Applying Random Arrival Models to Fixed Priority Analysis

I. Broster*, A. Burns
University of York, UK
{ianb, burns}@cs.york.ac.uk

Abstract

This paper addresses the use of a non-bounded interference function caused by random arrivals in worst case response time analysis. An outline of a probabilistic analysis is presented which is based on a simple random arrival model. The analysis produces a probability distribution of response times. The analysis is derived from an analysis of random faults on CAN and inherits some limitations from this. We report this analysis as a stand-alone result and observe that for some systems this analysis can be directly applied. However, work is in progress to remove these limitations.

1. Introduction

Worst case response time (WCRT) analysis can be used to guarantee deadlines in a fixed priority scheduler. By construction, it considers the the single worst case scenario (or a scenario which is equivalently as bad as the worst case). Thus, in WCRT analysis, it is necessary to place a bound on all the components of the analysis (execution times, arrivals, faults, jitter, blocking, interference *etc.*).

One of the fundamental assumptions that WCRT analysis makes is that the overhead of all process *arrivals* is bounded. Typically, it is assumed that arrivals are periodic or sporadic (i.e. with a minimum inter-arrival time, which is equivalent to the periodic model in the worst case). However for event-triggered systems which interact with the real-world, it is not necessarily the case that a useful bound can be placed on the arrival rate. Interrupt arrivals from external sensors or network interfaces, for example, may have no useful upper bound. The presence of transient faults is another example; modelling the effect of transient network faults is the original context of this analysis.

Assuming worst case bounds on all components of analysis is not only difficult (or impossible in some cases), but in practical situations the analysis may simply return “no guarantees are possible”. However, running such a system might show that deadlines are never (or rarely ever) missed. Thus, while the analysis is correct: there are no guarantees (there exists a scenario in which a deadline may be missed), the analysis fails to describe the system adequately.

In previous publications [3, 1, 2], a probabilistic response time analysis was developed for modelling the impact of transient faults caused by EMI on a CAN bus. It is not reasonable to assume that faults have a bounded impact: the presence of a fault at one instant does not guarantee that there will be no faults in the next instant. The probabilistic approach works very well in that domain. In this paper, we apply the same basic approach to a general fixed priority scheduling environment.

The advantages of this probabilistic analysis are that it is accurate, fast and simple. The disadvantage is that, in this form, it can only consider one random arrival stream. Section 2 explains how the original CAN analysis is applied to fixed priority scheduling. Section 3 introduces a way to apply this for multiple streams.

2. Probabilistic Worst Case Response Time

This section describes the analysis approach. We begin by a brief discussion of the CAN-based analysis, then describe it in the context of fixed-priority scheduling.

2.1. Modelling Faults in CAN

Broster *et al.*[3, 2] demonstrated that a probabilistic analysis approach can accurately model the impact on response times of a random fault arrival model. A key characteristic of a probabilistic analysis is that it is able to model the tail of a distribution, with probabilities that are difficult to consider using, for example, simulation or measurement techniques alone. Experiments [1] with CAN show that the results of the analysis are accurate and do not exhibit significant pessimism.

The fault model considered for the CAN analysis is that faults arrive randomly with a Poisson distribution (this standard distribution models random arrivals in many domains and is frequently applied to fault occurrences). Faults in CAN are effectively like high-priority interference, much like interrupts in CPU scheduling, although since a fault in CAN also results in a frame retransmission the overhead of a single fault has a larger impact than a typical small interrupt.

The analysis technique (specifically, the version given in [3]) is simple and has a low computational overhead to calculate. As a number of other probabilistic approaches emerge [5, 6], it is clear that the computational overhead of probabilistic analysis can be significant. The approach for CAN does not suffer this problem.

The overall result of the CAN analysis is a probability distribution of response times. This can be represented as a

* This work is supported by the IST Programme of the European Commission under project IST-2001-34820 (FIRST).

cumulative graph to provide a useful guide to the probability of successful delivery of any frame.

