情報処理教育のためのプログラムファイル容量と出力量の時間解について

（昭和53年10月31日　原稿受付）

西日本工業大学　情報処理センター　深川幸紀
九州工業大学　情報処理教育センター　積山洋子
〃　〃　中山泰雄
〃　〃　矢鳴虎夫
九州大学工学部　吉田将

The Time Dependent Solution on the Capacity of the Program File and Output Queue Length

by Yukinori FUKAGAWA
Yoko TSUMIYAMA
Yasuo NAKAYAMA
Torao YANARU
Sho YOSHIDA

At our Educational Center for Information Processing, the online system has been implemented for education on programming and this system is used effectively by students. But the online system requires the large capacity of the memory device for source programs, so we have designed the system to delete a memory segment in the device (program file) for one student if this segment be not accessed during past a constant period.

We have considered the capacity of the memory device (program file) and the amount of deleted (departed) segments under approximation that the segment-control-system is same as the multi-server queuing system. If all source programs and results of excution or log on access to the system of students be saved in the secondary memory device, those will be useful records for the education on programming.

1．まえがき

九州工業大学情報処理教育センターでは，複数台のディスプレイ装置を用いたオンラインシステムが学生のプログラミング教育のために開発され，効果をあげている。このオンラインシステムでは学生用のプログラムファイルが必須であり，この容量決定は大切である。我々はこの容量制御に関するモデルを $M/M/N$ 待ち行列で近似して，ファイル保持時間を制御する場合を検討した。[8] 定常状態の場合，簡単な制御でも効果が上がることがわかった。またファイル専有容量の立ち上がりの近似解についても検討した。[9]

さて，学生に対するプログラミング教育等，計算機システムを教育に用いる場合，学生が計算機をアクセスした情報全て（またはその一部分）を記録に残しておくことは，今後のシステムで考慮にいれるべきであろう。例えばプログラミング教育においてその学生のソースプログラムと実行結果のリスト等を記録して，その学生の演習に関するトレース等を調べることにより多くの情報が得られるであろう。

この時プログラムファイルはソースプログラムのみならず実行結果や学生個々に関する情報域をもっている必

要があり，大きなエリアを必要とするであろう。

本文では，出力情報記憶域として二次記憶域（具体的には磁気テープ）に記録されて，プログラムファイルセグメントが消去される時の出力セグメント数とプログラムセグメント数に関する状態をしらべた。

2．システムのモデル

学生は，ファイルの一定部分（これを1セグメントと呼ぶ）を固定長で用いるとする。このセグメントのなかに一人の学生のソースプログラムや実行結果のリスト等の情報が記憶されるとする。そしてある学生のためのセグメントが一次記憶域から消去されるとこれは二次記憶域に転写される。本文ではある学生のセグメントの有効期間分布が指数分布であるとして図-1，図-3のモデルすなわち無限呼源モデルと有限呼源モデルで主としてセグメント数と二次記憶容量の平均値を算出し，セグメント要求発生分布との関係を明らかにしたものである。

なお，ここでは一次記憶域（プログラム・ファイル）容量はM人分（有限の学生総数）または無限大としている。これをある値以下にした場合には解析が非常に複雑となる。

2.1. 有限呼源モデル

図-1に有限呼源モデルを示す。プログラムファイル使用者数をM人，プログラムセグメント総数をM，二次記憶域容量は無限大とする。システムの初期条件はいずれの記憶域も空とする。

