# Software fault-freeness and reliability predictions

Lorenzo Strigini and Andrey Povyakalo

Centre for Software Reliability,
City University London, U.K.
{L.Strigini, A.A.Povyakalo}@city.ac.uk
http://www.csr.city.ac.uk

**Abstract.** Many software development practices aim at ensuring that software is correct, or fault-free. In safety critical applications, requirements are in terms of probabilities of certain behaviours, e.g. as associated to the Safety Integrity Levels of IEC 61508. The two forms of reasoning – about evidence of correctness and about probabilities of certain failures – are rarely brought together explicitly. The desirability of using claims of correctness has been argued by many authors, but not been taken up in practice. We address how to combine evidence concerning probability of failure together with evidence pertaining to likelihood of fault-freeness, in a Bayesian framework. We present novel results to make this approach practical, by guaranteeing reliability predictions that are conservative (err on the side of pessimism), despite the difficulty of stating prior probability distributions for reliability parameters. This approach seems suitable for practical application to assessment of certain classes of safety critical systems.

**Keywords:** Correctness, survival probability, conservative bounds, software safety standards

## 1 Introduction

For critical applications of computers, it is important to demonstrate low enough likelihood that design faults (in particular software faults) cause, in operation, failures with severe consequences. A form of evidence that can support such claims is experience from either previous operation, in the same conditions as those for which the demonstration is needed, or "operational" testing that reproduced those conditions. Such evidence is often not offered in a suitable form.

Other forms of evidence are usually provided, often generically called "process" (or indirect) evidence: that methods believed to reduce the risk of defects were applied in development and verification and validation; and that the faults of concern are indeed likely to be absent (e.g., certain properties have been formally proved, stringent inspection or testing methods failed to detect faults, any fault revealed and considered important was fixed). Such process precautions are required by most standards for safety-critical (and security-critical) software.

Indeed, the process evidence required by a standard for a given criticality of the software's functions is often the *only* evidence brought to support a claim

that the software will fail with acceptably low probability. But this use of the evidence is not supported by current software engineering knowledge [1, 2]. The methods documented are considered by their proponents to reduce the *likelihood of faults*; but we have really no scientific bases for claiming that a specific set of precautions will achieve a *failure rate*, or a probability of failure per demand, below a specific threshold as required by the system of which the software will be a part (e.g. a SIL level in IEC61508 [3]). Even if these methods do reduce the number of faults, fault numbers or fault densities are not sufficient to enable estimation of a failure rate or *pfd*.

There is a use of "process" evidence that *could* directly support claims of low probability of failure: as supporting belief in *absence of faults*. The goal of the process precautions is to avoid faults; and for products that are inherently simple, this goal might be achieved. Software about which a verifier concludes, by detailed analysis and/or proof, that it is correct, sometimes *is* correct. A fault-free software product has zero failure probability, over any duration of operation. We cannot generally claim to know that a software product is fault-free with certainty; but we could bring convincing evidence about a probability of it being so, and this probability may be high enough to help in proving that the risk in operation is low enough. A formal introduction to this approach and more complete arguments in its favour were given many years ago [4]; we return to it to propose a concrete approach to its application. Advantages of reasoning this way would include:

- the probability of $pfd = 0$ is a lower bound on the software's probability of failure-free behaviour over any arbitrarily long period of operation (a serious advantage when making predictions for long-lived systems).
- while probabilities of failure per demand, or failure rates, depend on the frequencies of the various demands in the environment of use (the operational profile) of the system, a claim about probability of absence of faults would accompany a product to each new use for which the range of demands and the required responses are the same. Such relatively environment-independent claims would for instance be useful for the practice of "Safety Element out of Context" described in ISO26262 [5].

Many different words are used for properties similar to what we discuss, saying e.g. that the software is "correct", or "free from faults" (or "from defects"), or "perfect". We choose the term "fault-freeness". Independently of the name used, to avoid logical fallacies one needs to apply this term carefully according to the context. We are interested in safety critical, software-based items; then, we will mean by "software faults" those that would cause behaviours that violate safety requirements when the software is used in the context of interest; and by "failures" those with respect to these safety requirements. Similar restrictive definitions could be applied for the case of software that is critical for security. Different definitions will apply in different contexts. For instance, a subcontractor may wish instead to demonstrate that the software it delivers satisfies the written specifications in its contract, irrespective of whether they are correct, and thus define "faults" and "failures" with respect to these specifications.

