Statistical Analysis of WCET for Scheduling

Stewart Edgar* and Alan Burns
Real–Time Systems Research Group
Department of Computer Science
University of York, United Kingdom
e-mail: {sfe,burns}@cs.york.ac.uk

Abstract

To perform a schedulability test, scheduling analysis relies on a known worst-case execution time (WCET). This value may be difficult to compute and may be overly pessimistic. This paper offers an alternative analysis based on estimating a WCET from test data to within a specific level of probabilistic confidence. A method is presented for calculating an estimate given statistical assumptions. The implications of the level of confidence on the likelihood of schedulability are also presented.

1. Introduction

The traditional approach to scheduling analysis requires knowledge of an upper bound on the time taken for computation to take place to provide a guarantee of timeliness at run-time. There has been much work on improving the analysis of these systems particularly when they incorporate some of the more complex features of modern hardware [9, 12, 14]. However this need for careful and complex analysis contrasts with many situations where only measurement of the computational process may be possible or economically feasible. What is required is a method by which these measurements can be used in place of a deterministic analysis. Given measured values for the computation time of a particular task it is possible to construct a model of the length of time that it will take for this task to complete its execution [2]. It is also possible to construct an estimate for the *worst case execution time* (WCET) based only on the sample of data collected for this model. Since this estimate is not guaranteed, it must have an associated confidence level. The advantage of this approach is that we are freed from the restrictive assumptions that can make the construction of an analytically derived WCET impractical.

*This work has been partially funded by the EPSRC, via a research studentship and supported by the assistance of the DIRC project and City University.

An attempt has been made to construct a process that will take as its starting point a set of sample data, obtained through experimentation, for each task. From these data a model is constructed for the computation time of each task based on a probability density function, a mathematical expression approximating the real behaviour of the computation time [2]. From a particular model an estimate of the WCET for the task from which the sample data is taken can be derived for any chosen level of confidence that is required. A range of methods for combining these disparate pieces of information into an analysis of a system of tasks are then shown. This allows a more accurate analysis than considering all tasks separately, as the likelihood of underestimation of WCET for any particular task is low (by the nature of the problem). Hence, for a system of tasks with estimates of WCET, or prior determined values of computation for each task in a system, a value is obtained that provides a statistical measure of whether the WCET estimates are sufficient.

1.1. Objectives

The WCET is an upper bound on the greatest time that a task will take to perform its execution when it has sole access to all computational resources and under the complete set of all possible environmental conditions. It is also important to note, for the purposes of this paper, that this particular value will be the lowest possible upper bound, and be assumed to be possible under actual execution, so that under repeated executions of the same task under random environmental conditions the likelihood that the WCET will be encountered tends to 1. The assumption that timely completion of all execution for a task is required necessitates the penalty that underestimation of WCET is a problem for correct execution of a system inclusive of this task. For this reason it is helpful to add an assumption that failure due to an overload of processor resources is a possible outcome of *any* form of statistical analysis. Failure due to overrun of computation on a particular task is just one facet of a greater problem of failure that includes environmental issues and needs to be approached separately from the scheduling problem.

The restraint that this work is attempting to overcome is the requirement to have certain knowledge of the WCET. This value can only be obtained from detailed analysis of the behaviour of a task and the system in which it is expected to execute. However in many situations this detailed analysis cannot be performed. An alternative approach must be utilised due to practical difficulties, cost constraints or to a level of confidence required in high integrity applications, which for each generation of hardware becomes more difficult to ensure. Measurement of the time taken for computation is useful, as it requires no knowledge of the task in question. However any value of WCET constructed from measurement will not be certain knowledge; any measurement process will, in general, fail to be inclusive of all possible situations. If measurement is to be used then this possibility of failure must be quantified. The topic of ensuring measurement is representative of the subject in question, both generally in statistics and specifically in the situation of measuring computation time is not the issue of this paper and will not be discussed here, allowing the statistical theory employed through the rest of this paper to be applied to a larger range of potential problems. The emphasis is on the theoretical aspect of the problem as opposed to the implementation to a specific situation, although some example scenarios will be presented in later sections.

1.2. Process of Investigation

There are three stages in the construction of a schedule as seen by this paper. The first step consists of a detailed explanation of the problem at hand and an outline of the techniques used to solve it, as shown in section 2. There then follows a brief explanation of the process of modelling of data, a subject that has been examined previously [2]. Both this and the establishment of confidence levels for WCET are explained in detail in section 3. The second step is the construction of a schedule, or the application of a schedulability test. This step is not examined; the subject of scheduling algorithms has been approached from many angles in established work that are applicable to this subject [1, 7]. The major part of this work is examined in section 4, that of the construction of a confidence level argument for the system as a whole. This work is concluded with a section as to how these techniques may be applied to wider issues in scheduling in section 5 and an overview of the presented results in section 6.

2. Description of the Environment

Now that the rationale of this research has been established, the processes by which the estimation of WCET and establishment of a confidence for collections of tasks can be explained. In this section notation is given which will be used throughout this paper, together with an explicit declaration of the problem and how it is to be solved.

