THE·UNIVERSITY·OF

WARWICK

Research Report 396

THE IMPACT OF PREDICTIVE INACCURACIES ON EXECUTION SCHEDULING

Stephen A Jarvis, Ligang He, Daniel P Spooner and Graham R Nudd

Department of Computer Science
University of Warwick, Coventry CV4 7AL, UK

The Impact of Predictive Inaccuracies on Execution Scheduling*

Stephen A. Jarvis, Ligang He, Daniel P. Spooner and Graham R. Nudd

High Performance Systems Group, Department of Computer Science,

University of Warwick, Coventry CV4 7AL, UK

Email: Stephen.Jarvis@warwick.ac.uk

Tel: +44 (0)2476 524258 Fax: +44 (0)2476 573024

Abstract: This paper investigates the underlying impact of predictive inaccuracies on execution scheduling, with particular reference to execution time predictions. This study is conducted from two perspectives: from that of job selection and from that of resource allocation, both of which are fundamental components in execution scheduling. A new performance metric, termed the *degree of misperception*, is introduced to express the probability that the predicted execution times of jobs display different ordering characteristics from their real execution times due to inaccurate prediction. Specific formulae are developed to calculate the degree of misperception in both job selection and resource allocation scenarios. The parameters which influence the degree of misperception are also extensively investigated. The results presented in this paper are of significant benefit to scheduling approaches that take into account predictive data; the results are also of importance to the application of these scheduling techniques to real-world high-performance systems.

Index terms: performance prediction, execution time, scheduling, job selection, resource allocation, and performance evaluation

---

* This work is sponsored in part by grants from the NASA AMES Research Center (administrated by USARDSG, contract no. N68171-01-C-9012), the EPSRC (contract no. GR/R47424/01) and the

1. Introduction

Scheduling in a single processor environment is, at its most basic level, the task of determining the sequence in which jobs should be executed. In a multi-processor or multi-computer environment on the other hand, job scheduling also involves the process of resource allocation, that is, determining the resource to which a job should be sent for execution. The design of scheduling policies for parallel and distributed systems has received a good deal of attention [2, 4, 5, 14, 15]. These schemes are often based on the assumption that the job execution times are known somehow [5, 9]. While this is a useful assumption to make, information of this type is often unknown and must therefore be obtained through some predictive mechanism. A naïve approach to this problem might be to require the owner of the job to estimate the execution time based on their past experience. A more sophisticated approach might be to utilize performance prediction tools to perform this function. A number of increasingly accurate prediction tools have been developed that are able to predict the execution times of jobs based on performance models [3, 6, 7, 8] or historical data [1, 13].

In spite of this, an inevitable fact is that the prediction results are unlikely to be entirely accurate and as a result, this may have a fundamental impact on job selection and resource allocation.

In the case of job selection, the inaccurate prediction may mean that the scheduler has an incorrect view of the order in which the different jobs should execute. For example, it may be the case that the real execution time of job J_1 is greater than that of job J_2; because of the inaccurate prediction however, the scheduler may view job J_1 as having a shorter execution time than that of job J_2. Therefore, if the scheduling policy is based on job execution times (the shortest job serviced first, for example), then this

---

EPSRC e-Science Core Programme (contract no. GR/S03058/01).

misperception will impact on the order in which jobs are selected for execution. This will ultimately influence the scheduler and system performance.

When a scheduler receives a job in a parallel or distributed system, there may be a number of resources (processors or computers) available on which the job may be executed. If the resource allocation policy is also based on the expected execution time of the job on the different resources (select the computer that offers the shortest execution time, for example) then these inaccuracies might also cause the scheduler to wrongly select between them. Again, this misperception will impact on the scheduling and overall system performance.

The existence of misperception originates from the inaccurate prediction (of in this case execution time), and should be viewed as an inherent characteristic of any prediction-based scheduling scheme that operates in a highly-variable real-world system. This said, different scheduling policies will have different levels of sensitivity to the degree of misperception. Thus, the study of the impact of inaccurate prediction on scheduling performance can be decomposed into two levels – at the underlying level, as a study into the degree of misperception originating from inaccurate prediction; and at the higher level, as a study into the sensitivity of individual scheduling policies to this degree of misperception. This paper addresses the former, where the latter is the subject of future work.

Different predicted errors can lead to different degrees of misperception. The objective of this paper is to establish the relationship between the predicted error and the degree of misperception, for both job selection and resource allocation scenarios. This study provides an insight into the underlying impact of inaccurate prediction on job selection and resource allocation and significantly benefits the design and evaluation of scheduling policies that use predictive data.

The rest of this paper is organized as follows. Formula for the degree of misperception, and for job selection and resource allocation are developed in Section 2. The parameters that influence the degree of misperception are then extensively evaluated in Section 3. The paper concludes in Section 4.

2. An Analysis of the Degree of Misperception

2.1 Job Selection

When performance prediction tools are used to estimate the execution times of jobs, the predicted execution time usually lies in an interval around the actual execution time (of the job) according to some probability distribution [7, 8].

