Sampled Ready Queue Processing Time Estimation Using Size Measure Information In Multiprocessor Environment

Sarla More and Diwakar Shukla

•

Dr. Harisingh Gour University, Sagar, MP, India
sarlamore@gmail.com, diwakarshukla@rediffmail.com

Abstract

In a multiprocessor computer system, there exist a ready queue of large number of processes waiting for computing resources allocation by the processors. These jobs may have size measure, which are additional information priory known while entry to the ready queue. Suppose the sudden system breakdown occurs and recovery management is required immediately. At this stage, one can find some processes who are completely finished, some partially processed, some blocked by processors and remaining waiting for allocation in the ready queue. Prime act of a system manager is to evaluate the maximum time required to process all the remaining jobs. This paper presents an estimation strategy for such, derived by applying the lottery scheduling, sampling technique and imputation methodology. Expressions for mean squared error of the proposed strategy are derived and optimized for suitable selection of system parameters. Three cases are discussed and compared and consequent results are numerically supported. It is found that at the optimal choice of constants in the estimation methodology, the shortest confidence interval can be predicted estimating the remaining required time. Such findings are useful as a part of disaster management of a cloud based multiprocessor data centre.

Keywords: Ready Queue, Lottery scheduling, Multiprocessors, Simulation, Sampling, Random, Estimation

I. Introduction

Assume a computer system equipped with several processors having a ready queue, where processes are waiting for allocation of resources. Lottery scheduling is a type of priority scheduling where the computer system resources are allocated randomly to the waiting processes. In this, a bunch of token numbers are assigned to processes and multi-processors used to issue random numbers. A process who contains the token of issued number receives first the desired system resource. Using this, every process has chance of allocation of resources, sooner or later, therefore, probability of starvation vanishes. A process in a ready queue may have a predetermined measure of its size in terms of bytes which is additional information, can be used for better prediction of the remaining mean time of ready queue processing. This paper presents a strategy for effective use of known size measure of processes.

Let $t_1, t_2, t_3 \ldots\ldots t_k$ be the time of k processes and $x_1, x_2, x_3 \ldots\ldots x_k$ be their size measure. Figure 1 shows r processors ($r < k$) and waiting queue.

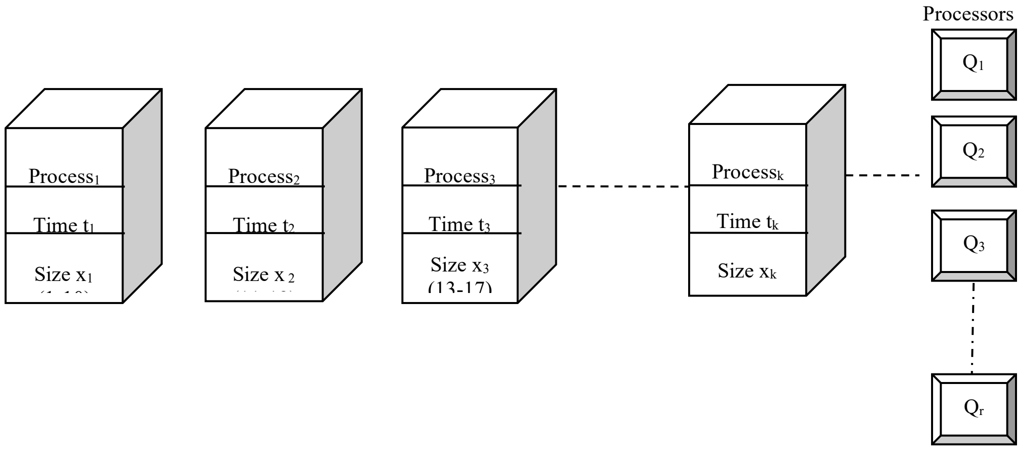


Figure 1: *Ready queue with waiting Processes and Multiprocessor*

Further assume processes A_1, A_2, A_3…..........are of small size with time consumption t_{11}, t_{12}, t_{13}……....... and size measure x_{11}, x_{12}, x_{13}…... (see figure 2). The large size processes areB_1, B_2, B_3…....... with time consumption t_{21}, t_{22}, t_{23}..…...... and size measure x_{21}, x_{22}, x_{23}…......…(see figure 3). All the A_i and B_i are to be processed by r processors, under the size measures. The case of partially processed and completely processed [23] exists when sudden breakdown occurs. One can further think of possibilities as under:

(a) inside multi-processors, some are completely processed,
(b) some are partially processed,
(c) some are blocked, and
(d) size measure of processes are known.

This paper extends approach of [23] in collective presence of (a), (b), (c) & (d) when multiprocessor computer system fails at an instant. The issue of estimation (prediction) of recovery time duration of the remaining is focused.

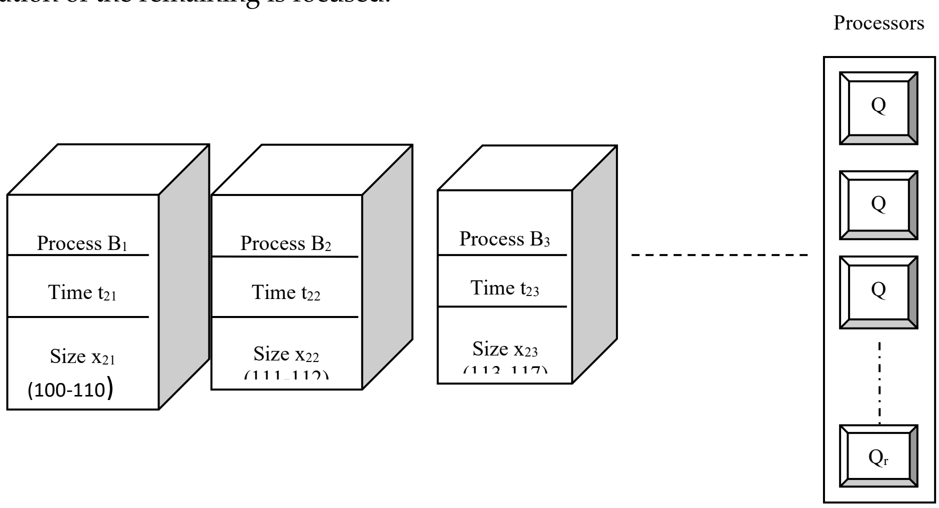


Figure 2: *Small size processes and Multiprocessors*

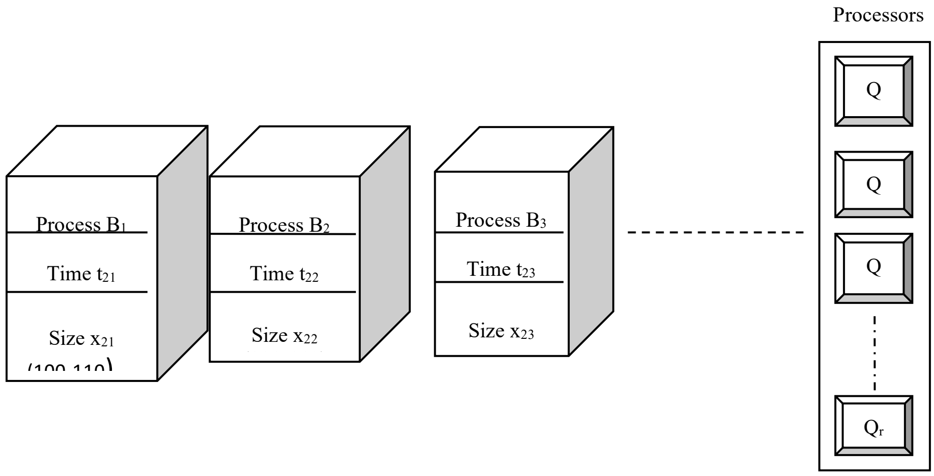


Figure 3: *Big size processes and Multiprocessors*

II. A Review

Lottery scheduling is a resource sharing technique [12] like a particular case of priority scheduling where the processes in ready queue are allotted bunches of ticket numbers. Process who receives maximum count of tickets has highest priority of being allocated the demanded system resources. Lottery scheduling is efficient and effective [22] in the framework of LINUX kernel also. It could be used as a model tool [4], [6] for estimating the mean time of processing of a ready queue where large number of jobs are in waiting but only some have processed. Completed jobs could be used as a sample just like a preliminary source of information for prediction. Concept of grouping of homogeneous jobs together [5] came into existence which has improved the prediction. Units in sample may have additional correlated variables which could be utilized for efficient computing ([7], [8], [9], [10], [11]) along with precise prediction. An exhaustive review [3] on the similar problem contributes few recent aspects of solutions extendable into [1] and [2]. Some authors have extended the lottery scheduling variants [20], [21] in the form of hybrid multi-level structure using Markov chain model along with analysis and chance based prediction.

Sampling techniques are useful tools for parameter estimation and value prediction. Random sampling schemes exist in statistical literature ([13], [15], [16]) who are widely used for parameter evaluation of a finite collection. Some popular schemes are like stratified sampling, cluster sampling, two stage sampling, systematic sampling, successive sampling etc.([14], [17], [18]) useful in varying situations of the aggregate . Moreover, such needs appropriate selection of methods also [19] to provide accurate confidence interval for unknown parameter.

