

逐次アクセス資源のある計算機システムの 待ち行列網による近似評価の一手法

木下 俊之[†] 高橋 幸雄[‡]

ファイルへのアクセスでは、一致性を保つためにあるジョブがアクセスしている間は他のジョブからのアクセスは禁止される場合がある。この逐次アクセス資源へのアクセスの衝突は、システム性能に大きな影響を及ぼす。これの性能評価には、通常の待ち行列網に資源と資源待ち行列を付加した網をマルコフ連鎖でモデル化し、その平衡方程式を数値的に解く方法が考えられる。しかしこの方法は、モデルの状態数が増えた時に計算量が増加し数値計算が困難になるという問題がある。そこで本報告ではこの数値計算上の困難を回避する近似手法を提案する。近似の考え方は、網の状態を各資源を要求/獲得しているジョブ数ごとに設定というもので、これにより状態数を大幅に削減し計算可能な範囲を拡大できる。この近似モデルについて、数値実験により状態数の削減効果と近似精度を検証した。

An Approximate Analysis of a Queueing Network Modeling for Computer Systems with Resource Requirements

TOSHIYUKI KINOSHITA[†] and YUKIO TAKAHASHI[‡]

In computer systems, job conflicts occur at accessing an *exclusively-used resource*. A typical example is conflicts at accessing a group of updatable files. These conflicts affect the performance of the system significantly. In our previous work, we introduced a queueing network model to analyze the influence of the conflicts on the performance of the computer systems. This analysis requires a large computational burden since the stationary distribution has to be calculated for a Markov chain with a large number of states. In this paper, we propose an approximate method of the model which reduces the number of states and our computation efforts. Its accuracy is analyzed by numerical experiments.

1. まえがき

計算機システム上の更新を伴うファイル処理では、ファイル・データの一貫性を保つため、ひとつのジョブの処理中は他のジョブからのアクセスが禁止される場合がある。このファイルへのアクセスの衝突は、計算機システムの性能に大きな影響を及ぼすにもかかわらず一般の待ち行列理論では取り扱うことが困難である。著者らはこの逐次アクセス資源を持つ計算機システムを性能評価する手法として、これを待ち行列網で記述しそのマルコフ連鎖の平衡方程式を数値的に解く方法を提案した(木下 [2])。この方法は逐次アクセス資源のある待ち行列網を必要だけ精密に解析できる反面、平衡方程式を数値的に解くので、待ち行列網の状態数が大きくなると数値計算が困難になるという問題をはらんでいる。

本報告では、この数値計算の手間と必要なメモリ

量を大幅に削減する近似手法を提案する。資源要求のある待ち行列網の解析が難しいのは、資源へのアクセスのぶつかりという非確率的な要因によって待ちが発生するため、網が積形解を持たなくなるためである。そこで提案する近似手法では、資源の要求/解放がない間(引き続き2つの要求/解放の間)の網のふるまいと、ジョブが資源を要求/解放するふるまいを分けて解析する。前者は資源のない通常の待ち行列網の積形解(Baskett [1])で近似し、後者はマルコフ連鎖を構成してその定常分布を数値的に求めることにより解析する。この近似手法でも後者でマルコフ連鎖の平衡方程式を数値的に解く必要があるが、その状態数は木下 [2] で扱う状態数に比べてはるかに少なく、計算可能な範囲を大幅に拡大できる。

2. モデルの記述

(1) 基本構成(図1のイメージ図参照)

網はひとつのCPUノードとM個のI/Oノードからなる閉セントラル・サーバ・モデルとし、ジョブ数をNとする。各ノードでのサービス分布は互

[†] (株)日立製作所システム開発研究所
Systems Development Laboratory, Hitachi, Ltd.
[‡] 東京工業大学
Tokyo Institute of Technology

