Demonstrating Software Reliability using Possibly Correlated Tests: Insights from a Conservative Bayesian Approach

Kizito Salako[a], Xingyu Zhao[b,c]

[a]*Centre for Software Reliability, City, University of London, Northampton Square, London EC1V 0HB, United Kingdom*
[b]*Department of Computer Science, University of Liverpool, Ashton Stree, Liverpool L69 3BX, United Kingdom*
[c]*Warwick Manufacturing Group, University of Warwick, Lord Bhattacharyya Way, Coventry CV4 7AL, United Kingdom*

ABSTRACT

This paper presents Bayesian techniques for conservative claims about software reliability, particularly when evidence suggests the software's executions are not statistically independent. We formalise informal notions of "*doubting*" that the executions are independent, and incorporate such doubts into reliability assessments. We develop techniques that reveal the extent to which independence assumptions can undermine conservatism in assessments, and identify conditions under which this impact is not significant. These techniques – novel extensions of *conservative Bayesian inference* (CBI) approaches – give conservative confidence bounds on the software's failure probability per execution. With illustrations in two application areas – nuclear power-plant safety and autonomous vehicle (AV) safety – our analyses reveals: **1)** the confidence an assessor should possess before subjecting a system to operational testing. Otherwise, such testing is futile – favourable operational testing evidence will eventually decrease one's confidence in the system being sufficiently reliable; **2)** the independence assumption supports conservative claims sometimes; **3)** in some scenarios, observing a system operate without failure gives less confidence in the system than if some failures *had* been observed; **4)** building confidence in a system is very sensitive to failures – each additional failure means significantly more operational testing is required, in order to support a reliability claim.

Abbreviations

pfd probability of failure per demand

pfe probability of failure per execution

pfm probability of fatality-event per mile

AV Autonomous Vehicle

CBI Conservative Bayesian Inference

NHPP non-homogeneous Poisson process

PK Prior Knowledge

SCS Safety-Critical System

SRGM software reliability growth model

Notation

T_i Random variable that indicates when the ith execution fails

X The unknown probability of an execution failing

Λ The unknown probability of the next execution failing when the last execution was a failure.

✉ k.o.salako@city.ac.uk (K. Salako); xingyu.zhao@liverpool.ac.uk (X. Zhao)
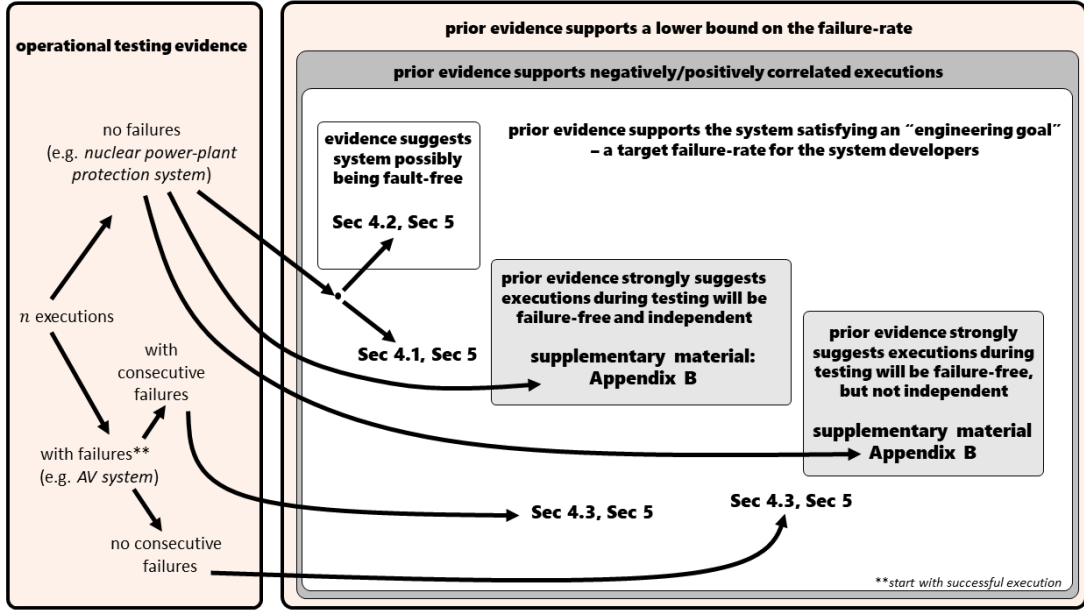ORCID(s): 0000-0003-0394-7833 (K. Salako); 0000-0002-3474-349X (X. Zhao)

Figure 1: various assessment scenarios and the sections in this paper that treat them. Each path through the diagram, starting at the "n executions" node on the left, indicates a system's behaviour (during operational testing) and the implications of reliability evidence considered by an assessor (before operational testing).

ϵ A bound on X representing an "engineering goal"

θ Prior confidence in the engineering goal being satisfied

b A required upper bound on X

c Posterior confidence in X satisfying the bound b

ϕ_1 Prior confidence in negatively dependent executions

ϕ_2 Prior confidence in positively dependent executions

n The total number of executions

s The number of executions that are failures

r The number of failures that are preceded by a failure

L The likelihood function

$\mathcal{R}$ The domain of the Klotz failure model

1. Introduction

It is prudent to be conservative when assessing a software-based *safety-critical system* (SCS), since software failure could significantly harm stakeholders in the system. Rigorous statistical arguments can give support for conservative claims about whether the software is sufficiently reliable, where such claims are based on evidence of achieved levels of reliability. In particular, Bayesian methods provide a natural formalism/calculus for combining various forms of reliability evidence, resulting in probabilistic measures that (given the evidence) articulate one's uncertainty about the reliability of the SCS (see Atwood et al. (2003)); in particular, the reliability of its software. Examples of evidence that can be utilised in Bayesian approaches for reliability assessment include the observed failure behaviour of (similar) software during past operation (e.g., see Thomas E. Wierman et al. (2001), Bunea et al. (2005) and Pörn (1996)).

When using statistical arguments in assessments, a central question is whether the system's software has statistically *independent and identically distributed* (i.i.d.) "executions". By "an execution" we mean *a set of actions performed by the software that can be regarded as a unit of software operation*[1]. For example, actions performed in response to *each* demand/input the software receives from its environment, or actions in response to *a sequence* of inputs (received over a unit amount of time or distance). In this paper, when a software execution occurs, either all of the actions performed[2] are the required actions or at least one of these actions is incorrect – i.e., a software execution is either correct or a failure. Our focus is the assessment of the system's software, so we consider only software failures (so, no hardware failures) as defining system failure.

Executions that are i.i.d. make sense for some systems, such as an on-demand system where demands rarely occur, and the system's state and operational environment are reset inbetween demand occurrences. But there are often reasons to doubt the i.i.d. assumption. An *autonomous vehicle* (AV) could experience sudden changes in driving conditions that make it very likely for the AV to make a series of consecutive mistakes; or an airplane (and its flight control systems) can be put under increasing operational stresses when they encounter aggressive weather mid-flight (with turbulent "air pockets"); and "failure clustering" has been observed in various (control) systems. In many situations, at least *some* doubt about independent executions is warranted.

Even when executions *are* assumed i.i.d., SCS software will typically be required to exhibit *no failures* over a large number of executions. An assessor might (must?) consider whether these successful executions *are* positively correlated after all, and might account for this possibility in assessments. Because, at face-value, an assumption of i.i.d. executions can seem quite strong – it significantly limits an assessor's hypothesis about which probabilistic laws could characterise a software's failure process. Consequently, one might suspect that assuming independence results in optimistic reliability claims – it's useful to ask whether this is *actually* the case. Are assessments significantly undermined by assuming independence?

The answer depends on: **i)** how reliable the software actually is, **ii)** the sequence of the software's successes/failures during operational testing, and **iii)** the nature of any dependence between executions. Prior to testing, the assessor is uncertain about (i)–(iii), and is reliant on reliability evidence to shape their beliefs about how reliable the system is. Operational testing provides more evidence that refines these beliefs further.

This process – of an assessor's uncertainties being initially shaped by evidence obtained prior to operational testing, and then further shaped by the system's performance during testing – is formalised in this paper in *conservative Bayesian inference* (CBI) terms, for an assessor making claims about the system's probability of failure per "*execution*" (*pfe*). An example *pfe* is the *probability of failure per demand* (*pfd*) for an on-demand system, and another example is the *probability of a fatality-event per mile* (*pfm*) for an AV; we consider both examples later in the paper. Our primary interest is in assessing software reliability – specifically, solving constrained optimisation problems, to obtain the least confidence an assessor can justifiably have in a system's *pfe* being "small enough".

CBI makes explicit how (i)–(iii) above affect an assessor's uncertainty under various assessment scenarios. Fig. 1 summarises these scenarios and indicates sections in this paper where the scenarios are treated. Prior to testing, an assessor may express some confidence in, say, **i)** the system being sufficiently reliable (e.g. confidence in the unknown *pfe* being smaller than some target value set for the system developers); **ii)** the system being fault-free; **iii)** future executions being negatively or positively dependent, and being failure-free. An assessor uses execution outcomes during testing – such as *no* failed executions occurring, or *some* failures separated by runs of successes occurring – to update their confidence.

Summary of the paper's contributions

1) extending CBI techniques that allow an assessor to quantify the potential negative impact of invalid statistical modelling assumptions on reliability claims (e.g., when software executions are assumed i.i.d. when, in actuality, they are not). See section 4;

2) showing how the outcomes of software executions – whether failures occurred and whether these were clustered or isolated – significantly affects how confident an assessor can (justifiably) be of a system being sufficiently reliable (sections 4, 5);

---

[1]This is also known as a "*run*" (e.g., see Strigini & Littlewood (1997)).

[2]Not performing any action could be the required action.

3) several closed-form solutions for conservative posterior confidence in an upper bound on *pfe* (see Fig. 1, sections 4, 5 and the supplementary material);

4) illustrating these findings in two scenarios: nuclear power-plant and autonomous vehicle (AV) safety (section 4). We give advice and caution for assessors/practitioners, concerning how confident they should be before embarking on operational testing (sections 4, 5).

The rest of the paper is organised as follows. Related work is detailed in section 2, while section 3 reviews CBI and the Klotz model for correlated executions. Formalisations of doubting i.i.d. executions are given in section 4, and used to derive conservative confidence bounds on system *pfe* under various scenarios. The sensitivity of these bounds to changes in model parameters is studied in section 5. Section 6 discusses results, and section 7 concludes the paper.

2. Related Work

The current paper directly continues our development of statistical techniques for conservative reliability assessment first reported in Salako & Zhao (2023). Consequently, the related works detailed in that paper continue to be relevant here; we highlight these works in this section.

2.1. Why is Modelling Correlated Executions Necessary?

An early model for sequences of statistically independent executions, used in works on random testing, is due to Thayer et al. (1978) (see Duran & Ntafos (1984) for an application of the model). However, reasons to expect correlated failed executions in various systems became well-known. For example, a system can exhibit "*failure clustering*" due to the system receiving sequences of inputs that cause the system to fail, where such inputs cluster into subsets of the system's failure region[3] – see Ammann & Knight (1988) and Bishop (1993). The system's operational environment generates input sequences as trajectories (within the set of all inputs) that eventually enter into, and linger in, these failure regions. This phenomenon motivated developing assessment approaches that account for positive failure correlation between executions – see Csenki (1993); Tomek et al. (1993); Huang et al. (2021).

Strigini (1996) gives other reasons for correlated executions; e.g. if the software's internal state is corrupted upon an initial failed execution, making subsequent executions more likely to fail. Or, if the system's operational environment becomes increasingly more stressful (i.e. there's an increasing probability of trajectories in the input space entering the failure region).

2.2. Statistical Models of Correlated Executions

A number of models with Markov dependence have been proposed for correlated executions. The binary Markov chain of Chen & Mills (1996); the Markov renewal process of Goseva-Popstojanova & Trivedi (2000) (that builds upon earlier work in Csenki (1993), Tomek et al. (1993)); and the Bondavalli et al. (1995) model that captures benign-failures, and the cumulative impact of such failures when assessing iterative software. Bondavalli et al. (1997) improve on this model, demonstrating the model's use with fitted steady-state and transition probabilities.

None of the aforementioned models are demonstrated using inference methods that explicitly account for one's uncertainty about whether the executions are i.i.d. or not. Nor do these models provide demonstrably conservative statistical support for reliability claims about software – where such support is justified by various forms of reliability evidence (in addition to the outcomes of *possibly* correlated executions during operational testing).

2.3. Conservative Bayesian Methods for Assessments

A number of studies have applied Bayesian methods to support software reliability assessment, e.g. Miller et al. (1992); Singh et al. (2001); Littlewood et al. (2002); Popov (2013). The utility of these methods is in the inference process. An assessor's beliefs about the reliability of a system are initially formed by evaluating relevant evidence. Then these beliefs are updated, upon seeing how the system performs during operation.

The usual challenge with Bayesian methods is the need to characterise one's initial (i.e. "*prior*") beliefs as a prior probability distribution – a distribution that captures all, and only all, of one's prior beliefs. Care must be taken when eliciting a prior distribution; an unrepresentative prior could lead to overly pessimistic, or dangerously optimistic, assessments.

---

[3]A software component's failure region is a geometrical, or mathematically topological, characterisation of those inputs that trigger the software to fail.

For reliability assessments, there is the added challenge that prior distributions often represent beliefs about continuous random variables, such as an on-demand system's unknown *pfd*. Requiring that an assessor specify beliefs about the *infinitely many* ranges of possible *pfd* values is often impractical.

CBI methods have been developed to address these challenges. CBI is related to *robust Bayesian analysis* which studies the sensitivity of the results of Bayesian inference to changes in the inference inputs – see Berger (1994, 1990); Lavine (1991); Berger & Moreno (1994). Inputs such as: the prior distribution; the statistical model that determines the likelihood function; and the posterior measure of interest. An assessor may not be able to use available evidence to fully specify a prior distribution, but the evidence may allow a much more limited *partial* specification of a prior – e.g. the assessor expects the prior, whatever it may be, to satisfy certain quantiles or moments. By considering *all* of those prior distributions that satisfy these specifications, CBI determines the most conservative inference result (from using these priors) to give support for a claim that the system is sufficiently reliable. This is one way in which conservatism in assessments is realised via Bayesian inference.

Bishop et al. (2011) introduced the CBI idea. A number of studies soon followed, applying CBI in various contexts. For example, **i)** in Strigini & Povyakalo (2013), Povyakalo *et al.* use CBI to obtain the smallest probability of the system's next *m* executions being successful, given prior evidence that the system is very reliable and its last *n* executions were successful; **ii)** in Zhao et al. (2015), with evidence to support some confidence in the system possibly being fault-free, and some confidence in the system being very reliable, Zhao *et al.* use CBI with operational testing to conservatively gain confidence in the system possibly being fault-free; **iii)** Salako (2020) bounds the reliability of a binary classifier, given evidence that the classifier's past performance was (un)likely; and **iv)** in Zhao et al. (2019), Flynn *et al.* apply CBI to the problem of assessing AV safety – highlighting circumstances under which attempts to demonstrate the required levels of safety via road testing are in vain. More CBI applications to assessing SCSs are found in Zhao et al. (2015, 2017, 2018, 2019).

These applications all involve "univariate" priors; i.e., distributions of a single unknown, typically the system *pfe*. More recently, CBI applications have involved "bivariate" priors. Littlewood et al. (2020) consider the assessment of a system in a "new" situation – either the system replaces an older system in a given operational environment, or the system has been deployed in a new environment after operating in a previous environment for some time. They demonstrate how CBI supports dependability claims, when evidence suggests the system's failure propensity in the new situation is "no worse" than the propensity in the "old" situation. Zhao et al. (2020) study "improvement arguments" of this kind in the context of assessing AV safety – but with different dependability measures of interest and a more general failure model for the system. While Salako et al. (2021) consider more general "improvement arguments" for an even wider range of assessment scenarios.

In Salako & Zhao (2023), we introduce a CBI technique for incorporating doubts about the i.i.d. assumption into conservative reliability claims. The statistical model used was the first CBI model to capture correlated executions, and is based on the Klotz (1973) model – a binary Markov chain that predates and agrees with Chen & Mills (1996) and Goseva-Popstojanova & Trivedi (2000). We illustrated how assessments where i.i.d. executions are assumed can be very optimistic. That paper was concerned with assessing on-demand systems where (despite the software containing faults) no software failures are observed during extensive operational testing. The current paper significantly extends this CBI approach, to apply to a wider range of assessment scenarios – e.g., scenarios where some failures are observed during extensive testing, and where the assessor can justify only very weak beliefs about the unknown *pfe*.

2.4. Assessing Continuously Operating Software

The current paper focuses on assessment scenarios where one observes the software's success/failure behaviour on each of a number of "unit" software operations – i.e., on each execution. Example scenarios include assessing on-demand systems (see PD IEC TR 63161:2022 (2022); Rausand (2014)). Hence, we employ "discrete-time" statistical models (i.e., *Bernoulli processes*) and our reliability measure of interest is *pfe*.

Contrastingly, for assessments where the reliability measure of interest is the software's failure-rate in continuous time, "continuous-time" statistical models are more appropriate (e.g., *non-homogeneous Poisson processes* (NHPPs)).

Many *Software reliability growth models* (SRGMs) – useful in predicting future reliability for continuously operating software (see Lyu (1996); Xie (1991); Musa et al. (1987)) – have been developed over the years; Singpurwalla & Wilson (1994), Miller (1986) and Bergman & Xie (1991) give good overviews of early SRGMs.

3. Preliminaries: a CBI Model of Correlated Executions

3.1. A Review of CBI

Consider the following scenario from Zhao et al. (2019). An on-demand system is subjected to operational testing, to determine if its *probability of failing on a randomly occurring demand* (i.e. *pfe*) is acceptably low. Let X be this unknown *pfe* – i.e. on a random demand, the system fails (with probability X) or succeeds (with probability $1 - X$). Demands occur randomly according to an *operational profile*, see Musa (1993).

Before operational testing, an assessor might have sufficient evidence to fully specify a *prior distribution* representing their beliefs about which values of X are likely to be the true value, and which values aren't. During operational testing, the system correctly responds to all n demands that occur – these successes are assumed to occur in an "*independent and identically distributed*" (i.i.d.) manner. If the value of X is x then the probability of observing these successes is $L(x; n) = (1 - x)^n$. Let b be a required upper bound on X. The assessor's confidence (after seeing the successes) in X being no larger than b is:

$$P(X \leqslant b \mid n \text{ successes}) = \frac{P(X \leqslant b, \, n \text{ successes})}{P(n \text{ successes})} = \frac{\mathbb{E}[L(X; n)\mathbf{1}_{X \leqslant b}]}{\mathbb{E}[L(X; n)]} \tag{1}$$

where $\mathbf{1}_S$ is an indicator function—it equals 1 when predicate S is true, and 0 otherwise.

Often, there isn't enough evidence to fully justify a specific prior distribution for (1), but there may be enough to justify weaker constraints on the prior (such as a few quantiles). We refer to such constraints as *prior knowledge* (PK). A basic form of PK is:

Prior Knowledge 1 (certainty in a lower bound). 100% *confidence in the system's pfe not being lower than p_l; i.e.,* $P(X \geqslant p_l) = 1$.

X is a probability, so p_l should be non-negative. p_l can be 0 (see Littlewood & Rushby (2012) for possibly "fault-free" software) or a very small number (e.g. the best reliability feasible for the system given current levels of technology).

Prior Knowledge 2 (confidence in satisfying an engineering goal). $\theta \times 100\%$ *confidence in the system's pfe being better than, or equal to, an upper bound ϵ; i.e.,* $P(X \leqslant \epsilon) = \theta$.

ϵ is an "engineering goal": a target *pfe* value that system developers try to achieve. ϵ is typically chosen to be much smaller than the required bound b, so $\epsilon \leqslant b$. While θ is how confident the assessor is, before operational testing, that the engineering goal *has* been achieved. θ has to be large enough to support conducting operational testing; reducing the chance that unreliable systems use up the operational testing budget.
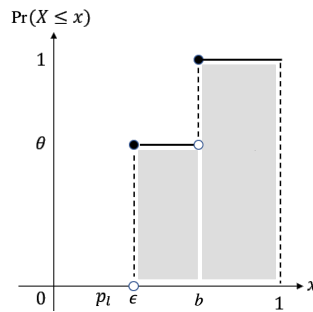

Figure 2: A conservative prior cumulative distribution function. Illustration from Salako et al. (2021).

The following theorem shows one can conservatively gain confidence in a bound b on X (see Zhao et al. (2019)).

Theorem 1 (univariate CBI). *Let $\mathcal{D}_u$ be the set of all probability distributions over the interval $[0, 1]$. Using (1), the optimisation problem*

$$\inf_{\mathcal{D}_u} P(X \leqslant b \mid n \text{ executions without failure})$$

$$s.t. \quad PK1, \quad PK2$$

is solved by Fig. 2's prior distribution: using this prior, $P(X < b \mid n$ executions without failure) equals the infimum[4].

Insight 1 (The basic CBI idea). One considers the set of all feasible priors that satisfy an assessor's PKs. For a given posterior measure of interest (e.g. posterior confidence in a bound on X), CBI determines a prior that gives the most pessimistic value for this measure – no feasible prior gives a more pessimistic value, and any prior that does must violate at least one PK. The CBI prior is referred to as a "worst-case" prior.