Suppose that the actual execution time of job J_i is x_i and that the predictive error, denoted by y_i, is a random variable in the range $[-ax_i, bx_i]$ following some probability density function, $g_i(y_i)$; where the possible value fields of a and b are [0, 100%] and [0, ∞), respectively. It is assumed that the predictive errors of different jobs are independent random variables. The predicted execution time of job J_i, denoted by z_i, is computed using Eq.1.

$$z_i = x_i + y_i \tag{1}$$

Suppose two jobs J_1 and J_2 have the actual execution times x_1 and x_2, where $x_1<x_2$. Then the predicted execution times of J_1 and J_2, that is z_1 and z_2, are x_1+y_1 and x_2+y_2, respectively. Given that $x_1<x_2$, a misperception happens if $z_1 \geq z_2$. The *degree of misperception for these two jobs*, denoted by $MD(x_1, x_2)$, is defined by the probability that $z_1 \geq z_2$ while $x_1<x_2$. This probability is denoted by $P_r(z_1 \geq z_2 | x_1<x_2)$. With Eq.1, this probability can be further transformed using Eq.2.

$$P_r(z_1 \geq z_2 | x_1<x_2)=P_r(x_1+y_1 \geq x_2+y_2 | x_1<x_2)=P_r(y_1 \geq y_2+x_2-x_1 | x_2-x_1>0) \tag{2}$$

so that the $MD(x_1, x_2)$ is computed by Eq.3.

$$MD(x_1, x_2)= P_r(y_1 \geq y_2+x_2-x_1 | x_2- x_1>0) \quad (3)$$

From Eq.3, it can be seen that the probability that the misperception happens is the probability that given $x_2-x_1>0$, the predictive error of J_1 (i.e., y_1) is greater than the predictive error of J_2 (i.e., y_2) plus the difference between x_2 and x_1. By constructing the coordinates of the predictive error y_1 and y_2, the inequality $y_1 \geq y_2+x_2-x_1$ means that y_1 and y_2 are given values from the area above the line $y_1=y_2+x_2-x_1$.

Fig.1.a, 2.a and 3.a illustrate the relation of predicted execution times of J_1 and J_2, while Fig.1.b, 2.b and 3.b show the corresponding value fields of the predictive error of J_1 and J_2 (y_1 and y_2) as well as the corresponding area in which $y_1 \geq y_2+x_2-x_1$.

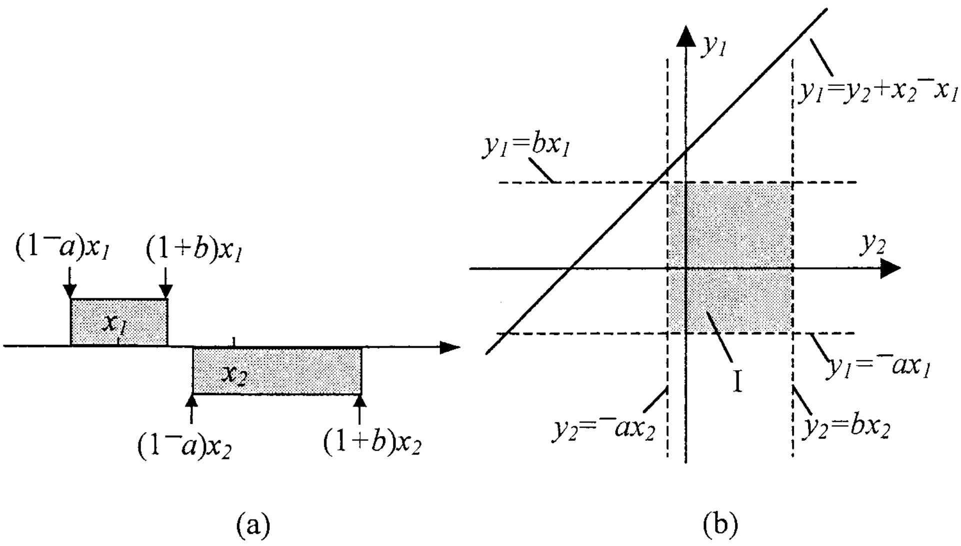


Figure.1. Case study 1 (a) the predicted execution times of jobs J_1 and J_2 do not overlap; (b) the corresponding coordinate area which the predictive errors y_1 and y_2 can be assigned values from.

Fig.1.a illustrates the case when the ranges of predicted execution times of J_1 and J_2 do not overlap. In this case a misperception will not occur even if the predictions are not accurate. Correspondingly, Fig1.b shows the coordinate area of the predicted errors of J_1 and J_2 (area I), which is the area surrounded by the lines: $y_1=-ax_1$, $y_1=bx_1$,

y_2=$-ax_2$ and y_2=bx_2. As can be seen from the figure, all of area I is below the line y_1=y_2+x_2–x_1. Hence, $P_r(y_1\geq y_2+x_2-x_1|\ x_2-x_1>0)$ in Eq.2 is equal to zero. This case is expressed more formally below.

$$P_r(y_1\geq y_2+x_2-x_1|\ x_2-x_1>0)=0 \qquad -ax_2>-bx_1+x_2-x_1,\ x_2-x_1>0 \tag{4}$$

