Worst Case Reliability Prediction Based on a Prior Estimate of Residual Defects

Peter G. Bishop
Adelard and Centre for Software Reliability
pgb@csr.city.ac.uk

Robin E. Bloomfield
Adelard and Centre for Software Reliability
pgb@csr.city.ac.uk

Abstract

In this paper we extend an earlier worst case bound reliability theory to derive a worst case reliability function R(t), which gives the worst case probability of surviving a further time t given an estimate of residual defects in the software N and a prior test time T.

The earlier theory and its extension are presented and the paper also considers the case where there is a low probability of any defect existing in the program. For the "fractional defect" case, there can be a high probability of surviving ***any*** *subsequent time t. The implications of the theory are discussed and compared with alternative reliability models.*

Keywords: reliability prediction, reliability testing, worst case reliability bound, residual fault prediction.

1. Introduction

In previous research [2] we derived a worst case bound on the expected failure rate given that a program contained *N* residual defects and had been tested for time *T*.

In this paper we extend this theory to derive a worst case reliability function R(*t*), which gives the worst case probability of surviving a further time *t* given *N* defects and a prior test time *T*.

The paper will first summarise the original worst case bound theory and then present an extension of the theory that derives a worst case reliability function. We also consider the case of a *fractional defect* where there is a low probability of a defect existing in the program. The implications of the theory are discussed and compared with a similar theory developed by Littlewood and Strigini [7].

2. Original worst case bound theory

The observed reliability of a system containing design defects depends on the failure rates of the defects (λ_1 .. λ_N) under a given operational profile. While there are a range of methods for estimating the likely number of software defects, *N*, there is no way to establish the failure rate for unknown software defects. However the theory developed in [2] can place a worst case bound on the failure rate for all the defects based on the amount of usage time and an estimate of the number of defects. The theory makes the relatively standard reliability modelling assumptions that:

- removing a defect does not affect the failure rates of the remaining defects
- the failure rates of the defects can be represented by λ_1 .. λ_N, which do not change with time (i.e. the input distribution *I* is stable)
- The defect failure regions are disjoint so the program failure rate is the sum of the defect failure rates
- any defect exhibiting a failure will be detected and corrected immediately

The basic idea behind the model is very simple; once the software has been operating for some time, defects with the highest failure rates will be removed, while defects with low failure rates only make a small contribution to the residual software failure rate. Thus for any time *T* there is a worst case defect failure rate which maximises the residual software failure rate.

Put more formally, using the assumptions given above, a defect *i* with a perceived failure rate λ_i can survive a usage time *T* with a probability of:

$$\mathrm{P}(T_{fail} > T) = e^{-\lambda T}$$

A defect can only contribute to the future unreliability of the program if it survives, so the expected failure intensity, $\theta_i(T)$, due to defect *i* after time *T* will be:

$$\theta_i(T) = \lambda_i \; e^{-\lambda_i T}$$

Differentiating with respect to λ_i, the maximum value of $\theta_i(T)$ occurs when:

$$e^{-\lambda_i T} - \lambda_i T \; e^{-\lambda_i T} = 0$$

i.e. when:

$$\lambda_i = \frac{1}{T}$$

Substituting back, it follows that the maximum expected failure intensity of any defect *i* after the software has operated for a time *T* is:

$$\theta_i(T) \le \frac{e^{-1}}{T}$$

This result is independent of the actual failure rate of the defect, as illustrated in the Figure 1 below where the expected failure intensity is plotted for different values of λ_i.