3.2. A Model of Correlated Software Executions

The following stochastic failure process for software exhibiting correlated executions – used in Salako & Zhao (2023) – is based on the Klotz (1973) model. A sequence of Bernoulli random variables $T_1, \dots, T_n$, each take on the values 1 or 0; indicating failure or success, respectively, on each of n software executions. Similar to section 3.1, let x be the unconditional *probability the next execution is a failure (pfe)*. Let λ be the probability that *a failure is followed by another failure*. That is,

$$P(T_i = 1) = 1 - P(T_i = 0) = x, \qquad\qquad i = 1, \dots, n$$
$$P(T_i = 1 \mid T_{i-1} = 1) = \lambda, \qquad\qquad i = 2, \dots, n$$

If the process is *1st-order stationary*, we obtain the Markov model in Fig. 3 (see Klotz (1973), Salako & Zhao (2023)).
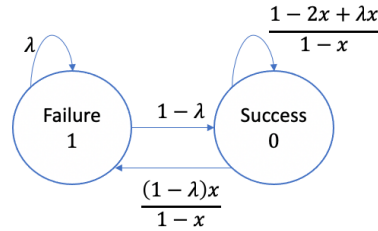

Figure 3: The Klotz model for Bernoulli trials with dependence. Illustration from Salako & Zhao (2023).

This stochastic process is well-defined if the transition probabilities lie between zero and one. That is, if

$$0 \leqslant x < 1, \quad \max\{0, (2x - 1)/x\} \leqslant \lambda \leqslant 1 \tag{2}$$

Inequalities (2) define the subset $\mathcal{R}$ of the unit square (see Fig. 4a).

Remark 1 (correlation in the Klotz model). The correlation coefficient for two successive executions is $\frac{\lambda - x}{1-x}\mathbf{1}_{0 \leqslant x < 1} + \mathbf{1}_{x=1}$. This defines 3 correlation types: **i)** when $x = \lambda$, the model exhibits independent execution outcomes; **ii)** when $\lambda > x$, execution outcomes tend to cluster more often (e.g. bursts of failures) – positive correlation; and **iii)** when $\lambda < x$, execution outcomes tend to alternate more often, between failure and success – negative correlation.

Over n executions, suppose the software makes: α transitions from a successful execution to a failed execution, β transitions from "success" to "success", γ from "failure" to "failure", and δ from "failure" to "success". So, $\alpha + \beta + \gamma + \delta + 1 = n$. Under the Klotz model, for given (x, λ), the probability of observing these transitions is the likelihood $L(x, \lambda; \alpha, \beta, \gamma, \delta)$:

$$L(x, \lambda; \alpha, \beta, \gamma, \delta) = \begin{cases} x\left(\frac{(1-\lambda)x}{1-x}\right)^{\alpha}\left(1 - \frac{(1-\lambda)x}{1-x}\right)^{\beta}\lambda^{\gamma}(1 - \lambda)^{\delta}; \\ \text{when the 1st execution is a failure} \\[2mm] (1 - x)\left(\frac{(1-\lambda)x}{1-x}\right)^{\alpha}\left(1 - \frac{(1-\lambda)x}{1-x}\right)^{\beta}\lambda^{\gamma}(1 - \lambda)^{\delta}; \\ \text{when the 1st execution is a success} \end{cases} \tag{3}$$

---

[4]To obtain this solution, one constructs sequences of feasible priors (where each subsequent prior in a sequence gives an increasingly worse value for the objective function) that converge "*almost everywhere*" to a prior that gives the infimum. This prior differs from the feasible priors converging to it at $X = b$. As a consequence, it is the value of $P(X < b \mid \dots)$ from this prior, rather than $P(X \leqslant b \mid \dots)$, that is the infimum.

During operational testing, an assessor observes α, β, γ and δ. But both x and λ are unknown to the assessor[5]; in particular, the assessor is uncertain about the *pfe* X. So, upon observing n executions of the system, an assessor's confidence in X being no larger than a bound b is:

$$P(X \leqslant b \mid \text{ outcomes of n executions}) = \frac{P(X \leqslant b, \text{ outcomes of n executions})}{P(\text{ outcomes of n executions})} = \frac{\mathbb{E}[L(X, \Lambda; \alpha, \beta, \gamma, \delta)\mathbf{1}_{X \leqslant b}]}{\mathbb{E}[L(X, \Lambda; \alpha, \beta, \gamma, \delta)]} \quad (4)$$

which generalises (1). It will be useful to refer to s – the number of failed executions – and r – the number of failed executions preceded by a failed execution ("consecutive failures", for short). The likelihood (3) can be re-expressed in terms of n, s and r, which are related to $\alpha, \beta, \gamma, \delta$ (see supplementary material, A.1). In particular, $r = \gamma$.

4. Conservative Upper Confidence Bounds on Probability of Failure per Execution

4.1. Practical Context for Applying CBI techniques

Upon observing n executions, what is the least confidence an assessor should have about X being "no bigger than" the bound b? We determine this for different scenarios by deriving the greatest lower bound for (4) using PKs 1, 2, 3, 4. Section 3.1 introduced PKs 1, 2, while section 4.2 introduces PKs 3, 4.

Practical implications of these results – with domain-specific PK parameterisations for nuclear power-plant safety protection systems and AV safety subsystems – are given in sections 4.3 and 4.4 respectively. The parameterisations are illustrative of plausible PK values when assessing functionally redundant software components for safety systems employing fault-tolerance – e.g., systems highlighted in Wood et al. (2010); Koopman & Wagner (2016); Hörwick & Siedersberger (2010); Sha (2001).

These CBI techniques/results – for conservatively gaining confidence in the software being sufficiently reliable – are primarily intended for use in assessments based on operational/statistical testing, where the software is treated as a black-box. In operational/statistical testing, the test cases for the software are randomly generated software inputs. To do this correctly, the probability of generating a given test case must be consistent with an "*operational profile*" – i.e., this probability must be the same as the probability of the same "case" occurring when the software is deployed in real operation. Test cases can take different forms, depending on the type of software under study. For example, consider a batch program that receives numerical values for all of its input variables at the start of an execution, it executes, and then it produces all of its outputs. A test case for such a program could be a fixed-length vector of numerical values – each number in the vector is the value for an input variable. Alternatively, consider a control program that receives multiple numerical values for each input variable over time; here, a test case could be a collection of numerical sequences – each sequence represents the changing values over time for an input variable. More examples of test cases can be found in Lyu (1996) and Strigini & Littlewood (1997).

Statistical testing supports direct estimates of reliability, for the purposes of reliability assessment and product acceptance. It also supports decisions on whether software is ready for use in a specific system. Thus, CBI techniques can be applied in any software development testing phase where statistical testing may be applied, and where decisions may be taken on whether software is ready to be deployed; e.g., integration or acceptance testing.

Of course, good practice for carrying out statistical testing must be followed when employing our theorems and results; e.g., Strigini & Littlewood (1997) give detailed guidance in this regard. See also Salako & Zhao (2023) for more discussion. Best practice approaches for expert belief elicitation should be followed to elicit the PKs; e.g., PRA Working Group (1994); O'Hagan et al. (2006).

Table 1 lists the worst-case priors used for the curves in the example plots. While these priors are consistent with those in Salako & Zhao (2023) (for assessors that can justify continuous marginal prior distributions of X), the current priors are applicable in many more scenarios (e.g., when assessors can justify only very limited properties of the marginal prior distribution of X, in the form of PKs 1 and 2).

4.2. Assessment with Doubts about i.i.d. Executions

Our first assessment scenario is a baseline. Before operational testing, an assessor uses reliability evidence to justify the engineering goal PK 2, and the following two PKs about the independence assumption (*cf.* Remark 1):

---

[5]In theory, knowing these values would allow the assessor to completely characterise the system's failure process; e.g., one may then compute the probability of future failure-free operation over a sequence of software executions.
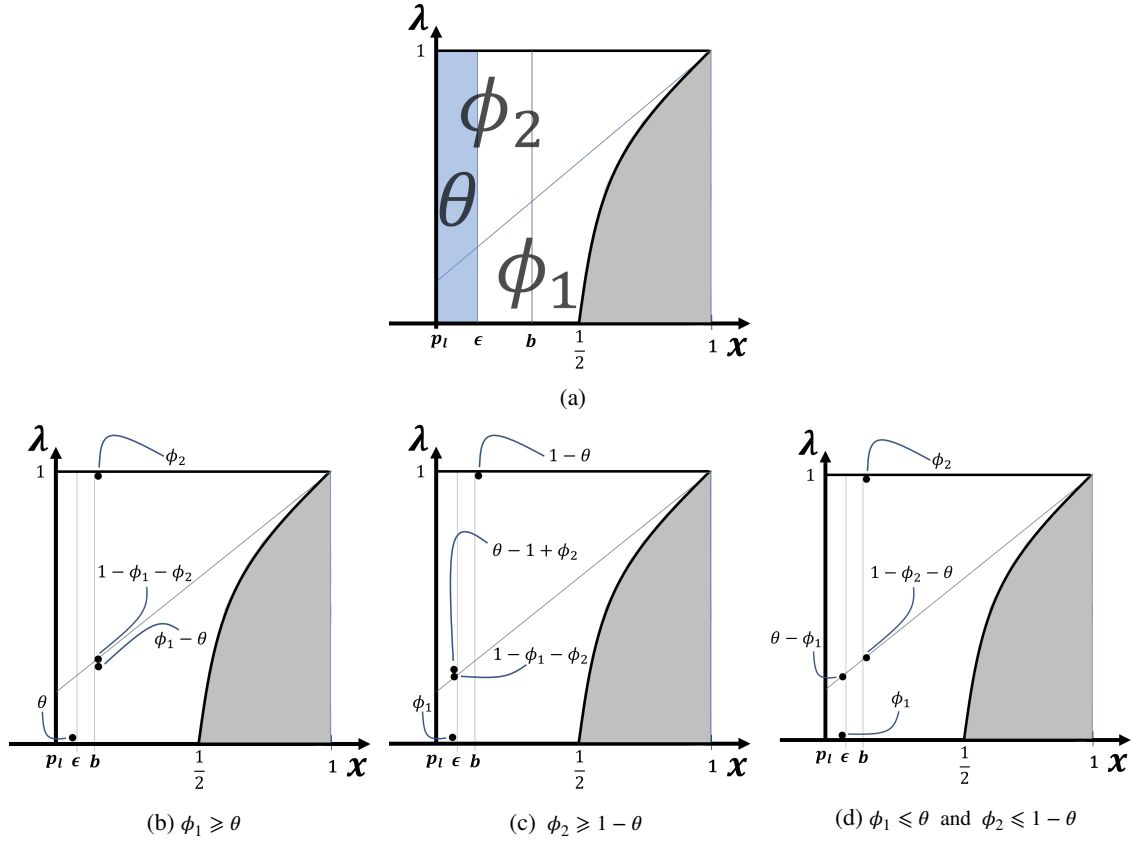
(a)

(b) $\phi_1 \geqslant \theta$

(c) $\phi_2 \geqslant 1 - \theta$

(d) $\phi_1 \leqslant \theta$ and $\phi_2 \leqslant 1 - \theta$

Figure 4: The support $\mathcal{R}$ of the Klotz likelihood function, and the subsets of $\mathcal{R}$ related to PKs 1, 2, 3 and 4, are shown in subfig. 4a. Upon observing executions with **no failures**, subfig.s 4b, 4c and 4d are 3 prior distributions that solve the optimisation problem in Theorem 2. These priors are relevant for the ranges of parameter values indicated in each subfigure.

Prior Knowledge 3 (confidence in negative dependence). $\phi_1 \times 100\%$ *confidence in successive software executions having negative dependence; i.e.,* $P(\Lambda < X) = \phi_1$.

Prior Knowledge 4 (confidence in positive dependence). $\phi_2 \times 100\%$ *confidence in successive software executions having positive dependence; i.e.,* $P(\Lambda > X) = \phi_2$.

Consequently, the assessor's *prior confidence in independence* is $(1 - \phi_1 - \phi_2)$, i.e. $P(\Lambda = X) = 1 - \phi_1 - \phi_2$. Note that in all of the remaining theorems in this paper, $\phi_1 = \phi_2 = 0$ is the special case of i.i.d. execution outcomes – in this limit, all the theorems agree with previously published univariate CBI results.

An assessor observes all n executions during testing are successful. Using the Klotz model, the CBI problem of determining the least amount of confidence the assessor can justifiably have, about the system *pfe* satisfying bound b, is the following constrained optimisation problem. Consider the support $\mathcal{R}$ of the Klotz likelihood, defined by (2) and depicted in Fig. 4a. Let $\mathcal{D}$ be the set of all prior probability distributions over $\mathcal{R}$, and $0 \leqslant p_l \leqslant \epsilon < b < \frac{1}{2}$. Then, the following theorem holds (generalisation proved in supplementary material, A.2).

Theorem 2. *Using* (3) *and* (4)*, the optimisation problem*

$$\inf_{\mathcal{D}} P(X \leqslant b \mid n \text{ executions without failure})$$

$$s.t. \quad PK1, \ PK2, \ PK3, \ PK4$$

is solved by the prior distributions in Fig.s 4b, 4c and 4d, since $P(X < b \mid n \text{ executions without failure})$ *from these priors equals the infimum.*

Each of Fig.s 4b, 4c and 4d shows the domain $\mathcal{R}$ of a joint prior distribution for (X, Λ) random variables, and the 4 points (i.e., black dots) in $\mathcal{R}$ assigned nonzero probabilities by this distribution. So, these joint priors are depicted as if one were looking down on the distribution and its domain "from above".

Example 1 (baseline). *Consider an on-demand SCS which acts only upon receipt of a demand from its environment. An assessor is 75% confident the software's pfe – i.e., its probability of failure per demand (pfd) – is no worse than $\epsilon = 10^{-5}$ (i.e. the engineering goal of PK2 with $\theta = 0.75$). After n failure-free tests of the SCS, the assessor is $c \times 100\%$ confident that the system meets Safety Integrity Level (SIL) 4, i.e. $b = 10^{-4}$ (see IEC (2010)). Fig. 5 shows three plots of c as a function of n using three Bayesian models[6]: univariate CBI (cf. Theorem 1), Bayesian Inference (BI) using a Beta prior[7] satisfying PK2, and CBI with confidence ϕ_1 and ϕ_2 in negative and positive dependence respectively (cf. Theorem 2).*

$$p_l = 0, \, b = 0.0001, \epsilon = 0.00001, \theta = 0.7$$

Legend:
- univariate CBI
- univariate BI with a prior of Beta(α=0.03, β)
- CBI with dependence, $\phi_1 = 0.8$, $\phi_2 = 0.05$

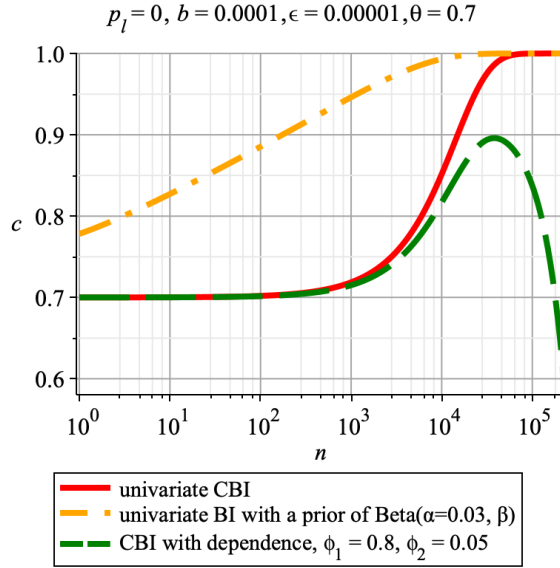**Figure 5:** (Example 1) $c \times 100\%$ posterior confidence in $[X \leqslant 10^{-4}]$ upon seeing n failure-free tests, from three Bayesian models with different PKs.

Univariate CBI and BI (using a Beta prior) both assume the executions are statistically independent (so, have likelihood $L(x; n) = (1 - x)^n$). In Fig. 5, as expected, univariate CBI gives less confidence (so, is more conservative) than BI using a Beta prior[8]; while both of these are more optimistic than CBI with doubts about independence. Failure-free evidence is always "good news" under univariate CBI and BI (i.e., the solid and dashed-dot curves monotonically increase to "certainty" in the bound). Contrastingly, with doubts in independence, such evidence can eventually undermine an assessor's posterior confidence in the 10^{-4} bound for *all* sufficiently large n (i.e., the uni-modal pattern of the dashed curve). This is because there are pessimistic reasons for why the failure-free tests could be occurring – reasons that suggest the successes are occurring despite the SCS not being very reliable. For example, the test-cases might be unrepresentatively "easy" for the SCS to correctly respond to, or there may be problems with the test oracle which mean some failures go undetected, see Littlewood & Wright (2007); Barr et al. (2015); Salako & Zhao (2023).

So, as failure-free evidence accumulates, *any* prior confidence the assessor has in the executions being positively correlated – i.e., any $\phi_2 > 0$ – will eventually undermine confidence in *any pfd* upper bound. On the other hand, prior confidence in negatively correlated executions (i.e. $\phi_1 > 0$) has a negligible impact on posterior confidence in the bound (see section 5's sensitivity analysis). Intuitively, the longer testing goes on for without failure, the greater the evidence *against* the tests being negatively dependent. An example of how negative dependence can occur is if an

---

[6]Note, with failure-free executions, the posterior confidence from CBI is not a function of p_l (since the distributions in Fig. 24 have no probabilities along the p_l line). Thus, *w.l.o.g.*, we set $p_l = 0$ in PK1.

[7]Specifically, to illustrate with a $Beta(\alpha, \beta)$, we first set the α parameter to 0.03, then fitted a "β" value to satisfy the quantile in PK2.

[8]This observation holds for any feasible prior, not only Beta distributions.

Table 1

A list of the prior distributions for the curves in the example plots of Section 4. See supplementary material for any listed figures not included in the paper.

	red solid	green dashed	blue dotted	orange (dash)dotted	pink spacedashed
Example 1 (Fig. 5)	Fig. 2	Fig. 4b	n/a	a Beta distribution∗	n/a
Example 2 (Fig. 6)	Fig. 2	Fig. 4b	Fig. 4b	n/a	n/a
Example 3 (Fig. 8)	Fig. 2	Fig. 4d	∗∗Fig.s 22b, 22d, 22f, 22h as n increases	∗∗Fig.s 22b, 22d, 22f, 22h as n increases	∗∗Fig.s 21b, 21d, 21f, 21h as n increases

∗An arbitrary Beta distribution satisfying the PKs.

∗∗This curve is a piecewise function – i.e. it's the confidence from the listed priors, in sequence, as n increases. The precise values n at which the curve switches between confidence from different priors depends on the execution outcomes (see proof in supplementary material, A).

assessor intentionally tries to "stress" the software during testing, by randomly including a disproportionate number of "difficult" demands – demands that are thought will likely cause software failure. So one might expect testing to reveal some negative dependence – a failure quickly followed by a success, then followed by another failure relatively soon afterwards, and so on. However, "no failures" may suggest the "difficult" demands are not actually difficult for the software.

4.3. Assessment with PKs for Nuclear Reactor Safety Systems

Next consider the assessment of a nuclear reactor safety protection system that is simple enough to possibly be fault-free (i.e., the system's *pfd* could be 0), see Littlewood & Rushby (2012). Typically, failure-free operational testing from such a system is required – otherwise, if a failure occurs, the system is taken offline and fixed, before testing resumes with a new sequence of demands. This scenario is very similar to the baseline of the last section – the testing evidence and most of the PKs are the same – but now, the engineering goal is "perfection" (i.e., PK2 with $\epsilon = 0$).

As before, we are interested in an assessor's confidence in a *pfd* bound b upon seeing n failure-free runs, where the assessor harbours doubts about the execution outcomes being i.i.d. (i.e., nonzero ϕ_1 or ϕ_2). This confidence in b is given by Theorem 2, simply by replacing PK2 with $P(X = 0) = \theta$; that is, $\epsilon = 0$ in the distributions of Fig.s 4b, 4c and 4d.

Example 2 (nuclear reactor protection systems). *Consider a nuclear reactor safety protection system that an assessor is 70% confident contains no faults (i.e., PK2 with $\epsilon = 0$ and $\theta = 0.7$). Upon seeing n failure-free tests, an assessor's conservative posterior confidence in the pfd bound 10^{-4} (SIL 4) is shown in Fig. 6, for three Bayesian models with different PKs: univariate CBI with $\epsilon = 0$, CBI with doubts in the independence assumption and $\epsilon = 10^{-5}$ (i.e., the baseline Example 1), and CBI with doubts in independence and $\epsilon = 0$.*

Example 2 highlights the benefit of the software possibly being fault-free: in contrast to Example 1, accumulated failure-free evidence *will not* eventually undermine posterior confidence in b. That is, the dotted curve in Fig. 6 is an increasing function of n that is asymptotic to the horizontal line $c = \frac{\theta}{\theta+(1-b)\phi_2}$, so it lies above the dashed curve[9], yet lies below the solid curve (i.e., its more conservative than the confidence from univariate CBI).