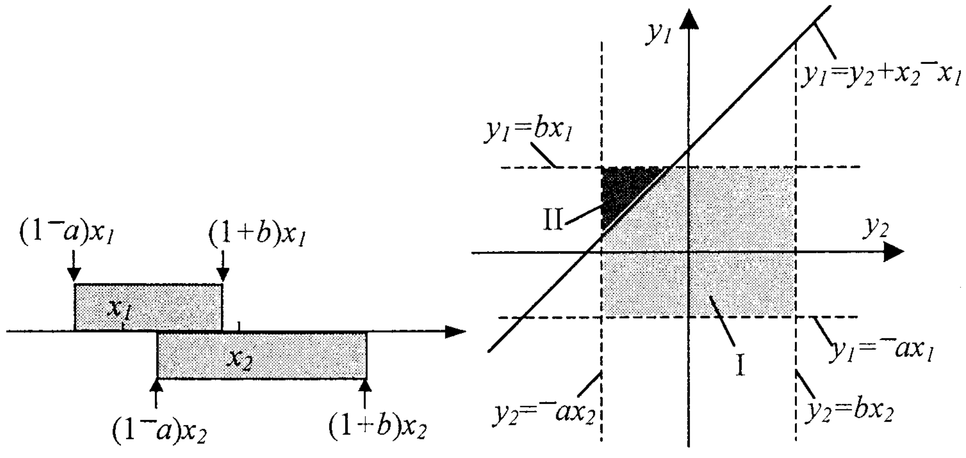


Figure.2. Case study 2 (a) the predicted execution times of jobs J_1 and J_2 overlap, however the lower limit of the predicted execution time for J_2 does not cover x_1; (b) the corresponding coordinate area of predicted errors of J_1 and J_2 (y_1 and y_2) in which the misperception occurs (the area is a triangle).

Fig.2.a illustrates the case when the predicted execution times of J_1 and J_2 overlap and the lower limit of the predicted execution time of J_2 is greater than x_1. The corresponding coordinate area of y_1 and y_2 is shown in Fig.2.b (area I). In area I, part of the area is above the line y_1=y_2+x_2–x_1 (area II), which is itself surrounded by the three lines: y_1=bx_1, y_2=$-ax_2$ and y_1=y_2+x_2–x_1. When y_1 and y_2 are assigned values from area II, y_1 and y_2 satisfy $y_1\geq y_2+x_2-x_1$, that is, a misperception will occur; a misperception will not occur if y_1 and y_2 are assigned values from any other area in I, although prediction errors will still exist. The probability in Eq.2 equals the double integral of the probability density functions of predicted errors (i.e., $g_1(y_1)$ and $g_2(y_2)$) on area II, which is calculated in Eq.5.

$$P_r(y_1 \geq y_2+x_2-x_1 \mid x_2-x_1>0)=\int_{-ax_2+x_2-x_1}^{bx_1}\int_{-ax_2}^{y_1-(x_2-x_1)} g_1(y_1)g_2(y_2)dy_2dy_1$$

$$-ax_1-(x_2-x_1)<-ax_2\leq bx_1-(x_2-x_1),\ x_2-x_1>0 \qquad (5)$$

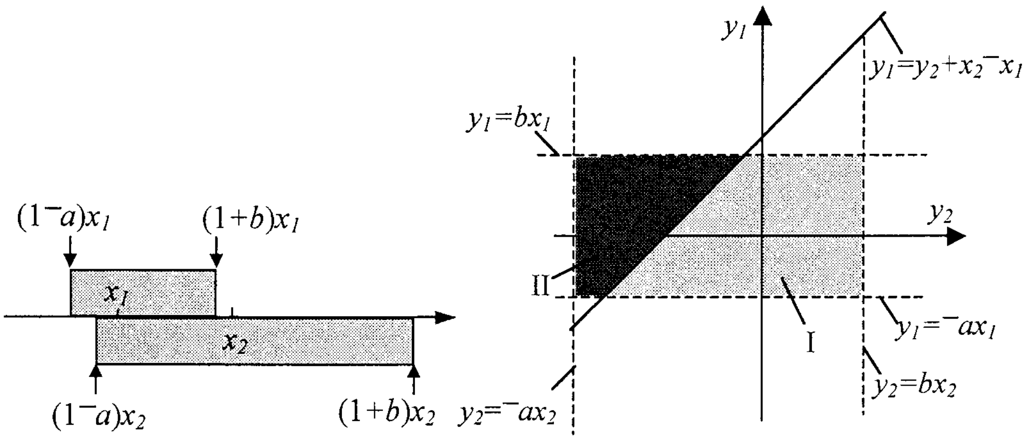


Figure.3. Case study 3 (a) the predicted execution times of jobs J_1 and J_2 overlap and the lower limit of predicted execution time of J_2 covers x_1; (b) the corresponding coordinate area of predicted errors of J_1 and J_2 (y_1 and y_2) in which the misperception occurs (the area is a trapezoid).

Fig.3.a illustrates a third case when the predicted execution times of J_1 and J_2 overlap. The difference from Fig.2.a is that the value field of J_2's predicted execution time covers x_1. The corresponding coordinate area in which y_1 and y_2 satisfy $y_1 \geq y_2+x_2-x_1$ is highlighted as area II in Fig.3.b. The area is a trapezoid rather than a triangle as in Fig.2.b. The formula for calculating the probability in Eq.2 is also different from Eq.5; this is shown in Eq.6.

$$P_r(y_1 \geq y_2+x_2-x_1 \mid x_2-x_1>0)=\int_{-ax_1}^{bx_1}\int_{-ax_2}^{y_1-(x_2-x_1)} g_1(y_1)g_2(y_2)dy_2dy_1$$

$$-ax_2\leq -ax_1-(x_2-x_1),\ x_2-x_1>0 \qquad (6)$$

Eq.4-6 account for all possible relations between the predicted execution times of jobs J_1 and J_2.

