Measuring Improved Reliability in Stochastic Systems

Jeremy Bradley* Neil Davies†

30 June 1999

Abstract

In this paper, we examine reliability of stochastic systems with specific application to Stochastic Process Algebras. We look at variance as a possible reliability metric and show how it can be reduced across different Stochastic Process Algebra combinators, thus giving more deterministic and reliable systems. We apply these techniques to analyse and improve a searching algorithm.

1 Introduction

Reliability is often seen as a cut-and-dried property. A computer model is proposed, a reliability criterion is defined and the model is analysed against that criterion to see whether it is or is not reliable.

In TIS [6] (a real-time modelling language), for example, specific amounts of time are allocated to actions of a model. Verification takes the form of asking whether intervals of operation take less than or greater than a hard bound. This kind of analysis is very useful if a system can be defined with such precision.

Stochastic Process Algebras (SPAs) allow us to represent temporal uncertainty; that is, the fact that we cannot tell how long a particular action may take. This is a powerful modelling ability—it is not completely arbitrary, not completely non-deterministic, but instead represents a weighted non-determinism.

As such SPAs are a half-way house between completely non-deterministic functional process algebras (such as CSP and CCS) and fully deterministic timed process algebras.

2 Defining Reliability

From an SPA point-of-view reliability is a slightly alien concept. It is generally considered a hard enough problem to obtain average behaviour for a stochastically defined system without having to worry about obtaining other measures. As a result, it is difficult to see how reliability could be extracted from current solution techniques which are centred around finding steady-state probabilities.

*Email: `bradley@cs.bris.ac.uk`

†Email: `neil@cs.bris.ac.uk` — Department of Computer Science, The University of Bristol, Merchant Venturers Building, Woodland Road, Bristol, BS8 1UB.

In a performance context, knowing how much a system can vary from its mean performance is an essential practical consideration. Or indeed, to put it the other way round, if the variation of behaviour is very large then having only the mean figure for a sojourn is probably not much use for modelling.

So, how to define reliability in the context of a stochastically defined system? If we consider a path in a stochastic transition system, we can aggregate the random variables that make up the path and obtain a distribution for the entire path [1]. This distribution will have a mean and a variance; as a starting point the variance will be a useful reliability statistic to gather. A path with large variance will slow down a system's convergence to stationary behaviour, whereas a path with small variance will make the system more predictable (or deterministic).

However, if we are able to derive the distribution completely for such a path, this represents a better measure of the path's reliability. If the distribution were asymmetric, for instance, this would be evident from the distribution but not reflected in the variance.

In the course of this paper, we will look at both aggregate distributions and variances. We will discover that general statements about reliability can be made according to how systems are constructed. In particular, the type of operator and modelling strategy used can dramatically affect the predicted reliability of a system.

3 Real-Time Systems Modelling

For systems for which reliability is important, average long-term behaviour is not the primary issue: hazards may occur or functionality may be at risk if individual instances of operation fall outside certain bounds. We want to be able to reason about local or transient behaviour as well as operation in the long term.

For instance, if we have a timeout race between an exponentially distributed action, X, and a deterministic cut-off, Y, if the timeout occurs first the system might well go into some kind of failure mode or error handler or may just deadlock.

$$P(X > Y) = P(X > 10) = \overline{F}_X(10) = e^{-100} \tag{1}$$

Such vanishingly small probabilities might form the basis of a safety case or an industrial standard for a component or hazard situation.

In MTL [4], typically, such local behaviour is looked at by examining the interval between two events where each event in the system is given a fixed duration. In a stochastic system we can use the notion of stochastic aggregation [1] to produce a similar, but stochastically defined, result.

So, given a distribution for a particular interval, how might reliability be defined on it? Various TIS papers [6, 5, 9] have suggestions on the subject, which can basically be summarized into two categories:

1. **One-tailed test:** $P(X < a) = p$ or $P(X > b) = q$
2. **Two-tailed test:** $P(a < X < b) = p$

where a, b, p and q would be specified for necessary reliable behaviour. These are easily attainable from the cumulative distribution function of an aggregate distribution.