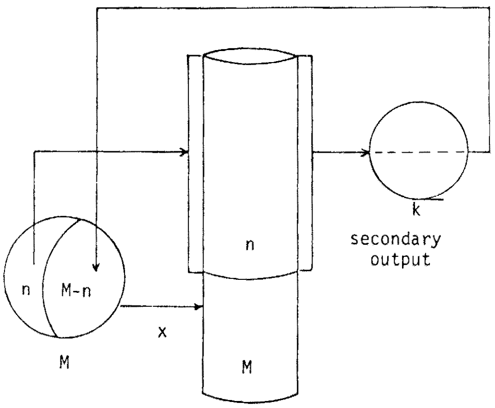

Fig. 1 Finite queue model

システムの時刻tにおける状態を$p_{n,k}(t)$とあらわす。すなわちtでプログラムファイルがnセグメント，二次記憶域容量がkセグメントが使用されている確率を$p_{n,k}(t)$とする。到着過程は平均λをもつポアソン分布とする。プログラムファイル保持時間は平均μをもつ指数分布とする。

システムの方程式は(1)〜(5)となる。

$$P'_{0,0}(t) = -M\lambda P_{0,0}(t) \tag{1}$$

$$P'_{0,0}(t) = -\{(M-n)\lambda + n\mu\}P_{n,0}(t) + \lambda\{M-(n-1)\}P_{n-1,0}(t) \quad n \geqq 1 \tag{2}$$

$$P'_{n,k}(t) = -\{(M-n)\lambda + n\mu\}P_{n,k}(t) + \lambda\{M-(n-1)\}P_{n-1,k}(t) + \mu(n+1)P_{n+1,k-1}(t) \tag{3}$$

$$k \geqq 1 \qquad 1 \leqq n \leqq \mu - 1$$

$$P'_{0,k}(t) = -M\lambda P_{0,k}(t) + \mu P_{1,k-1}(t) \tag{4}$$

$$P'_{M,k}(t) = -M\mu P_{M,k}(t) + \lambda P_{M-1,k}(t) \tag{5}$$

(1)〜(5)式を母函数 $G(x,y,t) \equiv G(x,y) = \sum_{n=0}^{M}\sum_{k=1}^{\infty} P_{n,k}(t)x^n y^k$ を用いてあらわせば(6)式となる。

$$\frac{\partial}{\partial t}G(x,y) = -M\lambda(1-x)G(x,y) + \{\lambda x(1-x) + \mu(y+x)\}\frac{\partial}{\partial t}G(x,y) \tag{6}$$

(6)式の特性方程式は

$$\frac{dt}{1} = \frac{-dx}{\lambda x(1-x)+\mu(y-x)} = \frac{dG}{-\mu\lambda(1-x)G} = \frac{dy}{0} \tag{7}$$

となる。ここでx_1とx_2は

$$x_1 \equiv \frac{-(1-\rho)+\sqrt{(1-\rho)^2+4\rho y}}{2\rho},$$

$$x_2 \equiv \frac{-(1-\rho)-\sqrt{(1-\rho)^2+4\rho y}}{2\rho}$$

でこれを用いて(7)を書きなおすと

$$\frac{dt}{1} = \frac{1}{\mu\rho}\frac{1}{x_1-x_2}\left\{\frac{1}{x-x_1} - \frac{1}{x-x_3}\right\}dx = \frac{dG}{-\mu\lambda(1-x)G} = \frac{dy}{0} \tag{8}$$

(8)式の第一，二項の関係より

$$\frac{dt}{1} = \frac{a}{\mu\rho}\left\{\frac{dx}{x-x_1} - \frac{dx}{x-x_2}\right\}$$

$$\therefore \quad e^{\theta t} = \left(\frac{x-x_1}{x-x_2}\right)\times c_1$$

$$c_1 = e^{\theta t}\left(\frac{x-x_1}{x-x_2}\right) \tag{9}$$

ここで c_1 は積分定数，$\theta = (x_1 - x_2)\mu\rho$

(9)式よりこれを変形すれば

$$1-x=\frac{(1-x_1)-(1-x_2)c_1e^{\theta t}}{1-c_1e^{\theta t}}$$

これを(8)式の第3項に代入して，第一項，第三項の関係より

$$\frac{dG}{G}=-M\lambda\frac{(1-x_1)-(1-x_2)c_1e^{\theta t}}{1-c_1e^{\theta t}}dt$$
$$=-\frac{M\lambda(1-x_1)}{1-c_1e^{\theta t}}dt+\frac{M\lambda(1-x_2)c_1e^{\theta t}}{1-c_1e^{\theta t}}dt$$

