ショートノート

等負荷ノードをもつ待ち行列網の高速計算法†

張　愛　英††　紀　一　誠†††

短期間，少工数で適確な性能評価を実行するための効果的技法として，BCMP 型待ち行列網が近年脚光を浴び，ソフトウェア・パッケージの開発も数多く報告されている．本稿では，等負荷ノード群を含む BCMP 型待ち行列網の計算法の高速化について示す．等負荷条件とは，情報処理システムの設計の初期段階でよく用いられるもので，システム内のある n 台の装置に関して等しい負荷（すなわち，JOB 当りのアクセス回数が各装置均等に割り振られる）を仮定することを意味している．等負荷条件をもつ待ち行列網については，等負荷ノードのたたみこみ計算に代えて，仮想的なノードのたたみこみ計算により，大幅に数値計算量を削減することができる．

1. まえがき

BCMP 型待ち行列網モデルは，迅速で的確に情報処理システムの性能評価を実行できる効果的なモデルとして近年発展を続け，ソフトウェア・パッケージの開発も数多く報告されている．待ち行列網モデルでは網内客数やノードの数が増加するにつれ数値計算量は飛躍的に増大する．したがって，待ち行列網モデルの実用化に際しては効率的な計算法の研究や開発が欠かせない．

本稿では，BCMP 型待ち行列網に基礎を置く性能評価用ソフトウェア・パッケージ QM-X[3] の使用経験から得られた問題の一つである，等負荷ノードをもつ待ち行列網の計算を高速化する方法について示す．

システム設計の初期段階では未検討，未確定な項目が数多く残されるため，ある装置群（たとえばデータベース格納用 DISK 装置群）に属する個々の装置当りのアクセス回数を正確には推定できない．そのため，この装置群全体へのアクセス回数を推定し，各装置へはアクセスが均等に出されるという等負荷条件を仮定して性能評価を行う場合が多い．等負荷条件の成り立つ網は等負荷ノードのたたみこみ結果がある仮想的なノードの状態確率に等しくなることを利用して計算の高速化がはかれる．

図1に等負荷ノード群を含むシステムモデルの例を，**図2**に JOB モデルを示す．等負荷条件とは各部分連鎖に属する JOB に閉じて仮定されるもので，連鎖の異なる JOB どうしでは等負荷ノードでもその JOB 当り使用時間は異なっていることに注意が必要である．

2. モデル

N個のノードとL個の閉鎖型部分連鎖をもつ待ち行列網を考える．部分連鎖mに従い網内を移動する客数を K_m，客数ベクトルを $\boldsymbol{K}=(K_1, K_2, \cdots, K_L)$ とする．すべてのノードは可変サービス率をもつタイプ1[1] のノードとする．すなわち，先着順サービス規律をもち，ノード i の客数が j のときにはサービス率 $\hat{\mu}_{i(j)}$ をもつ指数分布に従うサービス時間をもつとする．$\mu_i=\hat{\mu}_{i(1)}$, $\mu_{i(j)}=\hat{\mu}_{i(j)}/\mu_i$ とする．簡単のため，$1\sim n$ を等負荷ノード，$n+1\sim N$ をその他の一般のノードとする．さらに，等負荷ノードはすべて固定サービス率，すなわち $\mu_{i(j)}=1$, $j\geq 1$ をもつとする．

3. 積形式解と正規化定数の計算

本章では，前章のモデルに関する積形式表現とたたみこみによる正規化定数の計算法を準備する．積形式表現の簡略化のため，$\boldsymbol{a}, \boldsymbol{x}$ をL次元ベクトルとし，次の表記法を定義しておく．

$$|\boldsymbol{x}|=x_1+x_2+\cdots+x_L$$

$$\boldsymbol{x}!=x_1!\,x_2!\cdots x_L!$$

$$\boldsymbol{a}^{\boldsymbol{x}}=a_1^{x_1}a_2^{x_2}\cdots a_L^{x_L}$$

ノード i に存在する部分連鎖に従う客数を x_{im} とし，ノード i の状態ベクトルを $\boldsymbol{x}_i=(x_{i1}, x_{i2}, \cdots, x_{iL})$，網全体の状態ベクトルを $\boldsymbol{x}=(\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_N)$ とする．

† Computational Techniques for Queueing Networks Including Homogeneous Loading Nodes by Ai Ying Zhang (North China Institute of Computing Technology) and Issei Kino (C & C Systems Research Labs., NEC Corporation).