4 A Simple Stochastic Transition System

A second idea is to use the aggregation method again, but this time calculate the variance (and then, easily, the coefficient of variance). This is particularly attractive since it not only gives an idea of how "concentrated" or "diffuse" a distribution might be (which is a measure of determinism) but also allows us to aggregate in Laplace-space and generate the moments from the Laplace transform of the aggregate distribution [7]. Thus an inversion of a Laplace transform and an integration to the cumulative distribution function are avoided.

Let us consider the following state transition system, where each transition has a random variable, X, attributed to its execution:

$$P ::= X.P \;\mid\; [m]P + [1-m]P \;\mid\; \mathbf{0} \tag{2}$$

This represents a stochastic component where either transitions occur, taking a random amount of time, or a probabilistic branch occurs with probability m for one path and $(1-m)$ for the other.

Just by considering the variance of certain stochastic paths, we can gain some interesting insight. Define the following reliability measure, $\mathbf{R} :: P \longrightarrow \mathsf{R}$:

1. $\mathbf{R}\{\mathbf{0}\} = 0$
2. $\mathbf{R}\{X.P\} = \mathbf{Var}\{X\} + \mathbf{R}\{P\}$
3. $\mathbf{R}\{[m]P+[1-m]Q\} = m\mathbf{R}\{P\}+(1-m)\mathbf{R}\{Q\}+m(1-m)(\mathbf{E}\{P\}-\mathbf{E}\{Q\})^2$

where $\mathbf{E}\{P\}$ represents the average execution time of component P and $\mathbf{Var}\{X\}$ is the variance of random variable X. Property 2 is essentially a result of the convolution formula and shows that a finite system which is described by nothing but a sequence of events will have a monotonically increasing variance. Property 3 represents the formula for a probabilistic branch [7] and shows that this is where variance reduction might be achieved within a stochastic system.

Since many systems are largely sequential, this leads to a variance explosion problem, when modelling or simulating systems. The fact that probabilistic branching (and thus other constructs like competitive choice and interleaved synchronisations [2]) can reduce variance, means that there is a possibility of reducing this explosion, as we will demonstrate in the next section.

5 Improving Reliability of a Search Algorithm

As an example of how to improve the reliability of a system, we consider a search algorithm. The goal is to design a second algorithm which, when combined in a First-to-Finish strategy [2], would improve the reliability of the overall search. In this case, we consider 'improved' reliability to mean 'decreased variance'.

As an initial gambit we devise a naïve algorithm for searching an array of size N. We make the following assumptions about the search:

1. the data in the array is unordered
2. fetching any element from the array can be achieved in constant time, μ
3. there is only one element to find in the array and once it is found, the search can be terminated.

The naïve algorithm simply randomly selects an element from the array and then terminates if it is the correct element. No attempt is made to mark the element as searched or remove the element from the array if it is not the correct element; another random selection is made and the process continues.

The second algorithm is simply a sequential search through the array; on finding the correct element, it too terminates immediately. This has the obvious benefit of terminating successfully after at most time $N\mu$.

5.1 Analysis of Random Search Algorithm

Let T represent the amount of time spent searching for the element, as shown in figure 1.

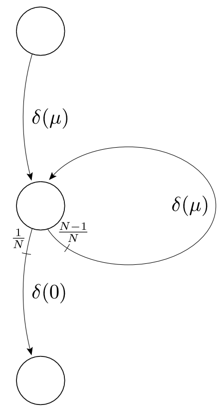

Figure 1: *Stochastic transition system for random search algorithm*

$$L_T(\omega) = \frac{e^{-\mu\omega}}{N-(N-1)e^{-\mu\omega}} \tag{3}$$

$$L'_T(\omega) = -\frac{\mu(N-1)e^{-2\mu\omega}}{(N-(N-1)e^{-\mu\omega})^2} - \frac{\mu e^{-\mu\omega}}{N-(N-1)e^{-\mu\omega}} \tag{4}$$