Assessments of the probability of operating for a period of time without safety failures are formally reliability predictions (reliability with respect to that subset of possible failures). In our mathematical treatment, we assume a system for which the reliability parameter is a probability of failure per demand (*pfd*); our approach can easily be extended to systems for which the reliability model is an exponential, continuous-time model, with a failure rate as its parameter.

In the rest of this paper, Section 2 examines how to use a claimed probability of fault-freeness $P_p$ towards claims of actual interest, namely probability of failure-free operation (reliability) over prolonged operation (possibly a system's lifetime). Section 3 discusses how to integrate via Bayesian inference evidence from failure-free operation to improve the claimed reliability; it shows how to avoid the crucial difficulty of choosing a prior distribution and obtain predictions that are guaranteed to be conservative (not to err on the side of optimism). Section 4 positions our contribution with respect to other past and ongoing work on related approaches. Last, Section 5 examines how a claim of a certain $P_p$ can be supported, and addresses the crucial issue that absolute certainty of fault-freeness can never bee achieved, even for a product that is indeed fault-free. The last section discusses the value of the reported results and future work.

## 2 Reliability predictions using a probability of fault-freeness

An advantage of reasoning with claims of fault-freeness is that they define lower bounds on long-term reliability, irrespective of the use (demand profile) to which the item will be subjected.

Reliability predictions based on a claimed probability of fault-freeness take a simple form: given a probability $P_p$ of fault-freeness, the reliability of the item at any future time $t$, $R(t)$, satisfies $R(t) \geq P_p$. Thus, in particular, being able to claim a reasonably high $P_p$ is a desirable option for systems with an operational life of many demands but that will not receive operational testing over a comparable number of demands. For instance, let us imagine a system with an intended lifetime of 10,000 (statistically independent) demands. To be 90% sure that it will not suffer any accident due to the software, we would need to demonstrate $pfd \leq 10^{-5}$; but we would get at least the same confidence if we could claim a 90% probability of fault-freeness.

We expect that claims based solely on probability of fault-freeness would not be accepted in many application domains: users, regulators and the general public would want to know some bound on the probability of failure for the case that the software *has* faults. But such confidence bounds on the *pfd* can be obtained from past operation or operational testing.

For instance, if we had 90% confidence that an item of avionic software has no faults such as to cause catastrophic failure, this by itself satisfies the regulatory requirement that catastrophic failures due to this equipment be "not anticipated to occur during the entire operational life of all airplanes of one type", "usually expressed" as "probability on the order of $10^{-9}$ or less" per

flight-hour [6] (the often-quoted "$10^{-9}$" requirement which has been forcefully argued to be infeasible to demonstrate statistically [1, 7]). Yet, the possibility that, in the unlikely (10% probability) case of faults being present, such faults cause a high probability (say 1%) of failure per flight, would probably seem unacceptable: some evidence would be required that even if faults are present the *pfd* would still be low. But this evidence would not need to demonstrate the $10^{-9}$ requirement. A much more modest claim, together with the probability of fault-freeness, would ensure a forecast of low enough risk during the early life of the aircraft type; and as operation continues, failure-free operation would increase the likelihood that the software is indeed fault-free, or if not, that its probability of failure is indeed very small.

We proceed to discuss this combination of evidence from operation or operational testing with probability of fault-freeness.

## 3 Inference from operation or testing

We examine now how to improve a reliability claim that includes probability of fault-freeness by adding evidence from operation or testing, if no failures (of the failure types of interest) have been observed.