積分 $\displaystyle\int\frac{dx}{a+be^{mx}}=\frac{1}{a_m}\{mx-\log(a+be^{mx})\}$

$$\int f(e^{ax})dx=\int\frac{f(y)}{ay}dy$$

を用いて上式を計算すれば

$$G=e^{-\mu\lambda\alpha_1 t}(1-c_1e^{\theta t})^{\gamma(\alpha_1-\alpha_2)}\times c_2 \tag{10}$$

ここで $\alpha_1=1-x_1$，$\alpha_2=1-x_2$，$\gamma=M\lambda/\theta$ で c_2は積分定数

(10)式に(9)式を代入すれば(6)式の一般解は(11)式となる。

$$G=e^{-\mu\lambda\alpha_1 t}\left(\frac{x_1-x_2}{x-x_2}\right)^{\gamma(\alpha_1-\alpha_2)}\phi\left\{e^{-\theta t}\frac{x-x_1}{x-x_2}\right\} \tag{11}$$

ここで ϕ は任意函数である。

さて ϕ を決定するために初期条件を用いる。

$$G(x,y,0)=1=\left(\frac{x-x_1}{x-x_2}\right)^{\gamma(\alpha_1-\alpha_2)}\phi\left\{\frac{x-x_1}{x-x_2}\right\}$$

$\dfrac{x-x_1}{x-x_2}=u$ とすると $x=\dfrac{x_1-x_2}{1-u}u$ となりこれを上式に代入すれば

$$\phi\{u\}=\frac{1}{(1-u)^{\gamma(\alpha_1-\alpha_2)}}\quad となる。$$

$$\therefore\quad \phi\left\{e^{-\theta t}\frac{x-x_1}{x-x_2}\right\}=\left\{1-e^{-\theta t}\frac{x-x_1}{x-x_2}\right\}^{-\gamma(\alpha_1-\alpha_2)} \tag{12}$$

(12)式を(11)式に代入して整理すれば

$$G(x,y,t)=\left\{\frac{1}{(x_1-x_2)}\right\}^M\{(x-x_2)-(x-x_1)e^{-\theta t}\}^M\times e^{-\mu\lambda\alpha_1 t} \tag{13}$$

(13)式を少し調べることにする。

(イ) (13)式は(6)式をみたす。

(ロ) $G(x,y,0)=1$ 初期条件をみたす。

(ハ) $G(1,1,0)=1$ 正規化条件をみたす。

(ニ) $G(x,1,t)=\left(\dfrac{\rho}{\rho+1}\right)^M\left\{\left(x+\dfrac{1}{\rho}\right)-(x-1)e^{-\mu(\rho+1)t}\right\}^M$

は既述の結果[9]と一致する。

$P_{n,k}(t)$ は(13)式の $x^n\,y^k$ の係数であるが非常に複雑となるので簡単に得られる平均値を計算する。

$L_x.(t)=\sum_{n=0}^{M}n\sum_{k=0}^{\infty}P_{n,k}(t)$ をプログラムファイルの平均値 $L.y(t)=\sum_{k=0}^{\infty}k\sum_{n=0}^{M}P_{n,k}(t)$ を二次記憶域の平均値とする。

$$L_x.(t)=\frac{\partial}{\partial x}G(x,1,t)\Big|_{x\to 1}$$
$$=\frac{\rho}{\rho+1}M\{1-e^{-\mu(\rho+1)t}\} \tag{14}$$

$$L.y(t)=\frac{\partial}{\partial y}G(1,y,t)\Big|_{y\to 1}$$
$$=\frac{\rho}{(\rho+1)^2}M\{e^{-\mu(1+\rho)t}+\mu(1+\rho)t-1\} \tag{15}$$

