数理解析研究所講究録
第 564 巻 1985 年 147-164

ブロッキングを伴う網型待ち行列モデルの数値解アルゴリズム

筑波大学 逆瀬川浩孝
Hirotaka Sakasegawa

1. はじめに

一人の客が任意複数箇所のサービスを受けるという現象を解析するための数学モデルとして、網型待ち行列モデルがある。閉型あるいは開型のネットワークの各ノードが一つの待ち行列モデルになっていて、客はパスに沿って与えられた規則で移動しながら各ノードの窓口でサービスを受ける、というものである。

この型のモデルの成功例は Jackson (1963) と Baskett et al. (1975) である。後者の云わゆる BCMP モデルは前者の一般化で、定常状態における各ノードの滞在客数の同時分布が積形式になることによって特徴付けられている。これは、ノード間の推移が一切の状況と独立に、確率的に行われることと、各ノードでのサービスは、そのノードに客がいれば、他のノードの様子に全く影響されない、という二つの基本的な仮定

に基づくものである。

後者の仮定は、各ノードの待ち合い室容量に制約がないことを意味するが、実際の現象では、制約のあるのが普通である。その結果、あるノードでのサービスは、他のノードの混雑具合によって影響を受け、サービスの効率を低下させることになる。サービスされるべき客がいるにも関らず、他のノードが混雑している等の理由で、窓口がサービスをしない（できない）状態を、その窓口はブロックされている、と言う。

ブロッキングを伴う待ち行列モデルとしては、ノードが線状に配置された直列型モデルについて、Hunt (1956) 以来、多くの解析がなされているが、一般の網型モデルについては、放置されたままになっている。網型モデルの最大の難点はシステムデッドロックと呼ばれる現象の取り扱いである。いくつかの窓口が同時にブロックされ、その状態を解除する手段が存在しない、という状態を（システム）デッドロックと呼ぶ。各ノードでのサービス時間が互いに独立であるような普通の網型モデルを考えた場合、閉じたパス（ループ）があって、そのループ上の各ノードの待ち合い室容量がすべて有限で、ループの外から客の到着がある、という条件が揃うと、デッドロック状態を避けることができないが、このような

条件は、ごく一般的なものである。デッドロックは、システムのいわば吸収状態であるから、興味のある結果は定常に達する前の過渡的状態の解析を通してでしか得られない。これが、ブロッキングを伴う網型モデルの解析が余りなされないことの最大の原因であろう。

この小論では、システムデッドロックを避けるために、ループを持たない網型モデルをとりあげ、その定常状態でのふるまいを解析する。例えば、図1のようなモデルである。待ち合い室容量が有限であるという点を除いて他はすべてBCMPモデルと同じ仮定をおいたとしても、このモデルの定常解は積形式解のような形にはならず、個々の同時確率を求めることによってしか得られない。したがってこれらを求めるために、大きな次元の連立方程式（平衡方程式）を数値的に解くことが必要になる。これは普通反復法によって解かれるが、次元が大きくなった時、かなり能率が悪くなるため、その収束を速めるためにさまざまな工夫が考案されている。ここでは条件付確率法とでも呼ばれる方法（Takahashi et al.(1976)）をとり入れ、収束の加速をはかっている。

以下、2節で、この小論で扱うモデルおよび記号の説明を行い、3節で、数値的アルゴリズムを提示する。4節では、若干の数値例を述べる。

2. モデル

Kケのノードからなるネットワークを考える。各ノードに単一窓口と、容量 N_k-1 の待ち合い室がある。客はパラメータ λ のポアソン過程にしたがってこの系に到着し、確率 p_{0k} で k ノードへ行く。もし、k ノードに N_k 人の客がすでに滞在していれば、新たな到着客は(何もサービスされないで、系から)退去する。これを、客の損失という。客に対する k ノードでのサービスは先着順で行われ、時間は平均 $1/\mu_k$ の指数分布に従う。サービス開始時点で、その客の行先ノードを確率分布 $\{p_{kj}\}_{j=0,1,\cdots,K}$ によって決定する。但し、ノード番号 0 は退去を表すものとする。サービス終了後、あらかじめ決められた番号のノードへ進む。