Bayesian inference from operational testing is well understood. The unknown *pfd* is seen as a random variable, which we will call $Q$, with a *prior* probability density function $f_Q(q)$. After observing success on $t_{past}$ independent demands, the *posterior* probability of surviving $t_{fut}$ further demands is:

$$R\left(t_{fut}|t_{past}\right) = \frac{\int_0^1 (1-q)^{t_{past}+t_{fut}} f_Q\left(q\right) dq}{\int_0^1 (1-q)^{t_{past}} f_Q\left(q\right) dq} \tag{1}$$

According to the previous discussion of probability of fault-freeness, the prior distribution for the unknown *pfd* has the form

$$R\left(t_{fut}|t_{past}\right) = f_Q\left(q\right) = P_p\,\delta\left(q\right) + \left(1-P_p\right) f_{Qn}\left(q\right) \tag{2}$$

where $\delta(q)$ is Dirac's delta function; $f_{Qn}\left(q\right)$ is itself a probability density function, for the random variable "value of the system *pfd* conditional on *pfd*$> 0$'.

After observing $t_{past}$ failure-free demands, the posterior reliability is

$$\begin{aligned} &\frac{\int_0^1 \left( (1-q)^{t_{past}+t_{fut}} \left(Pp\,\delta\left(q\right) + \left(1-Pp\right) f_{Qn}\left(q\right)\right)\right) dq}{\int_0^1 \left( (1-q)^{t_{past}} \left(Pp\,\delta\left(q\right) + \left(1-Pp\right) f_{Qn}\left(q\right)\right)\right) dq} \\ &\quad = \frac{P_p \;+\; \left(1-P_p\right) \int_0^1 (1-q)^{t_{past}+t_{fut}} f_{Q_N}\left(q\right) dq}{P_p \;+\; \left(1-P_p\right) \int_0^1 (1-q)^{t_{past}} f_{Q_N}\left(q\right) dq} \end{aligned} \tag{3}$$

We can describe the operation of Bayesian inference as reducing the values of the probability density function more for those values of *pfd* that are less likely to be true in view of the observed failure-free operation. Thus seeing no failures reduces the values of the probability density function for high values of *pfd*, and shifts probability mass towards the origin (towards $pfd = 0$).

### 3.1 Worst case prior distributions and worst-case reliability

A common problem in applying Bayesian inference is choosing a prior distribution for the unknown *pfd*, $Q$. Arguing from the *pfd* values observed for similar software will be difficult: there has been no systematic data collection activity that would allow this. "Expert judgement" tends then to be used. But all scientific evidence is that experts' judgement of probabilities tends only to be good for phenomena of which they have actual experience of prediction followed by feedback about its accuracy: textbook examples of observed good probability prediction ability are weather forecasters and horse-racing bookmakers, who on a daily basis assign probabilities to events and very soon observe whether the event occurs or not. On this basis, for an expert in critical software to be accurate in assigning a probability density function for a product's *pfd* would be unusual. We can expect an expert's direct experience to include comparatively few products, hardly any examples of *pfd* close to 1, and for those with high reliability, insufficient information to judge their true value of *pfd*.

But for safety it is usually acceptable to demonstrate *pessimistic* predictions. We can then look for a *worst case* prior distribution that one could assume for the inference. We can show that such a worst case does exist, as follows. Consider a probability density function for the unknown *pfd*, $Q$, made of two probability masses: a mass $P_p$ in $Q = 0$ and a mass $(1 - P_p)$ in $Q = q_N$. Now, if we assume $q_N$ to be close to either end of the interval $[0, 1]$, reliability predictions after observing failure-free demands will be very high. Indeed, in the two limiting cases, predicted reliability will be 1: if $q_N = 1$, one test is enough to show that $P(Q = q_N) = 0$ and thus $Q = 0$ with certainty; and if $q_N = 0$, $Q = 0$ with certainty to start with.In between these extreme values, successful tests will reduce $P(Q = q_N)$ and increase $P(Q = 0)$, but still leave a non-zero probability of future failure. Thus, posterior reliability as a function of $q_N$ is highest at the two ends of the interval, and must have a minimum somewhere in between. [1]

A proof of existence of this worst-case prior distribution of *pfd* has two steps:

1. as a consequence of the Lemma proved in Appendix A, of all the prior distributions with a probability mass in $Q = 0$, the worst-case one is indeed a two-point distribution as above

$$P_s \, \delta(q) + (1 - P_s) \, \delta(q - q_N) \tag{4}$$

[1] These considerations highlight another important point: a prior that is *pessimistic* in terms of the reliability it implies may produce *optimistic* inference. Here, moving $q_N$ closer to 1 implies, before failure-free operation, pessimism: a system likely to fail in few demands from the start of operation. But then observing it *not* failing over even few demands then logically makes it very likely to have 0 *pfd* (optimism). Which prior distributions will produce pessimistic posteriors depends both on which posterior prediction we wish to minimise (e.g. posterior reliability for $t_{fut}$ demands vs posterior probability of fault-freeness) and on the specific observations (here, the number of failure-free demands). It would thus be wrong to take from the worst-case posterior distribution we obtain here any measure different from posterior reliability for $t_{fut}$ demands, e.g. a certain percentile, or a posterior probability of fault-freeness, and believe it to be a conservative value for use in further claims about this system.

so that the posterior reliability (3) has the form

$$\frac{P_p \ + \ (1 - P_p) \ (1 - q_N)^{t_{past}+t_{fut}}}{P_p \ + \ (1 - P_p) \ (1 - q_N)^{t_{past}}} \tag{5}$$

2. among all such two points distribution, we can identify a value of $q_N \in (0, 1)$ that yields the lowest posterior reliability. Thus Bayesian inference can be applied on the basis of only $P_p$ and $N$, removing the major obstacle of assessing the whole $f_{Qn}(q)$ distribution.

Fig. 1a shows worst case posterior reliability as a function of the ratio $t_{fut}/t_{past}$, for various values of the prior probability of fault-freeness, $P_p$. We can read this figure in various ways:

- given a certain amount of observed failure-free operation, the worst-case reliability predicted for a comparable amount in the future is satisfactorily higher than $P_p$. So, for instance, given a prior probability of fault-freeness $P_p = 0.5$ for a safety critical system in — say — a car model, after observing failure-free operation of a car fleet using that system for one year, the worst-case probability of failure-free operation for another year, for constant fleet size, is above 80%. Given $P_p = 0.9$, it would be more than 95%. However, as the prediction extends further and further into the future, the statistical evidence becomes less and less adequate for confident prediction, and the worst-case reliability asymptotically falls back to $P_p$ as $t_{fut}$ tends to infinity;
- given the horizon $t_{fut}$ over which we want to predict reliability, the plot shows the number of $t_{past}$ observations that we need to reach for the worst-case prediction to hit our intended target. For instance, expecting a safety protection system to have to face 100 demands over its lifetime, $P_p = 0.9$ and statistical testing over 1000 demands will give probability of going through this lifetime without failures upwards of 95%. More detail for scenarios with very high required worst-case reliability, and extensive operational testing, is given in Fig. 1b; the $y$-axis represents the probability of at least one failure over $t_{fut}$ demands. If the number of test demands is much greater than the number of demands over the intended operational lifetime[2], even modest values of $P_p$ give substantial confidence in failure-free operational life.

A special case of reliability prediction is the reliability for $t_{fut} = 1$, i.e. the system *pfd*. Fig. 1d shows the number $t_{past}$ of failure-free demands one needs to observe to achieve a desired value for the worst case posterior *pfd*.

## 4 Related work

In computer science there has been for a long time an opinion sector opposing the very idea of software reliability assessment, on the grounds that software can be made, and thus ought to be made, correct: 100% reliable.

[2] An example of current interest concerns plans for testing of the protection system of the European Pressurised Reactor `http://www.hse.gov.uk/newreactors/reports/step-four/final-res-plans/resolution-plan-gi-ukepr-ci-02.pdf`