(14)は文献9と同一のものである。

図—2に平均値を示す。

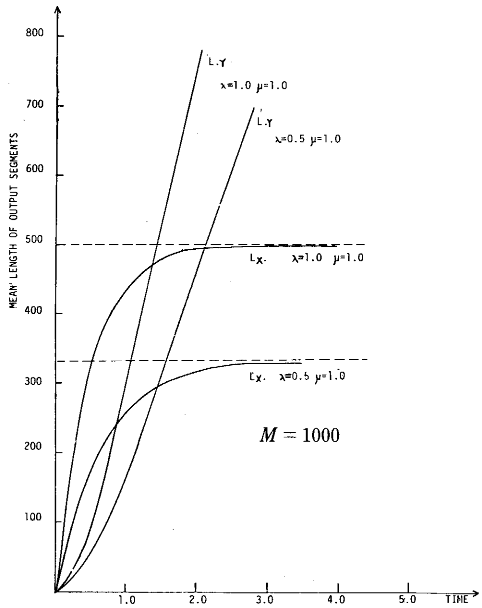

Fig. 2 Mean length ($L_x.$, $L\cdot_y$) of finite source model

$L.y(t)$ に関して，時間の経過とともに

$$L.y(t)\fallingdotseq\frac{M\lambda}{\rho+1}t-\frac{M\rho}{(\rho+1)^2} \tag{16}$$

の直線に近づいてゆくことがわかり

$\rho\gg 1$ の時　$L.y(t)\fallingdotseq M\mu t-\dfrac{M}{\rho}$　となり

$\rho\ll 1$ の時　$L.y(t)\fallingdotseq M\lambda t-M\rho$　となる。

$\rho\ll 1$ の時は処理率が大きい，すなわちフィイルの保持時間が短かい時，プログラムファイルは空で到着呼はすぐに二次ファイルに掃き出され，$\rho\gg 1$ の時プログラムファイルは一ぱいで，ここから到着率 $M\mu$ で二次ファイルに

掃き出されることを示している。(16)式をもとに図—2でこれを検討すると $L\cdot y(t)$ はよくあてはまっている。

2.2. 無限呼源モデル

図—3に無限呼源モデルを示す。プログラムセグメント総数は無限大，二次記憶域容量を無限大とする。システムの初期条件は両方のエリアも空とする。記号等は，2.1の定義と同様とする。

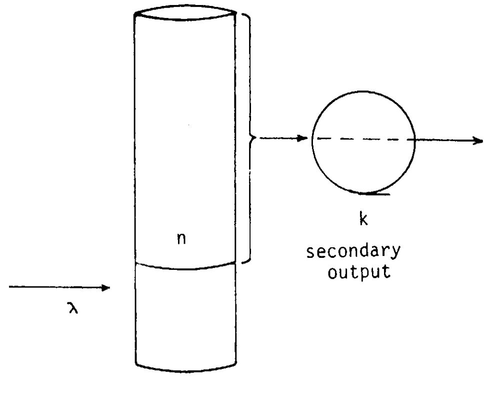

Fig. 3 Infinite queue model ($M/M/\infty$)

システムに関する式は(17)〜(20)式である。

$$P'_{0,0}(t) = -\lambda P_{0,0}(t) \tag{17}$$

$$P'_{n,0}(t) = -(\lambda+n\mu)P_{n,0}(t)+\lambda P_{n-1,0}(t), \quad n \geqq 1 \tag{18}$$

$$P'_{n,k}(t) = -(\lambda+n\mu)P_{n,k}(t)+\lambda P_{n-1,k}(t) + (n+1)\mu P_{n+1,k-1}(t) \qquad n \geqq 1,\ k \geqq 1 \tag{19}$$

$$P'_{0,k}(t) = -\lambda P_{0,k}(t)+\mu P_{1,k-1}(t) \tag{20}$$

母函数 $G(x,y,t) = G(x,y) \equiv \sum_{n=0}^{\infty}\sum_{k=0}^{\infty} P_{n,k}(t)y^k x^n$ とおいて(17)〜(20)式をあらわせば，

$$\frac{\partial}{\partial t}G(x,y,t) = -\lambda(1-x)G(x,y,t) + \mu(y-x)\frac{\partial}{\partial x}G(x,y,t) \tag{21}$$