Imputation is a methodology used when one or more values in a sample are found missing (or non-responded). For example, if a processor blocks a process then mean time parameter remains unpredicted using sample from the ready-queue. However, some processes may be blocked after the partial processing. For completely blocked processes, Random Imputation methods ([25], [26], [27]) could be used to recover information. In this, the missing values are selected randomly from the available part of sample values and replaced. Other popular imputation methods are mean imputation, deductive imputation, mean imputation within classes, deductive imputation within class, hot deck imputation, cold deck imputation etc. ([28], [29], [30], [31]). This paper considers the approach of [6] and [23] and extends using [10], [11], assuming situation when one process is blocked, one is partially processed and remaining others in a processor are completed before the occurrence of sudden breakdown.

I. Remaining Time Estimation Problem

Assume a large number of processes (say N) present in a ready queue of a multiprocessor computer system and only few of them (say n, n<N) have been processed before a fixed time instant. The remaining in the ready queue are (N-n) for whom the expected time computation is required. If sample mean time of those who already processed is Δ then remaining time estimate is $\delta = [(N-n)\ \Delta]$ which is an unknown quantity. For any two real numbers 'a' and 'b', if Δ is predicted as $\Delta \in (a,b)$ where (a,b) is an interval containing Δ with very high probability, then $\delta_1 = [(N-n)\ a]$is lowest, $\delta_2 = [(N-n)\ b]$ is the value of highest expected time. If highest expected time is precisely estimated then it could be used for backup management during system failure. The efficient estimation of this expected range is a problem which is undertaken in this paper for strategy formation in the multiprocessor setup with the consideration of multiple real life possibilities.

II. **Confidence Interval (CI)**

It is a statistical tool for evaluating the precision of mean time estimate. If catches the true unknown value then it is termed as a confidence interval. Let P[A] denotes the probability of happening of event A. In statistical theory, for any two real numbers a', b', the 95% confidence interval is defined as P[a' < true unknown value < b'] = 0.95**.** Define length of CI= l = (b'-a'). Let one confidence interval has length $l_1$obtained through a methodand other has length $l_2$obtained by another method. If $l_2 < l_1$then second one is said to be better than the first in terms of efficient prediction.

III. **Motivation**

Earlier contributions (specially [6], [23]) were under assumption that processes present in a multiprocessors system are completely processed before sudden failure. But this is not a practical reality. While sudden failure, some jobs may complete, some may partially processed and some may blocked by the processors [see figure 4]. The processed and unprocessed case was considered in [23] [see figure (5)]. This paper extends the approach of [23] by applying the tools of random imputation method against the blocked processes.

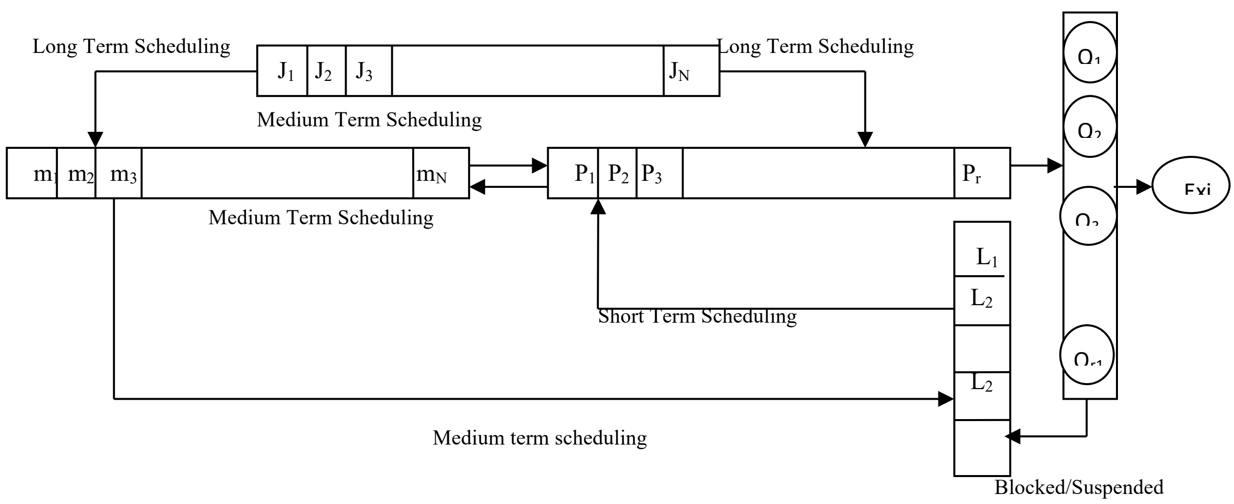


Figure 4: *Ready Queue Processing under Lottery Scheduling (due to [6])*

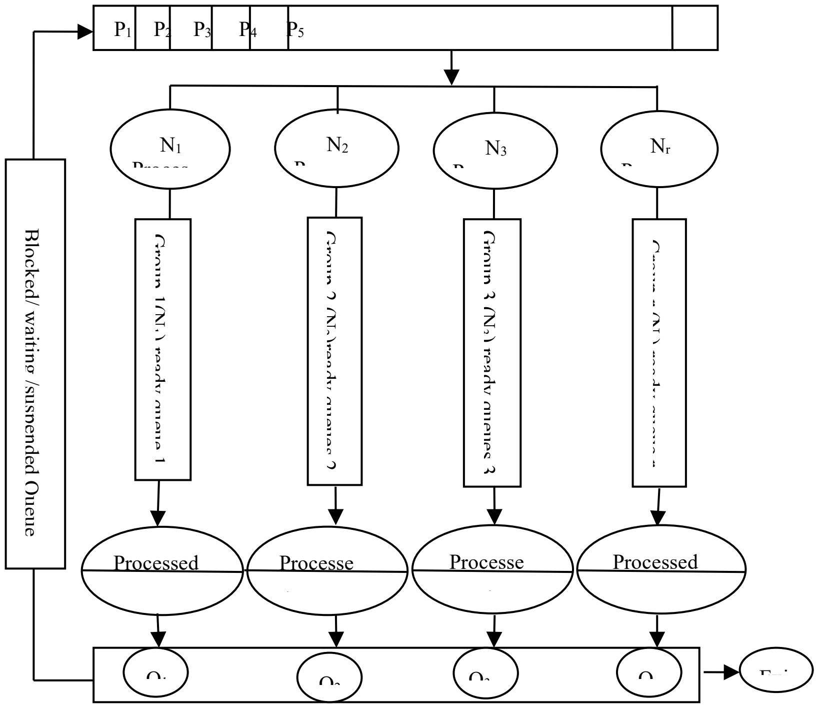


Figure 5: *Setup of ready queue and multiprocessor environment (due to[23])*

III. Proposed Computational Setup

Assume the existence a virtual sampled ready queue in a system of multiprocessors environment. Some jobs are randomly selected using lottery scheduling from the ready queue and placed in the sampled ready queue from top to bottom in the sequential manner of their selection. Processors are assigned processes in the ordered manner from top to bottom ofthe virtual sampled ready queue. Figure 6 shows basic setup of this approach without size measure while figure 5 shows the earlier approaches [4], [5], [6], [23]. Moreover, figure 7 reveals the special case when all sample units processed before the occurrence of breakdown.

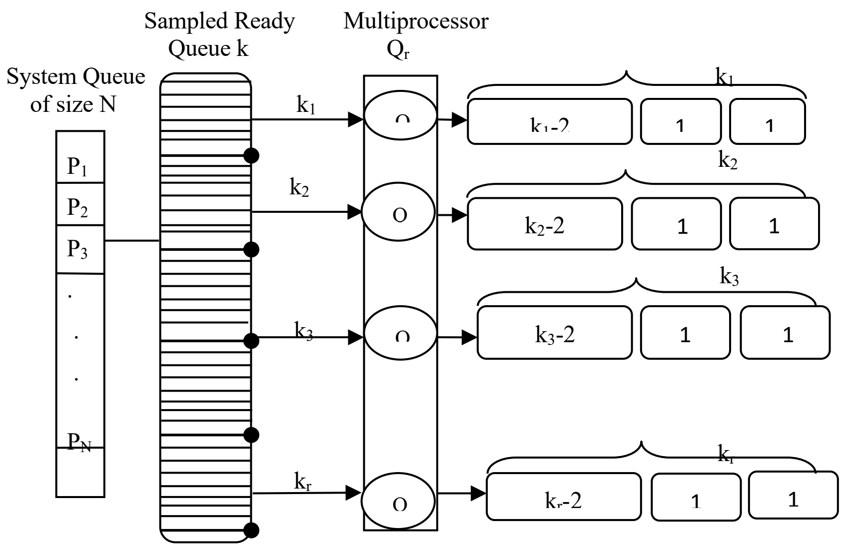


Figure 6: *Sampled Ready Queue Processing Time Estimation setup without size measure*

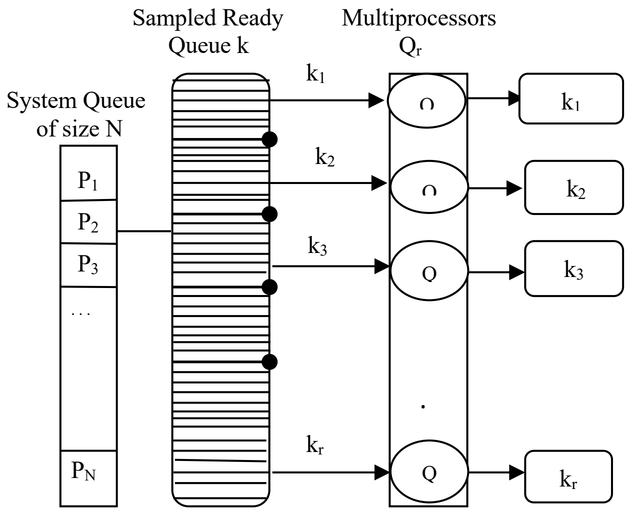


Figure 7: *Sampled Ready Queue Processing Time Estimation when all processed before breakdown*

I. Assumption and Model

In view of figure 6, let the selection of processes is as per priority scheduling, in particular, as per lottery scheduling. The process who selects first is placed at the top of the virtual queue who is segment or group of processes likely to allocate to the multi-processors.