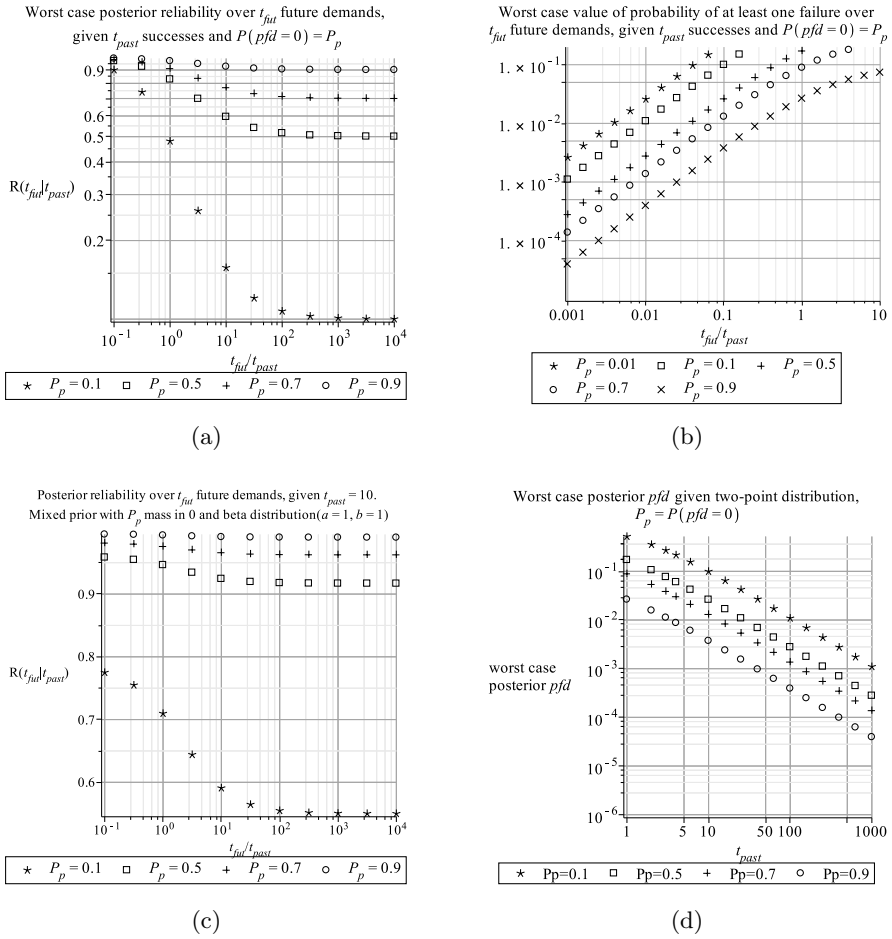


Fig. 1

Several authors [8, 4, 9] have argued for the usefulness of estimating a probability of correctness. Voas and co-authors [8] argued that given a lower bound on the *pfd* that any one fault can produce ("testability"), one can infer from failure-free testing a probability of correctness. Bertolino and Strigini published the inference procedure for this case [10, 11]; however, one cannot usually demonstrate a lower bound on the probability of failure; it is hard to demonstrate experimentally that an event that is very unlikely to start with, and never observed, is actually impossible. Thus the treatment used here, producing a worst-case prediction without such assumptions, is a major step towards practical applicability.

The approach of accepting an incomplete description of prior *pfd* distributions, and obtaining conservative predictions by finding – among those prior distributions that match this description – the one that, combined with the actual

observations, would yield the worst posterior value for some reliability measure has been studied: (1) for reliability given a mean *pfd* [12]; (2) for the mean posterior *pfd*, given, about the prior distribution of the *pfd*, (2a) a confidence level on an upper bound [13], or (2b) some combinations among a probability of $pfd = 0$, a confidence level on an upper bound and a certain upper bound [14].

Littlewood and Rushby showed how claims of "perfection" can aid assessment of fault-tolerant systems made of diverse redundant components [15, 16].

## 5 Evidence for fault-freeness and "quasi-fault-freeness"

Confidence in fault-freeness depends on "process" evidence being backed by arguments about why that process produces fault-free products with high enough probability. One might for instance reason that:

1. the current product is obtained by a process by which the same organisation produced a series of previous products (similar in their general requirements and complexity), that have proved to be fault-free; or
2. the verification steps that show this product to be correct (e.g., proof) are known by experience to catch a certain, high percentage of all faults, and together with estimates of the fault numbers to be expected before verification, this yields a probability of having *no* faults [17, 18, 19].

There are important difficulties. The relevance of past experience – whether the current product is somehow an anomaly in the "population" of the past products considered – is never certain. This is just the underlying difficulty of all statistically-based prediction, and we need not discuss it for this specific case. But here, the past experience itself is ambiguous: in argument 1 above, we cannot know *with certainty* that such past products were fault-free, but at most that they were scrutinised in many ways, operated for a long time without failures or problems that would cast any doubts on their correctness, and so on. As for argument 2, it assumes that in past experience we reached certainty about the number of faults in a product, and again such absolute certainty is impossible.

