Measuring with Slow Clocks

Heinz Beilner*

TR-88-003
July, 1988

ABSTRACT

This report describes a measurement technique and corresponding statistical evaluation options that can be used for assessing the mean duration of performing a particular operation, even when this duration is small compared with the resolution of an available, readable clock. The technique has been developed with regard to measuring operation durations of distributed system kernels, and to measuring durations of sub-activities embedded in these operations. The technique employs repetitive executions of the measured operation, but does not however depend on the usually employed "tight loop" around the operation. It also allows for simultaneous assessments of several different time intervals within the repetitive pattern. Based on an initial guess about the mean length of the smallest time interval to be measured, the necessary number of loop cycles can be determined before an experiment, for a selectable width of the confidence interval of the mean to be estimated, and at a selctable confidence level.

* Presently at: International Computer Science Institute, Berkeley
Permanent affiliation: Informatik IV, Universität Dortmund, Germany

JULY 1, 1988

1. Motivation

The ideas for this report arose during a visiting period with the DASH project at UC Berkeley. DASH studies general issues in the design of large, high-performance distributed systems, and is at the same time building an experimental system. An overview over the DASH project is given in ANFE88, additional references are supplied there.

In every such project, there is an obvious interest in experimentally verifying that all performance goals for the system are being met by the growing implementation. For distributed systems, interprocess communication performance is particularly interesting and, hence, latency and throughput/bandwidth figures for corresponding kernel operations need to be determined.

Measuring the time taken for executing any such operation can be difficult due to its small duration. Moreover, interpretations (and subsequent improvements) of operation durations necessitate a quantified break-down of operations, i.e. the assessment of shorter, operation-embedded sub-activities. With 1 MIPS and faster processors, sub-activities of about 50 μsec and below may well have to be measured.

Ideally, then, a good measurement environment would be asked for, such as the one used for measurements of the Accent system, cf. FIRA86, where special (profiling) versions of the system were employed, and where a high resolution (1 μsec) clock was available.

In the absence of such an environment, statistical techniques have to be employed, sometimes based on fairly low resolution (1...10 msec) clocks. A widely used measurement approach consists of executing the operation to be assessed "in a tight loop", i.e. repeating the operation over and over again. The low frequency clock is read before and after the loop, and the duration of the operation of interest (basically, of one loop cycle) estimated from the number of loop cycles executed and the total time used for executing the totality of loop cycles. For any clock frequency, a "sufficiently" high cycle number should exist to offset the low precision of the clock, for any required result precision.

The sketched approach may work well for larger, "total" operations. It fails, however, when sub-activities of operations must be assessed: Not every sub-activity lends itself for being measured within "its own" tight loop as the execution environment within the loop becomes increasingly atypical with shrinking size (buffer, cache, paging effects can only be reproduced to a lesser and lesser degree). One alternative for assessing sub-activities despite these difficulties consists of combining the tight loop measurement of the total operation with inspections of the (assembly level) code executed, in order to arrive at a quantitative break-down of execution time. This approach was for instance (in addition to further measurement instrumentation) applied in SCCO87. As easily understandable, a fully consistent, quantitative break-down of operations into sub-activities is still hard to obtain.

In the DASH environment, SUN 3 workstations have been used for the first implementations. On these workstations, a low resolution (10 msec) clock is available by standard. In addition, an adjustable frequency clock has been installed employing facilities of a presently unused RS232 port. This timing feature has been implemented by Shin-Yuan Tzou, who is also to be acknowledged for installing the measurement experiment to be reported on in section 4. The second clock improves the basis for measurements considerably; it must, however, be used with care as the interrupt handler for the additional timer consumes about 100 μsec of CPU time per interrupt - so, a theoretical limit (leaving no CPU time at all for actual program execution) is given by a resolution of 100 μsec; in practice, the frequency of the adjustable clock should be considerably lower - a frequency of 1000 Hz, i.e. a resolution of 1 msec, has often be employed in our measurements (using up about 10% of the available CPU capacity); measurement results can, of course, be adjusted to exclude the effects of the corresponding timing overhead.

In this report, a technique will be described which allows the measurement of small (execution) time intervals with slow clocks. The technique is based upon a repetitive execution of some regular execution pattern (in a tight loop), does, however, allow for simultaneous measurements of sub-activities within that execution pattern (within that loop). Given enough time for an experiment, sub-activities may be very short and the clock very slow; measurements of the indicated 50 μsec intervals with a 1...10 msec resolution clock typically have fairly reasonable execution times of a few minutes. The necessary number of loop cycles can be determined before the experiment, based on the approximate duration of the shortest sub-activity to be measured and upon the width of the confidence interval requested, at some selected confidence level.

