log\beta - \beta\log\alpha) + (\beta - 1)\sum_{i=1}^{n} \log(t_i) - \left(\frac{t_n}{\alpha}\right)^{\beta} \qquad \text{(B.7.9)} \end{aligned}$$

Where the last expression follows after taking into account that sums cancel each other out. A closed form solution for the MLE's is attainable in this case and is given by

$$\hat{\alpha} = \frac{t_n}{n^{1/\hat{\beta}}} \tag{B.7.10}$$

$$\hat{\beta} = \frac{n}{n\log(t_n) - \sum_{i=1}^{n} \log(t_i)} = \frac{n}{\sum_{i=1}^{n} \log(t_n/t_i)} \tag{B.7.11}$$

The variance and standard deviation can also be found by considering the inverted information matrix. This will give us estimates of the variance of the MLE's. For a 2×2 matrix the inverse of the matrix is given by

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \frac{1}{ad - bc}\begin{bmatrix} d & -b \\ -c & a \end{bmatrix} \tag{B.7.12}$$

Thus for the NHPP model the covariance matrix becomes

$$
\begin{aligned}
Var(\hat{\alpha},\hat{\beta}) &= \frac{1}{\frac{-\partial^2 l(\alpha,\beta|t)}{\partial\alpha^2}\frac{-\partial^2 l(\alpha,\beta|t)}{\partial\beta^2}-\left(\frac{-\partial^2 l(\alpha,\beta|t)}{\partial\alpha\partial\beta}\right)^2}\begin{bmatrix} \frac{-\partial^2 l(\alpha,\beta|t)}{\partial\beta^2} & -\frac{-\partial^2 l(\alpha,\beta|t}{\partial\alpha\partial\beta} \\ -\frac{-\partial^2 l(\alpha,\beta|t}{\partial\alpha\partial\beta} & \frac{-\partial^2 l(\alpha,\beta|t)}{\partial\alpha^2}\end{bmatrix} \\
&= \begin{bmatrix} \left(\frac{\hat{\alpha}}{\hat{\beta}}\right)^2(1+(\log n)^2) & \cdot \\ \cdot & \frac{\hat{\beta}^2}{n}\end{bmatrix} \qquad \text{(B.7.13)}
\end{aligned}
$$

Usually its the variance of the parameter β which are of most interest. Hence the variance of β is

$$Var(\hat{\beta}) = \frac{\hat{\beta}^2}{n} \qquad \text{(B.7.14)}$$

and consequently the standard deviation is

$$se(\hat{\beta}) = \frac{\hat{\beta}}{\sqrt{n}} \qquad \text{(B.7.15)}$$

B.7.4 Arithmetic Reduction of Age, (ARA)

For the ARA models the concept of virtual age, A_i has been implemented. In the ARA1 model the virtual age is calculated as

$$A_i = A_{i-1} + px_i = pt_i \qquad \text{(B.7.16)}$$

and for the ARA∞ model the virtual age is given as

$$A_i = p\left(A_{i-1} + x_i\right) = t_i - (1-p)\sum_{j=0}^{i-1} p^j t_{i-j} \qquad \text{(B.7.17)}$$

These follow a different conditional distribution than was the case in the NHPP model. Immediately after a the $(i-1)$th maintenance action has happened the system has the virtual age A_{i-1}. By considering the time until the next ith maintenance action, x_i, the probability that maintenance occurs in this interval is given from the conditional probability

$$
\begin{aligned}
P(T \le A_{i-1} + x_i | T > A_{i-1}) &= \frac{F(A_{i-1}+x_i) - F(A_{i-1})}{1 - F(A_{i-1})} \\
&= \frac{1 - R(A_{i-1}+x_i) - 1 + R(A_{i-1})}{R(A_{i-1})} \\
&= 1 - \frac{R(A_{i-1}+x_i)}{R(A_{i-1})} \qquad \text{(B.7.18)}
\end{aligned}
$$

Assuming a Weibull distribution of $R(\cdot)$ and denoting $P(A_{i-1}+x_i|A_{i-1})$ as $F(x_i)$ we have

$$F(x_i) = 1 - \exp\left[\left(\frac{A_{i-1}}{\alpha}\right)^{\beta} - \left(\frac{x_i + A_{i-1}}{\alpha}\right)^{\beta}\right] \tag{B.7.19}$$

Taking the derivative of the conditional probability with respect to x_i the corresponding conditional Weibull density function will become

$$f(x_i) = \frac{\partial F(x_i)}{\partial x_i} = \frac{\beta}{\alpha}\left(\frac{x_i + A_{i-1}}{\alpha}\right)^{\beta-1} \exp\left[\left(\frac{A_{i-1}}{\alpha}\right)^{\beta} - \left(\frac{x_i + A_{i-1}}{\alpha}\right)^{\beta}\right] \tag{B.7.20}$$

As was the case for the NHPP model the time up to the first interval do not follow this conditional distribution. It is determined by the regular Weibull density function, equation B.2.1. This can also be seen by setting $A_0 = 0$. Taking this into account the following expression for the likelihood function is derived

$$\begin{aligned} L(\alpha,\beta,p|x) &= \prod_{i=1}^{n} f(x_i) = f(x_1)\prod_{i=2}^{n} f(x_i) \\ &= \left(\frac{\beta}{\alpha^{\beta}}\right)^{n} x_1^{\beta-1}\exp\left[-\left(\frac{x_1}{\alpha}\right)^{\beta}\right]\prod_{i=2}^{n}(x_i + A_{i-1})^{\beta-1}\exp\left[\left(\frac{A_{i-1}}{\alpha}\right)^{\beta} - \left(\frac{x_i + A_{i-1}}{\alpha}\right)^{\beta}\right] \end{aligned} \tag{B.7.21}$$

$$\Downarrow$$

$$\begin{aligned} l(\alpha,\beta,p|x) =& n(\log\beta - \beta\log\alpha) - \left(\frac{x_1}{\alpha}\right)^{\beta} + (\beta-1)\left(\log(x_1) + \sum_{i=2}^{n}\log(x_i + A_{i-1})\right) \\ &+ \sum_{i=2}^{n}\left[\left(\frac{A_{i-1}}{\alpha}\right)^{\beta} - \left(\frac{x_i + A_{i-1}}{\alpha}\right)^{\beta}\right] \end{aligned} \tag{B.7.22}$$

Where the log-likelihood function for the respective ARA∞ and ARA1 model is given by considering the appropriate virtual age function A_i. No closed form solutions of the MLE's of α, β and p are attainable for the ARA models in this case. Thus the estimation of the MLE's highly rely on numerical maximization procedures.

C Plots of Expected Number of Maintenance Events

In this section the plots for the different systems considered in section 4 are given. The dotted lines represent the models and the black line indicates the dataset. Also note that in the cases where the ARA models gives a endpoint $p = 0$ or $p = 1$ for the p parameter the respective RP or NHPP lines is only plotted not the ARA line.

