

待ち行列システムの最適制御に関する 計算アルゴリズムについて

京大 工学部

大野勝久

1. はじめに

待ち行列システムの最適制御における主要な問題は、その最適制御政策のもつ性質を明らかにし、その政策を効率的に決定することである。しかし従来の研究は最適政策の状態に関する単調性等を示すにとどまり [1-5]、示された性質を有効に利用した最適政策の決定アルゴリズムまで論じたものは [6, 7] があるにすぎない。また、単調性等が示された問題も、 $M/G/1$, $GI/G/1$, $M/M/2$ システムにおけるサーバー数の制御問題、 $M/M/1$ システムにおけるサービス率の制御問題等簡単なものに限られている。

待ち行列システムの最適制御問題は通常セミ・マルコフ決定過程として定式化される。有限状態、有限決定のセミ・マルコフ決定過程にたいしては最適政策が定常になることが保証され、それを求めるアルゴリズムとして、

- (i) 政策反復法
- (ii) 逐次近似法
- (iii) 線形計画法
- (iv) 修正政策反復法

が知られている。したがって最適定常政策のもつ性質を何ん
ら知らなくても、これらアルゴリズムを用いれば最適定常政
策を容易に決定できるように思われるかもしれない。しかし
待ち行列システムの最適制御問題は簡単な問題でさえ状態数
が数百以上となり、(i) - (iii)のアルゴリズムでは容易に解
けない問題になる。本報告では、M/M/C システムおよ
びC段直列M/M型待ち行列システムにおけるサーバー数の
制御をも含むサービス率の制御問題を考え、その最適定常政
策を求める修正政策反復法 [8] によるアルゴリズムを提案
し、数値例によりその実用可能性を検討する。

2. M/M/C 待ち行列システムの最適制御

図 1 に示される M/M/C 待ち行列システムを考える。客
は平均到着率 λ のポアソン過程に従って到着し、行列長は N
に制限されているものとする。問題は、サーバー i ($i=1, \dots,$
 C) のとりうるサービス率が

$$\{0, \mu_i, \dots, M_i, \mu_i\}$$

で与えられるとき、システムの状態に応じて各サーバーのサ

サービス率をどのように制御すればよいかを求めることである。それに伴い、遊休中（サービス率が正でかつ客をサービスしていない）のサーバーが複数個存在する時、どのサーバーに客を割りあてるかも制御される。特に、全ての i で $M_i = 1$, $\mu_i = \mu$ とおけば、この問題は $M/M/C$ におけるサーバー数の制御問題になる。

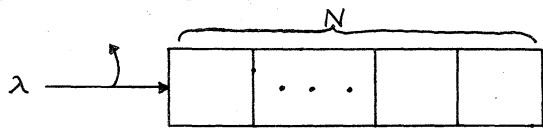


図 1

サーバー	サービス率
①	$\{0, \mu_1, \dots, M_1 \mu_1\}$
⋮	
②	$\{0, \mu_2, \dots, M_2 \mu_2\}$
⋮	
③	$\{0, \mu_c, \dots, M_c \mu_c\}$

このシステムの状態は客の到着あるいはサービスの終了時点でのみ変化し、したがってサービス率の制御およびサーバーへの客の割り当てもまた、客の到着あるいはサービスの終了直後に行なわれる。以下、自然な仮定として

- (C1) 休止中（サービス率が0）のサーバーへ客を割り当てることはない。
- (C2) 遊休中（サービス率が正でかつサービスをしていない）のサーバーは、待つている客がいる限り客を割り当てられる。
- (C3) 稼働中（現在サービス中）のサーバーのサービス

4

率は変えられず、サービス中の客を追いだすこともない。

をおき、費用として

r_1 : 待ち行列中の客の1人当り単位時間当り待ち費用

r_2 : サービス中の客の1人当り単位時間当り待ち費用

W_{1i} : サーバ- i の単位サービス率, 単位時間当り稼働費用

W_{2i} : サーバ- i の単位サービス率, 単位時間当り遊休費用

d_i : サーバ- i の単位サービス率 当り切り換え費用

e_i : サーバ- i の単位サービス率の切り換えに要する固定費用

を考える。

α : 割引率 ($\alpha > 0$)

