

福岡工業大学 学術機関リポジトリ

Approximation Method for Assembly-like Queueing Systems with Communication Blocking

メタデータ	言語: jpn 出版者: 公開日: 2021-02-16 キーワード (Ja): キーワード (En): 作成者: 宋, 宇, 上田, 文人 メールアドレス: 所属:
URL	http://hdl.handle.net/11478/00001650

通信型ブロッキングをもつ 組み立て型待ち行列システムの近似解析

宋 宇 (管理工学科)
上 田 文 人 (管理工学科)

Approximation Method for Assembly-like Queueing Systems with Communication Blocking

Yu SONG (Department of Management Engineering)

Fumito UEDA (Department of Management Engineering)

Abstract

We consider queueing systems with more than one assembly-like queues which are used for modelling assembly operation in production systems, and propose an approximation method to evaluate throughputs and stationary probabilities.

Key words : *assembly-like queues, production systems, approximation method*

1 はじめに

生産システムをモデル化し、解析を行なうために、待ち行列理論はよく用いられる。しかし、大規模なモデルは理論的な解析が不可能な上、ノード数が増えるにつれて状態数が指数的に増加するので、マルコフチェーンの解析などによる数値計算も事実上不可能である。このため、近似解法に関する研究が多くなされている。

これらの近似法が適用されるモデルは、ネットワークの構造から分類するとおおむね次のような3種類がある。

(1) 直列型待ち行列モデル。たとえば、[1, 3, 7, 8, 9, 10, 12, 13, 14] など多くの文献でこのようなモデルに対して近似解析が行なわれた。[7]では、Mitra と Mitrani は極小型ブロッキングをもつモ

デルに対して一種の分解法を用いて近似解析を行い、かなり精度の高い近似解を得た。

(2) 図1のような合流構造のもつネットワークモデル。ここでは、ノード1とノード2の出力はノード3の入力となっている。ノード3に到着した客は、ノード1と2のどちらのサービスを終了したかを問わずに同じバッファに並び、ノード3でのサービスを待つ。どちらかのサービスを終了した客がいればノード3がサービスを開始することができる [15, 16]。

(3) 本論文の研究対象となるのは、生産システムにおける組み立て型の生産をとらえるモデルである。ノード3で製品を組み立てるためにはノード1と2か

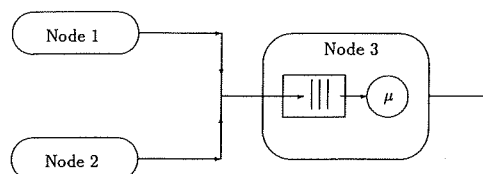


図1：合流構造

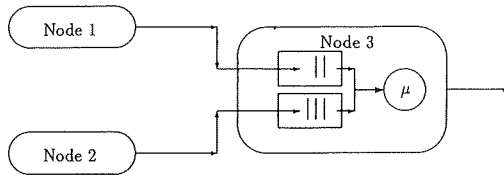


図2：組み立てノード

それぞれ一つずつ部品を取り寄せなければならない。したがって、モデルとしては図2のような組み立て構造をもつネットワークとなる。ノード3において、ノード1のサービスを終了した客とノード2のサービスを終了した客は別々のバッファにならび、ノード3がサービスを始めるために、両方のバッファに客が少なくとも一人ずついなければいけない。

このような組み立て型モデルについての研究のほとんどは、組み立てノードが一つだけのモデルを扱っている。[4]では無限バッファのモデルについて検討され、定常状態に達しないことが確認された。有限バッファのモデルに対しては、[2]では、応答時間分布について検討された。[6, 5]ではスループットや、平均在庫についての近似法が提案された。

本論文は、有限バッファをもつ組み立てノードが複数存在するネットワークモデルを解析の対象とする。有限バッファによって生じるブロッキングについては通信型ブロッキングルールを用いる。このようなモデルに対し、[7]で行なわれた極小型ブロッキングルールの直列型待ち行列モデルについての近似解析法を適用し、近似解を求める。

[7]で提案された近似法は一種の分解法である。そ

こでは、まずシステムをサブシステムに分解し解析する。そして、システム全体の挙動をとらえるために、サブシステム間を関連づける2つの到着率（本論文では、1つの到着率と1つの退去率）を推定する近似式の組み合わせをいくつか提案した。数値実験の結果、そのなかの一つ組み合わせ（以下組み合わせ1と呼ぶ）がもっともよい近似解をもたらすことを示した。

しかし、組み立て型モデルに対して近似解析を行なった結果、もう一つの組み合わせ（以下組み合わせ2と呼ぶ）によってもいい近似解が得られることが分かった。かんばん数などのパラメータによっては、組み合わせ1による近似の方が精度が高い場合と組み合わせ2の方が高い場合の両方がある。