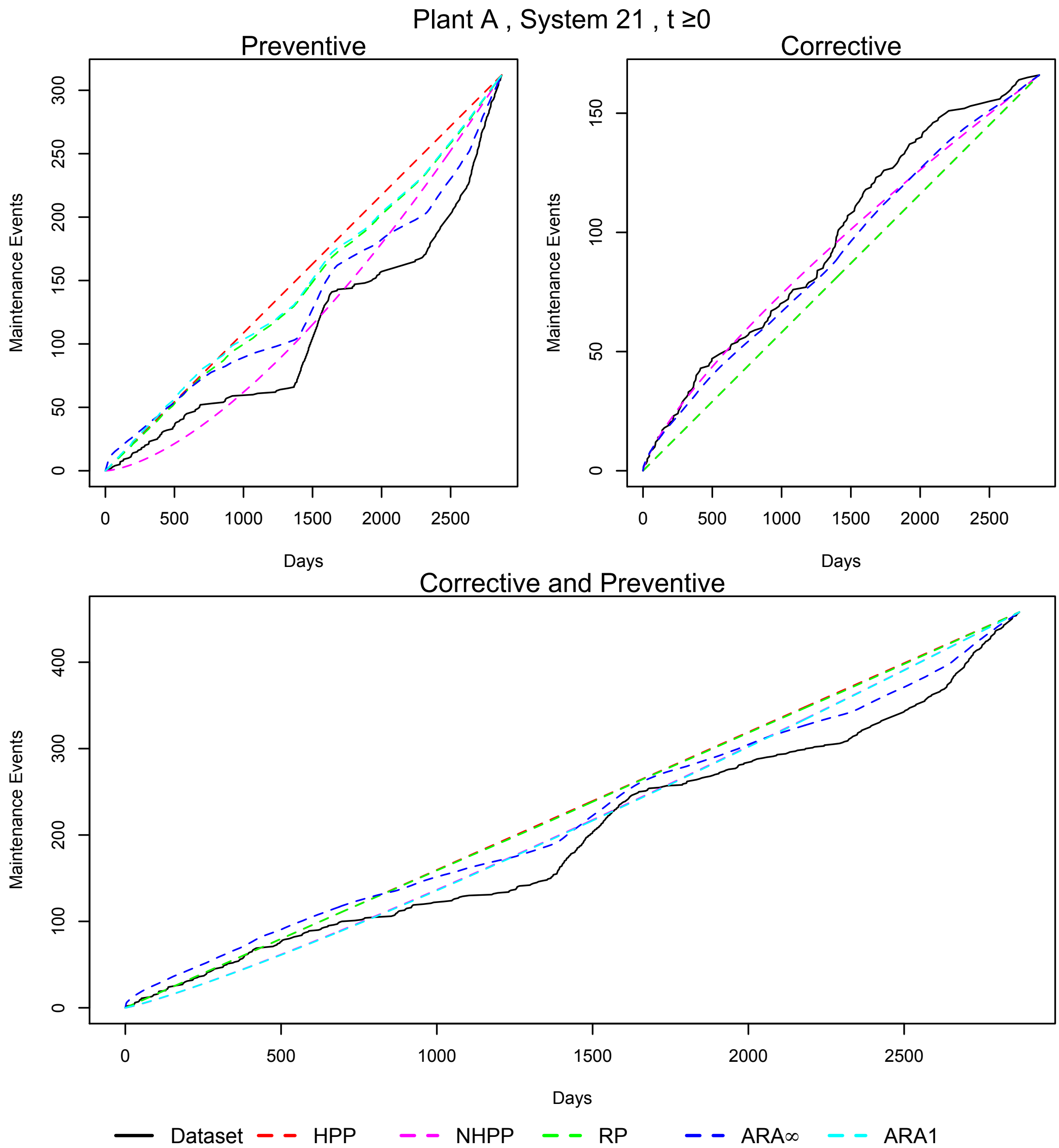


Figure C.1: Plot of expected number of maintenance events for system 21, plant A

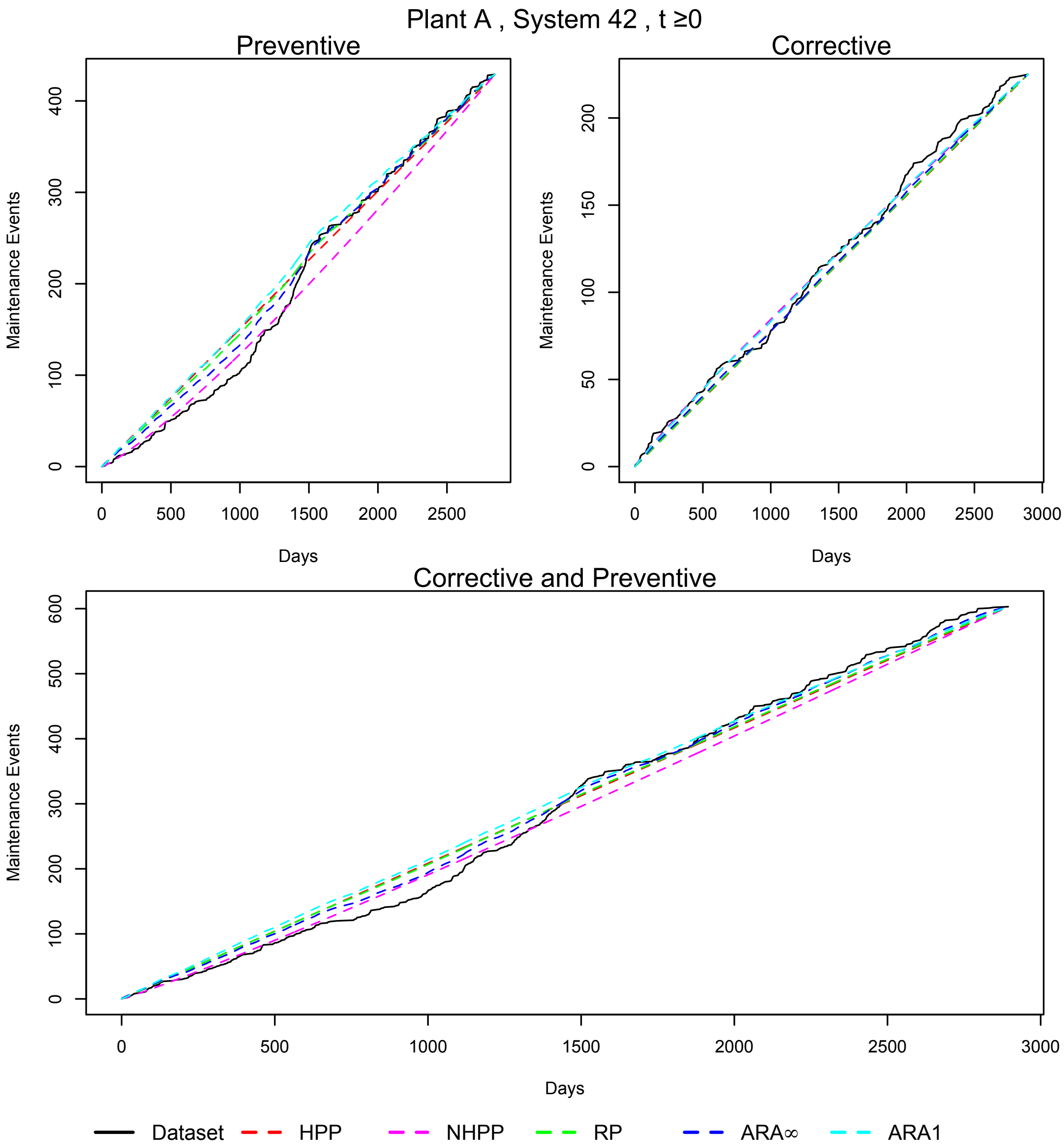


Figure C.2: Plot of expected number of maintenance events for system 42, plant A

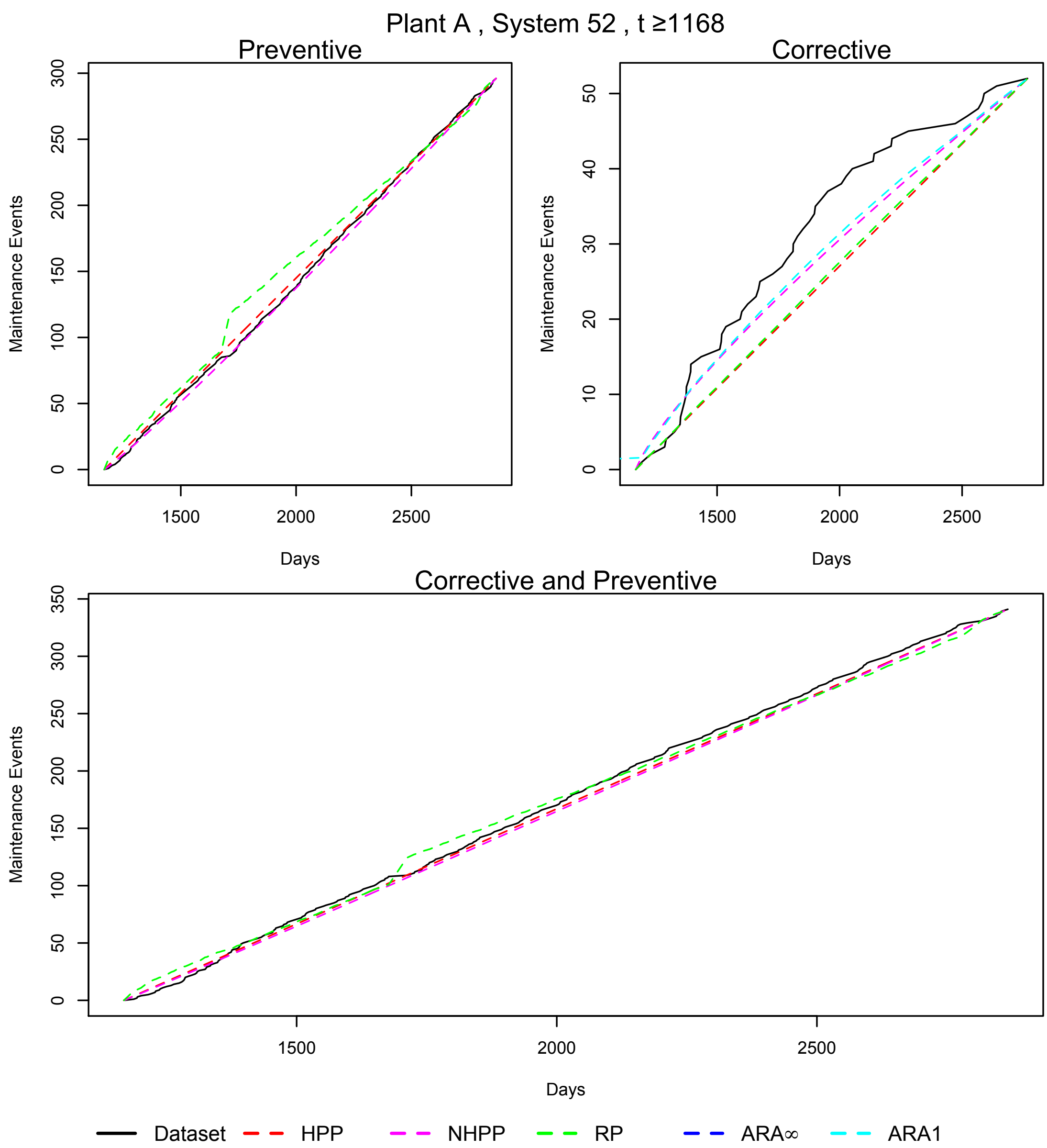