とおく。問題は、無限期間にわたる平均割引費用を最小にするサービス率および客の割り当てを与える最適定常政策を求めることである。

この問題をセミ・マルコフ決定過程として定式化する。まず客の到着あるいはサービスの終了直後におけるシステムの状態 S を

$$(2.1) \quad S = (X, Y_1, \dots, Y_c, Z_1, \dots, Z_c)$$

で表わす。ここで

$X \in \{0, 1, \dots, N\}$: 待ち行列長

$Y_i \in \{0, 1, \dots, M_i\}$: (サーバー i の現在のサービス率) / μ_i

$$Z_i = \begin{cases} 0, & \text{サーバー}i\text{が遊休あるいは休止中 } (i=1, \dots, c) \\ 1, & \text{サーバー}i\text{が稼働中} \end{cases}$$

である。 \bar{S} で可能な状態 S の集合を表わすことにすれば、 $(0, 0, \dots, 0)$ 等の状態は起りえないから、 \bar{S} の要素数はたかだか

$$(2.2) \quad 2^c (N+1) \prod_{i=1}^c (M_i+1)$$

である。システムが客の到着あるいはサービスの終了で状態 S に遷移したとき、その状態 S をみて決定の集合 $A(S)$ から決定 a が選ばれる。決定 a は

$$(2.3) \quad a = (Y'_1, \dots, Y'_c, Z'_1, \dots, Z'_c)$$

と表わされる。ここで

$Y'_i \in \{0, 1, \dots, M_i\}$: (サーバー i のサービス率) / μ_i

$$Z'_i = \begin{cases} 0, & \text{サーバー}i\text{に客を割り当てない。}(i=1, \dots, c) \\ 1, & \text{サーバー}i\text{に客を割り当てる。} \end{cases}$$

である。仮定 C1) ~ C3) から各状態 S で可能な決定の集合 $A(S)$ が定められる。たとえば

6

$$A(1, 2, 1, \dots, 1, 0, 1, \dots, 1) = \left\{ (0, 1, \dots, 1, 0, 1, \dots, 1), (1, 1, \dots, 1, 1, 1, \dots, 1), \dots, (M_1, 1, \dots, 1, 1, 1, \dots, 1) \right\}$$

である。特に、 $\mu_1 = \mu_2, M_1 = M_2, \dots, e_1 = e_2$ のようにサーバー 1, 2 が同じである場合、状態 $S = (1, 0, 0, \dots, 0, 0, \dots)$ における決定 $a_1 = (1, 0, \dots, 1, 0, \dots)$ と $a_2 = (0, 1, \dots, 0, 1, \dots)$ は本質的に同じ決定になり、このような決定のうち $1 \rightarrow a_2$ は $A(1, 0, 0, \dots, 0, 0, \dots)$ から除外される。

状態 $S = (X, Y_1, \dots, Y_c, Z_1, \dots, Z_c)$ で決定 $a = (Y'_1, \dots, Y'_c, Z'_1, \dots, Z'_c)$ をとると状態は瞬間的に中間状態

$$(2, 4) \quad S' = (X', Y'_1, \dots, Y'_c, Z'_1, \dots, Z'_c)$$

へ移る。ここで Y'_i, Z'_i は決定 a により定まる次の遷移までのサーバー i のサービス率と状態を示し、 X' は a によって変化し t 待ち行列長であり