2. Statistical Issues

Assume that we are interested in the duration, t(a), of a particular activity, a, and that we want t(a) to be measured. Also assume that t(a) is not constant but may vary due to certain environment influences, which assumption can be acknowledged by capturing the activity duration by a corresponding random variable, T(a). In the present context, "a" will often denote the execution of some particular piece of kernel code, such that we might be lead to argue t(a) were in fact constant. However, as masking and unmasking of interrupts at various levels constitute a mechanism inherently used by kernel code itself, environment influences (such as clock interrupts) cannot be strictly safeguarded against without impairing kernel dynamics, and thereby potentially invalidating any measurements. Moreover, if "a" also includes sub-activities on different, asynchronously operating hardware components (such as different CPUs and/or hosts, transmission media, etc.), acknowledging the variability of t(a) is obviously mandatory. Thus, measurement of t(a) will involve the collection of samples of t(a) and statistical estimation of distribution properties of T(a), from these samples. Although t(a) will not be constant, it may in the present context be justified to assume that its variation is not large, i.e., that T(a) exhibits a relatively small variance.

The standard set-up for corresponding experiments will then consist of

* embedding the activity, a, in a looping execution pattern;

* collecting the durations, $t_1(a), t_2(a),..., t_n(a)$, of the repetitively arising a-executions;

* statistically analysing the collected sample, ($t_i(a)$; i=1,2,...,n).

With the $T_i(a)$, i=1,2,...,n, considered identically distributed as some common random variable, T(a), with expected value, ET(a), we will for instance use the usual estimator, MT(a), for ET(a)

$$\text{(1a)} \quad MT(a) = \frac{1}{n} \sum_{i=1}^{n} T_i(a)$$

which is unbiased

$$\text{(1b)} \quad EMT(a) = ET(a)$$

and renders a sample estimate, mT(a), for ET(a) as

$$\text{(1c)} \quad mT(a) = \frac{1}{n} \sum_{i=1}^{n} t_i(a)$$

Other characteristics of the distribution of T(a) may obviously of interest, too.

Difficulties arise when the individual t_i's cannot be measured with reasonable accuracy, such as in the setting considered, where t_i's may range in the 50 µsec proximity whereas available discrete time clocks exhibit a resolution in the 1 ... 10 msec range. Fig. 2 depicts this situation.

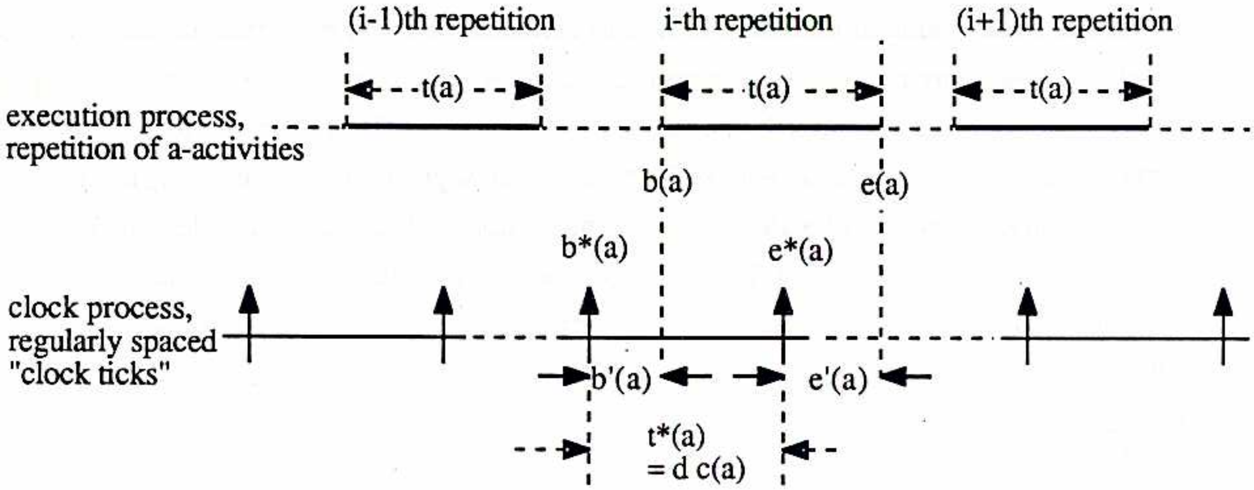


Figure 2: Execution and clock processes and their relationship

Following notation is used:

(3)	$t_i(a)$	(continuous time) duration of i'th repetition of a
	$b_i(a)$	(continuous time) begin of i'th repetition of a
	$e_i(a)$	(continuous time) end of i'th repetition of a
	$b_i{}^*(a)$	(discrete time) begin of i'th repetition of a, i.e. the clock time closest to, and prior to $b_i(a)$
	$e_i{}^*(a)$	(discrete time) end of i'th repetition of a, i.e. the clock time closest to, and prior to $e_i(a)$
	$b_i'(a)$	timing deviation at begin of i'th repetition of a, where $b_i^*(a) + b_i'(a) = b_i(a)$
	$e_i'(a)$	timing deviation at end of i'th repetition of a, where $e_i^*(a) + e_i'(a) = e_i(a)$
	$c_i(a)$	number of clock ticks during i'th repetition of a, rendering the discrete time duration of this repetition according to $t_i^*(a) = d \cdot c_i(a) = e_i^*(a) - b_i^*(a)$ (with a clock tick interval, d)

Please note that $c_i(a)=0$, i.e. $b_i^*(a)=e_i^*(a)$ is possible (and will in fact often occur) and also that, across the a-repetitions, $e_i^*=b_i^*=e_{i-1}^*=b_{i-1}^*= \dots$ is not unlikely to happen for small t(a)'s.