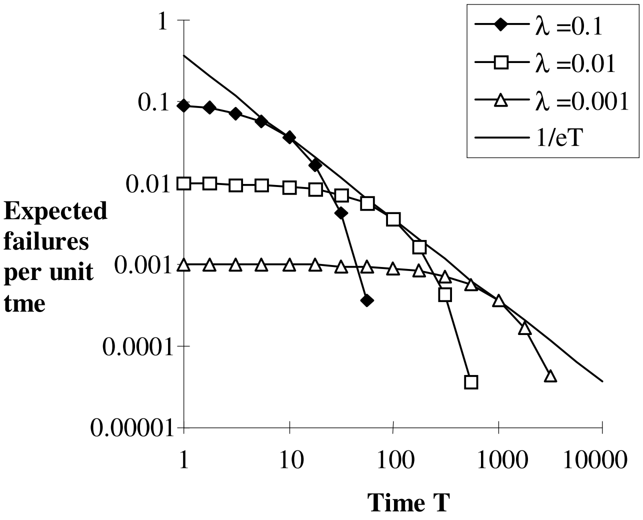

Figure 1. Illustration of the worst case bound

It is clear that, regardless of the value of λ_i, the expected failure intensity per test after *T* tests, $\theta_i(T)$, is bounded by 1/*eT*.

We can sum the worst bounds for all *N* defects to derive a worst case bound for the expected failure intensity of the whole program after time *T*, $\theta(T)$, i.e.:

$$\theta(T) < \frac{N}{eT}$$

If the model assumptions apply and we can estimate the number of defects *N* at *T*=0 (e.g. using estimation methods such as [3,6,8,9]) the reliability growth can be bounded at any future time *T*. Unlike conventional reliability models this theory takes no account of observed failures. The theory does not tell us when (or even if) the defects will be found, but it does set a quantitative bound on the failure intensity after any period of execution and correction and this bound always decreases with increasing execution time.

Note that this is the maximum *expected* failure intensity. For any specific test interval *T*, there is a finite probability that the observed failure intensity is greater than the expected value. For example, a defect with λ=4/*T* has approximately a 2% chance of surviving a test period *T.* So for 2 cases in 100 the subsequent failure intensity could be 4/*T*, and for the other 98 cases the subsequent failure intensity is zero. The expected failure intensity is the average over all cases so for a 4/*T* defect this average is approximately 0.05*T*. If the defect has the "worst case" failure rate, λ=1/*T,* then in 37% of cases (1/*e*) the subsequent failure intensity is 1/*T*, while in the remaining 63% of cases the failure intensity is zero. In this particular case the expected failure intensity is at its maximum value (1/e*T*).

So with a single defect, there is a bi-modal distribution of actual failure intensities around the expected value. However when we consider multiple defects with the same λ value, the sum of the failure intensities of the surviving defects tends to a normal distribution. If all N defects have the worst case failure rate of 1/*T*, the distribution has a mean of *N*/e*T* and a standard deviation of √*N*/e*T*. So for large *N* there is a 50% chance that the actual failure intensity is less than *N*/e*T* and a 98% chance of being less than (*N*+2√*N*)/e*T.* For other values of λ, the probability that the observed failure intensity is within the bound will be much higher. So the *actual* failure intensity is also likely to be within the bound in most cases.

3. Derivation of a worst case reliability function

To extend this theory to derive a worst case reliability function we can use the same idea of establishing a worst value for λ. For any defect with failure rate λ the chance of the defect surviving a test interval *T* is just:

$$\mathrm{P}(T_{\mathrm{det}} > T) = e^{-\lambda T}$$

and if the defect survived a test and correction interval *T*, the chance of operating for a further time *t* without failure is just:

$$\mathrm{P}(t_{\mathrm{fail}} > t) = e^{-\lambda t}$$

On the other hand, the chance of detecting and removing the defect in time *T* is:

$$\mathrm{P}(T_{\mathrm{det}} \le T) = 1 - e^{-\lambda T}$$

and in this circumstance, the probability of operating without failure for a further time t is unity.

Taking these two possible cases together, the chance of operating without failure for time t after prior usage T is:

$$\mathrm{R}(t \mid T) = (1 - e^{-\lambda T}) + e^{-\lambda T} \cdot e^{-\lambda t} \qquad (1)$$

One interesting feature of this equation is that the reliability function never falls to zero. For non-zero λ and T, the asymptote is $(1\text{–}e^{-\lambda T})$ rather than zero. This is simply because the defect may no longer be present so there is a finite chance of operating without failure until t reaches infinity.

Reducing λ has the effect of moving the asymptote closer to zero, but R(t) decreases more slowly with t. Increasing λ has the converse effect; a higher asymptote, but a more rapid fall towards it.

So there is some worst case value λ for every pair of T,t values that gives a worst case value of R(t|T) as shown by the thick line in the figure below.