†† 中国北京華北計算技術研究所
††† 日本電気(株) C&C システム研究所応用システム研究部

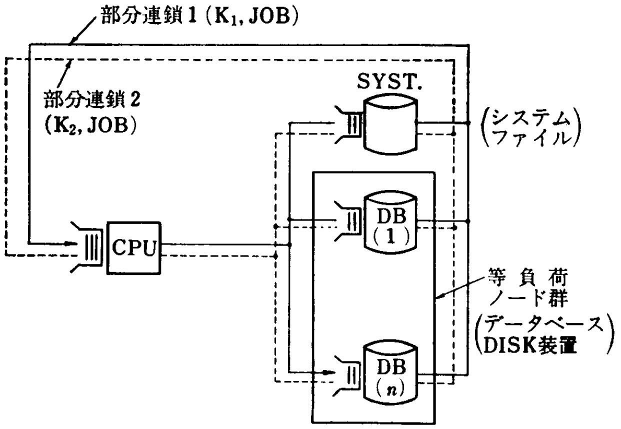

図 1 等負荷ノード群を含むシステム・モデルの例
Fig. 1 A system model including homogeneous loading nodes.

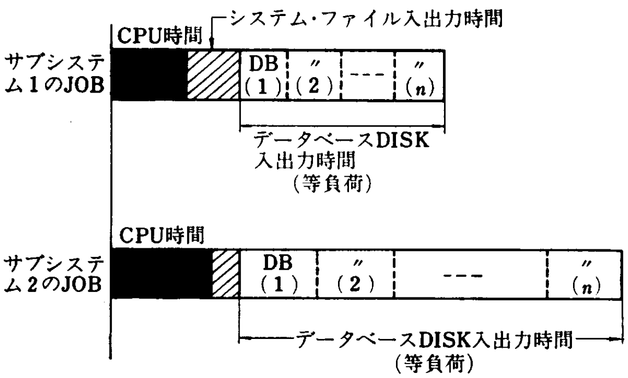

図 2 JOB 当りの資源使用時間 (JOB 特性) の例
Fig. 2 JOB characteristics of each subchain.

また，部分連鎖mに従う客のノードiへの相対平均訪問回数（部分連鎖 P_m を係数とする連立方程式の解として定数倍を除いて一意に定まる）を θ_{im} とし，負荷を $\rho_{im}=\theta_{im}/\mu_i$, 負荷ベクトルを $\boldsymbol{\rho}_i=(\rho_{i1}, \rho_{i2}, \cdots, \rho_{iL})$ とする．

さらに，ノード i の容量係数 $a_i(k)$ を次に定義する．

$$a_i(k)=\{\Pi_{j=1}^{k}\mu_i(j)\}^{-1},\ a_i(0)=1 \tag{3.1}$$

このとき，網の定常状態確率は次のような積形式に表現できることが知られている[1]．

$$P(\boldsymbol{x})=\Pi_{i=1}^{N}q_i(\boldsymbol{x}_i)/g(\boldsymbol{K}) \tag{3.2}$$

ここに，$g(\boldsymbol{K})$ は網の全状態にわたる状態確率の和が1なる条件から定まる正規化定数，$q_i(\boldsymbol{x}_i)$ は次式で与えられるノード i が状態 $\boldsymbol{x}_i$ である（正規化されない）状態確率である．

$$q_i(\boldsymbol{x}_i)=a_i(|\boldsymbol{x}_i|)\frac{|\boldsymbol{x}_i|!}{\boldsymbol{x}_i!}\boldsymbol{\rho}_i^{\boldsymbol{x}_i} \tag{3.3}$$

さらに，正規化定数はたたみこみ演算 $*$[4] を用いて次のごとくに計算できることが知られている[2),4)]．

$$g(\boldsymbol{K})=(q_1*q_2*\cdots*q_N)(\boldsymbol{K}) \tag{3.4}$$

4. 高速計算法

等負荷条件は，$\boldsymbol{\rho}=(\rho_1, \rho_2, \cdots, \rho_L)$ とし，次のごとくに表現される．

$$\boldsymbol{\rho}_1=\boldsymbol{\rho}_2=\cdots=\boldsymbol{\rho}_n=\boldsymbol{\rho} \tag{4.1}$$