1). Assume r processors in system and a ready queue of N processes denoted as [P_1, P_2, P_3.........P_N]who are waiting for allocation of resources.

2). The selection of process for resource allocation is on priority basis using lottery scheduling.

3). If all N are processed completely then time consumed by them are [t_1, t_2, t_3,.......t_N] and each process has size measure [x_1, x_2, x_3,.......x_N] who are priory known.

4). Define the whole ready queue mean time$\bar{t}$= $\frac{1}{N}\sum_{i=1}^{N} t_i$, size measure mean$\bar{x}$= $\frac{1}{N}\sum_{i=1}^{N} x_i$ and respective mean squares $S_t^2 = \frac{1}{N-1}\sum_{i=1}^{N}(t_i - \bar{t})^2$, $S_x^2 = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \bar{x})^2$

The process P_i of known size X_i consumes time t_i(i = 1,2,3,......N).

5). Hereby denote r multiprocessors as Q_1, Q_2, Q_3.....Q_r, (r < N) and time consumed by the i[th] process in the j[th] processor is t_{ij} with corresponding size measures x_{ij} (j = 1,2,3,......r)

6). Total completion time of ready queue is $N\bar{t}$, which is an unknown quantity. This paper is focused to estimate such using sampling methodology. Lottery scheduling is a tool for such estimation where process P_i has a bunch of token numbers and Q_j generates a random number. A process who receives the random number gets the desired resource from Q_j.

7). A virtual ready queue of size k (k < N, k>3r) exists to store sequentially the records of randomly selected k processes from N. The j[th] segment of virtual sampled queue is k_j(k $=\sum_{j=1}^{r} k_j$), who is allocated to the j[th] processor Q_j in sequential manner.

8). In sample let sx_{jl} denotes the file size measure and st_{jl} denotes time consumed by i[th] process in Q_j(l = 1,2,3,...k_j) when all processed completely who are included in the sample of size k.

(i). Sample mean of time $\overline{st}$= $\frac{1}{k}\sum_{j=1}^{r}\sum_{l=1}^{k_j} st_{jl}$

(ii). Sample mean square of time, $(es)_t^2 = \frac{1}{k-1}\sum_{j=1}^{r}\sum_{l=1}^{k_j}(st_{jl} - \overline{st})^2$

(iii). Sample mean of size, ($\overline{sx}$)= $\frac{1}{K-1}\sum_{j=1}^{k_j}\sum_{l=1}^{k_j}(sx_{jl})$

(iv). Sample mean square of size, $(es)_x^2 = \frac{1}{k-1}\sum_{j=1}^{r}\sum_{l=1}^{k_j}(sx_{jl} - \overline{sx})^2$

The term$\overline{st}$, $\overline{sx}$, $(es)_t^2$, $(es)_x^2$hold when system runs without failure.

9). Assume system breakdown occurs at the time instant T and there are (k_j - 2) processes who are finished in Q_j, but one remain partially processed and one remain unprocessed (blocked). This is an assumed model sh own in fig. 6 and fig.8.

10). Let $(st')_{jl}$ is time consumed by the l^{th} process in the processor Q_j [l =1, 2, 3... (k_j - 2)],who is among those processed completely before the occurrence of T.

11). Some sample mean related measures are:
   (i). Sample mean of (k_j - 2) process, $(\bar{st}')_j = \frac{1}{(k_j-2)}\sum_{l=1}^{(k_j-2)}(st'_{jl})$
   (ii). Sample mean square, $(es')_j^{\,2} = \frac{1}{(k_j-3)}\sum_{l=1}^{k_j-2}(st'_{jl} - (\bar{st}')_j)^2$
   (iii). Similar is for size measure also as (sx'_{jl}) represents size of l^{th} process who is in Q_j before T.
   (iv). Sample mean, $(\bar{sx}')_j = \frac{1}{(k_j-2)}\sum_{l=1}^{k_j-2}(sx'_{jl})$
   (v). $(\bar{sx})_j = \frac{1}{(k_j)}\sum_{l=1}^{k_j}(sx'_{jl})$ is sample mean of all k_j known values related to x in j^{th} segment of ready queue.
   (vi). Sample mean square, $(ex')_j^{\,2} = \frac{1}{(k_j-3)}\sum_{l=1}^{k_j-2}\left(sx'_{jl} - (\bar{sx}')_j\right)^2$
   (vii). Sample Covariance, $(es'x')_j = \frac{1}{(k_j-3)}\sum_{l=1}^{k_j-2}\left(st'_{jl} - (\bar{st}')_j\right)\left(sx'_{jl} - (\bar{sx}')_j\right)$

12). Assume t_m^* is partially processed time of a process in Q_j (j = m =1,2,3....r) whose sample mean under T is
$(\bar{t}^*/T) = \frac{1}{r}\sum_{m=1}^{r} t_m^*$
Variance $(\bar{t}^*/T) = V(\bar{t}^*/T) = \left(\frac{1}{r} - \frac{1}{N-k+r}\right) S_T^2$, where S_T^2 is the conditional ready queue mean square of the remaining unsampled part [N-K+r] expressed as:
$S_T^2 = \frac{1}{(N-k+r-1)}\sum_{i=i}^{N-k+r-1}(t_i - \bar{t}_T)^2$ where

$$\bar{t}_T = \frac{1}{N-k+r}\sum_{i=1}^{N-k+r}(t_i)$$

Herein to mention that S_T^2 and $\bar{t}_T$ contain time t_i only from non-sampled processes (N-k) of the main ready queue with the addition of those r who partially processed. For such, the size converts from N into (N-k+r) and only those processes are the part of $\bar{t}_T$ and S_T^2 who are in (N-K+r).

13). The r blocked processes are imputed by Random Imputation Method using random selection of a process among (k_j-2) relating to Q_j. Let for Q_j this imputed time is denoted as t_m^{**}.
   (i). Sample mean of imputed time, $\bar{t}^{**} = \frac{1}{r}\sum_{m=1}^{r} t_m^{**}$
   (ii). Variance of imputation under T, $V(\bar{t}^{**}/T) = \left(\frac{1}{r} - \frac{1}{k}\right)(es)^2$, $r < k$.

14). Sample based estimate of $(es)^2$ can be obtained by using all k values of time consumption in sample including the partially processed time t_m^* and imputed time value t_m^{**}. It is denoted as $(es^*)^2$ and mathematically expressed as
$(es^*)^2 = \frac{1}{k-1}\sum_{j=1}^{r}\sum_{l=1}^{k_j}(st^*_{jl} - \overline{st}^*)^2$ where (st^*_{jl}) and $\overline{st}^*$ include completely processed time st^*_{ij}, partially processed t_m^* and imputed t_m^{**}.

15). The sample estimate of S_T^2 is $(es')^2 = \frac{1}{r-1}\left[\sum_{m=1}^{r}(t_m^* - \bar{t}^*)^2\right]$

16). Bias of estimation strategy is assumed negligible wherever appears and applicable in mathematical expressions.

IV. Computational Set-up

The objective is to compute the remaining ready queue processing time while occurrence of sudden failure of system at time instant T. This is subject to condition that r processes are partially processed, r are unprocessed (blocked) and remaining (K-2r) are fully completed. Blocked and partially processed are one each from every Q_j and the available size measures are the part of computation. Some frequently used symbols for process time t and process size measure X are as under:

$\bar{t} = \frac{1}{N}\sum_{i=1}^{N} t_i = \frac{1}{N}\sum\sum t_{ij}$ …(4.1)

$\bar{t}^* = \frac{1}{r}\sum_{m=1}^{r} t_m^*$ …(4.2)

$\bar{t}^{**} = \frac{1}{r}\sum_{m=1}^{r-1} t_j^{**}$ …(4.3)

$(\overline{st}')_j = \frac{1}{(k_j-2)}\sum_{j=1}^{k_j-2}(st'_{jl})$ …(4.4)

$(\overline{sx}')_j = \frac{1}{(k_j-2)}\sum_{j=1}^{k_j-2}(sx'_{jl})$ …(4.5)

$(\overline{sx})_j = \frac{1}{(k_j)}\sum_{l=1}^{k_j}(sx'_{jl})$ …(4.6)

$(es')_j^2 = 1/(k_j-3)\sum_{l=1}^{k_j-2}(st'_{jl} - (\overline{st}')_j)^2$ …(4.7)

$(ex')_j^2 = 1/(k_j-3)\sum_{l=1}^{k_j-2}(sx'_{jl} - (\overline{sx}')_j)^2$ …(4.8)

$(es'x')_j = \frac{1}{(k_j-3)}\sum_{l=1}^{k_j-2}\left(st'_{jl} - (\overline{st}')_j\right)\left(sx'_{jl} - (\overline{sx}')\right)_j$ …(4.9)

$(es^*)^2 = \frac{1}{k-1}\sum_{j=1}^{r}\sum_{l=1}^{k_j}(st^*_{jl} - \overline{st}^*)^2$ …(4.10)

$R_{N_j} = \left[\frac{(\overline{st}')_j}{(\overline{sx}')_j}\right]$ …(4.11)

$\bar{t}_{rj} = [(\overline{st}')_j\left(\frac{(\overline{sx})_j}{(\overline{sx}')_j}\right)^{\alpha_j}]$, α_j being constant, $(0<\alpha_j<\infty)$ …(4.12)

I. Estimation Strategy

The sample based proposed estimation strategy for mean time is:

$(t_{mean}/T) = €_1 [\sum_{j=1}^{r} w_j\ (\bar{t}_{rj}/T)] + €_2\ (\bar{t}^*/T) + (1- €_1 - €_2)\ (\bar{t}^{**}/T)$

with condition that $\sum_{p=1}^{3} €_p = 1$ and $€_p$ denotes constants to be determine suitability and $w_j = (k_j/k)$ is known weight ($\sum w_j = 1$).