Our interest is in some sample of the t_i's where

$$t_i(a) = e_i(a) - b_i(a) \qquad i=1,2,...,n$$

With the T_i's (and, likewise, all other recurring time intervals) considered identically distributed, this relationship translates into its stochastic version

$$T(a) = E(a) - B(a)$$

from which we obtain with (3)

$$\begin{aligned} T(a) &= E^*(a) + E'(a) \; - \; (B^*(a) + B'(a)) \\ &= T^*(a) + E'(a) - B'(a) \\ \textbf{(4a)} \qquad &= d\cdot C(a) + E'(a) - B'(a) \end{aligned}$$

Now please convince yourself that (although being mutually dependent) E' and B' are identically distributed due to the assumed asynchronism of the clock and execution processes; in fact, both E' and B' are continuously uniformly distributed over (0,d). Hence, taking expectations on (4a) yields

(4b) $$ET(a) = d\cdot EC(a)$$

and the unbiased estimator, MC(a), for the expected number of clock ticks per a-activity

(5) $$MC(a) = \frac{1}{n}\sum_{1}^{n} C_i(a)$$

can be utilized to obtain, based on (1,4), an unbiased estimator MT(a), for ET(a)

(6a) $$MT(a) = \frac{d}{n}\sum_{1}^{n} C_i(a)$$

with the sample estimate, mT(a), based on a c_i-sample, given as

(6b) $$mT(a) = \frac{d}{n}\sum_{1}^{n} c_i(a)$$

Equ. (6) does not come as a surprise. A corresponding, more intuitive derivation could argue as follows: A total measurement interval, I, spanning c(I) clock ticks, has a duration of $d\cdot c(I)$. If, in this interval I, n a-activities are observed, with an assumed average duration of mT(a), then the portion

$$\frac{n \cdot mT(a)}{d \cdot c(I)}$$

of I will be "covered" by a-activities. Assuming a random distribution of a-activities across I, these activities should together collect the same portion of the total number of ticks "hitting" them, i.e.

$$c(\text{a-hits}) = c(I) \cdot \frac{n \cdot mT(a)}{d \cdot c(I)}$$
$$= \frac{n}{d} \cdot mT(a)$$

which coincides with (6b).

Compared with this intuitive derivation, equ.(6) provides a safer basis for assessing the sampling properties of MT(a); we shall in the following drop the explicit reference to activity, a, unless ambiguities arise. With respect to the sampling properties of MT we have:

(7a) as a repetition of the unbiasedness statement:

$$EMT = ET$$

(7b) using the usual limit theorem:
With n sufficiently large (n will in fact need to be fairly large)

MT normally distributed

(7c) as a warning factor:
With MT based upon the C_i's, and with the C_i's exhibiting considerable variance (think of ET small and d large), MT might possess a dangerously high variance. In fact, as well-known: If the C_i's could be considered mutually independent, an assumed variance, VC, of C would result in a variance, VMC, for MC corresponding to

$$VMC = VC/n$$

and hence, cf. (6), in a variance, VMT, for MT corresponding to

$$VMT = d^2 \cdot VC/n$$

We shall take these relationships for granted and return only later to the independence question.

What variance should we expect MC (and accordingly MT) to exhibit? Assume T had (as indicated before) in fact a relatively small variance, which we consider expressed through

(8) For

$$k \cdot d < ET < (k+1) \cdot d \qquad \text{k integer-valued}$$

the probabilities of an interval T being hit by less than k or more than k+1 clock ticks disappear (can be neglected), i.e.

$$P[< k \text{ clock ticks in } T] = P[> (k+1) \text{ clock ticks in } T] = 0$$

From (8) we obtain EC^2, the second moment of C

(9a) $$EC^2 = P[k \text{ hits}]\cdot k^2 + P[k+1 \text{ hits}]\cdot(k+1)^2$$

According to (8)

$$P[k \text{ hits}] + P[k+1 \text{ hits}] = 1$$

and, of course,

$$EC = P[k \text{ hits}]\cdot k + P[k+1 \text{ hits}]\cdot(k+1)$$

From the latter two equations

$$P[k \text{ hits}] = k + 1 - EC$$

$$P[k+1 \text{ hits}] = EC - k$$

easily derives; with (9a) we obtain the sought-for second moment of C

(9b) $$\begin{aligned} EC^2 &= (k+1-EC)\cdot k^2 + (EC-k)\cdot(k+1)^2 \\ &= (EC-k)\cdot(2k+1) + k^2 \end{aligned}$$

and the variance, VC, of C

(9c) $$\begin{aligned} VC &= EC^2 - E^2C \\ &= (EC-k) - (EC-k)^2 \end{aligned}$$

Now from (7c)

$$\begin{aligned} VMT &= d^2\cdot VC/n \\ &= d^2\cdot\{\,(EC-k) - (EC-k)^2\,\}/n \end{aligned}$$

with (4b)