2.1. Notation and Task Model

Measurement and the sampling of data is a complex science, especially when there may be a subclass of extreme situations that can affect the success of modelling. There is an assumption that the measurement process is fair, which is to say; each test of computation time is assumed to be independent, the sample of computation times has been taken over a range of environmental conditions and the sample is of a size which allows generalisation to be used. The first and second requirements are conditions of the experiment while the latter is simple to achieve since this branch of statistics is usually applied to situations with sparse data. This given, a model for each task requiring computational resources in a system can be constructed. It is important to note that in this paper the value that is being modelled is not the execution time, but specifically the worst-case execution time. A system is defined as a set of tasks $\Gamma = \{\tau_1 \ldots \tau_n\}$. Each task will have associated with it a number of features, represented by the following notation which will be used in this paper.

Symbol	Meaning
τ_i	Task number i.
D_i	The deadline of τ_i, typically dictated by the specification of the task.
max_i	The greatest value in the sample of times from τ_i
θ_i	The function representing the WCET of τ_i, a random variable. This statistic will be used to construct an estimate of the WCET.
ω_i	The estimate of WCET constructed in the process described in section 3.
ϵ_i	The confidence level with which the value of ω_i is held.

2.2. Defining the Problem

There are two problems that the process described can be used to solve. The first is to take a set of deadlines provided as part of the requirements for the system of tasks and the sample data and transform these into a set of estimates of WCET and a confidence with which the entire system will meet these deadlines. By writing ξ for the total confidence with which the whole system is held and x_{ij} for data item j of the sample for task τ_i the above relationship can be written as:

$$(\{D_i\}, \{x_{ij}\}) \mapsto (\{\omega_i\}, \xi)$$

The process described above is a three stage process. The first step is taking the sample data and using this to construct an estimate of the WCET, with a specified confidence level. The second step is to construct a schedule using the figures for WCET calculated previously. If the scheduling algorithm indicates that a system is unschedulable then there is the potential to recalculate estimates of WCET for lower confidence levels. The third stage is to take both the sample data and schedule and use them to calculate the confidence with which the system as a whole can be expected to meet all deadlines, ξ.

The second possible problem is of transforming a requirement for the reliability of a whole system, ξ and the sample data and producing a range of WCET estimates that will be ideal for producing a confidence of this desired level with the minimum use of computational resources and within specified deadlines. Then there will also be the possibility of using these estimates of WCET to perform a schedulability test.

$$(\{D_i\}, \{x_{ij}\}, \xi) \mapsto (\{\omega_i\}) \mapsto \{\text{true}, \text{false}\}$$

As in the previous case, the raw data has to be used to calculate appropriate models of computation time. However, instead of calculating the value of ϵ_i for a specific ω_i, the opposite operation is performed. (Note: it will be shown later that the relationship between ω_i and ϵ_i is a 1-1 mapping.) The value of ϵ_i for each task will be derived from the system confidence ξ and hence a suitable computation time estimate ω_i will be calculated. This can then be checked back with a required deadline. Both of these different processes are shown in figure 1.

Either of these techniques can be used as is seen fit, given pre-determined knowledge, and those unknowns that an experimenter seeks to find. The next step is to build the relationship between confidence level and WCET estimate.

3. Modelling and Estimation

It is necessary to describe the method by which WCET is estimated with respect to confidence level, and the models are introduced in this section. These models are chosen as being asymptotically correct for their chosen purpose, but they also exhibit useful properties which are important, not only for the model itself, but also for later stages when the information from a series of models is combined together. These properties are shown in this section and where appropriate, proven.

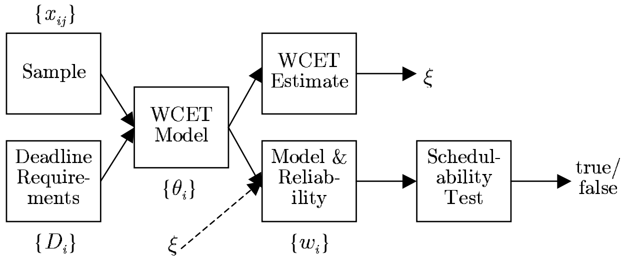

Figure 1. The process by which the two different problems can be solved using the same techniques.

3.1. WCET Estimation

The statement that an observer is '99% certain' that an event will occur is often said in conversation with little evidence that this statement is a correct one to make. Statistics provides techniques to make an assertion, and to test to see whether this assertion can be made with any surety given the data that have been observed. This *confidence level* is a value corresponding to the likelihood of a statement being true with the available evidence, and is very different from a statement of probability. For a prediction of WCET from statistical modelling there is a necessity that the value of prediction has associated with it a confidence with which that statement can be made.

When sample data are analysed, assumptions as to its underlying nature have to be made. These hypotheses can have testing applied to their veracity, this will lead to a confidence with which the hypothesis can be held. In a sense, confidence is a metric of 'belief' and it is subtly different to a value of probability. The probability of a statement S being true, say $P(S) = p$, is a mapping from the set of events onto the interval $[0, 1]$ of the real line that says that given a number of instances of S, a proportion p of them are likely to be true, and hence an individual instance of the statement will have a likelihood of either being true or false in proportions $p : (1 - p)$. A confidence is very different, and says that for a hypothesis with an associated confidence q, with which the evidence (i.e. the sample data and associated assumptions) indicates that the statement is factually correct *in all instances* with a probability of q. After the sampling of data only one fact is clearly known about the WCET of a task, it must be at least the greatest observed value, which is denoted max_i for a task τ_i.