With the help of Cochran [16; see page 166, page 27, 29] for t_{mean}, the expected value E[.] is expressed as:

$E[t_{mean}/T] = E[€_1 [\sum_{j=1}^{r} w_j (\bar{t}_{rj}/T)] + €_2 (\bar{t}^*/T) + (1- €_1 - €_2)(\bar{t}^{**}/T)]$

$= €_1 [\sum_{j=1}^{r} W_j E\ (\bar{t}_{rj}/T)] + €_2 E(\bar{t}^*/T) + (1- €_1 - €_2) E(\bar{t}^{**}/T)]$

$\neq \bar{t}$ which shows estimator (t_{mean}/T) is biased.

II. Mean Squared Error

Let MSE (.), V (.) and B (.) denote mean squared error, variance and bias respectively. One can express

MSE (t_{mean}/T) = Variance (t_{mean}/T) + [Bias $(t_{mean}/T)]^2$ which holds in general.

Assume the bias is small, therefore negligible (as in assumption no. 16)

MSE (t_{mean}/T) = Variance(t_{mean}/T)

$= €_1^2[\sum_{j=1}^{r} w_j^2\ MSE(\bar{t}_{rj}/T)] + €_2^2 V(\bar{t}^*/T) + (1- €_1 - €_2) V(\bar{t}^{**}/T)]$

$= €_1^2 [\sum_{j=1}^{r}\left(\frac{1}{(k_j-2)} - \frac{1}{k}\right) w_j^2 \{(es')_j^2 + \alpha_j^2 R_{N_j}^2 (ex')_j^2 - 2\alpha_j R_{N_j}(es'x')_j\}] + €_2^2 [\left(\frac{1}{r} - \frac{1}{N-k+r}\right) s_T^2]$

$+ (1- €_1 - €_2)^2 \sum_{j=1}^{r}\left(1 - \frac{1}{K_j-2}\right) w_j (es')_j^2$ (as per Cochran[16] page24, page29 and page164)

The expressions P, Q, R are in the sample based estimate form of population parameters

Let $P = \sum_{j=1}^{r} \left(\frac{1}{(k_j-2)} - \frac{1}{k}\right) w_j^2 \{ (es')_j^2 + \alpha_j^2 R^2_{Nj} (ex')_j^2 - 2\alpha_j R_{Nj} (es'x')_j \}$

$Q = \left(\frac{1}{r} - \frac{1}{N-k+r}\right) s_T^2$

$R = \sum_{j=1}^{r} \left(1 - \frac{1}{K_j-2}\right) w_j^2 (es')_j^2$

The above expression is re-written as:

$V[t_{mean/T}] = [€_1^2 P + €_2^2 Q + (1 - €_1 - €_2)^2 R]$ ignoring the covariance terms due to independency. For optimum variance, differentiate $V[t_{mean}/T]$ with respect to $€_1$ and $€_2$ and equate to zero, one gets

$(€_1)_{opt} = (QR) / [PQ+PR+QR] = QM$

$(€_2)_{opt} = PQ / [PQ+PR+QR] = PM$ where $M = R / [PQ+PR+QR]$

One can differentiate the variance expression by α_j also to get optimum value which is $(\alpha_j)_{opt} = [(es'x')_j / (R_{Nj} * (ex')_j^2)]$ Substituting optimum choices in expression, the optimum variance is:

$V[t_{mean}/T]_{opt} = (€_1)^2_{opt} P + (€_2)^2_{opt} Q + (1 - (€_1)_{opt} - (€_2)_{opt})^2 R]$ with $(\alpha_j)_{opt}$ as above.

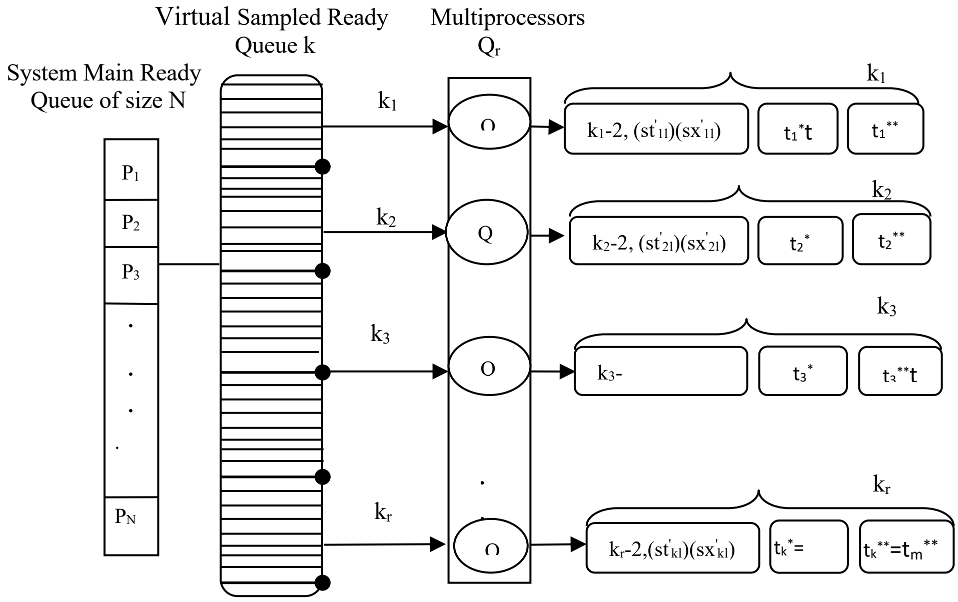


Figure 8: *Proposed Model of Virtual sampled Ready Queue Processing Time Estimation with size and Imputation*

V. Numerical Illustrations

Consider the 150 processes with processed CPU time whose details are in table 1 with assumption that all 150 processes have been completed.

Table 1: *System Ready Queue Processes with time (N = 150)*

Process	J_1	J_2	J_3	J_4	J_5	J_6	J_7	J_8	J_9	J_{10}	J_{11}	J_{12}	J_{13}	J_{14}	J_{15}
CPU Time	30	20	42	45	59	35	25	48	50	60	32	55	62	47	69
Process Size	41	71	103	142	316	82	199	163	220	127	76	192	251	52	133
Process	J_{16}	J_{17}	J_{18}	J_{19}	J_{20}	J_{21}	J_{22}	J_{23}	J_{24}	J_{25}	J_{26}	J_{27}	J_{28}	J_{29}	J_{30}
CPU Time	34	24	44	70	57	65	38	84	101	66	80	90	92	111	85
Process Size	318	202	106	181	242	148	46	252	136	222	261	97	109	271	116
Process	J_{31}	J_{32}	J_{33}	J_{34}	J_{35}	J_{36}	J_{37}	J_{38}	J_{39}	J_{40}	J_{41}	J_{42}	J_{43}	J_{44}	J_{45}
CPU Time	61	52	72	75	89	67	51	78	80	91	63	86	93	77	99
Process Size	172	243	253	262	83	203	183	166	219	193	223	272	281	301	289

Process	J_{46}	J_{47}	J_{48}	J_{49}	J_{50}	J_{51}	J_{52}	J_{53}	J_{54}	J_{55}	J_{56}	J_{57}	J_{58}	J_{59}	J_{60}
CPU Time	64	54	74	100	87	95	68	114	131	96	110	123	122	141	49
Process Size	205	244	223	254	146	263	53	218	273	139	282	302	173	309	290
Process	J_{61}	J_{62}	J_{63}	J_{64}	J_{65}	J_{66}	J_{67}	J_{68}	J_{69}	J_{70}	J_{71}	J_{72}	J_{73}	J_{74}	J_{75}
CPU Time	118	81	102	105	119	97	88	108	110	121	240	113	122	107	129
Process Size	313	194	153	255	225	169	206	264	58	274	283	303	184	291	216
Process	J_{76}	J_{77}	J_{78}	J_{79}	J_{80}	J_{81}	J_{82}	J_{83}	J_{84}	J_{85}	J_{86}	J_{87}	J_{88}	J_{89}	J_{90}
CPU Time	94	73	104	130	117	234	98	237	161	126	143	236	152	171	233
Process Size	207	246	228	360	256	275	217	265	226	195	284	292	304	300	280
Process	J_{91}	J_{92}	J_{93}	J_{94}	J_{95}	J_{96}	J_{97}	J_{98}	J_{99}	J_{100}	J_{101}	J_{102}	J_{103}	J_{104}	J_{105}
CPU Time	120	112	132	135	149	125	115	138	140	150	122	232	152	137	159
Process Size	247	79	208	276	285	257	56	293	266	187	305	178	310	299	215
Process	J_{106}	J_{107}	J_{108}	J_{109}	J_{110}	J_{111}	J_{112}	J_{113}	J_{114}	J_{115}	J_{116}	J_{117}	J_{118}	J_{119}	J_{120}
CPU Time	124	114	134	160	147	155	128	174	191	156	170	180	182	201	175
Process Size	277	286	211	248	227	294	157	258	229	267	196	298	188	306	270
Process	J_{121}	J_{122}	J_{123}	J_{124}	J_{125}	J_{126}	J_{127}	J_{128}	J_{129}	J_{130}	J_{131}	J_{132}	J_{133}	J_{134}	J_{135}
CPU Time	235	142	162	165	179	151	145	168	171	238	152	175	189	167	241
Process Size	287	278	295	197	249	307	268	311	213	350	112	314	259	297	230
Process	J_{136}	J_{137}	J_{138}	J_{139}	J_{140}	J_{141}	J_{142}	J_{143}	J_{144}	J_{145}	J_{146}	J_{147}	J_{148}	J_{149}	J_{150}
CPU Time	154	144	164	190	177	185	158	204	221	186	200	210	212	231	209
Process Size	214	250	260	279	288	296	308	269	312	245	317	198	319	315	239