$$(2, 5) \quad X' = X + \sum_{i=1}^c (Z_i - Z'_i)$$

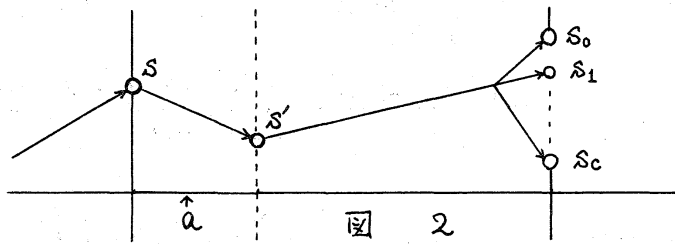
で与えられる。システムは次の客の到着あるいはサービス終了直前までこの状態 S' にとどまり、客が到着すれば

$$(2, 6) \quad S_0 = (X' + \delta(X'), Y'_1, \dots, Y'_c, Z'_1, \dots, Z'_c)$$

へ遷移し、サーバー i のサービスが終了すれば

$$(2, 7) \quad S_i = (X', Y'_1, \dots, Y'_c, Z'_1, \dots, 0, \dots, Z'_c)$$

へ遷移する。以上の状態遷移は図 2 のように図示される。



ここで、 $\delta(X)$ は

$$(2, 8) \quad \delta(X) = \begin{cases} 0, & X = N \text{ のとき} \\ 1, & X \neq N \text{ のとき} \end{cases}$$

である。状態 S で決定 a をとったとき状態 S_0 へ遷移する割引遷移確率 $f_{SS_0}(a)$ は

$$(2, 9) \quad f_{SS_0}(a) = \int_0^{\infty} e^{-\alpha t} \lambda \delta(X') \exp\{-\lambda \delta(X') t\} \exp\{-\sum_{i=1}^c Y_i' z_i' \mu_i t\} dt \\ = \frac{\lambda \delta(X')}{\lambda \delta(X') + \sum_{i=1}^c Y_i' z_i' \mu_i + \alpha}$$

で与えられる。同様に S_i への割引遷移確率 $f_{SS_i}(a)$ は

$$(2, 10) \quad f_{SS_i}(a) = \frac{Y_i' z_i' \mu_i}{\lambda \delta(X') + \sum_{i=1}^c Y_i' z_i' \mu_i + \alpha} \quad (i=1, \dots, c)$$

である。また、状態 S で決定 a をとったとき次の遷移までにかかる平均割引費用 $r(S, a)$ は、 $E(Y)$ を

$$(2, 11) \quad E(Y) = \begin{cases} 0, & Y = 0 \text{ のとき} \\ 1, & Y \neq 0 \text{ のとき} \end{cases}$$

で与えれば

$$(2.12) \quad r(s, a) = \sum_{i=1}^c \{ \mu_i d_i |Y_i - Y_i'| + e_i \varepsilon(Y_i - Y_i') \} \\ + \frac{h_1 X' + \sum_{i=1}^c \{ h_2 z_i + \mu_i Y_i' (w_{1i} z_i + w_{2i} (1 - z_i)) \}}{\lambda \delta(X') + \sum_{i=1}^c Y_i' z_i \mu_i + \alpha}$$

となる。したがって $M/M/C$ の最適制御問題は $s \in \bar{S}$ について

$$(2.13) \quad V^*(s) = \min_{a \in A(s)} \left\{ r(s, a) + \sum_{j=0}^c g_{ss_j}(a) V^*(s_j) \right\} \\ = r(s, f^*(s)) + \sum_{j=0}^c g_{ss_j}(f^*(s)) V^*(s_j)$$

をみたす最小費用 $V^*(s)$ および最適定常政策 $f^*(s)$ を求めセミ・マルコフ決定過程に帰着された。

3. 修正政策反復法

前節で定式化された問題を解く修正政策反復法を示す。

$$(3.1) \quad \alpha(s, a) = \frac{\sum_{j=0}^c g_{ss_j}(a)}{(\lambda \delta(X') + \sum_{i=1}^c \mu_i Y_i' z_i) / (\lambda \delta(X') + \sum_{i=1}^c \mu_i Y_i' z_i + \alpha)}$$

$$(3.2) \quad \beta = \max_{s, a} \{ \alpha(s, a) \} = \left(\lambda + \sum_{i=1}^c \mu_i M_i \right) / \left(\lambda + \sum_{i=1}^c \mu_i M_i + \alpha \right)$$

$$(3.3) \quad \gamma = \min_{s, a} \{ \alpha(s, a) \} = \min \{ \lambda, M_1, \dots, M_c \} / \{ \min \{ \lambda, M_1, \dots, M_c \} + \alpha \}$$