We outline here how this difficulty can be overcome in principle. Regarding for instance case 1 above, we consider that, if a past product successfully underwent stringent scrutiny and a long operational life, we cannot declare it fault-free with certainty, *but* it has a posterior distribution of *pfd* where most of the probability mass is either in 0 of close to it. The Lemma in appendix A shows that for worst-case reliability prediction, such a distribution can be substituted with a single probability mass in a point $q_S$ close to 0. Similar experience for multiple similar products will also give confidence, say a probability $P_s$, that the same property applies to the current product. So, for the current product, we can use a pessimistic probability density function of *pfd* similar to equation 4:

$$P_s\,\delta\,(q - q_S) + (1 - P_s)\,\delta\,(q - q_N) \tag{6}$$

(where $q_N$ accounts for the possibility that the *pfd* of the current system is *not* as low as that of the previous systems, and could be much worse); and find

a worst-case value of $q_N$, as in section 3.1. For $q_S$ close enough to 0 (in view of $t_{past}$ and $t_{fut}$), again the confidence $P_s$, together with worst case reasoning, could give useful reliability predictions, guaranteed to be conservative.

We have outlined this solution as a chain of reasoning steps: obtaining confidence in low *pfd*s of past products; then confidence that similar confidence in a low *pfd* applies to the new product; then using the latter for worst-case inference. This entire chain could instead be formalised in a single Bayesian model, in which all similar products have *pfd* distributions with parameters belonging in their turn to a common distribution, about which the operation of each product gives some information. Such models have been studied, e.g. [20]. A natural next step in this research is to apply them to the current problem, and check the feasibility of their use in concrete assessment and certification contexts.

## 6 Discussion and conclusions

We have shown a way of using the evidence usually collected for assurance about safety critical systems, together with experience from operation or realistic testing, to achieve reliability predictions that can be proved to be conservative.

This relies on (i) using the process evidence to support a claim of "probability of fault-freeness"; (ii) applying Bayesian inference from the observation of failure-free operation and (iii) given strong uncertainty about the prior distributions to use, applying worst-case reasoning.

This approach reduces the impact of important difficulties with the current ways of stating quantitative claims about software failure in critical applications: the lack of scientific bases for deriving claims about *pfd* from the kinds of favourable evidence usually produced about such software; the difficulty of achieving enough operational or test experience to demonstrate very high reliability over long lifetimes; last, the difficulty of choosing convincing prior distributions for Bayesian inference is obviated by the ability to do worst case inference.

The predictions thus obtained will not always be as high as desired, for at least two possible reasons: (i) the evidence may simply not be strong enough (not enough operational experience, not strong enough prior probability of fault-freeness) to warrant as high a predicted long-term reliability as we seek; (ii) these methods are intentionally biased towards conservatism; they avoid the risk of erroneous optimism by accepting potential errors in the direction of pessimism. Of course, choosing to err in the direction of pessimism is a two-edged sword; it avoids dangerous errors but may make the prediction useless. By way of comparison, we show in Fig. 1c the posterior reliability obtained by assuming, together with a certain probability $P_p$ of fault-freeness, that the *pfd*, if not zero, has a uniform distribution. This distribution might indeed be chosen as an "ignorance prior", when one does not know what prior to believe, and seem reassuringly conservative because *before observing failure-free demands*, it is indeed very pessimistic: it means that if there are faults the expected *pfd* is 0.5. But this prior conservatism is deceptive. It implies that observing failure-free demands very quickly builds up confidence in future reliability: comparing Fig. 1c with Fig. 1a

(and noting the different vertical scales in the two plots) shows how far this assumption is from being conservative. Thus, if one has strong reasons to support a specific prior distribution conditional on non-fault-freeness, ($f_{Qn}(q)$), by all means one should use it; but our worst case reasoning will illustrate how much of the predicted reliability hinges on believing that specific prior.