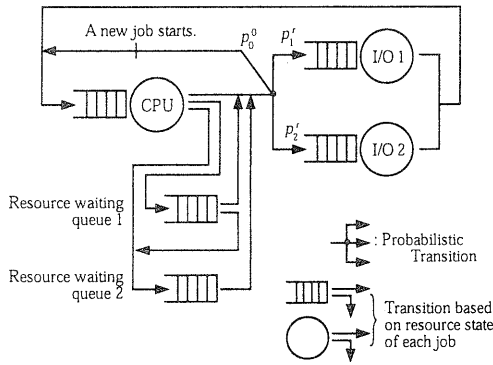


図 1: 資源要求のあるセントラル・サーバ・モデルのイメージ図 (資源が 2 個の場合)

いに独立な指数分布とし, CPU ノードでのサービス率を μ_0 , I/O ノード m でのサービス率を μ_m とする ($m = 1, 2, \dots, M$)¹. 全てのノードでジョブは FCFS (First Come First Served) 規律でスケジュールされる.

(2) 資源と資源待ち行列

(a) 資源の獲得/解放

F を資源数とし, 資源を S_f ($f = 1, 2, \dots, F$) で表わす. ジョブは, CPU ノードでのサービスの終了時に確率的に資源を要求/解放する.

ジョブの「資源状態」とは, その時点でジョブが必要としている資源, すなわちジョブが要求中または獲得中の資源のことである. したがってジョブ n の資源状態 r はこのジョブがその時点で必要としている資源の集合として $\{S_{f_1}, S_{f_2}, \dots, S_{f_{F_n}}\}$ ($0 \leq F_n \leq F$) と表わすことができる². このジョブの資源状態は CPU のサービス終了時に確率的に変更され, 資源状態が r_1 から r_2 に移る遷移確率を $p^{r_1 r_2}$ とする.

2 つ以上の資源を要求する際には, デッドロックを回避するために, 資源を獲得する順序が決められている. この資源獲得順序に違反して資源を獲得することは許されない.

(b) 資源待ち行列

ジョブが CPU サービス終了時に資源を要求した

時, もしその資源が既に他のジョブによって占有されている場合は, ジョブは対応する資源待ち行列に入って資源が解放されるのを待つ. 資源待ち行列ではジョブは FCFS 規律でスケジュールされる. 資源が解放されると, 資源待ち行列の先頭のジョブが直ちにこの資源を獲得する.

(3) ノード遷移

ジョブが CPU のサービス終了時に新たな資源を要求しなかったり, 資源を要求して直ちにそれを獲得できたり, 資源待ち行列にいるジョブが必要な資源が獲得できた時は, 次に CPU ノードか I/O ノードのいずれかに進む. ジョブの資源状態が r の時のノード m の選択確率を p_m^r とする. ただし次に CPU ノードに遷移する場合はそのジョブは新たなジョブに生まれ変わったと考える. この CPU ノードへの遷移はジョブが資源を獲得していない時にのみ起こる. すなわち $r > 0$ ならば $p_0^r = 0$ とする.

3. 近似モデル

前節で構成した資源要求のある待ち行列網の状態を δ とし, 定常状態確率 $P(\delta)$ とする. 性能評価に必要なビジー率, スループット, 平均応答時間などはこの $P(\delta)$ から求めることができる. しかしこれを求めるためには, 状態数と同数の未知数を含む平衡方程式 (連立一次方程式) を解かなければならない. この δ の状態数は 4. の表 1 に示すようにジョブ数や資源数が増加すると指数関数的に増加するので, この方法では限界がある. そこで近似モデルを導入してこの計算上の困難を回避することを考える.

近似モデルでは, 網内のジョブのふるまいを「資源の要求/解放がない間 (引き続く 2 つの要求/解放の間) のノード内でのふるまい」と「資源の要求/解放のふるまい」に分離し, 前者は (資源要求のない) 通常の待ち行列網の積形解で近似し, 後者はその資源の要求/解放のふるまいを記述するマルコフ連鎖を構成してその定常分布を求める. このマルコフ連鎖の状態は, δ を次のように集約 (aggregate) して構成する. すなわち各 δ について,

- (a) 各資源状態のジョブがノード内に何個ずつあるか (高々 1 個であるが).
- (b) 各資源状態のジョブが資源待ち行列に何個ずつあるか.

¹ $m = 0$ は CPU ノードを表わすものとする.
² $r = 0$ は資源状態が空集合, すなわち資源を何も必要としない状態を表わす.