$$\begin{aligned} L''_T(\omega) &= \frac{2\mu^2(N-1)e^{-2\mu\omega} + \mu^2(N-1)e^{-2\mu\omega}}{(N-(N-1)e^{-\mu\omega})^2} \\ &\quad + \frac{2\mu^2(N-1)^2e^{-3\mu\omega}}{(N-(N-1)e^{-\mu\omega})^3} + \frac{\mu^2 e^{-\mu\omega}}{N-(N-1)e^{-\mu\omega}} \end{aligned} \tag{5}$$

$$\mathbf{E}\{T\} = \mu N \tag{6}$$

$$\mathbf{E}\{T^2\} = (2N-1)N\mu^2 \tag{7}$$

$$\mathbf{Var}\{T\} = N(N-1)\mu^2 \tag{8}$$

Now looking at the Coefficient of Variation:

$$C_T = \frac{\sqrt{N(N-1)}\mu}{N\mu} = \sqrt{\frac{N-1}{N}} \tag{9}$$

Thus $C_T \longrightarrow 1$ as $N \longrightarrow \infty$. Given that the exponential distribution is the only distribution which has a Coefficient of Variation of 1, we approximate T by an exponential distribution with parameter λ:

$$\lambda = \frac{1}{N\mu} \tag{10}$$

5.2 Analysis of Sequential Search Algorithm

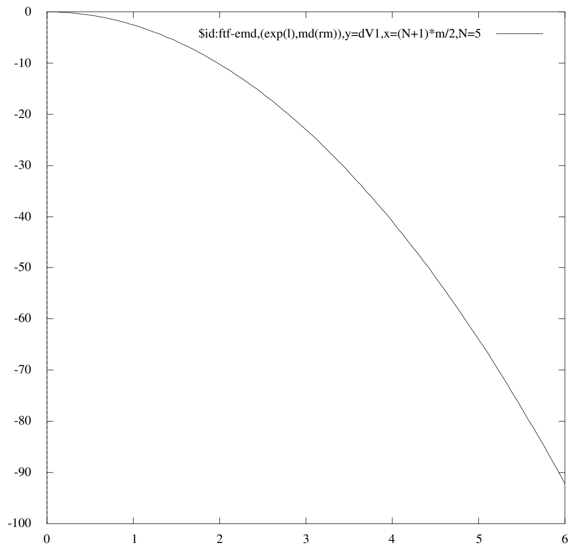

Figure 2: ΔV_1 *against* $\mathbf{E}\{T'\}$ *for fixed size array: the introduction of the linear search produces a large variance reduction*

We can immediately write down an expression for the overall evolution of the sequential search, given that after r stages:

1. The next element is fetched from the array, incurring a wait of length μ.
2. The search then terminates with probability $\frac{1}{N-r}$ if the element is the target, or continues with probability $\frac{N-r-1}{N-r}$.
3. The search proceeds to the $(r+1)$th stage. In the event that the search reaches the Nth stage, then the Nth element is going to be the target with probability 1, so the search terminates before the array ends.

Let T' represent the overall evolution of the sequential search:

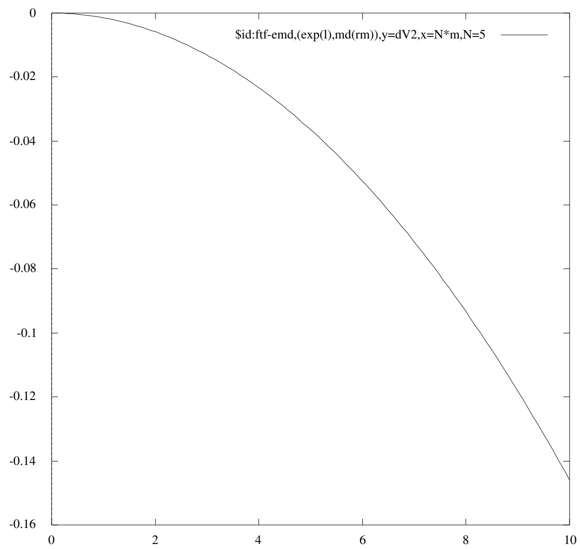

Figure 3: ΔV_2 *against* $\mathbf{E}\{T\}$ *for fixed size array: the introduction of the random search produces a small variance reduction*