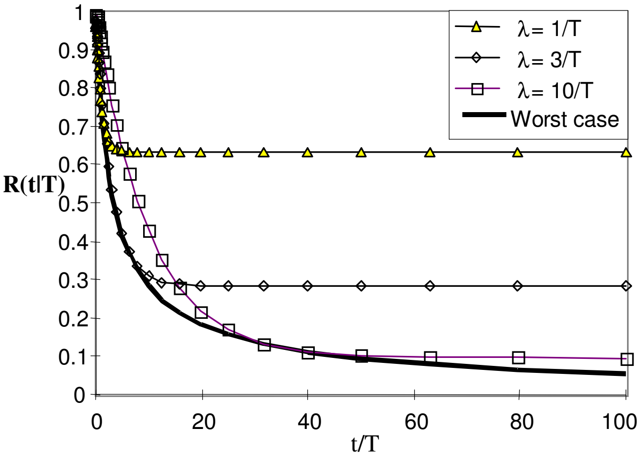

Figure 2: Long-term reliability for different λ values

We can differentiate (1) with respect to λ to find the minimum value for R(t|T). This occurs when:

$$\lambda = \frac{\ln(1 + t/T)}{t} \qquad (2)$$

Note that this is consistent with the original worst case bound theory since as t tends to zero the worst case value of λ tends to 1/T.

The worst value of λ defined in equation (2) can be substituted into (1) to derive a lower bound on the reliability function R of:

$$R(t \mid T) \geq 1 - \frac{t}{T+t} \cdot \exp(-\frac{T}{t} \ln(1 + \frac{t}{T})) \qquad (3)$$

For the case where $t << T$, ln(1 + t/T) approximates to t/T, so the reliability bound equation simplifies to:

$$R(t \mid T) \geq 1 - \frac{t}{e.(T+t)} \qquad (t << T)$$

While for the case where t >> T, the exponential term tends to unity, hence:

$$R(t \mid T) \geq 1 - \frac{t}{T+t} = \frac{T}{T+t} \approx \frac{T}{t} \qquad (\mathrm{t} >> \mathrm{T})$$

3.1. Naive exponential reliability model

By comparison, we can construct a naive reliability function using the maximum expected failure intensity at T, i.e. $\theta(T) = 1/\mathrm{e}T$. If we assume the program failure intensity remains constant with increasing t, the bound to the reliability function would be exponential, i.e.:

$$R(t|T) \geq \exp(-t / e.T)$$

For $t << T$ this approximates to:

$$R(t|T) \geq 1 - \frac{t}{eT}$$

So the naive exponential model at t=0 has the same initial slope of $-1/\mathrm{e}T$ as the worst case reliability function. The naive model is more pessimistic than the worst case reliability function for large values of t, as the worst case model tends to a reciprocal function rather than an exponential. This is illustrated in the figure below:

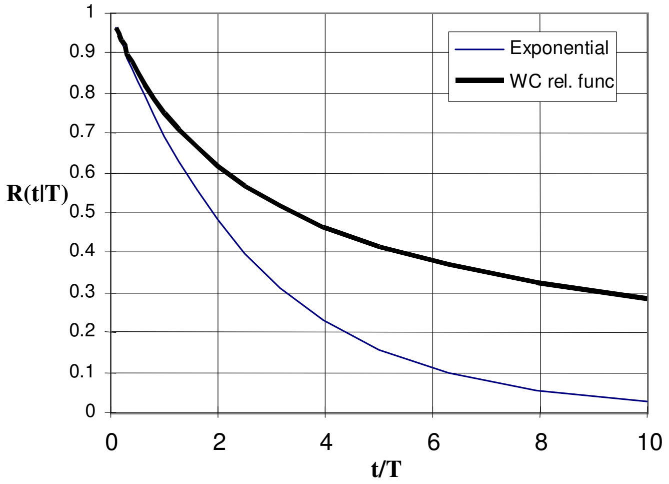

Figure 3: Naive exponential vs. worst case reliability function

For values of R(t) greater than 90%, the error in using the exponential mode is relatively small (as both are close to the linear slope approximation). For larger values of t, the error increases. For example, a 50% survival probability occurs at around $t = 2T$ on the naive exponential model but at around 3.5T using the worst case reliability model.