あるノードで、新たな客の到着によって待ち合い室が一杯

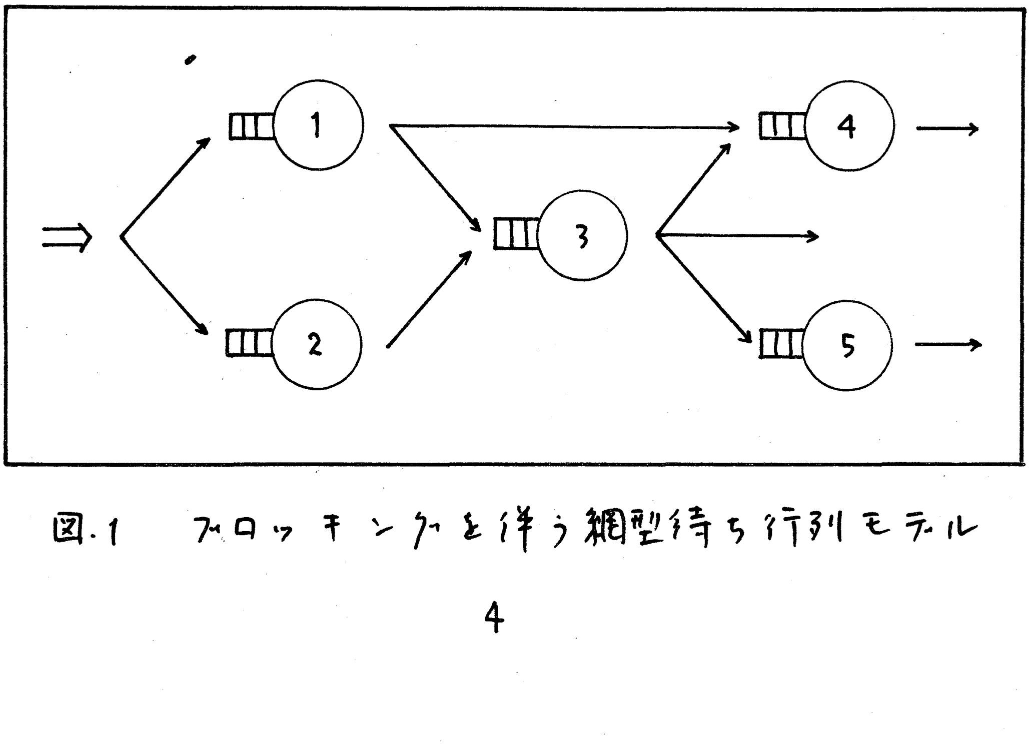

図.1　ブロッキングを伴う網型待ち行列モデル

になった時、そのノードを目指す総ての客のサービスは中断（停止）され、それらの窓口はブロックされる。ブロック状態は、そのノードのサービスが終了し、待ち合い室に一つの空きができるまで続く。

時刻 t における k ノードの客数、及びサービス中の客の行先ノード番号をそれぞれ $M_k(t)$, $T_k(t)$ で表す。定常状態におけるそれらの量をそれぞれ M_k, T_k で表す。

3. 解析

前節で定義されたモデルの定常状態でのふるまいを調べるのが目的である。モデルに含まれる確率要因のマルコフ性から、$(M_1(t), T_1(t), M_2(t), T_2(t), \cdots, M_K(t), T_K(t))$（$=(M_k(t), T_k(t))_k$ と書く）は非周期的マルコフ過程で、定常分布が存在し、その分布は、生成作用素を係数行列とする連立方程式の解として得られることがわかる。解は、積形式解のように陽には表せないので数値解法に頼らざるを得ない。そこで問題は、

(1) 与えられたパラメータ $(K, \lambda, (\mu_k), (N_k), (p_{jk}))$ から生成作用素 A をどのようにして生成するか、

(2) 大次元の連立方程式

(3.1) $\pi A = 0, \quad \pi \xi = 1$ （ξ はすべて1のベクトル）

をいかにして解くか.

