COMPUTER PROGRAM EXECUTION TIME AND DISCRETE CLOCK

M. W. Chan

IBM Corporation

## ABSTRACT

In validating a simulation model of a computer system, measurements of execution times for program modules are often required. Clocks available today for such purposes run discretely. They click at certain time intervals, some on the order of microseconds, others in milliseconds. For technical as well as economical reasons, the one we use clicks every 16.666 milliseconds. Therefore, if both the beginning and ending of the execution of a program module fall inside the 16.666 ms interval between two consecutive clicks, the execution time T will be recorded as zero. On the other hand, if the beginning of the execution occurs shortly before a click, the recorded execution time will be 16.666 ms, no matter how short the actual execution time may be, as long as it ends after the click. Based on such unreliable data, a procedure is derived to yield a reliable estimate of T.

## STATEMENT OF THE PROBLEM

Clocks for measuring the execution time of computer programs operate similar to industrial time clocks in that they run discretely. Instead of clicking every minute as most time clocks do, they click at much shorter time intervals, some on the order of microseconds, others in milliseconds. For technical as well as economical reasons, the one we use clicks every 16.666 milliseconds. Therefore, if both the beginning and ending of the execution of a program module fall inside the 16.666 ms interval between two consecutive clicks, the execution time T will be recorded as zero. On the other hand, if the beginning of the execution occurs shortly before a click, the recorded execution time will be 16.666 ms, no matter how short the actual execution time may be, as long as it ends after the click. For simplicity but without too much loss of reality, it usually can be assumed that, for a given program module, its execution time is approximately constant which is unknown and to be estimated.

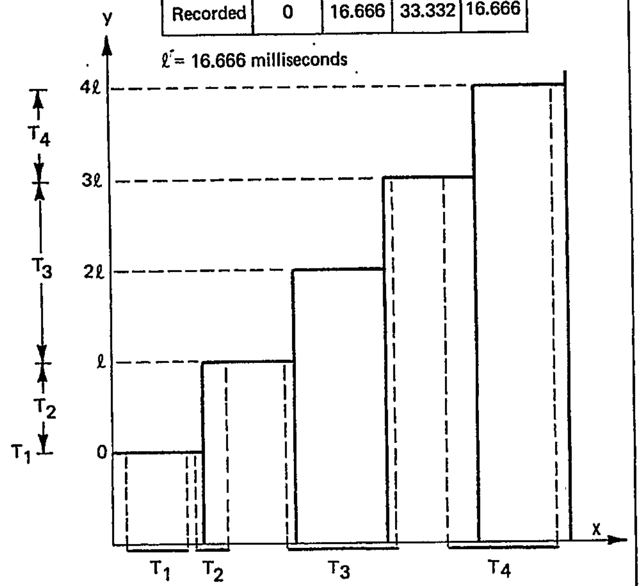
Briefly, the problem can be stated in terms of the following questions:

1. Is it possible to estimate T, based on such inaccurate data due to the discrete clock?
2. If the answer to question 1 is yes, what then is the estimate of T?
3. Further, assuming we are satisfied that the true value of T will fall within 10% of our estimate with probability of 0.95, what is the sample size n required to achieve it?
4. If we drop our confidence level from 0.95 to, say, 0.80, what then is the new sample size n needed?
5. For a fixed confidence level and again using 10% as the tolerance, what is the new n required if T increases or decreases?

ILLUSTRATION 1

Distortion of Program Module Execution Time due to the Use of Discrete Clocks

| | Execution Time (ms) | | | |
|---|---|---|---|---|
| | $T_1$ | $T_2$ | $T_3$ | $T_4$ |
| Actual | 10 | 5 | 20 | 20 |
| Recorded | 0 | 16.666 | 33.332 | 16.666 |



## SOLUTION

Let $\ell$ represent 16.666 ms, the length of time interval between two consecutive clicks, then we note that for any value of T between 0 and $\ell$, our observed values, denoted by random variable X, can only be 0 or $\ell$ with the corresponding probabilities

$$\frac{\ell - T}{\ell} \quad \text{and} \quad \frac{T}{\ell} \;,$$

respectively. Similarly, for any value of T between $k\ell$ and $(k+1)\ell$, $k = 1, 2, 3, \ldots$, the observed values can only be $k\ell$ or $(k+1)\ell$ with the same corresponding probabilities except the T value used in calculating the probabilities is $T@\ell$, where @ denotes modulo division in which the quotient is discarded and the remainder is retained.