The naive model is more pessimistic because it merges the defect survival and removal cases and uses a mean failure intensity. As we shall see later in section 3.3, a high probability of defect-free software can have a major effect on the worst case reliability function.

3.2. Extension to N defects

So far we have only considered the reliability function for a single defect. As we have assumed that the failure regions of the N defects are disjoint, a detected failure from one defect does not affect the survival probability of other defects. So the worst case reliability function for each defect is the same as equation (3), i.e.

$$R(t\,|\,T) \geq 1 - \frac{t}{T+t} \cdot \exp(-\frac{T}{t}\ln(1+\frac{t}{T}))$$

And since R(t|T) represents the worst case survival probability for one defect at time t, the worst case reliability function for N defects at time t is simply the product of the individual reliability functions, i.e.:

$$R(t\,|\,T,N) \geq \left(1 - \frac{t}{T+t} \cdot \exp(-\frac{T}{t}\ln(1+\frac{t}{T}))\right)^{N} \quad (4)$$

which for $N{\cdot}t << T$ approximates to:

$$R(t\,|\,T,N) \geq 1 - \frac{N \cdot t}{eT} \qquad (N{\cdot}t << T) \quad (5)$$

So the operating experience T needed to achieve a given reliability target is bounded by:

$$T \leq \frac{N \cdot t}{e \cdot (1 - R(t))} \qquad (N{\cdot}t << T) \quad (6)$$

As an illustration of equation (6), if there is a reliability requirement for a 90% probability of operating for 1 year without failure, we require of prior testing and correction time T under operational conditions of up to:

$$\frac{10N}{e} \text{ years}$$

Similarly a more severe requirement of 10^{-9} failures per hour would require prior operation and correction for up to10^{9}·N/e hours.

Note these are *worst case* test intervals—the actual test interval T will be less if the defect failure rates are widely spread over many orders of magnitude. For example, the reliability growth extrapolations performed by Butler and Finelli [4] resulted in test interval predictions of around $T \approx t/(1\text{-}R(t))$.

These results for high reliability requirements are basically the same as those derived using a naive exponential reliability function. For a lower reliability requirement over a more extended time interval, we need to use the full worst case reliability equation for N defects (equation 4).

3.3. Fractional defects

For a small safety-related program, the expected value of N could be less than 1. This becomes even more likely if we are only concerned with defects that have dangerous failure modes (e.g. where we exclude defects that have failsafe behaviour or only affect non-critical functions).

Let us assume that there is a finite probability of a perfect program, P(N=0), and let us further assume that only one defect can exist in such a program, i.e.:

$$P(N{=}1) = 1 - P(N{=}0)$$

In these circumstances, the expected number of defects in the program is a *fractional* value (N_f) where:

$$N_f = 1 - P(N{=}0)$$

For example, P(N=0) = 0.9 implies that 9 out of 10 implementations will be defect-free and $N_f = 0.1$. So in 9 out of 10 programs implemented, the failure rate is always zero. In the 10th program, the defect does exist initially and there is a worst case failure rate that depends on usage time. So the expected existence probability after usage time T is:

$$(1 - P(N{=}0)) \cdot e^{-\lambda T}$$

or equivalently:

$$N_f \cdot e^{-\lambda T}$$

and hence the chance of a defect being absent after usage time T is:

$$1 - N_f \cdot e^{-\lambda T}$$

When the defect is absent, the probability of executing for time t without failure is unity for all t. Therefore the overall reliability function for a given value of λ is:

$$R(t|T,N_f) = (1-N_f\cdot e^{-\lambda T}) + N_f\cdot e^{-\lambda T}\cdot e^{-\lambda t} \qquad (7)$$

Since this only differs by a constant from equation (1) the maximum occurs at the same point, i.e. when:

$$\lambda = \frac{\ln(1+t/T)}{t}$$

And similarly when substituted, the worst case reliability function is very similar to equation (3), i.e.:

$$R(t|T) \geq 1 - N_f \cdot \frac{t}{T+t}\cdot \exp(-\frac{T}{t}\ln(1+\frac{t}{T})) \quad (8)$$

So it is clear that the expected reliability never falls below $1-N_f$.