Table 2: *Descriptive Statistics of Table 1*

S.No.	Parameters Name	Calculated value
1	Number of Processes N	150
2	Mean time ($\bar{t}$)	122.51
3	Total sum of square $=\sum t_i^2$	2697717
4	Mean square St^2	3080.62

Assume that there are three processors Q_1, Q_2, Q_3in the system (r = 3) and a random sample of k = 30 is drawn from N = 150 by lottery scheduling. The sample k = 30 is divided into k_1 = 12, k_2= 10, k_3 = 8 in sequential manner for virtual sampled ready queue. The k_j process are assigned to Q_j(j = 1, 2, 3). Calculation is performed on 10 random samples each of size 30. Computation for only one sample is presented below:

I. CASE I: $\alpha_j = 0$ ($\alpha_1 = 0$, $\alpha_2 = 0$, $\alpha_3 = 0$)

Calculation for Sample No. 1 where sample size k=30

k_1:(J_{01}, 30,41), (J_{31},61,172), (J_{61},118,313), (J_{91},120,247), (J_{121},235,287), (J_{63},102,153), (J_{32},52,243), (J_{62},81,194), (J_{92},112,79), (J_{122},142,278), (J_3,42,103), (J_{33},72,253)

Partial Processed = (J_{03}, 42), (Processed= 22, unprocessed=20), Blocked = (J_{33},72),

Blocked replaced α' =(J_8,48)

[$\bar{st}_1{}'$ = 104.9, from eq.(4.4),$(es')_1^2$= 3317.65, from eq.(4.7)],

[$\bar{sx}_1$=2363/12 = 196.91, from eq. (4.5),$\bar{sx}_1{}'$=2007/10= 200.7, from eq. (4.6)], $(ex')_1^2$=8158.45, from eq.(4.8)] [$R_{N_1}=\left[\frac{(\bar{st}')_1}{(\bar{sx}')_1}\right]$ =0.52,from eq.(4.10)],[$(es'x')_1$ = 2982.52,from eq.(4.9)]

k2:(J_{49},100,254), (J_{34},75,262), (J_{64},105,255), (J_{94},135,276), (J_{124},165,197), (J_{135},241,230), (J_{35},89,83), (J_{65},119,225) (J_{95},149,285), (J_{125},179,249)

Partial Processed=(J_{95},149),(Processed=100, unprocessed=49), Blocked=(J_{125},179),

Blocked replaced β' =(J_{38}, 78)

[$\bar{st}_{2'}$ = 128.62, from eq. (4.4), $(es')_2{}^2$= 2843.98 from eq.(4.7)]

[$\bar{sx}_2$=2316/10 = 231.6, from eq. (4.5),$\bar{sx}_{2'} = \frac{1782}{8} =$ 222.75, from eq. (4.6), $(ex')_2{}^2$=3806.21, from eq.(4.8)]

[$R_{N_2}=\left[\frac{(\bar{st}')_2}{(\bar{sx}')_2}\right]$= 0.57, from eq.(4.10)], [$(es'x')_2$ = 281.75, from eq.(4.9)]

k3:(J_{29}, 111,271), (J_{59}, 141,309), (J_{89}, 171,300), (J_{96},125,257), (J_{119},201,306), (J_{149}, 231,315), (J_{67}, 88,206), (J_{97}, 115, 56)

Partial Processed = (J_{67}, 88), (Processed=40, unprocessed=48), Blocked = (J_{97}, 115),

Blocked replaced γ'=(J_{10}, 60)

[$\bar{st}_{3'}$ = 163.33, from eq.(4.4), $(es')_3{}^2$= 2152.66 from eq.(4.7)]

[$\bar{sx}_3$=2020/8=252.5,from eq. (4.5),$\bar{sx}_{3'} = \frac{1758}{6} =$ 293, from eq. (4.6), $(ex')_3{}^2$= 547.6, from eq.(4.8)]

[$R_{N_3}=\left[\frac{(\bar{st}')_3}{(\bar{sx}')_3}\right]$=0.55, from eq.(4.10)],[$(es'x')_3$ = 841.2, from eq.(4.9)]

$\bar{t}^*$ = (22+100+40)/3 = 54

$\bar{t}^{**}$ = (α' + β' + γ')/3 = (48+78+60) / 3 = 62

Estimated $S_T{}^2$ = 1658 (using point 15)

Let P = $\sum_{i=1}^{r}\left(\frac{1}{(k_j-2)}-\frac{1}{k}\right) w_j{}^2 \{(es')_j{}^2+\alpha_j{}^2 R^2_{Nj}(ex')_j{}^2 - 2\alpha_j R_{Nj}(es'x')_j\}$

Q = $\sum\left(\frac{1}{r}-\frac{1}{N-k+r}\right)$[estimated $S_T{}^2$]

R = $\sum\left(1-\frac{1}{kj-2}\right) w_j (es')_j{}^2$

R = $\sum\left(1-\frac{1}{kj-2}\right) w_j (es')_j{}^2$

Calculation of P, Q, R at $\alpha_1=\alpha_2=\alpha_3 = 0$

P = $\left(\frac{1}{10}-\frac{1}{30}\right)(0.4)^2\{3317.65\}+\left(\frac{1}{8}-\frac{1}{30}\right)(0.33)^2\{2843.98\}+\left(\frac{1}{6}-\frac{1}{30}\right)(0.26)^2\{2152.66\}$

= 0.066 *0.16*3317.65+ 0.092*0.1089*2843.98 +0.133*0.0676*2152.66= 82.88

Q = $\left(\frac{1}{3}-\frac{1}{150-30+3}\right)$ (1658)= (0.3252 *1658) =539.18

R = $\left(1-\frac{1}{10}\right)(0.4)^2$*3317.65+ $\left(1-\frac{1}{8}\right)(0.33)^2$*2843.98 +$\left(1-\frac{1}{6}\right)(0.26)^2$*2152.66

= 0.9 *0.16*3317.65+ 0.875*0.1089*2843.98 +0.833*0.0676*2152.66 = 869.95

Calculation of Mean and Variance V[t_{mean}/T]

$(€_1)_{opt}$= (QR) / [PQ+PR+QR] = QM = 539.18*869.95/[82.88*539.18+82.88*869.95+539.18*869.95]

= 469059.641/585848.3354= 0.8006

$(€_2)_{opt}$= PQ/ [PQ+PR+QR] = PM =82.88*539.18/[82.88*539.18+82.88*869.95+539.18*869.95]

= 44687.2384/ 585848.3354= 0.0762

$t_{mean}/T = (€_1)_{opt}[\sum_{j=1}^{r} w_j \bar{t}_{rj}] + (€_2)_{opt}(\bar{t}^*) + (1-(€_1)_{opt}-(€_2)_{opt})(\bar{t}^{**})$

t_{mean}/T= 0.8006 [0.4*104.7(196.91/200.7)+0.33*128.62(231.6/222.75) +0.26*163.33(252.5/293)]

+0.0762*54 +0.1232*62 = 0.8006 [41.08 +44.13 +36.59] +4.11+1.88= 97.51+4.11+7.63= 109.25

$V[t_{mean}/T] = (€_1)^2_{opt} P + (€_2)^2_{opt} Q + (1-(€_1)_{opt}-(€_2)_{opt})^2 R$

V[t_{mean}/T] = [(0.8006)² *82.88+ (0.0762)²*539.18+ 0.0152*869.95] = 53.12+3.13+13.22 = 69.47

The 95% confidence intervals for $\bar{t}$, P [$(t_{mean}/T) \pm 1.96\sqrt{[V(t_{mean}/T)]}$] = 0.95

= 109.25± $1.96\sqrt{69.47}$ = 109.25± 16.33 = (92.92, 125.58)

Table 3: *Estimated Sample Mean, Variance and Confidence Interval(CI) of Ten Random Samples*
CASE I: At $(€_1)_{opt}$, $(€_2)_{opt}$, $\alpha_j = 0$ ($\alpha_1 = 0$, $\alpha_2 = 0$, $\alpha_3 = 0$)