(10a) $$VMT = d^2\cdot\{\,(ET/d-k) - (ET/d-k)^2\,\}/n$$

Due to assumption (8), the term ET/d -k varies between 0 and 1, the bracketed {...} expression taking a maximum at ET/d-k = 1/2, with a value 1/4, such that we arrive at the tight bound

(10b) $$VMT \le d^2/(4n)$$

(10b) is independent of ET, which is not surprising as assumption (8) prescribes exactly that; a simpler derivation of (10b) could start directly from (8), realizing that under this assumption the maximum variance of T is attained when 50% of the measurements indicate a duration of $k \cdot d$, the remaining ones of $(k+1) \cdot d$, yielding the value $d^2/4$ for the variance of T. (8) is not a particularly fair restriction for the variance of T as the implied relative variation, as expressed by the corresponding coefficient of variation, becomes more stringent with larger ET. For relatively small ET, however, (8) seems justifiable and both (10a) and (10b) useful for assessing the variance of MT as obtained from (6); in fact, the really interesting case is for ET considerably smaller than d.

(6) covers, as a borderline case, the situation of a contiguous sequence of a-activities (cf. Fig.2). (6b) is then identical to the simple measurement approach of executing some a-activity "in a tight loop", and of determining the duration of "a" (in fact, that of one loop cycle time) directly from clock readings just before and after the total loop (the difference between those clock readings obviously agrees with the total number of clock ticks during the loop). The assessment of the result's precision, however, differs for the two interpretations. For the simple measurement approach, it would be argued that a clock resolution, d, renders a total c-sum precise up to +/- 1, and the mT-value correspondingly precise up to +/- {d/n}. The latter is only justified if constant t(a)'s can be absolutely guaranteed. For a statistics based assessment of VMT for that case, equ's (10a,b) could in principle be applied. It is, however, to be expected that the confidence intervals obtained would turn out rather large, i.e. very pessimistic, as the special situation of the contiguity of the measured intervals (an additional information) is not reflected in the derivation of (10a). We have for this case (cf. fig. 2)

$$\sum_i T_i = d \cdot \sum_i C_i + E_n' - B_1'$$

Continued use of the estimator (6a) gives

$$\begin{aligned} MT &= \frac{d}{n} \cdot \sum_i C_i \\ &= \frac{1}{n} \cdot \left(\sum_i T_i - E_n' + B_1' \right) \end{aligned}$$

which is still unbiased (E_n and B_1 identically distributed):

$$EMT = (1/n) \cdot n \cdot ET = ET$$

The variance of MT is provided by (T's, E_n', B_1' considered independent):

(10c) $VMT = (1/n^2) \cdot (n \cdot VT + VE_n' + VB_1')$

with the U(0,d) distribution of both E_n' and B_1'

$$= VT/n + (d/n)^2/6$$
$$\leq \{ S\_DT/\mathrm{sqrt}(n) + d/(6n) \}^2$$

and the standard deviation of MT correspondingly

(10d) $$S\_DMT = \sqrt{\frac{1}{n} VT + \frac{d^2}{6 \cdot n^2}} \leq \frac{S\_DT}{\sqrt{n}} + \frac{d}{6n}$$

The basic variability of MT, hence, stems from the variability of the T's; the correction term d/(6n) for which the clock resolution is responsible decreases faster, namely (for the standard deviation) linearly with n. An a priori forecast of VMT (without involving VT), like (10a,b) for the non-contiguous case, would require additional assumptions about the variability of T. As a comparison of the variability interpretation of the simple measurement approach (mT precise to =/- {d/n}) with the expressions derived here: Arguing from (10d), a 99%-confidence interval of MT would be given as

$$mT \pm \{ 2.58 \cdot \mathrm{sqrt}(VT/n + (d/n)^2/6) \}$$

if we stick with the normality assumption for MT (which is questionable, as MT is given as the sum of a term which is approaching a normal distribution and two uniformly distributed terms; as an alternative to assuming normality, the more conservative Chebychev Inequality would have to be used). For this interval to have a width of {2·d/n) it is required that

$$2.58 \cdot \mathrm{sqrt}\{ VT/n + (d/n)^2/6 \} \leq d/n$$

i.e.
$$VT/n \leq (d/n)^2/2.58 - (d/n)^2/6$$
$$VT \leq (d^2/n) \cdot 0.02209$$

which obviously does not automatically hold for all possible VT-values.