とおく

$$\text{スラック } \rho : \quad C_0 = \max_{s \in \bar{S}} \min_{a \in A(s)} \{ r(s, a) \} / (1 - \beta) \text{ を計算す}$$

る。

ステップ 1: 非負整数 m , 正定数 ε , 十分大きな正数 M を定め, $S \in \bar{S}$ について

$$U^0(S) = M, V^0(S) = C_0, A^0(S) = A(S)$$

とおき, $\eta_0 = 0, n = 1$ とおく。

ステップ 2: $S \in \bar{S}$ について

$$(3.4) \quad W^n(S) = \min_{a \in A^{n-1}(S)} \left\{ r(S, a) + \sum_{j=0}^{\varepsilon} g_{SS_j}(a) V^{n-1}(S_j) \right\}$$

を計算し, 同時に

$$(3.5) \quad A^n(S) = \left\{ a \in A^{n-1}(S); r(S, a) + \sum_{j=0}^{\varepsilon} g_{SS_j}(a) V^{n-1}(S_j) \leq U^{n-1}(S) - \alpha(S, a) \eta_{n-1} \right\}$$

を定める。 $f^{n-1}(S)$ が $W^n(S)$ を与えれば $f^n(S) = f^{n-1}(S)$ とおき, さもなければ $f^n(S)$ を $W^n(S)$ を与える任意の決定にとる。もし, $A^n(S)$ が全ての S で唯一となればステップ 5 へ行く。(注意, $r(S, a), g_{SS_j}(a), \alpha(S, a)$ はメモリーに記憶せず, 必要に応じて (2.9) - (2.12), (3.1) から計算される。)

ステップ 3: $Y^0(S) = W^m(S)$ とおき, $l = 0, 1, \dots, m-1$ について

$$(3.6) \quad Y^{l+1}(S) = r(S, f^l(S)) + \sum_{j=0}^{\varepsilon} g_{SS_j}(f^l(S)) Y^l(S_j)$$

($S \in S$) を計算し, $V^n(S) = Y^m(S)$ とおく。

ステップ 4:

$$(3.7) \quad \begin{cases} \Delta_n = \max_s \{ v^n(s) - v^{n-1}(s) \}, & \nabla_n = \min_s \{ v^n(s) - v^{n-1}(s) \} \\ a_n = \max_s \{ w^n(s) - v^n(s) \}, & b_n = \min_s \{ w^n(s) - v^n(s) \} \end{cases}$$

$$(3.8) \quad \begin{cases} \xi_n = \min \{ -\gamma^m b_n / (1 - \gamma^m), (\gamma \Delta_n + a_n) / (1 - \gamma) \} \\ \eta_n = (\beta \nabla_n + b_n) / (1 - \beta) \end{cases}$$

を計算する。もし

$$(3.9) \quad \xi_n - \eta_n \geq 2\varepsilon$$

ならば $U^n(s) = v^n(s) + \xi_n \quad (s \in \bar{S})$,

$n = n + 1$ とおき, ステップ 2 へ戻る。もし

ならば

$$(3.10) \quad v(s) = v^n(s) + (\xi_n + \eta_n) / 2 \quad (s \in \bar{S})$$

は $v^*(s)$ の ε 近似値である。

$$(3.11) \quad \begin{cases} \delta_n = \min_s \{ v(s) - r(s, f^n(s)) - \sum_{j=0}^{\infty} \beta^j s_{ss_j} (f^n(s)) v(s_j) \} \\ \varepsilon_n = \begin{cases} \varepsilon - \delta_n / (1 - \min_s d(s, f^n(s))), & \delta_n \geq 0 \text{ の時} \\ \varepsilon - \delta_n / (1 - \max_s d(s, f^n(s))), & \delta_n < 0 \text{ の時} \end{cases} \end{cases}$$