Suppose that the probability density function of a jobs' actual execution time in a job stream is $f(x)$ and that the value field of the real execution time x is $[xl, xu]$. The *degree of misperception of this job stream*, denoted by $\overline{MD}$, is defined by the average of the degree of misperception for any two jobs in the job stream. $\overline{MD}$ is computed using Eq.7, where $MD(x_1, x_2)$ is computed using Eq.4-6.

$$\overline{MD} = \int_{xl}^{xu}\int_{x_1}^{xu} f(x_1)f(x_2)MD(x_1,x_2)dx_2dx_1 \tag{7}$$

There are several independent parameters in Eq.7, as well as in Eq.4-6: a, b, xu, xl, $f(x)$, and $g_i(x_i)$. It is highly beneficial to study how these parameters influence the value of $\overline{MD}$; this is the subject of the investigation in Section 3.

2.2 Resource allocation

In dedicated environments, the execution time of one unit of work can be represented as a predicted point value [10, 11, 12]. However, in non-dedicated environments, the existence of background workloads on the resources causes a variation in unit execution times [3, 10, 11, 12]. Hence it can be assumed that the actual execution time of one unit of work locates across a range around the predicted point value following a certain probability [10, 12, 16].

Suppose there is a distributed system consisting of n heterogeneous computers c_1, c_2, …, c_n. Computer c_i is weighted w_i ($1 \le i \le n$), which represents the time it takes to perform one unit of computation. Now suppose for any i, j ($1 \le i, j \le n$), $w_i < w_j$ if $i < j$. A job with size s is therefore predicted to have the execution time sw_i on computer c_i. The predicted execution time of a job is denoted by z_{ci}, that is,

$$z_{ci} = sw_i \tag{8}$$

However, in shared environments the actual execution time for a job with size s on computer c_i may not be sw_i, because of the existence of background workload. The actual execution time of a job on c_i is therefore denoted by x_{ci}.

For a job with size s, its predicted error on computer c_i, denoted by y_{ci}, is computed using Eq.9:

$$y_{ci}=z_{ci}-x_{ci} \tag{9}$$

Suppose y_{ci} falls in the range $[-sw_i\times a, sw_i\times \mathrm{b}]$ following the probability density function $g_{ci}(y_{ci})$. Hence, x_{ci} locates in the range $[sw_i\times(1-a), sw_i\times(1+b)]$.

For two computers c_i and c_j, suppose that $w_i<w_j$. Then, the predicted execution time of a job with size s on c_i and c_j satisfy $sw_i<sw_j$. However, the range of the job's actual execution time on computer c_i is $[sw_i\times(1-a), sw_i\times(1+b)]$ and may overlap with that on computer c_j, which is $[sw_j\times(1-a), sw_j\times(1+b)]$. Consequently, the actual execution time on computer c_i may be greater than that on c_j. In this case, the inaccurate predictions cause a misperception in the order of the actual execution times on these two computers. Depending on the individual scheduling algorithm, this misperception may lead to the wrong selection of a computer to which the job is sent. Similarly, the *degree of misperception for a job with size s on two computers c_i and c_j*, denoted by $MD_c(c_i, c_j)$, is defined by the probability that $x_{ci}\geq x_{cj}$ while $z_{ci}<z_{cj}$. This probability is denoted by $P_r(x_{ci}\geq x_{cj}|z_{ci}<z_{cj})$, which can be further transformed by Eq.10 with Eq.8 and 9.

$$P_r(x_{ci}\geq x_{cj}|z_{ci}<z_{cj})=P_r(z_{ci}-y_{ci}\geq z_{cj}-y_{cj}|z_{ci}<z_{cj})=P_r(y_{cj}\geq y_{ci}+sw_j-sw_i|\ w_j-w_i>0) \tag{10}$$

That is, $MD_c(c_i, c_j)$ is computed using the following equation:

$$MD_c(c_i, c_j)=P_r(y_{cj}\geq y_{ci}+sw_j-sw_i|\ w_j-w_i>0) \tag{11}$$

Applying a similar method to that used to compute $MD(x_1, x_2)$, the equation for computing $MD_c(c_i, c_j)$ is expressed formally as follows.

$$MD_c(c_i,c_j)=\begin{cases}0 & bsw_j \le -asw_i + s(w_j - w_i) \\ \int_{-asw_i}^{bsw_j - s(w_j - w_i)} \int_{y_{ci}+s(w_j-w_i)}^{bsw_j} g_{ci}(y_{ci}) g_{cj}(y_{cj}) dy_{cj} dy_{ci} & \\ & -asw_i + s(w_j - w_i) < bsw_j < bsw_i + s(w_j - w_i) \\ \int_{-asw_i}^{bsw_i} \int_{y_{ci}+s(w_j-w_i)}^{bsw_j} g_{ci}(y_{ci}) g_{cj}(y_{cj}) dy_{cj} dy_{ci} & bsw_j \ge bsw_i + s(w_j - w_i)\end{cases} \tag{12}$$

The *degree of misperception for a job with size s for n heterogeneous computers* $c_1, c_2, \ldots, c_n$,denoted by $\overline{MD_c}$, is defined using the average of the degree of misperception for the job on any two computers, which can be computed using Eq.13.

$$\overline{MD_c} = \frac{1}{C_n^2} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} MD_c(c_i, c_j) \tag{13}$$

3. An Evaluation of the Degree of Misperception

In the previous section the general formula for the degree of misperception for both job selection and resource allocation were presented. In this section an investigation is conducted as to how the parameters in these formulae impact on the value assigned to the degree of misperception.