When performing inference using prior probability of fault-freeness (or of "quasi-fault-freeness"), failure-free operation gradually builds up confidence (increases predicted reliability). But observing even a single failure will radically undermine this confidence: the posterior probability of *pfd*=0 becomes zero; not being sure about our prior distribution for *pfd* when not zero ($f_{Qn}(q)$), and wishing to be pessimistic, we must conclude that the *pfd* is indeed very high. Some may object to this apparent "fragility" of the approach. We contend that it is an advantage: it represents correctly the way confidence is gained for many critical products. Indeed, if a product that was reputed to be practically immune from design faults suffers a possibly design-caused failure, a normal reaction is to take it out of service, find the design fault and fix it (creating a new product and a new prior distribution of *pfd*); or demonstrate that the failure was not due to a design fault; or that the design fault exists but brings an acceptably low *pfd* in operation. In any case, the previous argument is discarded when the failure undermines the belief in a *pfd* so low that the probability of seeing any failure is also low. Our Bayesian formalisation faithfully represents this effect.

We strongly believe that this approach can improve the way that critical software-based systems are assessed. However, we acknowledge that we advocate the use of general evidence about the effectiveness of development methods that is not widely available. For the time being, this approach may be useful to organisations with strong internal data collection processes: they may well have enough evidence to build arguments that they will consider sound for their own risk assessments, or might support a claim made to a client or a safety regulator. A safety regulator may use our kind of worst-case reasoning to compare against the predictions that it has to judge in order to approve or reject a claim that a system is safe enough for operation. A company for which wrongly optimistic reliability predictions bring large economic risks (e.g. an automobile manufacturers, for which a doubt of a possible safety-critical fault may lead to massive recalls) may use this approach to assess its own risk, both prior to deployment and at any point in the operational life of its products.

A straightforward extension of this approach is to the case of continuous reliability with the unknown parameter being a failure rate $\lambda$ instead of a *pfd*.

Another important question for further investigation is whether more complex Bayesian models, taking into account — for instance — experience in comparable products as in [20] can prove useful in practice, as suggested in Section 5, to ensure sound inference from "quasi-fault-freeness" of past products.

**Acknowledgements** This research was supported in part by the Leverhulme Trust via project UnCoDe ("Uncertainty and confidence in safety arguments: effect on expert decision makers") and by the SeSaMo project ("Security and Safety Modelling"), supported by the Artemis JU, and the United Kingdom

Technology Strategy Board (ID 600052). Our colleagues Bev Littlewood and David Wright contributed useful criticism to previous versions of this work.

## References

[1] B. Littlewood and L. Strigini, "Validation of ultra-high dependability for software-based systems," *CACM*, vol. 36, no. 11, pp. 69–80, 1993.
[2] B. Littlewood and L. Strigini, "'validation of ultra-high dependability...' - 20 years on," *Safety Systems, Newsletter of the Safety-Critical Systems Club*, May 2011.
[3] (IEC) International Electrotechnical Commission, "IEC 61508, functional safety of electrical/ electronic/programmable electronic safety related systems."
[4] 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.
[5] ISO, "ISO 26262 road vehicles – functional safety," 2011.
[6] FAA, "Federal aviation regulations far 25.1309," Advisory Circular AC 25.1309-1A, Federal Aviation Administration, 1985.
[7] R. Butler and G. Finelli, "The infeasibility of quantifying the reliability of life-critical real-time software," *IEEE TSE*, vol. 19, no. 1, pp. 3–12, 1993.
[8] J. Voas, C. Michael, *et al.*, "Confidently assessing a zero probability of software failure," *High Integrity Systems*, vol. 1, no. 3, pp. 269–275, 1995.
[9] W. Howden and Y. Huang, "Software trustability analysis," *ACM TOSEM*, vol. 4, no. 1, pp. 36–64, 1995.
[10] A. Bertolino and L. Strigini, "On the use of testability measures for dependability assessment," *IEEE TSE*, vol. 22, no. 2, pp. 97–108, 1996.
[11] A. Bertolino and L. Strigini, "Acceptance criteria for critical software based on testability estimates and test results," in *SAFECOMP'96*, pp. 83–94, Springer, 1996.
[12] L. Strigini, "Bounds on survival probabilities given an expected probability of failure per demand," DISPO2 Project Technical Report LS-DISPO2-03, Centre for Software Reliability, City University London, July 2003.
[13] B. Littlewood and A. Povyakalo, "Conservative bounds for the pfd of a 1-out-of-2 software-based system based on an assessor's subjective probability of 'not worse than independence'," CSR Technical Report, City University London, 2012.
[14] P. Bishop, R. Bloomfield, *et al.*, "Toward a formalism for conservative claims about the dependability of software-based systems," *IEEE TSE*, vol. 37, no. 5, pp. 708–717, 2011.
[15] B. Littlewood, "The use of proof in diversity arguments," *IEEE TSE*, vol. 26, no. 10, pp. 1022–1023, 2000.
[16] B. Littlewood and J. Rushby, "Reasoning about the reliability of diverse two-channel systems in which one channel is 'possibly perfect'," *IEEE TSE*, vol. 38, no. 5, pp. 1178 – 1194, 2012.
[17] T. Shimeall and N. Leveson, "An empirical comparison of software fault tolerance and fault elimination," *IEEE TSE*, vol. 17, pp. 173–182, 1991.
[18] B. Littlewood, P. Popov, *et al.*, "Modelling the effects of combining diverse software fault removal techniques," *IEEE TSE*, vol. SE-26, no. 12, pp. 1157–1167, 2000.
[19] R. Bloomfield and S. Guerra, "Process modelling to support dependability arguments," in *DSN 2002, International Conference on Dependable Systems and Networks*, (Washington, D.C., USA), IEEE Computer Society, 2002.

