

JAERI-M

6699

FACOM230-75システムの待行列
モデルによる解析

1976年9月

石 黒 美 佐 子

この報告書は、日本原子力研究所が JAERI-M レポートとして、不定期に刊行している研究報告書です。入手、複製などのお問い合わせは、日本原子力研究所技術情報部（茨城県那珂郡東海村）あて、お申しこしてください。

JAERI-M reports, issued irregularly, describe the results of research works carried out in JAERI. Inquiries about the availability of reports and their reproduction should be addressed to Division of Technical Information, Japan Atomic Energy Research Institute, Tokai-mura, Naka-gun, Ibaraki-ken, Japan.

FACOM 230-75 システムの待行列モデルによる解析

日本原子力研究所東海研究所計算センター

石黒美佐子

(1976年8月11日受理)

F230-75 におけるオープンバッチ多重ジョブ処理を、 $M/M/n$ 型待行列モデルにより解析する。このモデルでは、ジョブは、平均実行率が $\mu(n)$ の n 個の独立の計算機により処理されていると見なされる。ここで n は実行多重度で、その値は0から指定多重度 N まで動くものとする。

$\mu(n)$ は、多重度を n と固定した時の実行時における別の待行列モデルから別々に計算される。

実行経過時間、ターンアラウンドタイム、スループットなどの数値計算結果の比較が多重度の変化に対して図示されている。

Analysis for a Queuing Model of FACOM 230-75 System

Misako ISHIGURO

Computing Center, Tokai, JAERI

(Received August 11, 1976)

An open batch multi-job-processing for F230-75 System is analyzed by a M/M/n queuing model. In the model, jobs are considered to be processed with n independent machines whose mean processing rate is $\mu(n)$, where n is the actual multiplicity which varies from 0 to a designated multiplicity N.

The $\mu(n)$ is separately calculated from another queuing models of the executing stage in which multiplicity is held to n.

Numerical comparisons of processing time, turnaround time, throughput, etc. are shown in the figures to the variation of job multiplicities.

目 次

1	ま え が き	1
2	全体モデルの解析	2
2.1	M/M/n型モデル	2
2.2	確率計算	3
3	実行時モデル	8
3.1	確率計算	8
3.2	実行率 $\mu(n)$ の計算	12
4	実行経過時間の分布に関する考察	13
5	必要とするデータとその収集方法	15
6	解 析 結 果	18
7	検 討	29
謝 辞		37
参 考 文 献		37

Contents

1. Introduction.....	1
2. Analysis for Full Model.....	2
2.1 M/M/n Queuing Model.....	2
2.2 Probability Calculation.....	3
3. Queuing Model for Executing Stage.....	8
3.1 Probability Calculation.....	8
3.2 Calculation for Job Processing Rate $\mu(n)$	12
4. Investigation for Job Processing Time Distribution.....	13
5. Basic Data and Its Gathering Method.....	15
6. Results of the Analysis.....	18
7. Discussion.....	29
Acknowledgement.....	37
References.....	37

1 ま え が き

大型計算機の性能評価について、前回までは、測定を中心としたデータ集めに重点が置かれていた。

今回は、F230-75をM/M/nの待行列モデル化し、理論的に解析し、実測だけでは判断できない事柄、例えば、指定多重度の変更、ジョブの負荷の増減、実装装置台数の増減、装置の処理能力の増減に対して、スループット、ターンアラウンドタイム等コンピュータの総合性能がどうなるか検討する。

多重プログラミングシステムに対しては、種々の待行列モデルによる解析がなされている〔1〕、〔2〕、〔3〕。いずれも実行時モデルの解析に重点が置かれている。ここでは、大型センターで不特定多数のユーザを相手にオープンバッチ処理を行う場合のシステム解析、特にターンアラウンドタイム、ジョブの実行経過時間、ジョブの実行待行列、装置の並行処理率とジョブ多重度との関係等、実行時モデルを包括するバッチ処理の全容をモデル化し解析する。

システム全体を一挙にモデル化するとモデルが複雑になり過ぎて解析が困難であるために、まずバッチ処理を入力 → 実行 → 出力に区分し、実行時モデルと切り離して別に計算し全体モデルにはめ込むことにする。実行時モデルからジョブの平均の実行率 $\mu(n)$ を実行多重度 n の関数として計算し、それを全体モデルの解析に使用する。ここでの考え方は、実行時には指定多重度 N またはそれ以下の数、 n 台分の計算機が並行に動作していて、各々が平均実行時間 $e(n)$ のジョブを実行していると見立ててモデル化することである。指定多重度が N でも、実際はある確率 $P_r(n)$ で N 以下の多重度 n で実行されている場合があり、実行時モデルは実行多重度 n にしたがって変動すると考える。

このように、このモデルでは、指定多重度以下で実行されている場合の確率的取扱を含んでいるのでより現実を反映している。また実行時モデルをサブモジュールとして別に解析し、全体モデルにはめ込む一つの方法を与え解析の単純化をはかっている。

以下2ではオープンバッチ処理の全体をM/M/n型にモデル化し解析する。3では実行段階を、多重度が n で平衡状態となっている場合を想定し解析し、平均ジョブ実行率 $\mu(n)$ を算定する。4においては、平均実行経過時間について、実行時モデル解析時の幾何分布による取扱と、全体モデルにおける指数分布の取扱を関連づける。5では解析結果を数値計算するのに必要な基本データを前回の測定結果〔6〕から作成し、6でターンアラウンドタイム、実行経過時間等解析結果を図示し、7でこのモデルによる制限事項等を明確にする。

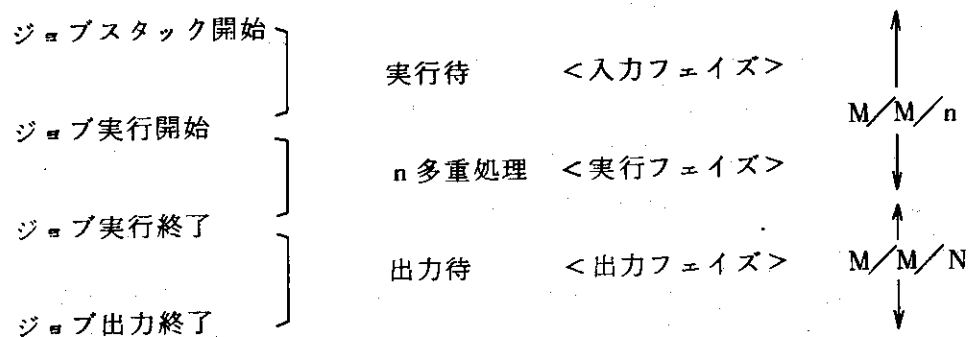
2 全体モデルの解析

2.1 M/M/n型待行列モデル

図1は、オープンバッチシステムを、入力、実行、出力のフェイズに分けてモデル化したものである。ジョブの到着はパラメータ λ のポアソン分布にしたがうものとし、到着したジョブはFCFS (First Come First Served) のスケジュールにしたがって実行される。ジョブの優先権、ジョブクラスは考慮しない。ジョブはあらかじめ指定した多重度 N で実行されるものとし、実行多重度が N 未満の場合に入力したジョブはただちに実行が開始される。

ジョブの実行経過時間(ジョブ開始から終了まで)は、その時の実行多重度 n ($0 \leq n \leq N$)に依存したパラメータ $\mu(n)$ を持つ指数分布にしたがうものとする。 $\mu(n)$ は実行時のモデルの解析から別途算定される値であり、平均実行経過時間 $e(n)$ の逆数である。

実行終了後FCFSにより L 台の出力装置(ラインプリンタを想定している)により並行して出力される。 L 台の出力装置が稼働中のときは、出力待行列に入って待たされる。出力所要時間もパラメータ μ_z を持つ指数分布にしたがうものとする。 L 台のどの装置も同一のパラメータ μ_z の出力率を持つものとする。



一般に、指数分布 $f(x) = 1 - e^{-\mu x}$ はマルコフ過程をなす性質を持つことはよく知られている[1]。

すなわち、ある時間 t_0 における次のふるまい(=状態) $X(t)$ は、現在の状態 $X(t_0)$ のみ依存して決まり、 t_0 以前の過去の状態には依存しない。これを指数分布のmemoryless propertyと呼ぶ。

ジョブ到着がポアソン分布(つまり指数分布)、ジョブのサービス時間の分布が同じく指数分布で、 n 台の窓口からサービスを受けるとき $M/M/n$ 型モデルと呼ぶ。ここで M はMarkovの意味である。一番単純な待行列モデルは $M/M/1$ 型であり、いろいろな解析を行っていく上で、式の導出が容易であるという理由から最も都合の良い仮定である。

ランダムな到着はポアソン分布にしたがうことについては現実の事象とよい一致が見られる。到着に比して、サービス時間の分布は、計算機の使用時間のようなものに対しては指数分布に適合しないものが見受けられる[5]。したがってより一般的な分布を用いて待行列モデルを解析する必要があり、このような場合には $M/G/n$ モデルと呼ぶ。 G はGeneralから来る。

この場合もマルコフ・チェインにはめ込んで解析可能である。

我々の解析では、解析の容易さからM/M/n型モデルにより取扱うことにする。現実のシステムに対する到着分布、サービス分布に対する考察は〔5〕でなされている。サービス分布は $1 - e^{-a\sqrt{x}}$ の型に適合が見られる。

2.2 確率計算

図1より実行待行列の長さを x ($x \geq 0$)、実行時のジョブの多重度を n ($0 \leq n \leq N$)、出力待行列の長さに現在出力中のジョブを加えたものを Z ($Z \geq 0$) とし、それらの値が (x, n, z) となる確率を $p(x, n, z)$ で表わすことにする。ここで N は指定多重度とする。

(x, n, z) の状態遷移を図2で示す。図2から平衡状態における方程式は以下のようになる。

(a) $x \geq 1, n = N, z > 1$ のとき

$$\lambda p(x-1, N, z) + N\mu(N) p(x+1, N, z-1) + \ell_{z+1} \mu_z p(x, N, z+1) \\ = \{ \lambda + N\mu(N) + \ell_z \mu_z \} p(x, N, z)$$

$$\text{ここで } \ell_z = \begin{cases} z & , 1 \leq z \leq L \\ L & , L \leq z \end{cases} \quad (1)$$

(b) $x \geq 1, n = N, z = 0$ のとき

$$\lambda p(x-1, N, 0) + \mu_z p(x, N, 1) = \{ \lambda + N\mu(N) \} p(x, N, 0) \quad (2)$$

(c) $x = 0, n = N, z \geq 1$ のとき

$$\lambda p(0, N-1, z) + N\mu(N) p(1, N, z-1) + \ell_{z+1} \mu_z p(0, N, z+1) \\ = \{ \lambda + N\mu(N) + \ell_z \mu_z \} p(0, N, z) \quad (3)$$

(d) $x = 0, n = N, z = 0$ のとき

$$\lambda p(0, N-1, 0) + \mu_z p(0, N, 1) = \{ \lambda + N\mu(N) \} p(0, N, 0) \quad (4)$$

(e) $1 \leq n < N, z \geq 1$ のとき ($x = 0$)

$$\lambda p(0, n-1, z) + (n+1)\mu(n+1) p(0, n+1, z-1) \\ + \ell_{z+1} \mu_z p(0, n, z+1) = \{ \lambda + n\mu(n) + \ell_z \mu_z \} p(0, n, z) \quad (5)$$

(f) $1 \leq n < N, z = 0$ のとき ($x = 0$)

$$\lambda p(0, n-1, 0) + \mu_z p(0, n, 1) = \{ \lambda + n\mu(n) \} p(0, n, 0) \quad (6)$$

(g) $n = 0, z \geq 1$ のとき ($x = 0$)

$$\mu(1) p(0, 1, z-1) + \ell_{z+1} \mu_z p(0, 0, z+1) = \{ \lambda + \ell_z \mu_z \} p(0, 0, z) \quad (7)$$

(h) $n = 0, z = 0$ のとき ($x = 0$)

$$\mu_z p(0, 0, 1) = \lambda p(0, 0, 0) \quad (8)$$

$\lambda, \mu(1) \sim \mu(N), L, \mu_z$ を既知量として $p(x, n, z)$ を求める。

次の三つの関係式は方程式(1)~(8)式を満す。

$$\textcircled{1} \quad p(x, N, z) = \frac{\lambda}{N\mu(N)} p(x-1, N, z), \quad x \geq 1 \quad (9)$$

$$\textcircled{2} \quad p(0, n, z) = \frac{\lambda}{n\mu(n)} p(0, n-1, z), \quad 1 \leq n \leq N \quad (10)$$

$$\textcircled{3} \quad p(x, n, z) = \frac{\lambda}{\ell_z \mu_z} p(x, n, z-1), \quad z \geq 1 \quad (11)$$

これらの関係式から方程式の特殊解 $p(x, n, z)$ を次のように得る。

$$p(x, N, z) = \left\{ \frac{\lambda}{N\mu(N)} \right\}^x p(0, N, z) \quad (\textcircled{1} \text{より})$$

$$= \left\{ \frac{\lambda}{N\mu(N)} \right\}^x \left\{ \prod_{i=1}^N \frac{\lambda}{i\mu(i)} \right\} p(0, 0, z) \quad (\textcircled{2} \text{より})$$

$$= \left\{ \frac{\lambda}{N\mu(N)} \right\}^x \left\{ \prod_{i=1}^N \frac{\lambda}{i\mu(i)} \right\} \left\{ \prod_{i=1}^z \frac{\lambda}{\ell_i \mu_z} \right\} p(0, 0, 0)$$

($\textcircled{3}$ より)

(12)

