

資源要求のある待ち行列網のモデル化の一提案

5L-2

木下 俊之  
(株)日立製作所  
システム開発研究所

高橋 幸雄  
東京工業大学

1. はじめに

計算機の性能評価手法のひとつに待ち行列網による解析の手法がある。これは対象とする計算機のCPUやI/O装置などの構成要素を、待ち行列網でモデル化し、装置のビジー率、滞留ジョブ数、応答時間などを解析するものである。

一方、計算機の処理には、ファイルのデータ項目の更新のように、ひとつのジョブがそれをアクセスしている間は、他のジョブからのアクセスを禁止するケースが多く存在する。このような対象をここでは共有資源、または単に資源と呼ぶ。この資源へのアクセスの競合は、計算機の性能に大きな影響を与える。しかし従来の待ち行列網では、この資源を厳密に扱うことが出来なかった(1,2)。そこで本発表では待ち行列網に資源の概念を付加し、マルコフ連鎖にもとづく状態方程式を厳密に解くことにより、資源の計算機性能に及ぼす影響を解析した。

待ち行列網としては、計算機の性能評価に広く用いられている閉のセントラルサーバモデル(Central Server Model: CSMと略す)を取り上げる。これは単一のCPUと複数のI/O装置から成り、CPUでの処理を中断されたジョブは次にI/O装置の何れかに入って処理を受け、その後さらにCPUでの処理を受けるといふ、計算機内部での基本的な動作をモデル化したものである。

2. 資源要求のあるセントラルサーバモデル

資源として、I/O装置上に散在したファイルデータを、複数のジョブが排他的に使用するという状況を考える。一般にCSMでは、CPU処理の完了時に次に遷移するノードを確率的に決めるが、資源要求のあるCSMでは、これと同時に資源を取得するか否かも確率的に決定する。それが資源取得要求だった場合、既に他のジョブによって資源が保持されていれば、資源を要求したジョブは資源待ちキューに入って資源が解放されるのを待つ。その後、資源解放時に資源を取得して、後続の処理を実行する。(図1参照)

以下、次の記号を用いる。

N: 総ジョブ数、 K: ジョブクラス数

$N_k$ : ジョブクラスkのジョブ数 ( $k=1 \sim K, \sum_{k=1}^K N_k = N$ )

M: I/Oノード数(全ノード数は M+1)

(CPUノード:  $m=0$ , I/Oノード:  $m=1 \sim M$ )

$\mu_k$ : CPUノードでのクラスkのジョブのサービス率

$\nu_{km}$ : I/Oノードmでの

( $m=1 \sim M$ )

$p_{km}^{gn}, p_{km}^{ng}, p_{k0}^{gn}, p_{k0}^{ng}$ : クラスkのジョブのCPUノードの前後での資源取得/解放の遷移確率。右上の二重添え字の前者はCPU到着時の資源確保の状態を、後者はCPU処理完了時の資源確保の状態を表わし、'g'は資源を保持していることを、'n'は保持していないことを表わす。

$p_{km}^{gn}, p_{km}^{ng}$ : クラスkのジョブの

CPUノードからノードm

( $m=0 \sim M$ )への遷移確率。

右上の添え字'g','n'は、

CPUノードでの資源の有

無を表わす。

(ただし、 $p_{k0}^{gn}$ はジョブの

完了確率、また  $p_{k0}^{ng}=0$ )

資源は複数種類を考慮することができるが、ここでは1種類とする。また全てのノードおよび資源待ちキューで、FCFS (First Come First Served)スケジュールが行なわれる。サービス時間は負の指数分布に従い、互いに独立である。

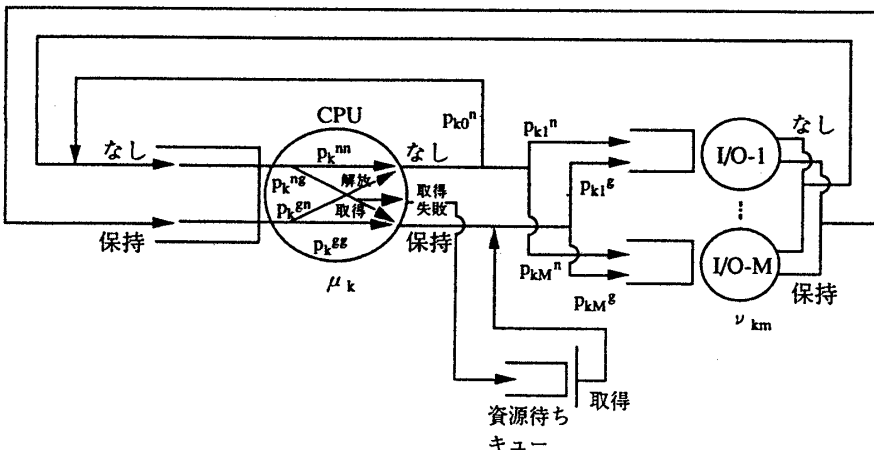


図1 資源要求のあるセントラルサーバモデルの構成 (資源数=1の場合)

A Queueing Network Model with Resource Requirements  
Toshiyuki KINOSHITA, Hitachi Systems Development Laboratory  
Yukio TAKAHASHI, Tokyo Institute of Technology

3. 数値解析

K=1, M=2 の場合について定常状態方程式を解いて、厳密な数値解析を行なった。N=2 の場合、

$$\nu_1 P(1,1a,,) = p_{0n} p_{1n} \mu P(1,1,,) + p_{0n} p_{1n} \mu P(1a,1,,) + p_{0n} p_{1n} p_{1n} \mu P(1a,,1)$$