3.1 Job Selection

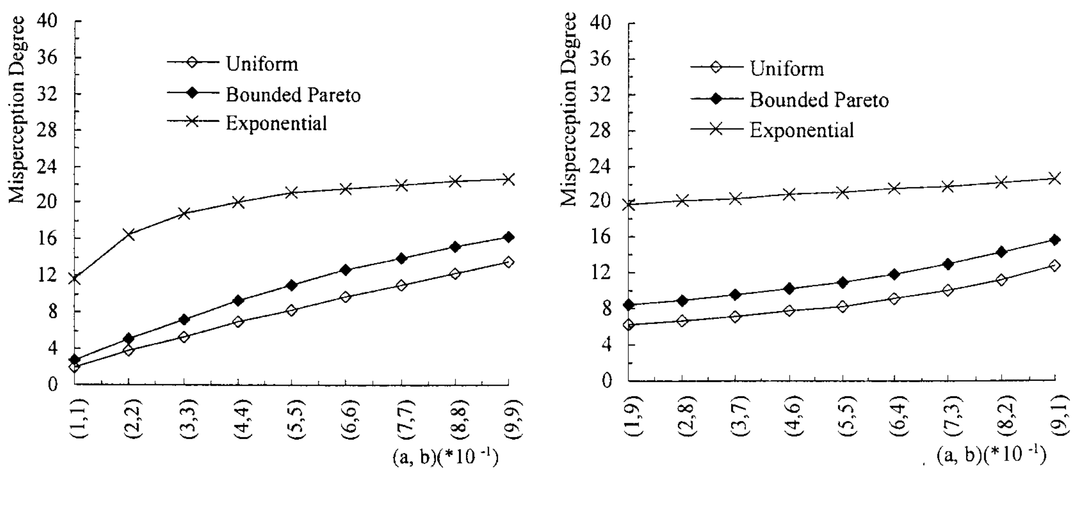
The parameters a and b represent the range of predicted errors in Eq.4-6. Fig.4.a and Fig.4.b show the impact of the parameters a and b on the value $\overline{MD}$. It is difficult to evaluate the impact of these parameters if the probability density function of predicted errors takes a general form. Therefore in the following parameter evaluation, the predicted error for the execution time of x is assumed to follow a uniform distribution in $[-ax, bx]$, whose probability density function $g_x(y_x)$ is expressed as:

$$g_x(y_x) = \frac{1}{(b+a)x}$$

Three types of job stream are investigated. The actual execution times in these job streams follow a uniform, Bounded Pareto and Exponential distribution, respectively. Their probability density functions $f(x)$ are shown in Table 1. In the job stream following the exponential distribution, the job execution time has no upper limit. Only execution time values in [10, 14.6] are considered, as 99% of the execution times locate in this range according to its probability density function. This simplification does not impact on the accuracy of the results.

Table 1. The range of job execution times in the three job streams

	Uniform	Bounded Pareto	Exponential
$f(x)$	$\frac{1}{xu - xl}$	$\frac{\alpha \times xl^{\alpha}}{1-(xl/xu)^{\alpha}} x^{-\alpha-1}$ $(\alpha = 1)$	$\frac{1}{\beta} e^{-(x-xl)/\beta}$ $(\beta = 1)$
$[xl, xu]$	[10, 100]	[10, 40]	[10, 14.6]



(a) (b)

Figure.4. Impact of the parameters *a* and *b* on $\overline{MD}$ (a) the impact of the range size of predicted errors (b) the impact of the range location of predicted errors

In Fig.4.a, a and b increase from 10% to 90% with increments of 10%. This means that the range of predicted error for the actual execution time of x increases from $[-0.1x, 0.1x]$ to $[-0.9x, 0.9x]$, while the average predicted error remain unchanged (at 0).

As can be observed in Fig.4.a, under all three probability distributions, the degree of misperception increases as a and b increase. This is because as a and b increase the predicted execution times of jobs have a higher probability of overlapping with each other, which leads to the overall increase in $\overline{MD}$. This result suggests that when the average predicted error is the same, the range size of predicted errors is critical for the value of $\overline{MD}$.

It can also be observed that under the same a and b, the degree of misperception is highest under the exponential distribution, second highest under the Bound Pareto distribution and the lowest under the uniform distribution. This is because the size of the range of actual execution times is smallest when the execution times follow an exponential distribution and the largest when following a uniform distribution. This result suggests that the actual execution times will also influence the degree of misperception. This is demonstrated in Fig.5.

In Fig.4.b, the range size of the predicted error for the actual execution time of x remains unchanged (at x), while the location of the range shifts towards the left from $[-0.1x, 0.9x]$ to $[-0.9x, 0.1x]$. The result of this is that the degree of misperception increases as the range location shifts leftwards (see Fig.4.b). The reason for this is as follows. Consider Fig.2.a and Fig.3.a. When (a, b) is (0.1, 0.9), the range of predicted errors for x_1 and x_2 are $[-0.1x_1, 0.9x_1]$ and $[-0.1x_2, 0.9x_2]$, respectively. The size of the range where the predicted execution times for x_1 and x_2 overlap is

$$0.9x_1+0.1x_2-(x_2-x_1)$$

When (a, b) is (0.9, 0.1) however, the range of predicted errors are $[-0.9x_1, 0.1x_1]$ and $[-0.9x_2, 0.1x_2]$, respectively, and the size of the overlapping ranges is

$$0.9x_2+0.1x_1-(x_2-x_1)$$

Since x_2 is greater than x_1, hence,

$$0.9x_1+0.1x_2-(x_2-x_1)<0.9x_2+0.1x_1-(x_2-x_1)$$

this means that in general the size of the overlapping ranges is greater when (a, b) is (0.9, 0.1) than when (a, b) is (0.1, 0.9). This therefore leads to the increased degree of misperception. This result suggests that compared with an overestimation of execution time, the same level of underestimation may result in a higher degree of misperception.