以下同様にして,

$$p(x, N, 0) = \left\{ \frac{\lambda}{N\mu(N)} \right\}^x \left\{ \prod_{i=1}^N \frac{\lambda}{i\mu(i)} \right\} p(0, 0, 0) \quad (13)$$

$$p(0, n, z) = \left\{ \prod_{i=1}^n \frac{\lambda}{i\mu(i)} \right\} \left\{ \prod_{i=1}^z \frac{\lambda}{\ell_i \mu_z} \right\} p(0, 0, 0) \quad (14)$$

$$p(0, 0, z) = \left\{ \prod_{i=1}^z \frac{\lambda}{\ell_i \mu_z} \right\} p(0, 0, 0) \quad (15)$$

$$p(0, n, 0) = \left\{ \prod_{i=1}^n \frac{\lambda}{i\mu(i)} \right\} p(0, 0, 0) \quad (16)$$

$$\text{ここで } \ell_z = \begin{cases} z & , \quad 1 \leq z \leq L \\ L & , \quad z \geq L \end{cases}$$

$$1 \leq n \leq N$$

一方、 $\sum_{z=0}^{\infty} \left\{ \sum_{n=0}^{N-1} p(0, n, z) + \sum_{x=0}^{\infty} p(x, N, z) \right\} = 1$ より、(12)~(16)式を入れて計算すると、

$$\begin{aligned}
 & p(0, 0, 0) + \sum_{n=1}^{N-1} p(0, n, 0) + \sum_{x=0}^{\infty} p(x, N, 0) \\
 & + \sum_{z=1}^{\infty} \left[p(0, 0, z) + \sum_{n=1}^{N-1} p(0, n, z) + \sum_{x=0}^{\infty} p(x, N, z) \right] = 1, \\
 & \left(1 + \sum_{z=1}^{\infty} \prod_{i=1}^z \frac{\lambda}{\ell_i \mu_z} \right) \left(1 + \sum_{n=1}^{N-1} \prod_{i=1}^n \frac{\lambda}{i \mu(i)} + \prod_{i=1}^N \frac{\lambda}{i \mu(i)} \sum_{x=0}^{\infty} \left\{ \frac{\lambda}{N \mu(N)} \right\}^x \right)
 \end{aligned}$$

$$p(0, 0, 0) = 1$$

$$\begin{aligned}
 A(N) = & \left\{ 1 + \frac{\lambda}{\mu(1)} + \frac{\lambda^2}{2\mu(1)\mu(2)} + \dots \right. \\
 & + \frac{\lambda^{N-1}}{(N-1)! \mu(1)\mu(2)\dots\mu(N-1)} \\
 & \left. + \frac{\lambda^N}{(N-1)! \mu(1)\mu(2)\dots\mu(N-1) \{ N\mu(N) - \lambda \}} \right\}
 \end{aligned}$$

とおくと

$$\left(1 + \sum_{z=1}^{\infty} \prod_{i=1}^z \frac{\lambda}{\ell_i \mu_z} \right) A(N) p(0, 0, 0) = 1$$

$$\text{ゆえに } p(0, 0, 0) = \left\{ \left(1 + \sum_{z=1}^{\infty} \prod_{i=1}^z \frac{\lambda}{\ell_i \mu_z} \right) A(N) \right\}^{-1} \quad (17)$$

但し(17)式は $\lambda < N\mu(N)$ の範囲で成立する。 $\lambda \geq N\mu(N)$ の場合は系が暴走し、このモデルでは計算が不可能である。このような場合には実行待行列が無限に長くなり、確率の計算をする際に、 x が十分大きい場合を無視できないので、有限の変数範囲におさまらないという不都合が起る。

表1に、指定多重度を N とした場合に計算実行可能な λ の値の上限を示す。この値は、以下で計算される $N\mu(N)$ の値である。

表1 モデルが計算可能なλの値

指定多重度 N	入力率 (λ) の上限
3	0.0172 (ジョブ/秒)
4	0.0198
5	0.0214
6	0.0223
7	0.0228
8	0.0232
9	0.0234
10	0.0235
11	0.0236
12	0.0236

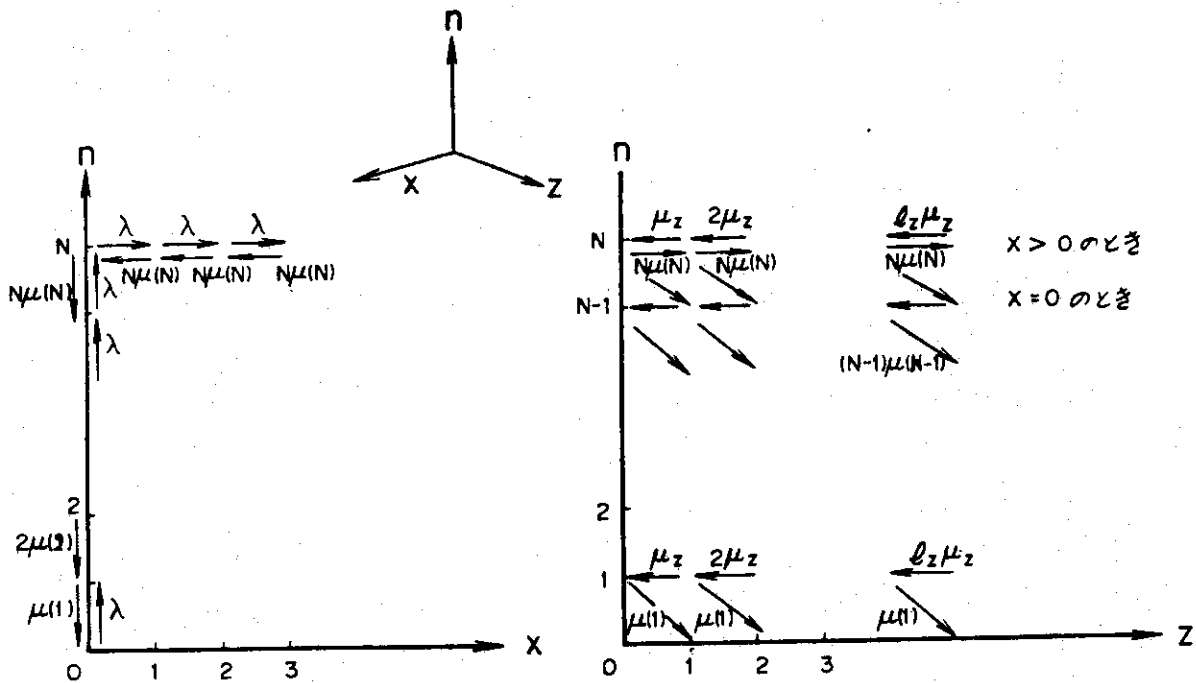


図2 状態遷移図

Fig.2 Figure of State Transition

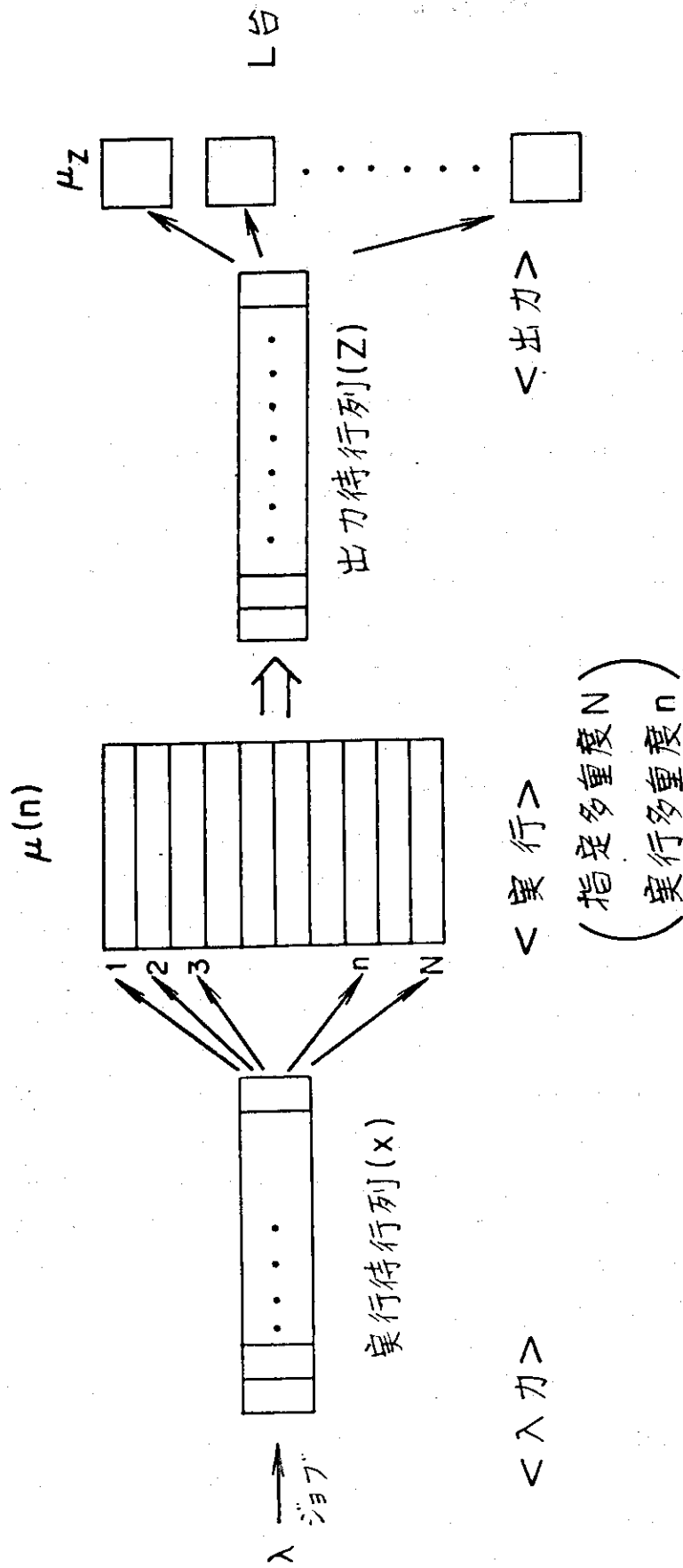


図1 オープンバッチ処理のモデル

Fig.1 Model of an Open Batch Processing System

3 実行時モデル

3.1 確率計算

図3に示すような実行多重度が n のときの実行時モデルを作る。ジョブの実行過程は図3に示すようにCPU使用、チャンネル使用が交互にくり返される m 回のパスから成るものとする。タイムスライスによるタスクの切換えは考慮しないものとする。CPU、チャンネルの使用にはOSのオーバーヘッドも含まれていると見なす。そして1回にCPUを継続して使用する時間はパラメータ μ を持つ指数分布にしたかうものとする。

計算機の構成は、大型システムを想定し、二つのCPUが同一の能力を有し平等に使用されるMP構成とする。チャンネルは、磁気ドラム用のHSC (High Speed Channel) と磁気ディスク用のBMC (Block multiplexer Channel) の二つのみを考慮する。ラインプリンターなどの入出力装置のためのマルチプレキサー・チャンネルは実行のフェーズでは使用しないものとし、ディスクを経由して入力・出力フェーズで処理されると考える。磁気テープはオープンバッチ処理という処理形態から使用を禁止されているものとし、その代り大容量の補助記憶装置を持つファイル・オリエンティド・システムとして運用されているとする。

1回のCPU使用後は必ずいずれかのチャンネルが使用されるがその比を σ_1, σ_2 とする。ここで $\sigma_1 + \sigma_2 = 1$ である。それぞれのチャンネルの使用時間は、パラメータ μ_1, μ_2 を持つ指数分布にしたかうものとする。

ジョブの実行開始は、最初にCPU待行列に入った時点とし、実行終了は最後に特定のチャンネル(ここではHSC)の処理を終了し系を出ていく時点とする。いずれかのCPUが空くとFCFSのスケジュールでCPU待行列から次の処理要求が一つ取出され処理される。現実のシステムでは実行優先権にもとづいてCPU割当が行われているが、ここでは優先処理を考慮しないことにする。またジョブ(=タスク)はすべて主記憶上にあると仮定する。CPU使用後はHSCかBMCのいずれかのチャンネル待行列に入りFCFSのスケジュールで処理される。この場合も優先処理をしないこととする。HSC使用後は ρ の確率でジョブが終了し、実行時モデルの外に出る。BMC使用後は必ずCPU待行列にもどるものとする。

実行時モデルにおいては、1ジョブ当りのCPUチャンネルのパスのくり返し数の平均 m が 10^3 程度の値だから $\rho \approx \frac{1}{m}$ は小さい値であり、HSC使用後もCPU待行列にもどる場合が確率 $(1 - \rho)$ であるから、圧倒的に多い。またジョブが終了した場合は ρ の確率となるが、系が多重度 n で平衡状態を保っているという仮定を置くと、ただちに次のジョブがCPU待行列に入ってくると考えることができる。したがってHSC使用後も必ず待行列にもどると考えてさしつかえない。

CPU待行列の長さ x とCPU使用中のジョブ(=タスク)の和を x 、チャンネル待行列の長さ y_1 とチャンネル使用中のもの y_2 の和をそれぞれ y_1 (HSC), y_2 (BMC) とする。実行多重度が n のとき、平衡状態におけるこれらの値が (x, y_1, y_2) となる確率 $q(x, y_1, y_2)$ を求める。

$$x + y_1 + y_2 = n, \quad 0 \leq x, y_1, y_2 \leq n$$

図3から平衡状態における方程式は以下のようなになる。

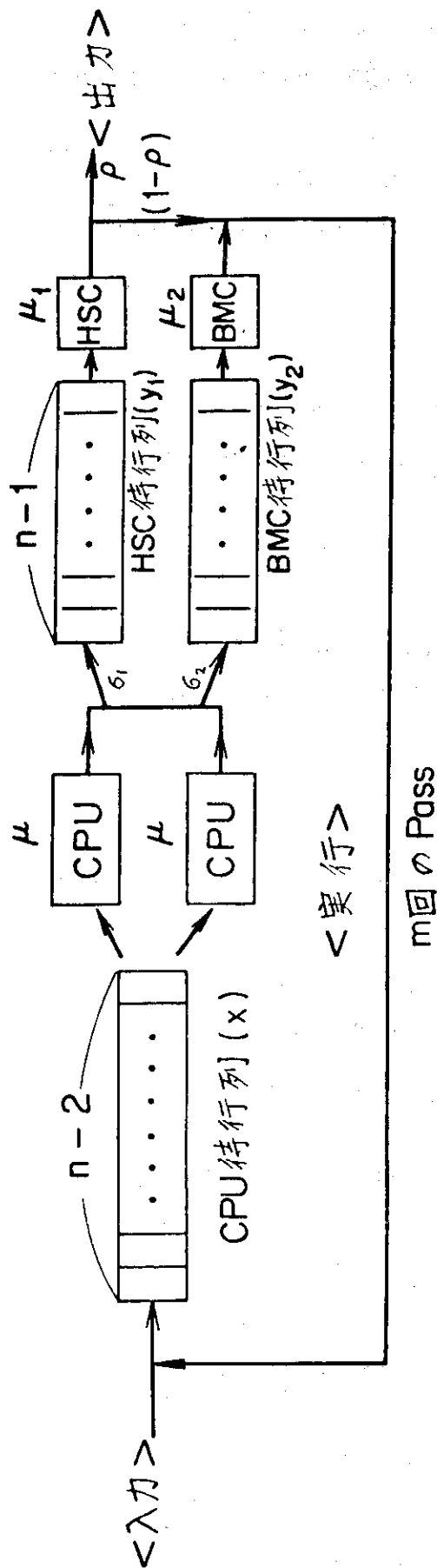


図 3 実行時モデル (多重度 = n)