3.1.1 Probabilistic Model

A knowledge of the average timing behaviour of a task is not an answer to the problem at hand, that of the worst-case behaviour. Thus the model that is constructed must reflect, not the distribution of the computation time, but the expected maximum computation given that the value max_i will be the smallest possible value that the statistic can take. The model chosen to represent either or both of these facets of the task in question can be chosen from a wide variety of different statistical equations. If the underlying distribution of computation times, say X is described by the function $F(x) = P(X < x)$ then the maximum of k independent executions will be described by $F^k(x) = [P(X < x)]^k$. It is the asymptotic behaviour of this function that is of interest. If there is a limit to the asymptotic behaviour then the function will be of only three different forms [4].

The particular model chosen is that provided by the Gumbel distribution [6], which by the Fisher-Tippet theorem is the limiting distribution for a random sample of independent observations which has no upper bound. The distribution of the WCET will follow the maximal model given the information that the value of the maximum will be at least max_i. To achieve this combination of features a truncated and scaled Gumbel distribution is utilised [2]. An example of this is the curve, θ_i, in figure 2, as the solid black line approaching a value of 1 as the time allowance is increased. The dotted line represents the sample data and the dashed line is a model of execution time. There are two other possible limiting distributions which can be models for the maximum of a distribution. Together all three models form a family of distributions, of which the Gumbel is the most general. For the purposes of this paper it is assumed that the Gumbel is a suitable model for the data and a standard statistical test, such as a χ^2-test has been performed [3].

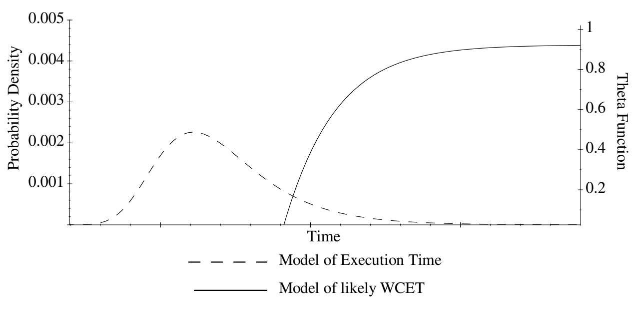

Figure 2. Graph of probability density and WCET estimate against time for hypothetical data.

Definition 1 *The Gumbel distribution function approximating the maximum of a set of sample data of the form $\{x_1, \ldots, x_k\}$, with parameters μ and σ has a cumulative distribution function representing the likelihood that the maximum will be equal to, or less than, x. This function is represented by equation (1) [4]. The function* $\exp(y)$ *represents exponentiation, i.e.* $\exp(y) \approx 2.71828^y$.

$$G_{(0,\mu,\sigma)}(x) = \exp(-\exp(-(x-\mu)/\sigma)),\ x > \mu \qquad (1)$$

3.1.2 Calculation of Parameters

A model will be constructed solely from a *random sample* of measured data. A sample will be of size n, and consist of data items $\{x_1, \ldots, x_n\}$. From this a model of the computational cost of the task is constructed using the Gumbel distribution as a statistical basis. The Gumbel distribution function is referred to as $G_{(0,\mu,\sigma)}(t)$ for variable t which will represent time. There is a requirement for three parameters, a *shape* parameter, which for the Gumbel distribution is zero, a *location* parameter, μ, and a *scale* parameter, σ. Mathematically, μ and σ must be estimated using a *statistic*, a function of the random sample [4, 5]. An *unbiased estimator* of a parameter is a function of the random sample which, under the rules of expectation, will be mathematically equal to the parameter in question. Put formally:

Definition 2 *If α is an unknown parameter of a random variable and $\hat{\alpha}$ is an estimate for this parameter then $\hat{\alpha}$ is an unbiased estimator if $E(\hat{\alpha}) = \alpha$, for all α. $E(\hat{\alpha})$ is notation for $E(f_{\hat{\alpha}}(X_1, \ldots, X_n))$, where $f_{\hat{\alpha}}$ is some function on a sample of random variables $X_1, \ldots, X_n$ [10].*

The unbiased estimators for the location parameter, μ is based on the sample mean $\bar{x}$, and for scale parameter σ is based on the sample variance $\hat{\sigma}^2$ and Euler's constant λ.

$$\mu = \bar{x} - \sigma\lambda\ , \quad \bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i\ , \quad \lambda \approx 0.5772$$

$$\sigma = \frac{\sqrt{6}}{\pi}\hat{\sigma}\ , \quad \hat{\sigma}^2 = \frac{1}{(n-1)}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

Hence it is trivial to construct a model for computation time from sample data that is deemed to be representative of the full conditions under which a task is expected to perform. The value of μ, the location parameter, is estimated by the mean of the sample while σ, the scale parameter, is estimated by the standard deviation of the sample.

3.2. Constructing a WCET Estimate

Equation (1) does not give the expected value for the worst case execution time (WCET) of a task, since this is a model of the time taken to execute once - it is necessary for the estimation of WCET to only consider values greater than the largest piece of sample data denoted by max. This value can be calculated using the *pertaining excess distribution function* over a threshold u, representing the time after a certain predetermined cut-off point.