の2つである。

先ず, 方程式の(行列 A の)次元を求める。このマルコフ過程の状態集合を $\mathcal{S}$ とし, $\mathcal{S}$ の部分集合 $\mathcal{S}_n$ ($n=0,1,\cdots,N$; 但し $N=\Sigma_k N_k$) を次で定義する。

$$\mathcal{S}_n = \{ (n_k, t_k)_k \in \mathcal{S} \ ; \ \Sigma_k n_k = n \}$$

$\mathcal{S}_n$ は系内に n 人の客がいる状態をすべて集めたもので, お互いに排反で, すべて集めると $\mathcal{S}$ になっている。$\mathcal{S}_n$ の要素の個数は次の関数 $g(\cdot,\cdot)$ を使って $g(K,n)$ と表される。

$$g(1,n) = \begin{cases} 1 & (n=0) \\ r_1 & (0<n \leqq N_1) \\ 0 & (\text{それ以外の } n) \end{cases}$$

$$g(k,n) = \begin{cases} g(k-1,n) + r_k \Sigma_{j=1}^{N_k} g(k-1,n-j) & \\ & (0 \leqq n \leqq \Sigma_{j=1}^{k} N_j) \\ 0 & (\text{それ以外の } n) \end{cases}$$

但し, r_k は $p_{kj} \neq 0$ となる j の個数.

したがって, 方程式の次元 $|\mathcal{S}|$ は, $g(K,0)+g(K,1)+\cdots+g(K,N)$ で与えられる。

次に $\mathcal{S}_n$ の要素に対し, 辞書的順序の意味で整列させる(通し番号を振る)関数 h_n を次のように定義する。

$$h_n : \mathcal{S}_n \longrightarrow \{0,1,2,\cdots, g(K,n)-1\}$$

$$h_n((n_k, t_k)_k) = \langle t_1 \rangle$$
$$+ \sum_{k=2}^{K} \left\{ \sum_{j=0}^{n_k-1} r_k(j)\, g(k-1, m_k-j) + \langle t_k \rangle\, g(k-1, m_{k-1}) \right\}$$

但し、$\langle t_k \rangle$ は $p_{kj} \neq 0$ となる j を小さい順に並べて 0 から $r_k - 1$ まで番号を付けた時の t_k の番号,

$r_k(j) = 1$ (if $j=0$), $= r_k$ (if $j>0$),

$m_k = n_1 + n_2 + \cdots + n_k$ とする。

h_n の逆関数、すなわち、S_n の辞書的順序の意味で m 番目の要素（状態ベクトル）を生成する関数は下のような手続きによって定義される。

$$h_n^{-1} : \{0, 1, 2, \cdots, g(K, n)-1\} \longrightarrow S_n$$

input x ; output $(n_k, t_k)_k$;

$m := x$; $l := n$; $k := K$;

repeat

if $m < g(k-1, l)$ then $\{ n_k := 0$; $t_k := 0 \}$

else $\{ n_k := \min_j \left(m - \sum_{i=0}^{j} r_j(i)\, g(k-1, l-i) < 0 \right)$;

$$\langle t_k \rangle := \left[\frac{m - \sum_{i=0}^{j-1} r_j(i)\, g(k-1, l-i)}{g(k-1, l-j)} \right]_{\text{Gauss}} ;$$

$m := m - \sum_{i=0}^{j-1} r_j(i)\, g(k-1, l-i) - \langle t_k \rangle\, g(k-1, l-j)$;

$l := l - n_k$; $k := k-1 \}$

until $k = 1$;

$n_1 := l$; $\langle t_1 \rangle := m$.

このように定義された h_n を使って S_n の要素を自然な順序で並べ、S 全体も S_n の自然な順序に従って並べておけば、生成作用素 A は次のような特徴を持つことになる。すなわち S_n に対応させて A を小行列に分解して考えると、小行列を要素とする行列の意味で三重対角：

$$A = \begin{bmatrix} P_0 & Q_0 & & & & \\ R_1 & P_1 & Q_1 & & 0 & \\ & R_2 & P_2 & Q_2 & & \\ & & \ddots & \ddots & \ddots & \\ & 0 & & R_{N-1} & P_{N-1} & Q_{N-1} \\ & & & & R_N & Q_N \end{bmatrix}$$

となり、特に主対角行列 P_n は上三角行列となる。A の非 0 要素は、例えば、$((n_k, t_k)_k)$ から $((n_k', t_k')_k)$ へ rate α で推移する可能性があるならば、

$$a_{jk} = \alpha$$

$$j = h_n((n_k, t_k)_k) + \sum_{\ell=0}^{n-1} g(K, \ell) \quad , n = \sum n_k$$

$$k = h_{n'}((n_k', t_k')_k) + \sum_{\ell=0}^{n'-1} g(K, \ell) \quad , n' = \sum n_k'$$