とおけば, f^n は ε_n -最適政策である。

ステップ 5: $\{ f^n(s); s \in \bar{S} \}$ は唯一の最適定常政策である。

$$(3.12) \quad c_n = \max_s \{ w^n(s) - v^n(s) \}, \quad d_n = \min_s \{ w^n(s) - v^n(s) \}$$

を計算すれば, $v^*(s)$ は $s \in \bar{S}$ について

$$(3.13) \quad v(s) = w^n(s) + \frac{1}{2} \{ \gamma c_n / (1 - \gamma) + \beta d_n / (1 - \beta) \}$$

で評価される。

上記アルゴリズムで記憶すべきものは、 $v^{n-1}(s), v^n(s), w^n(s), u^n(s), A^n(s), f^n(s)$ であり、必要な記憶容量は大体

$$(3.14) \quad 5|S| + \sum_{s \in S} |A(s)|$$

である。ここで $|\cdot|$ は \cdot の要素数を表わす。

上記アルゴリズムでは v^0 としてステップ 0 でえられた C_0 を用いたが、(3.8)式を一部修正すれば任意の v^0 を用いることができる。また $m=0$ とすれば上記アルゴリズムは (ii) 逐次近似法に一致し、 $m \rightarrow \infty$ とすれば (i) 政策反復法における値決定ルーチンを逐次近似法で解くアルゴリズムに一致する。詳細は [8] を参照されたい。

4. 数値例

$$C = 2, \mu_1 = \mu_2 = 0.01, M_1 = M_2 = 4, \lambda = 0.06, N = 10$$

とおいた例を考える。サーバー 1, 2 は同一であるから、たとえば状態 $(1, 3, 2, 0, 1)$ と $(1, 2, 3, 1, 0)$ は区別できず、可能な状態数は 740 となる。また、可能な決定の総数は 4980 となる。

この問題を (i) 政策反復法で解くためには 740 元連立一次方程式を解かねばならず、ガウスの消去法では相当困難で

ある。

(iii) 線形計画法で解くためには、4980変数740制約のLP問題を解かねばならず、やはり困難である。以下前節のアルゴリズムによる結果を示す。

$$(1) \quad h_1 = 10, \quad h_2 = 0, \quad w_{11} = w_{12} = 500, \quad w_{21} = w_{22} = 250 \\ d_1 = d_2 = 5000, \quad e_1 = e_2 = 0$$

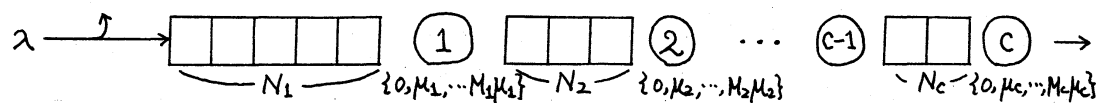
とした問題が $m=10$, $\varepsilon=0.01$ とおいた修正政策反復法で約19秒で解かれ、唯一の最適定常政策(ステップ5)がえられた。[9]のアルゴリズムで状態を整理した結果が最終頁に与えられている。

5. タンデム待ち行列システムの最適制御

下図に示されるC段直列型M/M待ち行列システムを考える。客は平均到着率 λ のポアソン過程に従って到着し、 i 段の行列長は N_i ($i=1, \dots, C$)に制限されているものとする。問題は i 段のサーバー(サーバー i とよぶ)のとりうるサービス率が

$$\{0, \mu_i, \dots, M_i \mu_i\} \quad (i=1, \dots, C)$$

で与えられる時、システムの状態に応じて各サーバーのサービス率をどのように制御すればよいかを決定することである。



システムの状態は、§2と同様客の到着あるいはサービスの終了時点に変化する。以下、§2の仮定 (C1) - (C3) を仮定し、同じ費用構造を採用する。まず、客の到着あるいはサービスの終了直後におけるシステムの状態 S を

$$(5.1) \quad S = (X_1, \dots, X_c, Y_1, \dots, Y_c, Z_1, \dots, Z_c)$$

で表わす。ここで

$X_i \in \{0, 1, \dots, N_i\}$: i 段の待ち行列長 ($i=1, \dots, c$)