3.4. Dealing with uncertainty in N

More generally, if there is uncertainty about our prior belief in N, we can characterise this as a probability distribution, $P(N=n)$. In this case the expected value of the reliability function is:

$$R(t\,|\,T,P(N)) = \sum P(N=n)\cdot R(t\,|\,T)^n$$

and the expected value of N is:

$$\overline{N} = \sum P(N=n)\cdot n$$

Using Jensen's Inequality [5], it can be shown that for any distribution of N:

$$\sum P(N=n)\cdot R(t\,|\,T)^n \leq R(t\,|\,T)^{\overline{N}}$$

and hence it is conservative to use the expected value for N in equation (4). So for any distribution of values of N, the worst case reliability function can be conservatively approximated by:

$$R(t\,|\,T,P(N)) \geq \left(1 - \frac{t}{T+t}\cdot\exp(-\frac{T}{t}\ln(1+\frac{t}{T}))\right)^{\overline{N}} \quad (9)$$

And for $\overline{N}\cdot t \ll T$, equation (9) can be conservatively approximated to:

$$R(t\,|\,T,P(N)) \geq 1 - \frac{\overline{N}\cdot t}{eT} \qquad (\overline{N}\,t \ll T) \qquad (10)$$

while for $t \gg T$, the equation (9) tends to:

$$R(t\,|\,T,P(N)) > \left(\frac{T}{T+t}\right)^{\overline{N}} \qquad (t \gg T) \qquad (11)$$

In addition, for $\overline{N}<1$, it is always the case that for a distribution $P(N)$:

$$P(N=0) \geq 1-\overline{N}$$

and since the expected value of $R(t) > P(N=0)$ for all t, we can also conservatively bound the worst case reliability function by:

$$R(t\,|\,T,P(N)) > 1-\overline{N} \qquad (\overline{N} < 1) \qquad (12)$$

These results show that is not necessary to define a precise distribution for N, a worst case reliability function using the expected value of N will be conservative with respect to the "proper" worst case reliability function derived using a distribution for N.

3.5. Imperfect diagnosis

The foregoing analysis assumes perfect detection and correction of software defects. This may not occur in practice. We can represent imperfect detection with an additional parameter d that represents the number of failures that occur before the defect is corrected. In these circumstances, the detection rate is λ/d, so the probability of a defect surviving for a time T is:

$$P(T_{fail} > T) = e^{-\lambda T/d}$$

While the probability of subsequent failure free operation for time t if a defect is present is unchanged, i.e.

$$P(t_{fail} > t) = e^{-\lambda t}$$

The equations are formally identical to the earlier analysis except that T is replaced by T/d. It follows that equations 2 to 12 can be generalized by replacing T by T/d. The maximum expected failure intensity at t=0 would therefore be d/eT, and for N defects the value is Nd/eT. This is consistent with the result derived in [2] for imperfect diagnosis.

This correction for imperfect detection can be applied to other reliability models, so imperfect correction is not considered when comparing the worst case reliability function against alternative models in the section below.

4. Comparison with alternative models

The worst case reliability function can be compared with the reliability function derived by Littlewood and Strigini [7] where, if zero failures are observed in time T_0, the reliability function is:

$$R(t|T_0) = T_0/(T_0+t) \qquad (13)$$

This reliability function is derived using a Bayesian analysis where there is a prior belief that all failure rates are equally likely, and this prior belief is modified by failure free testing. We will term this the "black-box Bayesian" reliability function as no effort is made to represent the actual distribution of failure rates.

By contrast, in our model we always assume each defect has a worst failure rate, and that we know how many defects exist in the software prior to testing .

Another key difference between the models is that our model can be applied *regardless of the failures* observed in time T while the T_o in the black-box Bayesian model only relates to the last failure free interval. In practice however, an estimate of $\overline{N}$=0.1 residual defects would have to be revised in the light of a failure (since we would then know that N was at least one, and possibly more). So in practice, a worst case reliability function with $\overline{N} < 1$ would require failure free working during the test interval T, i.e. $T \equiv T_0$ for $\overline{N} < 1$.