とすればよく、h_n^{-1} を使えば、A の非 0 要素を規則的に生成していくことができる。

次に、A を係数行列とする平衡方程式を数値的に効率的に解く方法を考える。このマルコフ過程の定常分布 π も、A の

分解に対応して

$$\pi = (\pi_0, \pi_1, \cdots, \pi_N)$$

π_n は $g(K, n)$ 次元ベクトル

と書いておくと，定常方程式 (3.1) は，

(3.2) $$\pi_{n-1} Q_{n-1} + \pi_n P_n + \pi_{n+1} R_{n+1} = 0 \qquad (n = 0, 1, \cdots, N)$$

但し，$Q_{-1} = 0$，$R_{N+1} = 0$

となる。これを満足する π_n は

$$\pi_n^{(k+1)} = -(\pi_{n-1}^{(k+1)} Q_{n-1} + \pi_{n+1}^{(k)} R_{n+1}) P_n^{-1}$$

又は

$$\pi_n^{(k+1)} = -\pi_{n-1}^{(k+1)} Q_{n-1} + \pi_n^{(k)} (I - P_n) - \pi_{n+1}^{(k)} R_{n+1}$$

によって求めることができるが、実際に計算してみると収束が極めて遅い。そこで条件付分布を計算する次の方法を用いることにする。すなわち

$$p_n = P\{(M_k, T_k) \in \mathcal{S}_n\}$$

(3.3) $$\tau_n = \frac{1}{p_n} \pi_n$$

とおくと，(3.2) 式は

(3.4) $$\frac{p_{n-1}}{p_n} \tau_{n-1} Q_{n-1} + \tau_n P_n + \frac{p_{n+1}}{p_n} \tau_{n+1} R_{n+1} = 0$$

と書ける。一方 A の行和が 0 であること：

$$R_n \xi + P_n \xi + Q_n \xi = 0$$

と、(3.2) 式とから

$$\pi_{n-1} Q_{n-1} \xi = \pi_n R_n \xi$$

が導かれ，

$$(3.5) \qquad \frac{p_{n-1}}{p_n} = \frac{\tau_{n-1} Q_{n-1} \xi}{\tau_n R_n \xi}$$

が得られる。したがって適当な $\{\tau_n\}$ の初期値から出発し，(3.5) 式を使って $\{\frac{p_{n-1}}{p_n}\}$ を計算し，それを (3.4) 式に代入して新しい $\{\tau_n\}$ を求める，という操作を繰り返せば (3.4) 式をみたす $\{\tau_n\}$, $\{\frac{p_{n-1}}{p_n}\}$ を求めることができる。$\{p_n\}$ は正規化条件を用いて $\{\frac{p_{n-1}}{p_n}\}$ から計算できるので，その $\{p_n\}$ と $\{\tau_n\}$ を使えば，(3.3) 式から $\{\pi_n\}$ が求まる。これら一連の計算手順をまとめると以下のようになる。

アルゴリズム

step. 1 （初期値設定）

$$\tau_n^{(0)} := 1 / g(K, n) \qquad (n = 0, 1, \cdots, N),$$

$$\alpha_n^{(0)} := \tau_n^{(0)} Q_n \xi \qquad (n = 0, 1, \cdots, N-1),$$

$$\beta_n^{(0)} := \tau_n^{(0)} R_n \xi \qquad (n = 1, 2, \cdots, N),$$

$$k := 1.$$

step. 2 （k回目の反復計算）

$$\tau_n^{(k)} := -\left(\frac{\beta_n^{(k-1)}}{\alpha_{n-1}^{(k)}} \tau_{n-1}^{(k)} Q_{n-1} + \frac{\alpha_n^{(k-1)}}{\beta_{n+1}^{(k-1)}} \tau_{n+1}^{(k-1)} R_{n+1} \right) P_n^{-1}$$
$$(n = 1, 2, \cdots, N),$$