Figure C.3: Plot of expected number of maintenance events for system 52, plant A

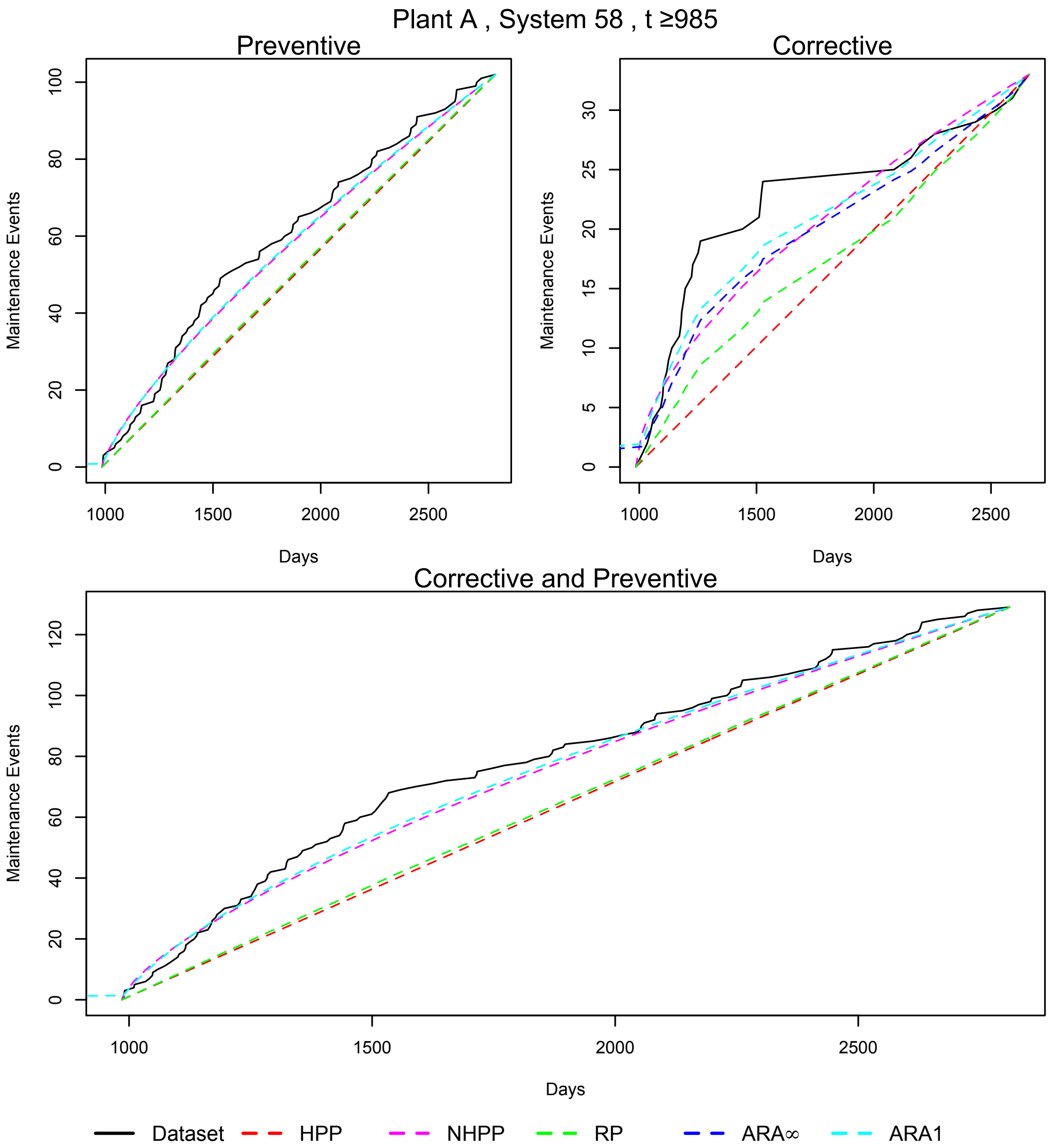


Figure C.4: Plot of expected number of maintenance events for system 58, plant A

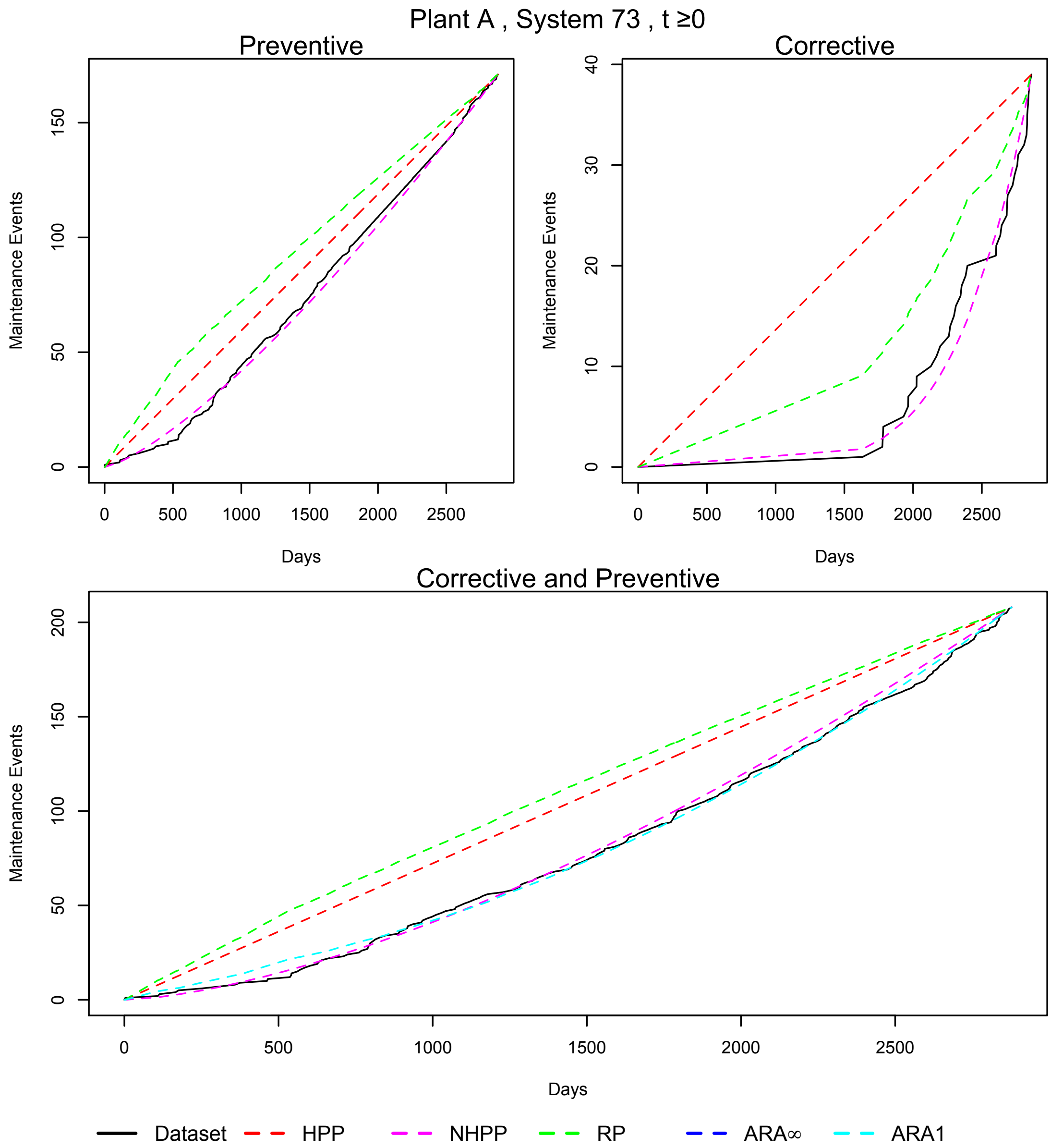


Figure C.5: Plot of expected number of maintenance events for system 73, plant A