We must return to the independence question, pushed aside when discussing (7c). If independence of the C's is not assumed, the variance of MC is, as well-known:

$$\text{(11)} \qquad \begin{aligned} VMC &= E[\,(MC - EMC)^2\,] \\ &= E[\,(\frac{1}{n}\sum_i C_i - EMC)^2] \\ &= E[\,\frac{1}{n^2}\cdot\{\,\sum_i (C_i\text{-}EC)^2 + \sum_i\sum_{j\neq i} (C_i\text{-}EC)\cdot(C_j\text{-}EC)\,\} \\ &= \frac{1}{n}\cdot VC + \frac{1}{n^2}\sum_i\sum_{j\neq i} E[\,(C_i\text{-}EC)\cdot(C_j\text{-}EC)\,] \end{aligned}$$

The covariance terms can be >0 or <0 and will disappear for independent C's, then rendering the VMC-expression of (7c). Without further assumptions, all three possibilities may become effective: Examples can be easily drawn up with a deterministic setting of the a-activities relative to the clock

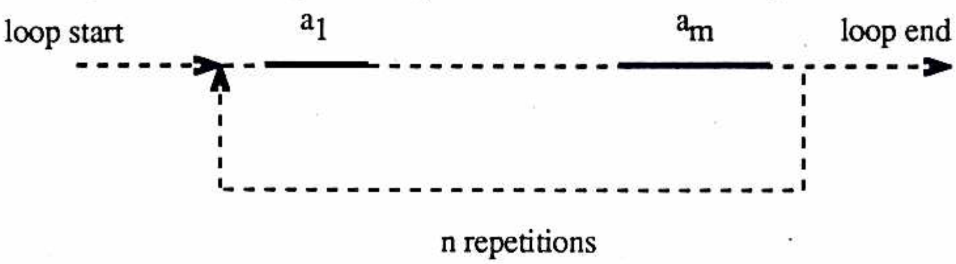


Figure 12: Several activities of a sequential execution pattern observed in one loop

process, in which case the execution and clock processes are involved in a repetitive pattern; the sum of the covariance terms may then turn out smaller or larger than 0 or may disappear depending on the sample size, n. Experience with using (10) has shown that point estimates for VMT, as obtained in actual experiments, turn out slightly smaller than predicted by (10), such that (10) may be regarded as somewhat pessimistic (and therefore: safe) variance predictors. A numerical example of these observations is provided in sect. 4.

3. Practical considerations

For purposes of measuring the execution time of some activity, "a", this activity will be included in a looping execution pattern. At the beginning and at the end of "a" an available clock will be read, the difference between these two clock readings determined and the consecutively obtained differences accumulated. The clock readings and, optionally, the updating of the cumulation counter have to be installed in the software to be measured, at the appropriate points in code. Upon leaving the loop, the total number of clock ticks is known that occurred while a-instances were active. The approach is advantageous in that

* it provides the possibility for executing "a" in a realistic execution environment,
* several different activities, $a_1, a_2, ..., a_m$, all included in the loop, can be assessed in one experiment, for which purpose cumulative clock tick counts must be collected for each activity; cf. fig. 12.

It is mandatory to assess two experiment parameters before actual experimentation, namely

* the clock interval, d, of the clock used; this may be a fixed value, or (for an adjustable clock) a genuine experiment parameter;
* the number of loop cycles, n.

The values of d and n will, together with the expected durations of the a_i, determine the precision of the measurements, as expressed by equ. (10).

The normality of every estimator $MT(a_i)$ allows a quantification of its confidence intervals, once the variance of MT is known. r%-confidence intervals have a total width of $w_r \cdot$S_DMT (again: S_DMT denotes the standard deviation of MT) with the pairs (r, w_r) obtainable from corresponding tables. A

required precision (r,p)	wr	required number of cycles, n, for d/ ET(amin) = 20	40	200
(90,0.1)	3.30	20700	42500	216700
(95,0.05)	3.94	29500	60500	308900
(99,0.01)	5.16	50600	103800	529800

Table 15: Minimal sample size (number of loop cycles)
for different precision requirement (r,p)
and different relations d/ET (clock resolution / activity duration)

few customarily applied pairs:

r	90	95	99
w_r	3.30	3.94	5.16

The usual requirement will consist of demanding an r%-confidence interval of a total width of no more than (100·p)% of the mean value of a, i.e.

(13) $$w_r \cdot S\_DMT(a) \leq p \cdot ET$$

With (10a):

$$(w_r \cdot d)^2 \cdot \{ (ET/d\text{-}k) - (ET/d\text{-}k)^2 \} / n \leq p^2 \cdot E^2T$$

and finally

(14a) $$n \geq (w_r/p)^2 \cdot \{1\text{-}k \cdot d/ET\} \cdot \{(1+k) \cdot d/ET - 1\}$$

The shortest activity measured, a_{min}, is likely to take an expected duration smaller than the clock resolution, i.e.

$$ET(a_{min}) < d$$

which results in a marginal form of (14a)

(14b) $$n \geq (w_r/p)^2 \cdot \{d/ET(a_{min}) - 1\}$$

We may want to see some actual numbers for a few interesting cases:

In table 15, the precision requirement (90,0.1) should be considered very moderate, whereas

(99,0.01) is very demanding; the values for r and p can, of course, be selected independently of each other. d/ET=20 corresponds, e.g., to a clock resolution of 1 msec and a measured time interval of 50 µsec, d/ET=200 corresponds to the same time interval measured with a 10 msec resolution clock.