$$\alpha_n^{(k)} := \tau_n^{(k)} Q_n \xi \qquad (n=1,2,\cdots,N-1),$$

$$\beta_n^{(k)} := \tau_n^{(k)} R_n \xi \qquad (n=1,2,\cdots,N).$$

step. 3 （収束判定）

$$\| \tau_n^{(k)} - \tau_n^{(k-1)} \| < \varepsilon \qquad (n=1,2,\cdots,N)$$

がなりたたなければ　$k := k+1$　として step 2 へもどる。

step 3′ （収束判定…流入・流出率の等価性の判定）

(1)　$\lambda \sum_k p_{0k} P\{k$ノードに到着できる$\}$,

$\sum_k \mu_k P\{k$ノードで次に退去する客に対してサービス中$\}$

の2つの値が誤差の範囲で等しいことを確かめる。

(2)　$\mu_k P\{k$ノードがサービス中$\}$,

$\sum_j \mu_j P\{j$ノードで次にkノードへ進む客に対してサービス中$\}$

の2つの値が誤差の範囲で等しいことを，すべての k について確かめる。

step. 4 （分布の計算）

$$p_0 := \left\{ \sum_{n=0}^{N} \prod_{j=1}^{n} \frac{\beta_j^{(k)}}{\alpha_{j-1}^{(k)}} \right\}^{-1},$$

$$p_n := p_0 \prod_{j=1}^{n} \frac{\beta_j^{(k)}}{\alpha_{j-1}^{(k)}}, \quad (n=1,2,\cdots,N),$$

$$\pi_n := p_n \tau_n^{(k)}. \qquad (n=0,1,\cdots,N).$$

このアルゴリズムが正常に働くことは，上の(3.3)式,(3.4)式(3.5)式からあきらかである。

なお、一回の反復（step. 2）の計算量（乗算の回数のオーダ）は、

$$\frac{K}{2}\left(r_0+\#(j : p_{j0}\neq 0)\right)\sum_n g(K, n-1)\, g(K, n)$$

である。但し、$g(K, n)$ は $g(K, 0)=1$ から単調に増加し、$n=\frac{N}{2}$ 付近で最大になり、そこから単調に減少して、$g(K, N)= r_1 r_2 \cdots r_K$ となる。最大値のオーダは

$$r_1 r_2 \cdots r_K N_1 N_2 \cdots N_K$$

である。

4. 数値例

前節で提示されたアルゴリズムは FORTRAN 77 によってコード化され、FUJITSU M380 OSⅣ/F4 MSP を使っていくつかのモデルについて試算された。プログラムステップは約500、使用メモリ量は、生成作用素の疎率（一行あたりの非

$$\lambda = 1.5$$

$$(\mu_k) = (2,\ 1,\ 2,\ 1.5,\ 1)$$

$$(N_k) = (3,\ 2,\ 3,\ 3,\ 2)$$

$$(P_{jk}) = \begin{bmatrix} 0 & 0.7 & 0.3 & 0 & 0 & 0 \\ 0 & 0 & 0.7 & 0.3 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0.3 & 0 & 0 & 0 & 0.5 & 0.2 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

表.1　テスト計算のためのパラメータの値

零要素数）を S_p ，整数型・実数型変数一つあたりのバイト数をそれぞれ I, R とした時

$$(I+R)(1+S_p)\Sigma_n g(K,n)$$

である。ここでは I=4, R=8 を用いた。

一つの数値例として、図1のモデルを使って試算した結果について述べる。テスト計算に用いられたモデルのパラメータは表.1の通りである。

方程式は2520次元となり、精度を 10^{-9} とした時の計算時間は3.9秒であった。Aの各小行列の次元、すなわち $g(K,n)$ の値は

$$(g(K,n))_n = (1, 8, 32, 88, 183, 302, 407, 452, 416, 314, 191, 90, 30, 6),$$

Aの非0要素の数は18675、したがって一行あたりの非0要素の平均個数は7.4である。また、N_5 を1から9まで1きざみで変化させたモデルについて、方程式の次元、計算時間

N_5	dim.	time	sparsity
1	1680	2.2 sec.	7.1
2	2520	3.9	7.4
3	3360	6.3	7.6
4	4200	9.6	7.6
5	5040	13.0	7.7
6	5880	16.6	7.7
7	6720	20.3	7.8
8	7560	24.2	7.8
9	8400	27.8	7.8

表.2 待ち合い室容量とモデルの規模

一行あたりの平均非0要素数は表.2のようになった。なお、計算時間は精度の対数と大体線形関係にあった。

次に各ノードの待ち合い室容量とブロック特性の関係についていくつか調べてみる。容量がどれぐらいあれば無限待ちモデルと見做せるか、を知るのが主な目的である。各ノードごと別々に、他の条件を一定にして、N_kを1から9まで変化させて種々の特性の変化を調べる。以下のグラフで、欄外に(1), (2), .. とあるのは、それぞれ N_1, N_2, .. を変化させるという意味である。