(21)式の特性方程式は

$$\frac{dt}{1} = \frac{dx}{-\mu(y-x)} = -\frac{dG}{\lambda(1-x)G} \tag{22}$$

(22)式の第一，二項より

$$c_1 = e^{-\mu t}(x-y) \tag{23}$$

第二，三項より

$$G = c_2 e^{\rho x}(x-y)^{\rho(y-1)} \tag{24}$$

ここで c_1，c_2 は積分定数である。

(21)式の一般解は

$$G(x,y,t) = e^{\rho x}(x-y)^{\rho(y-1)}\phi\{(x-y)e^{-\mu t}\} \tag{25}$$

ここで $\rho = \lambda/\mu$，ϕ は任意函数である。

初期条件より ϕ を決定すれば，

$$G(x,y,0) = 1 = e^{\rho x}(x-y)^{\rho(y-1)}\phi\{(x-y)\}$$

$x-y = u$ とおけば

$$\phi\{u\} = e^{-\rho(u+y)}u^{\rho(1-y)}$$

$$\therefore\quad \phi\{(x-y)e^{-\mu t}\} = e^{-\rho\{y+(x-y)e^{-\mu t}\}}(x-y)^{\rho(1-y)} \cdot e^{\mu t+(1-y)\rho}$$

$$\therefore\quad G(x,y,t) = e^{\rho(1-e^{\mu t})x}e^{\rho\{\mu t+e^{-\mu t}-1\}y}e^{-\lambda t} \tag{26}$$

(26)式が(17)〜(21)式に関する解である。

$P_{n,k}(t)$ は(26)式の x^n，y^k の係数であり，

$$P_{n,k}(t) = \frac{\rho^n A^n}{n!}\frac{\rho^k(\mu t-A)^k}{k!}e^{-\lambda t} \tag{27}$$

ここで $A = 1-e^{-\mu t}$

(27)式は(17)〜(21)式をみたす，又 $\sum_{k=0}^{\infty} P_{n,k}(t) = \frac{\rho^n A^n}{n!}e^{-\rho A}$ は良く知られた $M/M/\infty$ 待ち行列における確率である。

平均値に関して

$$L.y(t) = \frac{\partial}{\partial y}G(1,y,t)\Big|_{y\to 1} = \lambda t-\rho A \tag{28}$$

$$Lx.(t) = \frac{\partial}{\partial x}G(x,1,t)\Big|_{x\to 1} = \rho(1-\rho^{-\mu t}) \tag{29}$$

$L\cdot y(t)$ と $Lx.(t)$ を図—4に示す。

(29)式はよく知られた結果である。(28)式で時間が経過すれば $L\cdot y(t) \fallingdotseq \lambda t-\rho$ となり，図—4でそれが認められる。

又，統計的平衡でプログラムファイル容量が同じになるような有限，無限呼源モデルでは，$L\cdot y(t)$ はほぼ同じであることもわかった。

3. 検　討

本文では主として平均値について結果がえられた。二つのモデルにおいて二次記憶容量は時間の経過とともに直線に近づくが，λt に比例している。このことは二次記憶域の容量は学生が計算機システムにアクセスする時間間隔によっていることがわかる。すなわち計算機にアクセスする利用者の様態をつかむことが大切である。次に統計的平衡状態でプログラムファイル容量が同じになるような有限，無限呼源の二つのモデルは $L.y(t)$ はほぼ同じであることもわかった。図-5はそれを示す。