In summary, we have the following:

For $0 \leq T < \ell$,

| X | Probabilities |
|---|---|
| 0 | $\frac{\ell - T}{\ell}$ |
| $\ell$ | $\frac{T}{\ell}$ |

For $\ell \leq T < 2\ell$,

| X | Probabilities |
|---|---|
| $\ell$ | $\frac{\ell - (T@\ell)}{\ell}$ |
| $2\ell$ | $\frac{T@\ell}{\ell}$ |

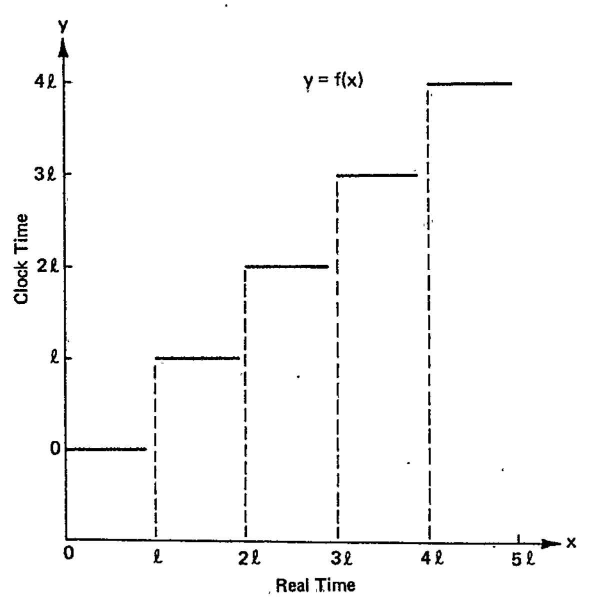
For $k\ell \leq T < (k+1)\ell$, $k = 0, 1, 2, \ldots$

| X | Probabilities | |
|---|---|---|
| $k\ell$ | $\frac{\ell - (T@\ell)}{\ell}$ | or $1 - p$ |
| $(k+1)\ell$ | $\frac{T@\ell}{\ell}$ | or p |

where $\frac{T@\ell}{\ell}$ is denoted by p.

Now we have arrived at the following conclusion. Suppose a program module has been executed n times, and the module execution time is recorded as $(k+1)\ell$ with a frequency of d out of the n times, and recorded as $k\ell$ with a frequency of $(n-d)$ times. We can say the true module execution time is between $k\ell$ and $(k+1)\ell$ and its best estimate is

$$\left(k + \frac{d}{n}\right)\ell \;.$$




This is based on the fact that X is a binomial random variable, the estimate of its parameter p is d/n.

Let $\hat{p} = d/n$, then for fairly large n, the normal approximation applies, and we have the following statement giving an approximate 95 percent confidence interval for p;

$$P\left\{\hat{p} - 1.96\sqrt{\frac{\hat{p}(1-\hat{p})}{n}} < p < \hat{p} + 1.96\sqrt{\frac{\hat{p}(1-\hat{p})}{n}}\right\} \cong 0.95 \;. \qquad (1)$$

Converting statement (1) to correspond to the appropriate confidence interval for T, we note that

$$T = k\ell + T@\ell = k\ell + p\ell \qquad (2)$$

and its estimate $\hat{T}$ is

$$\hat{T} = k\ell + \frac{d}{n}\ell = k\ell + \hat{p}\ell \;. \qquad (3)$$

Combining (1), (2), and (3) we have

$$P\left\{\hat{T} - \ell\left(1.96\sqrt{\frac{\hat{p}(1-\hat{p})}{n}}\right) < T < \hat{T} + \ell\left(1.96\sqrt{\frac{\hat{p}(1-\hat{p})}{n}}\right)\right\} = 0.95 \quad . \tag{4}$$