Fig.3 Model of the Executing Stage

(a) $x \geq 2, y_1 > 0, y_2 > 0$ のとき

$$\begin{aligned} & \mu_1 q(x-1, y_1+1, y_2) + \mu_2 q(x-1, y_1, y_2+1) \\ & + 2\mu\sigma_1 q(x+1, y_1-1, y_2) + 2\mu\sigma_2 q(x+1, y_1, y_2-1) \\ & = (2\mu + \mu_1 + \mu_2) q(x, y_1, y_2) \end{aligned} \quad (18)$$

(b) $x \geq 2, y_1 = 0$ のとき ($y_2 = 0$ の場合も同様)

$$\begin{aligned} & \mu_1 q(x-1, 1, y_2) + \mu_2 q(x-1, 0, y_2+1) + 2\mu\sigma_2 q(x+1, 0, y_2-1) \\ & = (2\mu + \mu_2) q(x, 0, y_2) \quad , \\ & \quad x + y_2 = n \end{aligned} \quad (19)$$

(c) $x = 1, y_1 > 0, y_2 > 0$ のとき

この場合は CPU が 1 台だけ動いている状態である。

$$\begin{aligned} & \mu_1 q(0, y_1+1, y_2) + \mu_2 q(0, y_1, y_2+1) + 2\mu\sigma_1 q(2, y_1-1, y_2) \\ & + 2\mu\sigma_2 q(2, y_1, y_2-1) \\ & = (\mu + \mu_1 + \mu_2) q(1, y_1, y_2) \\ & \quad y_1 + y_2 = n - 1 \end{aligned} \quad (20)$$

(d) $x = 1, y_1 = 0$ のとき ($y_2 = 0$ の場合も同様)

$$\begin{aligned} & \mu_1 q(0, 1, y_2) + \mu_2 q(0, 0, y_2+1) + 2\mu\sigma_2 (2, 0, y_2-1) \\ & = (\mu + \mu_2) q(1, 0, y_2) \\ & \quad x + y_2 = n \end{aligned} \quad (21)$$

(e) $x = 0, y_1 > 0, y_2 > 0$ のとき

$$\begin{aligned} & \mu\sigma_1 q(1, y_1-1, y_2) + \mu\sigma_2 q(1, y_1, y_2-1) = (\mu_1 + \mu_2) q(0, y_1, y_2) \quad , \\ & \quad y_1 + y_2 = n \end{aligned} \quad (22)$$

(f) $x = 0, y_1 = 0$ のとき ($y_2 = 0$ の場合も同様)

$$\mu\sigma_2 q(1, 0, n-1) = \mu_2 q(0, 0, n) \quad (23)$$

方程式 (18) ~ (23) 式を満す次の関係式を得る。

$$\begin{aligned} \textcircled{1} \quad q(x, y_1, y_2) &= \frac{\mu_1}{2\mu\sigma_1} q(x-1, y_1+1, y_2) \\ &= \frac{\mu_2}{2\mu\sigma_2} q(x-1, y_1, y_2+1) \end{aligned}$$

$$x + y_1 + y_2 = n, \quad 2 \leq x \leq n, \quad 0 \leq y_1, y_2 < n \quad (24)$$

$$\begin{aligned} \textcircled{2} \quad q(1, y_1, y_2) &= \frac{\mu_1}{\mu\sigma_1} q(0, y_1+1, y_2) \\ &= \frac{\mu_2}{\mu\sigma_2} q(0, y_1, y_2+1) \end{aligned}$$

$$y_1 + y_2 = n - 1, 0 \leq y_1, y_2 < n \quad (25)$$

これらの関係式から $q(x, y_1, y_2)$ の特殊解として以下を得る。

$$q(x, y_1, y_2) = \left(\frac{2\mu\sigma_1}{\mu_1}\right)^{y_1} \left(\frac{2\mu\sigma_2}{\mu_2}\right)^{y_2} q(n, 0, 0) \quad (26)$$

$$x > 0, x + y_1 + y_2 = n$$

$$q(0, y_1, y_2) = \frac{1}{2} \left(\frac{2\mu\sigma_1}{\mu_1}\right)^{y_1} \left(\frac{2\mu\sigma_2}{\mu_2}\right)^{y_2} q(n, 0, 0) \quad (27)$$

$$x = 0, y_1 + y_2 = n$$

一方, $\sum_{y_1=0}^n \sum_{y_2=0}^{n-y_1} q(n-y_1-y_2, y_1, y_2) = 1$ から, (26), (27) 式を代入すると $q(n,$

$0, 0)$ を以下のようにして計算できる。

$$\begin{aligned} & \sum_{y_1=0}^n \sum_{y_2=0}^{n-y_1} q(n-y_1-y_2, y_1, y_2) \\ &= \sum_{y_1=0}^{n-1} \sum_{y_2=0}^{n-1-y_1} q(n-y_1-y_2, y_1, y_2) + \sum_{y_1=0}^n q(0, y_1, n-y_1) \\ &= \sum_{y_1=0}^{n-1} \sum_{y_2=0}^{n-1-y_1} \rho_1^{y_1} \rho_2^{y_2} q(n, 0, 0) + \frac{1}{2} \sum_{y_1=0}^n \rho_1^{y_1} \rho_2^{n-y_1} q(n, 0, 0) \\ &= \left\{ \sum_{y_1=0}^{n-1} \rho_1^{y_1} \left(\frac{1-\rho_2^{n-y_1}}{1-\rho_2} \right) + \frac{\rho_2^n}{2} \sum_{y_1=0}^n (\rho_1/\rho_2)^{y_1} \right\} q(n, 0, 0) \\ &= \left[\frac{1}{1-\rho_2} \sum_{y_1=0}^{n-1} \{ \rho_1^{y_1} - \rho_2^n (\rho_1/\rho_2)^{y_1} \} + \frac{\rho_2^n}{2} \sum_{y_1=0}^n (\rho_1/\rho_2)^{y_1} \right] q(n, 0, 0) \\ &= \left\{ \frac{1}{1-\rho_2} \left(\frac{1-\rho_1^n}{1-\rho_1} - \frac{\rho_2^n - \rho_1^n}{1 - \frac{\rho_1}{\rho_2}} \right) + \frac{1}{2} \frac{\rho_2^{n+1} - \rho_1^{n+1}}{\rho_2 - \rho_1} \right\} q(n, 0, 0) \\ &= \left\{ \frac{1}{(1-\rho_1)(1-\rho_2)} + \frac{-2(\rho_2 - \rho_1) + 2\rho_2(1-\rho_1) - \rho_1(1-\rho_1)(1-\rho_2)}{2(1-\rho_1)(1-\rho_2)(\rho_2 - \rho_1)} \rho_1^n \right. \\ & \quad \left. - \frac{2\rho_2 - \rho_2(1-\rho_2)}{2(1-\rho_2)(\rho_2 - \rho_1)} \rho_2^n \right\} q(n, 0, 0) \\ &= \left\{ \frac{1}{(1-\rho_1)(1-\rho_2)} + \frac{1}{2(\rho_2 - \rho_1)} \left(\frac{1+\rho_1}{1-\rho_1} \rho_1^{n+1} - \frac{1+\rho_2}{1-\rho_2} \rho_2^{n+1} \right) \right\} q(n, 0, 0) \quad (28) \end{aligned}$$

ゆえに

$$q(n, 0, 0) = \left\{ \frac{1}{(1-\rho_1)(1-\rho_2)} + \frac{1}{2(\rho_2-\rho_1)} \left(\frac{1+\rho_1}{1-\rho_1} \rho_1^{n+1} - \frac{1+\rho_2}{1-\rho_2} \rho_2^{n+1} \right) \right\}^{-1}$$

$$\text{ここで } \rho_1 = \frac{2\mu\sigma_1}{\mu_1}, \quad \rho_2 = \frac{2\mu\sigma_2}{\mu_2} \quad (29)$$

ここで得た $q(x, y_1, y_2)$ は $x + y_1 + y_2 = n$ の場合の条件付確率である。 $q(n, 0, 0)$ は $\rho_1 < 1, \rho_2 < 1$ のときは $(1-\rho_1)(1-\rho_2)$ に近づく。

3.2 実行率 $\mu(n)$ の計算

多重度が n の場合の平均実行率、つまり単位時間あたりのジョブの終了率 $\mu(n)$ を計算する。単位時間あたりシステムが実行するジョブの数 $\mu_T(n)$ は以下のようにして算定できる。

$$\mu_T(n) = u_{HSC}(n) \mu_1 \rho \quad (30)$$

ここで $u_{HSC}(n)$ は HSC が使用中となる確率であり (34) 式で計算される。 ρ は HSC 使用後系を出て行く確率である。つまり $\mu_T(n)$ は、実行多重度が n の場合のシステム全体の実行時のスループットである。図 1 のモデルの考え方は、実行時は n 台の計算機が同時動作をするということなので、1 台あたりの実行率は、システム全体の $1/n$ となる。

$$\mu(n) = \mu_T(n) / n \quad (31)$$

ジョブの平均実行経過時間 $e(n)$ は $\mu(n)$ の逆数であるから、

$$e(n) = 1 / \mu(n) \quad (32)$$

平均 CPU 使用率、チャネル使用率 $u_{CPU}(n)$ 、 $u_{HSC}(n)$ 、 $u_{BMC}(n)$ は以下のようにして計算できる。

$$u_{CPU}(n) = 1 - \sum_{y_1=0}^n q(0, y_1, n-y_1) + \sum_{y_1=0}^{n-1} q(1, y_1, n-y_1) \times 0.5 \quad (33)$$

$$u_{HSC}(n) = 1 - \sum_{x=0}^n q(x, 0, n-x) \quad (34)$$

$$u_{BMC}(n) = 1 - \sum_{x=0}^n q(x, n-x, 0) \quad (35)$$

装置間の同時使用率も同様にして求められる。

4 実行経過時間の分布に関する考察

ここでは、実行時モデルと全体モデルを結ぶ $\mu(n)$ の算定方式に矛盾がないことを明確にするために、実行経過時間の分布に関する考察を行う。

全体モデルでは、実行時は、そのサービス時間が $\mu(n)$ をパラメータとする指数分布にしたがう n 個の計算機で処理を行っているとは仮定している。一方、実行時モデルでは、平均 m 回のパスのくり返しによりジョブが実行されるという仮定から、実行経過時間が幾何分布に従うものとして取扱っている。全体モデルと実行モデルの間の一致点は次の観点から見出し得る。

全体モデルにおける実行経過時間、いわゆるサービス分布は

$$f(x) = 1 - e^{-\mu(n)x} \quad (36)$$

により与えられるものとしている。

実行時モデル(図3)の解析から、1回のパスに対してジョブが系から出ていく確率は $\sigma_1\rho$ であり、再びCPU待行列にもどる確率は $(1-\sigma_1\rho)$ である。したがってジョブが k 回のパスで終了する確率 P_k は幾何分布となり、

$$P_k = \sigma_1\rho(1-\sigma_1\rho)^{k-1} \quad (37)$$

で与えられる。ここで σ_1 はHSCを使用する率、 ρ はHSC使用后系から出ていく率である。 m をジョブ当りの平均パス数とすると次の関係が成立する。

$$\sigma_1\rho = \frac{1}{m} \quad (38)$$

$$P_k = \frac{1}{m} \left(1 - \frac{1}{m}\right)^{k-1} \quad (39)$$

実行多重度が n のときの1パス当りの経過時間の平均を $w(n)$ とすれば、 P_k はジョブの実行経過時間が $kw(n)$ となる確率である。したがってジョブの実行経過時間が x 以下となる確率は、 $x = kw(n)$ とおくと、

$$\begin{aligned} g(x) &= \frac{1}{m} + \frac{1}{m} \left(1 - \frac{1}{m}\right) + \dots + \frac{1}{m} \left(1 - \frac{1}{m}\right)^{\left\{\frac{x}{w(n)} - 1\right\}} \\ &= 1 - \left(1 - \frac{1}{m}\right)^{\frac{x}{w(n)}} \end{aligned} \quad (40)$$

一方、平均のジョブの実行率は、ジョブ実行率 $\mu(n)$ の逆数であるから、

$$\mu(n) = 1/mw(n) \quad \text{より} \quad \frac{1}{w(n)} = m\mu(n)$$

したがって次式を得る。

$$g(x) = 1 - \left(1 - \frac{1}{m}\right)^{m\mu(n)x} \quad (41)$$

$$\lim_{m \rightarrow \infty} \left(1 - \frac{1}{m}\right)^m = e^{-1}$$

より、 m が十分大きい場合は

$$g(x) \approx 1 - e^{-\mu(n)x} = f(x) \quad (42)$$

m は平均の入出力の総数だから、現実には 10^3 程度の値を取る。したがって実行経過時間の分布を指数分布であると考えることができる。

一方、実行時モデルから平均パス数を計算すると m となることがわかる。

$$\begin{aligned} \sum_{k=1}^{\infty} k P_k &= \sum_{k=1}^{\infty} \frac{k}{m} \left(1 - \frac{1}{m}\right)^{k-1} \\ &= \frac{1}{m} \sum_{k=1}^{\infty} k \left(1 - \frac{1}{m}\right)^{k-1} \\ &= m f'(m) \end{aligned} \quad (43)$$