At the other extreme, regard the longest activity measured, a_{max}, which is very likely the full loop cycle time. To arrive at an impression of the time needed for an experiment, consider the cases that $ET(a_{max}) = 50 \cdot ET(a_{min})$ and $ET(a_{max}) = 100 \cdot ET(a_{min})$. With $ET(a_{min}) = 50$ µsec, this would correspond to loop cycle times of 2.5 and 5 msec. The total experiments will then require the following execution times:

* low precision: (r,p) = (90,0.1)
  reasonable clock resolution: d = 1 msec
  $ET(a_{min}) \approx 50$ µsec, $ET(a_{max}) \approx 2.5$ msec
  experiment duration: 52 sec

* reasonable precision: (r,p) = (95,0.05)
  reasonable clock resolution: d = 1 msec
  $ET(a_{min}) \approx 50$ µsec, $ET(a_{max}) \approx 2.5$ msec (5 msec)
  experiment duration: 74 sec (145 sec)

* high precision: (r,p) = (99,0.01)
  low clock resolution: d = 10 msec
  $ET(a_{min}) \approx 50$ µsec, $ET(a_{max}) \approx 5$ msec
  experiment duration: 2650 sec

Although the extreme cases are obviously extreme in experiment duration too, they are not altogether infeasible; moderate cases, on the other hand, have very reasonable total experiment times.

Yet some additional illustrative numbers: With the necessary number of loop cycles determined by the presision requirements for minimum duration activities, the longer activities will automatically end up with higher precisions. Assume a reasonable precision of (r,p) = (95,0.05) required for $ET(a_{min})$. For $ET(a_{max})/ET(a_{min}) = 50$ and $d/ET(a_{min}) = 20$ we obtain from (14a) and table 15 the corresponding

$w_r \leq \mathrm{sqrt}(184000) \approx 430$

i.e., for p=0.05 a $w_r \leq 21.5$ and consequently $r > 99$

for p=0.01 a $w_r \leq 4.3$ and consequently $95 < r < 99$

as higher confidence levels for MT(a_{max}).

Concluding this section, I should like to point out that publications on measurements of kernel code (the area of central interest here) do not always pay sufficient attention to the statistical interpretation of experiment results. Not always are samples of the size reported in table 15 considered necessary, and not always are experimenters aware of the consequently restricted statistical precision of their results.

Let me quote from one particular example: Durations of executions of certain parts of kernel code were measured with a clock resolution of (d=) 10 msec. The simple measurement approach was used of running every activity measured"in a tight loop", i.e. loop cycle times were to be assessed. The loop was ("typically"?) executed (n=) 1000 times. The author uses the obvious formula (6b) for estimating the mean cycle times of the various loops; with respect to precision, total experiment durations are reported precise up to +/- 10 msec and individual loop cycle times precise up to +/- 10 µsec. The latter, as we observed, can only be accepted if loop cycle times can be guaranteed absolute constancy. If not, as is very likely, the estimated means have a variance as given by equ. (10d). A 99%-confidence interval of +/- {10µsec}is therefore only correct if a T-variance smaller than $0.02209 \cdot d^2/n$ can be guaranteed, i.e. a standard deviation