Insight 2 (For a possibly perfect system, failure-free testing cannot undermine confidence in a bound). As more successful executions are observed, the more likely it is that these observations are the result of either a fault-free system (so $\epsilon = 0$) or a perfectly positively correlated system (so $\lambda = 1$). That is, as n increases, the distribution in Fig. 4b tends to a distribution that has probability mass at only two points: a probability $\frac{\theta}{\theta+(1-b)\phi_2}$ at $(0,0)$, and a complementary probability $\frac{(1-b)\phi_2}{\theta+(1-b)\phi_2}$ at $(b, 1)$. In the limit of large n, the assessor will need more evidence to distinguish between these two possibilities.

The sensitivity of these insights – to changes in the strength of the PKs – is explored in section 5. While further implications of these insights are discussed in section 6.

---

[9]The asymptote is obtained by setting $\epsilon = 0$ in the distribution of Fig. 4b, and computing $\lim_{n\to\infty} P(X < b \mid n$ executions without failure).
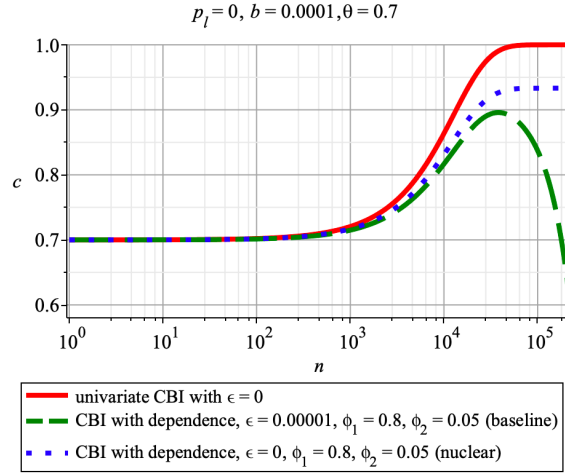
Figure 6: (Example 2) $c \times 100\%$ posterior confidence in $[X \leqslant 10^{-4}]$ upon seeing n failure-free tests, from three Bayesian models with different PKs.

4.4. Assessment with PKs for Autonomous Vehicles

We turn our attention to assessing AV-safety. In line with Kalra & Paddock (2016) and Zhao et al. (2019), the *pfe* is the *probability of a fatality-event per mile* (*pfm*). Here, each mile is treated as a "unit distance" over which software that enacts AV safety functions must correctly operate[10]. Unlike the (possibly fault-free) protection software of subsection 4.3, AV software cannot be expected to be fault-free – it relies on imperfect, sophisticated machine learning solutions performing a complex driving task. Consequently, unfortunately, AV safety software failures (leading to fatalities, in particular) have been known to occur; see National Highway Traffic Safety Administration (2022). This suggests a nonzero lower bound p_l on the *pfm* (see PK1), and an engineering goal of a "safe enough"[11] system rather than "perfection".

The following theorem gives conservative confidence in the *pfm* bound b (see general proof in supplementary material, A). Failures change the form of the Klotz likelihood from the one in Theorem 2 (see (3)). Here, an assessor doubts the independence of successive software executions as the AV drives over successive miles.

Theorem 3. *Using* (3) *and* (4), *the optimisation problem*

$$\inf_{D} P(\, X \leqslant b \mid n \text{ executions, } s \text{ failures, } r \text{ consecutive failures})$$

$$s.t. \quad PK1, \; PK2, \; PK3, \; PK4$$

is solved by prior distributions such as those in Fig. 7, since $P(\, X < b \mid n \text{ executions, } s \text{ failures, } r \text{ consecutive failures})$ *from these priors equals the infimum.*

Example 3 (AVs). *Consider an assessor's confidence in an AV being as safe as the average human driver[12] in terms of pfm (so* $b = 10^{-8}$*), after a fleet of AVs have driven millions of miles. Using PK parameter values from Zhao et al. (2020), we have: the engineering goal,* $\epsilon = 10^{-10}$*, is 2 orders of magnitude safer than the pfm for human drivers; and the lower bound on pfm is* $p_l = 10^{-15}$*. Fig. 8 shows confidence in b under different values of s (number of failures), r (consecutive failures),* ϕ_1*, and* ϕ_2*.*

Three observations from Fig. 8: **i)** like previous scenarios, the dashed curve shows that doubts in independent executions eventually undermine confidence in b when no failures occur (*cf.* Fig.s 5,6). However, the other curves in

---

[10]This is a coarse model of operation represented by a Bernoulli process: AV software responds to, say, discrete visual/positional stimulus it receives at a constant rate per mile. It would be interesting to see if the conclusions from the following analyses change significantly with a more sophisticated model of continuous operation (e.g., failures occurring according to an NHPP).

[11]How safe is "safe enough" is a separate topic, see Liu et al. (2019). A typical target would be several times safer than the average human driver.

[12]The exact statistic in the U.S. (2013) was 1.09e–8, as used by Kalra & Paddock (2016). For simplicity we round this to 10^{-8} in the examples.

(a) $\phi_2 \geqslant 1 - \theta$

(b) $\phi_1 \leqslant \theta$ and $\phi_2 \leqslant 1 - \theta$

(c) $\phi_1 \geqslant 1 - \theta$

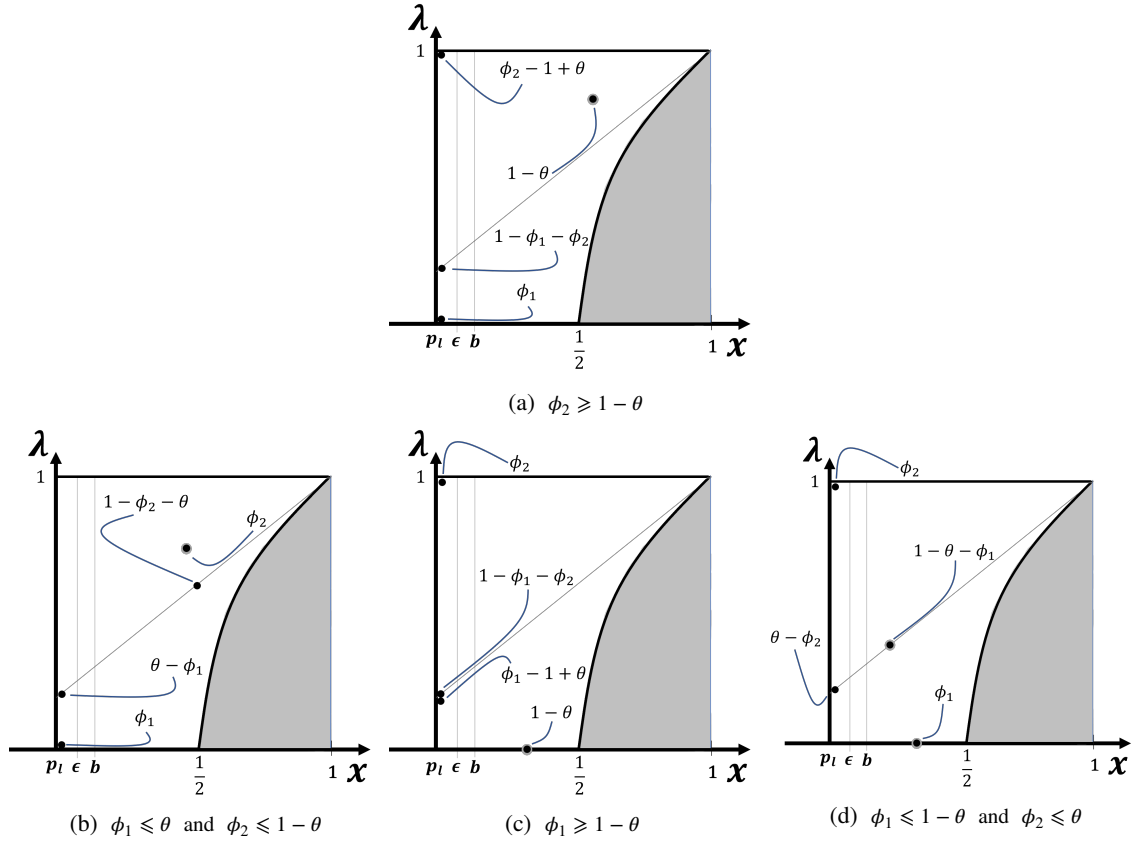(d) $\phi_1 \leqslant 1 - \theta$ and $\phi_2 \leqslant \theta$

Figure 7: Some examples of worst-case priors that give nonzero infima for the optimisation in Theorem 3; please see Figs. 21, 22, 23 of the supplementary material for all of the priors. When the software executes with **some isolated and consecutive failures** (i.e. $r > 0$), the worst-case priors can look like those in subfig.s 7a and 7b. And, if the executions contain some **isolated failures, but no consecutive failures** (i.e. $r = 0$), worst-case priors can look like subfig.s 7c and 7d instead. The exact locations of the support of these distributions (i.e. the black dots) depend on the values of the exponents in the likelihood function, and whether the 1st execution is a failure or not.

Fig. 8 (except the solid curve) show that failures allow confidence c to eventually approach 1. This is explained in Insight 3; **ii)** unsurprisingly, more failures requires more testing for confidence in b to grow (i.e., the dash-dot curve lies to the right of the dotted curve); and **iii)** more consecutive failures requires even more testing (i.e., the space-dash curve lies to the right of the dash-dot curve). Section 5's sensitivity analysis explores this further.

Insight 3 (failures can allow confidence in b to grow to 1). Consider the following two possibilities when failures occur: **i)** *no consecutive failures* (so $s > r = 0$), *and prior evidence weakly supports positively correlated executions* ($\phi_2 \leqslant \theta$). Then initially, execution outcomes are evidence of negative correlation (possibly from a system with *pfm* larger than b). However, as the number of successful executions increases (with no more failures), it becomes less likely that the executions are negatively (or positively) correlated; otherwise, more failures should have been observed. Instead, it's more likely that the successes are occurring because the *pfm* is smaller than b; **ii)** *consecutive failures* (so $s > r > 0$), *and prior evidence weakly supports negatively correlated executions* ($\phi_1 \leqslant \theta$). Again, initially, correlated executions (possibly from a system with *pfm* larger than b) are most likely. And like the previous case, as the successful executions increase (with no more failures), it becomes less likely that the executions *are* correlated, and more likely that the successes are due to the *pfm* being smaller than b.

5. The Sensitivity of Confidence Bounds to Changes in Prior Knowledge and Evidence

For the assessor that is uncertain about PK values, this section illustrates how to check the sensitivity/robustness of confidence in b to changes in PK values. The analyses also gives insight into how confidence responds to a
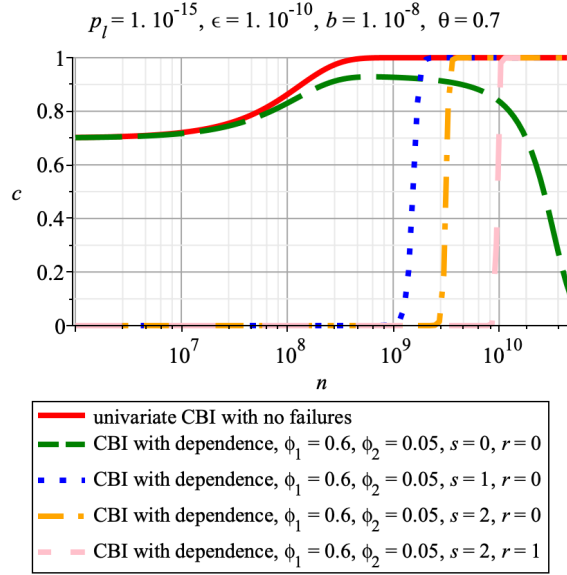
Figure 8: (Example 3) $c \times 100\%$ posterior confidence in $[X \leqslant 10^{-8}]$ upon seeing s failures (r of them consecutive) in n tests, from CBI models with different PKs.
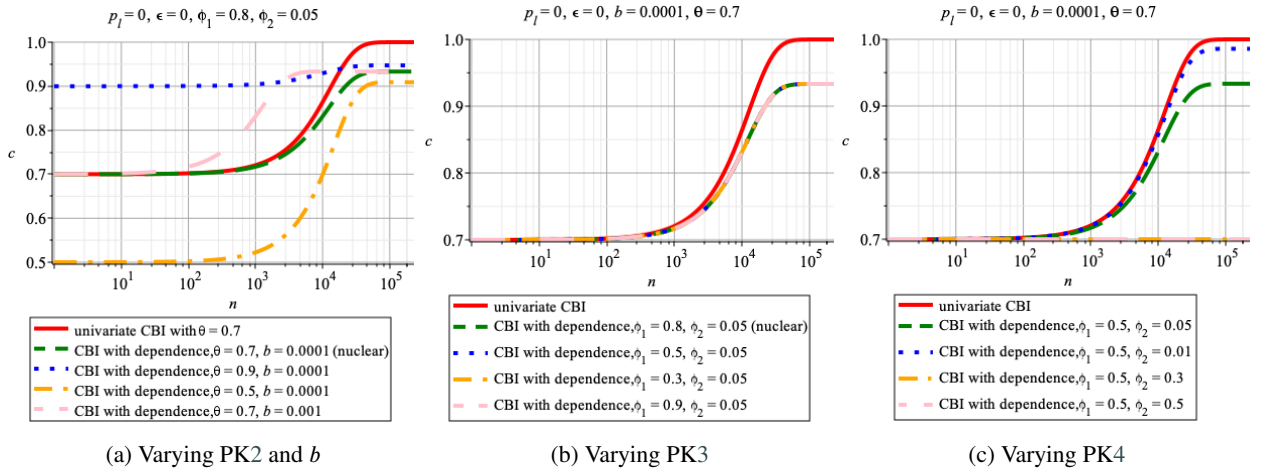

Figure 9: Sensitivity analysis varying PKs for the nuclear reactor safety protection system scenario.

"strengthening" of prior reliability evidence. We systematically vary PK parameters, the bound b, and the execution outcomes. The prior distributions used in the numerical analyses are summarised in Table 2.

5.1. The Nuclear Safety Protection System Scenario

Fig. 9 provides 3 sub-figures highlighting the effects of changes in PK parameters (except PK1).

In Fig. 9a, "CBI with dependence" curves are asymptotically more conservative than the univariate CBI solid curve. However, because the system could be fault-free, the curves show that confidence in b always increases with increasing failure-free operational evidence. Also, the smaller ϕ_2 becomes, or the bigger b or θ are, the greater confidence in b can become.

Fig. 9b illustrates how changes in ϕ_1 have no apparent impact on confidence in b. However, Fig. 9c shows that the smaller ϕ_2 becomes, the closer the "CBI with dependence" curve gets to the univariate CBI curve. While, for $\phi_2 \geqslant 1 - \theta$, the confidence in b is the constant horizontal line $c = \frac{\theta}{\theta + (1-b)(1-\theta)}$ (see Fig. 4c with $p_l = \epsilon = 0$).

Table 2
A list of all prior distributions used in the sensitivity analysis of Section 5. See the supplementary material for any listed figures not included in the paper.

Sensitivity Analysis in Section 5							
Scenarios with various PKs		red solid	green dashed	blue dotted	orange dashdotted	pink spacedashed	black spacedotted
Nuclear reactor protection system (Fig. 9)	Varying PK2 and b (Fig. 9a)	Fig. 2	Fig. 4b	Fig. 4c	Fig. 4b	Fig. 4b	n/a
	Varying PK3 (Fig. 9b)	Fig. 2	Fig. 4b	Fig. 4c	Fig. 4c	Fig. 4b	n/a
	Varying PK4 (Fig. 9c)	Fig. 2	Fig. 4c	Fig. 4c	Fig.s 4c/4d	Fig. 4c	n/a
AVs with no consecutive failures (Fig. 10)	Varying PK1, PK2 and b (Fig. 10a)	Fig.s 22b, 22d, 22f, 22h as n increases	Fig.s 22b, 22d, 22f, 22h as n increases	Fig.s 22b, 22d, 22f, 22h as n increases	Fig.s 22b, 22d, 22f, 22h as n increases	Fig.s 22b, 22d, 22f, 22h as n increases	Fig.s 22b, 22d, 22f, 22h as n increases
	Varying PK3 (Fig. 10b)	Fig.s 22b, 22d, 22f, 22h as n increases	Fig.s 22b, 22d, 22f, 22h as n increases	Fig.s 22a, 22c, 22e, 22g as n increases*	Fig.s 22b, 22d, 22f, 22h as n increases	Fig.s 22b, 22d, 22f, 22h as n increases	Fig.s 22a, 22c, 22e, 22g as n increases
	Varying PK4 (Fig. 10c)	Fig.s 22a, 22c, 22e, 22g as n increases*	Fig.s 22a, 22c, 22e, 22g as n increases*	Fig.s 22a, 22c, 22e, 22g as n increases*	Fig.s 22a, 22c, 22e, 22g as n increases*	Fig.s 22a, 22c, 22e, 22g as n increases	Fig.s 23b, 23d, 23f, 23h as n increases
AVs with consecutive failures (Fig. 11)	Varying PK1, PK2 and b (Fig. 11a)	Fig.s 21b, 21d, 21f, 21h as n increases	Fig.s 21b, 21d, 21f, 21h as n increases	Fig.s 21b, 21d, 21f, 21h as n increases	Fig.s 21b, 21d, 21f, 21h as n increases	Fig.s 21b, 21d, 21f, 21h as n increases	Fig.s 21b, 21d, 21f, 21h as n increases
	Varying PK3 (Fig. 11b)	Fig.s 21b, 21d, 21f, 21h as n increases	Fig.s 23a, 23c, 23e, 23g as n increases	Fig.s 21b, 21d, 21f, 21h as n increases	Fig.s 21b, 21d, 21f, 21h as n increases	Fig.s 23a, 23c, 23e, 23g as n increases	Fig.s 21b, 21d, 21f, 21h as n increases
	Varying PK4 (Fig. 11c)	Fig.s 21b, 21d, 21f, 21h as n increases	Fig.s 21a, 21c, 21e, 21g as n increases	Fig.s 21a, 21c, 21e, 21g as n increases	Fig.s 21b, 21d, 21f, 21h as n increases	Fig.s 21a, 21c, 21e, 21g as n increases	Fig.s 21a, 21c, 21e, 21g as n increases

*The set of worst-case priors can be replaced by Fig.s 22b, 22d, 22f, 22h as n increases, since the parameters chosen cover these cases.

5.2. The AV Scenario

Recall that, unlike the baseline and nuclear protection system scenarios, failures occur (albeit rarely[13]) during sufficiently long road testing campaigns (so $s > 0$). The confidence in bound b from Theorem 3 is dependent on whether some of these failures are consecutive ($r > 0$) or not ($r = 0$). We conduct two sets of analyses[14] along these possibilities.

5.2.1. With No Consecutive Failures ($s > r = 0$)

In Fig. 10a confidence in b changes in response to increases in s and b. The increase in ϵ has no noticeable impact – the solid and dashed curves overlap. However, when b is increased, the required number of executions n to support a given confidence level reduces by a similar order of magnitude. In contrast, an additional failure increases n significantly – so the dotted and solid curves lie to the left of the spacedashed and dashdotted curves, respectively. This is consistent with the findings of Littlewood & Wright (1997).

Changes in ϕ_1 have no noticeable effect on confidence in b (see Fig. 10b). Perhaps because there is very little operational evidence to support negative correlations – i.e. only few instances of "switching" between failure and success.

On the other hand, changes in ϕ_2 have a clear impact, as shown in Fig. 10c. An increase in ϕ_2 requires an increase in n. Moreover, when $\phi_2 \geqslant \theta$, confidence in b becomes 0 *for all* n (see the prior distributions for $r = 0$ in Fig. 23).

---

[13]Due to the severe negative impact failures can have in SCSs, we only consider operational campaigns with no more than a few failures.
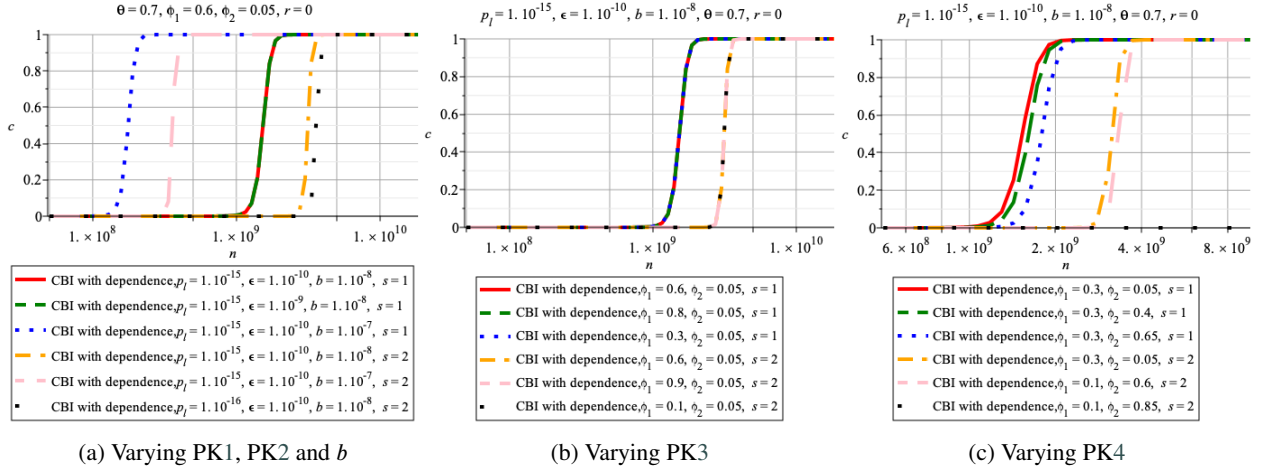[14]Note, Fig. 8 shows the result of r becoming nonzero for fixed s.