2.2. Framework for Fixed Priority Scheduling

We introduce the analysis in the context of a general fixed priority environment using the framework suggested by Burns *et al.*[4]. Thus we begin with the familiar WCRT equations. Following the general framework approach, we may break the interference into parts. $\mathsf{hpp}(i)$ is the set of processes with priority greater than i which have a bounded arrival model, typically periodic (or periodic in the worst case). Likewise $\mathsf{hpn}(i)$ is the set of processes with priority greater than i which do not have a simple bounded interference function. The worst case response time for a process i is given by:

$$R_i = C_i + B_i + \sum_{j \in \mathsf{hpp}(i)} \left\lceil \frac{R_i}{T_j} \right\rceil C_j + \sum_{k \in \mathsf{hpn}(i)} A_k(R_i) C_k \qquad (1)$$

where B_i is the worst case blocking that process i can experience, T_i is the period of process i, C_i is the worst case execution time of process i. Slightly different to original framework[4], however, we use $A_k(t)$ to be a random variable with the meaning "the number of random arrivals of event k in an interval of time t".

2.3. Arrival Model

The original analysis for CAN deals with a single stream of random fault arrivals. In CAN, the worst case overhead of one fault is equivalent to any other (faults are neither specific to, nor related to, the frames they affect). Thus the inherent limitation of the CAN analysis, when applied to this general framework, is that it deals with only one random arrival stream. Multiple streams are considered in Section 3. The rest of this section assumes one arrival stream.

We consider a random arrival model with a random distribution. A Poisson distribution is used, $A_k \sim \mathscr{P}o(\lambda)$. The Poisson distribution has the property that the distribution is memoryless, which is required for the correctness of the analysis. Therefore, we define the probability of exactly m arrivals occurring in any time interval t as:

$$\mathrm{P}(m,t) = \mathrm{P}_t(A_k = m) = \frac{e^{-\lambda t}(\lambda t)^m}{m!} \qquad (2)$$

The worst case overhead of each arrival of process k is C_k. Therefore the function describing the non-bounded arrivals is a random distribution:

$$N_k(t) = \begin{cases} 0 & \text{with probability } \mathrm{P}(0,t) \\ C_k & \text{with probability } \mathrm{P}(1,t) \\ 2C_k & \text{with probability } \mathrm{P}(2,t) \\ \cdots & \cdots \end{cases} \qquad (3)$$

or more generally:

$$N_k(t) = A_k(t) C_k \qquad (4)$$

2.4. Pre-computing Response Times

As the example later in this paper will illustrate, the shape of the probability distribution output is 'stepped'. The cause is the simple nature of the overhead function (3). It is noted, therefore, that there are only a relatively small number of possible worst case response times that this analysis will generate. A large number of different scenarios contribute to the probability of each response time value; the probability of each response time is the sum of the probabilities of these scenarios.

Therefore, it is possible to pre-compute the set of possible response times up to some point (such as the period, which is the limit of the analysis) and then calculate the possible scenarios which contribute to each response time.

Note that we use the notation $R_{i|m}$ to mean the worst case response time given that m events arrive before the process completes an invocation. Pre-computing the response times is done in the expected manner, by forming a recurrence relation from equation (5) for all m up to the deadline, D_i, or period, T_i, such that $R_{i|m} < D_i \leq T_i$.

$$R_{i|m} = C_i + B_i + \sum_{j \in \mathsf{hpp}(i)} \left\lceil \frac{R_{i|m}}{T_j} \right\rceil C_j + mC_k \qquad (5)$$

2.5. Scenarios

After pre-computing the possible worst case response times, it is necessary to consider the scenarios that contribute to each possible value. The following discussion of these scenarios is useful to aid understanding of the scheme.

Equation (5) generates a set of non-overlapping intervals over m, as shown in Figure 1. The notation $e(n)$ is used to denote the number of arrivals that occur in time interval $(R_{i|n-1}, R_{i|n}]$ (or $(0, R_{i|n}]$ where $n = 0$).

0	$R_{i\|0}$	$R_{i\|1}$	$R_{i\|2}$	$R_{i\|n-1}$	$R_{i\|n}$
$e(0)$	$e(1)$	$\cdots$	$e(n-1)$		

Figure 1. Possible Worst Case Response Times for a Given Number of Faults.