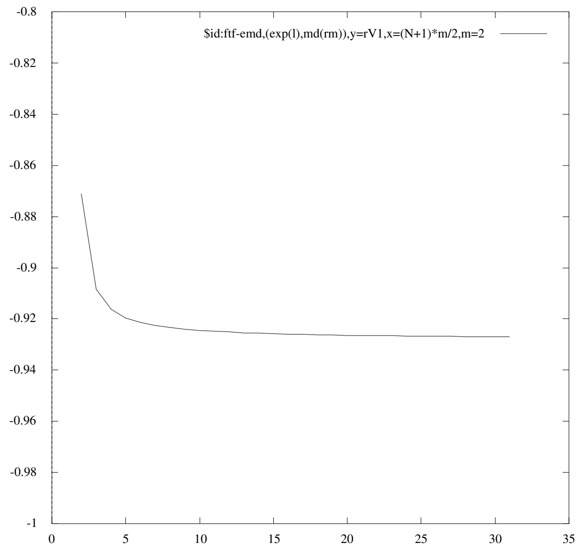

Figure 4: rV_1 *against* $\mathbf{E}\{T'\}$ *for variable size array: relative to the original random search, the linear search reduces the variance by up to 93% for large N*

$$\begin{aligned} L_{T'} &= e^{-\mu\omega}\left(\frac{1}{N} + \frac{N-1}{N}e^{-\mu\omega}\left(\frac{1}{N-1} + \frac{N-2}{N-1}e^{-\mu\omega}\left(\frac{1}{N-2} + \cdots \right.\right.\right. \\ &\quad \left.\left.\left. \cdots\left(\frac{1}{2} + \frac{1}{2}e^{-\mu\omega}\right)\cdots\right)\right)\right) \end{aligned} \tag{11}$$

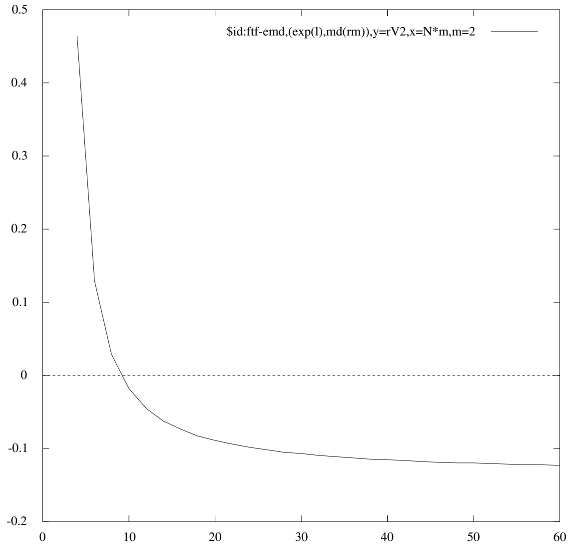

Figure 5: rV_2 *against* $\mathbf{E}\{T\}$ *for variable size array: relative to the original linear search, the random search still reduces the variance by up to 12% for large* N

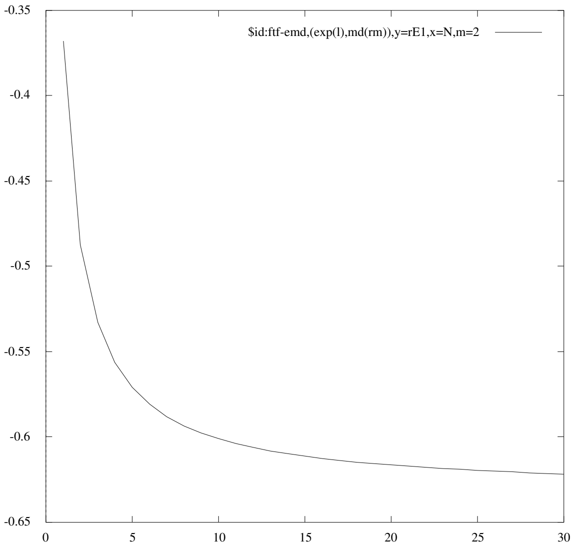

Figure 6: rE_1 *against* N*: the mean reduces by up to 62% when the linear search is combined with the random search, for large* N

which simplifies to:

$$L_{T'} = \frac{1}{N} \sum_{r=1}^{N} e^{-r\mu\omega} \tag{12}$$

$$\mathbf{E}\{T'\} = \frac{\mu}{N} \sum_{r=1}^{N} r = \frac{N+1}{2}\mu \tag{13}$$