(a) Varying PK1, PK2 and b (b) Varying PK3   (c) Varying PK4

Figure 10: Sensitivity analysis varying PKs for the AV-safety scenario with no consecutive failures.

(a) Varying PK1, PK2 and b (b) Varying PK3   (c) Varying PK4
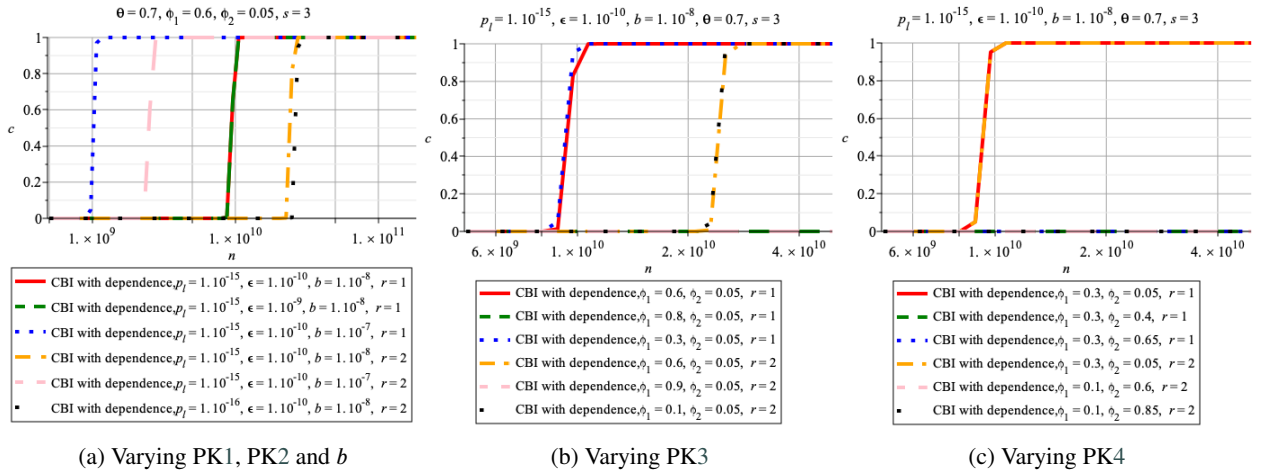
Figure 11: Sensitivity analysis varying PKs for the AV-safety scenario with some consecutive failures (note, when $\phi_2 \geqslant 1 - \theta$, the plots are effectively 0, but not 0).

5.2.2. With Some Consecutive Failures ($s > r > 0$)

Despite consecutive failures, Figs. 11a and 10a broadly give the same insights.

When prior confidence in negative correlations is strong (i.e. $\phi_1 \geqslant \theta$), Fig. 11b shows confidence in b is 0 *for all* n, just like the condition on ϕ_2 in the $r = 0$ scenario (see the left column of prior distributions in Fig. 23). While a large ϕ_2 (i.e. $\phi_2 \geqslant 1 - \theta$) gives practically 0 confidence in b in Fig. 11c.

We now summarise the results of the sensitivity analysis:

Insight 4 (results of sensitivity analysis). Confidence in b from "CBI with dependence" is insensitive to changes in ϕ_1, although confidence is 0 for $\phi_1 \geqslant \theta, r > 0$. Confidence in b is sensitive to ϕ_2. Both the number of failures s and the number of consecutive failures r have a significant effect on confidence in b. When there are no failed executions p_l is irrelevant, with some effect when failures occur. The impact of the engineering goal, ϵ, is consistent with previous CBI models, e.g. Zhao et al. (2020) – smaller ϵ gives greater confidence in b when no failures are observed, but has no significant impact when there are failures.
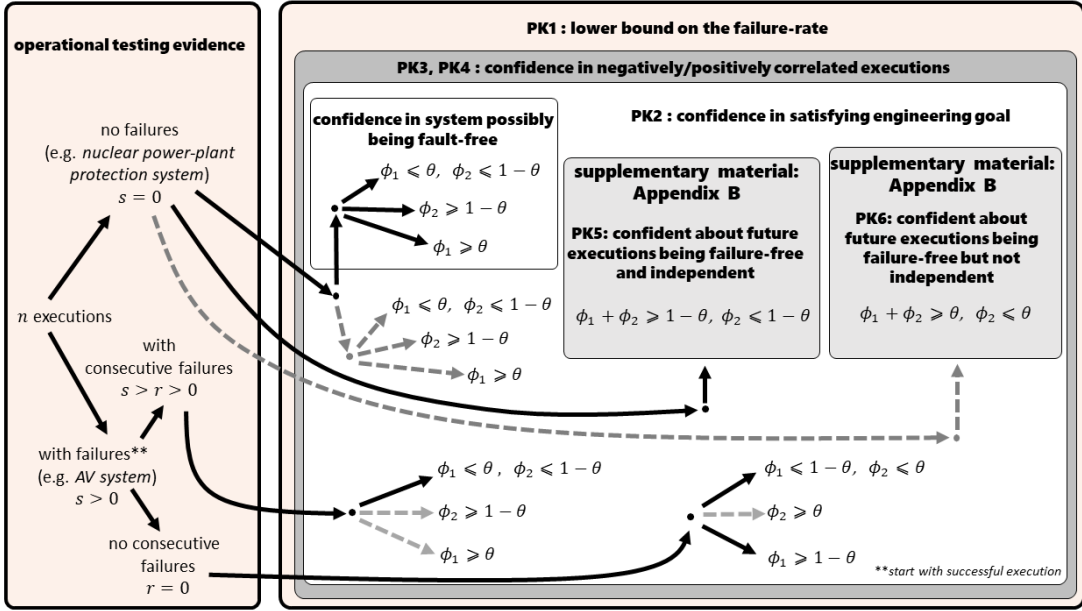
Figure 12: An updated overview of the various assessment scenarios analysed in this paper (*cf.* Fig. 1), indicating the possible testing outcomes and prior confidence an assessor could have. The dashed paths are scenarios where either testing is futile in supporting reliability claims, or the scenarios are of little practical interest.

6. Discussion

6.1. Incorporating Doubts about Model Assumptions into Reliability Assessments

In Bayesian assessments, one should always question the properties of the statistical model (such as independent executions), and whether these properties are appropriate in a given real-world context. Our use of CBI illustrates a general, formal method for incorporating such doubts – about the suitability of any statistical model properties – directly into the assessment. Since the results of CBI are guaranteed to be conservative (see Insight 1), this is a conservative version of a Bayesian approach first proposed by Draper (1995). Draper suggests that if one is uncertain about the suitability of model properties, one should perform the inference with an "expanded" model that weakens the properties in question and has the original model as a special case. In this sense, our choice of the Klotz model is not arbitrary: it is the simplest model we know that exhibits dependent, stationary Bernoulli trials, and that has "independent executions" as a special case. This approach is incremental; if one is doubtful of the Klotz model, a model expansion of the Klotz model can be used for assessment.

Using this approach in Salako & Zhao (2023), we illustrated how to formally check the impact of one's doubts (about i.i.d. execution outcomes) on reliability claims, where such claims are based on Bayesian inference with operational testing data. The results suggested that, for on-demand systems, using the i.i.d. assumption in assessments could result in extremely optimistic claims, but not always. By weakening the prior knowledge an assessor must justify before commencing operational testing, the current paper extends this previous work to a much wider class of scenarios. For example, there are assessment scenarios for which the i.i.d. assumption supports claims "close to being" conservative. "How close" will depend on the strength of reliability evidence available (e.g., Fig. 8 shows how accumulating operational evidence can make i.i.d.-based claims less optimistic).

6.2. Which Prior Beliefs give Conservative Confidence Bounds?

Only certain prior beliefs about the *pfe*, X, and the dependence between executions, Λ, will ensure an assessor's posterior confidence in a *pfe* upper bound b is conservative. The CBI solutions of section 4 make clear what these beliefs are – i.e., these beliefs are encoded in the prior distributions that solve Theorems 2 and 3. Specifically, the beliefs are encoded as those (x, λ) locations in region $\mathcal{R}$ that each of these distributions assign nonzero probability to (e.g., see Fig.s 4, 7, 25). Four main factors determine such beliefs: **i)** the execution outcomes during operational

testing (i.e., the successes/failures); **ii)** prior knowledge, e.g. evidence strongly suggests the executions are positively correlated (i.e., large ϕ_2), not negatively correlated (i.e., small ϕ_1); **iii)** which beliefs about (X, Λ) are *least likely* to have produced the testing outcomes, if the *pfe* is less than b; and **iv)** which beliefs are *most likely* to have produced the outcomes, if the *pfe* is larger than b. Here, "least likely" and "most likely" are determined by the Klotz likelihood.

In addition to ensuring an assessor's confidence is conservative, here are two more reasons for why such beliefs are important. Firstly, they are consistent with the available evidence, since the beliefs are encoded in prior distributions that are (the limits of sequences of prior distributions that are) consistent with the evidence. So, the assessor cannot "rule out" these beliefs without extra evidence, and the consequences of these beliefs should be taken seriously. Secondly, when reliability evidence is "weak", these beliefs can make operational testing futile: the more one observes successful executions, the more doubtful one becomes about the *pfe*. Assessors should have enough evidence before embarking on operational testing; we elaborate on these points below.

For example, consider when all of the executions are successful (e.g. Example 1 of section 4). Superficially, this suggests the executions are strongly positively correlated, or the system's *pfe* is low. However, the assessor can take the more conservative view that these successful executions are evidence the system is not quite good enough. The assessor does this by holding the following beliefs: **i)** if the *pfe* is larger than b, then successful tests are most likely if the following two beliefs are true: the executions are "perfectly positively correlated" (i.e. $\lambda = 1$), and the *pfe* is "as small as possible, but no smaller than b". In terms of the Klotz likelihood, these beliefs are encoded as the location[15] $(b, 1)$ in $\mathcal{R}$; **ii)** if instead, the *pfe is* smaller than b, then successful tests are least likely if the following two beliefs are true: the executions are as "negatively correlated" as possible, and the *pfe* is "as big as possible, but no bigger than b". These beliefs are encoded as the location $(b, 0)$.

PKs refine these beliefs, giving the prior distributions shown in Fig.s 4 and 24. These beliefs imply that as failure-free executions increase without bound, in order for the assessor to be conservative, their confidence in the bound must diminish (e.g., see dotted curve in Fig. 6). Because the increasing number of successful executions makes all other beliefs unlikely, except the beliefs that the executions are "perfectly positively correlated" and the *pfe* is "as small as possible, but no smaller than b". The Klotz likelihood tends to zero at all of the nonzero probability locations in Fig. 4, except at the point $(b, 1)$ where the likelihood has the constant value $(1-b)$ for all n. Failure-free executions eventually undermine confidence.

It is possible that the successful executions are occurring because the *pfe* is very small. The problem with this possibility is that any very small *pfe* eventually becomes too unlikely to have produced a sufficiently large number of successful executions. Moreover, there are more pessimistic reasons (not disallowed by the evidence) for runs of successful executions. For example, all of the test inputs may have been "easy" for the software to respond to. This could happen by chance: e.g., the operational environment just happens to be submitting a sequence of easy inputs. Overcoming such problems of chance may require an infeasible amount of testing. Another pessimistic reason for the successful executions could be an error in the test-case generation procedure, which systematically fails to generate inputs that lie in the system's failure region. Or an error in the test oracle, which fails to indicate true failures. Such possibilities are consistent with previous works that show how "favourable" operational evidence can undermine confidence during assessments; e.g., Littlewood & Wright (2007); Salako & Zhao (2023). To make progress with using failure-free evidence, an assessor must use appropriate additional evidence to rule out pessimistic reasons for not observing any failures.

If the system *could* be fault-free – in modelling terms, the engineering goal ϵ is zero – then a fault-free system *could* produce the increasing number of successful executions. This possibility allows the assessor's confidence in the bound b to grow, because the confidence an assessor has in the bound being satisfied is never smaller than their confidence in the system being fault-free. The confidence in the system being fault-free increases as the number of successful executions increases, thus increasing confidence in b too. See Fig. 6, where confidence in b increases from θ to $\frac{\theta}{\theta+(1-b)\phi_2}$ along the dotted curve. Contrarily, the successes could also be produced by more pessimistic reasons, so confidence in these more pessimistic reasons also increases. For example, the dotted curve in Fig. 6 and the prior distribution in Fig. 4b that produced this curve, together imply that this confidence increases from $\frac{(1-b)\phi_2}{\theta+(1-b)(1-\theta)}$ to $\frac{(1-b)\phi_2}{\theta+(1-b)\phi_2}$, since $\phi_2 \leqslant 1 - \theta$ is assumed. So, the assessor will always be uncertain about whether the *pfe* is better than b.

---

[15] $(b, 1)$ is a *limit point* – the limit of a sequence of $\mathcal{R}$ locations that represent increasingly pessimistic beliefs. See supplementary material, A.

So far we have discussed conservative beliefs when only failure-free executions are observed. Ironically, a few failures during testing can help overcome the pessimism we have highlighted. As the number of successful executions increases, any initial failures become evidence that the executions cannot be "perfectly positively correlated". Otherwise, if the executions *were* this strongly correlated, either no successes or no failures would have occurred! The previously pessimistic belief implied by location $(b, 1)$ – where $\lambda = 1$ – must now move to a different pessimistic location where $x < \lambda < 1$ (indicating less strong positive correlation). Now, as successful executions increase, it eventually becomes most likely that the successes are being produced by a system with a *pfe* smaller than b. Consequently, the assessor's confidence in b eventually approaches certainty, although this can require considerable amounts of successful executions because of the few failures (e.g. Fig. 8).

Like the "failure-free" scenario, there are also situations where testing is futile when some failures occur. The futility here is even more extreme than before: confidence in the bound is now identically zero, no matter how many more successful tests are observed. For instance, note the zero confidence in Fig.s 10 and 11 from priors such as those in Fig. 23. If the failures during testing are isolated and few, then strong confidence in the executions being positively correlated (i.e., $\phi_2 \geqslant \theta$) undermines confidence in the bound b. If the failures are clustered and few, then strong confidence in the executions being negatively correlated (i.e., $\phi_1 \geqslant \theta$) undermines confidence in b. In both situations, the assessor's prior confidence in the system being "very reliable" is simply not strong enough to rule out pessimistic causes for the testing outcomes.

The various assessment scenarios are summarised in Fig. 12 (an updated version of Fig. 1). Each path through the figure – starting from the far-left at the "n executions" node – gives the operational evidence and the prior confidence (i.e., PK parameter ranges) an assessor could have. The paths containing dashed lines are paths to be weary of; paths that our analyses reveal to be asymptotically futile or of little practical interest. Assessor's on such paths should seek stronger evidence of the system being sufficiently reliable prior to commencing testing.

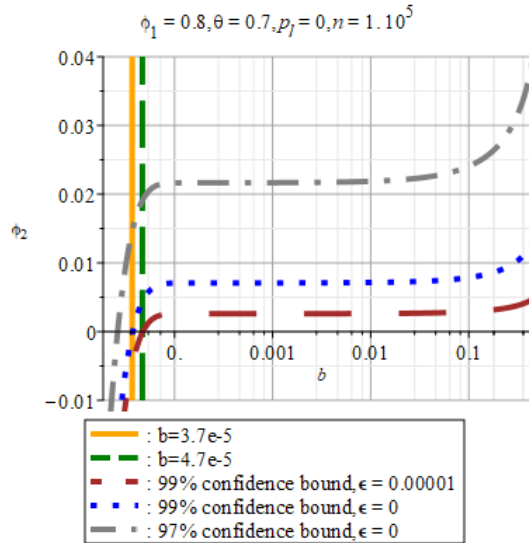### 6.3. Is assuming Independence always very Optimistic?

Figure 13: The relationship between prior confidence ϕ_2 in positively correlated executions and the $(1 - \alpha) \times 100\%$ confidence bound b, when the system is subjected to 10^5 tests without failure. The curves are obtained from posterior confidence given by the prior in Fig. 4b. The values of α plotted here are curves for $\alpha = 0.01, 0.03$. The smaller ϕ_2 becomes, the smaller the *pfe* upper bound b that can be supported at a given level of confidence.

Concerning the impact of assuming independent executions, Chen & Mills (1996) observe the following when using classical inference with their Markov model: for a given number of successes/failures during testing, **i)** positively correlated executions give bigger confidence bounds (i.e., worse values for *pfe* estimates), compared with using Thayer *et al.*'s independent executions model; **ii)** a "*not so big*" positive correlation gives confidence bounds that are comparable to those given by Thayer *et al.*'s model. This is consistent with our findings.

Indeed, consider the examples in Fig. 13, where all 10^5 executions are successful, and the system could be "fault-free". Then, for instance, the 99% confidence bound from univariate CBI (i.e., CBI that assumes independence) is 3.7×10^{-5} – the smallest b-value from the 99% confidence bound (dotted) curve, precisely when $\phi_2 = 0$. All other 99% confidence bounds from this curve – i.e., all 99% confidence bounds from CBI using the Klotz model, $\epsilon = 0$ and $\phi_2 > 0$ – are larger. Moreover, as ϕ_2 increases, the confidence bounds b increase. While, the smaller the assessor's prior confidence in positive correlation (i.e., ϕ_2 decreases), the closer the Klotz confidence bound becomes to the confidence bound under independent executions (i.e., the intersection of the curves with the horizontal axes).

If the system cannot be fault-free, CBI with the Klotz model is significantly more conservative. The long-dashed curve gives the 99% confidence bounds when $\epsilon = 10^{-5}$. Here, the univariate CBI 99% confidence bound is 4.7×10^{-5} (at $\phi_2 = 0$) – "4 orders of magnitude" smaller than the 99% confidence bound 0.1 from the Klotz model with $\phi_2 = 3.5 \times 10^{-3}$.

Interestingly, unlike Chen and Mill's other observation (that a reduction in confidence bounds accompanies an increase in negative correlation), section 5 suggests that confidence bounds from failure-free testing are insensitive to prior confidence in negative correlation ϕ_1. In fact, since $\phi_1 \geqslant \theta$ in Fig. 13, the relevant posterior confidence is given by the prior in Fig. 4b as $\frac{(1-\epsilon)(1-\epsilon/_{1-\epsilon})^{n-1}\theta}{(1-\epsilon)(1-\epsilon/_{1-\epsilon})^{n-1}\theta+(1-\theta-\phi_2)(1-b)^n+(1-b)\phi_2}$, which doesn't depend on ϕ_1. "No failures" supports confidence in positive correlation ϕ_2, and undermines confidence in negative correlation ϕ_1.

So, the plots illustrate how conservative confidence bounds can be very sensitive to confidence in positive correlation – i.e., small changes in ϕ_2 can result in "orders of magnitude" changes in confidence bounds. If evidence strongly supports the executions being positively correlated, the confidence bounds obtained under assuming independence can be quite optimistic. On the other hand, B and Fig. 26 of the supplementary material show how being skeptical about independent executions can be initially optimistic; giving smaller confidence in b than the confidence from univariate CBI. While strongly believing in independence can be conservative initially. As successes accumulate, these roles between "skepticism" and "strong belief" are reversed, with "skepticism" in independence eventually giving conservative confidence and "strong belief" giving optimistic confidence.

6.4. Limitations, Generalisations and Future Work

The following Klotz model limitations, first highlighted in Salako & Zhao (2023), remain.

Using a relatively simple model of dependent Bernoulli trials – i.e., the Klotz model – we have illustrated how one might account for dependent executions in conservative reliability assessments. Of course, there is scope for studying the implications of more expressive failure-models. For instance, many systems experience different types of failure, some of which may be considered benign. Some Markov models that capture this include the models of Csenki (1993); Goseva-Popstojanova & Trivedi (2000); Bondavalli et al. (1999). In contrast, the Klotz model treats all failure-types (and all successes-types) identically, in terms of how likely they are to occur, and the model ignores variations in the impact different failure-types have on system stakeholders when they occur.

Another Klotz model limitation is that positive correlation in both of its forms – i.e., whether failures are likely to follow previous failures or successes follow previous successes – are captured by the size of parameter λ relative to x (see Remark 1). A further limitation is that the type of dependence – i.e., whether executions are positively or negatively correlated, or independent – is fixed for the duration of the system's operation. Certainly, there are practical situations where dependence can vary significantly over time; e.g., a change in the system's internal state makes failures much more/less likely. Or dependence can exist between several executions separated in time. Or, the sequence of executions could be halted whenever a failure occurs and the software could be fixed, before the software is allowed to resume executing – thus altering the faults the software contains and the dependence among execution outcomes. Accounting for such dependence variation requires a failure-model that explicitly captures time-dependent correlations. These scenarios justify a weakening of the conditional independence in the Klotz model: in the model, T_i is conditionally independent of $T_{i-2}, T_{i-3}, \ldots, T_1$ given T_{i-1}. In future work, it will be interesting to consider longer dependence structures (over several "time steps" into the past) – e.g., T_i being dependent on the last "$i - 1$" execution outcomes. By applying the general conservative approach illustrated in this paper, an assessor can check the robustness of assessment claims based on models with more general dependence structures.