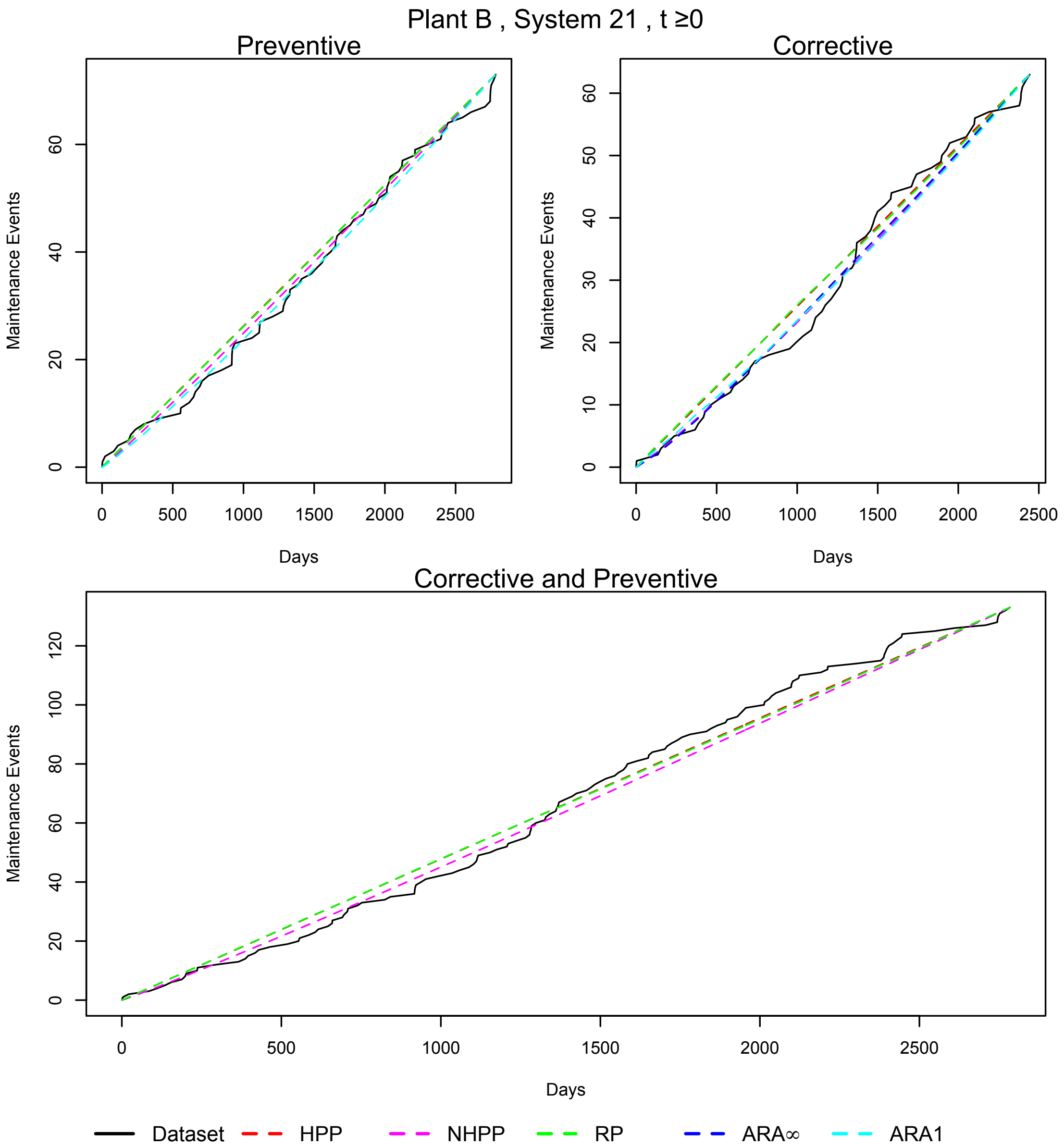


Figure C.6: Plot of expected number of maintenance events for system 21, plant B

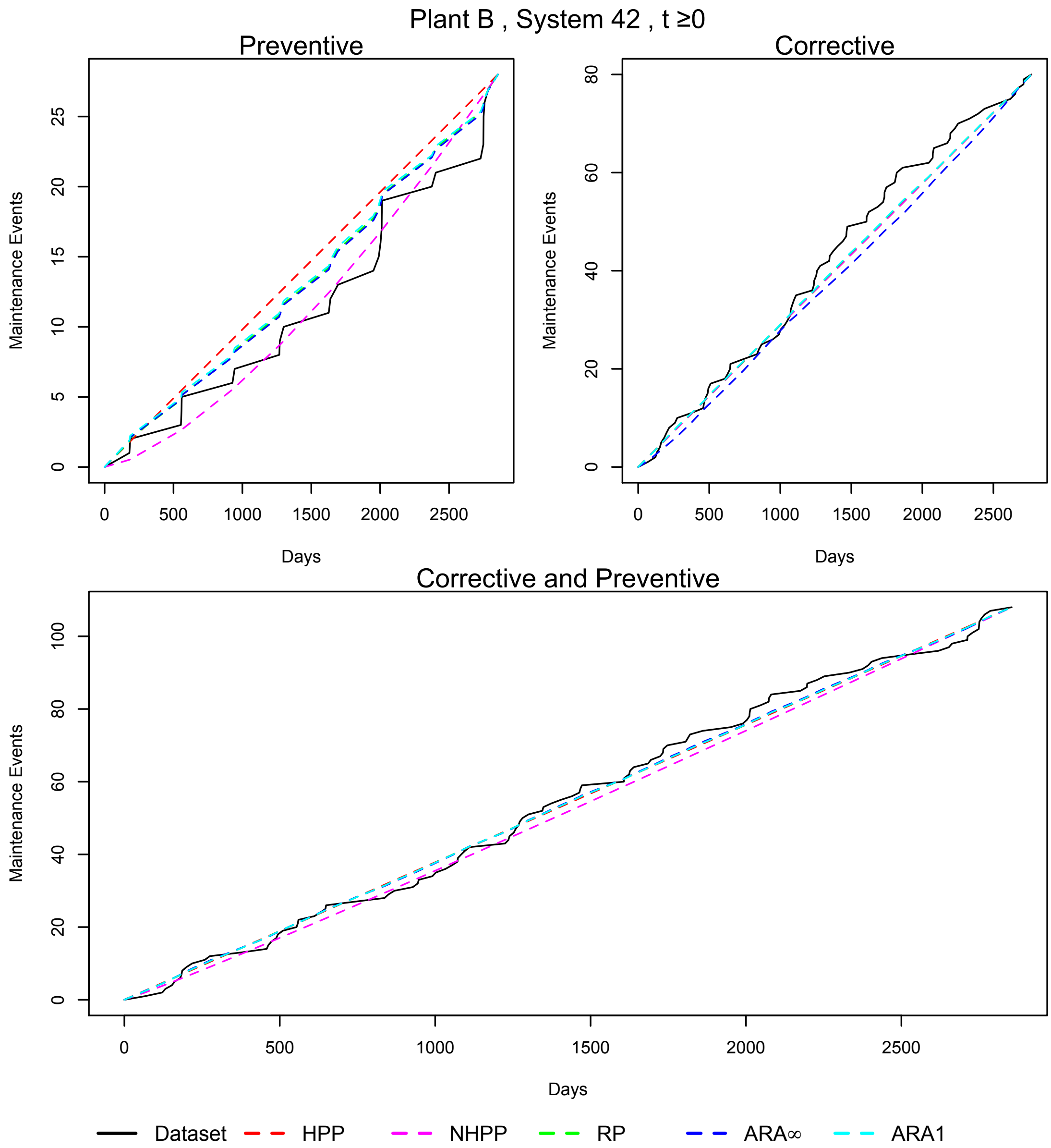


Figure C.7: Plot of expected number of maintenance events for system 42, plant B

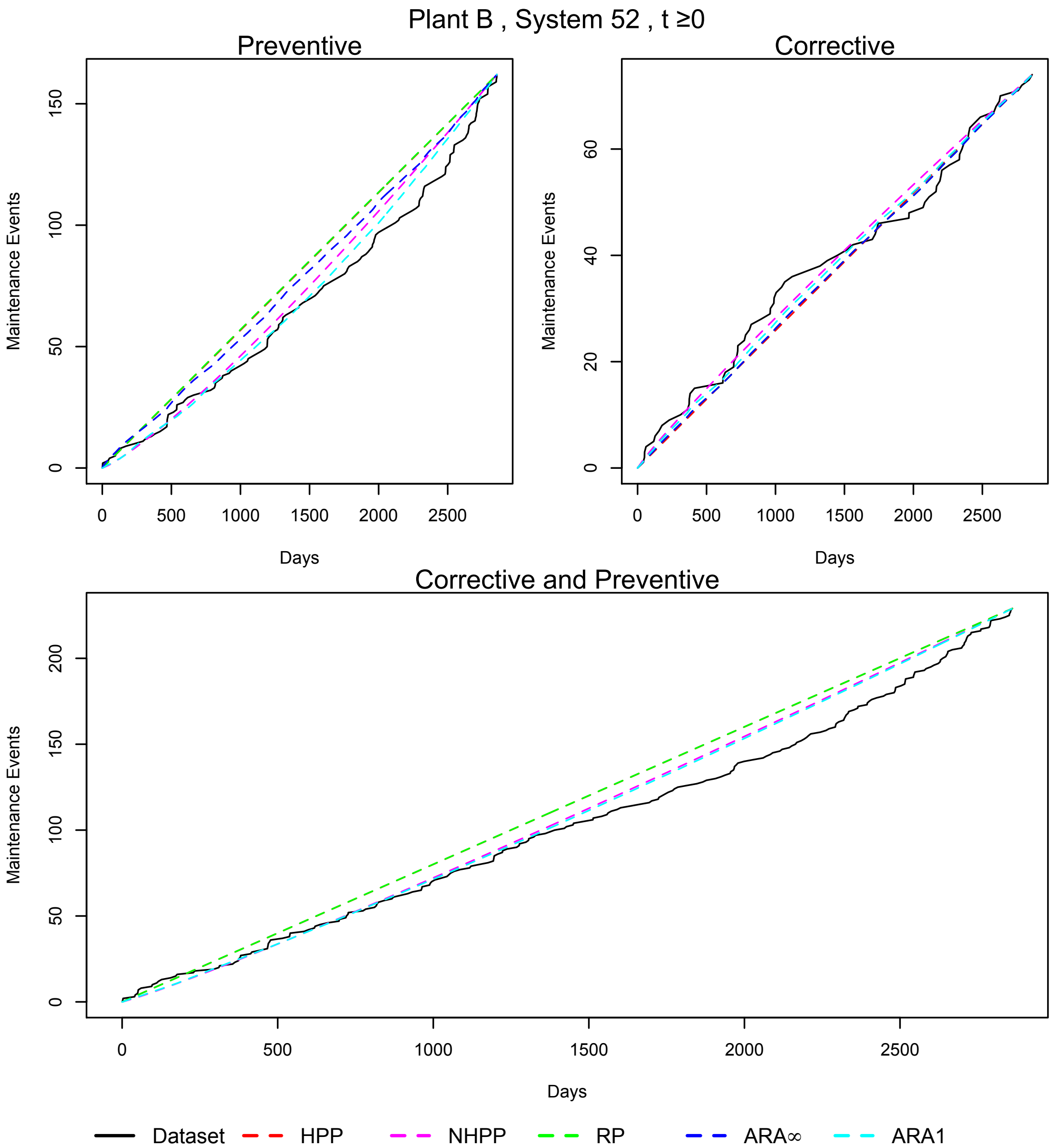