If there are no failures during T, the model parameters are identical (i.e. T in our model is the same as T_0 in the black-box Bayesian model) so it is possible to make valid comparisons between the two models.

Some sample results from the two models are shown in figure 4. It is interesting to note that for $N \leq 2$, the worst case reliability function yields a higher reliability prediction than the black-box Bayesian model for most values of t. For larger values of N (e.g. N=5), $R(t)$ is worse than the Bayesian model, but it is should be noted that for $N \geq 1$, the results might not be directly comparable as T covers the whole test period, while the Bayesian model T_0 only relates to the last failure free interval.

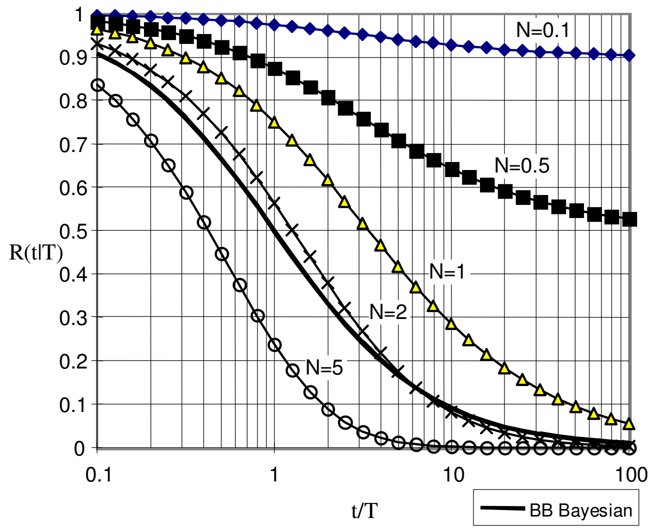

Figure 4. Worst case reliability predictions for different estimates of N vs. black-box Bayesian model

Some key values from these curves are summarised in the table below.

Table 1. Reliability vs. time for different models

Model	R(t)			
	t=0.1T	t=T	t=10T	T=100T
N_f = 0.1	0.996	0.975	0.928	0.905
N_f = 0.5	0.982	0.875	0.642	0.527
N = 1	0.965	0.75	0.265	0.55
N = 2	0.931	0.562	0.081	0.003
N = 5	0.837	0.237	0.002	4.8×10^{-7}
Bayesian	0.909	0.50	0.091	0.010

A comparison of equations (11) and (13) shows that the worst case and black-box Bayesian models tend towards the same asymptotic value for the case where N=1 and t>>T. However it can be seen in the figure that the convergence is fairly slow. The Bayesian model predicts that there is a 50% chance of surviving another interval of T (which seems intuitively reasonable) while the "worst case" reliability function predicts a 75% survival probability.

The difference is more marked if we estimate there is only a 0.1 chance of a defect (N_f=0.1). In this case the 1 probability is much greater, i.e. 97.5%. For larger values of t the difference is even greater—at t=100T the reliability bound is around 90% for the N_f=0.1 model, but a reliability of only 1% is predicted by the black-box Bayesian model.

This arises because the N_f = 0.1 model has a worst case survival probability R(t) which is asymptotic to 0.9, while the Bayesian model (and all worst cast functions where $N \geq 1$) are asymptotic to zero. This means there are very significant gains in predicted reliability if a low probability of a defect can be justified.

The “fractional defect” variant of the worst case reliability function can also be compared with the “probability of perfection” approach used by Bertolino and Strigini [1]. While [1] does not directly derive a reliability function, it shows that the probability of failure over *any* period t is always less than $1-P_{perf}$ In conventional reliability function terms, the probability of surviving any period t is therefore: $R(t) > P_{perf}$.

In our worst case reliability function model for a fractional defect, equation (12) shows that $R(t) > 1-N_f$. for all t. Since a fractional defect N_f is equivalent to $1-P_{perf}$, the asymptotes for both models are similar (but, in the Bayesian approach used in [1], P_{perf} can increase with increasing failure free operation).

It should also be noted that the worst case reliability function is consistent with the earlier worst case bound theory [2]. The initial slope of the reliability curve at t=0 gives the instantaneous failure intensity $\theta(T)$, and the value of N/eT is consistent with the original worst case bound theory.