Definition 3 *For a random variable X with distribution function F the pertaining excess distribution function over a threshold u representing the distribution of a random variable greater than a particular value can be written as equation (2).*

$$F^{(u)}(x) = P(X - u \leq x \mid X > u) \tag{2}$$

Theorem 1 *The estimate of the WCET value of task τ_i is represented by the pertaining excess distribution function with a threshold max_i of a parameterized Gumbel distribution. This function, denoted θ_i, can be written as equation (3) [11]. In this definition X is a random variable with Gumbel distribution $G = G_{(0,\mu_i,\sigma_i)}$.*

$$\theta_i(x) = \frac{G(x + max_i) - G(max_i)}{1 - G(max_i)} \tag{3}$$

Proof: The distribution function of the maximum of the sample will be $X \sim G_{(0,\mu_i,\sigma_i)}$. Hence the probability that a measurement will be equal or less than max_i will be:

$$\begin{aligned} & P(X - max_i \leq x \mid X > max_i) \\ = & \frac{P(max_i < X \leq (x + max_i))}{P(X > max_i)} \\ = & \frac{G(x + max_i) - G(max_i)}{1 - G(max_i)} \qquad \square \end{aligned}$$

Equation (3) allows a direct comparison between values of the WCET and the degree to which these values will be correct given the evidence at hand. This is the equation from which the estimate of WCET is derived. For any estimate ω_i the probability that the most extreme event (*i.e.* the WCET) will occur at, or below, this value will be dependent on the model, data and the chosen value of ω_i, as shown in equation (4). The likelihood at which we expect an estimate of WCET to be exceeded will be denoted by ϵ_i. Note that this symbol is chosen to emphasise that this value is small, not that ϵ_i corresponds to an error.

$$G^{(max_i)}_{(0,\mu_i,\sigma_i)}(\omega) = 1 - \epsilon_i \tag{4}$$

Example 1

The above techniques are shown in an example program to illustrate how estimates of a fixed quantity, the WCET, can be constructed from a random data sample. A particular program with a known structure and behaviour yet variable execution time, due to the effects of superscalar architecture features in the processor, has its computation time measured. The program under observation has a single path and hence variation in execution time is not the result of control flow issues [2]. It was run 100 times, with a maximal reading of 963, a mean of 669.025 and a standard deviation of 143.378. Forming the approximation above yields a computational model with equation (5), all values correct to three significant figures.

$$\theta_{(\mu_i,\sigma_i)}(x) = -7.28{+}8.28\exp(-\exp(-0.00697x{+}4.67)) \tag{5}$$

From this it is possible to evaluate the confidence with which we are prepared to hold the estimates that we choose for the WCET. A graph of this is shown in figure 3. Values of confidence for various times were calculated numerically and the times are shown in the table below. This particular example is chosen to be small, so it may be possible to find an upper bound for the WCET with a slight pessimism due to the assumptions that have to be made to find this. In general this value relies on a detailed knowledge of the environment in which the programme is executing, which may not be known.

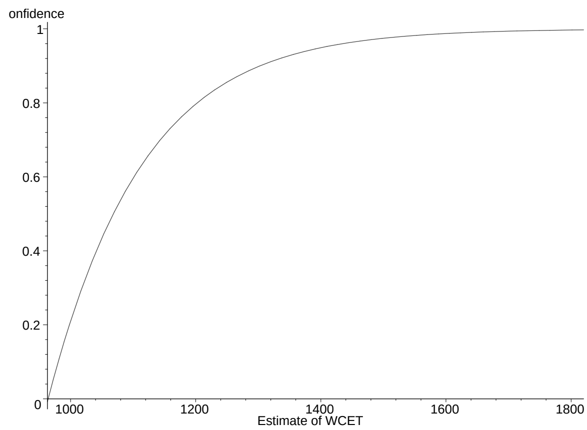

Figure 3. The graph of confidence against computation time estimate.

Estimate of WCET	Confidence Level
963	0
1000	0.216
1100	0.600
1200	0.798
1300	0.899
1400	0.950

3.3. Calculating WCET confidence to maximize system performance

Assume that tasks $\tau_1, \ldots, \tau_n$ are ordered by increasing deadline. The likelihood that the first task will meet a deadline is $P(\theta_1 < D_1)$. The likelihood that the second task will meet its deadline will depend on the amount of time taken by the first task - if the actual WCET is less than the estimate then this time will allow an underestimation of the WCET of τ_2 to be handled given enough spare time. It must be emphasized that the times of the individual tasks will not be varying at all. In actual operation it may be that all tasks take far less than the individual WCET calculated to complete and correct execution will occur with time to spare with a far greater likelihood than is being estimated. At this stage, the only concern is with the WCET behaviour and the repercussions of using estimates of a WCET value, and hence all execution times of a task must be considered potential worst-cases.