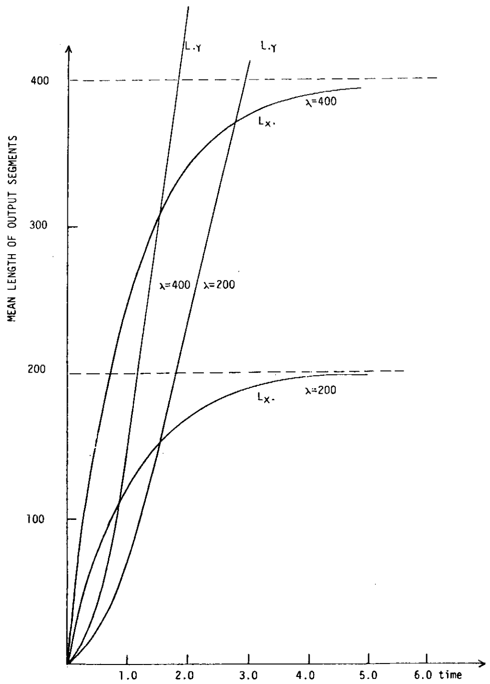

Fig. 4 Mean length of $M/M/\infty$ model ($L_{x\cdot}$ and $L_{\cdot y}$)

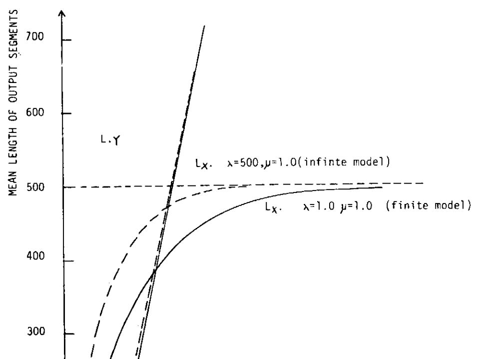

Fig. 5 Mean length ($L_{x\cdot}$, $L_{\cdot y}$)

立ち上がりでは有限呼源モデルの方が，早く，プログラム容量が大きくなる。なおここで得られた結果は，プログラムファイル容量が使用者に対して充分なだけスペースがある場合であり，本文の結果はかぎられたスペース割り当ての場合の目安をつける場合に役立つであろう。プログラム容量が小さい場合の結果は，文献 10）にみられるように非常に複雑になることが予想される。

謝　　辞

本文をまとめるにあたり，オンラインシステム開発にたずさわったセンター夜間運転員諸君と図画作成を行なっていただいたセンター野田事務官に謝意を表すとともに，センターシステムの開発体制を作っていただいた磯センター長に深く感謝いたします。

参 考 文 献

1）本間：〝待ち行列の理論″，理工学社.
2）鈴木：〝待ち行列″，裳華房.
3）T. L. Saaty: "Elements of Queuing Theory with Application", McGraw-Hill, 1961.
4）L. Kleinrock: "Queuing System Theory Vol. I", John Wiley & Sons, 1975.
5）A. T. Bharucha-Reid: "Elements of the Theory of Markov Processes and their Application".
6）N. T. J. Boiley: "The Elements of Stochastic Processes", John Willey & Sons, 1963.
7）H. Greenberg and I. Greenberg: "The Number Served in a Queue", O. R. S. A, 1965.
8）深川他：〝集団情報処理教育におけるプログラム用ファイル容量設定に関する一考察″，九工大研究報告（工学），No. 36, 1978.
9）深川他：情報処理教育のためのプログラムファイル容量の時間変動について″，九工大研究報告（工学），No. 37, 1978.
10) T. L. Saaty: "Time-Dependent Solution of the Many Server Poisson Queue", O. R. S. A., 1960.
11) N. M. Mirasol: "The Output of an $M/G/\infty$ Quening System is Poisson", Opns. Res. 11, 1963.
12）寺本他訳：〝生物学における確率過程の理論″，参業図書，1978.