The Klotz model is a 1st-order stationary stochastic process (see Klotz (1973); Salako & Zhao (2023)). *pfe*s used in reliability assessment make sense when the failure process is stationary. Because then, the probability of the system failing its n-th execution is the same for all n, and it equals the *pfe*. This, despite the conditional probability of failing the very next execution being dependent on, say, the success/failure of the last execution. Consequently,

upper confidence bounds on such *pfe*s are useful measures of reliability in those practical scenarios characterised by a stationary failure process. But when failure probabilities are time-dependent, one should forego using *pfe*s in assessment claims and opt for more suitable reliability measures, such as the probability of failure-free operation in the future (see Strigini (1996)).

Even with a 1st-order stationary model, it's still worth studying the impact of the independence assumption on reliability measures like the probability of future failure-free operation. Previous CBI studies have shown that an assessor's justifications for a conservative claim are often different for different measures – even if the justifications are ultimately based on the same PKs. Also, some measures may be more sensitive to PK changes than other measures.

7. Concluding Remarks

Statistically independent software executions are often assumed when assessing software reliability. If inappropriate, this assumption can result in (dangerously) optimistic reliability claims. By formalising informal notions of "doubting" the independence assumption, and by employing conservative Bayesian methods, this work demonstrates how such doubts can be accounted for in assessments.

This paper contains analyses of various assessment scenarios. This involved the constrained mathematical optimisation of an assessor's confidence in an upper bound on the probability of failure per execution (*pfe*), after observing the system in operation. The work highlights a number of practical considerations. For example, a system exhibiting no failures during operation can give *less* confidence in a *pfe* bound, compared with if the system *had* exhibited failures. Or confidence can be very sensitive to failures; each additional failure means significantly more failure-free operation is needed for confidence to grow.

The scope of the results makes clear that a nuanced answer is required to the question of whether assuming independence undermines assessments. The answer depends, often sensitively, on various factors outlined in the paper. So that sometimes, the independence assumption has "*little to no*" impact on conservatism. And sometimes, the impact is simply too great to ignore. A "case-by-case" approach to estimating this impact in practice is advised, and the methods and many solutions in this paper provide assessors/practitioners with the means to do this.

Acknowledgements

References

Ammann, P. E., & Knight, J. C. (1988). Data diversity: an approach to software fault tolerance. *IEEE transactions on computers*, *37*, 418–425.

Atwood, C. L., Laboratories, S. N., & Commission, U. S. N. R. (2003). *Handbook of parameter estimation for probabilistic risk assessment*. Division of Risk Analysis and Applications, Office of Nuclear Regulatory Research, U.S. Nuclear Regulatory Commission Washington, DC.

Barr, E. T., Harman, M., McMinn, P., Shahbaz, M., & Yoo, S. (2015). The oracle problem in software testing: A survey. *IEEE Transactions on Software Engineering*, *41*, 507–525.

Berger, J., & Moreno, E. (1994). Bayesian robustness in bidimensional models: Prior independence. *Journal of statistical planning and inference*, *40*, 161–176.

Berger, J. O. (1990). Robust Bayesian analysis : Sensitivity to the prior. *journal of statistical planning and inference*, *25*, 303–328.

Berger, J. O. (1994). An overview of robust Bayesian analysis. *Test*, *3*, 5–124.

Bergman, B., & Xie, M. (1991). On Bayesian software reliability modelling. *Journal of Statistical Planning and Inference*, *29*, 33–41.

Bishop, P. (1993). The variation of software survival time for different operational input profiles (or why you can wait a long time for a big bug to fail). In *FTCS-23 The Twenty-Third International Symposium on Fault-Tolerant Computing* (pp. 98–107). IEEE.

Bishop, P., Bloomfield, R., Littlewood, B., Povyakalo, A., & Wright, D. (2011). Toward a formalism for conservative claims about the dependability of software-based systems. *IEEE Transactions on Software Engineering*, *37*, 708–717.

Bondavalli, A., Chiaradonna, S., Di Giandomenico, F., & La Torre, S. (1997). Modelling the effects of input correlation in iterative software. *Reliability Engineering & System Safety*, *57*, 189–202.

Bondavalli, A., Chiaradonna, S., Di Giandomenico, F., & Strigini, L. (1995). Dependability models for iterative software considering correlation between successive inputs. In *Proc. of IEEE Int. Computer Performance and Dependability Symposium* (pp. 13–21). Erlangen, Germany: IEEE.

Bondavalli, A., Chiaradonna, S., Di Giandomenico, F., & Strigini, L. (1999). A contribution to the evaluation of the reliability of iterative-execution software. *Software testing, verification & reliability*, *9*, 145–166.

Bunea, C., Charitos, T., Cooke, R. M., & Becker, G. (2005). Two-stage Bayesian models—application to ZEDB project. *Reliability Engineering & System Safety*, *90*, 123 – 130.

Chen, S., & Mills, S. (1996). A binary Markov process model for random testing. *IEEE Transactions on Software Engineering*, *22*, 218–223.

Copson, E. T. (1968). *Metric Spaces*. Cambridge Tracts in Mathematics. Cambridge University Press.

Csenki, A. (1993). Reliability analysis of recovery blocks with nested clusters of failure points. *IEEE transactions on reliability*, *42*, 34–43.

Draper, D. (1995). Assessment and propagation of model uncertainty. *Journal of the Royal Statistical Society. Series B (Methodological)*, *57*, 45–97.

Duran, J. W., & Ntafos, S. C. (1984). An evaluation of random testing. *IEEE Transactions on Software Engineering*, *SE-10*, 438–444.

Goseva-Popstojanova, K., & Trivedi, K. S. (2000). Failure correlation in software reliability models. *IEEE Transactions on Reliability*, *49*, 37–48.

Hörwick, M., & Siedersberger, K.-H. (2010). Strategy and architecture of a safety concept for fully automatic and autonomous driving assistance systems. In *2010 IEEE Intelligent Vehicles Symposium* (pp. 955–960). IEEE.

Huang, R., Sun, W., Xu, Y., Chen, H., Towey, D., & Xia, X. (2021). A survey on adaptive random testing. *IEEE Transactions on Software Engineering*, *47*, 2052–2083.

IEC (2010). *IEC61508, Functional Safety of Electrical/ Electronic/Programmable Electronic Safety Related Systems*. International Electrotechnical Commission (IEC).

Kalra, N., & Paddock, S. (2016). Driving to safety: How many miles of driving would it take to demonstrate autonomous vehicle reliability? *Transp. Research Part A: Policy and Practice*, *94*, 182–193.

Klotz, J. (1973). Statistical Inference in Bernoulli Trials with Dependence. *The Annals of Statistics*, *1*, 373–379.

Koopman, P., & Wagner, M. (2016). Challenges in autonomous vehicle testing and validation. *SAE International Journal of Transportation Safety*, *4*, 15–24.

Lavine, M. (1991). Sensitivity in Bayesian statistics: The prior and the likelihood. *Journal of the American Statistical Association*, *86*, 396–399.

Littlewood, B., Popov, P., & Strigini, L. (2002). Assessing the reliability of diverse fault-tolerant software-based systems. *Safety science*, *40*, 781–796.

Littlewood, B., & Rushby, J. (2012). Reasoning about the reliability of diverse two-channel systems in which one channel is 'possibly perfect'. *IEEE Tran. on Software Engineering*, *38*, 1178–1194.

Littlewood, B., Salako, K., Strigini, L., & Zhao, X. (2020). On reliability assessment when a software-based system is replaced by a thought-to-be-better one. *Reliability Engineering & System Safety*, *197*, 106752.

Littlewood, B., & Wright, D. (1997). Some conservative stopping rules for the operational testing of safety critical software. *IEEE Transactions on Software Engineering*, *23*, 673–683.

Littlewood, B., & Wright, D. (2007). The use of multilegged arguments to increase confidence in safety claims for software-based systems: A study based on a BBN analysis of an idealized example. *IEEE Transactions on Software Engineering*, *33*, 347–365.

Liu, P., Yang, R., & Xu, Z. (2019). How safe is safe enough for self-driving vehicles? *Risk Analysis*, *39*, 315–325.

Lyu, M. R. (Ed.) (1996). *Handbook of Software Reliability Engineering*. USA: McGraw-Hill, Inc.

Miller, D. R. (1986). Exponential order statistic models of software reliability growth. *IEEE Transactions on Software Engineering*, *SE-12*, 12–24.

Miller, K. W., Morell, L. J., Noonan, R. E., Park, S. K., Nicol, D. M., Murrill, B. W., & Voas, M. (1992). Estimating the probability of failure when testing reveals no failures. *IEEE Transactions on Software Engineering*, *18*, 33–43.

Moreno, E., & Cano, J. A. (1991). Robust Bayesian analysis with ϵ-contaminations partially known. *Journal of the Royal Statistical Society. Series B (Methodological)*, *53*, 143–155.

Musa, J. D. (1993). Operational profiles in software-reliability engineering. *IEEE Software*, *10*, 14–32.

Musa, J. D., Iannino, A., & Okumoto, K. (1987). *Software reliability: measurement, prediction, application*. McGraw-Hill, Inc.

National Highway Traffic Safety Administration (2022). *Summary Report: Standing General Order on Crash Reporting for Level 2 Advanced Driver Assistance Systems*. Technical Report DOT HS 813 325 U. S. Department of Transportation.

O'Hagan, A., Buck, C., Daneshkhah, A., Eiser, J., Garthwaite, P., Jenkinson, D., Oakley, J., & Rakow, T. (2006). *Uncertain Judgements: Eliciting Experts' Probabilities*. Statistics in Practice. Wiley.

PD IEC TR 63161:2022 (2022). *Assignment of safety integrity requirements: basic rationale*. Standard IEC Geneva, Switzerland.

Popov, P. (2013). Bayesian reliability assessment of legacy safety-critical systems upgraded with fault-tolerant off-the-shelf software. *Reliability engineering & system safety*, *117*, 98–113.

PRA Working Group (1994). *A Review of NRC Staff uses of Probabilistic Risk Assessment*. Technical Report NUREG-1489 U. S. Nuclear Regulatory Commission.

Pörn, K. (1996). The two-stage Bayesian method used for the T-Book application. *Reliability Engineering & System Safety*, *51*, 169 – 179.

Rausand, M. (2014). *Reliability of safety-critical systems: theory and applications*. Hoboken, New Jersey: Wiley & sons.

Rudin, W. (1976). *Principles of Mathematical Analysis*. International series in pure and applied mathematics (3rd ed.). McGraw-Hill.

Salako, K. (2020). Loss-Size and Reliability Trade-Offs Amongst Diverse Redundant Binary Classifiers. In M. Gribaudo, D. N. Jansen, & A. Remke (Eds.), *Quantitative Evaluation of Systems* (pp. 96–114). Cham: Springer International Publishing volume 12289 of *LNCS*.

Salako, K., Strigini, L., & Zhao, X. (2021). Conservative Confidence Bounds in Safety, from Generalised Claims of Improvement & Statistical Evidence. In *51st Annual IEEE/IFIP International Conference on Dependable Systems and Networks* DSN'21 (pp. 451–462). Taipei Taiwan: IEEE/IFIP.

Salako, K., & Zhao, X. (2023). The unnecessity of assuming statistically independent tests in bayesian software reliability assessments. *IEEE Transactions on Software Engineering*, *49*, 2829–2838.

Sha, L. (2001). Using simplicity to control complexity. *IEEE Software*, *18*, 20–28.

Singh, H., Cortellessa, V., Cukic, B., Gunel, E., & Bharadwaj, V. (2001). A Bayesian approach to reliability prediction and assessment of component based systems. In *Proc. 12th International Symposium on Software Reliability Engineering* (pp. 12–21).

Singpurwalla, N. D., & Wilson, S. P. (1994). Software reliability modeling. *International Statistical Review / Revue Internationale de Statistique*, *62*, 289–317.

Strigini, L. (1996). On testing process control software for reliability assessment: the effects of correlation between successive failures. *Software Testing, Verification and Reliability*, *6*, 33–48.

Strigini, L., & Littlewood, B. (1997). *Guidelines for Statistical Testing*. Technical Report (PASCON/WO6-CCN2/TN12) ESA/ESTEC project PASCON.

Strigini, L., & Povyakalo, A. (2013). Software fault-freeness and reliability predictions. In F. Bitsch, J. Guiochet, & M. Kaâniche (Eds.), *Computer Safety, Reliability, and Security* (pp. 106–117). Berlin, Heidelberg: Springer Berlin Heidelberg volume 8153 of *LNCS*.

Thayer, R. A., Lipow, M., & Nelson, E. C. (1978). *Software Reliability*. North-Holland.

Thomas E. Wierman, Scott T. Beck, Michael B. Calley, Steven A. Eide, Cindy D. Gentillon, & William E. Kohn (2001). *Reliability Study: Combustion Engineering Reactor Protection System, 1984–1998*. Technical Report NUREG/CR-5500 Vol.10 Idaho National Engineering and Environmental Laboratory U.S. Nuclear Regulatory Commission Washington, DC.

Tomek, L. A., Muppala, J. K., & Trivedi, K. S. (1993). Modeling correlation in software recovery blocks. *IEEE Transactions on Software Engineering*, *19*, 1071–1086.

Wood, R. T., Belles, R., Cetiner, M. S., Holcomb, D. E., Korsah, K., Loebl, A., Mays, G. T., Muhlheim, M. D., Mullens, J. A., Poore, W. P., III, Qualls, A. L., Wilson, T. L., & Waterman, M. E. (2010). *Diversity Strategies for Nuclear Power Plant Instrumentation and Control Systems*. Technical Report ORNL/TM-2009/302; NUREG/CR-7007 Oak Ridge National Lab. (ORNL).

Xie, M. (1991). *Software reliability modelling*. World Scientific.

Zhao, X., Littlewood, B., Povyakalo, A., Strigini, L., & Wright, D. (2017). Modeling the probability of failure on demand (pfd) of a 1-out-of-2 system in which one channel is "quasi-perfect". *Reliability Engineering & System Safety*, *158*, 230–245.

Zhao, X., Littlewood, B., Povyakalo, A., Strigini, L., & Wright, D. (2018). Conservative claims for the probability of perfection of a software-based system using operational experience of previous similar systems. *Reliability Engineering & System Safety*, *175*, 265 – 282.

Zhao, X., Littlewood, B., Povyakalo, A., & Wright, D. (2015). Conservative claims about the probability of perfection of software-based systems. In *26th Int. Symp. on Software Reliability Eng.* (pp. 130–140). IEEE.

Zhao, X., Robu, V., Flynn, D., Salako, K., & Strigini, L. (2019). Assessing the Safety and Reliability of Autonomous Vehicles from Road Testing. In *the 30th Int. Symp. on Software Reliability Engineering* (pp. 13–23). Berlin, Germany: IEEE.

Zhao, X., Salako, K., Strigini, L., Robu, V., & Flynn, D. (2020). Assessing safety-critical systems from operational testing: A study on autonomous vehicles. *Information and Software Technology*, *128*, 106393.

Kizito Salako: is a Lecturer at the department of computer science, City, University of London. He holds a double honours (1st-class) degree in mathematics and statistics from the University of Lagos; a Master of advanced study in mathematics degree from the University of Cambridge (where he was both a Shell Centenary scholar and a Commonwealth scholar); and a PhD in computer science from City, university of London. Kizito is passionate about applications of probability theory, Bayesian statistics, geometry and machine-learning, when simulating, assessing and forecasting the (failure) behaviour of software-based systems. His research produces statistical techniques that support conservative dependability claims for safety-critical systems. He also builds simulations of large-scale, complex, interdependent, critical infrastructure, in order to forecast the occurrence and impact of cascading failures, cyber-attacks and the efficacy of mitigation strategies.

Xingyu Zhao: is a Assistant Professor in Safety-Critical Systems at WMG, University of Warwick. After receiving Bachelor and Masters degrees from the Beihang University, he earned a PhD in computer science at the Centre for Software Reliability, City, University of London in 2017. His research expertise covers probabilistic verification of autonomous systems, Bayesian inference with partial/vague prior knowledge, reliability assessment and safety assurance, and trustworthy AI. He has published 40+ papers in interdisciplinary fields of software engineering, AI, and system safety and reliability. Beyond publications, he has secured funding from UK EPSRC, UK DSTL and Innovate UK as a co-Investigator.

A. Conservative Bayesian Assessment

A.1. Klotz Failure-Model Likelihood Function

The Klotz failure-model is naturally expressed in coordinates (x, λ), where x is the Bernoulli frequency parameter and λ is the model's correlation parameter. If λ^* denotes $\max\left\{0, \frac{2x-1}{x}\right\}$, the Klotz model likelihood function is well-defined[16] over the region $\mathcal{R}$ defined by $0 \leqslant x \leqslant 1$, $\lambda^* \leqslant \lambda \leqslant 1$ (depicted in Fig.s 4a and 14a). $\mathcal{R}$ ensures the likelihood's magnitude is no greater than 1. The likelihood has two primary forms:

$$L(x, \lambda; \alpha, \beta, \gamma, \delta) = \begin{cases} x\left(\frac{(1-\lambda)x}{1-x}\right)^{\alpha}\left(1 - \frac{(1-\lambda)x}{1-x}\right)^{\beta}\lambda^{\gamma}(1-\lambda)^{\delta}; \\ \text{when the 1st execution is a failure} \\ \\ (1-x)\left(\frac{(1-\lambda)x}{1-x}\right)^{\alpha}\left(1 - \frac{(1-\lambda)x}{1-x}\right)^{\beta}\lambda^{\gamma}(1-\lambda)^{\delta}; \\ \text{when the 1st execution is a success} \end{cases} \tag{5}$$

where the greek exponents in the likelihood are fixed by the outcomes of n system executions and satisfy $\alpha, \beta, \gamma, \delta \geqslant 0$.

In practice, the Klotz likelihood is used as follows. Consider a system executing n demands, with s failed executions, and r of these failures being *consecutive failures* – i.e. when a failure immediately follows a previous failure. Then, the Klotz likelihood becomes one of the following four expressions, depending on the particular values of the greek exponents in (5). That is, depending on whether the n executions,

1) begin with a failure and end with a failure:

$$x\left(\frac{(1-\lambda)x}{1-x}\right)^{s-r-1}\lambda^{r}(1-\lambda)^{s-r-1}\left(1 - \frac{(1-\lambda)x}{1-x}\right)^{n-2s+r+1} \tag{6}$$

when $\alpha = s - r - 1$, $\beta = n - 2s + r + 1$, $\gamma = r$ and $\delta = s - r - 1$;

2) begin with a success and end with a failure:

$$(1-x)\left(\frac{(1-\lambda)x}{1-x}\right)^{s-r}\lambda^{r}(1-\lambda)^{s-r-1}\left(1 - \frac{(1-\lambda)x}{1-x}\right)^{n-2s+r} \tag{7}$$

when $\alpha = s - r$, $\beta = n - 2s + r$, $\gamma = r$ and $\delta = s - r - 1$;

3) begin with a success and end with a success:

$$(1-x)\left(\frac{(1-\lambda)x}{1-x}\right)^{s-r}\lambda^{r}(1-\lambda)^{s-r}\left(1 - \frac{(1-\lambda)x}{1-x}\right)^{n-2s+r-1} \tag{8}$$

when $\alpha = s - r$, $\beta = n - 2s + r - 1$, $\gamma = r$ and $\delta = s - r$;

4) begin with a failure and end with a success:

$$x\left(\frac{(1-\lambda)x}{1-x}\right)^{s-r-1}\lambda^{r}(1-\lambda)^{s-r}\left(1 - \frac{(1-\lambda)x}{1-x}\right)^{n-2s+r} \tag{9}$$
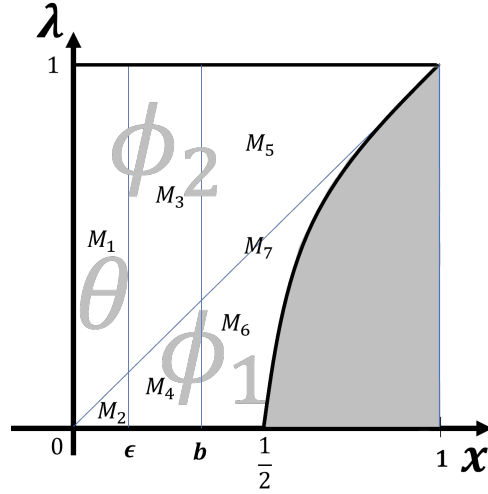
when $\alpha = s - r - 1$, $\beta = n - 2s + r$, $\gamma = r$ and $\delta = s - r$.

In alternative coordinates (λ, y), defined by the transformation $\lambda = \lambda$ and $y = \frac{(1-\lambda)x}{1-x}$ from (x, λ) coordinates, the Klotz likelihood (5) has the equivalent forms:
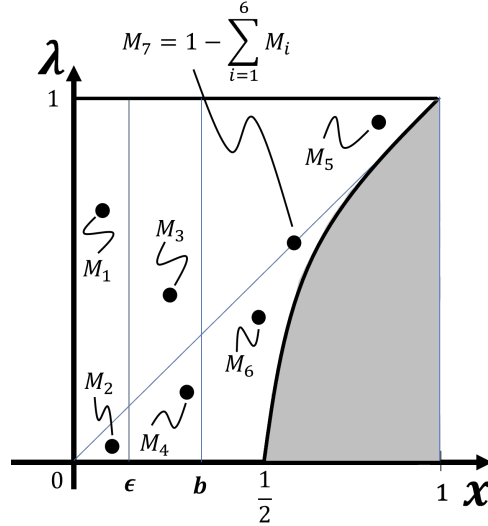
$$L(\lambda, y; \alpha, \beta, \gamma, \delta) = \begin{cases} \frac{y}{y+1-\lambda}y^{\alpha}(1-y)^{\beta}\lambda^{\gamma}(1-\lambda)^{\delta}; \\ \text{when the 1st execution is a failure} \\ \\ \frac{1-\lambda}{y+1-\lambda}y^{\alpha}(1-y)^{\beta}\lambda^{\gamma}(1-\lambda)^{\delta}; \\ \text{when the 1st execution is a success} \end{cases} \tag{10}$$

In what follows, we will use the likelihood function in either (x, λ) or (λ, y) coordinates (whichever is more convenient to work with), for suitable α, β, γ and δ.