Figure C.8: Plot of expected number of maintenance events for system 52, plant B

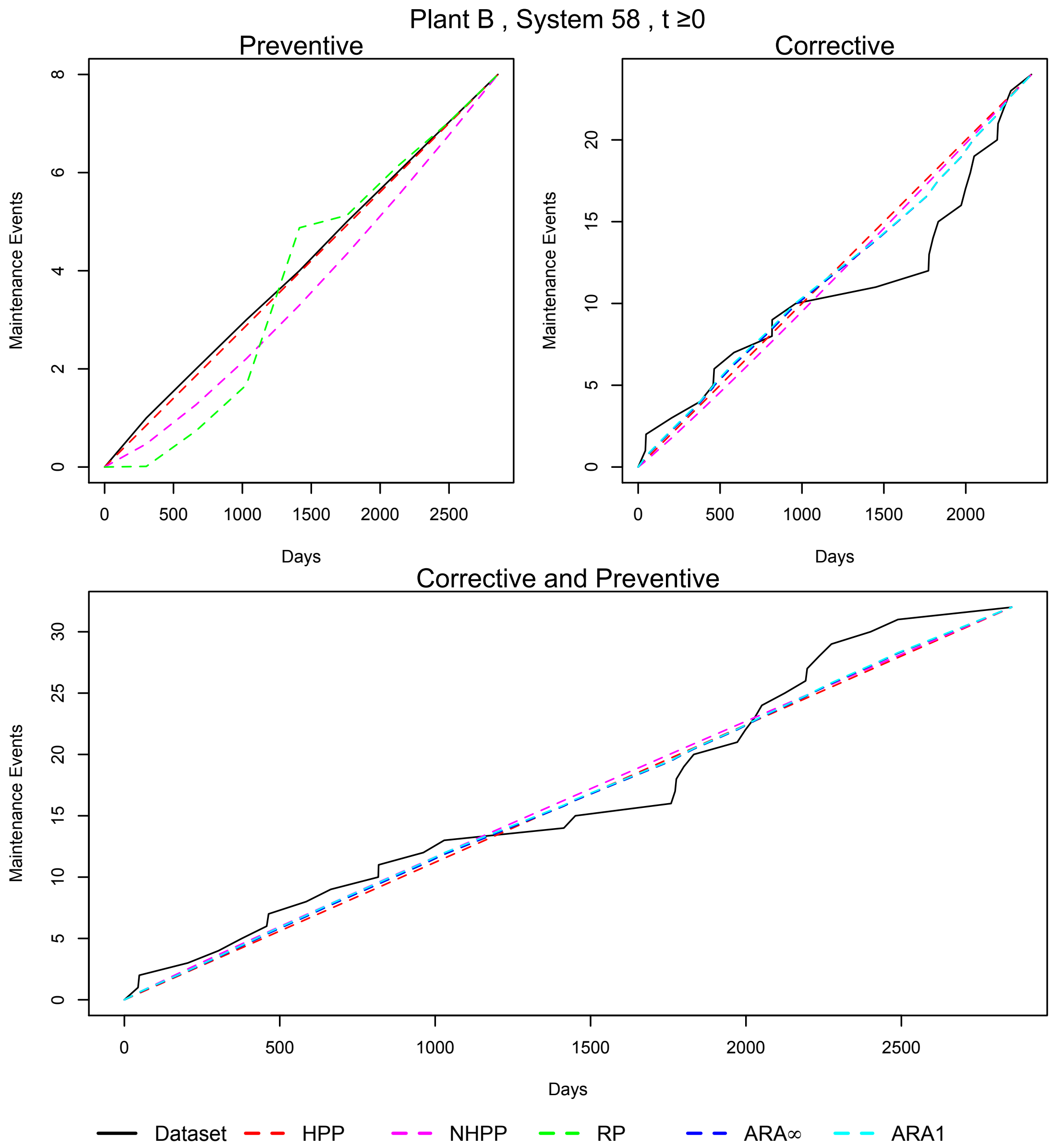


Figure C.9: Plot of expected number of maintenance events for system 58, plant B

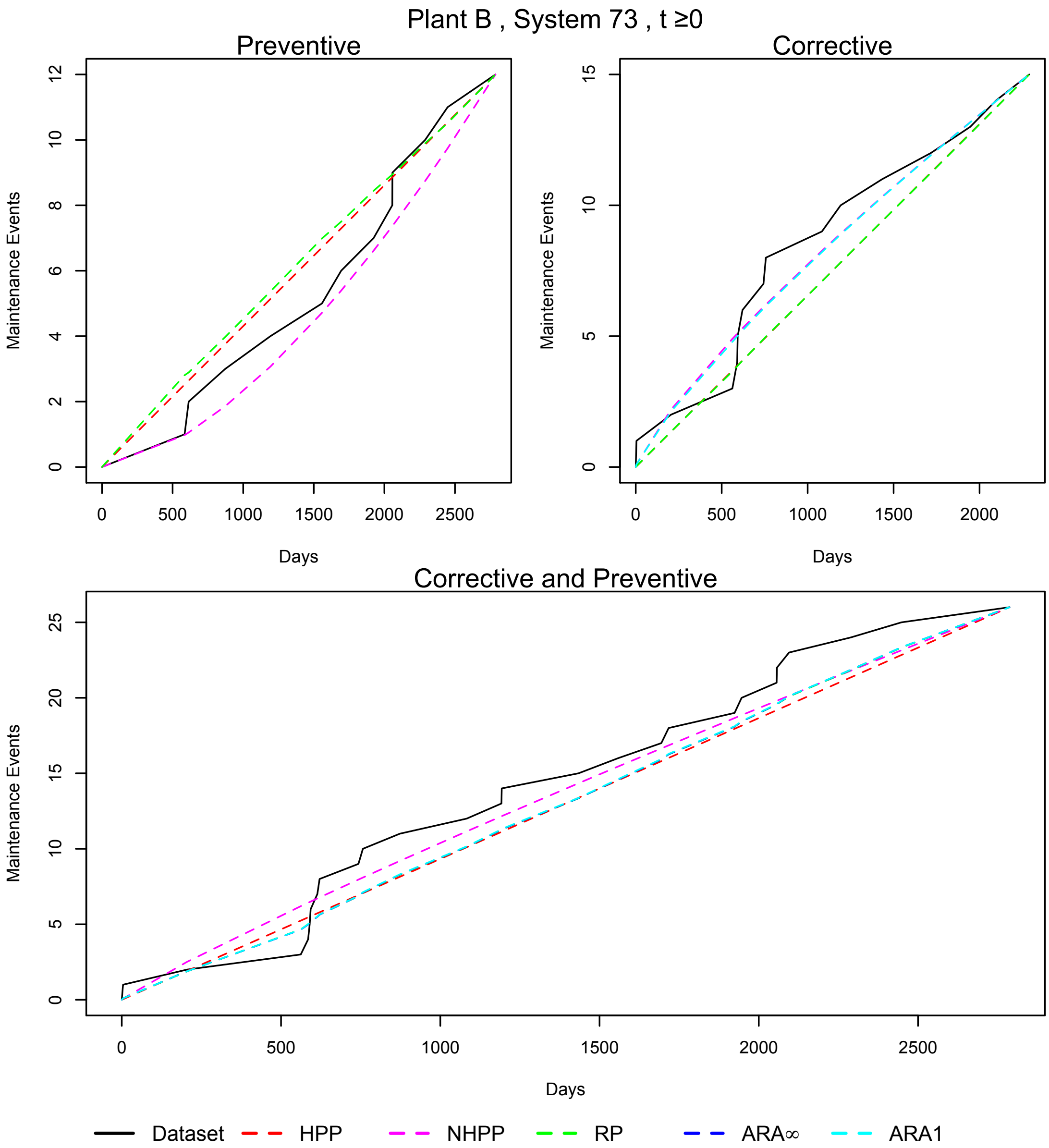


Figure C.10: Plot of expected number of maintenance events for system 73, plant B