また，仮定より等負荷ノードはすべて $a_i(k)=1$ である．

これらの条件を利用して，(3.4)のたたみこみ計算を高速化できる．本章では高速化に必要な二つの Lemma を導き，それらに基づく計算アルゴリズムを示す．

等負荷ノードに関するたたみこみ結果$q^{(n)}(\boldsymbol{x})$ および $[x]_n$ を (4.2), (4.3) に定義しておく．

$$q^{(n)}(\boldsymbol{x})=(q_1*q_2*\cdots*q_n)(\boldsymbol{x}),\ \boldsymbol{0}\leq\boldsymbol{x}\leq\boldsymbol{K} \tag{4.2}$$

$$[x]_n=x(x+1)\cdots(x+n-1),\ [x]_0=1 \tag{4.3}$$

このとき，次の Lemma 1 が成り立つ．

Lemma 1. $\boldsymbol{0}\leq\boldsymbol{x}\leq\boldsymbol{K}$ で次が成り立つ．

$$q^{(n)}(\boldsymbol{x})=\frac{[|\boldsymbol{x}|+1]_{n-1}}{(n-1)!}\frac{|\boldsymbol{x}|!}{\boldsymbol{x}!}\boldsymbol{\rho}^{\boldsymbol{x}} \tag{4.4}$$

証明） 数学的帰納法による． 1) $j=1$ のとき与式の成立は明らか． 2) $j\,(1\leq j\leq n-1)$ に対して成立を仮定．このとき，$q^{(j+1)}(\boldsymbol{x})=(q^{(j)}*q_{j+1})(\boldsymbol{x})$ は次．

$$q^{(j+1)}(\boldsymbol{x})=\frac{1}{(j-1)!}\frac{\boldsymbol{\rho}^{\boldsymbol{x}}}{\boldsymbol{x}!}\sum_{y=0}^{|\boldsymbol{x}|}y![y+1]_{j-1}(|\boldsymbol{x}|-y)! \cdot\sum_{|\boldsymbol{y}|=y}\frac{\boldsymbol{x}!}{\boldsymbol{y}!(\boldsymbol{x}-\boldsymbol{y})!} \tag{4.5}$$

ここで次の関係が成り立つことに注意する．

$$\binom{|\boldsymbol{x}|}{y}=\sum_{|\boldsymbol{y}|=y}\frac{\boldsymbol{x}!}{\boldsymbol{y}!(\boldsymbol{x}-\boldsymbol{y})!} \tag{4.6}$$

$$\textstyle\sum_{r=1}^{j}[r]_{k+1}=(j+k+1)!/(k+2)(j-1)! \tag{4.7}$$

(4.6), (4.7) を用い (4.5) を整理することにより，$j+1$ のときの与式の成立が示される．（証明終り）

(3.4)および Lemma 1 からただちにわかるように，等負荷ノードに関するたたみこみ結果は容量係数 $a^{(n)}(k)$ が次式で与えられるノードの状態確率に等しい．

$$a^{(n)}(k)=[k+1]_{n-1}/(n-1)! \tag{4.8}$$

したがって，(3.4)の計算に際しては，等負荷ノー

ドのたたみこみに代えて，(4.8)に示される容量係数をもつ仮想的なノードを設定し，次のごとくに計算することにより計算の大幅な高速化を実現することができる．

$$g(\boldsymbol{K})=(q^{(n)}*q_{n+1}*\cdots*q_N)(\boldsymbol{K}) \tag{4.9}$$

次に $q^{(n)}(\boldsymbol{x})$ の実現に必要な次の Lemma 2 を準備する．

ただし $\boldsymbol{e}_m$ はm方向の単位ベクトルとする．

Lemma 2. $\boldsymbol{0}\leq\boldsymbol{x}\leq\boldsymbol{K}$ で次が成り立つ．

$$q^{(n)}(\boldsymbol{x})=\frac{|\boldsymbol{x}|+n-1}{|\boldsymbol{x}|}\sum_{m=1}^{L}\rho_m q^{(n)}(\boldsymbol{x}-\boldsymbol{e}_m) \tag{4.10}$$

証明） (4.4)より容易に確かめられる．

(4.10)を用いて $q^{(n)}(\boldsymbol{x})$ を効率的に計算するためのアルゴリズムを以下にプログラム風に示す．ただし，For $\boldsymbol{x}$: c は条件 c を満たす可能な $\boldsymbol{x}$ の組合せすべてについての実行を指示するものとする．