$Y_i \in \{0, 1, \dots, M_i\}$: (サーバー i の現在のサービス率) / μ_i

$Z_i = \begin{cases} 0, & \text{サーバー } i \text{ が遊休あるいは休止中} \\ 1, & \text{稼動中} \\ 2, & \text{ブロックされている。} \end{cases} \quad (i=1, \dots, c-1)$

$Z_c = \begin{cases} 0, & \text{サーバー } c \text{ が遊休あるいは休止中} \\ 1, & \text{稼動中} \end{cases}$

である。状態 S における決定 a は

$$(5.2) \quad a = (Y'_1, \dots, Y'_c)$$

で表わされる。ここで

$Y'_i \in \{0, 1, \dots, M_i\}$: (サーバー i のサービス率) / μ_i

である。ブロックされているサーバーのサービス率も変えられるものとするれば、 S でとりうる決定の集合 $A(S)$ は、

$$Z_i = 0 \text{ or } 2 \quad (i=1, \dots, c-1), \quad Z_c = 0$$

となる各サーバー i について Y'_i を $0 \sim M_i$ まで変化させてえ

られる全ての組み合わせからなる。

状態 $S = (X_1, \dots, X_c, Y_1, \dots, Y_c, Z_1, \dots, Z_c)$ で決定
 $a = (Y'_1, \dots, Y'_c)$ をとると, §2 と同様中間状態

$$(5.3) \quad S' = (X'_1, \dots, X'_c, Y'_1, \dots, Y'_c, Z'_1, \dots, Z'_c)$$

へ遷移する。ここで Y'_i は決定 a から定まる次の遷移までの
 のサーバー i のサービス率を示し, X'_i, Z'_i は次式から定
 められる。

$$(5.4) \quad Z'_c = \begin{cases} \varepsilon(X_c), & Y'_c \neq 0, Z_c = 0 \text{ のとき} \\ Z_c, & \text{そうでないとき} \end{cases}$$

$$(5.5) \quad X'_c = \begin{cases} X_c - \varepsilon(X_c), & Y'_c \neq 0, Z_c = 0, Z_{c-1} \neq 2 \text{ のとき} \\ X_c, & \text{そうでないとき} \end{cases}$$

$$(5.6) \quad Z''_c = \begin{cases} 0, & Y'_c \neq 0, Z_c = 0, Z_{c-1} = 2 \text{ のとき} \\ Z_{c-1}, & \text{そうでないとき} \end{cases}$$

であり, $i = c-1, \dots, 1$ について

$$(5.7) \quad Z'_i = \begin{cases} \varepsilon(X_i), & Y'_i \neq 0, Z''_{i+1} = 0 \text{ のとき} \\ Z''_{i+1}, & \text{そうでないとき} \end{cases}$$

$$(5.8) \quad X'_i = \begin{cases} X_i - \varepsilon(X_i), & Y'_i \neq 0, Z''_{i+1} = 0, Z_{i-1} \neq 2 \text{ のとき} \\ X_i, & \text{そうでないとき} \end{cases} \quad (Z_0 \triangleq 0)$$

$$(5.9) \quad Z''_i = \begin{cases} 0, & Y'_i \neq 0, Z''_{i+1} = 0, Z_{i-1} = 2 \text{ のとき} \\ Z_{i-1}, & \text{そうでないとき} \end{cases} \quad (i \geq 2)$$

である。ここで $\delta_i(X)$ は (2.11) で与えられている。(2.8)