A useful early result to achieve is to value all tasks equally within the parameters we are allowed to vary. Given that these parameters are those of the statistical model chosen to represent the execution time of a task, the parameter that will have to be chosen is either ω_i, the WCET estimate, or ϵ_i, the confidence with which this estimate may be held. There will be an upper bound on the value of ω_i, the deadline, although in general it will not be possible for every task to be assigned this maximum amount of computation. In general - the value of ω_i will not be given as part of the specification. This leaves the value of ϵ_i to be calculated in the construction of a schedule. If there is a method of constructing an 'average' confidence ϵ then the expected likelihood that the system will execute correctly at run-time, using the simplest analysis, will be $o(n\epsilon)$.

$$\begin{aligned} P(\text{Exceedance of limits}) &= 1 - \prod_{i=1}^{n}(1-\epsilon_i) \\ &\approx \sum_{i=1}^{n} \frac{n!}{(n-i)!}(-1)^{i-1}\epsilon^i \\ &= o(n\epsilon) \end{aligned}$$

It may be advantageous to have varying levels of confidence in the estimates of WCET that are calculated - some tasks will be more important for the correct running of a system than others. If there is a functional dependence between individual tasks, τ_i and τ_j, then the promptness of the combined task $\tau_i\tau_j$ will depend more on the effects of τ_i than on the effects of τ_j. An assumption made at this point is that information about the dependence of tasks and their importance are not known, or given as part of the specification, and that all tasks are independent in terms of their WCET, apart from the competition for resources. It has been established that a statistical WCET estimate can be constructed from measured data given a confidence with which this estimate is to be held. The values of these confidences, denoted as ϵ_i for task τ_i, are not given but must be chosen as part of the estimation process. There will be a large number of different criteria affecting the choice of the ϵ_i due to the different nature of each task. However the problem as constructed is to maximise the timeliness of the whole system. To do this it is advantageous to choose the values for all ϵ_i to be identical. This can be proved simply by noting a few important characteristics of the functions used to model the timing behaviour of tasks in the system.

The function θ_i, representing the likelihood of an estimate of time is a monotonically increasing function that tends to 1. This is to be expected - with an increasing allowance of time the more likely from the evidence in the sample data that the estimate will prove accurate. Of more interest is the differential of the function θ_i which represents the probability density of the function above, as can be seen in figure 4. This function is monotonically decreasing for values of time greater than max, the maximum observed value. This is to expected as well. Each point along the graph (or to be more accurate; the area under the graph between two successive points around an integer value of time) represents the increase in probability that an estimate is accurate that will be gained by increasing the 'allowance' of time accorded to that task by 1 unit.

The principal to note is that in figure 4, the decreasing function means that the corresponding areas under the curve are also decreasing. That is to say, the improvement in confidence by increasing the allowance of time given to a task from t to $t+1$ will be less than the increase in confidence by increasing the allowance from $t+1$ to $t+2$. How does this result affect the estimates for a collection of tasks? Assume that there is no slack in a schedule that has been calculated using estimates for WCET - this is reasonable, since the greater the allowance for a task, the greater the confidence in the correct execution which is the desired goal. Since all time has been taken, the only way of increasing the confidence of any one task is to remove the allowance of another task. That is to say:

- τ_i has time increased from t_i to $t_i + 1$, resulting in a change in confidence from $\theta_i(t_i) = (1-\epsilon_i)$ to $\theta_i(t_i + 1)$.

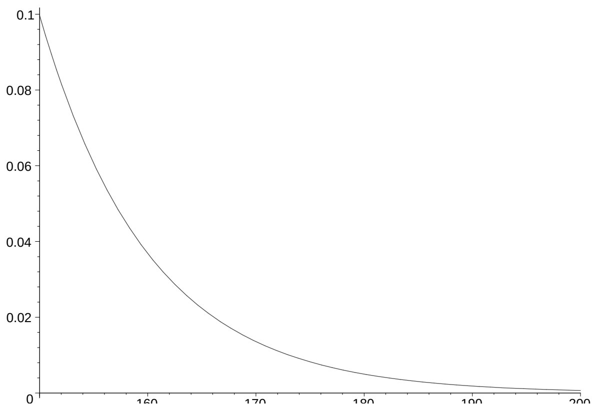

Figure 4. The graph of θ_i with respect to time t showing probability densities corresponding to WCET estimates. ($\mu = 100$, $\sigma = 10$, $max = 150$).

- τ_j has time decreased from t_j to $t_j - 1$, resulting in a change in confidence from $\theta_j(t_j) = (1 - \epsilon_j)$ to $\theta_j(t_j - 1)$.

By assumption the two tasks will have equal errors $\epsilon = \epsilon_i = \epsilon_j$. Thus the total change in confidence for the system if these two tasks will have as a lower bound:

$$\Delta = [\theta_j(t_i + 1)\theta_i(t_j - 1)] - [\theta_i(t_i)\theta_j(t_j)] \tag{6}$$

The above expression is evaluated by substituting in the definitions for theta and expanding the value of $-\Delta$. If the function G_i denotes $G_{(0,\mu_i,\sigma_i)}$, equation (6) can be expressed as:

$$\frac{f_1 + (f_2 - f_3)}{(1 - G_i(max_i))(1 - G_j(max_j))} \tag{7}$$

In equation (7) the value of the denominator is positive, since G is a probability function defined on $[0, 1]$, while the three functions f_1, f_2, f_3 are:

$$\begin{aligned} f_1 &= G_i(max_i + t_i)G_j(max_j + t_j) \\ &\quad -G_i(max_i + t_i + 1)G_j(max_j + t_j - 1) \\ f_2 &= G_j(max_j)[G_i(max_i + t_i + 1) - G_i(max_i + t_i)] \\ f_3 &= G_i(max_i)[G_j(max_j + t_j) - G_j(max_j + t_j - 1)] \end{aligned}$$

Both f_1 and $(f_2 - f_3)$ will be positive since the advantage gained from increasing a WCET estimate by one unit of time is less than the disadvantage of reducing another estimate by one (see figure 4). For $-\Delta > 0$, there is negative improvement in the confidence of a system by altering the allowance for each task in a system, and so the optimal solution will be found for equal values of error, *i.e.* when $\epsilon_i = \epsilon_j$.