S.No.	True Mean	Estimated Sample Mean	V[t_{mean}/T]	95% Confidence Interval (CI)	CI Length
1	122.51	109.25	69.47	(92.92, 125.58)	32.66
2	122.51	123.30	61.64	(107.92, 138.68)	30.76
3	122.51	107.67	75.92	(90.59, 124.74)	34.15
4	122.51	114	289.87	(80.63, 147.37)	66.74
5	122.51	128.09	285.83	(94.95, 161.22)	66.27
6	122.51	113.82	30.09	(103.07, 124.57)	21.50
7	122.51	119.23	39.79	(106.87, 131.59)	24.72
8	122.51	113.51	185.98	(86.78, 140.23)	53.45
9	122.51	133.73	175.83	(107.74, 159.30)	51.56
10	122.51	111.47	56.65	(96.72, 126.22)	29.5
			Average Length (411.31/10)		41.13

CASE II: $\alpha_j = 1$ ($\alpha_1 = 1$, $\alpha_2 = 1$, $\alpha_3 = 1$)

Calculation for Sample No. 1 , k=30, on above sample and P, Q, R at ($\alpha_1 = 1$, $\alpha_2 = 1$, $\alpha_3 = 1$)

P = $\left(\frac{1}{10}-\frac{1}{30}\right)$ $(0.4)^2$ {3317.65+1*0.52*0.52*8158.45-2*1*0.52*2982.52}
+$\left(\frac{1}{8}-\frac{1}{30}\right)$ $(0.33)^2$ {2843.98+1*0.57*0.57*3806.21-2*1*0.57*281.75}
+$\left(\frac{1}{6}-\frac{1}{30}\right)$ $(0.26)^2$ {2152.66+1*0.55*0.55*547.6-2*1*0.55*841.2}
= 0.066 *0.16*2421.87+ 0.092*0.1089*3759.42 +0.133*0.0676*1392.98 = 75.76
Q = $\left(\frac{1}{3} - \frac{1}{150-30+3}\right)$ 1658= 0.3252 *1658= 539.1816
R = $\left(1 - \frac{1}{10}\right)$ $(0.4)^2$ *3317.65+ $\left(1-\frac{1}{8}\right)$ $(0.33)^2$ *2843.98+$\left(1-\frac{1}{6}\right)$ $(0.26)^2$ *2152.66
= 0.9 *0.16*3317.65+ 0.875*0.1089*2843.98+0.833*0.0676*2152.66 = 869.95

Calculation of Mean and Variance V[t_{mean}/T] at ($\alpha_1 = 1$, $\alpha_2 = 1$, $\alpha_3 = 1$)

$(€_1)_{opt}$= (QR) / [PQ+PR+QR] = QM = 539.1816*869.95/[75.76*539.1816+75.76*869.95+539.1816*869.95]
= 469061.03292/575816.842936= 0.8146
$(€_2)_{opt}$= PQ/ [PQ+PR+QR] = PM =75.76*539.1816/[75.76*539.1816+75.76*869.95+539.1816*869.95]
= 40848.398016/ 575816.842936= 0.0709
(t_{mean}/T) = $(€_1)_{opt}$[$\sum_{j=1}^{r} w_j \bar{t}_{rj}$] + $(€_2)_{opt}$ ($\bar{t}^{*}$) + (1- $(€_1)_{opt}$– $(€_2)_{opt}$) ($\bar{t}^{**}$)
(t_{mean}/T)= 0.8146[0.4*104.7(196.91/200.7)+0.33*128.62(231.6/222.75) +0.26*163.33(252.5/293)]
+0.0709*54 +0.1145*62 = 0.8146[41.08+44.13 +36.59] +3.82+7.09=99.21+4.44+7.09= 110.74
V[t_{mean}/T] = $(€_1)^2_{opt}$ P + $(€_2)^2_{opt}$ Q + (1- $(€_1)_{opt}$– $(€_2)_{opt})^2 R$] with α_j = 1for all j = 1,2,3
V[t_{mean}/T] = [$(0.8146)^2$ *75.76+ $(0.0709)^2$*539.1816+ 0.0131*869.95] = 50.27+2.71+11.39=64.37
The 95% confidence intervals for $\bar{t}$,P [(t_{mean}/T)± 1.96$\sqrt{[\mathrm{V}\,(t_{mean}/T)}$] = 0.95
=110.74±1.96$\sqrt{64.37}$ =110.74±15.73 = (95.01, 126.46)

Table 4: *CASE II: At $(€_1)_{opt}$, $(€_2)_{opt}$, $\alpha_j = 1$ ($\alpha_1 = 1$, $\alpha_2 = 1$, $\alpha_3 = 1$)*
Sample Mean , Variance and Confidence Interval(CI) of Ten Random Samples

S.No.	True Mean	Estimated Sample Mean	V[t_{mean}/T]	95% Confidence Interval	CI Length
1	122.51	110.74	64.37	(95.01, 126.46)	31.45
2	122.51	125.72	49.78	(111.89, 139.55)	27.66
3	122.51	112.61	54.66	(98.11, 127.10)	28.99
4	122.51	113.87	305.14	(79.63, 148.10)	68.47
5	122.51	127.45	235.98	(97.35, 157.55)	60.2
6	122.51	113.63	62.92	(98.08, 129.18)	31.1
7	122.51	119.64	37.80	(107.58, 131.69)	24.11
8	122.51	144.02	144.34	(120.48, 167.57)	47.09
9	122.51	133.45	171.53	(107.77, 159.12)	51.35
10	122.51	122.85	40.12	(110.44, 135.26)	24.82
				Average Length (395.2/10)	39.52

III. CASE III: $\alpha_j = \alpha_{opt}$ where $(\alpha_{opt})_j = (es'x')_j / (R_{Nj}^* (ex')_j^2)$

Calculation for Sample No. 1, sample size k=30, $\alpha_1 = (\alpha_{opt})_1 = 0.70$, $\alpha_2 = (\alpha_{opt})_2 = 0.13$, $\alpha_3 = (\alpha_{opt})_3 = 2.79$and P, Q, R at $\alpha_j = (\alpha_{opt})_j$[$\alpha_1 = (\alpha_{opt})_1$, $\alpha_2 = (\alpha_{opt})_2$, $\alpha_3 = (\alpha_{opt})_3$]

P = $\left(\frac{1}{10}-\frac{1}{30}\right)$ $(0.4)^2$ {3317.65+0.70*0.70*0.52*0.52*8158.45-2*0.70*0.52*2982.52}
+$\left(\frac{1}{8}-\frac{1}{30}\right)$ $(0.33)^2$ {2843.98+0.13*0.13*0.57*0.57*3806.21-2*0.13*0.57*281.75}
+$\left(\frac{1}{6}-\frac{1}{30}\right)$ $(0.26)^2$ {2152.66+2.79*2.79*0.55*0.55*547.6-2*2.79*0.55*841.2}
= 0.066 *0.16*2227.34+ 0.092*0.1089*2823.12 +0.13*0.0676*860.45 = 59.36

Q = $\left(\frac{1}{3}-\frac{1}{150-30+3}\right)$ 1658= 0.3252 *1658 = 539.1816

R = $\left(1-\frac{1}{10}\right)$ $(0.4)^2$*3317.65+ $\left(1-\frac{1}{8}\right)$ $(0.33)^2$*2843.98 +$\left(1-\frac{1}{6}\right)$ $(0.26)^2$*2152.66
= 0.9 *0.16*3317.65+ 0.875*0.1089*2843.98 +0.833*0.0676*2152.66 = 869.95

Calculation of Mean and Variance V[t_{mean}/T]at$\alpha = (\alpha_{opt})_j$

$(€_1)_{opt}$= (QR) / [PQ+PR+QR] = QM = 539.1816*869.95/[59.36*539.1816+59.36*869.95+539.1816*869.95]
= 469061.03292/552707.084696= 0.8486

$(€_2)_{opt}$= PQ/ [PQ+PR+QR] = PM =59.36*539.1816/[59.36*539.1816+59.36*869.95+539.1816*869.95]
= 32005.819776/ 552707.084696= 0.0579

(t_{mean}/T)= $(€_1)_{opt}$[$\sum_{j=1}^{r} w_j \bar{t}_{rj}$] + $(€_2)_{opt}$ ($\bar{t}^*$) + (1- $(€_1)_{opt}$– $(€_2)_{opt}$) ($\bar{t}^{**}$)

(t_{mean}/T)= 0.8486[0.4*104.7(196.91/200.7)+0.33*128.62(231.6/222.75) +0.26*163.33(252.5/293)]
+0.0579*54 +0.0935*62 = 0.8486[41.08+44.13 +36.59] +3.13+5.797= 103.35+3.13+5.79= 112.27

V[t_{mean}/T] = $(€_1)^2_{opt}$ P + $(€_2)^2_{opt}$ Q + $(1- (€_1)_{opt}– (€_2)_{opt})^2 R$]

V[t_{mean}/T] = [$(0.8486)^2$ *59.36+ $(0.0579)^2$*539.1816+ 0.0087*869.95] = 42.74+1.80+7.56=52.10

The 95% confidence intervals for $\bar{t}$ P [(t_{mean}/T) ± 1.96$\sqrt{[V(t_{mean}/T)]}$ = 0.95
= 112.27± 1.96$\sqrt{52.10}$ = 112.27± 14.14 = (98.13, 126.41)

Table 5: *CASE III: At $(€_1)_{opt}$, $(€_2)_{opt}$, $\alpha_j = (\alpha_{opt})_j$ (j = 1, 2, 3)*
Estimated Sample Mean, Variance and Confidence Interval (CI) of Ten Random Samples

S.No.	True Mean	Estimated Sample Mean	V[t_{mean}/T]	95% Confidence Interval (CI)	CI Length
1	122.51	112.27	52.10	(98.13, 126.41)	28.28
2	122.51	126.91	44.04	(113.90, 139.91)	26.01
3	122.51	110.92	50.71	(96.96, 124.88)	27.92
4	122.51	114.44	32.79	(103.22, 125.66)	22.44
5	122.51	127.37	230.85	(97.59, 157.14)	59.55
6	122.51	114.59	26.79	(104.44, 124.74)	20.30
7	122.51	121.13	30.35	(110.33, 131.93)	21.60
8	122.51	139.34	105.75	(119.18, 159.49)	40.31
9	122.51	129.54	84.13	(111.56, 147.52)	35.96
10	122.51	123.66	31.31	(112.69, 134.62)	21.93
				Average Length (304.3/10)	30.43

IV. CASE IV: At $€_1 = 1$, $€_2 = 0$, with $\alpha_j = (\alpha_{opt})_j$

It is the case when no imputation used and partially processed situation not considered. But it is away from practical situation.