5. Discussion

We have shown the worst case bound theory is consistent with the earlier worst case bound result [2] and is asymptotic to the reliability predictions derived in [1] and [7] in cases where the underlying assumptions are comparable. The main question to ask is whether such a worst case reliability function has any practical advantages over Bayesian methods for estimating reliability. The features of the two approaches are summarised below.

Table 2. Comparison of models

"Worst case" reliability	*Black-box Bayesian*
prediction before operation	inference from operation
postulates perfect failure detection, or adjustment for imperfect detection	exactly the same
postulates perfect fault fixing	postulates that one does not change the program at all
requires prior beliefs about N	requires prior beliefs about failure rate
given valid prior beliefs about N, there are some approximations which guarantee pessimism in the predictions	checking for pessimism is not always trivial

One merit of the worst case bound approach is that there is no necessity for prior beliefs about the distribution of possible failure rates, which is required in a Bayesian analysis. On the other hand, it is necessary to have some means for estimating N so, from a Bayesian perspective, the estimate for N is an alternative form of prior belief. There are a variety of methods for estimating N (e.g. [3,6,8,9]). The main limitation is that is difficult to get assurance that the prior estimate of N (or distribution of N) is close to the actual number of defects present in the software. This is particularly difficult when the expected value if N is predicted to be less than one as the uncertainties in the actual value become greater.

It should also be noted that, the expected value of N can be reduced if we are only concerned with a subset of the residual defects. For example, from a safety perspective, we are only interested in the number of *hazardous* defects (i.e. those that cause *dangerous* failures). In some continuous time safety-related systems, quite low percentages of hazardous defects have been observed (e.g. 3 to 10%). If this were the case then even with a prior estimate of 10 residual defects, the number of hazardous defects, N could be quite low (possibly less than one) so our model might still be less pessimistic than the black-box Bayesian model.

The instantaneous failure intensity at time T, $\theta(T)$ can be derived from the initial slope of the reliability curve and the failure intensity predictions at t=0 for the two models are shown on the table below.

Table 3. Comparison of back box Bayesian and "worst case" failure intensity predictions at *t=0*

Model	$\theta(T)$
black-box Bayesian	$1/T_0$
Worst case reliability function	N/eT

So the two models yield the same estimate when $N = eT/T_0$. So for $N < 3$, the worst case bound theory gives a less pessimistic prediction. If failures are observed in the test interval (i.e. $T_0 < T$), this would increase the difference between the two predictions (although this is based on an assumption of perfect correction of defects which may not be plausible).

These results could have been obtained using the original worst case bound theory [2[. However if requirements are expressed in terms of a lower probability of survival over a long time period we should use the full worst case reliability function (equation 4)..

The difference between the reliability functions at lower survival probabilities can be illustrated by calculating the test time required for a 50% chance of surviving 1000 time units, as shown in the following table.

Table 4. Comparison of tests needed for a 50% chance of execution without failure up to t=1000

Model	T for a 50% probability of surviving time t =1000 in operation
Worst case (N_f=0.1)	0
Worst case (N_f=0.5)	0
Worst case (N=1)	298
Worst case (N=2)	821
Black-box Bayesian	1000
Worst case (N=5)	2437

This indicates that if N is large, it would be better to base the reliability estimation on the observed failure free interval T_0 using the black-box Bayesian method. It should be noted however that the test interval T predicted by the worst case model, e.g. 2437 for 5 defects, may still be needed to achieve the desired reliability goal if there are residual defects in the software.

It can be seen that potentially the greatest gain would be when we believe there is a finite chance that the program is defect free (or at least free of dangerous defects). For example if our prior belief is that P(N=0) > 0.5, no tests are strictly necessary for a 50% survival probability. Clearly this is very sensitive to our belief about the existence of a dangerous defect—we are making a strong additional assumption that 1 out of 2 programs produced would be defect free (i.e. would never fail on demand regardless of the amount of testing). Given the uncertainties in the value of N it might be desirable to always test for the $N=1$ case even if we have a strong belief that N could be zero (i.e. the program is perfect).