Using the shorthand, $\langle 210 \rangle$ to mean the scenario e(0)=2, e(1)=1, e(2)=0, Table 1 shows the scenarios which contribute to a given response time[1]. Note that (for example) the sequence $\langle 1020 \rangle$ cannot contribute to $R_{i|3}$, since the sequence begins $\langle 10 \rangle$ which contributes only to $R_{i|1}$ because at time $R_{i|1}$, there has been only one arrival therefore the iteration of the WCRT equation terminates. It would be pessimistic to attach the scenario $\langle 1020 \rangle$ to the probability of the response time for 3 arrivals, $R_{i|3}$.

1 The sequence of the number of scenarios which constitute each response time grows rapidly. It begins 1, 1, 2, 5, 14, 42, 132, 429, 1430, 4862, 16796, 58786, 208012, 742900, and is known as the Catalan Se-

Response Time	Possible Scenarios (Shorthand)	Number of Scenarios
$R_{i\|0}$	$\langle 0 \rangle$	1
$R_{i\|1}$	$\langle 10 \rangle$	1
$R_{i\|2}$	$\langle 200 \rangle$, $\langle 110 \rangle$	2
$R_{i\|3}$	$\langle 3000 \rangle$, $\langle 2100 \rangle$, $\langle 2010 \rangle$, $\langle 1200 \rangle$, $\langle 1110 \rangle$	5
$R_{i\|4}$	$\langle 40000 \rangle$, $\langle 31000 \rangle$, $\langle 30100 \rangle$, $\langle 30010 \rangle$, $\langle 21100 \rangle$, $\langle 21010 \rangle$, ...	14

Table 1. Enumeration of Scenarios.

2.6. Efficient Probabilistic Analysis

From the precomputed response times, an efficient probabilistic analysis can be used to find the probabilities of the response times without enumerating all the scenarios. The technique is presented in this section.

2.6.1. Calculating $\mathrm{P}(R_{i|0})$ Considering the worst case response time with no arrivals, $R_{i|0}$: $\mathrm{P}(R_{i|0})$ is the upper bound on the probability of arrivals not causing a process i to exceed this time. It is simply the probability that there are no arrivals of the random stream in the interval $(0, R_{i|0}]$. As Table 1 showed, this is the only possible scenario that can produce a response time of $R_{i|0}$.

$$\mathrm{P}(R_{i|0}) = \mathrm{P}(0, R_{i|0})$$

2.6.2. Calculating $\mathrm{P}(R_{i|1})$ For the response time $R_{i|1}$, *i.e.* 1 arrival, there is only one scenario which can cause this. Table 1 shows this to be $\langle 10 \rangle$, there must be exactly one arrival in $(0, R_{i|0}]$ and no arrivals in $(R_{i|0}, R_{i|1}]$. The previous section suggested that the probability of $\langle 10 \rangle$ may be calculated by summing the probabilities of the scenarios (just one in this case).

However, an alternative approach is to begin with the probability of having exactly one arrival in the interval $(0, R_{i|1}]$, which is $\mathrm{P}(1, R_{i|1})$. This can occur in only two ways: $\langle 01 \rangle$ or $\langle 10 \rangle$, of which only $\langle 10 \rangle$ is of interest. The probability of scenario $\langle 01 \rangle$ is already partially calculated because this is $\mathrm{P}(R_{i|0})$ multiplied by the probability of 1 arrival in $(R_{i|0}, R_{i|1}]$.

$$\begin{aligned} \mathrm{P}(1, R_{i|1}) = &\mathrm{P}(R_{i|1}) && \langle 10 \rangle \\ &+ \mathrm{P}(R_{i|0})\,\mathrm{P}(1, R_{i|1} - R_{i|0}) && \langle 01 \rangle \end{aligned}$$

Hence:

$$\mathrm{P}(R_{i|1}) = \mathrm{P}(1, R_{i|1}) - \mathrm{P}(R_{i|0})\,\mathrm{P}(1, R_{i|1} - R_{i|0})$$

2.6.3. Calculating $\mathrm{P}(R_{i|m})$ Likewise, to calculate $\mathrm{P}(R_{i|2})$, is it possible to begin with the probability that there must be exactly two arrivals in $(2, R_{i|2}]$ and then exclude the scenarios where there were exactly 0 arrivals in $(0, R_{i|0}]$, or exactly 1 arrival in $(0, R_{i|1}]$ since these scenarios would give rise to smaller response times.