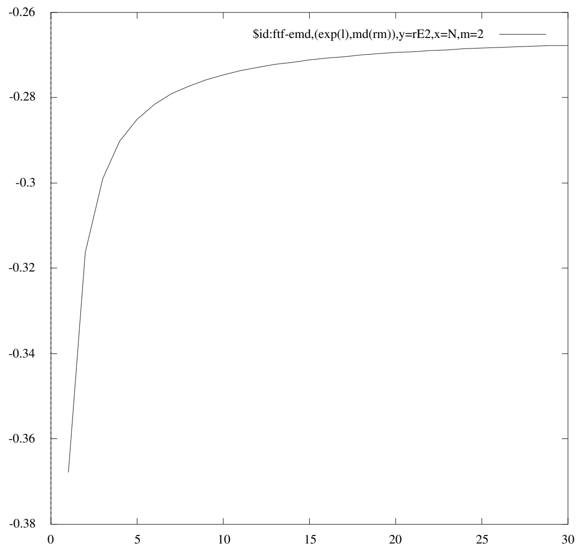

Figure 7: rE_2 *against* N*: the mean reduces by 27% when the random search is combined with the linear search, for large* N

$$\mathbf{E}\{T'^2\} = \frac{\mu^2}{N}\sum_{r=1}^{N} r^2 = \frac{1}{6}(2N+1)(N+1)\mu^2 \tag{14}$$

$$\mathbf{Var}\{T'\} = \frac{1}{12}(N+1)(N-1)\mu^2 \tag{15}$$

5.3 Analysis of Concurrent Search Process

We now require to combine the two strategies under a First-to-Finish synchronisation. The improvement in variance can then be calculated.

$$\overline{F}_{T \overset{\|}{\text{FTF}} T'}(t) = \frac{1}{N} e^{-\lambda t} \sum_{r=1}^{N} H(r\mu - t) \tag{16}$$

$$f_{T \overset{\|}{\text{FTF}} T'}(t) = \frac{1}{N} \sum_{r=1}^{N} (\lambda e^{-\lambda t} H(r\mu - t) + e^{-\lambda t}\delta(t - r\mu)) \tag{17}$$

This leads to the following results for the overall FTF synchronisation process:

$$\mathbf{E}\{T \overset{\|}{\text{FTF}} T'\} = \frac{1}{N\lambda} \sum_{r=1}^{N} (1 - e^{-\lambda r\mu}) \tag{18}$$

$$\mathbf{Var}\{T \overset{\|}{\text{FTF}} T'\} = \frac{2}{N\lambda^2} \sum_{r=1}^{N} (1 - (1 + \lambda r\mu) e^{-\lambda r\mu}) \tag{19}$$

We now calculate:

$$\Delta V_1 = \mathbf{Var}\{T \overset{\|}{\text{FTF}} T'\} - \mathbf{Var}\{T\} \text{ and } \Delta V_2 = \mathbf{Var}\{T \overset{\|}{\text{FTF}} T'\} - \mathbf{Var}\{T'\} \tag{20}$$

These are plotted (figures 2, 3) against the means of T' and T respectively to show how much delay needs to be introduced into the synchronising process to produce a given change in variance. These clearly show that the synchronisation is successful in reducing the variance, although it can be seen that the random algorithm has far less effect on the variance of the linear algorithm (figure 3).