S_DT ≤ 0.0479 msec

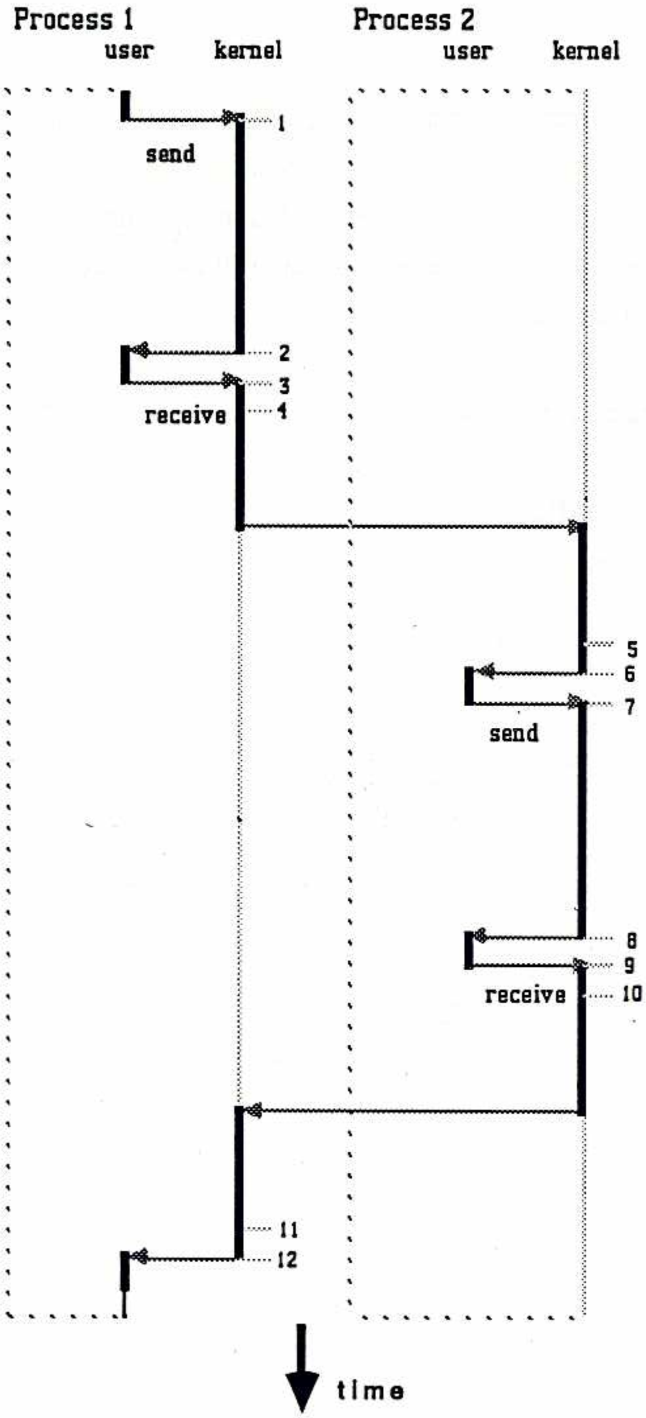
which may or may not be the case.

4. First Applications

The proposed measurement technique was first applied for quantifying the time needed in passing messages via the DASH kernel. More precisely: Two user processes were set up passing a minimum size ("null") message repeatedly back and forth. The two processes resided in separate user spaces, on a single host (a SUN 3-50). The processes consisted of executing the following two programs (specified here without any reference to actual syntax):

Program / Process # 1:	Program / Process # 2:
DO n TIMES	DO >n TIMES
send; receive;	receive; send;
OD;	OD
display results	

The applied message-passing operations are implemented in DASH such that they trap into the kernel and return to user space upon completion. The receive operation will potentially block until a corresponding message is available. The above processes are (after an initial round which is not included in the above program specifications) engaged in a synchronized, repetitive pattern as depicted in fig. 16. Four measurement points were implanted in the DASH kernel code: One just after trapping into the kernel, one just before returning to user space, one just after recognizing the necessity of a context switch and one just before completing a context switch. These measurement points appear (in different roles) as points 1 through 12 in fig. 16.



Measurement points 1,3,7,9:
Trapping into the kernel

Measurement points 2,6,8,12:
Returning to user space

Measurement points 4,10:
Starting a context switch

Measurement points 5,11:
Completing a context switch

Depicted time intervals are only roughly proportional to measured durations; indicated times of actual switching within context switch activity arbitrarily selected.

Figure 16: Timing diagram for the synchronized behaviour pattern of processes #1 and #2

All occurring time intervals between time-adjacent measurement points were to be assessed, i.e. times (1,2), (2,3), ... , (11,12), (12,1); additionally, as a minimal consistency check, the total loop cycle time (1,1).

The initial timing guess was for a minimum of these intervals at $ET(a_{min}) \approx 100\,\mu sec$. With the intention of obtaining noticeable result variations across repetitions of the sketched scheme, and consulting table 15, the adjustable clock was set to d = 1 msec and the number of loop cycles to n = 10000, well below the 20700 required for a (90,0.1)-precision. The whole scheme was executed 10 times (i.e. 10 experiments with 10000 loop cycles each were conducted).

Table 17a supplies the resulting clock tick counts, c(i,j), for all (i,j) and for all 10 experiment repetitions. Also supplied are the sums of clock ticks over all 10 experiments for all (i,j), which are taken as an equivalent of a single experiment with a loop cycle number of 100000 (moving us well above the extreme (99,0.01)-precision - see table 15). Table 17b supplies point estimates for the ET(i,j), on the basis of the high precision column of table 17a, following equ.(6b). These times should not be taken as an indication of effective DASH performance: The experiments were conducted less than one week after the measured operations became operational; another week later, the round trip (i.e. cycle) time was measured at less than 50% of that reported in table 17, due to initial tuning work. Eventual DASH performance figures will be supplied in a separate paper. Table 17b also supplies the standard deviations, S_DMT, of all interval means (for the original n=10000, "small", samples), obtained as