［アルゴリズム Q］

Dimension $q(K_1, K_2, \cdots, K_L)$

$q(\boldsymbol{0})\leftarrow 1$

For $k=1$ to $|\boldsymbol{K}|$

$a=(k+n-1)/k$

For $\boldsymbol{x}$: $|\boldsymbol{x}|=k$

$$q(\boldsymbol{x})\leftarrow a\times\left\{\sum_{m=1}^{L}\rho_m q(\boldsymbol{x}-\boldsymbol{e}_m)\right\}$$

Next $\boldsymbol{x}$

Next k

以上の［アルゴリズムQ］の実行が完了した時点で記憶領域 $q(\boldsymbol{x})$，$\boldsymbol{0}\leq\boldsymbol{x}\leq\boldsymbol{K}$ 上に (4.4) に示される $q^{(n)}(\boldsymbol{x})$ が実現されている．

5. 混合型待ち行列網への拡張[4]

前章の結果を混合型待ち行列網へと拡張することは容易である．2章に定義したモデルに加えて，$L+1, L+2, \cdots, M$ なる開放型部分連鎖の存在する網を考える．

各開放型部分連鎖に従う客の網への到着はパラメータ λ_m のポアソン到着とする．ノード i への連鎖mに従う客の到着率は $\lambda_m\theta_i$ となる．ただし，θ_i は開放型部分連鎖 P_m を係数とする連立方程式の解であり，その存在は仮定されるものとする．さらに，$e_m=\lambda_m\theta_i/\mu_i$ とし，$\rho_{i0}=\sum_{m=L+1}^{M}e_m$ とする．このとき，ρ_{i0} は開放型部分連鎖に従うすべての客からかかるノード i への負荷を意味している．等負荷条件は，閉鎖型部分連鎖に関する等負荷条件 (4.1) に加えて開放型部分連鎖に関する次の条件が加わる．

$$\rho_{10}=\rho_{20}=\cdots=\rho_{n0}=\rho_0 \tag{5.1}$$

このとき，閉鎖型部分連鎖に属する客だけに関する周辺分布 $q^{(n)}(\boldsymbol{x})$ に関して，次の Lemma 3 が成り立つ．

Lemma 3. $\boldsymbol{0}\leq\boldsymbol{x}\leq\boldsymbol{K}$ で次が成り立つ．

$$q^{(n)}(\boldsymbol{x})=\frac{[|\boldsymbol{x}|+1]_{n-1}}{(1-\rho_0)^{(|\boldsymbol{x}|+n)}(n-1)!}\frac{|\boldsymbol{x}|!}{\boldsymbol{x}!}\rho^{\boldsymbol{x}} \tag{5.2}$$

証明） 略．(文献 4) 参照)

(5.2)をもとに前章同様計算の高速化が可能である．

6. む す び

等負荷ノードをもつ待ち行列網に関する計算の高速化について示した．待ち行列網を情報処理システムの性能評価のためのモデルとして実用化するためには，システムの特徴をうまく取り入れ計算の効率化をはかることが欠かせない．本研究結果は性能評価用ソフトウェア・パッケージ QM-X に採り入れられ実用化される予定である．

謝辞 本研究を行う機会と多くの支援を与えていただいた，C & C システム研究所 三上徹所長代理，応用システム研究部 竹谷誠研究課長，情報処理営業支援本部 高橋晃本部長，第1システム支援部篠沢昭二部長，金森吾一課長に感謝いたします．

参 考 文 献

1) Baskett, F. et al.: Open, Closed and Mixed Networks with Different Classes of Customers, *J. ACM*, Vol. 22, No. 2, pp. 248-260 (1975).
2) Reiser, M. and Kobayashi, H.: Queueing Networks with Multiple Closed Chains: Theory and Computational Algorithms, *IBM J. Res. Dev.*, Vol. 19, No. 3, pp. 283-294 (1975).
3) 納富，北浦：性能予測ツール QM-X とその応用，情報処理学会，計算機システムの制御と評価研究会資料，17-8 (1982).
4) 紀 一誠：混合型待ち行列網の計算アルゴリズム，同上，17-4 (1982).

(昭和 58 年 5 月 26 日受付)
(昭和 58 年 7 月 19 日採録)