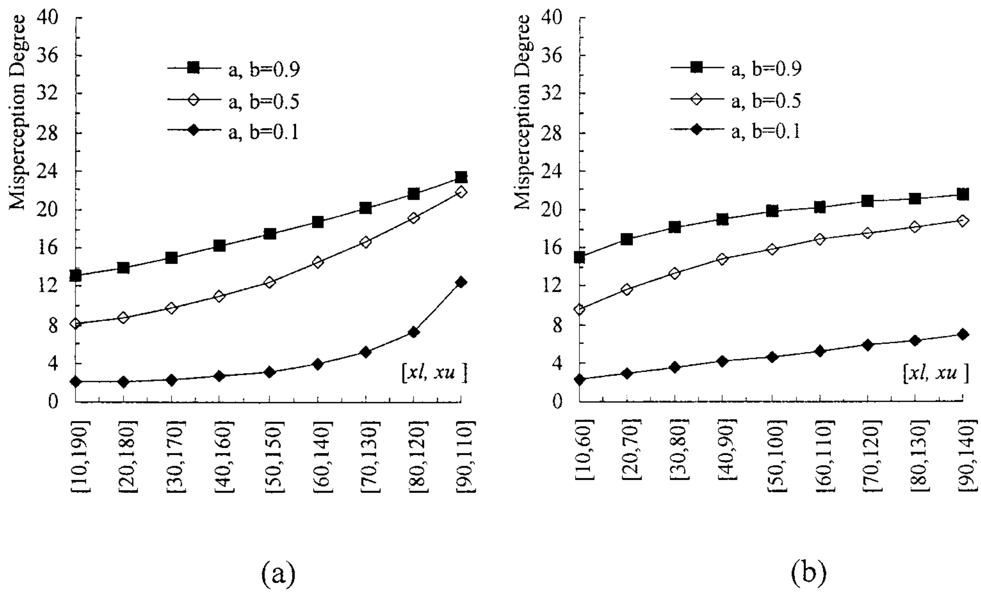


Figure.5. The impact of actual execution times on the degree of misperception (a) the impact of the range size of actual execution times (b) the impact of the range location of actual execution times

Fig.5.a and Fig.5.b show the impact of actual execution times on the degree of misperception. The results show the data for the actual execution times following a uniform distribution; the results for the Bounded Pareto distribution display a similar pattern. This study does not consider the execution times with an exponential distribution, since their range size ($xu-xl$) is fixed when 99% of the execution times are considered.

Fig.5.a shows the impact of the size of the range of actual execution times (i.e., $xu-xl$). In this same figure, the average of the actual execution times remains the same (at 100) while the range size of execution times decreases from 180 to 20 with decrements of 20. This experiment is conducted with different values of a and b. It can be observed in Fig.5.a that for the same values of a and b, the degree of misperception increases as the range size decreases. The reason for this is as follows: as the range size of the actual execution times decrease, the value of x_2-x_1 (in Fig.2.a or Fig.3.a) decreases on average under the same a and b. As a result of this, the overlapping area of the two predicted execution times increases, which finally leads to an increase in $\overline{MD}$. This result suggests that when the average execution times are the same, a greater variance in execution time is of benefit, as this will reduce the degree of misperception.

Fig.5.b demonstrates the impact of the location of the range of actual execution times. In Fig.5.b, the range size of execution times remains constant (at 50) while the range shifts from [10, 60] to [90, 140]. As can be seen in Fig.5.b, the degree of misperception increases in all cases as the range location shifts from [10, 60] to [90, 140]. The reason for this is that as the range location shifts, the mean execution time increases. Under the same a and b, the larger the actual execution time, the greater the range of its predicted error. Consequently, corresponding predicted execution times

have higher probability of overlapping with each other, which then incurs a higher degree of misperception. This result shows that when other parameters remain constant, the job stream with the greater average execution time tends to cause the highest degree of misperception.

3.2 Resource allocation

In Eq.12 and 13, the parameters that influence $\overline{MD_c}$ include the error range parameters a and b, the computer weight w_i and the probability density function of predicted errors $g_{ci}(y_{ci})$. In the following experiments, the values of these parameters are given in Table 2 unless otherwise stated.