D *R*-code

```
# Function for finding the MLE estimates of the models
##################
## INPUT ##
# data: the dataset we want to fit
# system: the systems we want to fit it on
# rocof: The models we try to fit to the data
#              The possible models are:
#              HPP, RP, PL, Ara1, AraInf
# tint: time interval we want to consider, (tint,->)
## OUTPUT ##
# mle's: maximum likelihood estimates of the parameters in the model
# se.mle's: standard errors of the mle's based on inverting the hessian
# logL: log-likelihood
# p.value: p-value corresponding to the test statistic
# test.statistic: Test statistic for the given method
# n: number of registered events
##################
fitMLE = function(data,system,rocof,tint,dup=FALSE){
        # First we check if we want to consider a specific system or the plants as a whole
        if(is.na(system)){sys = data}
        else if(system==1220){sys = subset(data,data$Plant==system,drop=T)}
        else if(system==1221){sys = subset(data,data$Plant==system,drop=T)}
        else{sys = subset(data,data$System==system,drop=T)}

        #If we have too few registered events we display the result as NA
        if(length(sys$Bas..start.date)<=2){
                res = list(n=length(sys$Bas..start.date),logL=NA,alpha=NA,p=NA,beta=NA,se.alpha
                    =NA,se.beta=NA,se.p=NA,VP1=NA,pvalueP1=NA,VB1=NA,pvalueB1=NA,VP0=NA,
                    pvalueP0=NA,message=NA,method=NA,conv=NA,p.value=NA,pvalue=NA,KS=NA,par=NA
                    )
                }
        else{
        # Sort after the column Bas..start.date
        sys = sys[order(sys$Bas..start.date),]
        #We call the function IntOc to find the interoccurence times
        sys = IntOc(sys,tint)
        # Edit time column if tint is present
        sys$Bas..start.date = sys$Bas..start.date - tint
        # If dup is TRUE we only consider the rows where the values of sys$Bas..start.date is
            not the same
        if(dup==TRUE){ sys = sys[!is.na(sys$IntDup),]}
        # We also check if the rocof function given is supported
        if(all(rocof != c("PL","Ara1","AraInf","RP","HPP"))){stop("ROCOF function isn't 
            supported")}
        # After removing dup entries we again has check if we have to few observations
        if(length(sys$Bas..start.date)<=2){res = list(n=length(sys$Bas..start.date),logL=NA,
            alpha=NA,p=NA,beta=NA,se.alpha=NA,se.beta=NA,se.p=NA,VP1=NA,pvalueP1=NA,VB1=NA,
            pvalueB1=NA,VP0=NA,pvalueP0=NA,message=NA,method=NA,conv=NA,p.value=NA,pvalue=NA,
            KS=NA,par=NA)}
        else{
                t = sys$Bas..start.date
                ###
                if(rocof=="HPP"){ #RESULTS FOR THE HPP MODEL
                        n = length(t)
                        maxt=max(t)
                        alpha = maxt/n
                        s = maxt/(n^(3/2))
                        logL = -n*log(alpha)-maxt/alpha
                        #Kolmogorov-Smirnov test
                        KS = KStest(time=t,param=alpha,rocof=rocof)
                        res = list(n=n,logL=logL,alpha=alpha,se.alpha=s,KS=KS$KS$p.value,NT=KS$
                            NT)
                        }
                ###
                else if(rocof=="PL"){ #RESULTS FOR THE NHPP MODEL
                        n = length(t)
                        par = c()
                        par[2] = n/sum(log(t[n]/t)) #beta
                        par[1] = t[n]/(n^(1/par[2])) #alpha
                        #First we find the test statistic before we display the results
                        n = length(t)
                        V = 2*sum(log(t[n])-log(t))
                        if(par[2]>1){p = pchisq(V,2*(n-1))}
                        else{p = pchisq(V,2*(n-1),lower.tail=FALSE)}
                        if(V<0){p = NA}
                        #Kolmogorov-Smirnov test
                        KS = KStest(time=t,param=par,rocof=rocof)
                        #standard deviation
                        s = sqrt(c((1/n)*((par[1]/par[2])^2)*(1+(log(n))^2),(par[2]^2)/n))
```

```
        res = list(n=n,logL=logL(par,time=t,Rocof=rocof),alpha=par[1],se.alpha=
              s[1],beta=par[2],se.beta=s[2],test.statistic=V,p.value=p,df=2*(n
              -1),KS=KS$KS$p.value,NT=KS$NT,par=par)
        }
###
else if(rocof=="RP"){ #RESULTS FOR THE RP MODEL
        logLRPtrans = function(param,time){
                n = length(time)
                xi = append(diff(time,1),time[1],after=0)
                a = param[1]
                b = param[2]
                res = n*(b-a*exp(b))+(exp(b)-1)*sum(log(xi))-sum((xi/exp(a))^
                      exp(b))
                return(res)
                }
        opRP = optim(par=c(0,0),fn=logLRPtrans,gr=NULL,time=t,method="BFGS",
              control=list(maxit=10000,fnscale=-1),hessian=FALSE)
        parRP = c(exp(opRP$par))
        n = length(t)
        lb1 = n*(log(n)-1-log(max(t)))
        VRP = -2*(lb1-opRP$value)
        if(VRP<0){pvalue=NA}
        else(pvalue=pchisq(VRP,1,lower.tail=FALSE))
        if(is.na(pvalue)){sig = "No"}
        else if(pvalue>0.05){sig = "No"}
        else{sig = "Yes"}
        hessian = hessian(logL,x=parRP,time=t,Rocof=rocof)
        s = sqrt(-diag(solve(hessian)))
        #Kolmogorov-Smirnov test
        KS = KStest(time=t,param=parRP,rocof=rocof)
        res = list(n=n,logL=opRP$value,alpha=parRP[1],se.alpha=s[1],beta=parRP
              [2],se.beta=s[2],par=parRP,test.statistic=VRP,pvalue=pvalue,sig=
              sig,KS=KS$KS$p.value,NT=KS$NT)
        }
###
else if(rocof=="Ara1"){ #RESULTS FOR THE ARA1 MODEL
        tr = transAra1(time=t,start=c(0,0,0))
        #First we find the test statistics
        TS = testStat(time=tr$time,param=tr$par,logLik=tr$op$value,logLP0=tr$
              logLP0,logLP1=tr$logLP1)
        if(tr$par[3]==0){
                par =tr$par[-3]
                hessian = hessian(logL,x=par,time=tr$time,Rocof="RP")
                s = sqrt(-diag(solve(hessian)))
                s[3]=NA
                }
        else if(tr$par[3]==1){
                par = tr$par[-3]
                n = length(tr$time)
                s = sqrt(c((1/n)*((par[1]/par[2])^2)*(1+(log(n))^2),(par[2]^2)/
                      n))
                s[3]=NA
                }
        else{
                hessian = try(hessian(logL,x=tr$par,time=tr$time,Rocof=rocof),
                      silent=TRUE)
                sol = try(solve(hessian),silent=TRUE)
                if(class(sol)=="try-error"){
                        s = c(NA,NA,NA)
                        }
                else{
                        s = sqrt(-diag(sol))
                        }
                }
        #Kolmogorov-Smirnov test
        KS = KStest(time=t,param=tr$par,rocof=rocof)
        res = list(n=length(tr$time),logL=tr$op$value,alpha=tr$par[1],se.alpha=
              s[1],beta=tr$par[2],se.beta=s[2],p=tr$par[3],se.p=s[3],message=tr$
              op$message,conv=tr$op$convergence,method=tr$method,VP1=TS$VP1,
              pvalueP1=TS$pvalueP1,VB1=TS$VB1,pvalueB1=TS$pvalueB1,VP0=TS$VP0,
              pvalueP0=TS$pvalueP0,KS=KS$KS$p.value,NT=KS$NT,par=tr$par)
        }
###
else if(rocof=="AraInf"){ #RESULTS FOR THE ARAINF MODEL
        tr = transAraInf(time=t)
        #Fist we find the test statistics
        TS = testStat(time=tr$time,param=tr$par,logLik=tr$op$value,logLP0=tr$
              logLP0,logLP1=tr$logLP1)
        if(tr$par[3]==0){
                par = tr$par[-3]
                hessian = hessian(logL,x=par,time=tr$time,Rocof="RP")
                s = sqrt(-diag(solve(hessian)))
                s[3]=NA
```

```
                    }
            else if(tr$par[3]==1){
                    par = tr$par[-3]
                    n = length(tr$time)
                    s = sqrt(c((1/n)*((par[1]/par[2])^2)*(1+(log(n))^2),(par[2]^2)/
                        n))
                    s[3]=NA
                    }
            else{
                    hessian = try(hessian(logL,tr$par,time=tr$time,Rocof=rocof),
                        silent=TRUE)
                    sol = try(solve(hessian),silent=TRUE)
                    if(class(sol)=="try-error"){
                            s = c(NA,NA,NA)
                            }
                    else{
                            s = sqrt(-diag(sol))
                            }
                    }
            #Kolmogorov-Smirnov test
            KS = KStest(time=t,param=tr$par,rocof=rocof)
            res = list(n=length(tr$time),logL=tr$op$value,alpha=tr$par[1],se.alpha=
                s[1],beta=tr$par[2],se.beta=s[2],p=tr$par[3],se.p=s[3],message=tr$
                op$message,conv=tr$op$convergence,method=tr$method,VP1=TS$VP1,
                pvalueP1=TS$pvalueP1,VB1=TS$VB1,pvalueB1=TS$pvalueB1,VP0=TS$VP0,
                pvalueP0=TS$pvalueP0,KS=KS$KS$p.value,NT=KS$NT,par=tr$par)
            }
        ###
        }
        }
    return(res)
    }
###############################################

###############################################
# Function for finding the inter-occurence times.
IntOc = function(system,tint=0){
        if(length(system$Bas..start.date)<=1){
                res = system
                }
        else{
                system = system[order(system$Bas..start.date),]
                temp = diff(system$Bas..start.date-tint,1)
                system$IntOc = append(temp,system$Bas..start.date[1]-tint,after=0)
                system$IntDup = system$IntOc
                system$IntDup[system$IntDup==0]=NA
                rm(temp)

                system = system[order(system$Bas..start.date),]
                res = system
                }
        return(res)
        }
###############################################

###############################################
#Function for finding the MLE's for the ARAinf model
transAraInf = function(time){
        sumAI = function(p,time){
                        n = length(time)
                        t1 = time[1]

                        Xi = append(diff(time,1),t1,after=0)
                        res =c(p*Xi[1])
                        iter = 2
                        while(iter<=n-1){
                                res[iter]=p*(res[iter-1]+Xi[iter])
                                iter = iter + 1
                                }
                        return(res)
                        }
        FN3 = function(param,time){
                n = length(time)
                a = param[1]
                b = param[2]
                p = param[3]
                t1 = time[1]

                pmark = exp(p)/(1+exp(p)) # p between 0,1
                sumai = sumAI(pmark,time)
                Xi = diff(time,1)
                Ai = Xi + sumai
```

```
          Aiminus = sumai

          res = n*(b-a*exp(b))-(t1/exp(a))^exp(b)+(exp(b)-1)*log(t1)+(exp(b)-1)*sum(log(
               Ai))+sum((Aiminus/exp(a))^exp(b))-sum((Ai/exp(a))^exp(b))
          return(res)
          }
t = time
n = length(t)
start = expand.grid(a=0,b=0,p=c(0,1,2,3,-1,-2,-3))
opOld = list(value=-10000)
parOld = 0
methodOld = "Failure"
i = 1
while(i<=length(start$a)){
          op = try(optim(par=start[i,], fn=FN3, gr=NULL, time=t, method="BFGS", control=list(
               maxit=20000,fnscale=-1), hessian=TRUE), silent=TRUE)

          if(class(op)=="try-error"){
                    op2 = try(spg(par=start[i,], fn=FN3, gr=NULL, method=3,control=list(maxit
                         =20000,maxfeval=100000,maximize=TRUE, gtol=1E-4,trace=FALSE),time=t
                         ), silent=TRUE)
                    if(class(op2)=="try-error"){
                              op = list(value=-10000)