S_DMT(i,j) = sqrt (VMT(i,j))

where VMT(i,j) is calculated from equ.(10a) with the high precision mMT(i,j) substituted for ET(i,j) as a good approximation. Equ. (10b) supplies an upper bound of 5 μsec, for all intervals. The last column of table 17b provides point estimates for the standard deviations, s_dMT, of the low precision means, calculated on the basis of the samples of table 17a (size: 10) following the standard formula

$$\text{s\_dMT} = \sqrt{\frac{\sum_i (\text{mMT}_i)^2 - \frac{1}{10}\left(\sum_i \text{mMT}_i\right)^2}{9}}$$

The tables verify equ's (10a,10b) in the sense that these equations indeed provide safe bounds for the variability of the measurement results, in the conducted experiment.

	d = 1000			n = 10000							
repetition #	1	2	3	4	5	6	7	8	9	10	sums
measured:	# of clock ticks										
c(1,1)	56913	56855	56856	56853	56854	56854	56854	56854	56853	56856	568602
c(1,2)	11931	11949	11931	11899	11912	11906	11940	11902	11963	11935	119268
c(2,3)	820	806	817	829	849	821	821	882	832	811	8288
c(3,4)	1858	1846	1847	1843	1845	1882	1823	1816	1832	1846	18438
c(4,5)	11995	12023	12033	12013	11953	11981	12028	12014	11972	12029	120041
c(5,6)	894	872	859	864	909	871	852	832	888	847	8688
c(6,7)	1425	1403	1404	1455	1436	1399	1459	1488	1461	1428	14358
c(7,8)	11918	11918	11924	11875	11870	11959	11867	11882	11860	11902	118975
c(8,9)	844	867	873	905	892	830	892	865	884	898	8750
c(9,10)	1819	1823	1801	1757	1806	1806	1806	1775	1826	1774	17993
c(10,11)	9629	9585	9597	9634	9606	9604	9606	9636	9587	9628	96112
c(11,12)	855	859	830	843	835	846	865	859	852	839	8483
c(12,1)	2925	2904	2940	2936	2941	2949	2895	2903	2896	2919	29208

Table 17a: Clock tick measurement results for various activities

	high precision mMT(i,j) µsec	low precision S_DMT(i,j) µsec	low precision s_dMT(i,j) µsec
(1,1)	5686	**4.64**	**1.86**
(1,2)	1193	**3.94**	**2.14**
(2,3)	83	**2.76**	**2.22**
(3,4)	184	**3.88**	**1.83**
(4,5)	1200	**4.00**	**2.75**
(5,6)	87	**2.82**	**2.33**
(6,7)	144	**3.51**	**2.96**
(7,8)	1190	**3.92**	**3.19**
(8,9)	88	**2.83**	**2.41**
(9,10)	180	**3.84**	**2.31**
(10,11)	961	**1.93**	**1.92**
(11,12)	85	**2.79**	**1.15**
(12,1)	292	**4.55**	**2.03**

Table 17b: High precision point estimates for mean interval durations,
low precision calculated standard deviations of means,
low precision estimatated standard deviations of means

5. Conclusion

A measurement technique has been described that allows the estimation of means of operation durations and of means of operation-embedded sub-activities even though an available discrete-time clock might exhibit an inter-tick interval larger than (possibly much larger than) the time intervals to be measured. Our first experience with using the technique has been sketched and has shown to be fairly positive.

There is certainly room for further work that would improve the technique:

* Comparisons of calculated and estimated variances / standard deviations (see table 17) should be conducted on larger samples (in the sense of a larger number of experiment repetitions) in order to hopefully further confirm the usability of the variance predictors, equ's (10a,b).
* The question of independence/dependence of the c-samples, cf. equ's (7c,11), should be studied more thoroughly in order to obtain
  - either: the conditions under which the c-sample is independent or at least its components pairwise non-correlated,
  - or: a provable expression for the covariance term of equ. (11), or a bound for it, to be used as an adjustment term in equ's (10a,b).
* For the special case of a contiguous series of intervals to be measured (the loop cycle duration case), both intuition and experience to-date seem to indicate that the variance predictor (10a) is too large and that a smaller variance could possibly be determined. This suggestion should be studied further and would, if correct, result in the possibility of conducting smaller experiments (i.e., with smaller "n"), if only the loop cycle time (and not any embedded time intervals) are of interest.

As in all experiments so-far the variance predictors (10a,b) have turned out slightly pessimistic (i.e.: safe!), the technique can be supplied with sufficient confidence.

References

ANFE88 Anderson,D.P./Ferrari, D.;
The DASH Project: An Overview;
Technical Report No. 88/405, UC Berkeley, Computer Science Dpt., 1988

FIRA86 Fitzgerald,R./Rashid,R.F.;
The Integration of Virtual Memory Management and Interprocess Communication in Accent; ACM TOCS, vol.4 (1986), no.2, pp.147-177

SCCO87 Scott,M.L./Cox,A.L.;
An Empirical Study of Message-Passing Overhead;
Proc. 7th ICDCS, Berlin, 1987