といった各ジョブが資源を要求/獲得しているパターンによって集約する。すなわちこのパターンが同じであるような δ を集約して、新しいひとつの状態 γ とする。資源の要求/解放については、この γ を状態とするマルコフ連鎖を考えてその平衡方程式を数值的に解いて定常分布を求める。

この近似モデルでは定常分布 $P(\gamma)$ について平衡方程式を数值的に解く必要があるが、表1に示すようにその状態数(γ の個数)は集約する前の状態数(δ の個数)に比べてはるかに少なく、数値計算の手間(計算量とメモリ量)を大幅に削減できる。

4. 数値実験

(1) パラメータ

近似モデルによる状態数の削減効果と近似精度を調べるために、木下[2]と同一の設定で数値実験を行った。設定したパラメータは、次の通りである。

- (a) ジョブ数: $N = 2, 4, 8, 12$
- (b) I/O ノード数: $M = 2$
- (c) CPU ノードでのサービス率: $\mu_0 = 2.0$
- (d) I/O ノード 1, 2 でのサービス率: $\mu_1 = \mu_2 = 1.0$
- (e) 資源数: $F = 2$
- (f) 資源状態の遷移確率

現実のオンライン実時間システムにおいて排他制御を伴う入出力が全入出力の十数回に1回(約8%)であることから、 $\bar{p}^{01} = 0, 0.02, 0.04, 0.06, \dots, 0.14$ とする。

(i) 資源が1個の場合

$$\begin{pmatrix} \bar{p}^{00} & \bar{p}^{01} \\ \bar{p}^{10} & \bar{p}^{11} \end{pmatrix} = \begin{pmatrix} 1 - \bar{p}^{01} & \bar{p}^{01} \\ 0.5 & 0.5 \end{pmatrix}$$

(ii) 資源が2個の場合

$$\begin{pmatrix} \bar{p}^{r_1 r_2} \end{pmatrix}_{\substack{r_1 = 0, 1, 2, 3 \\ r_2 = 0, 1, 2, 3}} = \begin{pmatrix} \bar{p}^{00} & \bar{q} & \bar{q} & (1 - \sqrt{\bar{p}^{00}})^2 \\ 0.5 & 0.5 & 0 & 0 \\ 0.5 & 0 & 0.5 & 0 \\ 0.5 & 0 & 0 & 0.5 \end{pmatrix}$$

ただし $\bar{q} = \sqrt{\bar{p}^{00}}(1 - \sqrt{\bar{p}^{00}})$ とする。

(g) CPU ノードからの推移確率

CPU ノードから次のノードへの遷移は資源が1個の場合も2個の場合も、またジョブの資源状態に関係なく、次のようにおく。

表 1: 待ち行列網の状態数 ($M = 2$)

ジョブ数 (N)	資源数 $F = 1$		$F = 2$	
	δ	γ	δ	γ
2	21	3	78	12
4	120	5	1,383	55
6	406	7	13,684	154
8	1,035	9	100,077	333

$$(\bar{p}_0^0, \bar{p}_1^0, \bar{p}_2^0) = (p_0^0, p_1^0, p_2^0) = (0.2, 0.4, 0.4)$$

$$(\bar{p}_0^1, \bar{p}_1^1, \bar{p}_2^1) = (p_0^1, p_1^1, p_2^1) = (0.0, 0.4, 0.6)$$

$$(r = 1, 2, 3)$$

(2) 状態の削減効果

厳密モデルと近似モデルの状態数を表1に示す。このように近似モデルの状態数は著しく減少していることが分かる。表1によると $F = 2$ の時に $P(\delta)$ を求めることは $N = 6$ の場合が実用上の限度で、 $N \geq 8$ では困難と考えられるが、 $P(\gamma)$ はこの範囲においても十分に計算可能である。