$(\alpha_{opt})_j = (es'x')_j / (R_{Nj} * (ex')_j^2)$

Calculation for Sample No. 1, sample size k=30, when $€_1$ = 1, $€_2$ =0 with $\alpha_j = \alpha_{opt}$ and P, Q, R at $(\alpha_{opt})_j$

P = 59.36, Q = 539.1816, R = 869.95

Calculation of Mean and Variance V[t_{mean}/T] at $(\alpha_{opt})_j$ with ($€_1$= 1, $€_2$=0)

$(t_{mean}/T) = €_1 [\sum_{j=1}^{r} w_j \bar{t}_{rj}] + €_2 (\bar{t}^*) + (1 - €_1 - €_2)(\bar{t}^{**})$

(t_{mean}/T)= 1* [0.4*104.7(196.91/200.7)+0.33*128.62(231.6/222.75) +0.26*163.33(252.5/293)]

= [41.08+44.13 +36.59] +0+0= 121.8

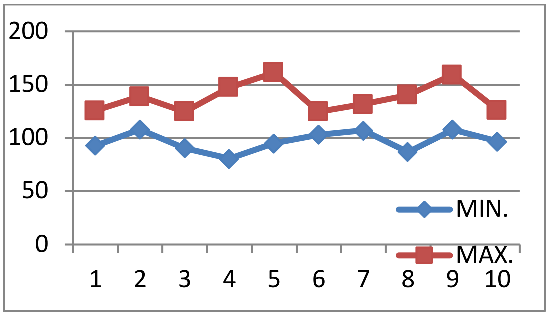
V[t_{mean}/T] = [$(1)^2$ *59.36+ $(0)^2$*539.1816+ 0*869.95] = 59.36

The 95% confidence intervals for $\bar{t}$ are P [$t_{mean}/T \pm 1.96\sqrt{[V(t_{mean}/T)]}$] = 0.95

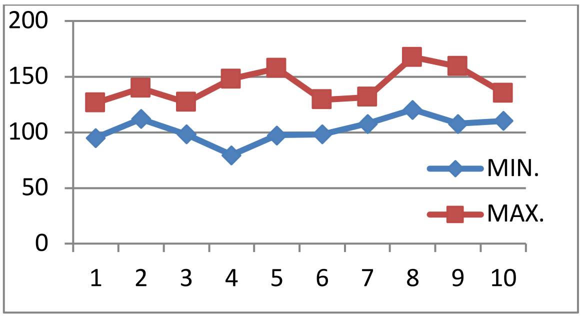
= 121.8± $1.96\sqrt{59.36}$ = 121.8± 15.10 = (106.70, 136.90)

Table 6: *CASE IV: when[$€_1 = 1$, $€_2 = 0$ $\alpha_j = (\alpha_{opt})_j$]*
Estimated Sample Mean, Variance and Confidence Interval (CI)of Ten Random Samples

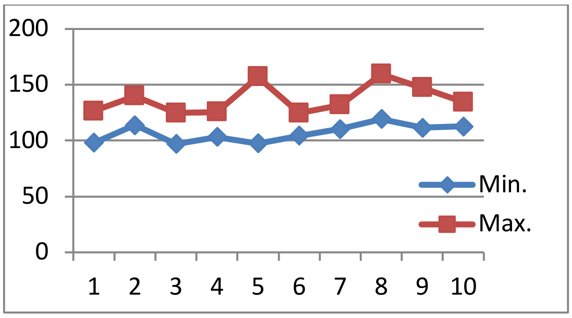
S.No.	True Mean	Estimated Sample Mean	V[t_{mean}/T]	95% Confidence Interval (CI)	CI Length
1	122.51	121.8	59.36	(106.70, 136.90)	30.20
2	122.51	136.51	50.11	(121.63, 150.38)	28.75
3	122.51	117.57	56.95	(60.62, 132.36)	71.74
4	122.51	119.77	47.64	(106.24, 133.29)	27.05
5	122.51	121.23	44.23	(108.19, 134.27)	26.08
6	122.51	125.01	27.56	(114.72, 135.30)	20.58
7	122.51	127.21	34.62	(115.67, 138.74)	23.07
8	122.51	116.64	58.63	(101.63, 131.64)	30.01
9	122.51	114	41.37	(101.40, 126.60)	25.20
10	122.51	127.23	30.69	(116.37, 138.09)	21.72
				Average Length	30.44



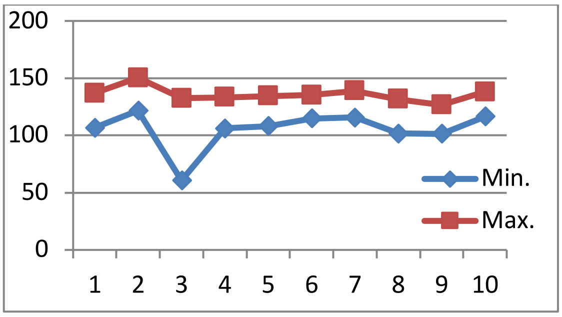
(a) Case I



(b) Case II



(c) Case III



(d) Case IV

Figure 9: *(a), (b), (c), (d) are graphical representation of Confidence Interval range of Ten Random Samples for four different cases of Table 3,4,5 and 6 (X-axis has sample number as shown in table 3,4,5,6)*

Table 7: *Comparison Between Cases I,II, and III*

S. No.	CASE 1 $\alpha_j = 0$ ($\alpha_1 = \alpha_2 = \alpha_3 = 0$)		CASE II $\alpha_j = 1$ ($\alpha_1 = \alpha_2 = \alpha_3 = 1$)		CASE III $(\alpha)_j = (\alpha_{opt})_j^2$		CASE IV [$\epsilon_1 = 1, \epsilon_2 = 0$] with $\alpha_j = \alpha_{opt}$	
	95%Confidence Interval	Length	95%Confidence Interval	Length	95%Confidence Interval	Length	95%Confidence Interval	Length
1.	(92.92, 125.58)	32.66	(95.01, 126.46)	31.45	(98.13, 126.41)	28.28	(106.70, 136.90)	30.20
2.	(107.92, 138.68)	30.76	(111.89, 139.55)	27.66	(113.90, 139.91)	26.01	(121.63, 150.38)	28.75
3.	(90.59, 124.74)	34.15	(98.11, 127.10)	28.99	(96.96, 124.88)	27.92	(60.62, 132.36)	71.74
4.	(80.63, 147.37)	66.74	(79.63, 148.10)	68.47	(103.22, 125.66)	22.44	(106.24, 133.29)	27.05
5.	(94.95, 161.22)	66.27	(97.35, 157.55)	60.2	(97.59, 157.14)	59.55	(108.19, 134.27)	26.08
6.	(103.07, 124.57)	21.50	(98.08, 129.18)	31.1	(104.44, 124.74)	20.30	(114.72, 135.30)	20.58
7.	(106.87, 131.59)	24.72	(107.58, 131.69)	24.11	(110.33, 131.93)	21.60	(115.67, 138.74)	23.07
8	(86.78, 140.23)	53.45	(120.48, 167.57)	47.09	(119.18, 159.49)	40.31	(101.63, 131.64)	30.01
9.	(107.74, 159.30)	51.56	(107.77, 159.12)	51.35	(111.56, 147.52)	35.96	(101.40, 126.60)	25.20
10.	(96.72, 126.22)	29.5	(110.44, 135.26)	24.82	(112.69, 134.62)	21.93	(116.37, 138.09)	21.72
Average Length		41.13		39.52		30.43		30.44

VI. Comparison and Discussion

The proposed setup has three parameters $€_1$, $€_2$and α ($0< \alpha < \infty$) whose suitable choices provide the best estimate. The case I has $\alpha_j= 0$ (for all j) which means there is no consideration of size measure in the strategy. Case II considers $\alpha_j= 1$ (for all j) indicating for the presence of size measure x in the estimation strategy but at a particular choice. Case III considers $\alpha_j= (\alpha_{opt})_j$ (for all j) where size measure is at the best (optimal) fractional level incorporated in strategy of prediction. All the three cases (see table 7) are showing the average length of confidence intervals, but smallest average interval length is 30.43 obtained by the case III where choices $(€_1)_{opt}$, $(€_2)_{opt}$ and$(\alpha_{opt})_j$ are used. The ten sample average confidence intervals are in table 8. Fig.9 shows smooth, increasing, condensed and controlled variations of lower and upper limits of CI, best found in the case III which deserved for recommendation.