同様, $i=1, \dots, c$ について

$$(5.10) \quad \delta_i(X) = \begin{cases} 0 & , X = N_i \text{ のとき} \\ 1 & , \text{ そうでないとき} \end{cases}$$

を定義し,

$$(5.11) \quad \phi_j(z) = \begin{cases} 1 & , z = j \text{ のとき} \\ 0 & , \text{ そうでないとき} \end{cases} \quad (j=1, 2)$$

とおく。中間状態 S で客が到着すればその直後に

$$(5.12) \quad S_0 = (X'_1 + \delta_1(X'_1), X'_1, X'_2, \dots, X'_c, Y'_1, \dots, Y'_c, z'_1, \dots, z'_c)$$

へ遷移し, サーバ i のサービスが終了すればその直後に

$$(5.13) \quad S_i = (X'_1, \dots, X'_i, X'_{i+1} + \delta_{i+1}(X'_{i+1}), \dots, X'_c, \\ Y'_1, \dots, Y'_c, z'_1, \dots, z'_{i-1}, 2(1 - \delta_{i+1}(X'_{i+1})) \\ z'_{i+1}, \dots, z'_c) \quad (i < c)$$

$$(5.14) \quad S_c = (X'_1, \dots, X'_c, Y'_1, \dots, Y'_c, z'_1, \dots, z'_{c-1}, 0)$$

へ遷移する。割引遷移確率は δ と同様

$$(5.15) \quad g_{ss_0}(a) = \lambda \delta_1(X'_1) / \{ \lambda \delta_1(X'_1) + \sum_{i=1}^c Y'_i \mu_i \phi_2(z'_i) + \alpha \}$$

であり, $i=1, \dots, c$ について

$$(5.16) \quad g_{ss_i}(a) = Y'_i \mu_i \phi_1(z'_i) / \{ \lambda \delta_i(X'_i) + \sum_{i=1}^c Y'_i \mu_i \phi_2(z'_i) + \alpha \}$$

である。ブロックされているサーバには遊休費用 w_{2i} が

かかるものとし、客にも待ち費用 h_i がかかるものとするれば、平均割引費用 $r(s, a)$ は

$$(5.17) \quad r(s, a) = \sum_{i=1}^c \{ \mu_i d_i |Y_i - Y_i'| + e_i \varepsilon(Y_i - Y_i') \} \\ + \frac{1}{\{ \lambda \delta_1(X_1) + \sum_{i=1}^c Y_i' \mu_i \phi_1(z_i) + \alpha \}} \times \left\{ h_1 \left(\sum_{i=1}^c X_i' + \sum_{i=1}^c \phi_2(z_i) \right) \right. \\ \left. + h_2 \sum_{i=1}^c \phi_1(z_i) + \sum_{i=1}^c \mu_i Y_i' (w_{1i} \phi_1(z_i) + w_{2i} (1 - \phi_1(z_i))) \right\}$$

で与えられる。

6. 数値例

$$c = 2, N_1 = 10, N_2 = 2, \lambda = 0.03, \mu_1 = \mu_2 = 0.02$$

$$M_1 = M_2 = 3, \alpha = 0.01, h_1 = h_2 = 10, w_{11} = w_{12} = 250$$

$$w_{21} = w_{22} = 125, d_1 = d_2 = 5000, e_1 = e_2 = 0$$

を § 3 の修正政策反復法を用いて解き、唯一の最適定常政策 (ステップ 5) が 58 秒でえられた。ただし、 $m = 10$, $\varepsilon = 0.01$ とした。この問題の状態数は 1588, くりうる決定の総数は 11530 であり、他の手法ではほぼ不可能な規模である。

7. おわりに

§ 4, 6 で $M/M/2$, $M/M \rightarrow M$ 待ち行列システムのサービス率の最適定常制御政策が § 3 の修正政策反復法で比較的容易にえられることが示された。今後より一層アルゴリ