ここで、

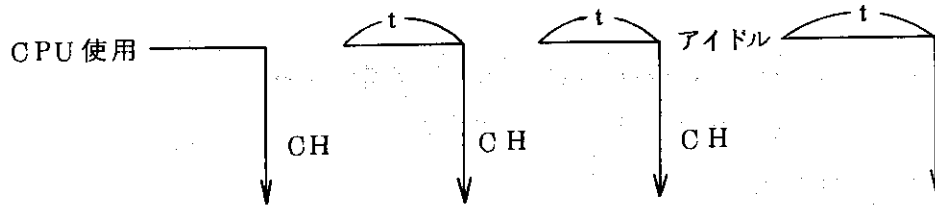
$$\begin{aligned} f(m) &= \sum_{k=0}^{\infty} \left(1 - \frac{1}{m}\right)^k = m \\ f'(m) &= \sum_{k=1}^{\infty} k \left(1 - \frac{1}{m}\right)^{k-1} \frac{1}{m^2} = 1 \end{aligned}$$

5 必要とするデータとその収集方法

(1) 1パス当りの平均のCPU使用率(μ)

測定時間(T)内の平均CPU使用率 U_{CPU} を測定装置マイクロサムを使用して測定する。

各チャンネルのアクセス回数 A_1, A_2 をユニバーサル・スケータでカウントする(Start I/O 命令が出された回数をカウントする)



チャンネルを呼出す時間間隔の平均 t がわかれば $\mu = \frac{1}{t}$ により μ が算定できる。

$$t = \frac{2TU_{CPU}}{A_1 + A_2}, \quad 0 \leq U_{CPU} \leq 1 \quad (44)$$

両CPUの使用比率を仮に U_{CPU0} と U_{CPU1} とする(ほぼ同じ値を取る)。

$$U_{CPU} = \frac{1}{2} (U_{CPU0} + U_{CPU1})$$

$$\mu = \frac{A_1 + A_2}{2TU_{CPU}}, \quad \sigma_1 = \frac{A_1}{A_1 + A_2}, \quad \sigma_2 = \frac{A_2}{A_1 + A_2}$$

$$2\mu\sigma_1 = \frac{A_1}{T \cdot U_{CPU}}, \quad 2\mu\sigma_2 = \frac{A_2}{T \cdot U_{CPU}} \quad (45)$$

(2) 各チャンネルの使用比(σ_1, σ_2)

$$\sigma_1 = \frac{A_1}{A_1 + A_2}, \quad \sigma_2 = \frac{A_2}{A_1 + A_2} \quad (46)$$

(3) 1パス当りの各チャンネル使用率(μ_1, μ_2)

平均チャンネル使用時間の逆数として求める。平均の各チャンネルの使用時間 t_i は、測定時間 T 内の平均の使用率 U_{HSC}, U_{BMC} をマイクロサムを使用して測定し、平均CPU継続使用时间 t を求めたのと同じ方法で算定できる。

$$t_1 = \frac{T \cdot U_{HSC}}{A_1}$$

$$t_2 = \frac{T \cdot U_{BMC}}{A_2}$$

$$\mu_1 = \frac{1}{t_1} = \frac{A_1}{T \cdot U_{HSC}}$$

$$\mu_2 = \frac{1}{t_2} = \frac{A_2}{T \cdot U_{BMC}} \quad (47)$$

(4) ジョブ当りの平均パスの数 (m)

- ① 会計情報よりジョブあたりのユーザCPU使用時間 J_{CPU} の平均値 $M_{CPU} = \bar{J}_{CPU}$ を得る。
- ② 測定時間 T 内のユーザのCPU使用時間の総数 $T_{CPU} = \sum_{JOB} J_{CPU}$ を計算する。
- ③ システムオーバヘッド率 S を次式から計算する。

全CPU使用時間 = ユーザ使用時間 (1 + S)

$$T \cdot U_{CPU} = \frac{1}{2} T_{CPU} (1 + S), \quad S = \frac{2TU_{CPU}}{T_{CPU}} - 1 \quad (48)$$

- ④ 1ジョブあたりの平均パスの数 m を次のようにして算定する。

$m =$ ユーザCPU使用時のパス数 (1 + S)

$$= \frac{M_{CPU} (1 + S)}{t} \quad (49)$$

(5) 出力率 (ρ)

1ジョブあたりの平均のパスの数が m であるから、

$$\frac{1}{m} = \frac{A_1 \rho}{A_1 + A_2} \quad \text{より} \quad \rho = \frac{A_1 + A_2}{A_1 m}$$

$$\text{ゆえに} \quad \rho = \frac{1}{m \sigma_1} \quad (50)$$

(6) ジョブ入力率 (λ)

会計情報からジョブ入力時間間隔の平均値を求めその逆数をとる。

(7) ラインプリンタ出力率 (μ_2)

ジョブ実行終了から出力終了までの時間の平均値を求め、その逆数を取る。このデータにはラインプリンタがすべて使用中の場合の待時間も含まれていることを考慮しなければならない。

(8) 多重度が n のときの平均のジョブの実行率 $\mu(n)$, $n = 1, 2, \dots, N$

実行時モデル (図3) の解析結果による。

(9) 出力装置台数 (L), 指定多重度 (N)

自明である。

以上(1)~(9)には、原研システムBに対して、現在収集可能な方法と算定式を示した。この結果、以後の計算に対して使用した数値を表2で示す。

チャンネルの使用回数 (A_1, A_2) と平均使用時間 ($1/\mu_1, 1/\mu_2$) はF75用効率測定用モニタPANSYにより、より正確な値が取れる。PANSYを用いてチャンネル使用状況のカウンタを取る機能がメーカ側から提供されるのは51年度末頃の予定である。

表2 基本データ(システムB)

項 目	値	記 号	根 拠
平均入力間隔	45秒	$1/\lambda$	会計情報の1ヶ月分の平均(9.00~17.00)
平均出力時間	100秒	$1/\mu_z$	同 上
ラインプリンタ台数	5	L	
マイクロサムによる測定時間	24540秒 (1975.910)	T	
チャンネル使用数			
HSC	1500000	A_1	推定(オーバーヘッドを含む)
BMC	525000	A_2	会計情報よりマイクロサムによる測定時間内のアクセス数
CPU使用率	0.89	U_{CPU}	マイクロサムによる測定(両CPUの平均)
HSC使用率	0.60	U_{HSC}	同 上(2つのHSCの和)
BMC使用率	0.28	U_{BMC}	同 上(2つのBMCの和)
ユーザCPU使用時間	25000秒	T_{CPU}	会計情報(T内)
ジョブ数	387	N_{Job}	同 上
入力率	0.0222	λ	
LP出力率	0.0100	μ_z	
平均ユーザCPU使用時間	64.6秒/ジョブ	M_{CPU}	計 算
実行時のCPU処理率	46タスク/秒	μ	計 算
実行時のHSC処理率	100タスク/秒	μ_1	計 算
実行時のBMC処理率	76タスク/秒	μ_2	計 算
平均CPU継続使用時間	0.022秒	$1/\mu$	計 算
平均HSC使用時間	0.01秒	$1/\mu_1$	計 算
平均BMC使用時間	0.013秒	$1/\mu_2$	計 算
OSオーバーヘッド	0.3		推 定
パス当りの出力率	0.00035	ρ	計 算
HSC使用比	0.74	σ_1	計 算
BMC使用比	0.26	σ_2	計 算
ジョブ当りの平均パスの数	3893	m	計 算

6 解析結果

いくつかの解析結果を多重度の関数として与え、F230-75システムBに対して数値計算結果を与える。ここで使用する基本データは表2のとおりである。

(1) 平均ジョブ実行経過時間 (図4)

(31), (32)式で与えた $\mu(n)$, $e(n)$ を計算する。 $e(n)/n$ は、 n がある値より大きいときはほぼ一定値をとる。これはHSCの使用率が增加しないことによる(**)。

(2) 平均ターンアラウンド・タイム (図5)

オープンバッチ・システムのターンアラウンド・タイムは、ジョブを入力してから出力終了までの時間であると定義できる。指定多重度が N のときのジョブの平均ターンアラウンド・タイム $w(N)$ の算式を示す。

ターンアラウンド・タイムは次の三項目の和である。

① 実行待行列上での待時間

$$\begin{cases} 0 & , n < N \\ (x+1)/N\mu(N) & , n = N \end{cases}$$

② 実行時間 $1/\mu(n)$ ③ 出力待および出力時間 $1/\mu_z + \max(0, z-L)/(L\mu_z)$

$$\begin{aligned} w(N) = & \sum_{n=0}^{\infty} \sum_{z=0}^{N-1} \left\{ 1/\mu(n) + \max(0, z-L)/(L\mu_z) + 1/\mu_z \right\} p(0, n, z) \\ & + \sum_{x=0}^{\infty} \sum_{z=0}^{\infty} \left\{ (x+N+1)/(N\mu(N)) + \max(0, z-L)/(L\mu_z) + 1/\mu_z \right\} \\ & p(x, N, z) \quad (51) \end{aligned}$$

図5では、 $\lambda=0.0222$ (ジョブ/45秒)と場合と $\lambda=0.0204$ (ジョブ/49秒)および $\lambda=0.0182$ (ジョブ/55秒)の場合について図示している。入力率が約一割減少することにより、ターンアラウンド・タイムはかなり減少することがわかる。

(3) スループット

スループットは、全体モデルのジョブの平均終了率を意味する。計算式を次に示す。

$$\begin{aligned} T(N) = & \sum_{z=0}^{\infty} \ell_z \mu_z \left\{ \sum_{n=0}^{N-1} p(0, n, z) + \sum_{x=0}^{\infty} p(x, N, z) \right\} \\ \ell_z = & \begin{cases} z & , z \leq L \\ L & , z > L \end{cases} \quad (52) \end{aligned}$$

スループットは、M/M/1モデルでは入力率に等しいことが知られている。この場合のようにモデルがある確率でM/M/1からM/M/Nの間を変化するときも $T(N)$ は次式のよう

に指定多重度 N に依存しない式として表わせる。(17)式の導出の過程から(注*)

$$T(N) = \sum_{z=1}^{\infty} \ell_z \mu_z \left(\prod_{i=1}^z \frac{\lambda}{\ell_i \mu_i} \right) A(N) p(0,0,0) \quad (53)$$

$A(N)$ は(17)式の定義と同一のものであり、これを代入すると、

$$T = \frac{\sum_{z=1}^{\infty} \ell_z \mu_z \left(\prod_{i=1}^z \frac{\lambda}{\ell_i \mu_i} \right)}{1 + \sum_{z=1}^{\infty} \left(\prod_{i=1}^z \frac{\lambda}{\ell_i \mu_i} \right)} \quad (54)$$

しかし導入時の100ジョブベンチマークテスト等でスループットが多重度によって変わることは実際では有り得ることである。それは、到着分布、サービス分布が指数分布ではないという現実とのずれから来るものである。100ジョブテストの場合は、実行待行列がいつも空でない状態で、いわば指定多重度 N で平衡状態で運用されている場合でありこのモデルとは合致しない。

注(*)

$$\begin{aligned} & \sum_{n=0}^{N-1} p(0,n,z) + \sum_{x=0}^{\infty} p(x,N,z) \\ &= p(0,0,z) + \sum_{n=1}^{N-1} p(0,n,z) + \sum_{x=0}^{\infty} p(x,N,z) \\ &= \prod_{i=1}^z \frac{\lambda}{\ell_i \mu_i} \left\{ 1 + \sum_{n=1}^{N-1} \prod_{i=1}^n \frac{\lambda}{\ell_i \mu_i} + \prod_{i=1}^N \frac{\lambda}{\ell_i \mu_i} \sum_{x=0}^{\infty} \left\{ \frac{\lambda}{N \mu(N)} \right\}^x \right\} p(0,0,0) \\ &= \prod_{i=1}^z \frac{\lambda}{\ell_i \mu_i} A(N) p(0,0,0) \end{aligned}$$

(4) 平均実行多重度(表3)

$$M(N) = \sum_{z=0}^{\infty} \left\{ \sum_{n=0}^{N-1} n p(0,n,z) + N \sum_{x=0}^{\infty} p(x,N,z) \right\} \quad (55)$$

(5) 指定多重度で実行される確率(図6)

$$R(N) = \sum_{z=0}^{\infty} \sum_{x=0}^{\infty} p(x,N,z) \quad (56)$$

通常10多重で運用した場合に、実際に多重度10でシステムが動くのは0.6の確率であることが図6からわかる。

(6) 平均実行待行列長(図7)

$$I(N) = \sum_{z=0}^{\infty} \sum_{x=0}^{\infty} x \cdot p(x, N, z) \quad (57)$$

図7で見られるように、入力率 λ がジョブの処理能力 $N\mu(N)$ を越えた場合は、実行待行列が無限に大きくなる。これは定常的に入力率が処理能力を上まわる場合の話であって実際は、昼食時とか夕刻時のような時間的変動や、ユーザの自己規制によりジョブの入力をあきらめることにより λ が小さくなるのでこの図のようなことは起らない。入力率と処理能力の関係については表1を参照。

通常は実行待ジョブが10個位という結果を得ている。

(7) 装置の処理率 (図8)

$$U_{CPU}(N) = \sum_{z=0}^{\infty} \sum_{n=0}^{N-1} u_{CPU}(n) p(0, n, z) + u_{CPU}(N) R(N) \quad (58)$$

$$U_{HSC}(N) = \sum_{z=0}^{\infty} \sum_{n=0}^{N-1} u_{HSC}(n) p(0, n, z) + u_{HSC}(N) R(N) \quad (59)$$

$$U_{BMC}(N) = \sum_{z=0}^{\infty} \sum_{n=0}^{N-1} u_{BMC}(n) p(0, n, z) + u_{BMC}(N) R(N) \quad (60)$$

$N > N_0$ のとき (ここで N_0 は $\lambda / N\mu(N) > 1$ を満す最低の値であり表1で示される) ,

$U_{CPU}(N)$, $U_{HSC}(N)$, $U_{BMC}(N)$ は N に関係なく等しい値となっている (***)。ジョブ処理能力が入力率を下まわるときは、実行多重度に対して装置の使用率があまり変化しないことによるものと考えられる。

(8) 装置の同時使用率 (図9)

算定方法は、(33)~(35)式に準じる。図は、CPU、HSC、BMCの同時使用(並行動作)率の平均を全体を1としてその割合を区分したものである。

例えば、CPU・HSC・BMCとあるのはCPUとHSCとBMCが同時に動作しているという解釈をとる。

当然のことながら実行多重度が上ると同時動作の割合が大きくなり、実行多重度が下るとアイドルが多くなる。

(9) ラインプリンタ使用率 (図10, 11)