$$\begin{aligned} \mathrm{P}(R_{i|2}) = &\mathrm{P}(2, R_{i|2}) \\ &- \mathrm{P}(R_{i|1})\,\mathrm{P}(1, R_{i|2} - R_{i|1}) \\ &- \mathrm{P}(R_{i,|0})\,\mathrm{P}(2, R_{i|2} - R_{i|0}) \end{aligned}$$

The result is generalised as follows. The probability of exactly m arrivals in $R_{i|m}$ is derived directly from the Poisson distribution equation, $\mathrm{P}(m, R_{i|m})$. However only some permutations of arrivals can possibly lead to such a response time. The permutations which cannot lead to $R_{i|m}$ are those which would lead to a response time $R_{i|j}$ where $j < m$.

If there are j arrivals in $(0, R_{i|j}]$ then (because there are m arrivals in $(0, R_{i|m}]$) there must be $m - j$ arrivals in $(R_{i|j}, R_{i|m}]$. So, the probability of j arrivals in $(0, R_{i|j}]$ given that there are m arrivals in $(0, R_{i|m}]$ is $\mathrm{P}(R_{i|j})\,\mathrm{P}(m - j, R_{i|m} - R_{i|j})$. This value can then be subtracted from the probability $\mathrm{P}(R_{i|m})$.

The resulting general equation for the upper bound on the probability of worst case response time $R_{i|m}$ is:

$$\begin{aligned} \mathrm{P}(R_{i|m}) = &\mathrm{P}(m, R_{i|m}) \\ &- \sum_{j=0}^{m-1} \mathrm{P}(R_{i|j})\,\mathrm{P}(m - j, R_{i|m} - R_{i|j}) \end{aligned} \tag{6}$$

Finally, the probability of deadline failure for a process i is given by equation (7).

$$\mathrm{P}_i(failure) = 1 - \sum_{\forall m | R_{i|m} < D_i} \mathrm{P}(R_{i|m}) \tag{7}$$

Implementation of this is trivial, so code is not shown. A software implementation based on equations (5) and (6) was used to calculate several examples shown next.

2.7. Probability Distribution

The immediate result of the analysis is a set of pairs $R_i = \langle t_i, \mathrm{P}(t_i) \rangle$. More usefully, a cumulative probability distribution can be plotted. The cumulative probabilities represent an upper bound on the probability of the corresponding response time being exceeded.

An example from the original CAN-based analysis is presented in Figure 2. It shows the general 'stepped' shape of the output for a number of different messages at different priorities. Note that the lowest point on each line represents the probability of deadline failure, equation (7).

A further example illustrates the accuracy of the analysis and also how the analysis can consider the tail of a distribution. The curved line in Figure 3 is generated by accurate simulation, the other line is by analysis. The deviation at low priorities is caused by the simulation not generating the infrequent, but possible long response times.

3. Multiple Streams

The limitation of the analysis in Section 2 is that only one random arrival stream can be accommodated. This section explains why multiple streams is not trivial but gives

ries [7] given by the formula: $\frac{(2m)!}{m!(m+1)!}$ where m is the number of arrivals (as in $R_{i|m}$).

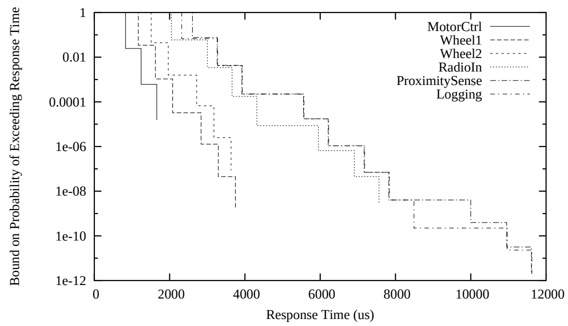

Figure 2. Example of Analysis Output

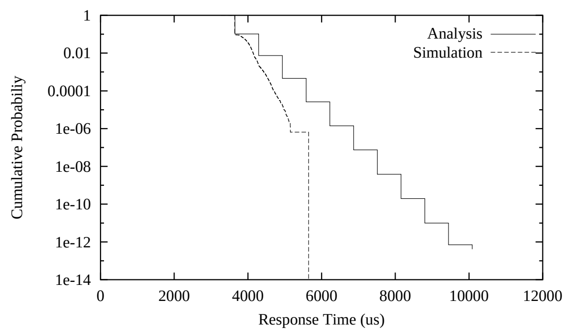

Figure 3. Comparing to Simulation

an intuition into how this scheme may be adapted for multiple streams with random arrivals.