ズムを改良するとともに実行しやすい準最適政策の性質を見出し、その性質を上手く利用したアルゴリズムを開発することが残された課題である。

最後に常日頃御指導いただき、三根久教授に深謝するとともに全ての数値計算を実行していただいた市木邦美君に感謝いたします。

参考文献

- [1] T.B. Crabill, D. Gross and M.J. Magazine, "A survey of research on optimal design and control of queues," Tech. paper T-280. The George Washington Univ. 1973.
- [2] T.B. Crabill, D. Gross and M.J. Magazine, "A classified bibliography of research on optimal design and control of queues," Operations Res., 25, 219-232 (1977).
- [3] N. Prabhu and S. Stidham, Jr. "Optimal control of queueing systems," in "Mathematical methods in queueing theory," pp.263-294. Springer, 1974.
- [4] M.J. Sobel, "Optimal operation of queues," ibid. pp.231-261, Springer, 1974.
- [5] 大野勝久, "待ち行列システムの最適制御" 数理解析研究所講究録 No.425, pp.106-127, 1981
- [6] C.E. Bell, "Characterization and computation of optimal policies for operating an M/G/1 queueing system with removable servers," Operations Res., 19. 208-218 (1971).
- [7] H.C. Tijms, "An algorithm for average costs denumerable state semi-Markov decision problems with applications to controlled production and decision

- processes," in "Recent development in Markov decision processes," ed. by R. Hartley et al. pp.143-179. Academic. 1980.
- [8] K. Ohno, "A unified approach to algorithms with a suboptimality test in discounted semi-Markov Decision Processes," J. Oper. Res. Soc. Japan, 24, 296-324 (1981).
- [9] B.L. Fox and D.M. Landi, "An algorithm for identifying the ergodic subchains and transient states of a stochastic matrix," Comm. ACM 11, 619-621 (1968).

LAM= 0.06000 MU= 0.01000 AL= 0.01000 LP= 0.01000
 HC= 10.00000 HS= 0.0 W1= 500.00000 W2= 250.00000 SF= 5000.00000 ST= 0.0 M= 10
 NUMBER= 0 360MSEC

NUMBER= 1 1263980.89851 1206MSEC
 NUMBER= 2 314595.94907 1219MSEC
 NUMBER= 3 30699.60082 1226MSEC
 NUMBER= 4 20923.03452 1264MSEC
 NUMBER= 5 15682.92022 1220MSEC
 NUMBER= 6 3202.40208 1201MSEC
 NUMBER= 7 2353.45927 1252MSEC
 NUMBER= 8 614.97090 1284MSEC
 NUMBER= 9 224.00045 1282MSEC
 NUMBER= 10 82.50336 1192MSEC
 NUMBER= 11 36.61056 1100MSEC
 NUMBER= 12 16.21434 1074MSEC
 NUMBER= 13 7.23058 1052MSEC
 NUMBER= 14 3.26068 1041MSEC
 NUMBER= 15 1.44431 1056MSEC
 NUMBER= 16 0.63842 1029MSEC
 CONVERGENCE 297MSEC

NUMBER OF ITERATIONS 17

UNIQUE-OPTIMAL POLICY IS OBTAINED.

CPU= 19355MSEC

1000 1000 3225.818
 1100 1100 3264.053
 1110 1110 3727.362
 2000 2000 3218.675
 2100 2100 3256.410
 2101 2101 3720.219
 2110 2110 3674.732
 2200 2200 3295.146
 2210 2210 3677.674
 3000 3000 3238.052
 3100 3100 3280.360
 3101 3101 3716.400
 3110 3110 3660.423
 4000 3000 3288.052
 4100 3100 3330.360
 4101 4101 3730.380
 4110 4110 3687.211

3200 3200 3322.667
 3201 3201 3678.720
 3210 3210 3668.115
 3300 3200 3372.667
 3310 3310 3698.888

4200 3200 3372.667
 4201 4201 3714.924
 4210 4210 3698.150