In order to establish relative improvement in variance, we need to see how ΔV_1 and ΔV_2 change relative to the original variances, $\mathbf{Var}\{T\}$ and $\mathbf{Var}\{T'\}$. To this end, figures 4, 5 show:

$$rV_1 = \frac{\Delta V_1}{\mathbf{Var}\{T\}} \text{ and } rV_2 = \frac{\Delta V_2}{\mathbf{Var}\{T'\}} \tag{21}$$

Given that $\mathbf{Var}\{\cdot\} \geq 0$ so it implies that $rV_N \geq -1$ with $rV_N = -1$ indicating a deterministic evolution and a 100% reduction in variance.

Figure 4 represents the reduction of variance relative to the random search, given the introduction of the linear search; it can be seen that for increasing size of array (which linearly increases the size of $\mathbf{E}\{T'\}$), the variance is reduced by approaching 93%. Figure 5 shows the reduction of variance relative to the linear search, given the introduction of the random search as a synchronising process; even here, despite the fact that the random algorithm has a much larger variance, its introduction will still reduce the overall variance by approaching 12% for large N. This demonstrates our assertion that, in this case anyway, a concurrent algorithm can be employed to improve the reliability of the overall system.

Producing similar relative mean plots, figures 6 and 7[1], shows that the means are reduced as well through such an interaction, however an instance could be envisaged when a modeller might be prepared to put up with larger average operation in return for a more reliable system.

6 Conclusion

In this paper we have discussed the difference between average behaviour or performance and variant behaviour or reliability. Performance in stochastic systems is usually obtained from stationary distributions and represents long-run state probabilities. As such, it automatically discounts variation or short-term extreme behaviour.

What has been attempted here, therefore, is to show that by identifying a key transition or set of transitions and by employing techniques to reduce the variance of such a transition, the reliability (i.e. determinacy) of the whole system can be improved.

We demonstrated this variance reduction technique on a simple searching example which might have been a crucial element of a wider communicating system. In doing so, metrics for reliability measurement were implemented and used to demonstrate improvement.

As further work, we need to develop some more of these ideas across process algebra models and try and find stochastic structures which are particularly amenable to variance reduction. We have so far only demonstrated variance

[1] $rE_1 = (\mathbf{E}\{T \underset{\mathrm{FTF}}{\|} T'\} - \mathbf{E}\{T\})/\mathbf{E}\{T\}$, $rE_2 = (\mathbf{E}\{T \underset{\mathrm{FTF}}{\|} T'\} - \mathbf{E}\{T'\})/\mathbf{E}\{T'\}$

properties for convolution, probabilistic branching and First-to-Finish synchronisation; we have further results on competitive choice, and Last-to-Finish synchronisation which will be presented soon.

References

[1] Bradley, J. T., and Davies, N. J. An aggregation technique for analysing some generally distributed stochastic processes. Tech. Rep. CSTR–99–003, Department of Computer Science, University of Bristol, March 1999.

[2] Bradley, J. T., and Davies, N. J. Reliable performance modelling with approximate synchronisations. Tech. Rep. CSTR–99–002, Department of Computer Science, University of Bristol, February 1999.

[3] Feller, W. *An Introduction to Probability Theory and its Applications*, third ed., vol. 1. John Wiley, 1968.

[4] Koymans, R. Specifying real-time properties with metric temporal logic. *Real Time Systems 4*, 2 (1990), 255–290.

[5] Schieferdecker, I. Verifying performance oriented properties of TIS specifications. *Journal of Mathematical Modelling and Simulation in Systems Analysis* (October 1993).

[6] Schieferdecker, I., and Wolisz, A. Verifying deterministic performance bounds of communication protocols. In *Third International Workshop on Responsive Computer Systems* (Lincoln, New Hampshire, October 1993).

[7] Trivedi, K. S. *Probability and Statistics with Reliability, Queuing and Computer Science Applications.* Prentice-Hall, 1982.

[8] Watson, E. J. *Laplace Transforms and Applications.* Van Nostrand Reinhold, 1981.

[9] Wolisz, A., Schieferdecker, I., and Walch, M. An integrated approach to the design of communication protocols. In *Fourth Workshop on Future Trends of Distributed Computer Systems* (Lisbon 1993).