ラインプリンタの使用状況を、平均出力時間を100秒とした場合(図10)と150秒とした場合(図11)について図示している。

会計情報の9時から17時までのジョブに対する1ヶ月の出力時間の平均は111秒である。これはジョブの実行終了から出力終了までの時間を取っているため、ラインプリンタが使用中の場合の待時間も含まれている。したがって出力所要時間は約100秒と見てよい。しかしながら長い出力を要するジョブの偶然の重り合いを考慮して150秒の場合の計算結果も併記し、混雑のピーク時に備える。

結果的には、使用時間が100秒の場合は5台全部動く場合は0.1の確率であり、1台も動かない確率と等しい。

多重度が10でラインプリンタの台数がL台の場合に、出力待行列の長さがzとなる確率 $P_L(z)$ の計算式は次のとおりである。

$$P_L(z) = \sum_{n=0}^{N-1} p(0, n, z) + \sum_{x=0}^{\infty} p(x, N, z) \quad (61)$$

10) ラインプリンタ設置台数と出力待行列の長さ(図12)

ラインプリンタの設置台数を ℓ とした場合の出力待行列の長さの平均 $L(\ell)$ は次のように計算される。

$$L(\ell) = \sum_{z=1}^{\infty} \ell_z \left\{ \sum_{n=0}^{N-1} p(0, n, z) + \sum_{x=0}^{\infty} p(x, N, z) \right\} \quad (62)$$

図12は、平均出力時間が100秒の場合と150秒の場合におけるラインプリンタ設置台数と出力待行列の長さを示している。

これらの結果から、現在の負荷なら3台、少し負荷を上げて4台あれば、ラインプリンタ台数は十分であると言える。

11) CPUの速度が2倍あると仮定した場合の計算(図13, 14)

CPUの使用率が高いので、仮にCPUが2倍の速度を持つとした場合に、実行経過時間、ターンアラウンド・タイムがどのように短くなるかを計算し、図示したのが図13, 図14である。この計算にあたり実行時のタスク実行率 μ を2倍にし、確率 q, p の計算を行い、(31)式と(51)式で与えた式を使用した。CPU速度が2倍に上っても今までどおりのジョブの投入だと仮定すると、ほとんど即座に実行が終了し、実行待はほとんど起らない。平均の実行多重度は3.4となりシステムの遊びが大きくなると予想される。

注(**) HSCの使用率は(29), (34)式より

$$\begin{aligned} u_{HSC}(n) &= 1 - \sum_{x=0}^n q(x, 0, n-x) \\ &= 1 - \sum_{x=0}^n \rho_2^{n-x} q(n, 0, 0) \\ &= 1 - \frac{1 - \rho_2^{n+1}}{1 - \rho_2} q(n, 0, 0), \quad \rho_1 = \frac{2\mu\sigma_1}{\mu_1} \doteq 0.7, \quad \rho_2 = \frac{2\mu\sigma_2}{\mu_2} \doteq 0.3 \end{aligned}$$

注(***) 指定多重度がNの場合のHSCの平均使用率を例にとって示す。

(59)式より

$$U_{HSC}(N) = \sum_{z=0}^{\infty} \sum_{n=0}^{N-1} u_{HSC}(n) p(0, n, z) + u_{HSC}(N) R(N)$$

前注(**)より、 $n > n_0$ に対し $u_{HSC}(n) \doteq U_{HSC}(\text{一定})$

$$U_{HSC}(N) = \sum_{z=0}^{\infty} \sum_{n=0}^{n_0} u_{HSC}(n)$$

$$+ \sum_{z=0}^{\infty} \left\{ \sum_{n=n_0+1}^{N-1} u_{\text{HSC}} p(0, n, z) + u_{\text{HSC}} \sum_{x=0}^{\infty} p(x, N, z) \right\}$$

第2項に(12)~(16)式を代入する。

$$\begin{aligned} & \sum_{z=0}^{\infty} \left\{ \sum_{n=n_0+1}^{N-1} u_{\text{HSC}} p(0, n, z) + u_{\text{HSC}} \sum_{x=0}^{\infty} p(x, N, z) \right\} \\ &= \left(1 + \sum_{z=1}^{\infty} \prod_{i=1}^z \frac{\lambda}{\ell_i \mu_z} \right) U_{\text{HSC}} \left\{ \sum_{n=n_0+1}^{N-1} \prod_{i=1}^n \frac{\lambda}{i \mu(i)} + \sum_{x=0}^{\infty} \prod_{i=1}^N \frac{\lambda}{i \mu(i)} \left(\frac{\lambda}{N \mu(N)} \right)^x \right\} \\ & \quad i > n_0 \text{ のとき } \frac{\lambda}{i \mu(i)} \rightarrow \rho \text{ より [注(**)より],} \\ &= \left(1 + \sum_{z=1}^{\infty} \prod_{i=1}^z \frac{\lambda}{\ell_i \mu_z} \right) U_{\text{HSC}} \prod_{i=1}^{n_0} \frac{\lambda}{i \mu(i)} \left(\sum_{n=n_0+1}^{N-1} \rho^{n-n_0} + \rho^{N-n_0} \cdot \frac{1}{1-\rho} \right) \\ &= \left(1 + \sum_{z=1}^{\infty} \prod_{i=1}^z \frac{\lambda}{\ell_i \mu_z} \right) U_{\text{HSC}} \prod_{i=1}^{n_0} \frac{\lambda}{i \mu(i)} \left\{ \frac{\rho(1-\rho^{N-n_0-1}) + \rho^{N-n_0}}{1-\rho} \right\} \\ &= \left(1 + \sum_{z=1}^{\infty} \prod_{i=1}^z \frac{\lambda}{\ell_i \mu_z} \right) U_{\text{HSC}} \prod_{i=1}^{n_0} \frac{\lambda}{i \mu(i)} \frac{\rho}{1-\rho} \quad (\text{一定}) \end{aligned}$$