---

[16]Strictly speaking, the likelihood function in (5) is not well-defined at the point $(1, 1)$ except for appropriate ranges of the greek exponents. The likelihood's value at this point is 0, and thus well-defined, if either the demand executions begin with a success, or $\delta > 0$ (so failure is not an absorbing state). These sufficient conditions are satisfied in all practical scenarios of interest.

(a) A feasible joint distribution of (X, Λ), over the region $\mathcal{R}$, allocates probability masses $\{M_i\}$ over $\mathcal{R}$ subsets. The masses must satisfy $M_1 + M_3 + M_5 = \Phi_2$, $M_2 + M_4 + M_6 = \Phi_1$, and probability $M_7 = 1 - \sum_{i=1}^{6} M_i$ is allocated to the diagonal.

(b) A discrete joint prior distribution over $\mathcal{R}$. It allocates probability masses M_i to single points within each subset and along the diagonal.

Figure 14:

A.2. Posterior Confidence in a Bound on the Probability of Failure per Execution

We prove the following theorem.

Theorem 4. *Let $\mathcal{D}$ be the set of all prior distributions over the region $\mathcal{R}$ and let $0 \leqslant \epsilon < b < \frac{1}{2}$ (see Fig. 14a). The optimisation problem*

$$\inf_{\mathcal{D}} P(X \leqslant b \mid \text{outcomes of n executions})$$

$$s.t. \quad P(\Lambda < X) = \phi_1, \quad P(\Lambda > X) = \phi_2, \quad P(X \leqslant \epsilon) = \theta$$

is solved by prior distributions such as those in Fig.s 21 – 24, since $P(X < b \mid \text{outcomes of n executions})$ from these priors (for $p_l = 0$) equals the infimum.

Proof. The optimisation constraints can be used to partition $\mathcal{R}$ into 7 disjoint subsets (with one of the subsets being the 45° diagonal). Each prior distribution $F \in \mathcal{D}$ must assign 7 probabilities $\{M_i\}_{i=1}^{7}$ to these subsets, in such a way as to satisfy the constraints of the optimisation problem (see Fig. 14a).

The proof progresses in 6 stages:

1. restrict the optimisation from $\mathcal{D}$ to its subset $\mathcal{D}'$ of discrete prior distributions. An arbitrary discrete prior assigns its probabilities M_i to 7 arbitrary points $\{(x_i, \lambda_i)\}_{i=1}^{7}$ within $\mathcal{R}$. Hence, the objective function becomes a rational function of the "x_i"s, "λ_i"s and "M_i"s;

2. show that the gradient of this objective function is determined by the gradient of the Klotz model likelihood;

3. show the likelihood is unimodal along vertical and horizontal lines in $\mathcal{R}$, as well as along the 45° diagonal line;

4. show that the likelihood is also unimodal over all of $\mathcal{R}$, and it attains its maximum either at a stationary point in the interior of $\mathcal{R}$, or along the boundary of $\mathcal{R}$;

5. the previous steps in the proof imply the following: starting from any $F \in \mathcal{D}'$, and the probabilities $\{M_i\}$ assigned by F, we can construct a new prior that gives a smaller value for the objective function (compared with F's objective function value). We simply use the gradient of the likelihood to determine new locations within each of the 7 $\mathcal{R}$-subsets, and reassign the "M_i"s to these new locations. This reassignment produces a new prior distribution, which in turn can have *its* probabilities reassigned to new points (and so on, indefinitely). In the limit, depending on the values of α, β, γ and δ, the sequence of new points obtained by successive reassignments will converge to *limit points*[17] in each $\mathcal{R}$-subset. That is, the objective function values converge in a monotonically decreasing manner, as the sequence of "reassigned" priors converge to a limiting distribution with support at, no more than, 7 *limit points*;

6. finally, determine the values for the "M_i"s that a limiting distribution should assign to *limit points* – to ensure that the related sequence of objective function values converge to the infimum. Determining these worst-case "M_i"s is a constrained *linear fractional programming* problem. One may solve this either numerically, or by a logical allocation of probability masses to the relevant *limit points* in $\mathcal{R}$. For the CBI solutions in this paper, we use the latter approach. These final forms of limiting distribution (illustrated in Fig.s $21 - 24$) are worst-case prior distributions; so-called because $P(X < b \mid outcomes\ of\ n\ executions)$ for these distributions equals the infimum we seek.

Let us proceed with the proof:

stage 1) By definition, for any $F \in \mathcal{D}$,

$$P(X \leqslant b \mid outcomes\ of\ n\ executions) = \frac{\mathbb{E}[L(X, \Lambda; \alpha, \beta, \gamma, \delta)\mathbf{1}_{X \leqslant b}]}{\mathbb{E}[L(X, \Lambda; \alpha, \beta, \gamma, \delta)]} = \frac{\int_{[0,b] \times [\lambda^*, 1]} L(x, \lambda; \alpha, \beta, \gamma, \delta)\, \mathrm{d}F(x, \lambda)}{\int_{[0,1] \times [\lambda^*, 1]} L(x, \lambda; \alpha, \beta, \gamma, \delta)\, \mathrm{d}F(x, \lambda)}$$

However, the set $\mathcal{D}$ can be restricted to the subset $\mathcal{D}'$ of discrete joint distributions – i.e., to those distributions that assign their "M_i"s to single points within each $\mathcal{R}$ subset (e.g. see Fig. 14b), see Moreno & Cano (1991). So that, for any $F \in \mathcal{D}'$, the objective function of the optimisation becomes

$$\begin{aligned}
P(X \leqslant b \mid outcomes\ of\ n\ executions) &= \frac{\int_{[0,b] \times [\lambda^*, 1]} L(x, \lambda; \alpha, \beta, \gamma, \delta)\, \mathrm{d}F(x, \lambda)}{\int_{[0,1] \times [\lambda^*, 1]} L(x, \lambda; \alpha, \beta, \gamma, \delta)\, \mathrm{d}F(x, \lambda)} \\
&= \frac{\sum_{i=1}^{7} L(x_i, \lambda_i; \alpha, \beta, \gamma, \delta)\mathbf{1}_{x_i \leqslant b}\, M_i}{\sum_{i=1}^{7} L(x_i, \lambda_i; \alpha, \beta, \gamma, \delta)\, M_i} \\
&= \frac{Num}{Denum}
\end{aligned}$$

Consequently the objective function has become $\frac{Num}{Denum}$; a rational function of the "x_i"s, "λ_i"s and "M_i"s.

---

[17]Definition: for a given topology (e.g., the "open balls" topology associated with 2D Euclidean space), a *limit point* of a subset of the plane is a point that is arbitrarily well-approximated by sequences of points within the subset (see Rudin (1976); Copson (1968)).

stage 2) Consider how this objective function changes when restricted to a vertical line in the subset of $\mathcal{R}$ where $x \leqslant \frac{1}{2}$. The rate of change of $\frac{Num}{Denum}$ with respect to λ is then

$$\frac{\partial}{\partial \lambda} \left(\frac{Num}{Denum} \right) = \frac{\frac{\partial}{\partial \lambda} Denum}{Denum} \left(\frac{\frac{\partial}{\partial \lambda} Num}{\frac{\partial}{\partial \lambda} Denum} - \frac{Num}{Denum} \right) \tag{11}$$

Since $\frac{Num}{Denum}$ is a rational function of λ, it is smooth (except where $Denum = 0$). Consequently, the sign of $\frac{\partial}{\partial \lambda} \left(\frac{Num}{Denum} \right)$ indicates how to move the location of the "M_i"s along vertical lines in each $\mathcal{R}$ subset, in order to minimise $\frac{Num}{Denum}$.

The following argument shows how the sign of $\frac{\partial}{\partial \lambda} \left(\frac{Num}{Denum} \right)$ is determined by the sign of $\frac{\partial}{\partial \lambda} L(x, \lambda; \alpha, \beta, \gamma, \delta)$ and the size of x. Observe that $\frac{\partial}{\partial \lambda} \left(\frac{Num}{Denum} \right) \geqslant 0$ *if, and only if,* $\frac{\partial}{\partial \lambda} Denum$ and $\left(\frac{\frac{\partial}{\partial \lambda} Num}{\frac{\partial}{\partial \lambda} Denum} - \frac{Num}{Denum} \right)$ have the same sign. When the $\mathcal{R}$ region satisfies $x \leqslant b$, this implies $\frac{\frac{\partial}{\partial \lambda} Num}{\frac{\partial}{\partial \lambda} Denum} = 1$ for λ from that region. Substituting 1 for $\frac{\frac{\partial}{\partial \lambda} Num}{\frac{\partial}{\partial \lambda} Denum}$ in (11), and noting that the objective function $\frac{Num}{Denum}$ is a probability (hence must lie between 0 and 1), we have that $\frac{\partial}{\partial \lambda} \left(\frac{Num}{Denum} \right)$ and $\frac{\partial}{\partial \lambda} Denum$ share the same sign when $x \leqslant b$. When $x \geqslant b$ instead, $\frac{\partial}{\partial \lambda} \left(\frac{Num}{Denum} \right)$ and $\frac{\partial}{\partial \lambda} Denum$ have opposite signs[18]. So, the sign of $\frac{\partial}{\partial \lambda} Denum$ (i.e. the sign of $\frac{\partial}{\partial \lambda} L(x, \lambda; \alpha, \beta, \gamma, \delta)$), and the value of x, together determine the sign of $\frac{\partial}{\partial \lambda} \left(\frac{Num}{Denum} \right)$. A similar argument shows that along horizontal lines in $\mathcal{R}$, $\frac{\partial}{\partial x} L(x, \lambda; \alpha, \beta, \gamma, \delta)$ and x's value determine the sign of $\frac{\partial}{\partial x} \left(\frac{Num}{Denum} \right)$. And thus, they determine where the "M_i"s should be allocated to minimize $\frac{Num}{Denum}$ along that line.

stage 3) Along any vertical line in $\mathcal{R}$ where $x \leqslant \frac{1}{2}$ is satisfied, $L(x, \lambda; \alpha, \beta, \gamma, \delta)$ is a non-negative unimodal function of λ. To see this, note that $\frac{\partial}{\partial \lambda} L(x, \lambda; \alpha, \beta, \gamma, \delta) = 0$ has non-trivial solutions at λ values where two quadratic functions of λ intersect. That is, solutions to

$$\left(1 - \frac{(1-\lambda)}{1-x} x \right) (\gamma - \lambda(\gamma + \delta)) = \lambda \left(\alpha - (\alpha + \beta) \frac{(1-\lambda)}{1-x} x \right) \tag{12}$$

An illustration of these two functions is given in Fig. 15. One function has two roots of opposite sign (at $\lambda = \frac{2x-1}{x}, \frac{\gamma}{\gamma+\delta}$) and a maximum, while the other function has a root at $\lambda = 0$ and a minimum. This means at least one solution to (12) cannot lie within $\mathcal{R}$ – it must be non-positive. And the other solution must be positive and represents a maximum turning point. Because the l.h.s of (12) is bigger than the r.h.s. for λ values slightly smaller than the positive solution, and the l.h.s. is smaller than the r.h.s. for all λ values bigger than the positive solution.

Thus, as λ grows from 0 to 1 along any vertical line in $\mathcal{R}$ (where $x \leqslant \frac{1}{2}$), there is (at most) one stationary point at which the likelihood is maximum. The likelihood is monotonic on either side of this maximum along the vertical line.

$L(x, \lambda; \alpha, \beta, \gamma, \delta)$ is also unimodal along the 45° diagonal (i.e., when $x = \lambda$). Because it has only one non-trivial stationary point[19], and this must be a maximum since the likelihood is non-negative with value 0 at the endpoints of the diagonal.

Analogously, $L(x, \lambda; \alpha, \beta, \gamma, \delta)$ is unimodal along any horizontal line within $\mathcal{R}$, since it has (at most) one stationary point at which it attains a maximum. The stationary point solves $\frac{\partial}{\partial x} L(x, \lambda; \alpha, \beta, \gamma, \delta) = 0$ non-trivially. Equivalently, the stationary point satisfies the leftmost intersection between a straight line and a quadratic function in x (see Fig. 16). This leftmost intersection must occur to the left of $x = \frac{1}{2-\lambda}$, ensuring that the stationary point lies in $\mathcal{R}$. The fact that the line lies above the quadratic before this intersection, and then below the quadratic immediately after, ensures that, as x increases from 0, the $\frac{\partial}{\partial x} L(x, \lambda; \alpha, \beta, \gamma, \delta)$ transitions from being positive to being negative. That is, the stationary point is a maximum.

stage 4) Finally, the likelihood either has a single stationary point within $\mathcal{R}$ at which it attains a maximum value over $\mathcal{R}$, or it attains its maximum value over $\mathcal{R}$ on the boundary of $\mathcal{R}$. When the single stationary point lies within

---

[18]These statements exclude the unimportant edge cases when $\frac{Num}{Denum} = 0, 1$.

[19]This stationary point is located at either $x = \frac{1+\alpha+\gamma}{1+\alpha+\gamma+\beta+\delta}$ or $x = \frac{\alpha+\gamma}{1+\alpha+\gamma+\beta+\delta}$, depending on whether executions begin with a failure or success respectively.
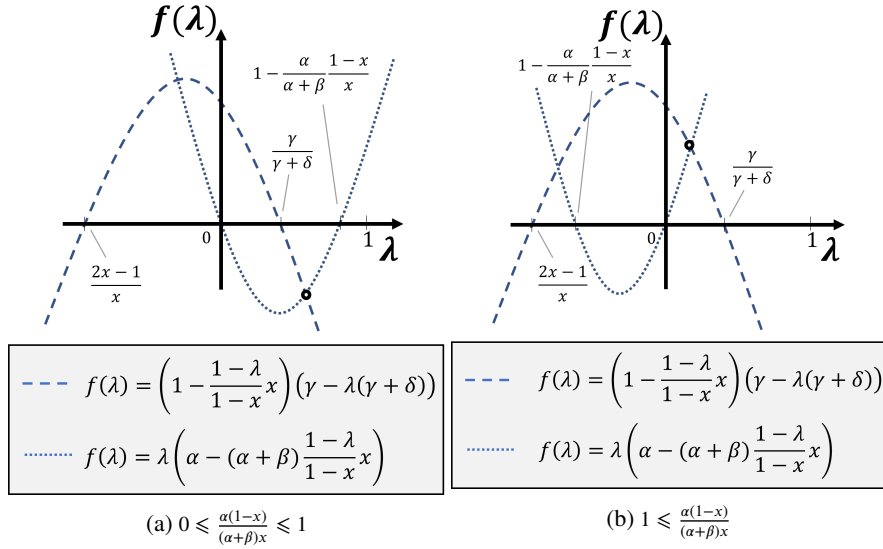
Figure 15: Two illustrations of two quadratic functions of λ having, at most, one intersection over the range $0 < \lambda < 1$. For fixed x, this geometric fact implies $L(x, \lambda; \alpha, \beta, \gamma, \delta)$ is unimodal over any vertical line in $\mathcal{R}$ such that $x \leqslant \frac{1}{2}$.
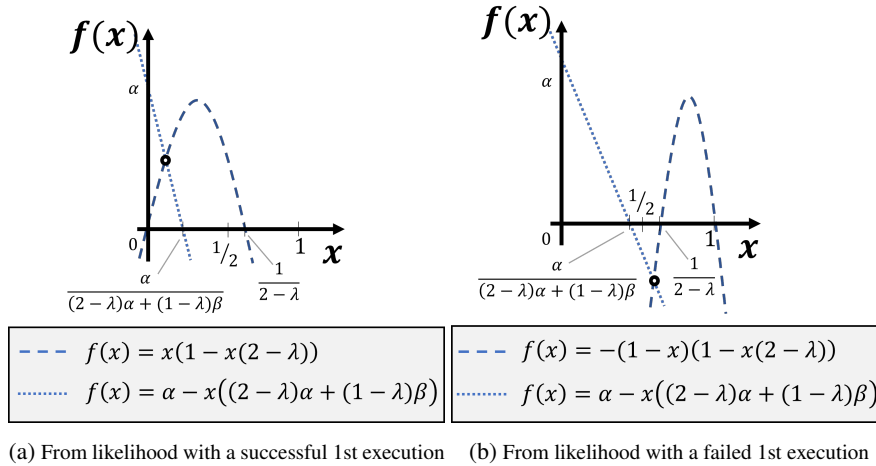

Figure 16: Two illustrations of quadratic and linear functions of x having, at most, one intersection over the range $0 < x \leqslant \frac{1}{2}$. For fixed λ, this geometric fact implies $L(x, \lambda; \alpha, \beta, \gamma, \delta)$ is unimodal over any horizontal line in $\mathcal{R}$.

$\mathcal{R}$, it can be determined by solving $\frac{\partial}{\partial \lambda} L(\lambda, y; \alpha, \beta, \gamma, \delta) = 0$ and $\frac{\partial}{\partial y} L(\lambda, y; \alpha, \beta, \gamma, \delta) = 0$ simultaneously for λ and y. The non-trivial solutions for this simultaneous system of equations are given by the intersections of pairs of curves, as illustrated in Fig. 17.

If the stationary point lies within $\mathcal{R}$ then it must be a maximum; because the stationary curves in Fig. 17 imply that, from any point along the boundary of $\mathcal{R}$, we can always move away from that point along an appropriate path within $\mathcal{R}$ to increase the likelihood's value.

stage 5) stages 1-4 of this proof demonstrate the existence and uniqueness of locations in $\mathcal{R}$ that are local or global maxima, as exemplified in Fig. 18. For the region $x \leqslant b$ in $\mathcal{R}$, the "further away" from maxima the locations a prior assigns probabilities to, the smaller the objective function. For $x > b$, the "closer" the nonzero probability locations are to the maxima, the smaller the objective function. Here, "further away" and "closer" are in terms of the Klotz likelihood's gradients.

(a) From likelihood with a successful 1st execution, $\gamma + \delta \neq 0$ and $\alpha + \beta \neq 1, 0$

(b) The stationary curves of Fig. 17a depicted in $\mathcal{R}$.

(c) From likelihood with a failed 1st execution, $\alpha + \beta \neq 0$ and $\gamma + \delta \neq 1, 0$

(d) The stationary curves of Fig. 17c depicted in $\mathcal{R}$.
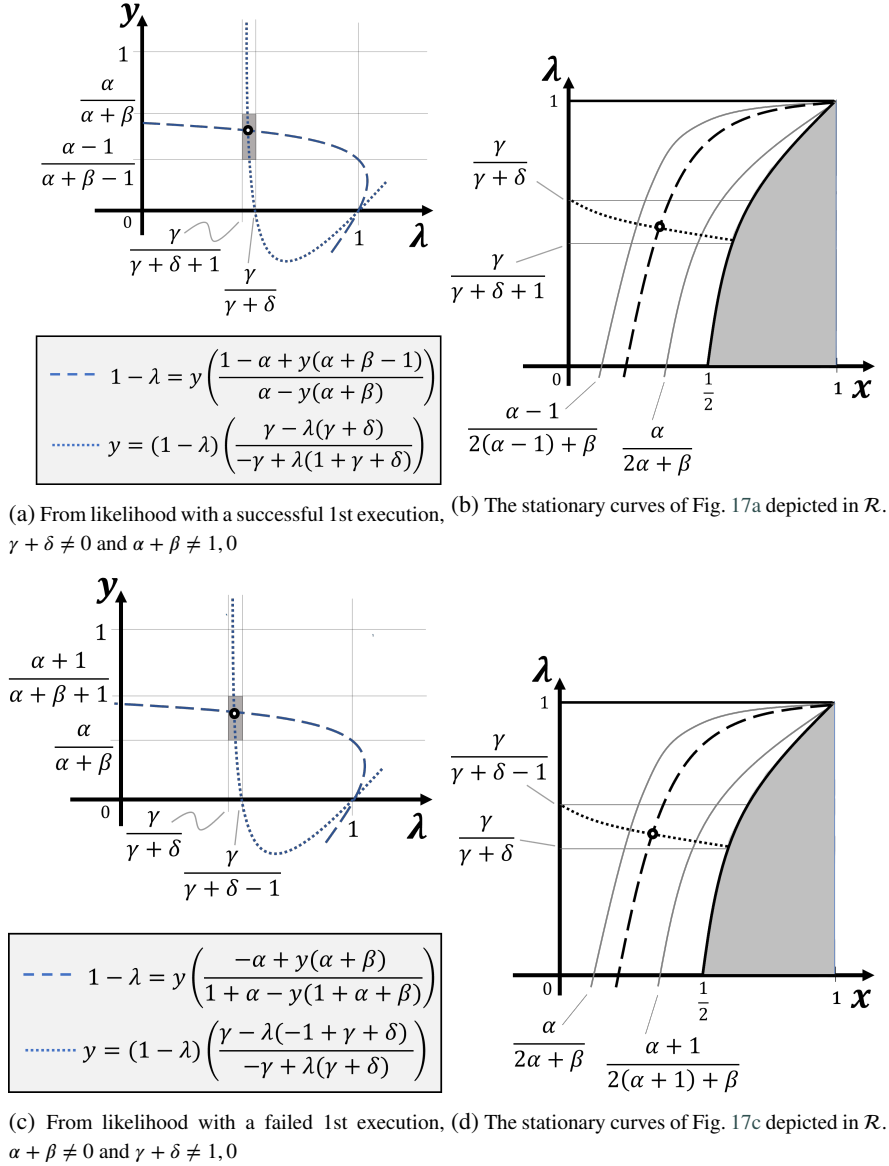
Figure 17: Two illustrations of pairs of curves intersecting, at most, once, over the range $0 < \lambda < 1, 0 < y < 1$. This geometric fact implies $L(x, \lambda; \alpha, \beta, \gamma, \delta)$ is unimodal over $\mathcal{R}$ with, at most, one stationary point in the interior of $\mathcal{R}$.