With our tolerance fixed at 10% of $\hat{T}$, we have

$$\ell\left(1.96\sqrt{\frac{\hat{p}(1-\hat{p})}{n}}\right) = 0.10\hat{T} = 0.10(k+\hat{p})\ell$$

$$1.96\sqrt{\frac{\hat{p}(1-\hat{p})}{n}} = 0.10(k+\hat{p})$$

$$n = \frac{100\,(1.96^2)\,\hat{p}(1-\hat{p})}{(k+\hat{p})^2} \quad . \tag{5}$$

Equation (5) is the answer to questions 1, 2, and 3 of our problem stated earlier.

To illustrate the use of (5), suppose we have recorded the execution of a program module 2000 times, 400 of which we observed by the discrete clock to have an execution time of 16.666 ms, the rest being 0 ms. We thus have the following:

$$k = 0$$

$$\hat{p} = \frac{400}{2000} = 0.2$$

$$\hat{T} = k\ell + \hat{p}\ell = 0.2 \times 16.666 = 3.333 \text{ ms} \quad .$$

The question is simply: Is the sample size, in this case 2000, large enough so that we can say with confidence (i.e., with probability 0.95) the true value of module execution time is within 10% of our estimate of 3.333 ms ?

Employing (5), the minimum sample size required is

$$n = \frac{100\,(1.96^2)\,(0.2)\,(0.8)}{(0.2)^2}$$

$$= \frac{384.16\,(0.8)}{0.2}$$

$$\cong 1536 \quad .$$

Hence, our sample size of 2000 is indeed large enough to make the confidence statement.

If we lower the confidence level from 0.95 to 0.90, 0.85, 0.80, then by replacing the figure 1.96 in (5) with appropriate numbers we obtain respectively:

$$n = \frac{100\,(1.645^2)\,\hat{p}(1-\hat{p})}{(k+\hat{p})^2} \quad , \tag{5a}$$

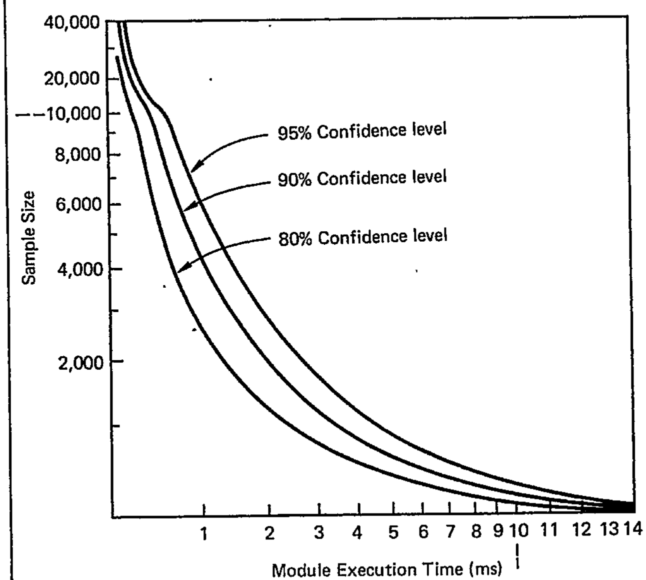
$$n = \frac{100\,(1.44^2)\,\hat{p}(1-\hat{p})}{(k+\hat{p})^2} \quad , \tag{5b}$$

$$n = \frac{100\,(1.28^2)\,\hat{p}(1-\hat{p})}{(k+\hat{p})^2} \quad . \tag{5c}$$

For questions 4 and 5 posted earlier, the solutions are provided in graphic form in Illus. 3. The curves show the relationship between sample size required (y-axis) and the estimated module execution time (x-axis). The different curves represent different confidence levels, while they all maintained the same 10% tolerance.

ILLUSTRATION 3

Relationship between Sample Size and Module Execution Time




Note: For module execution time of 15 ms or longer, a sample size of 45 is sufficient for all cases.