These three results are important for the scheduling problem that is faced. We now know that instead of having to choose the individual ϵ_i confidence levels for n different tasks in a system, the problem is reduced to choosing a single $\bar{\epsilon}$ for the whole system. This will simplify the scheduling problem, as after the process of modelling has been completed the difficulty remains in choosing a value of $\bar{\epsilon}$ to ensure schedulability.

4. Scheduling with WCET Estimates

The previous sections have established a method of transforming sample data into models of computation time for specific computational tasks, and how to develop these models into a process of forming a relationship between the estimate of WCET and the confidence with which this estimate can be held. Lastly it was shown that a schedule can be constructed by choosing an identical confidence in all of the tasks in a system. Now it remains to show how this analysis will combine with the information about scheduling in a system to allow a confidence with which a constructed schedule can be held.

4.1. Apportioning Computation Time

There are, of course, many different ways of scheduling a system of tasks. In this section we examine a simple scheme to emphasise the statistical implications over more complex scheduling issues. However the results presented are generally applicable - with the caveat that some details may have to be modified. We assume the non-preemptive execution of a series of tasks $\{\tau_1, \ldots, \tau_n\}$. The task sequence must be completed by a deadline, $\bar{D}$. This is exactly the scheme within a cyclic executive and is the basic behaviour of many other schemes. Context switch times are either assumed to be zero, or if non-zero are used to reduce the effective value of $\bar{D}$. Each task has a separate deadline D_i, so that $\max D_i = \bar{D}$. A statistically modelled task will be based on a function which can vary over a range of possible values. It is helpful to have bounds under which the estimate of WCET can vary. Each task can have an estimate between a minimum value, which will be max the greatest value obtained during measurement, and a maximum value. This will be the greatest time between the deadline of a task and the earliest possible invocation of a task, determined either by an offset, O_i, or by the sum of all other prior executed tasks. (Mathematically this value is the sum of measured values max_j, $j < i$.) Hence:

$$\omega_i \in \left[max_i, \min\left(D_i - \sum_{j<i} max_j, D_i - O_i\right)\right] = \left[max_i, D_i - \max\left(\sum_{j<i} max_j, O_i\right)\right]$$

Given that the same confidence $\bar{\epsilon}$ is required for all tasks in a system, a method is required for apportioning computation time between the separate tasks to achieve this requirement. The following algorithm achieves the goal correctly, if inefficiently.

1. Note that the deadline $\bar{D} > \sum max_i$, so there is no evidence from the measured data to show that the task set will be unschedulable.
2. Assign task τ_i the time max_i, so that there is a minimum measured time to allow execution to take place in these situations.
3. Calculate the probability that each estimate of WCET is accurate given the evidence.
4. Choose the smallest confidence level and give a spare unit of time to this task, if the upper bound on computation time has not yet been reached. (If more than one task has the smallest amount, as is the situation at the beginning, chose the task with the smallest σ_i).
5. Repeat 3 and 4 until all spare computational resource is used.

The result will be an increase in the amount of time that it is estimated that the WCET will be. This increase will not be in proportion to the size of the original allocation, the value of the original maximal observed computation time from the sample, but instead will be increased by an analysis of the sample data. To see this process in action it is useful to give two examples.

Example 2

As an example a set of data is constructed to show the process of calculating the 'allowances' of computation time for each task. A value of $\bar{D} = 650$ is chosen.

Task	μ_i	σ_i	max_i
τ_1	50	10	85
τ_2	100	20	115
τ_3	250	15	315

One unit of computation is initially given to τ_1, then τ_3, then τ_2. Computation is then given as in the algorithm above. For the task set the result is:

Task	max_i	ω_i	$\theta_i(\omega_i)$	% increase
τ_1	85	114	0.9442	34.1
τ_2	115	177	0.9441	53.9
τ_3	315	359	0.9464	14.0

Example 3

In a second example, two tasks modelled on real-world data [2] are used to show the applicability of the method to non-constructed data. A value of $\bar{D} = 1900$ is chosen.

Task	μ_i	σ_i	max_i
τ_1	609.301	162.568	911
τ_2	524.405	116.11	780

Task	max_i	ω_i	$\theta_i(\omega_i)$	% increase
τ_1	911	1034	0.5113	13.5
τ_2	780	866	0.5094	11.0

4.2. Calculating System Confidence

A model exists by which a task requiring computational resources can be represented as a parameterised function. From this model it is possible to calculate the probability that a chosen WCET estimate is correct (confidence), or vice-versa. This allows a probabilistic assessment of a single source of uncertainty. What needs to be ascertained is; what are the effects of a number of modelled tasks in a single system?