That is, given any $F \in \mathcal{D}'$, we can reassign the probabilities $\{M_i\}$ that F allocates to points in $\mathcal{R}$, to new points suggested by the likelihood's gradients – resulting in a new prior with a smaller objective function value. Such reassignments can be carried out indefinitely, creating a sequence of priors with an associated, monotonically decreasing sequence of objective function values. And the *completeness of the real numbers* guarantees that this sequence of objective function values converge[20]. Being discrete distributions, it is also clear that these reassigned priors, themselves, converge to some limiting discrete distribution. Examples of limiting distributions converged to in this manner are illustrated in Fig. 20. The points in each subfigure indicated by black dots are the limits of the sequences of new points chosen for reassignments – so-called *limit points*.

---

[20]Note that the objective function is a probability, and is therefore bounded.

(a) A successful 1st execution
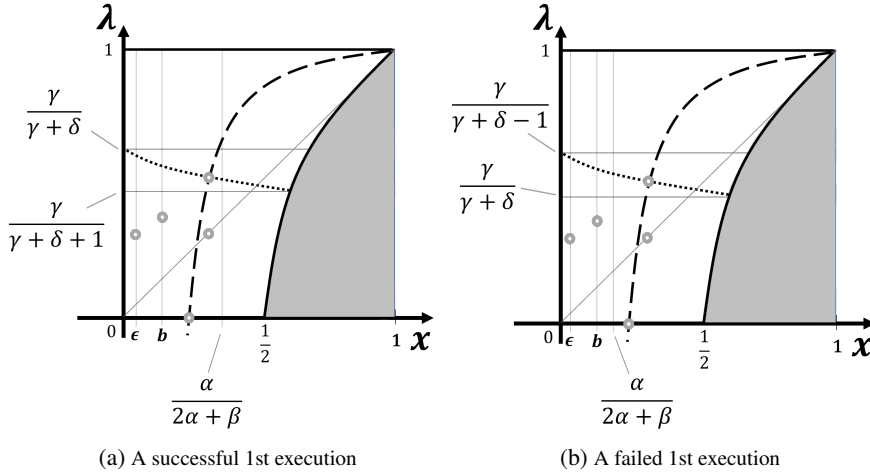
(b) A failed 1st execution

Figure 18: Examples of locations in $\mathcal{R}$ (indicated by grey circles) at which local and global maxima of the Klotz likelihood occur. Here, $\alpha, \beta, \gamma, \delta \geqslant 1$.

stage 6) So, the limiting distributions assign probabilities only to certain limit points of the 7 $\mathcal{R}$-subsets. The exact values of the probabilities will depend on which initial prior F (with its probabilities $\{M_i\}$) was chosen to create the "reassigned" priors sequence. To determine those values for the "M_i"s that ensure the sequence of objective function values converges to the infimum, one can systematically allocate probability masses to the limit points. We will now illustrate this, and show how the priors (when $p_l = 0$) in Fig.s 21a, 21b, 22a, 22b, 23a, 23b and 24 can be obtained from the limiting distribution forms in Fig. 20. All of the priors in Fig.s $21 - 24$ (when $p_l > 0$) are similarly obtained.

For example, consider the limit points in Fig.20b, for the case when $\phi_1 \geqslant \theta$ and no failures are observed. Focus on the subset $x \leqslant \epsilon$ and recall the requirement $P(X \leqslant \epsilon) = \theta$. To be pessimistic, we must allocate probability θ to those limit points within this subset at which the likelihood is smallest. This is the limit point $(\epsilon, 0)$. Since $P(X \leqslant \Lambda) = \phi_1 \geqslant \theta$, all of the θ probability can be allocated to this limit point "from below" the 45° diagonal and "from the left" of the line $x = \epsilon$ (see Fig. 19a). Consequently, because $P(X \leqslant \epsilon) = \theta$, no more probability can be allocated to any other limit points in $x \leqslant \epsilon$.
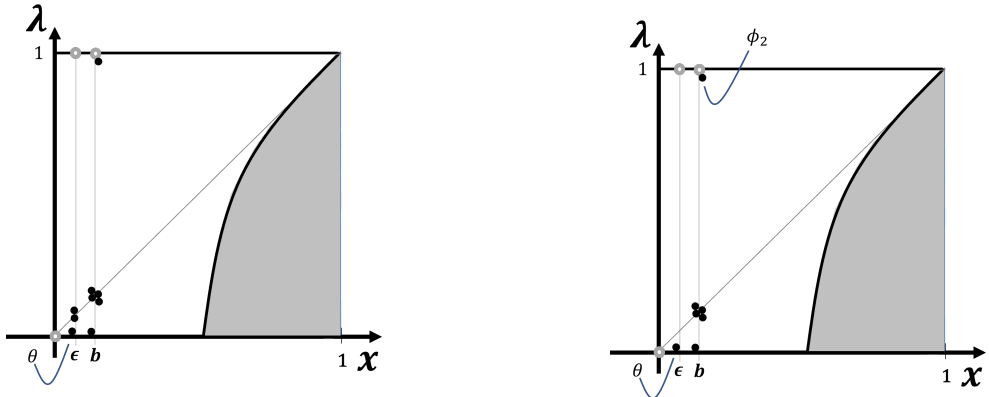
Now we need to assign probability $1 - \theta$ to the remaining limit points in $\mathcal{R}$. There are two alternative limit points above the diagonal where we may assign the ϕ_2 probability. Assigning to the point $(b, 1)$ gives more pessimistic results than assigning to (b, b). We can see this by sharing the ϕ_2 probability between the two points, and noting that the objective function monotonically decreases as the amount of ϕ_2 allocated to $(b, 1)$ increases. In effect, all of ϕ_2 should be allocated to any sequence of points that approximate $(b, 1)$ arbitrarily-well, "from the right" of the line $x = b$. This justifies Fig. 19b. Similar reasoning shows that allocating probability $\phi_1 - \theta$ to the point (b, b), "from the right" of $x = b$, is more pessimistic than allocating it to $(b, 0)$ "from the left". Thus justifying Fig. 19c. Note that these allocations are possible and do not violate the constraints, because $1 - \phi_1 - \phi_2 \geqslant 0$ and $\phi_1 \geqslant \theta$ imply $0 \leqslant \phi_1 - \theta + \phi_2 \leqslant 1 - \theta$.

Finally, using similar "approximation"-based reasoning to how ϕ_2 and $\phi_1 - \theta$ were allocated, we must assign the remaining probability $1 - \phi_1 - \phi_2$ to the point (b, b). Via any sequence of points that approximate (b, b) arbitrarily-well "from the right", along the diagonal (see Fig. 19d).

Note that probabilities were assigned to limit points that lie along the line $x = b$, but only by assigning the probabilities to points that approximate these limit points "from the right" of the line $x = b$. Consequently, our final limiting distribution gives the value of the infimum in the optimisation problem, but only by computing "$P(X < b \mid \dots)$" for this distribution, and not by computing "$P(X \leqslant b \mid \dots)$".
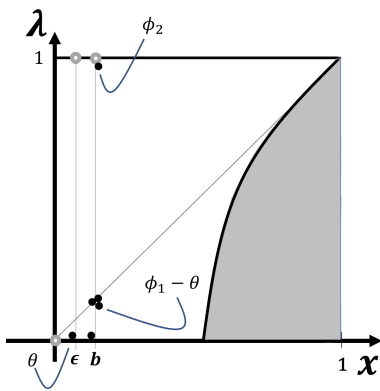
Using similar arguments to allocate probabilities to limit points, all of the remaining worst-case distributions in Fig.s $21 - 24$ are constructed from limit points analogous to those in Fig. 20. ∎

Remarks : with very few modifications, the foregoing arguments can be used to derive worst-case prior distributions subject to the additional constraint of PK1, i.e. $P(X \geqslant p_l) = 1$. Such priors solve the optimisation problem in Theorem 3. Indeed, after observing n executions of a system (which include some consecutive, failed executions),
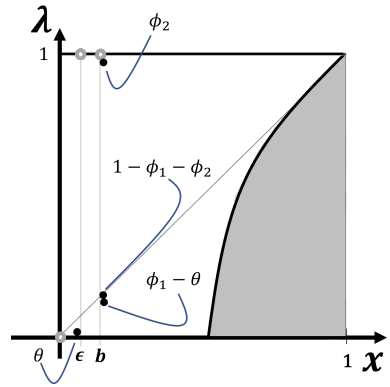
(a) For $x \leqslant \epsilon$, assign θ to limit point where likelihood is smallest, i.e. $(\epsilon, 0)$.

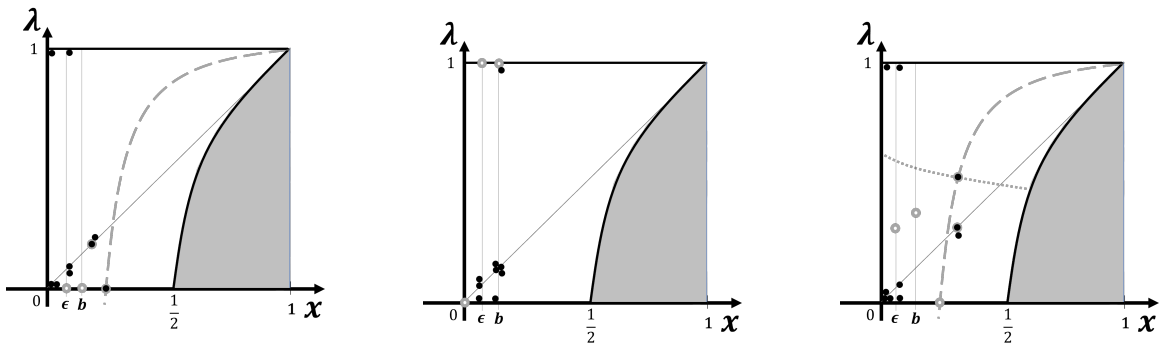(b) Assign ϕ_2 to limit point where $x \geqslant b$, i.e. $(b, 1)$.

(c) Assign $\phi_1 - \theta$ to limit point where $x \geqslant b$, i.e. (b, b).

(d) Assign $1 - \phi_1 - \phi_2$ to limit point on diagonal where $x \geqslant b$, i.e. (b, b).

Figure 19: A systematic allocation of probabilities to limit points, that demonstrates how Fig. 24a is obtained from Fig.20b.

(a) No consecutive failures (so $\gamma = 0$ and $\alpha, \beta, \delta \geqslant 1$)

(b) No failures (so $\alpha, \gamma, \delta = 0$)

(c) No restrictions on failures (so $\alpha, \beta, \gamma, \delta \geqslant 1$)

Figure 20: Examples of 3 limiting distributions for sequences of prior distributions (in D') that give progressively smaller posterior confidence in the failure-rate bound b. These distributions must allocate mass only at certain limit points of each $\mathcal{R}$-subset, as indicated by the black circles. Some relevant stationary points in $\mathcal{R}$ are also indicated as grey circles.

figure 21 illustrates 2 groups of priors (4 priors in each group) that give the smallest posterior confidence in the system's unknown *pfe* being no worse than the bound b. These solutions are subject to PKs 1, 2, 3 and 4.

Figure 21: Worst case prior distributions that solve the optimisation problem in Theorem 3 when consecutive failures are observed (i.e. $r > 0$). It's important to note the following: the precise locations of the "black dots" for each such distribution are determined by 1) the values of α, β, γ and δ, 2) whether the first execution is a success or failure, and 3) the indicated parameter ranges in each subfigure. The location (x^*, λ^*) of the global maximum for the Klotz likelihood is indicated by the grey circle. The 4 priors, illustrated in subfigures 21a, 21c, 21e and 21g, are solutions when $\phi_2 \geqslant 1 - \theta$. While the priors in 21b, 21d, 21f and 21h solve the problem when $\phi_2 \leqslant 1 - \theta$ and $\phi_1 \leqslant \theta$. These solutions assume $\alpha, \beta, \gamma, \delta > 0$.

(a) $\phi_1 \leqslant 1 - \theta$ and $\phi_2 \leqslant \theta$

(b) $\phi_1 \geqslant 1 - \theta$

(c) $\phi_1 \leqslant 1 - \theta$ and $\phi_2 \leqslant \theta$

(d) $\phi_1 \geqslant 1 - \theta$

(e) $L(p_l, p_l; \alpha, \ldots) \leqslant L(\epsilon, \epsilon; \alpha, \ldots)$ and $\phi_1 \leqslant 1 - \theta$ and $\phi_2 \leqslant \theta$

(f) $L(p_l, p_l; \alpha, \ldots) \leqslant L(\epsilon, \epsilon; \alpha, \ldots)$ and $\phi_1 \geqslant 1 - \theta$

(g) $L(p_l, p_l; \alpha, \ldots) \geqslant L(\epsilon, \epsilon; \alpha, \ldots)$ and $\phi_1 \leqslant 1 - \theta$ and $\phi_2 \leqslant \theta$

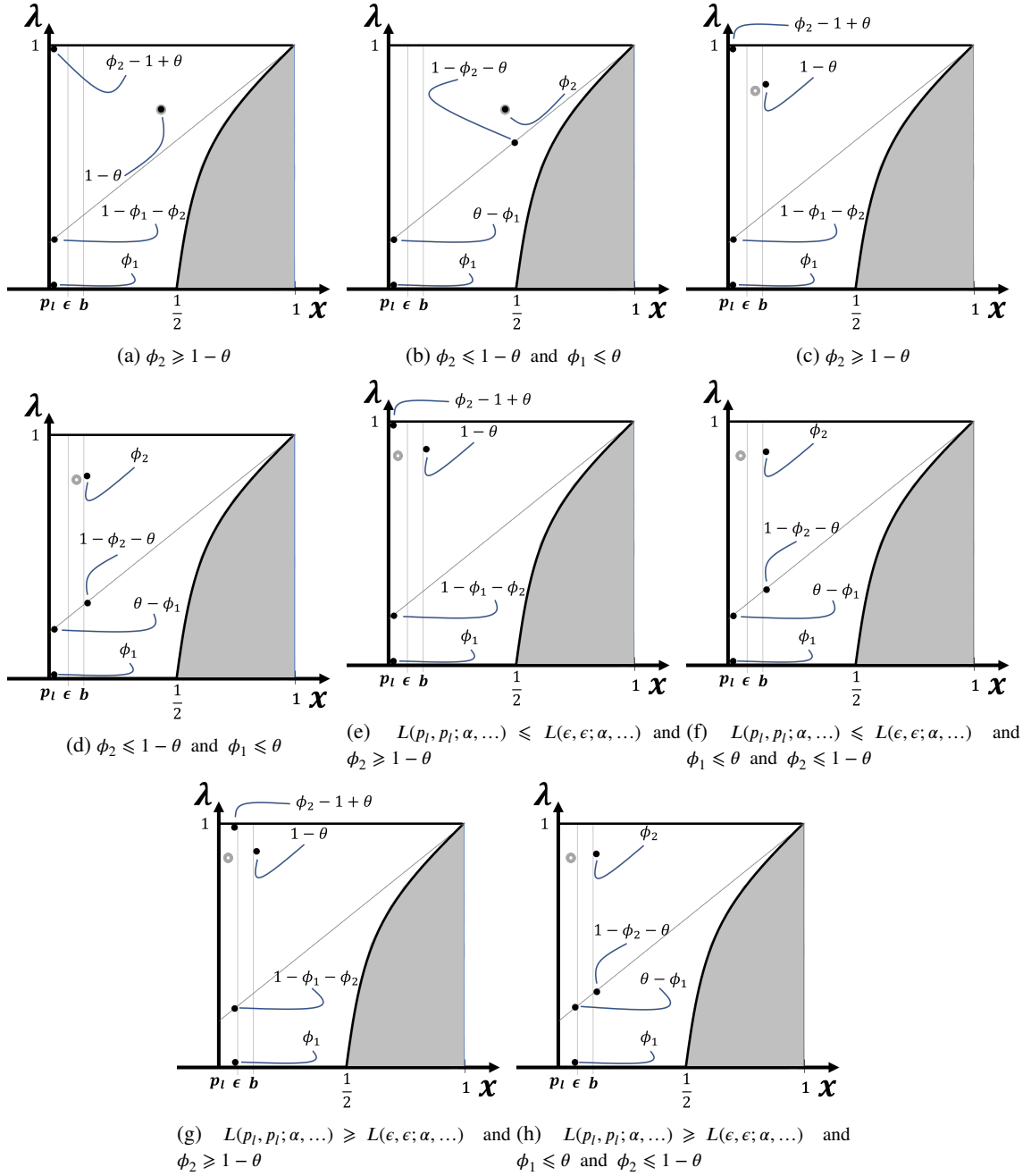(h) $L(p_l, p_l; \alpha, \ldots) \geqslant L(\epsilon, \epsilon; \alpha, \ldots)$ and $\phi_1 \geqslant 1 - \theta$
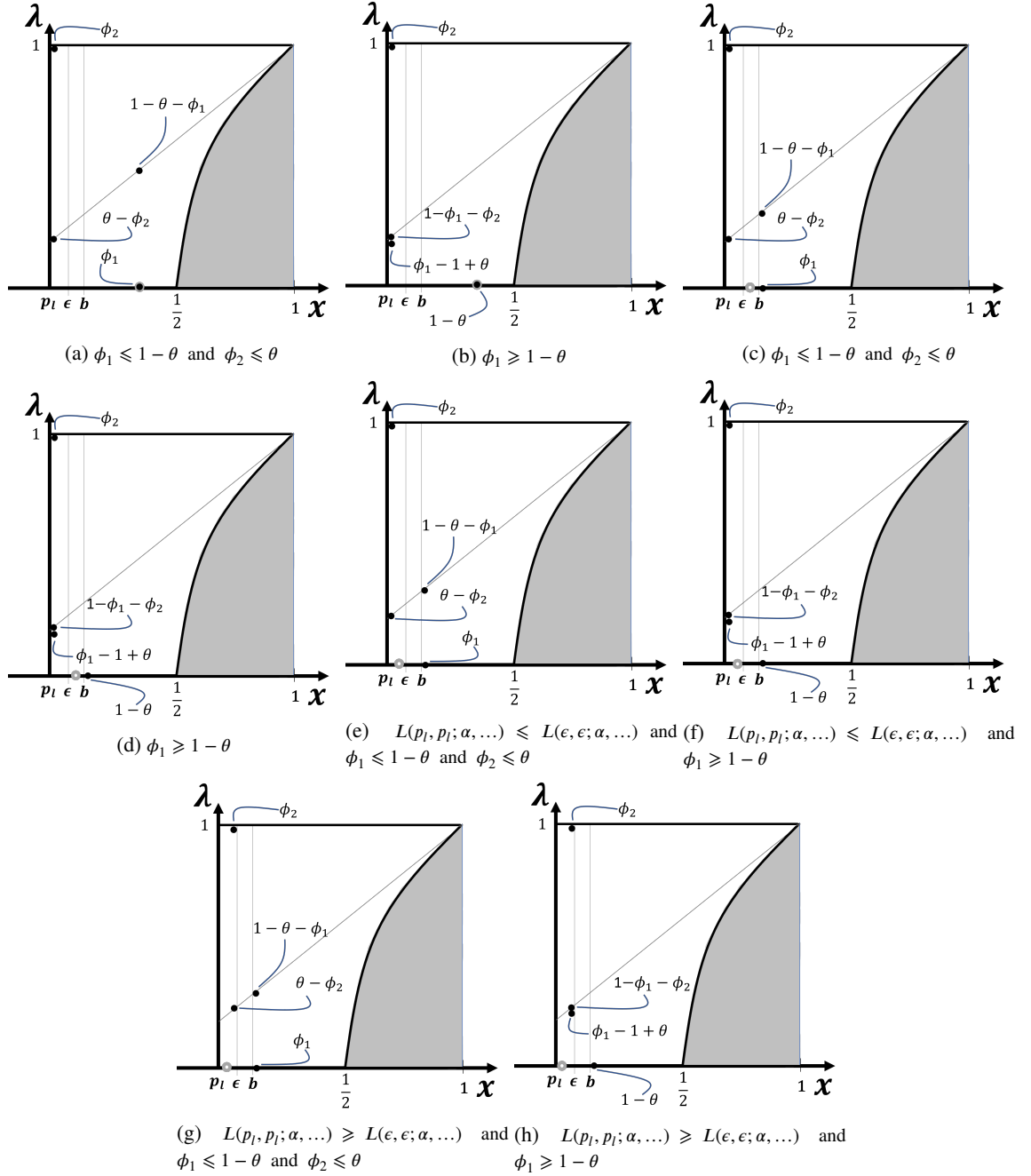
Figure 22: Worst case prior distributions that solve the optimisation problem in Theorem 3 when failures are observed, without any consecutive failures (i.e. $r = 0$). Each distribution's support is determined by $\alpha, \beta, \gamma, \delta$, and whether the first execution succeeds or fails. The location (x^*, λ^*) of the global maximum for the Klotz likelihood is indicated by the grey circle. The 4 priors, illustrated in subfigures 22a, 22c, 22e and 22g, are solutions when $\phi_1 \leqslant 1 - \theta$ and $\phi_1 \leqslant \theta$. While the priors in 22b, 22d, 22f and 22h solve the problem when $\phi_1 \geqslant 1 - \theta$. These solutions assume $\alpha, \beta, \gamma, \delta > 0$.