[20] B. Littlewood and D. Wright, "Reliability prediction of a software product using testing data from other products or execution environments," DeVa Project Technical Report 10, City University London, 1996.

## Appendix A: General lemma

If the prior probability density function of the *pfd* of a system is a mixture of probability density functions $f_{Qi}$, then substituting any subset of these component distributions with a set of single-point probability masses, one for each of the $f_{Qi}$ thus substituted, will yield a pessimistic prediction of posterior reliability after observing failure-free demands.

Formally: let random variable $Q$ have probability density function (*pdf*)

$$f_Q(q) = \sum_{i=1}^{n} p_i f_{Q_i}(q) dq, \tag{7}$$

where $\int_0^1 f_{Q_i}(q)dq = 1;\ p_i > 0, i = 1..n$ ; and $\sum_{i=1}^n p_i = 1$.

Then, for any two natural numbers $t_{past}$ and $t_{fut}$,

$$\frac{\sum_{i=1}^n p_i \int_0^1 (1-q)^{t_{past}+t_{fut}} f_{Q_i}(q)dq}{\sum_{i=1}^n p_i \int_0^1 (1-q)^{t_{past}} f_{Q_i}(q)dq} \geq \\ \frac{p_1(1-q_1)^{t_{past}+t_{fut}} + \sum_{i=2}^n p_i \int_0^1 (1-q)^{t_{past}+t_{fut}} f_{Q_i}(q)dq}{p_1(1-q_1)^{t_{past}} + \sum_{i=2}^n p_i \int_0^1 (1-q)^{t_{past}} f_{Q_i}(q)dq}, \tag{8}$$

(where without loss of generality we have substituted the single component distribution $f_{Q_1}$; the l.h.s. of (8) is the posterior reliability), the value of $q_1$ is

$$q_1 = 1 - \left( \int_0^1 (1-q)^{t_{past}} f_{Q_1}(q)dq \right)^{\frac{1}{t_{past}}},\ i = 1..n, \tag{9}$$

and the bound (8) is attained if $f_{Q1}(q) = \delta(q - q_1)$, where $\delta(x)$ is Dirac's delta function.

### Proof

By Hölder's inequality,

$$\int_0^1 (1-q)^{t_{past}} f_{Q_i}(q)dq \leq \left( \int_0^1 (1-q)^{t_{past}+t_{fut}} f_{Q_i}(q)dq \right)^{\frac{t_{past}}{t_{past}+t_{fut}}},$$

i.e.

$$\int_0^1 (1-q)^{t_{past}+t_{fut}} f_{Q_i}(q)dq \geq \\ \left( \int_0^1 (1-q)^{t_{past}} f_{Q_i}(q)dq \right)^{\frac{t_{past}+t_{fut}}{t_{past}}} = (1-q_i)^{t_{past}+t_{fut}}, \tag{10}$$

and (10), together with substituting (9) in (8), proves the lemma. **QED**.