The fundamental difficulty with multiple streams is that the number of combinations of different possible arrivals can be very large. For example, considering only two random streams, if we wish to find the probability of say $R_i > d$ then we would need to find all the possible combinations of these two streams which can give rise to response times in the range 0 to d. This might be expressed: $R_{i|0,0}$, $R_{i|1,0}$, $R_{i|0,1}$ etc. The complexity grows rapidly. Therefore, although entirely possible in theory, the practical implications make this approach redundant.

Next, we outline an approach to consider multiple streams. From equation (1), we note that the interference from arrivals for all random streams and non-random streams are simply summed. Therefore, as far as any analysis is concerned, we may neglect any indication of priority, and just consider the set of processes with higher priority than the process under analysis.

Therefore, we may attempt to group together all the random arrivals into a single arrival stream. Equation (4) for the worst case interference function of these streams becomes:

$$N(t) = A_a(t)C_a + A_b(t)C_b + \cdots \qquad (8)$$

These streams can be combined using the standard techniques for making composite probability distributions, modified slightly since each original stream may have a different worst case computation time (C_a etc.)

Finally, for this combined stream, the probability of individual response times can be computed in a similar way to before. However, given that the combined interference function is much more complex that before, it may also be necessary to group a set of response times together into one (slightly pessimistic) response time in order to maintain a simple ‘stepped’ interference function.

The details of this approach are still to be finalised, but we contend that this current work in progress will lead to a practical analysis for random arrivals in fixed priority scheduling.

4. Conclusion

In this paper, a probabilistic analysis, originally derived for fault modelling on a network, has been applied to a general fixed priority scheduling system. The analysis calculates the effect that random distributions of arrivals have on response times. The analysis has uses modelling interrupts, faults and other external interactions. The analysis has the advantages that it is simple and not computationally expensive too perform.

The original analysis for CAN was limited to one single random arrival stream, which although useful in that domain (to model transient faults on a bus), becomes a limitation in the general fixed priority framework. However, an approach is also outlined which allows the analysis of multiple random streams.

References

[1] I. Broster. *Flexibility in Dependable Communication*. PhD thesis, Department of Computer Science, University of York, York, YO10 5DD, UK, Aug 2003.

[2] I. Broster, A. Burns, and G. Rodríguez-Navas. Probabilistic analysis of can with faults. In *Proceedings of the 23rd Real-time Systems Symposium*, Austin, Texas, Dec 2002. IEEE.

[3] I. Broster, A. Burns, and G. Rodríguez-Navas. Comparing real-time communication under electromagnetic interference. In *Proceedings of the 16th Euromicro Conference on Real-Time Systems*, pages 45–52, Catania, Italy, July 2004. IEEE Computer Society, IEEE. The version in the proceedings had an error in equation (15) and Table 3. The version at `http://www.cs.york.ac.uk/rts/` is corrected.

[4] A. Burns, G. Bernat, and I. Broster. A probabilistic framework for schedulability analysis. In R. Alur and I. Lee, editors, *Proceedings of the Third International Embedded Software Conference, EMSOFT*, number LNCS 2855 in Lecture Notes in Computer Science, pages 1–15, 2003.

[5] J. L. Díaz, D. F. García, K. Kim, C.-G. Lee, L. L. Bello, J. M. López, S. L. Min, and O. Mirabella. Stochastic analysis of periodic real-time systems. In *Proceedings RTSS*, 2002.

[6] A. Leulseged and N. Nissanke. Stochastic analysis of periodic real-time systems. In *9th Intl. Conf. on Real-Time and Embeded Computing Systems and applications (RTCSA 2003)*, Taiwan, 2003.

[7] N. J. A. Sloane. Sequence a000108—catalan numbers. Available from `http://www.research.att.com/~njas/sequences/Seis.html`, 2003. On-Line Encyclopedia of Integer Sequences.