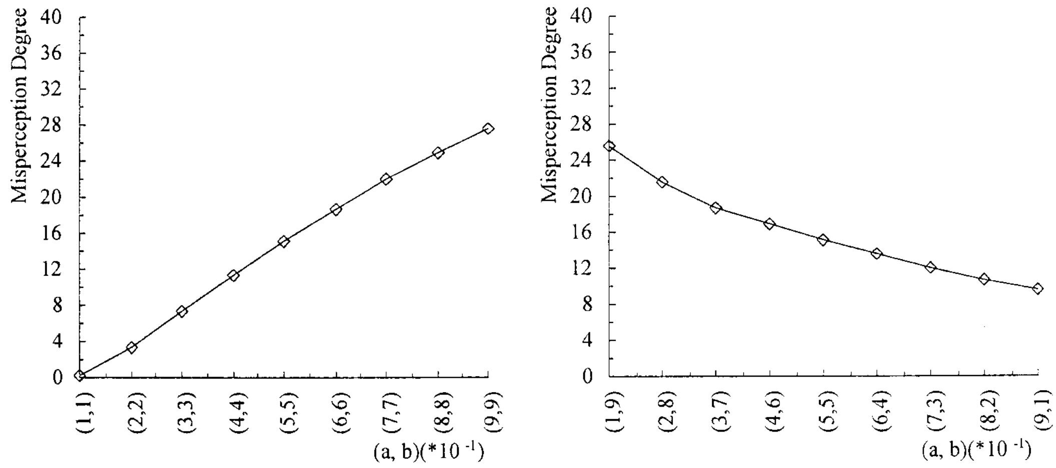
Table 2. The default values of the parameters for the experimentation

w_1	w_i-w_{i-1} ($2 \le i \le n$)	n	s
10	5	6	50

In the following figures, the predicted error for the execution time of x is also assumed to follow a uniform distribution in [$-asw_i$, bsw_i], whose probability density function $g_{ci}(y_{ci})$ is expressed as follows:

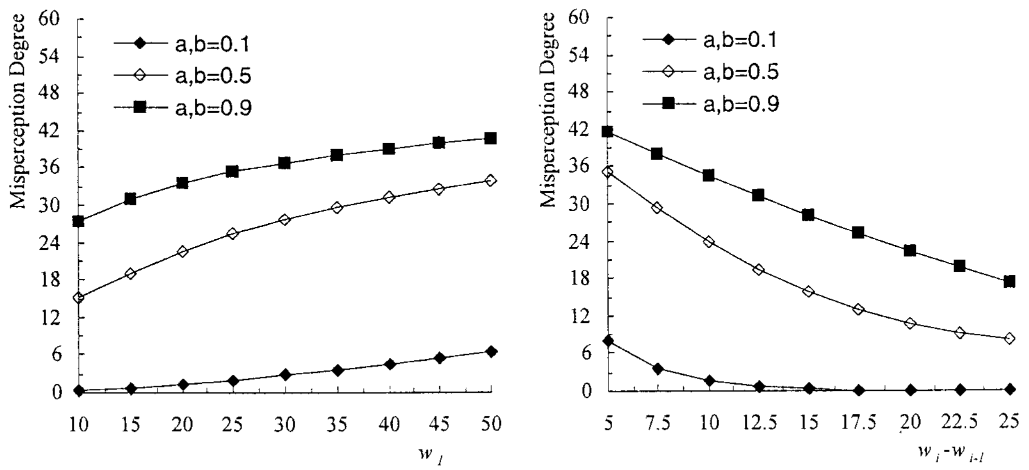
$$g_{ci}(y_{ci}) = \frac{1}{(b+a)sw_i}$$

The parameters a and b indicate the range of the predicted error. Fig.6.a shows the impact of the range size on $\overline{MD_c}$. The result has a similar pattern to that seen in Fig.4.a, which suggests that the range size of predicted errors is also critical for the value of $\overline{MD_c}$. Similarly to Fig.4.b, Fig.6.b demonstrates the impact of the range location on $\overline{MD_c}$. In the study of resource allocation, [$sw_i(1-a)$, $sw_i(1+b)$] represents the range of actual execution times. Hence, the process where (a, b) shifts from (1, 9) to (9, 1) means that the predicted execution time sw_i changes gradually from an underestimate to an overestimate. In Fig.6.b it can be seen that $\overline{MD_c}$ decreases as (a, b) shifts from (1, 9) to (9, 1). These results coincide with those seen in Fig.4.b, in that compared with an overestimate of execution time, the same level of underestimation may incur a higher degree of misperception.



(a) (b)

Figure.6. Impact of the parameters *a* and *b* on $\overline{MD_c}$ (a) the impact of the range size of predicted errors (b) the impact of the range location of predicted errors



(a) (b)

Figure.7. The impact of computer weight on $\overline{MD_c}$ (a) the impact of the size of computer weights (b) the impact of the weight difference between computers

Fig.7.a shows the impact of computer weight on $\overline{MD_c}$. In Fig.7.a, the difference of the weight between computer c_i and c_{i-1} is fixed (at 5). As w_1 increases (which means that resource c_1 becomes slower), the weights of all the remaining computers increase accordingly. As can be observed in Fig.7.a, $\overline{MD_c}$ increases as w_1 increases under all values of a and b. This is because as w_i increases, the range of the actual execution time ($[sw_i(1-a), sw_i(1+b)]$) also increases. This in turn increases the probability that the range of actual execution times on different computers overlap, which finally leads to an increased $\overline{MD_c}$. This result suggests that slower computers tend to generate higher degrees of misperception than faster computers.

Fig.7.b demonstrates the impact of weight difference between computers. In Fig.7.b, the difference between w_i and w_{i-1} ($2 \le i \le n$) increases while the mean of these computer weights remains constant (at 70). It can be observed from this figure that $\overline{MD_c}$ decreases as w_i-w_{i-1} increases. This is because as w_i-w_{i-1} increases, the difference between the predicted execution times on two computers c_i and c_j (i.e., sw_j-sw_i) increases, which in turn reduces the probability that the ranges of their actual execution times overlap. This result suggests that resource pools with higher heterogeneity will achieve a lower degree of misperception.