(イ) 損失確率(図2)

ノード1のバッファ効果が最も高く、$N_1=9$としてもまだ有限であることが無視できないことを示している。一方、ノード4, 5では待ち合い室容量がそれぞれ5, 2もあれば、無限にある場合と殆んど変りがないので、ノード1, 2あるいは3における混雑現象の解析は、それら3つのノードだけからなる、より小

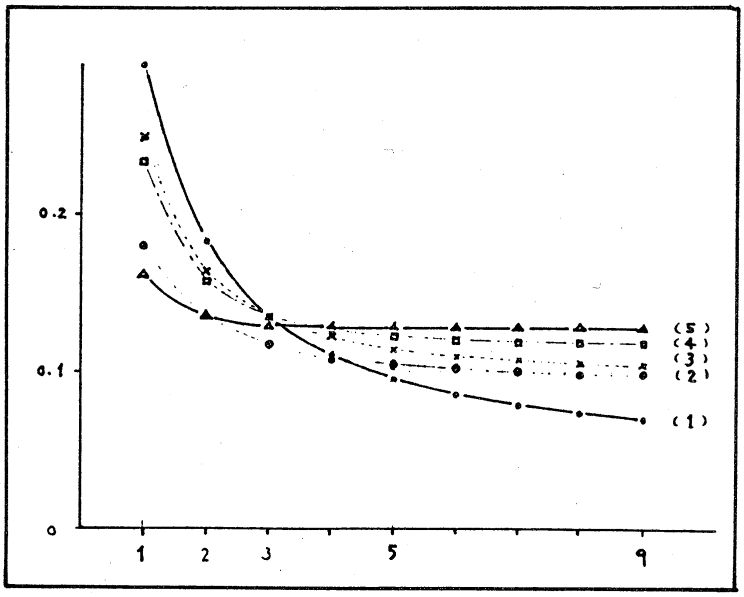

図.2 損失確率

さいモデルで解析しても良い近似が得られることが期待できる。

(ロ) 流出率（図3）

すべてのN_kを∞にした時の流出率と比較すると、ノード2については$N_2=7$で殆んど∞の場合（→で示す）と同じになっている。ノード1,2および3だけからなるモデルの損失確率を0.12とした時（図.2参照），N_4, N_5だけを∞にした時の最大流出率は，すべてのN_kを∞にした時の値に0.88を乗じたものになるが，それぞれ$N_4=4$, $N_5=2$で殆んどこの値（--->で示す）に収束している。