表 3 平均実行多重度

指定多重度	平均実行多重度
5	-
6	5.8
7	6.5
8	7.1
9	7.7
10	8.3
11	8.8
12	9.4

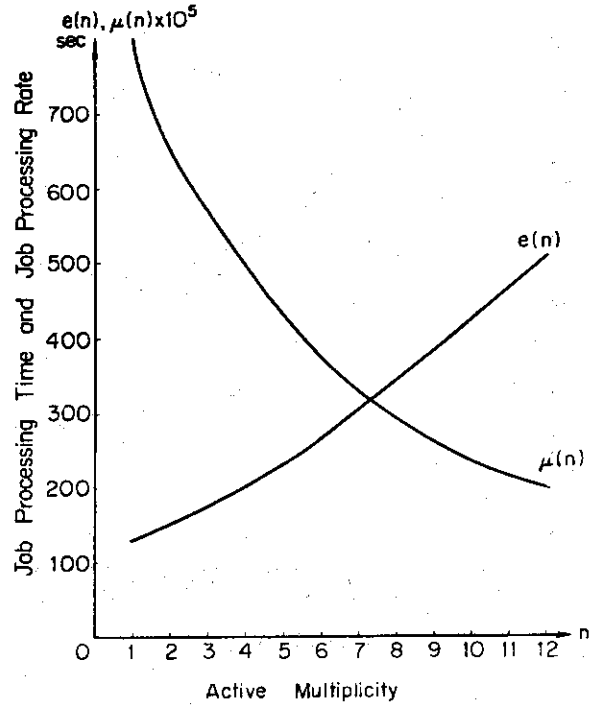


図4 平均ジョブ実行経過時間と実行率

Fig.4 Mean Job Processing Time and Processing Rate

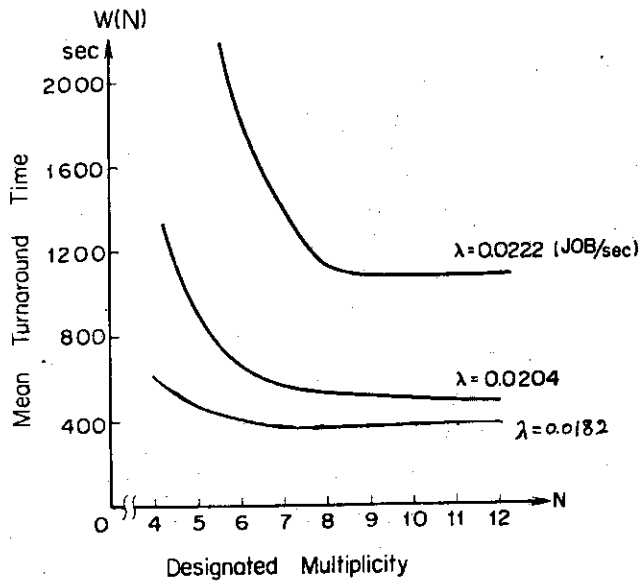


図5 平均ジョブターンアラウンドタイム

Fig.5 Mean Job Turnaround Time

注 定常的に入力率 (λ) がジョブの実行能力 ($N\mu(N)$) を越えると、実行時行列が無限大となり、ターンアラウンドタイムが無限に大きくなる。

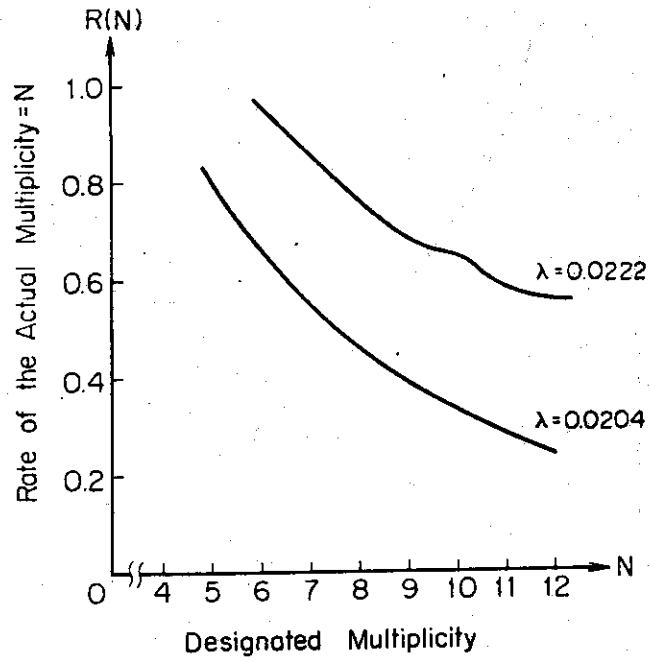


図6 実行多重度が指定多重度と一致する率
 Fig.6 Rate of the Actual Multiplicity=N

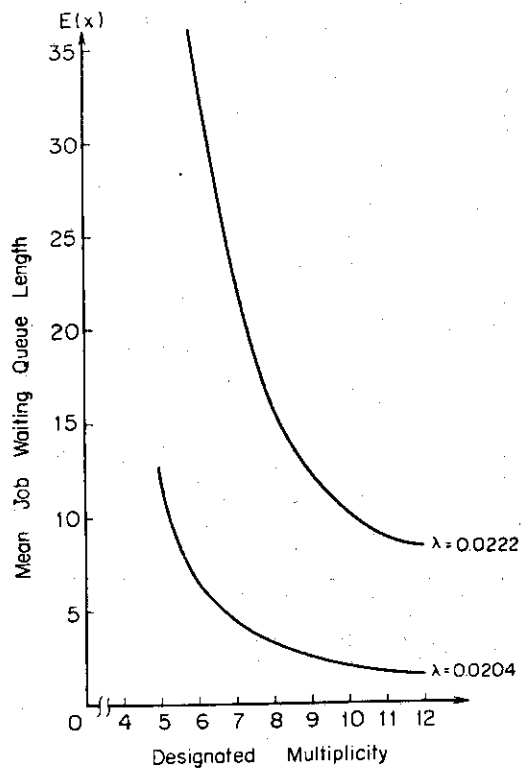


図7 平均実行待行列の長さ
 Fig.7 Mean Job Waiting Queue Length

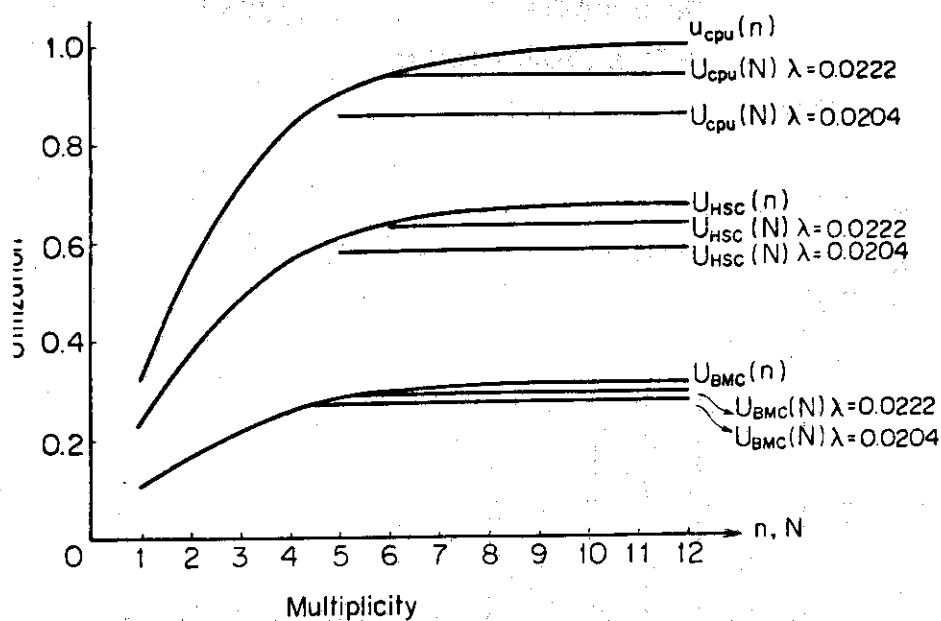


図8 CPU・チャネルの使用率
Fig.8 CPU and Channels Utilization

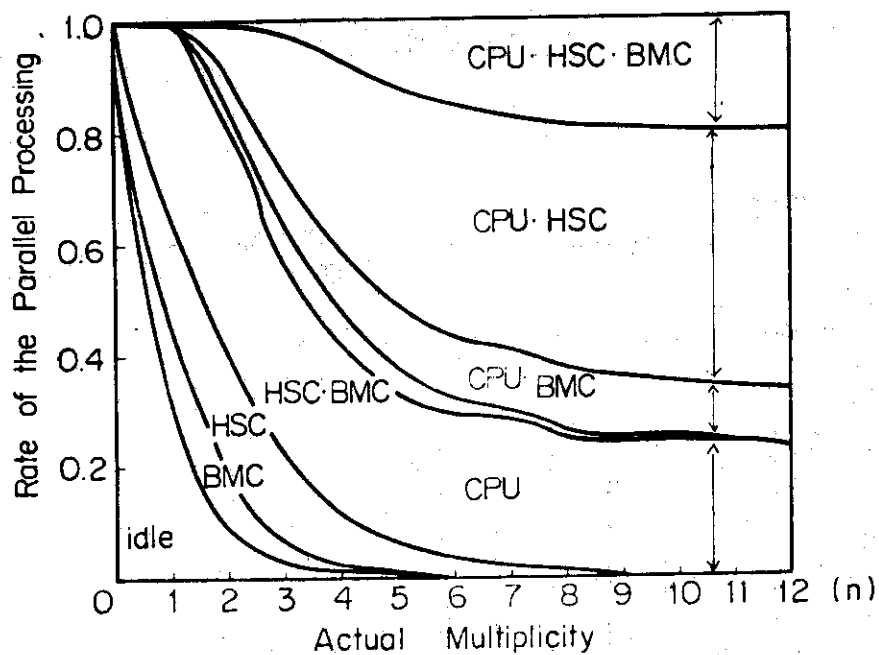


図9 各種装置の並行処理率
Fig.9 Rate of the Parallel Processing Between the Devices

指定多重度=10, ラインプリンタ台数=5
 平均出力時間=100秒

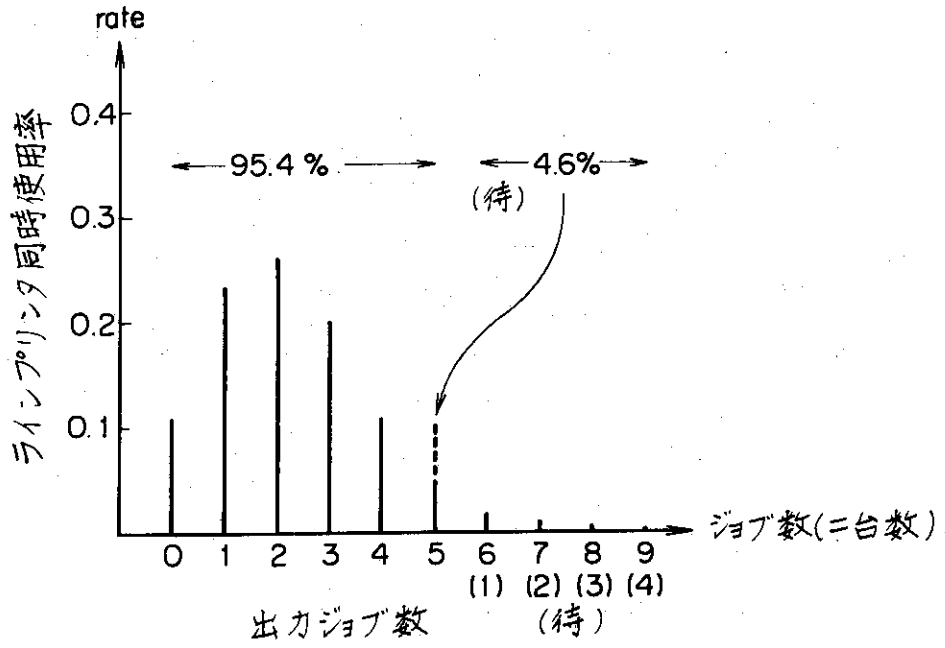


図10 ラインプリンタ同時使用率(システムB)

平均出力時間=150秒

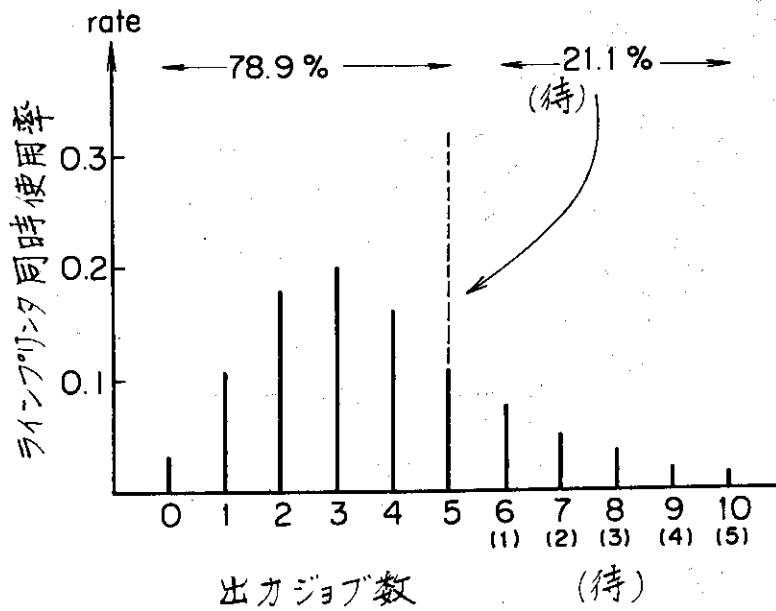


図11 ラインプリンタ同時使用率(システムB)

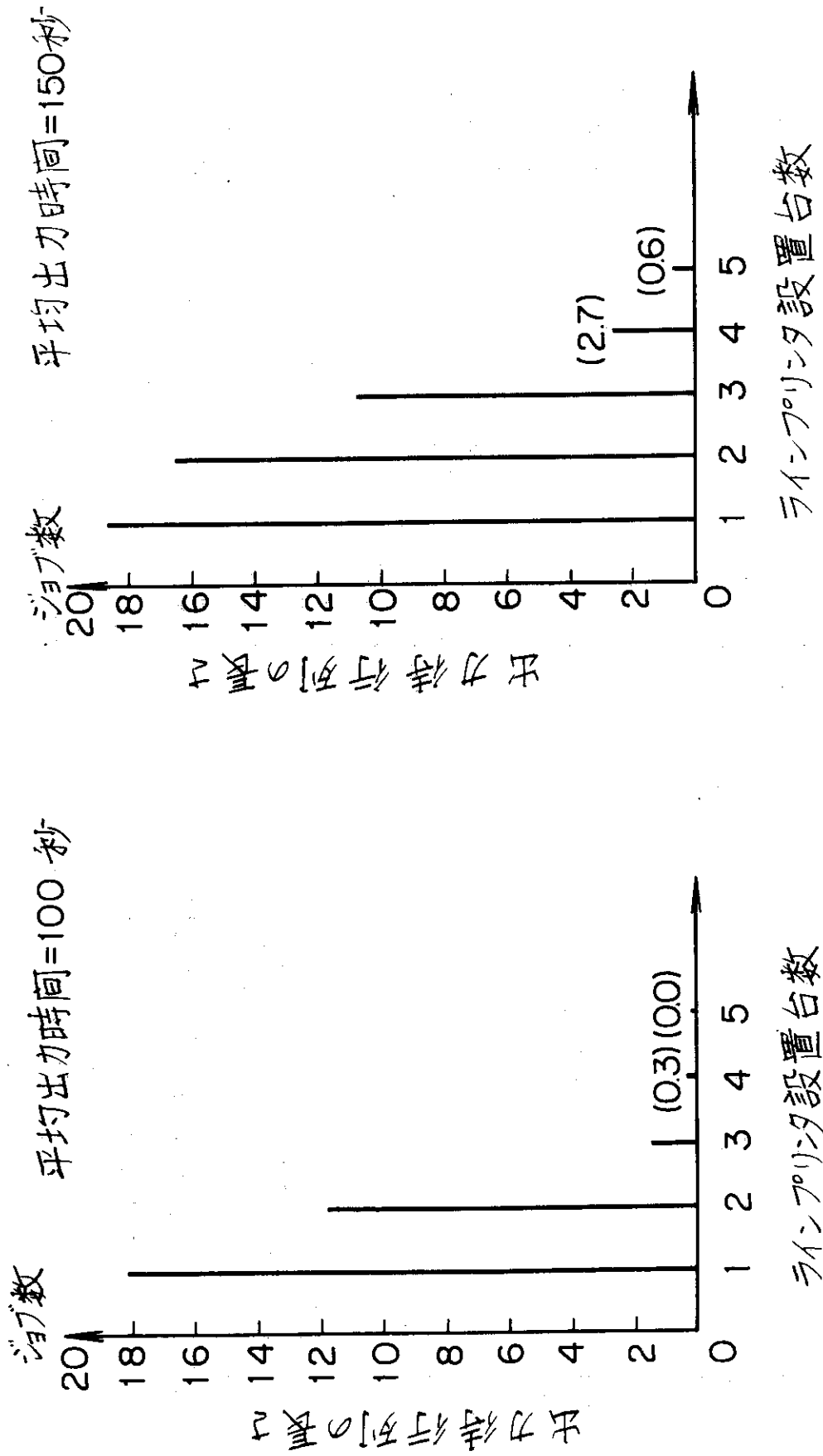


図12 ラインプリンタ設置台数と出力待ジョブ数

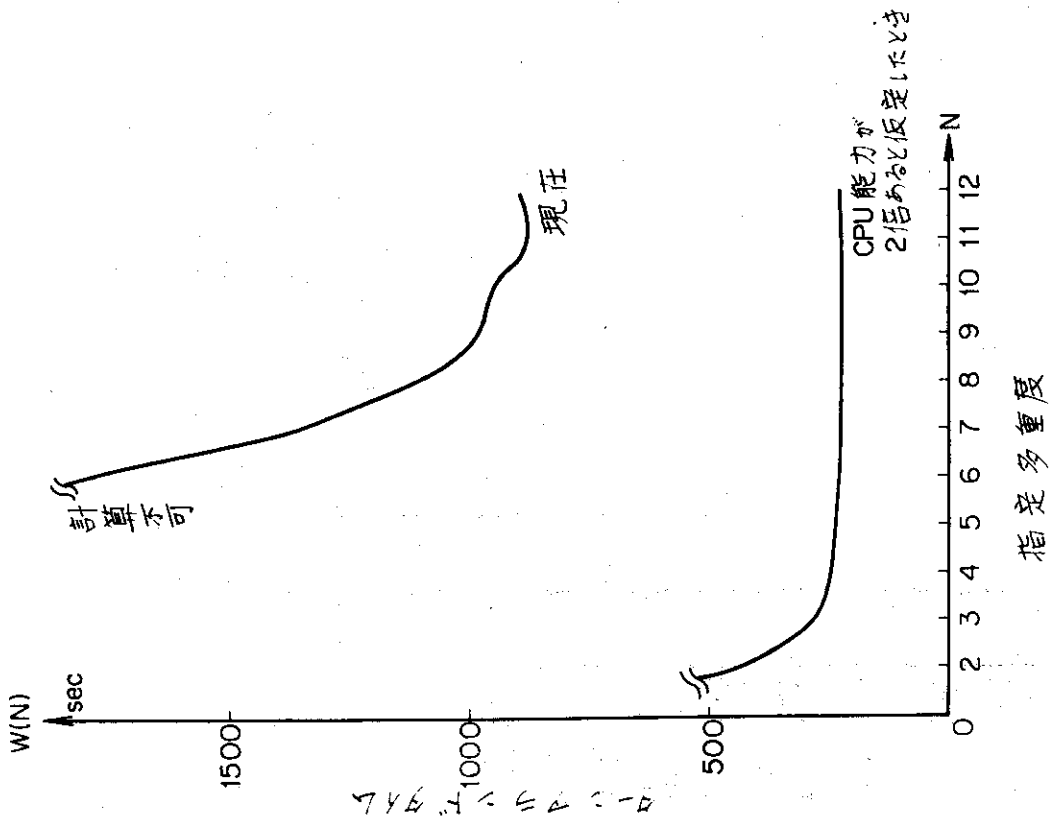


図14 CPUの能力が2倍となった場合のターンアラウンド
時間

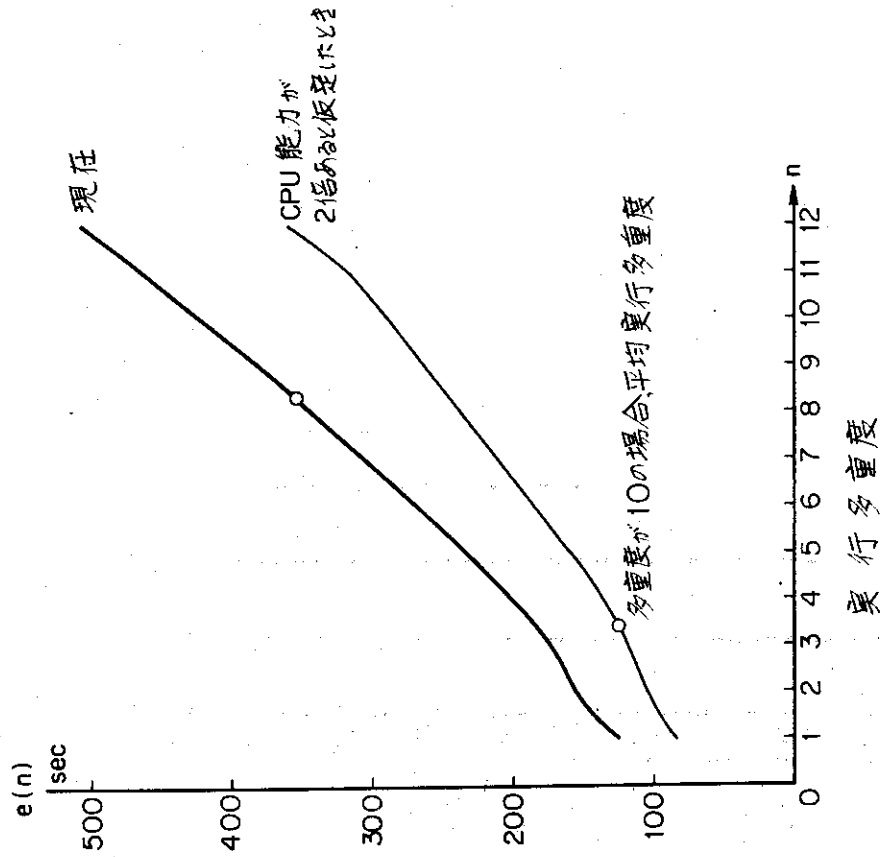


図13 CPU能力が2倍になった場合の実行経過時間

7 検 討

最後に、ここで解析した計算機モデルに現実のシステムの動きを十分に反映し得なかった事項を明記し、今後の発展への足がかりとしたい。

(1) OSのオーバヘッド(表4)