                              par = c(1,1,1)
                              method = "failure"
                              }
                    else{
                              op = op2
                              par = c(exp(op2$par[1]),exp(op2$par[2]),exp(op2$par[3])/(1+exp(
                                   op2$par[3])))
                              method = "spg"
                              }
                    }
          else{
                    op = op
                    par = c(exp(op$par[1]),exp(op$par[2]),exp(op$par[3])/(1+exp(op$par[3]))
                         )
                    method = "optim"
                    }
          if(op$value>opOld$value){
                    opOld = op
                    parOld = par
                    methodOld = method
                    }
          else{
                    opOld = opOld
                    parOld = parOld
                    methodOld = methodOld
                    }
          i = i+1
          }
#Then the result is given as
op = opOld
par = parOld
method = methodOld

### Check the endpoints (p=0 or p=1) to see if they have greater logLik

# First we find the loglik for p=1 and p=0
# checking p = 1
bPL = n/sum(log(t[n]/t)) #beta
aPL = t[n]/(n^(1/bPL)) #alpha
logLP1 = logL(c(aPL,bPL),time=t,Rocof="PL")
parP1 = c(aPL,bPL,1)
opP1 = list(value=logLP1,convergence=0,message="NULL")

#checking p = 0
logLRP = function(param,time){
          n = length(time)
          xi = append(diff(time,1),time[1],after=0)
          a = param[1]
          b = param[2]
          res = n*(b-a*exp(b))+(exp(b)-1)*sum(log(xi))-sum((xi/exp(a))^exp(b))
          return(res)
          }
opP0 = optim(par=c(0,0),fn=logLRP, gr=NULL, time=t, method="BFGS", control=list(maxit
      =10000,fnscale=-1),hessian=TRUE)
logLP0 = opP0$value
parP0 = c(exp(opP0$par),0)

## Compare the loglik
if((op$value-logLP1) > 1E-5 && (op$value-opP0$value) > 1E-5){
          #Here the ARA1 model is the correct
```

```
            res = list(op=op,par=par,method=method,time=t,logLP0=opP0$value,logLP1=logLP1)
            }
        else{
            if((logLP1-opP0$value)>1E-5){
                # Here the NHPP model is the correct
                res = list(op=opP1,par=parP1,method="Endpoint, p=1",time=t,logLP0=opP0$
                    value,logLP1=logLP1)
                }
            else{
                # Here the RP model is the correct
                res = list(op=opP0,par=parP0,method="Endpoint,p=0",time=t,logLP0=opP0$
                    value,logLP1=logLP1)
                }
            }

        return(res)
        }
################################################

transAra1 = function(time,start){
        # here we try Ara1
        # First we find the log-likelihood for the transformed ARA1 model.
        FN2 = function(param,time){
            n = length(time)
            a = param[1]
            b = param[2]
            p = param[3]
            tminus = time[-n]
            t1 = time[1]

            pmark = exp(p)/(1+exp(p)) # p between 0,1
            Xi = diff(time,1)
            Ai = Xi+pmark*tminus
            Aiminus = pmark*tminus

            res = n*(b-a*exp(b))-(t1/exp(a))^exp(b)+(exp(b)-1)*log(t1)+(exp(b)-1)*sum(log(
                Ai))+sum((Aiminus/exp(a))^exp(b))-sum((Ai/exp(a))^exp(b))
            return(res)
            }
        t = time
        n = length(t)
        start = expand.grid(a=0,b=0,p=c(0,1,2,3,-1,-2,-3))
        opOld = list(value=-10000)
        parOld = 0
        methodOld = "Failure"
        i = 1
        while(i<=length(start$a)){
            op = try(optim(par=start[i,], fn=FN2, gr=NULL,time=t,method="BFGS",control=list(
                maxit=20000,fnscale=-1),hessian=TRUE),silent=FALSE)

            if(class(op)=="try-error"){
                op2 = try(spg(par=start[i,], fn=FN2, gr=NULL, method=3,control=list(maxit
                    =10000,maxfeval=100000,maximize=TRUE, gtol=1E-4,trace=FALSE),time=t
                    ),silent=TRUE)
                if(class(op2)=="try-error"){
                    op = list(value=-10000)
                    par = c(1,1,1)
                    method = "failure"
                    }
                else{
                    op = op2
                    par = c(exp(op2$par[1]),exp(op2$par[2]),exp(op2$par[3])/(1+exp(
                        op2$par[3])))
                    method = "spg"
                    }
                }
            else{
                op = op
                par = c(exp(op$par[1]),exp(op$par[2]),exp(op$par[3])/(1+exp(op$par[3]))
                    )
                method = "optim"
                }
            if(op$value>opOld$value){
                opOld = op
                parOld = par
                methodOld = method
                }
            else{
                opOld = opOld
                parOld = parOld
                methodOld = methodOld
                }
            i = i+1
```

```
                }
        #Then the result is given as
        op = opOld
        par = parOld
        method = methodOld

        ### Check the endpoints (p=0 or p=1) to see if they have greater logLik

        # First we find the loglik for p=1 and p=0
        # checking p = 1 this equals the NHPP model.
        bPL = n/sum(log(t[n]/t)) #beta
        aPL = t[n]/(n^(1/bPL)) #alpha
        logLP1 = logL(c(aPL,bPL),time=t,Rocof="PL")
        parP1 = c(aPL,bPL,1)
        opP1 = list(value=logLP1,convergence=0,message="NULL")

        #checking p = 0 this equals the RP model
        logLRP = function(param,time){
                n = length(time)
                xi = append(diff(time,1),time[1],after=0)
                a = param[1]
                b = param[2]
                res = n*(b-a*exp(b))+(exp(b)-1)*sum(log(xi))-sum((xi/exp(a))^exp(b))
                return(res)
                }
        opP0 = optim(par=c(0,0),fn=logLRP,gr=NULL,time=t,method="BFGS",control=list(maxit
             =10000,fnscale=-1),hessian=TRUE)
        parP0 = c(exp(opP0$par),0)

        ## Compare the loglik
        if((op$value-logLP1) > 1E-5 && (op$value-opP0$value) > 1E-5){
                #Here the ARA1 model is the correct
                res = list(op=op,par=par,method=method,time=t,logLP0=opP0$value,logLP1=logLP1)
                }
        else{
                if((logLP1-opP0$value)>1E-5){
                        # Here the NHPP model is the correct
                        res = list(op=opP1,par=parP1,method="Endpoint, p=1",time=t,logLP0=opP0$
                             value,logLP1=logLP1)
                        }
                else{
                        # Here the RP model is the correct
                        res = list(op=opP0,par=parP0,method="Endpoint,p=0",time=t,logLP0=opP0$
                             value,logLP1=logLP1)
                        }
                }

        return(res)
        }


##############################################
## Function for finding the log-likelihood
logL = function(param,time,Rocof){
        if(Rocof=="PL"){
                n = length(time)
                a = param[1]
                b = param[2]
                res = n*(log(b)-b*log(a))+(b-1)*sum(log(time))-(max(time)/a)^b
                return(res)
                }
        else if(Rocof == "Ara1"){
                n = length(time)
                a = param[1]
                b = param[2]
                p = param[3]
                tminus=time[-n]
                t1 = time[1]
                Xi = diff(time,1)
                Ai = Xi+p*tminus
                Aiminus = p*tminus
                res = (n*(log(b)-b*log(a))+(b-1)*log(t1)-(t1/a)^b+(b-1)*sum(log(Ai))+sum((
                     Aiminus/a)^b)-sum((Ai/a)^b))
                return(res)
                }
        else if(Rocof=="AraInf"){
                n = length(time)
                a = param[1]
                b = param[2]
                p = param[3]
                t1 = time[1]
                sumAI = function(p,time){
                        n = length(time)
                        t1 = time[1]
```

```
                    Xi = append(diff(time,1),t1,after=0)
                    res =c(p*Xi[1])
                    iter = 2
                    while(iter<=n-1){
                            res[iter]=p*(res[iter-1]+Xi[iter])
                            iter = iter + 1
                            }
                    return(res)
                    }
            sumai = sumAI(p,time)
            Xi = diff(time,1)
            Ai = Xi + sumai
            Aiminus = sumai
            res = (n*(log(b)-b*log(a))+(b-1)*log(t1)-(t1/a)^b+(b-1)*sum(log(Ai))+sum((
                Aiminus/a)^b)-sum((Ai/a)^b))
            return(res)
            }
        else if(Rocof=="RP"){
            n = length(time)
            a = param[1]
            b = param[2]
            Xi = append(diff(time,1),time[1],after=0)
            res = n*(log(b)-b*log(a))+(b-1)*sum(log(Xi))-sum((Xi/a)^b)
            return(res)
            }
        }
################################################


################################################
#function for estimating the test statistics for Ara1 and AraInf
testStat = function(time,param,logLik,logLP0,logLP1){
        n = length(time)
        ## test for testing if p not equal 1 ##
        #Likelihood ratio test
        VP1 = -2*(logLP1-logLik)
        if(VP1<0){pvalueP1 = NA}
        else{pvalueP1 = pchisq(VP1,1,lower.tail=FALSE)}

        ## test for testing if b not equal 1 ##
        aB1 = time[n]/n
        logLB1 = (-n*log(aB1)-time[n]/aB1)
        #Likelihood ratio test
        VB1 = -2*(logLB1-logLik)
        if(VB1<0){pvalueB1=NA}
        else{pvalueB1 = pchisq(VB1,1,lower.tail=FALSE)}

        ## test for testing if p not equal 0 ##
        #Likelihood ratio test
        VP0 = -2*(logLP0-logLik)
        if(VP0<0){pvalueP0 = NA}
        else{pvalueP0 = pchisq(VP0,1,lower.tail=FALSE)}
        #return the results
        res = list(logLik=logLik,logLP1=logLP1,VP1=VP1,pvalueP1=pvalueP1,logLB1=logLB1,VB1=VB1,
            pvalueB1=pvalueB1,logLP0=logLP0,VP0=VP0,pvalueP0=pvalueP0)
        return(res)
        }
################################################
#Function for the Kolmogorov-Smirnov test
KStest = function(time,param,rocof){
        if(rocof=="Ara1"){
                NT = cumsum(ZARA1(time,param))
                }
        else if(rocof=="AraInf"){
                NT = cumsum(ZARAINF(time,param))
                }
        else if(rocof=="RP"){
                NT = cumsum(ZRP(time,param))
                }
        else if(rocof=="HPP"){
                a = param[1]
                NT = time/a
                }
        else if(rocof=="PL"){
                a = param[1]
                b = param[2]
                NT = (time/a)^b
                }
        N = length(time)
        freq = cumsum(rep(1,N)/N)

        ks = ks.test(freq,NT/N)
        res = list(KS=ks,NT=NT)
```

```
        return(res)
        }
################################################

################################################
# Function for finding the expected number of events in each interval
ZRP = function(time,param){
        a = param[1]
        b = param[2]
        Xi = append(diff(time,1),time[1],after=0)
        res = (Xi/a)^b
        return(res)
        }
ZARA1 = function(time,param){
        a = param[[1]]
        b = param[[2]]
        p = param[[3]]
        if(p==0){
                par = param[-3]
                res = ZRP(time,par)
                }
        else if(p==1){
                par = param[-3]
                res = ZPL(time,par)
                }
        else{