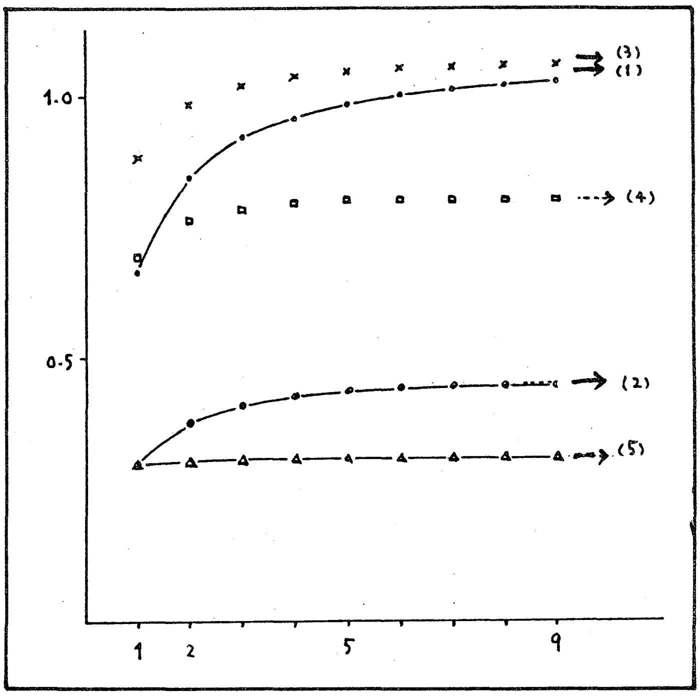

図.3 流出率

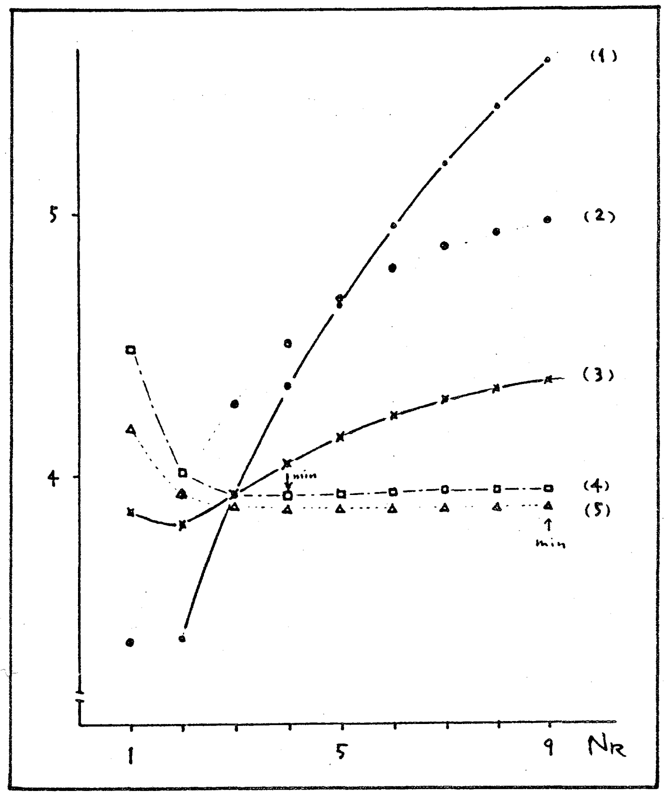

図4. 平均系内客数

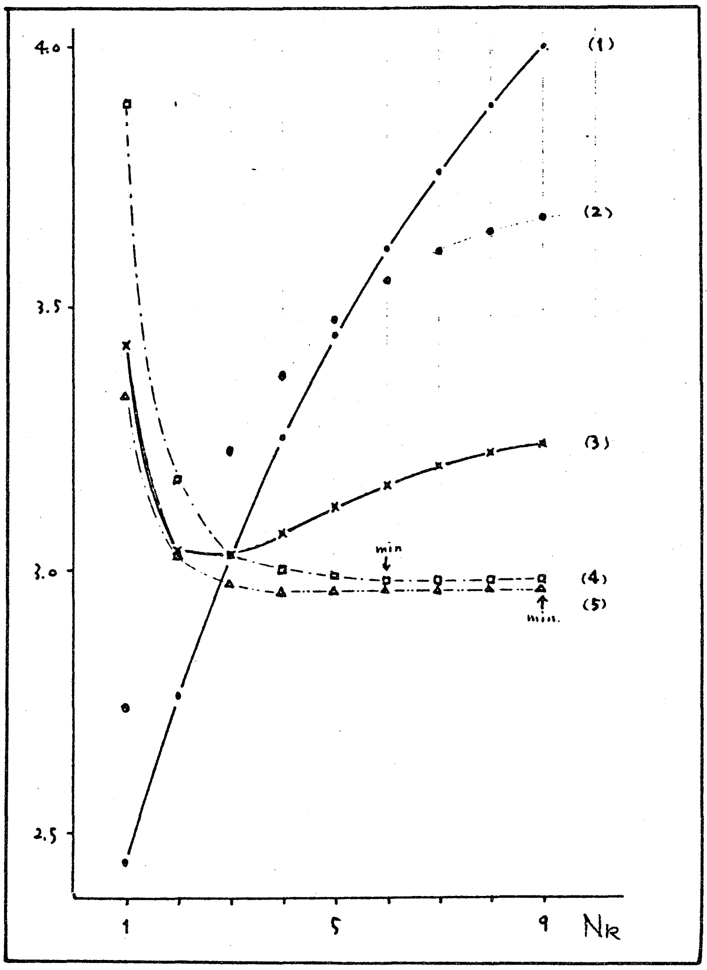

図5　平均滞在時間

(ハ) 平均系内客数(図4)

N_1, N_2 に関して単調増加であるが、N_3, N_4 に関しては、それぞれ $N_3=2$, $N_4=4$ に最小値があり、U型である。N_5 に関しては単調減少であるが、$N_5 \geqq 4$ で殆んど一定の値になっている。