Table8: *Ten Sample average Confidence Interval & estimated total processing Remaining time for Recovery Management*

	Case I (Without size measure)	Case II (With size measure)	Case III (With size measure)	True Value
Average Interval (Over 10 samples)	(96.8 - 137.9)	(102.6 - 142.1)	(106.8 - 137.2)	122.51
CI Length	41.1	39.5	30.4	
Lowest Predicted Remaining time	(N-k)*96.8 = 11,616 units	(N-k)*102.6 = 12312 units	(N-k)*106.8 = 12816 units	------
Highest Predicted Remaining time	(N-k)*137.9 =16,548 units	(N-k)*142.1 = 17052 units	(N-k)*137.2 = 16464 units	

To note that average intervals (table 8) are producing the same length as shown in table 7.Define relative efficiency measure in terms of percentage as:

$$\text{Percentage Relative Efficiency (PRE)} = \left[\frac{[\text{LengthofCIofcaseI}]-[\text{LengthofCIofothercases}]}{\text{LengthofCIofcaseI}}\right] X\ 100$$

Table 9: *Percentage Relative efficiency (PRE)*

Case II with respect to Case I	Case III with respect to Case I
PRE = 3.91 %	PRE = 26.01 %

The case III is more efficient (26.02%) than the case II with respect to case I as base where no size measure considered for estimation. In fact, all the sample computed confidence intervals are catching the true value (122.51) which is the strength of the proposed method. The minimal highest predicted time required to process the remaining jobs in ready queue (after breakdown)is 16464 units which is in case III (see table 8).

VII. Conclusion

On recapitulation, the paper considers the practical problem of remaining time estimation of processes in ready queue, after the occurrence of system failure in a multiprocessor computer system. While sudden breakdown how much backup time and computer related infrastructure required? This time duration and maximum time estimation are done using the tools of sampling theory and assumption of lottery scheduling. This scheduling opens avenues for application of random sampling tools and techniques. A concept of virtual ready queue is added as a new feature, who found useful in allocating the processes to multi-processors. The virtual ready queue along with lottery scheduling have created environment for the estimation of remaining time of main ready queue. The proposed estimation strategy is capable enough to predict for mean time. For efficient estimation $(€_1)_{opt}$, $(€_2)_{opt}$ and (α_{opt}) are used who provide the lowest length confidence interval. The Case III found best estimated and predicted than case I and case II. The case III also provides prediction indicating the minimal highest remaining time to arrange backup accordingly while failure. Imputation has improved the level of estimation and use of additional information (size measure) contributed a lot for higher precision. Such estimates are useful for backup and recovery management while the occurrence of system breakdown. Such findings are useful for risk evaluation and disaster management in setup of cloud computing and data centre.

References

[1] More Sarla, and Shukla Diwakar, Some new methods for ready queue processing time estimation problem in a multiprocessor environment, Social Networking and Computational Intelligence, Lecture notes in Networks and Systems, Springer, Singapore, March 2020; 100: 661-670.
[2] More, Sarla and Shukla Diwakar, Analysis, and extension of methods in ready queue processing time Estimation in Multiprocessor Environment, Proceedings of International Conference on Sustainable Computing in Science, Technology and Management (SUSCOM), Amity university Rajasthan, Jaipur-India, February 2019; 1558-1563.
[3] More, Sarla and Shukla Diwakar, "A review on ready queue processing time estimation problem and methodologies used in multiprocessor environment", International Journal of Commuter Science and Engineering, 2018; 6(5): 1186-1191.
[4] Shukla Diwakar, Jain Anjali, and Choudhary Amita, Estimation of ready queue processing time under SL scheduling scheme in multiprocessors environment, International Journal of Computer Science and Security, 2010; 4(1): 74-81.
[5] Shukla Diwakar, Jain Anjali and Choudhary Amita, "Estimation of ready queue Processing time under usual group lottery scheduling (GLS) in multiprocessor environment", International Journal of Commuter Applications, 2010; 8(14): 39-45.
[6] Shukla Diwakar, Jain Anjali and Choudhary Amita, Prediction of ready queue processing time in multiprocessor environment using lottery scheduling (ULS), International Journal of Commuter Internet and Management, 2010; 18 (3):58-65.
[7] Shukla Diwakar, and Jain Anjali, Analysis of ready queue processing time under PPS-LS and SRS-LS scheme in multiprocessing environment, GESJ: Computer Science and Telecommunications, 2012; 33(1): 54-61.
[8] Shukla Diwakar and Jain Anjali, Estimation of ready queue processing time using efficient factor type estimator (E-F-T) in multiprocessor environment, International Journal of Computer Applications, 2012; 48(16): 20-27.
[9] Shukla Diwakar and Jain Anjali, Ready queue mean time estimation in lottery scheduling using auxiliary variables in multiprocessor environment, International Journal of Commuter Applications, 2012; 55(13):13-19.
[10] Jain, Anjali and Shukla, Diwakar, Estimation of ready queue processing time using factor type (F-T) estimator in multiprocessor environment, COMPUSOFT, An International Journal of Advanced Computer Technology, 2013; 2(8): 256-260.
[11] Shukla Diwakar , Jain Anjali and Verma Kapil, Estimation of ready queue processing time using transformed factor- type (T-F-T) estimator in multiprocessor environment, International Journal of Computer Applications, 2013;79(16): 40-48.
[12] Carl A. Waldspurger and E William Weihl, Lottery Scheduling: Flexible proportional share resource management, The 1994 Operating systems design and implementation conference (OSDI '94), Monterey, California.

[13] Johnnie Daniel, Sampling Essentials: Practical Guidelines for Making Sampling Choices, Sage Publication, 2011.

[14] Paul S. Levy and Stanley Lemeshow, Sampling of Populations: Methods and Applications, Volume 543 of Wiley Series in Survey Methodology, Wiley, 2008.

[15] Sampath, S.,Sampling Theory and Methods, Alpha Science International Publication, 2005.

[16] Cochran, W.G, Sampling Technique, Wiley Eastern publication, New Delhi,2005.

[17] Poduri S. R. S. Rao, Sampling Methodologies with Applications, Texts in Statistical Science, Chapman and Hall/CRC Press, 2000.

[18] Ranjan K. Som, Practical Sampling Techniques, Second Edition Statistics: A Series of Textbooks and Monographs, CRC Press, 1995.

[19] Steven K. Thompson, Sampling, Volume 272 of Wiley Series in Probability and Statistics, 1992.

[20] Shweta Ojha, Saurabh Jain and Diwakar Shukla, Hybrid lottery multi-level queue scheduling with a Markovian model, GESJ: Computer Science and Telecommunications, 2011; 3(32): 86-97.

[21] Pradeep kumarJatav, Rahul Singhai, and Saurabh Jain, Analysis of hybrid lottery scheduling algorithm using Markov chain model, International Journal for Research in Engineering Application and Management (IJREAM), 2018; .04 (09): 180-192.

[22] MaríaMejía, Adriana Morales-Betancourt and Tapasya Patki, Lottery scheduler for the Linux kernel, The author; licensee universidad Nacional de Colombia. Dyna 82 (189), Medellín. Printed, Online DOI: http://dx.doi.org/10.15446/dyna.v82n189.43068, 2015; 216-225.

[23] Diwakar Shukla and Sarla More, Modified group lottery scheduling algorithm for ready queue mean time estimation in multiprocessor environment, Reliability: Theory & Applications (RT&A), 2020; 15, 4(59): 69-85.

[24] Carl A. Waldspurger, Lottery and stride scheduling: Flexible proportional-share resource management, Massachusetts Institute of Technology, September 1995.

[25] Singh H.P., Gupta A. and Tailor R., Estimation of population mean using a difference-type exponential imputation method, https://doi.org/10.1007/s42519-020-00151-2, Journal of Statistical Theory and Practice 2021; 15(19).

[26] Kone Dramane, Goore Bi Tra, and Dr. Kimou Kouadio Prosper, New hybrid method for efficient imputation of discrete missing attributes, International Journal of Innovative Science and Research Technology, 2020; 5(11): 983-991.

[27] GaribNath Singh, Mohd. Khalid and Jong-Min Kim, Some imputation methods to deal with the problems of missing data in two-occasion successive sampling, Communications in Statistics-Simulation and Computation, 2019; 50(2):557-580.

[28] Diwakar Shukla, Narendra Singh Thakur and Sharad Pathak, Some new aspects on imputation in sampling, African Journal of Mathematics and Computer Science Research, 2013; 6(1): 5-15.

[29] Diwakar Shukla, D. S. Thakur and N. S. Thakur, Utilization of mixture of $\bar{x}$, $\bar{x}_1$, and $\bar{x}_2$ in imputation for missing data in post-stratification, African Journal of Mathematics and Computer Science Research, 2012; 5(4):78-89.

[30] Diwakar Shukla, Narendra Singh Thakur, Dharmendra Singh Thakur and Sharad Pathak, Linear combination based imputation method for missing data in sample, International Journal of Modern Engineering Research (IJMER) 2011;1(2): 580-596.

[31] Diwakar Shukla and Narendra Singh Thakur, Imputation Methods in Sampling Hardcover, Aman Prakashan, January 2014, available at *Amazon:https://www.amazon.in/Imputation-Methods-Sampling-DiwakarShukla/dp/B07VY2B59D.*