4. Conclusions

This paper investigates the underlying impact of inaccurate prediction on job selection and resource allocation. A new performance metric, termed the *degree of misperception*, is introduced in order to facilitate this exposition. General formulae have been developed in order to calculate the degree of misperception for a variety of job streams and for distributed resource pools of varying levels of heterogeneity. The parameters that influence the degree of misperception are also investigated. This study will benefit the design and evaluation of different scheduling mechanisms for parallel and distributed systems that take prediction into account. It is likely that different scheduling policies will have different levels of sensitivity to this degree of misperception. Further work is planed to investigate how individual scheduling policies and specific performance measures are affected by this new performance metric.

References

[1] D.A Bacigalupo, S.A Jarvis, L He, G.R Nudd, "An Investigation into the Application of Different Performance Prediction Techniques to e-Commerce Applications," Submitted to the *International Workshop on Performance Modelling, Evaluation and Optimization of Parallel and Distributed Systems*, 18th IEEE International Parallel and Distributed Processing Symposium 2004 (IPDPS'04), April 26-30, Sante Fe, New Mexico, USA.

[2] J Cao, D.P Spooner, S.A Jarvis, S Saini, G.R Nudd. "Agent-based Grid Load Balancing using Performance-driven Task Scheduling," *17th IEEE International Parallel and Distributed Processing Symposium 2003 (IPDPS'03)*, April 22-26, Nice, France.

[3] Linguo Gong, Xian-He Sun and Edward F. Watson, "Performance Modelling and Prediction of Nondedicated Networking Computing," IEEE Transactions on Computers 2002, pp. 1041-1055.

[4] Ligang He, S.A Jarvis, G.R Nudd, "Dynamic Scheduling of Parallel Real-time Jobs by Modelling Spare Capabilities in Heterogeneous Clusters," *2003 IEEE International Conference on Cluster Computing* (Cluster 2003), December 1-4, 2003.

[5] Ligang He, S.A Jarvis, D.P Spooner, G.R Nudd, "Performance-based Dynamic Scheduling of Hybrid Real-time Applications on a Cluster of Heterogeneous Workstations," *International Conference on Parallel and Distributed Computing* (*Euro-Par 2003*), 26th - 29th August 2003, Klagenfurt, Austria.

[6] S. A Jarvis, D. P Spooner, HN Lim Choi Keung, J Cao, S Saini, GR Nudd. "Performance Prediction and its use in Parallel and Distributed Computing Systems," *International Workshop on Performance Modelling, Evaluation and Optimization of Parallel and Distributed Systems*, 17th IEEE International Parallel and Distributed Processing Symposium 2003 (IPDPS'03), April 22-26, Nice, France.

[7] G.R. Nudd, D.J.Kerbyson et al, "PACE-a toolset for the performance prediction of parallel and distributed systems," *International Journal of High Performance Computing Applications, Special Issues on Performance Modelling*, 14(3), 2000, 228-251.

[8] E. Papaefstathiou, D.J. Kerbyson, G.R. Nudd, D.V. Wilcox, J.S. Harper, S.C. Perry, "A Common Workload Interface for the Performance Prediction of High Performance Systems," *IEEE International Symposium On Computer Architecture, Workshop on Performance Analysis in Design (PAID'98)*, Barcelona, 1998.

[9] X. Qin and H. Jiang, "Dynamic, Reliability-driven Scheduling of Parallel Real-time Jobs in Heterogeneous Systems," *30th International Conference on Parallel Processing*, Valencia, Spain, September 3-7, 2001.

[10] Jennifer M. Schopf and Francine Berman, "Performance Prediction in Production Environments," *12th International Parallel Processing Symposium on Parallel and Distributed Processing (IPPS/SPDP '98)*, Orlando, Florida, 1998.

[11] Jennifer M. Schopf and Francine Berman, "Using Stochastic Information to Predict Application Behavior on Contended Resources," *International Journal of Foun-*

dations of Computer Science, Special Issue on Parallel Distributed Computing, 12(3), 2001, 341-364.

[12] Jennifer M. Schopf and Francine Berman, "Stochastic Scheduling," *SuperComputing '99*, November 13-19, Portland Oregon, USA.

[13] W. Smith, I. Foster, and V. Taylor, "Predicting Application Run Times Using Historical Information," *Proceedings of the 4th Workshop on Job Scheduling Strategies for Parallel Processing*, 1998.

[14] D.P Spooner, S.A Jarvis, J Cao, S Saini, GR Nudd. "Local Grid Scheduling Techniques using Performance Prediction," *IEE Proc. Comp. Digit. Tech.*, 150(2):87-96, 2003.

[15] X.Y. Tang, S.T. Chanson, "Optimizing static job scheduling in a network of heterogeneous computers," *29th International Conference on Parallel Processing (ICPP 2000)*, August 21-24, Toronto, Canada.

[16] Lingyun Yang, Jennifer M. Schopf, and Ian Foster, "Conservative Scheduling: Using Predicted Variance to Improve Scheduling Decisions in Dynamic Environments," *SuperComputing 2003*, November 15-21, Phoenix, Arizona, USA.