OSのオーバヘッドは次のように算定している。

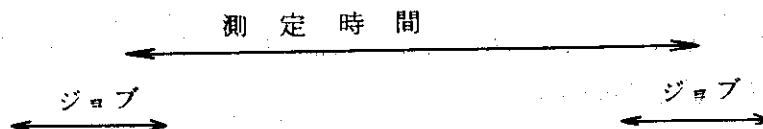
$$\text{測定時間} = \frac{\text{ユーザCPU使用時間}}{2} + \text{OSオーバヘッド} + \text{アイドル}$$

CPU使用率

これらのうち測定できる量は、CPU使用率(マイクロサムによる)、ユーザCPU使用時間(会計情報から)である。したがってOSのオーバヘッドは、マイクロサムによる測定時間(T)にCPU使用率をかけ、ユーザ使用時間の $\frac{1}{2}$ を引いたものとして算定できる。このようにして測定したOSのオーバヘッドは約15~40%となっている。〔6〕

一般にOSのオーバヘッドは多重度が上ると増加するものと予想されるが、原研システムにおける測定ではこれをはっきり立証づけるデータが得られなかった。よって測定時のユーザ使用時間との比の平均値(約40%)をや、下まわる30%をOSのオーバヘッドとした。これはOSのオーバヘッドのうち実行時の分は30%、残り10%は入出力時に使用すると推定した。

OSのオーバヘッドをハードウェア的に測定できることが望ましい。というのは、ユーザのCPU使用時間は、会計情報から取ると測定時間の両端にまたがるジョブのCPU使用時間が正確に取れないからである。



現在のところ、ジョブのCPU使用時間に対し、測定時間内の経過時間比をかけて補正しているが5~10%の誤差がでる。

(2) ジョブクラスと優先権(図15, 図16)

実行待行列からジョブを取出すためのジョブ優先権、実行時にCPU使用権を与えるための実行優先権、ジョブクラス等によるジョブまたはタスクの割当はこのモデルでは考慮していない。

ジョブの優先処理は図15に示すようなプライオリティ待行列モデルを作り、Priority Queueの理論により解析できる。

ジョブクラスについては、このモデルの考え方から拡張が容易であろう。図16に示すよう

な全体モデルを作り、次のように考える。

ジョブクラスA 短時間ジョブで unlimited クラス (つまり多重度に余裕があればそのクラスの制限を起して実行される。)

ジョブクラスB 中時間ジョブで Limited クラス

ジョブクラスJ 長時間ジョブで Limited クラス

それぞれの指定多重度を N_A, N_B, N_J とし実行多重度を n_A, n_B, n_J とすると次式を満す。

$$n_B \leq N_B, \quad n_J \leq N_J, \quad n_A \leq N - n_B - n_J$$

$$n = n_A + n_B + n_J, \quad N = N_A + N_B + N_J$$

これらの制限を取り入れて定常時の方程式を作り、解を求める。

現実のシステムに近いモデルを作るためには、これらを同時に取入れることになるが、常とう手段ではなくモジュール化し、解析を簡略化する必要があり、今後の課題となる。

(3) ラインプリンタ出力分布 (図17, 18)

このモデルでは、出力時間はパラメータ μ_z を持つ指数分布にしたがうと仮定してきた。現実のシステムでは図17, 18で示すように完全に指数分布と一致していない。出力時間が長いジョブは指数分布によるものより現実の方が少なめである。

(4) 実行経過時間の分布 (図19, 20)

実行経過時間の分布をパラメータ $\mu(n)$ を持つ指数分布であると仮定した。実際は図19, 20で示すように、指数分布よりはむしろ

$$f(x) = 1 - e^{-a\sqrt{x}}, \quad a \text{ は 常数}$$

の形に近い分布関数となっていることがわかる。但し実行多重度毎のデータではなく、一括したものである。CPUの使用時間の分布がこの形をなすことは〔5〕で報告されている。

指数分布に比して、短い時間で終了するジョブ数が多く、長い時間要するジョブ数が少い傾向にあることがわかる。

(5) 入力分布

入力分布は指数分布と良い一致が見られることが〔5〕で報告されている。しかしこれはオープンシステムに限って言えることである。

(6) ロールアウトされているジョブ

このモデルでは、主記憶装置(実記憶)にすべてのジョブが存在する場合のみを想定している。現実のシステムでは指定多重度が10の場合に0.5ジョブの割合、つまり9.5ジョブが主記憶上にあるという状況(これは、ソフトウェアモニタPANSYによる測定結果である)から、今回は考慮しなかった。

(7) タイムスライス (図21)

F230-75システムでは、バッチジョブに対するタイムスライスが0.1秒と大きく、その効果が少い。というのは、平均のCPU使用継続時間が表2に示すように0.022秒であるから、0.1秒もCPU使用が継続する確率はかなり小さい。実際はファイルへの入出力以外に、(ラインプリンタへの)出力や(カードからの)入力などがあるわけで、0.1秒もCPU使用

が継続するという事は 10^5 ステップ間入出力を行わないことに等しい。タイムスライスを考慮したモデルは図 21 で示される。実行時モデル (図 3) に対して CPU 使用後再び CPU 使用にもどる場合を追加している。その確率 σ_s は、次式で計算できる。

$$\sigma_s = e^{-\mu t_0}$$

ここで t_0 はタイムスライスとする。この式は CPU 使用時間が t_0 より大きい確率を出している。

(8) 実行多重度の変動

ジョブが実行開始後実行終了まで、実行多重度が開始時の値を保つものとして解析されている。この点はこのモデルから来る制約である。

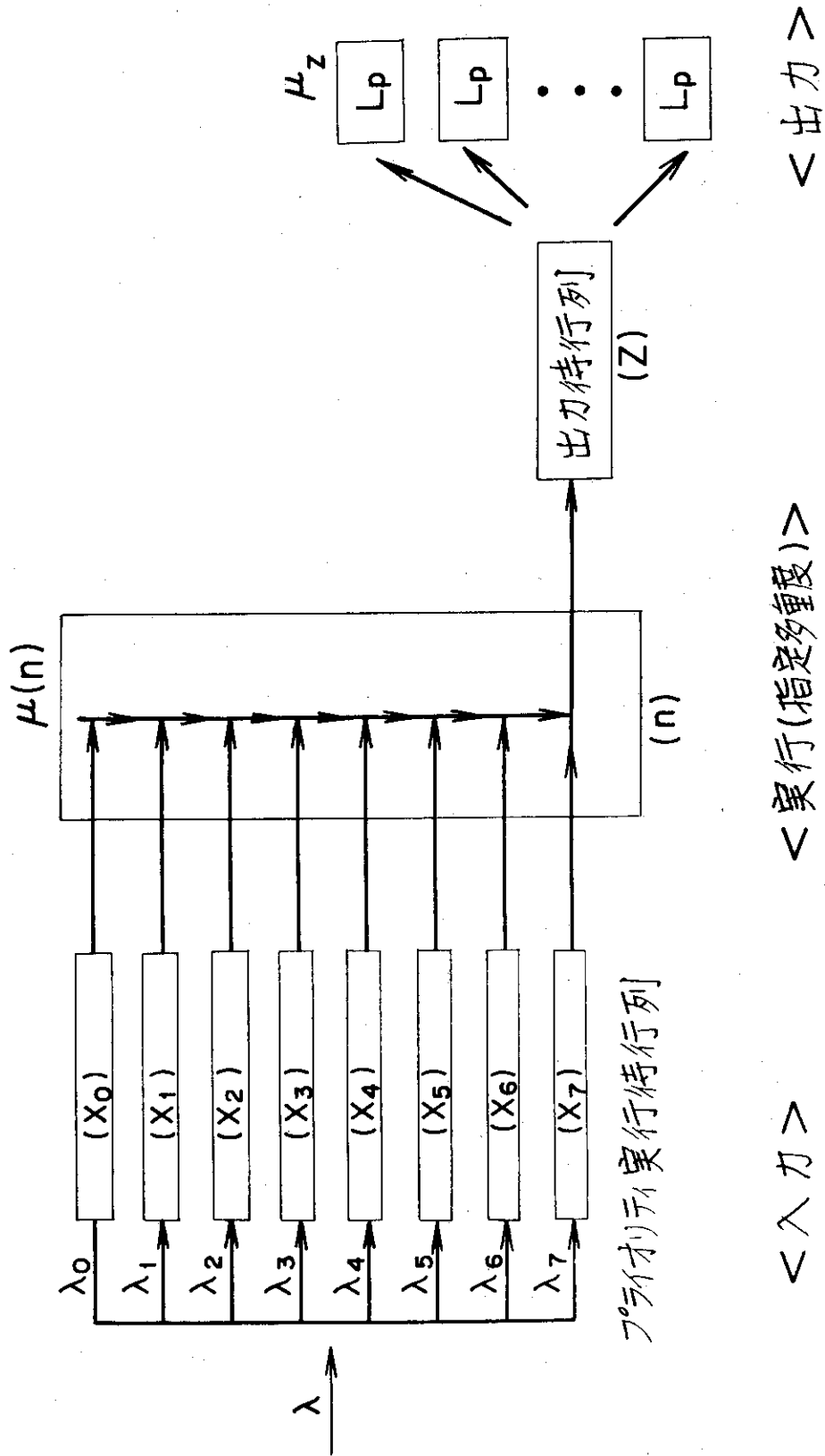
これを避けるためにはモデルを、サブモジュール化しないで、図 1 と図 3 を一体としたモデルを作り解析することが必要である。

表 4 CPU 使用状況 (百分比)

測定日時	ユーザ使用***	OS オーバヘッド	アイドル*	平均実行多重度**
1975. 8. 11 14.10~16.09 システム A	44.3%	11.0	44.7	3.4
1975. 8. 13 10.50~13.59 システム A	53.2	32.6	15.2	5.7
1975. 9. 10 10.02~16.51 システム A	63.8	26.0	10.2	8.9
1975. 9. 26 11.31~17.15 システム A	37.5	21.8	40.7	5.2
1975. 10. 28 12.00~16.00 システム B	71.5	16.8	11.7	7.6
1975. 10. 30 13.11~15.50 システム B	51.8	24.2	24.0	5.3