If there is a system of tasks in a system with no slack in the schedule, what is the likelihood that the system as a whole will behave as expected even if the actual WCET of individual tasks differ from the estimates used in the construction of a schedule? A simple method of calculating the overall likelihood that the system will complete is to ignore the interaction between the two tasks and require that they both finish execution in the expected time. Under these circumstances the confidence level is the product $\prod_{i=1}^{n} \epsilon_i$. For example if every one of n tasks in a system has a confidence level of $1 - 10^{-k}$ then the confidence of the system as a whole will be $(1 - 10^{-k})^n$. For example for 10 tasks each with a computation time estimate with a 99% confidence level the lower bound on system level scheduling will $(0.99)^{10} \approx 0.904$.

As may be deduced this is an overly pessimistic approach. It is possible to exploit an important effect of the modelling procedure, that a WCET for a particular task is far more likely to be overestimated than underestimated. The result that is required, of this section of the paper, relates the confidences for individual tasks and the system as a whole. It will enable us to prove a statement along the lines of "a system of tasks each with a WCET estimated to a confidence of $1 - 10^{-k}$ that has a schedulability guarantee means the confidence with which the system can be said to behave timely at run-time will also be $1 - 10^{-k}$."

Definition 4 *If a system of n tasks $\Gamma = \{\tau_1 \ldots \tau_n\}$, each with a WCET estimated to a confidence of $(1 - \epsilon)$, has a schedulability guarantee and the confidence with which the* *system can be said to behave timely at run-time is also* $(1 - \epsilon)$ *then this system of tasks is said to be composable with confidence* $(1 - \epsilon)$ *and* $\Gamma \in S_n$.

As the above definition suggests a more formal theorem can be proved using a inductive argument on n, the number of different tasks in a system. To help in this proof it is useful to make two additional definitions.

Definition 5 *Let* c_1 *be the probability that a system* Γ *is still composable with confidence* $(1 - \epsilon)$, $\Gamma \in S_{n+1}$, *given an overestimation of the first* n *tasks and an underestimation of the* $n + 1$ *task.*

Definition 6 *Let* c_2 *be the probability that a system* Γ *is still composable with confidence* $(1 - \epsilon)$, $\Gamma \in S_{n+1}$, *given an underestimation of the first* n *tasks and an overestimation of the* $n + 1$ *task.*

The value of the first situation is likely to be relatively high and the value of the second situation relatively low, since the cumulative likelihood of n tasks being underestimated is $o(\epsilon^n)$, i.e. in relative terms, insignificant. With these values in mind it is now possible to construct a proof of the following theorem. The advantage of this proof over other methods is that we only require a similar degree of timeliness, rather than the calculation of an accurate probability value.

Theorem 2 *If* $c_1 + c_2 \geq 1$ *then for a system of tasks each with a WCET estimated to a confidence of* $(1 - \epsilon)$, *that also has a schedulability guarantee, the confidence with which the system can be said to behave timely at run-time will be at least* $(1 - \epsilon)$.

Proof: The case of $n = 1$ is trivially true, if there is only one task then the system will be schedulable with a value $(1 - \epsilon)$. Now assume that the above statement is true for n tasks, S_n is true. What is required is a proof that it will also be true for $(n + 1)$ tasks. Assume that the $(n + 1)$ tasks are independent of each other. This is a restrictive assumption in the sense that, if the method of observation under which the results were obtained is not fair then the possibility is that the errors encountered in this stage of modelling are dependent on each other.

Hence a situation exists similar to that shown in figure 5. The first n tasks, shown as one task, τ_n will be initiated at time $t = 0$ and finish execution at a time estimated to be $t = \sum_{i=1}^{n} \omega_i$. At this point task τ will be initiated. (Remember, there is an assumption that there is no penalty for context switching.) This task has been estimated to complete by a time $t = D$. There is no slack in the schedule. There thus exist four different situations.

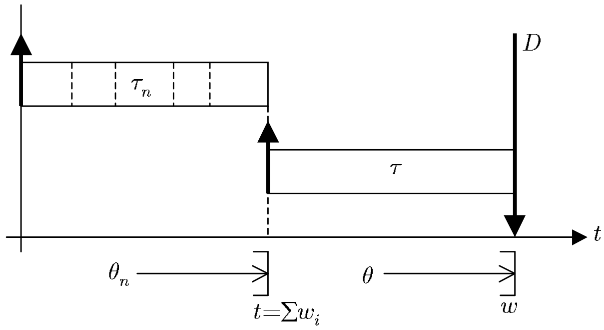

Figure 5. Diagram of two tasks against time.