As noted by other researchers (e.g. [1, 10]) the concept of “probability of perfection” can be used to “scale-up” reliability estimate made by testing. As with the other methods, the observation of a failure refutes the premise that the program is defect-free. In our method there is no standard way of updating the prior estimate for N if testing shows our prior estimate is invalid (i.e. when the number of defects detected exceeds the predicted value).

If we could employ Bayesian methods to update the prior distribution for N based on defects are detected during testing (and perhaps the expected distribution of failure rates), we would be less vulnerable to errors in the initial estimate for N. This is an area that merits further research

6. Conclusions and further work

We have derived a worst case reliability function that is relatively easy to apply and results in less pessimistic reliability predictions than our earlier theory [2] and can be less pessimistic than the black-box Bayesian reliability equation [7] if the predicted number of defects, N, is small.

We also show that for the case where the expected value of N<1, i.e. where there is a finite chance that there are no dangerous defects, there can be dramatic increases in predicted reliability. This “fractional defect” parameter is closely related to the “probability of perfection” parameter used [1], and both result in reliabilities that are asymptotic to a non-zero value.

Strictly speaking we do not need to believe in complete perfection if we are only concerned with dangerous defects—we only need to believe it is possible that there are no *dangerous* defects, i.e. the expected number of dangerous defects is less than unity.

While such “probability of perfection” scaling is an attractive concept, especially for the ultra-high reliability area, it is difficult to construct convincing arguments to justify that belief. This could be a fruitful area for further research.

Another limitation of the worst case model is that the estimate of N is fixed prior throughout testing. Clearly the predicted worst case bound is invalid if more than N defects are detected in practice. It would therefore be fruitful to generalise the approach so that a prior distribution of N values is updated in the light of operational experience.

7. Acknowledgements

This research was undertaken for the UK Nuclear Industry C&I Research Programme under Scottish Nuclear (now British Energy Generation UK) contract PP/114163/HN. The paper reporting the research was produced under the EPSRC research interdisciplinary programme on dependability (DIRC).

8. References

1. A. Bertolino and L.Strigini, "Assessing the risk due to software faults: estimates of failure rate vs evidence of perfection", Software Testing, Verification and Reliability, vol. 8, no.3, pp.155-166, 1998.
2. P.G. Bishop, R.E. Bloomfield, “A Conservative Theory for Long-Term Reliability Growth Prediction”, IEEE Trans. Reliability, vol. 45, no. 4, pp 550-560, Dec. 1996.
3. P.G. Bishop, “Estimating Residual Faults from Code Coverage”, SAFECOMP 2002, 10-13 September, Catania, Italy, 2002.
4. R.W. Butler and G.B. Finelli, “The Infeasibility of Experimental Quantification of Life-Critical Software Reliability”, in ACM SIGSOFT ’91 Conference on Software for Critical Systems, in ACM SIGSOFT Software Eng. Notes, Vol. 16 (5), New Orleans, Louisiana, pp.66-76, 1991.

5. J.L.W.V. Jensen, “Sur les fonctions convexes et les inégalités entre les valeurs moyennes,” Acta Mathematica, pp. 175-193, Uppsala, Sweden, 1906.
6. T.M. Khoshgoftaar and J.C. Munson, ‘Predicting software development errors using complexity metrics”, IEEE J of Selected Areas in Communications, 8(2), pp. 253-261, 1990.
7. B. Littlewood and L. Strigini, ‘Validation of Ultra-High Dependability for Software-based Systems”, Communications of the ACM, 36 (11), pp.69-80, 1993.
8. Y.K. Malaiya and J. Denton, ‘Estimating the Number of Residual Defects”, HASE' 98, 3rd IEEE Int' l High-Assurance Systems Engineering Symposium, Maryland, USA, November 13-14, 1998.
9. K. Yasuda, ‘Software Quality Assurance Activities in Japan”, Japanese Perspectives in Software Engineering, pp. 187-205, Addison-Wesley, 1989.
10. J.M. Voas, C.C. Michael, and K.W. Miller ‘Confidently Assessing a Zero Probability of Software Failure”. High Integrity Systems. 1, 3,.pp. 269-275, 1995.