(ニ) 平均滞在時間(図5)

$L=\lambda W$ の関係式が成り立つので平均系内客数と同じ傾向を示すが、N_3, N_4 に関して、その最小値をとる値はすこしずれがある。

5. 近似式との比較

Takahashi et al. (1980) によってこの型のモデルの近似式が提案されている。同時多発的なブロックを無視した時、サービス時間とブロックされている時間の和の期待値がみたさなければならない方程式系を考え、その解を各ノードの平均サービス時間とみなして積形式解を計算することによって

ブロック確率を計算することなしに近似値を求めようというものである。ここで扱っているモデルと異なる点はブロックの方式で、サービス終了後に先のノードが一杯ならば、その時点で窓口をブロックするという点である。表1のモデルで、N3だけを1から9にかえた時の損失確率と平均滞在時間について、真値と近似値を比較したのが図6と図7である。近似はあまり良いとは言えないが、定性的には原モデルを良く説明していると言える。

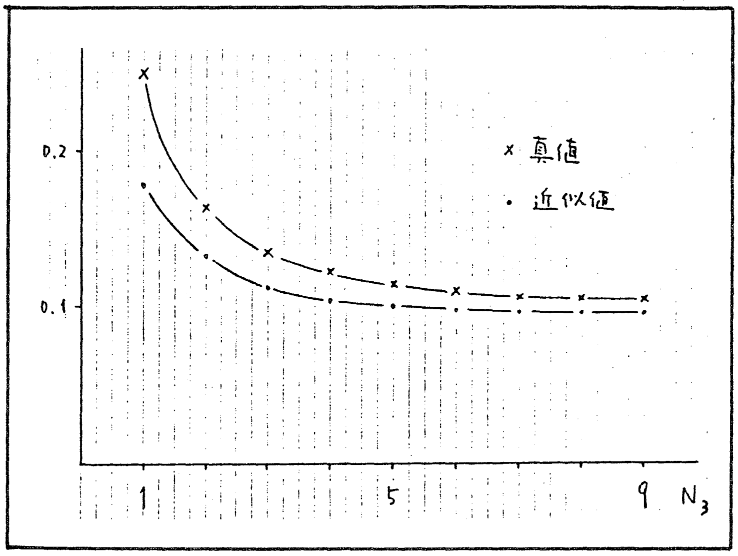

図6．損失確率

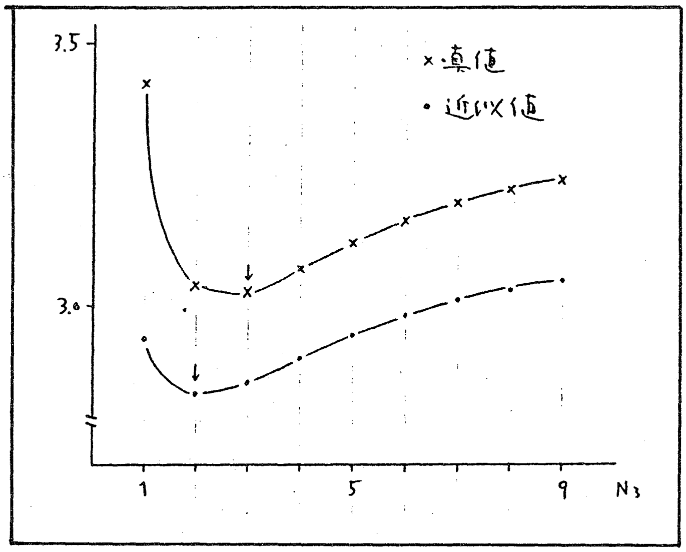

図7．平均滞在時間

参考文献

Baskett,F., Chandy,K.M., Muntz,R.R. and Palacios,F.G. (1975) Open, closed and mixed networks of queues with different classes of customers. JACM 22.2, 248-260.

Hunt,G.C. (1956) Sequential arrays of waiting lines. OR 4, 674-683.

Jackson,J.R. (1963) Jobshop-like queueing systems. MS 10.1, 131-141.

Takahashi,Y., Miyahara,H. and Hasegawa,T. (1980) An approximation method for open restricted queueing networks. OR 28.3, 594-602.

Takahashi,Y. and Takami,Y. (1976) A numerical method for the steady-state probabilities of a GI/G/c queueing system in a general class. JORSJ 19.2, 147-157.