マイクロサム* , PANSY サブモニター** による測定および会計情報からの集計*** による。

但し、ユーザ使用率は数%の誤差を含む。



<実行(指定多重度)>

<入力>

<出力>

図15 F230-75プライオリティ待行列モデル

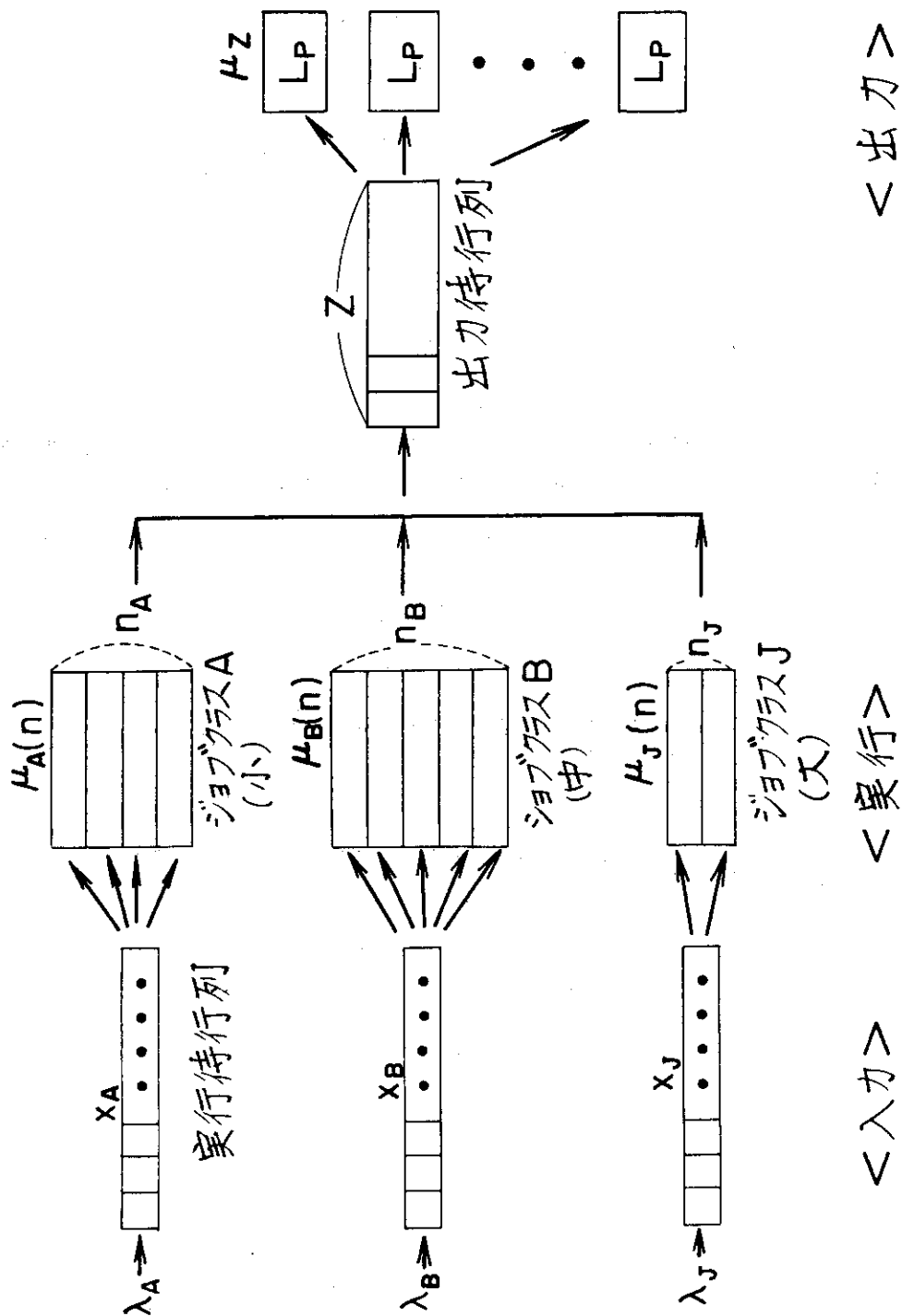
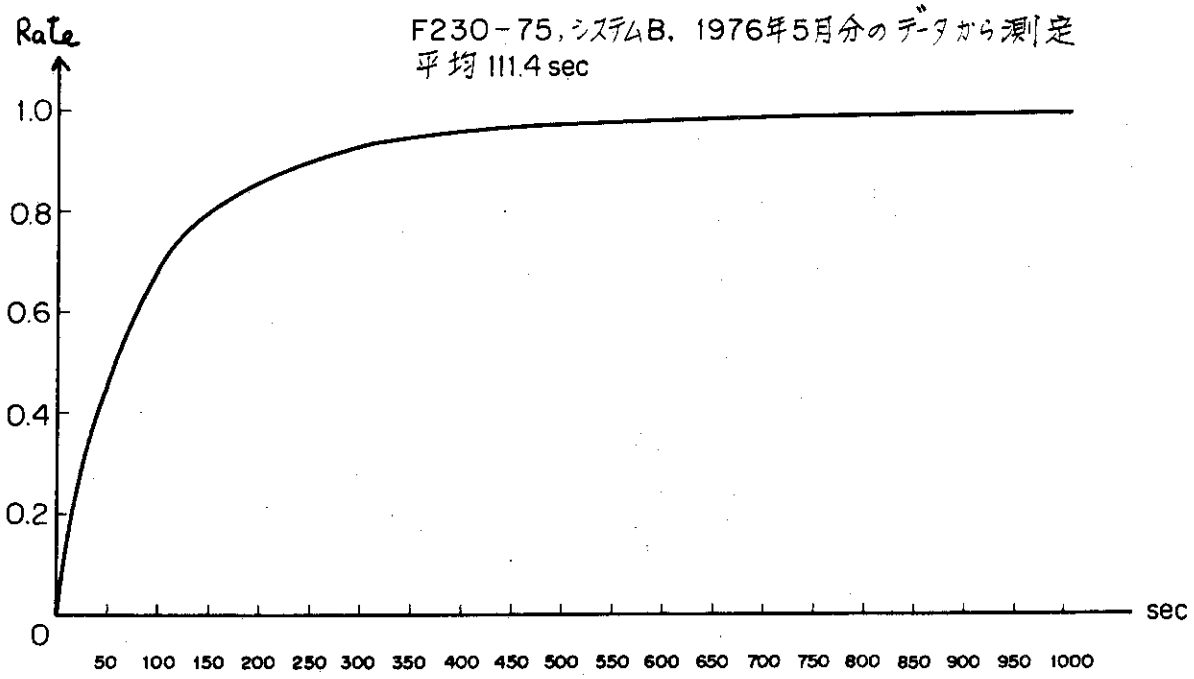
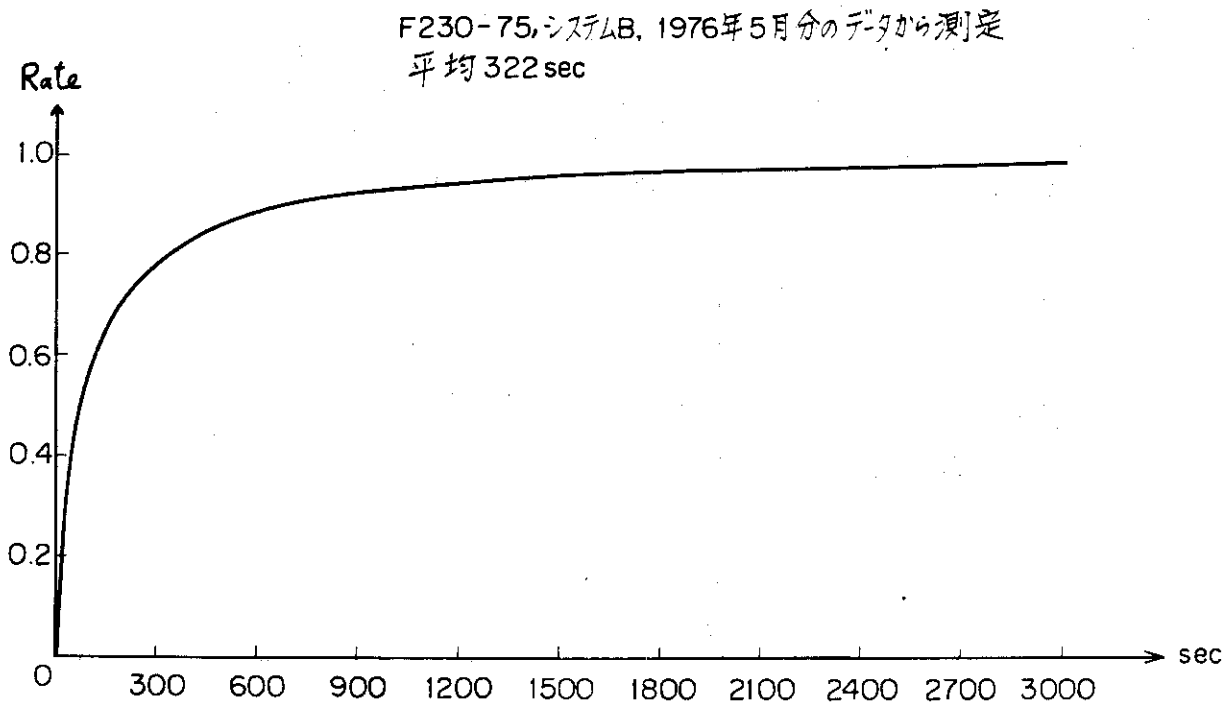


図16 F230-75 システムモデル (ジョブクラス導入)



出力時間

図17 出力時間の分布



実行経過時間

図19 実行経過時間の分布

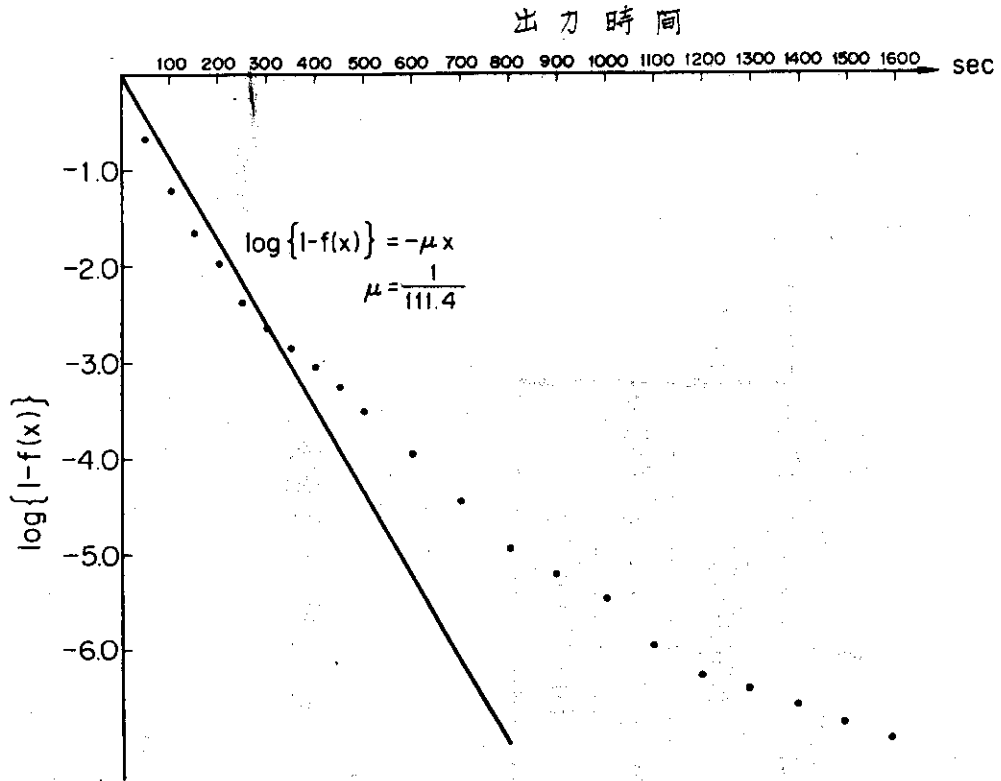


図 18 出力時間の分布, 実測値と指数分布の誤差

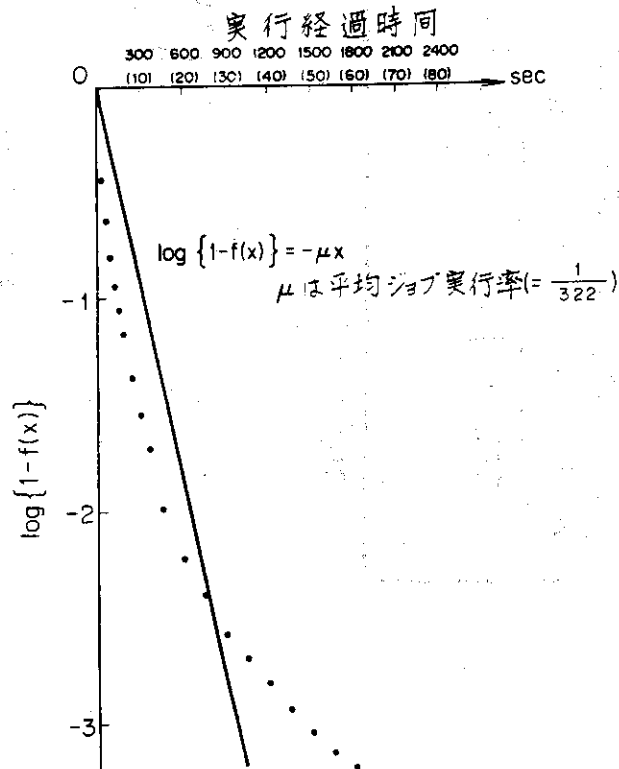
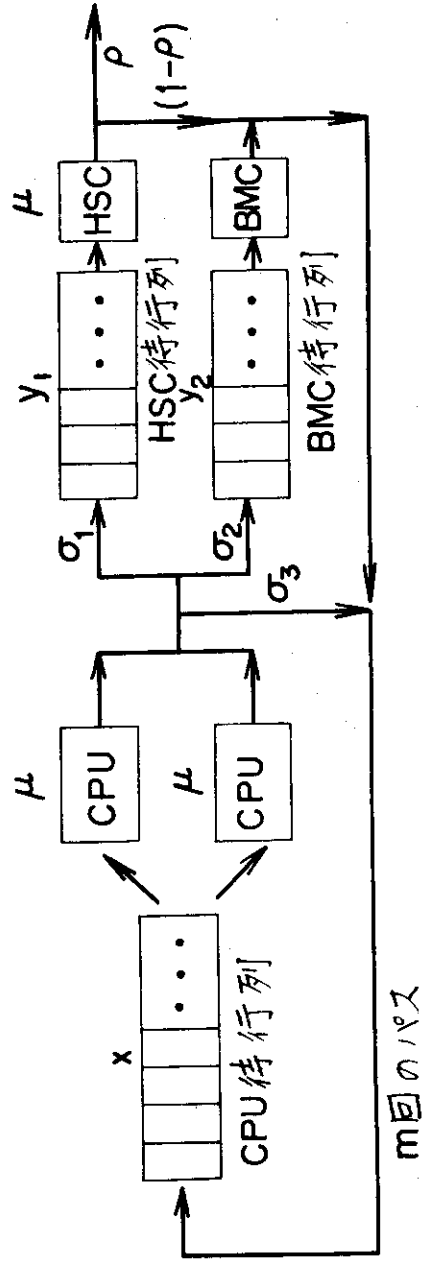


図 20 実行経過時間の分布, 実測値と指数分布の誤差



$$\sigma_1 + \sigma_2 + \sigma_3 = 1$$

$$x + y_1 + y_2 = n, \quad n \text{ は実行多重度}$$

図 21 タイムスライスを考慮した実行待モデル

謝 辞

この報告書を作成するにあたり，“Operating Systems Theory”のゼミの指導をしていただいている茨城大学の磯田先生をはじめ、計算センターの浅井清氏以下ゼミのメンバーに、結果の検討と有効な助言をいただきましたことを感謝します。

参 考 文 献

- [1] E.G. Coffman, J.R. and P.J. Denning : "Operating Systems Theory", Prentice-Hall, 1973.
- [2] I. adiri, M.Hofri and M. Yadin :
"a multiprogramming Queue", J.ACM. V20-4, Oct. 1973, p589~603.
- [3] 宮崎他 : "多重プログラミングシステムの効率について", 情報処理学会誌,
V16-12, Dec. 1975, p1040~1047.
- [4] 紀 一誠 : "資源切り出し型待行列モデルによるメモリ競合の解析", システム性能
評価研究会資料13, 1976年1月
- [5] 石黒美佐子 : "FACOM 230-60システムのCPUアイドルの解析",
JAERI-M 5657, 1974.
- [6] 石黒美佐子 : "導入過渡期におけるF230-75のシステム効率とジョブ数",
1976.

謝 辞

この報告書を作成するにあたり、"Operating Systems Theory"のゼミの指導をしていただいている茨城大学の磯田先生をはじめ、計算センターの浅井清氏以下ゼミのメンバーに、結果の検討と有効な助言をいただきましたことを感謝します。

参 考 文 献

- [1] E.G. Coffman, J.R. and P.J. Denning : "Operating Systems Theory", Prentice-Hall, 1973.
- [2] I. adiri, M.Hofri and M. Yadin :
"a multiprogramming Queue", J. ACM. V20-4, Oct. 1973, p589~603.
- [3] 宮崎他 : "多重プログラミングシステムの効率について", 情報処理学会誌,
V16-12, Dec. 1975, p1040~1047.
- [4] 紀 一誠 : "資源切り出し型待行列モデルによるメモリ競合の解析", システム性能
評価研究会資料13, 1976年1月
- [5] 石黒美佐子 : "FACOM 230-60システムのCPUアイドルの解析",
JAERI-M 5657, 1974.
- [6] 石黒美佐子 : "導入過渡期におけるF230-75のシステム効率とジョブ数",
1976.