- Both the first n tasks and the $(n + 1)$th task are overestimated, *i.e.* take less time than expected. In this case the resulting system will still be schedulable in all cases.
  $P(S_{n+1} \mid \tau_n \text{ over}, \tau \text{ over}) = [(1 - \epsilon) \times (1 - \epsilon) \times 1]$
- Both the first n tasks and the $(n + 1)$th task are underestimated. In this case the system will definitely not be schedulable.
  $P(S_{n+1} \mid \tau_n \text{ under}, \tau \text{ under}) = [\epsilon \times \epsilon \times 0]$
- The first n tasks are underestimated and the $(n + 1)$th task overestimated. There remains a question of schedulability in this scenario.
  $P(S_{n+1} \mid \tau_n \text{ under}, \tau \text{ over}) = [\epsilon \times (1 - \epsilon) \times c_2]$
- The first n tasks are overestimated and the $(n + 1)$th task is underestimated. There remains a question of schedulability in this scenario.
  $P(S_{n+1} \mid \tau_n \text{ over}, \tau \text{ under}) = [(1 - \epsilon) \times \epsilon \times c_1]$

Since all four situations are mutually exclusive, the probability that the resulting system of tasks will be composable will be the total of the four probabilities. The total probability will be $(1 - \epsilon)[1 - \epsilon + \epsilon(c_1 + c_2)]$ which we require to be at least $(1 - \epsilon)$. In other words we require $[1 - \epsilon + \epsilon(c_1 + c_2)]$, which will be positive if $c_1 + c_2 \geq 1$. $\square$

5. Future Work

When the requirement to have a known upper bound on computation time is removed then the ability to give a qualitative answer to the question of schedulability is also removed. Instead there will be a quantitative or probabilistic approach where schedulability can be seen as a possibility with a value of the likelihood that schedulability will occur with a specified time frame, such as the hyper-period. Alternatively schedulability can remain a fixed event and the confidence with which the evidence supports this view can be the metric used in the assessment of the system. Also, the analysis until this point has depended on the assumption of a single equal deadline by which all tasks have to complete all computation. This is a restrictive assumption to make and an adjustment to the analysis is required to create a method that is applicable to a greater range of scheduling problems. The following stage of research is to show that this technique offers the potential for improved scheduling of real-time systems particularly where the majority of the work performed is of a soft or firm real-time nature.

6. Conclusions

This paper explains a process by which sampling the execution times of a particular task allows the construction of an estimate by which a value for WCET can be estimated. This value will either be dependent on the choice of the confidence with which this value can be held, or on the amount of time that is assigned to a task to complete its execution. Calculating estimates of WCET for individual tasks is useful. However the result that is required is not the likelihood that an individual task performs correctly, an entire system has to work within estimates to be acceptable. If all actual computation times C_i are less than or equal to the estimates of WCET, ω_i, then a system will be schedulable. This will provide a lower bound on the likelihood of successful execution, but it will not take account of the overestimation of values for WCET cancelling out the underestimation that may occur. An important additional result is thus found, that the confidence required for a whole system of tasks is equal to that which should be chosen for individual tasks as long as the modelling process is fair and independent. The conflicts between tasks are investigated by starting out by examining the conflict between a task and those executing previously and then using the result of this to generalise behaviour for a greater number of tasks. The method examined is a simple method, which is still pessimistic, but provides a suitable lower bound for the overall confidence level of a system.

References

[1] N. C. Audsley, A. Burns and A. J. Wellings. Deadline Monotonic Scheduling Theory and Application. *Control Eng. Practice*. Vol. 1, No. 1, pp. 71–78, 1993.

[2] A. Burns and S. Edgar. Predicting Computation Time for Advanced Processor Architectures. *Proc. of the 12th Euromicro Conference on Real-Time Systems*, pp. 89–96, 2000.

[3] R. B. D'Agostino and M. A. Stephens (editors). Goodness-of-fit Techniques. Statistics: textbooks and monographs, vol. 68, Dekker, 1986.

[4] R. A. Fisher and L. H. C. Tippett. Limiting forms of the frequency distribution of the largest or smallest member of a sample. *Proceedings of the Cambridge Philosophical Society*. Vol. 24, pp. 180–190, 1928.

[5] J. Galambos. The Asymptotic Theory of Extreme Order Statistics. Wiley, 1978.

[6] E. J. Gumbel. Statistics of Extremes. *Columbia University Press*. 1958.

[7] M. Joseph and P. Pandya. Finding Response Times in a Real-Time System. *The Computer Journal*, 29(5), pp. 390–395, 1986.

[8] C. L. Liu and J. W. Layland. Scheduling Algorithms for Multiprogramming in a Hard-Real-Time Environment. *JACM*, 20(1), pp. 46–61, 1973.

[9] C. Y. Park. Predicting program execution times by analyzing static and dynamic program paths. *Real–Time Systems*, vol. 5, pp. 31–62, 1993.

[10] J. Pickands. Statistical inference using extreme value order statistics. *Ann. Statist.* 3, pp. 119–131, 1975.

[11] R.-D. Reiss and M. Thomas. Statistical Analysis of Extreme Values. *Birkhäuser* 1997.

[12] J. Sun and J. Liu. Bounding Completion Times of Jobs with Arbitrary Release Times and Variable Execution Times. *Proc. IEEE Real–Time Systems Symposium 1996*, pp. 2–12.

[13] M. H. Woodbury Analysis of Execution Time of Real–Time Tasks *Proc. IEEE Real–Time Systems Symposium 1986*, pp. 89-96.

[14] N. Zhang, A. Burns, and M. Nicholson. Pipelined Processors and Worst-Case Execution Times. *Real-Time Systems*, 5(4):319–343, Oct. 1993.