B. Independent Executions and Conservative Assessments

B.1. Which Independence Beliefs give Conservative Results?

Fig. 6 already shows that assessments based on independent executions *are* conservative initially, when the number of inputs during operational testing is relatively small. That is, the univariate CBI curve initially overlaps with the CBI
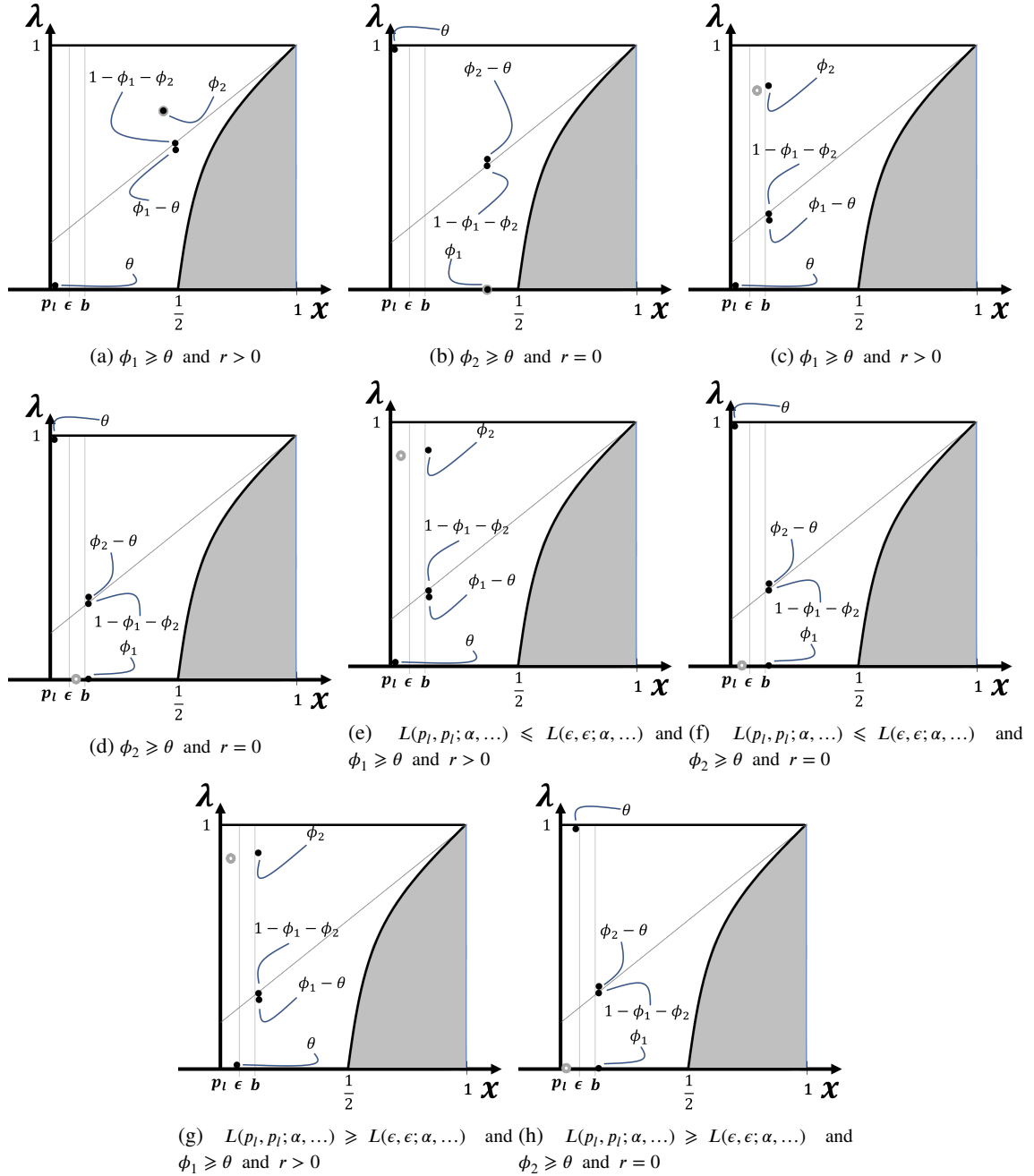
(a) $\phi_1 \geqslant \theta$ and $r > 0$

(b) $\phi_2 \geqslant \theta$ and $r = 0$

(c) $\phi_1 \geqslant \theta$ and $r > 0$

(d) $\phi_2 \geqslant \theta$ and $r = 0$

(e) $L(p_l, p_l; \alpha, \ldots) \leqslant L(\epsilon, \epsilon; \alpha, \ldots)$ and $\phi_1 \geqslant \theta$ and $r > 0$

(f) $L(p_l, p_l; \alpha, \ldots) \leqslant L(\epsilon, \epsilon; \alpha, \ldots)$ and $\phi_2 \geqslant \theta$ and $r = 0$

(g) $L(p_l, p_l; \alpha, \ldots) \geqslant L(\epsilon, \epsilon; \alpha, \ldots)$ and $\phi_1 \geqslant \theta$ and $r > 0$

(h) $L(p_l, p_l; \alpha, \ldots) \geqslant L(\epsilon, \epsilon; \alpha, \ldots)$ and $\phi_2 \geqslant \theta$ and $r = 0$

Figure 23: Worst case prior distributions that solve the optimisation problem in Theorem 3 when failures are observed, giving 0 posterior confidence. Each distribution's support is determined by $\alpha, \beta, \gamma, \delta$, and whether the first execution succeeds or fails. The location (x^*, λ^*) of the global maximum for the Klotz likelihood is indicated by the grey circle. The 4 priors, illustrated in subfigures 23a, 23c, 23e and 23g, are solutions when $\phi_1 \geqslant \theta$ and $r > 0$. While the priors in 23b, 23d, 23f and 23h solve the problem when $\phi_2 \geqslant \theta$ and $r > 0$. These solutions assume $\alpha, \beta, \gamma, \delta > 0$.

curve for dependent executions. But, posterior confidence based on independence becomes increasingly optimistic after a large number of tests. That is, the curves begin to deviate significantly as the number of tests increases. So, only assessments that allow for doubt in independence – i.e. nonzero ϕ_1 or ϕ_2 – can support (in the long run) more pessimistic claims about the bound b.

(a) $\phi_1 \geqslant \theta$

(b) $\phi_2 \geqslant 1 - \theta$

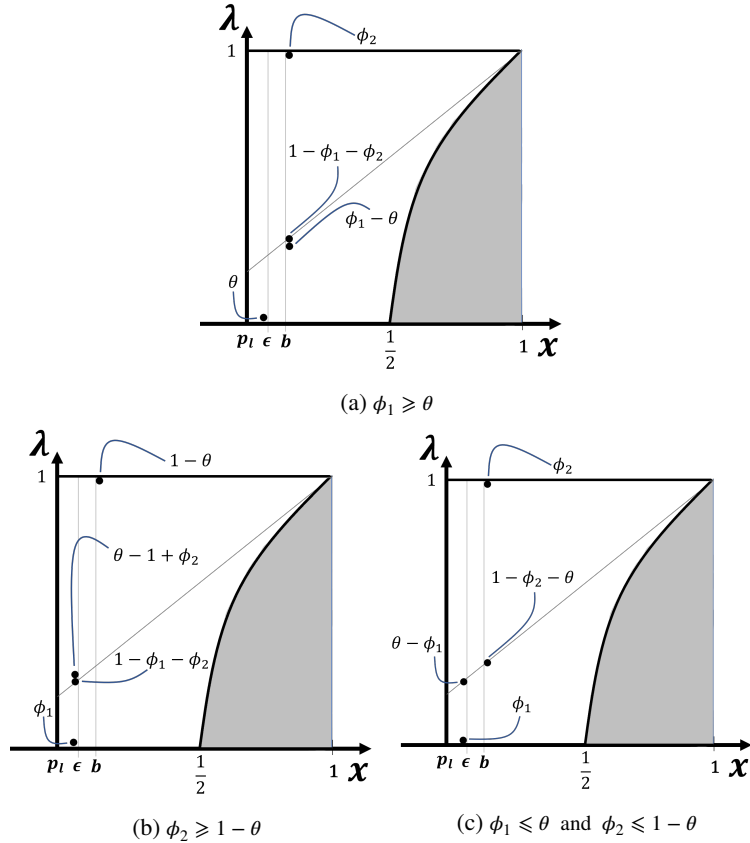(c) $\phi_1 \leqslant \theta$ and $\phi_2 \leqslant 1 - \theta$

Figure 24: For executions with **no failures**, these worst-case prior distributions solve the optimisation problem in Theorem 4. These priors are relevant for the ranges of parameter values indicated in each subfigure. The support of each distribution is determined by β.

Once some doubt in independence has been expressed, an assessor might want to allow for operational evidence to "slowly" allay such doubts. Or, instead, allow for the evidence to "quickly" convince them otherwise – that independence does *not* hold! PK5 represents an assessor who's initially very confident the system will exhibit independent, failure-free executions.

Prior Knowledge 5 (strong belief in independence). *The probability P(executions will be independent and failure-free), from the joint prior distribution of (X, Λ), has a value that is the solution to the optimisation problem:*

$$\sup_{D} P(\text{ executions will be independent and failure-free })$$
$$s.t. \quad PK1, \quad PK2, \quad PK3, \quad PK4$$

While PK6 is held by an assessor who's initially very doubtful the system will exhibit independent, failure-free executions.

Prior Knowledge 6 (weak belief in independence). *The probability P(executions will be independent and failure-free), from the joint prior distribution of (X, Λ), has a value that is the solution to the optimisation problem:*

$$\inf_{D} P(\text{ executions will be independent and failure-free })$$
$$s.t. \quad PK1, \quad PK2, \quad PK3, \quad PK4$$

Note the difference between the optimisation problems in these PKs, and those in CBI Theorems 1, 2 and 3. Here, the optimisations *constrain* the prior distribution in how it assigns probability mass; hence why these are PKs. While

the previous optimisations *are contrained by* the prior distributions – specifically, constrained by the PKs the priors must satisfy.
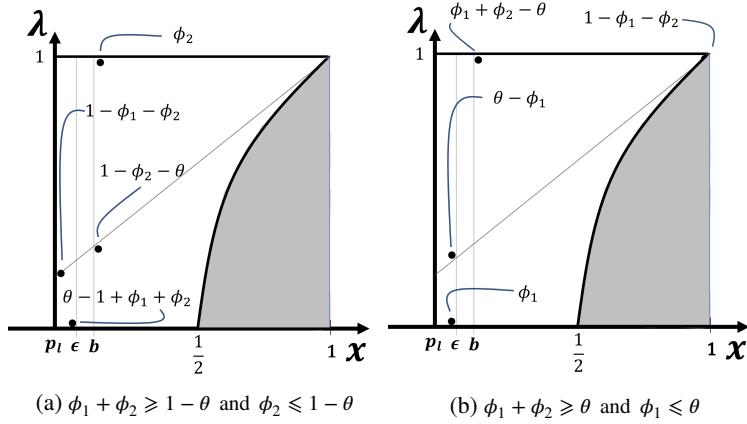

(a) $\phi_1 + \phi_2 \geqslant 1 - \theta$ and $\phi_2 \leqslant 1 - \theta$ (b) $\phi_1 + \phi_2 \geqslant \theta$ and $\phi_1 \leqslant \theta$

Figure 25: Prior distributions representing extreme beliefs about whether the executions will be independent (i.e. whether $[X = \Lambda]$), for the ϕ_1, ϕ_2, θ ranges indicated. Consider all prior distributions with the largest prior probability of observing n independent executions with no failures. The most pessimistic posterior confidence in b from these priors is given by the prior in Fig. 25a. Similarly, for all priors with the smallest prior probability of n independent failure-free executions, Fig. 25b gives the most pessimistic posterior confidence.

Theorems 5 and 6 below give the pessimistic posterior confidence in b, for assessors who hold PK5 or PK6 beliefs, respectively. Proved in B.2, some prior distributions that give the pessimistic posterior confidence in these theorems are shown in Fig. 25. And the confidence from these priors, as well as from priors in Theorems 1 and 2, are compared in Fig. 26.

Theorem 5. *Using* (3) *and* (4)*, the optimisation problem*

$$\inf_{D} P(X \leqslant b \mid n \text{ executions without failure})$$

$$s.t. \quad PK1, \quad PK2, \quad PK3, \quad PK4, \quad PK5$$

has the prior distribution in Fig. 25a as a solution, since $P(X < b \mid n \text{ executions without failure})$ *from this prior equals the infimum.*

Theorem 6. *Using* (3) *and* (4)*, the optimisation problem*

$$\inf_{D} P(X \leqslant b \mid n \text{ executions without failure})$$

$$s.t. \quad PK1, \quad PK2, \quad PK3, \quad PK4, \quad PK6$$

has the prior distribution in Fig. 25b as a solution, since $P(X < b \mid n \text{ executions without failure})$ *from this prior equals the infimum.*

Fig. 26 clearly shows that the confidence from the very skeptical "PK6"-believing assessor is initially the most optimistic (i.e. the widely-spaced dotted curve lies above all of the other curves). Noticeably more optimistic than even the confidence based on independent executions (i.e. the solid curve). This is in contrast to the assessor who holds the strong PK5 belief in independence. Such beliefs actually support conservative claims initially, even as claims based on independence start becoming optimistic – as suggested by the overlap of the dashed curve and the narrowly-spaced dotted curve, where the solid curve lies above both of them. Eventually, however, the roles are reversed as PK5 supported claims become optimistic, while PK6 supported claims become conservative and agree with the dashed curve. In this sense, "strongly believing" and "being skeptical of" the independence assumption are two halves of "conservatively doubting" the independence assumption. This is the behaviour for the range of PK parameter values in Fig. 26.

Why does PK5 initially support less optimistic claims than PK6, then less pessimistic claims as the number of successes n rises? It has to do with which prior beliefs about (X, Λ) – i.e. which locations in $\mathcal{R}$ – are crucial for
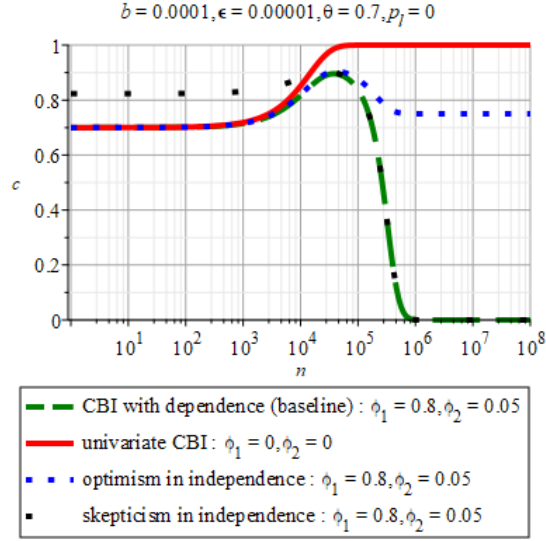
Figure 26: A comparison of posterior confidence in the bound b after operational testing, showing the impact of some confidence in the system's executions being independent. For the parameter values in this example, being very skeptical about independent executions (i.e. the prior in Fig. 25b) gives the most optimistic posterior confidence in the bound b shown here. While strong beliefs in independent executions (i.e. the prior in Fig. 25a) almost results in the most pessimistic confidence in b, at least for $n < 4 \times 10^4$ approximately.

conservative confidence in b. There are two principal beliefs a conservative assessor must hold: **i)** strong doubts of failure-free operation being evidence of a "sufficiently reliable" system – i.e., being evidence of a system with a *pfe* just smaller than b; **ii)** strong confidence in failure-free operation being evidence of an "almost sufficiently reliable" system – i.e., being evidence of a system with a *pfe* slightly worse than b, that exhibits perfectly positively correlated executions. With a PK5 belief, failure-free operation initially supports confidence in an "almost sufficiently reliable" system (hence, initially conservative confidence in b). But eventually, an increasing number of successes could also be due to the system being fault-free (because there is a nonzero probability at (p_l, p_l) in Fig. 25a and $p_l = 0$ in Fig. 26). So, the dotted curve reaches a horizontal asymptote. It's the reverse with a PK6 belief, where an "almost sufficiently reliable" system is initially very unlikely (hence initially optimistic confidence in b), but becomes arbitrarily more likely as n grows (hence conservative confidence in b).

When $\epsilon = 0$ in PK2, both PK5 and PK6 support conservative confidence in b as n grows large – i.e., Theorems 2, 5 (with PK5) and 6 (with PK6) agree eventually (see Fig. 27).

B.2. Proof of Theorem 5

We derive the prior distribution in Fig. 25a that solves the optimisation problem in Theorem 5. Analogous steps can be used to derive the prior in Fig. 25b which solves Theorem 6.

Theorem. *The optimisation problem*

$$\inf_{D} P(X \leqslant b \mid n \text{ executions without failure})$$

$$s.t. \quad PK1, \ PK2, \ PK3, \ PK4, \ PK5$$

has the prior distribution in Fig. 25a as one of its solutions, since $P(X < b \mid n \text{ executions without failure})$ from this prior equals the infimum.

Proof. For the prior distributions that solve PK5, the size of

$$P(n \text{ independent failure-free executions}) = \mathbb{E}[L(X, 1; n, 0, 0)]$$

is made as big as possible (for all $n \geqslant 0$) by assigning as much probability mass as possible to locations along the diagonal in $\mathcal{R}$ where the Klotz likelihood is largest. The likelihood is largest at (p_l, p_l) and monotonically decreases
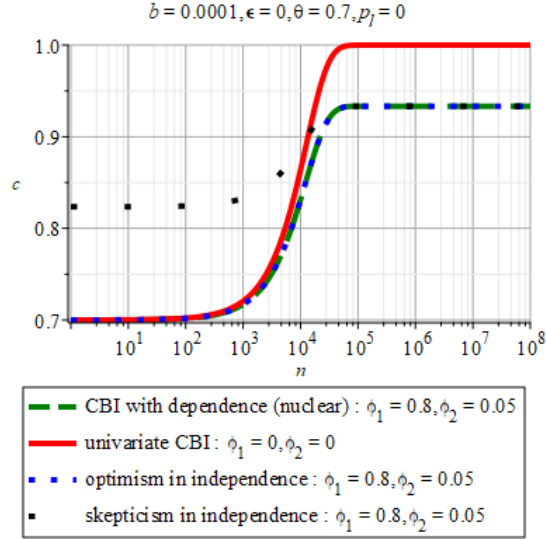
Figure 27: A similar comparison to that of Fig. 26, but with the added constraint that there is a probability $\theta = 0.7$ of the system containing no faults (i.e., $\epsilon = 0$). Now, a strong belief in independent executions gives the most pessimistic posterior confidence in b – i.e. the dashed "*optimism in independence*" and the dotted "*CBI with dependence*" curves are now indistinguishable.

to 0 at $(1, 1)$. Thus, **i)** if $\theta \geqslant 1 - \phi_1 - \phi_2$ then the prior assigns all of the mass along the diagonal (i.e. $1 - \phi_1 - \phi_2$) to the location (p_l, p_l). This gives the largest possible value for $\mathbb{E}[L(X, 1; n, 0, 0)]$ as $(1 - \phi_1 - \phi_2)(1 - p_l)^n$; **ii)** if instead $\theta \leqslant 1 - \phi_1 - \phi_2$, then the largest amount of probability mass that the prior can assign to (p_l, p_l) is θ, while the remaining $1 - \phi_1 - \phi_2 - \theta$ mass must be assigned to the limit point of the range $x > \epsilon$ (along the diagonal) where the likelihood is largest – the limit point (ϵ, ϵ). In this case, the prior must be the limit of a sequence of priors that re-assign the remaining mass via a sequence of points that converge to (ϵ, ϵ) from the right. The largest value of $\mathbb{E}[L(X, 1; n, 0, 0)]$ in this case is thus $\theta(1 - p_l)^n + (1 - \phi_1 - \phi_2)(1 - \epsilon)^n$.

Consequently, the priors that solve PK5 – i.e., the feasible priors in theorem 5 – must allocate probability along the diagonal in one of the two ways just outlined. In particular, from among those priors that allocate all of the $1 - \phi_1 - \phi_2$ mass to the point (p_l, p_l), the methods of A justify the prior in Fig. 25a as a solution of theorem 5 . ∎