といった方程式が現われる(ここで記号(1a,,1)は、CPUノードに1個のジョブが資源を保持して処理中で、I/Oノード1,2にはジョブはなく、資源待ちキューに1個のジョブが存在する状態を表わす)。この右辺の第3項目は、資源を保持してCPUで処理を受けていたジョブがCPU処理完了時に資源を解放したので(その後、I/Oノード1に遷移した)、資源待ち中のジョ

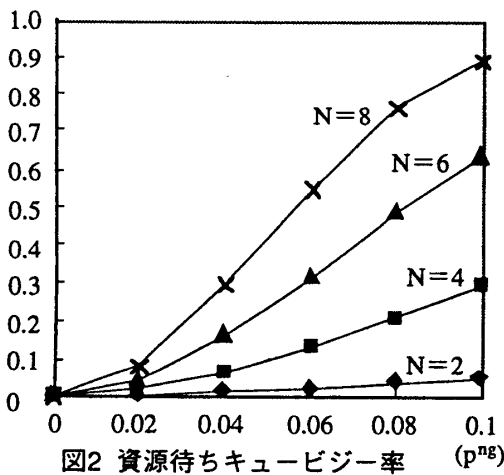


図2 資源待ちキュービジー率 (p<sup>ng</sup>)

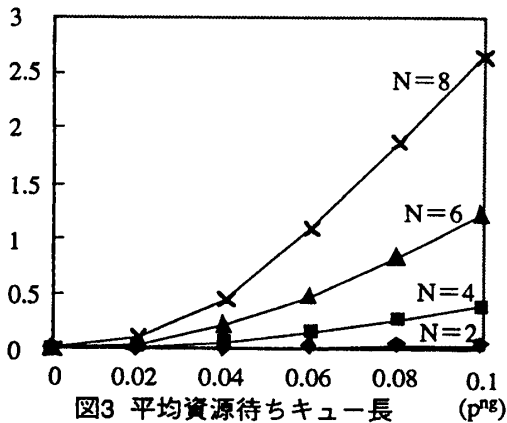


図3 平均資源待ちキュー長 (p<sup>ng</sup>)

表1 ジョブの応答時間の1次近似 (N=6)

p <sup>ng</sup>	応答時間 (t)	微分係数 (a)	1次近似 (t <sub>0</sub> + a · p <sup>ng</sup> )	相対誤差
0.00	17.601	39.367	17.601	0.00%
0.02	18.435		18.388	-0.25
0.04	19.382		19.176	-1.06
0.06	20.481		19.963	-2.53
0.08	21.755		20.750	-4.62
0.10	23.200		21.538	-7.16

ブが資源を取得してI/Oノード1に遷移し、全体として左辺の状態に遷移したことを表わしている。

設定したパラメータは、次の通りである。(ジョブクラスを表わす 'k' の添え字は省略する)

$$\mu = 2.0, \nu_1 = \nu_2 = 1.0$$

$$p_{1n} = 0.4, p_{2n} = 0.6, p_{0n} = 0.5, p_{0n} = 0.5$$

従っていったん資源を取得すると、平均2回のCPU処理とI/O処理を受けてから資源を解放する。p<sub>0n</sub> と p<sub>0n</sub> は、ひとつのジョブが完了までにCPU処理を受ける回数(の平均値が)常に一定値5となるように、次の様に対にして設定した。

$$(p_{0n}, p_{0n}) = (0.0, 0.200), (0.2, 0.208), (0.4, 0.216), (0.6, 0.224), (0.8, 0.232), (1.0, 0.240)$$

以上の結果を、図2、図3に示す。

また資源取得確率 p<sup>ng</sup> が小さい場合は、資源要求がぶつかる確率は非常に小さいと考えられる(p<sup>ng</sup> の2次の無限小)のでこれを無視できる。すると、資源を保持しているジョブと保持していないジョブが異なるジョブクラスに属すると考えることで、積形解を持つ待ち行列網となる。これを解いて、p<sup>ng</sup>=0 の近傍における1次近似を行なった。(表1参照)

結果は、次の様にまとめられる。

- (1) 資源取得確率 p<sup>ng</sup> ≤ 0.04(資源取得がI/Oアクセスの25回に1回以下)、ジョブ数 N ≤ 8 の範囲では、資源が性能ボトルネックとはならないが、p<sup>ng</sup> ≥ 0.1、N ≥ 8 になると、資源がボトルネックとなる(N=8の時、資源待ちキューのビジー率は90%に達し、8個のジョブのうち約1/3が資源待ちキュー中にある)。
- (2) 資源取得確率 p<sup>ng</sup> が小さい範囲では、資源要求のぶつかりを無視したモデルで、精度の良い近似が得られる。p<sup>ng</sup> ≤ 0.8、N ≤ 6 の範囲で、ジョブの応答時間の相対誤差は5.0%以内である。

4. おわりに

これまで近似のみであった資源要求のある待ち行列網を、厳密に解析した。さらに資源取得確率が小さい場合の簡便な近似について、精度検証した。

今後は、資源が複数の場合などを扱う予定である。

[参考文献]

- 1) C.H.Sauer : Approximate Solution of Queueing Networks with Simultaneous Resource Possession, IBM Journal of Res.& Dev., Vol.25, No.6, (1981.11)
- 2) 池原: パッシブ・サーバをもつネットワーク型待ち行列を用いた計算機の性能評価法、情報処理学会論文誌, Vol.20, No.2, (1979.3)