                Xi = append(diff(time,1),time[1],after=0)
                n = length(time)
                Aiminus = 0
                res = c(((Xi[1]+Aiminus)/a)^b)
                iter = 2
                while(iter<n+1){
                        Aiminus = Aiminus + p*Xi[iter-1]
                        res[iter] = ((Xi[iter]+Aiminus)/a)^b-((Aiminus/a)^b)
                        iter = iter + 1
                        }
                }
        return(res)
        }
ZARAINF = function(time,param){
        a = param[[1]]
        b = param[[2]]
        p = param[[3]]
        if(p==0){
                par = param[-3]
                res = ZRP(time,par)
                }
        else if(p==1){
                par = param[-3]
                res = ZPL(time,par)
                }
        else{
                Xi = append(diff(time,1),time[1],after=0)
                n = length(time)
                Aiminus = 0
                res = c(((Xi[1]+Aiminus)/a)^b)
                iter = 2
                while(iter<n+1){
                        Aiminus = p*(Aiminus + Xi[iter-1])
                        res[iter] = ((Xi[iter]+Aiminus)/a)^b-((Aiminus/a)^b)
                        iter = iter + 1
                        }
                }
        return(res)
        }
ZHPP = function(time,param){
        a = param[1]
        Xi = append(diff(time,1),time[1],after=0)
        res = Xi/a
        }
ZPL = function(time,param){
        a = param[1]
        b = param[2]
        n = length(time)
        tminus = c(0,time[-n])
        res = (time/a)^b-(tminus/a)^b
        return(res)
        }
################################################

################################################
#Function for plotting the accumulated number of maintenance events against time for the
     dataset
```

```
ecdfH = function(data,system,dup=FALSE,tint=0,yrange=NULL){
        if(is.na(system)){sys = data}
        else if(system==1220){sys = subset(data,data$Plant==system,drop=T)}
        else if(system==1221){sys = subset(data,data$Plant==system,drop=T)}
        else if(all(system != levels(data$System))){stop("System doesn't exist")}
        else{sys = subset(data,data$System == system, drop = T)}

        if(length(sys$Bas..start.date)>1){
                sys = sys[order(sys$Bas..start.date),] #Sort after coulumn Bas..start.date
                sys = IntOc(sys,tint)
                # If dup is TRUE we only consider 1 event per day
                if(dup==TRUE){sys = sys[!is.na(sys$IntDup),]}
                }
        # Here I check if the system has enough recorded events
        if(length(sys$Bas..start.date)<=2){
                plot(0,0,main = "Too few recorded events, <=2")
                obs=0
                N=0
                }
        #Then I calculate the cumulative registered events and plot
        else{
                tid = sys$Bas..start.date
                N = length(tid)
                obs = cumsum(c(0,rep(1,N)))
                if(tint==0){XLIM=c(0,max(tid))}
                else{XLIM=c(tint,max(tid))}
                par(mar=c(4,4,1,1)+0.1)
                plot(c(tint,tid),obs,type="l",xlim=XLIM,ylim=yrange,xlab="Days",ylab="
                    Maintenance Events",main=NULL)
                }
        return(list(sys=sys,obs=obs,N=N))
        }
#################################################

# Function for plotting the fitted NHPP models together with the maintenance against time
fitPlot = function(data,system,dup=FALSE,tint=0,yrange=NULL,legend=TRUE,Type="Max"){
        labels = c()
        labels[1]="Maintenance vs time"
        labels[2]="HPP"
        labels[3]="NHPP"
        labels[4]="RP"

        colorPL = "magenta"
        colorRP = "green"
        colorARA1 = "cyan"
        colorARAINF = "blue"
        colorHPP = "red"
        colors = c("black",colorHPP,colorPL,colorRP)

        LTYM = 2
        LTY = c(1,LTYM,LTYM,LTYM)

        if(tint==0){
                # Here we plot if tint is not present
                e = ecdfH(data,system,dup=dup,tint=tint,yrange=yrange)
                time = e$sys$Bas..start.date
                if(length(e$sys$Bas..start.date)<=2){res=e}

                else{
                        h = fitMLE(data,system,"HPP",tint,dup)
                        NTHPP = c(0,h$NT)
                        lines(c(0,time),NTHPP,type="l",col=colorHPP,lty=LTYM,pch=25) #HPP
                        f = fitMLE(data,system,"PL",tint,dup)
                        NTPL = c(0,f$NT)
                        lines(c(0,time),NTPL,type="l",col=colorPL,lty=LTYM,pch=21) #NHPP
                        r = fitMLE(data,system,"RP",tint,dup)
                        NTRP = c(0,r$NT)
                        lines(c(0,time),NTRP,type="l",col=colorRP,lty=LTYM,pch=22) #RP
                        a = fitMLE(data,system,"Ara1",tint,dup)
                        NTARA1 = c(0,a$NT)
                        ainf = fitMLE(data,system,"AraInf",tint,dup)
                        NTARAINF = c(0,ainf$NT)

                        #find out if we need to plot ARA1 or ARAinf
                        if(ainf$p!=0 && ainf$p!=1){
                                lines(c(0,time),NTARAINF,type="l",col=colorARAINF,lty=LTYM,pch
                                    =24)
                                labels = c(labels,expression("ARA"* infinity))
                                colors = c(colors,colorARAINF)
                                LTY = c(LTY,LTYM)
                                }
                        if(a$p!=0 && a$p!=1){
                                lines(c(0,time),NTARA1,type="l",col=colorARA1,lty=LTYM,pch=23)
```

```
                    labels = c(labels,"ARA1")
                    colors = c(colors,colorARA1)
                    LTY = c(LTY,LTYM)
                    }

            if(legend==TRUE){
                    legend("topleft",bty = "n",legend=labels,col=colors,lty=LTY)
                    }

            KS = list(ksHPP=h$KS,kRP=r$KS,kPL=f$KS,ksAinf=ainf$KS,ksA1=a$KS)
            res = list(fitPL = f, fitRP = r, fitAra1 = a, fitAraInf = ainf, fitHPP
                = h, sys = e$sys,KS=KS)
            }
        return(res)
        }
        #### Here we plot for t < tint and t >= tint
else{
        # I first find the subset of the system I'm interested in
        if(is.na(system)){data = data}
        else if(system==1220){data = subset(data,data$Plant==system,drop=T)}
        else if(system==1221){data = subset(data,data$Plant==system,drop=T)}
        else if(all(system != levels(data$System))){stop("System doesn't exist")}
        else{data = subset(data,data$System == system, drop = T)}

        datamin = subset(data,data$Bas..start.date<tint,drop=T)
        datamax = subset(data,data$Bas..start.date>=tint,drop=T)

        if(Type=="Min"){
        #Plot for datamin
        emin = ecdfH(datamin,system,dup=dup,tint=0,yrange=yrange)
        time = emin$sys$Bas..start.date
        if(length(emin$sys$Bas..start.date)<=2){
                fmin=0
                rmin=0
                amin = 0
                ainfmin = 0
                hmin = 0
                emin = emin
                KSmin=0
                }
        else{
        hmin = fitMLE(datamin,system,"HPP",tint=0,dup)
        NTHPP = c(0,hmin$NT)
        lines(c(0,time),NTHPP,type="l",col=colorHPP,lty=LTYM,pch=25)
        fmin = fitMLE(datamin,system,"PL",tint=0,dup)
        NTPL = c(0,f$NT)
        lines(c(0,time),NTPL,type="l",col=colorPL,lty=LTYM,pch=21)
        rmin = fitMLE(datamin,system,"RP",tint=0,dup)
        NTRP = c(0,rmin$NT)
        lines(c(0,time),NTRP,type="l",col=colorRP,lty=LTYM,pch=22)
        amin = fitMLE(datamin,system,"Ara1",tint=0,dup)
        NTARA1 = c(0,amin$NT)
        ainfmin = fitMLE(datamin,system,"AraInf",tint=0,dup)
        NTARAINF = c(0,ainfmin$NT)

        #find out if we need to plot ARA1 or ARAinf
        if(ainfmin$p!=0 && ainfmin$p!=1){
                lines(c(0,time),NTARAINF,type="l",col=colorARAINF,lty=LTYM,pch=24)
                labels = c(labels,expression("ARA"* infinity))
                colors = c(colors,colorARAINF)
                LTY = c(LTY,LTYM)
                }
        if(amin$p!=0 && amin$p!=1){
                lines(c(0,time),NTARA1,type="l",col=colorARA1,lty=LTYM,pch=23)
                labels = c(labels,"ARA1")
                colors = c(colors,colorARA1)
                LTY = c(LTY,LTYM)
                }

        if(legend==TRUE){
                legend("topleft",bty = "n",legend=labels,col=colors,lty=LTY)
                }
        }
        }
        else if(Type=="Max"){
        #Plot for datamax
        emax = ecdfH(datamax,system,dup=dup,tint=tint,yrange=yrange)
        time = emax$sys$Bas..start.date
        if(length(emax$sys$Bas..start.date)<=2){
                fmax=0
                rmax=0
                amax = 0
                ainfmax = 0
```

```
                hmax = 0
                emax = emax
                }
        else{
        t = (time−tint)
        hmax = fitMLE(datamax,system,"HPP",tint,dup)
        NTHPP = c(0,hmax$NT)
        lines(c(tint,time),NTHPP,type="l",col=colorHPP,lty=LTYM,pch=25)
        fmax = fitMLE(datamax,system,"PL",tint,dup)
        NTPL = c(0,fmax$NT)
        lines(c(tint,time),NTPL,type="l",col=colorPL,lty=LTYM,pch=21)
        rmax = fitMLE(datamax,system,"RP",tint,dup)
        NTRP = c(0,rmax$NT)
        lines(c(tint,time),NTRP,type="l",col=colorRP,lty=LTYM,pch=22)
        amax = fitMLE(datamax,system,"Ara1",tint,dup)
        NTARA1 = c(0,amax$NT)
        ainfmax = fitMLE(datamax,system,"AraInf",tint,dup)
        NTARAINF = c(0,ainfmax$NT)

        #find out if we need to plot ARA1 or ARAinf
        if(ainfmax$p!=0 && ainfmax$p!=1){
                lines(c(0,time),NTARAINF,type="l",col=colorARAINF,lty=LTYM,pch=24)
                labels = c(labels,expression("ARA"* infinity))
                colors = c(colors,colorARAINF)
                LTY = c(LTY,LTYM)
                }
        if(amax$p!=0 && amax$p!=1){
                lines(c(0,time),NTARA1,type="l",col=colorARA1,lty=LTYM,pch=23)
                labels = c(labels,"ARA1")
                colors = c(colors,colorARA1)
                LTY = c(LTY,LTYM)
                }

        if(legend==TRUE){
                legend("topleft",bty = "n",legend=labels,col=colors,lty=LTY)
                }
        }
        }
        if(Type=="MinMax"){
                KSmin = list(ksHPP=hmin$KS,kRP=rmin$KS,kPL=fmin$KS,ksAinf=ainfmin$KS,
                    ksA1=amin$KS)
                KSmax = list(ksHPP=hmax$KS,kRP=rmax$KS,kPL=fmax$KS,ksAinf=ainfmax$KS,
                    ksA1=amax$KS)
                res = list(fitminPL=fmin,fitmaxPL=fmax,fitminRP=rmin,fitmaxRP=rmax,
                    fitminAra1=amin,fitmaxAra1=amax,fitminAraInf=ainfmin,fitmaxAraInf=
                    ainfmax,fitminHPP=hmin,fitmaxHPP=hmax,sysmin=emin$sys,sysmax=emax$
                    sys,KSmin=KSmin,KSmax=KSmax)
                return(res)
                }
        else if(Type=="Min"){
                KSmin = list(ksHPP=hmin$KS,kRP=rmin$KS,kPL=fmin$KS,ksAinf=ainfmin$KS,
                    ksA1=amin$KS)
                res = list(fitminPL=fmin,fitminRP=rmin,fitminAra1=amin,fitminAraInf=
                    ainfmin,fitminHPP=hmin,sysmin=emin$sys,KSmin=KSmin)
                return(res)
                }
        else if(Type=="Max"){
                KSmax = list(ksHPP=hmax$KS,kRP=rmax$KS,kPL=fmax$KS,ksAinf=ainfmax$KS,
                    ksA1=amax$KS)
                res = list(fitmaxPL=fmax,fitmaxRP=rmax,fitmaxAra1=amax,fitmaxAraInf=
                    ainfmax,fitmaxHPP=hmax,sysmax=emax$sys,KSmax=KSmax)
                return(res)
                }
        }
}
```