(3) 近似モデルの精度検証

表2~4に数値解析結果を示す。3.で述べたように、近似モデルでは資源の要求/解放がない間のジョブのノード内でのふるまいを通常の待ち行列網の積形解で近似する。すなわち網を状態 γ に制限をした場合のジョブのふるまいは、定常分布で近似できると仮定している。しかしこれは厳密には正しくなく、網を γ に制限をした場合のふるまいは γ に遷移する直前の状態に依存する(すなわち「定常状態」ではなく「遷移状態」である)。このような定常性を仮定することは網の構造やジョブのふるまいを単純化することにつながり、性能値は負荷が軽い方の値にシフトすると考えられる。近似モデルの誤差は主にこれによると考えられる。

この数値実験例では、資源数が2個の場合に資源待ち行列について誤差率が最大で20%程度に達するケースがあるが、CPU ビジー率、ジョブのスループットや平均応答時間といった資源待ち行列以外の性能値の誤差率は最大で9%程度であり、これは実用上有用な範囲内と言える。

5. むすび

ファイル資源のある計算機システムの性能上の特性を、待ち行列論により解析する際に問題となる数値計算の計算量を削減するための近似モデルを導入し、数値実験により計算量の削減効果と近似精度を確認した。今後は網内の各ジョブのふるまい（特にジョブが資源待ち行列につながるごとの応答時間への影響）について解析していく。

参考文献

- [1] F. Baskett, K.M. Chandy, R.R. Muntz and F.G. Palacios, "Open, Closed, and Mixed Networks of Queues with Different Classes of Customers," J. ACM, vol.22, no.2, pp.248-260, Apl. 1975.
- [2] 木下俊之, 高橋幸雄, "逐次アクセス資源のある計算機システムの待ち行列網によるモデル化と評価法," 電子通信情報学会論文誌 (D-I), vol. J82-D-I, no.6, pp.701-710, Jun. 1999.

表 2: CPU ビジー率 (ρ_i : 厳密モデル, ρ'_i : 近似モデル)

ジョブ数 (N)	資源要求 確率 (p^{01})	資源数 $F = 1$			$F = 2$		
		ρ_1	ρ'_1	$(\rho'_1 - \rho_1)/\rho_1$	ρ_2	ρ'_2	$(\rho'_2 - \rho_2)/\rho_2$
4	0.02	0.761	0.762	0.13 %	0.761	0.762	-0.13 %
	0.08	0.740	0.739	-0.14	0.752	0.754	0.27
	0.14	0.701	0.702	0.14	0.734	0.739	0.68
8	0.02	0.901	0.903	0.22	0.898	0.902	0.45
	0.08	0.843	0.844	0.12	0.878	0.890	1.37
	0.14	0.747	0.758	1.47	0.836	0.840	0.48

表 3: 1 回の資源待ち行列平均滞在時間 (t_i : 厳密モデル, t'_i : 近似モデル)

ジョブ数 (N)	資源要求 確率 (p^{01})	資源数 $F = 1$			$F = 2$		
		t_1	t'_1	$(t'_1 - t_1)/t_1$	t_2	t'_2	$(t'_2 - t_2)/t_2$
4	0.02	4.505	4.053	-10.03 %	3.354	3.881	15.71 %
	0.08	5.370	4.888	8.98	4.699	4.184	-10.96
	0.14	6.186	5.632	8.96	5.004	4.345	-13.17
8	0.02	9.305	10.065	8.17	7.284	8.523	17.01
	0.08	17.961	17.522	-2.44	11.352	10.282	-9.43
	0.14	23.478	21.733	-7.43	14.922	12.014	-19.49

表 4: ジョブの平均応答時間 (T_i : 厳密モデル, T'_i : 近似モデル)

ジョブ数 (N)	資源要求 確率 (p^{01})	資源数 $F = 1$			$F = 2$		
		T_1	T'_1	$(T'_1 - T_1)/T_1$	T_2	T'_2	$(T'_2 - T_2)/T_2$
4	0.02	13.142	13.145	0.02 %	13.137	13.112	-0.19 %
	0.08	13.514	13.533	0.14	13.335	13.069	-1.99
	0.14	14.269	14.245	-0.17	13.610	13.060	-4.04
8	0.02	22.200	22.212	0.05	22.236	22.101	-0.61
	0.08	23.733	23.692	-0.17	22.793	21.798	-4.37
	0.14	26.775	26.369	-1.52	23.861	21.736	-8.91