次節ではモデルを定義し、3節では近似法の適用を提案する。そして、4節で数値結果に基づいてこの近似法の有効性を検討する。

2 組み立て型待ち行列ネットワークモデル

図3のようなK個のノードからなるツリー状の待ち行列ネットワークを考えよう。

各ノードは単一窓口であり、ノードkの窓口のサービス時間はサービス率 μ_k の指数分布に従うとする。

我々はこのK個のノードを3つのグループに分ける。

グループ1はノード1からノード K_1 の K_1 個のノードからなり、各ノードの前にはそれぞれ無限大の客のプールがあって、常に客がサービスを待っている。これらのノードでのサービスを終了した客は残りの

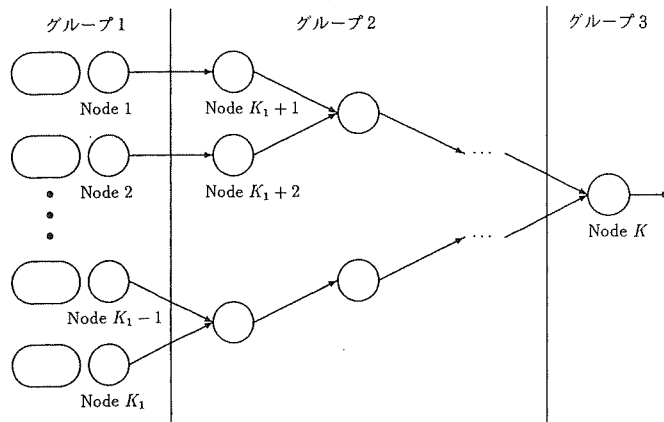


図3：組み立て型待ち行列ネットワークモデル

$K-K_1$ 個のノード (ノード K_1+1, K_1+2, \dots, K) のうちの一つに到着し、そこでのサービスを受ける。

グループ3に含まれるのはノード K だけであり、ここでサービスを受けた客はシステムから退去する。グループ1とグループ3以外のノードはグループ2に属する。

グループ2, 3のノードの入力はグループ1またはグループ2のノードでサービスを終了した客である。ノード k ($k > K_1$) への入力は、ノード j_1, j_2, \dots, j_k ($j_1, j_2, \dots, j_k < k$) からの出力とすれば、ノード k には各ノードから入力のためにそれぞれバッファ j_1, j_2, \dots, j_k を用意する。窓口にも各入力ノードのためにそれぞれ一つの位置が用意されており、すべての位置に客がいる場合に限り、集団でサービスを受ける。空位置が一つでもあれば、ノード k はサービスを始めることができない。

グループ1とグループ2の各ノード k ($k < K$) でのサービスを終了した客が、下流のノードにあるバッファ k に入り、下流のノードでのサービスを待つ。バッファ k のサイズを C_k-1 とする。ノード k で新しい客のサービスを始める前にまずバッファ k の状態を見る。バッファ k がいっぱいになっていれば、客が窓口にとどまってノード k のサービスをブロックする。バッファ k に空きができるとサービスが開始される。このブロッキングルールは通信型ブロッキングと呼ばれている。

[7]で用いられた極小型ブロッキングは直列型待ち行列に対して定義されたものである。そこでは、バッファ k の状態に関係なくノード k でのサービスが始まることができる。サービスの終了時、バッファ k がいっぱいになってもすぐには窓口をブロックしない。まず上流のバッファ $k-1$ にまだノード k のサービスを受けていない客がいるかどうかを調べ、いるならばその客と位置交換してバッファ $k-1$ をあたかもバッファ k の一部のように使用する。そして、ノード k の窓口は新しい客のためにサービスを行なう。バッファ $k-1$ が全部ノード k でのサービスを終了した客で埋まったらはじめてノード k をブロックする。

しかし、ノード k が組み立てノードの場合、サービスを受ける前の客と終了後の客とはかなり異質なものになっているので、上流のバッファもバッファ k の一部として使用する考えはあまり適切ではない。

通信型ブロッキングのシステムは一種のかんばん生産システムとして理解することもできる。すなわち、

バッファ k には C_k 枚のかんばんがあり、バッファに入る客に必ず一枚もたせる。客はバッファ k から窓口に進みサービスを受けている間もそのかんばんを持ち続け、サービスを終了してからかんばんを返す。したがって、バッファ k で待機できる客は C_k によってただか C_k-1 人に制限される。バッファ k のかんばんが無くなったなら、ノード k の窓口がブロックされ、サービスを始めることができない。

バッファ k のかんばんをもつ客の数を n_k とすれば、われわれはベクトル

$$(n_1, n_2, \dots, n_{K-1}), n_k = 0, 1, 2, \dots, C_k$$

をもってシステム全体の状態を表し、マルコフ連鎖でシステムの確率的挙動を記述することができる。定常状態において、バッファ k の周辺確率 $P(X_k = n_k)$ とし、システムのスループットを T とおく。

3 近似法

直列型で、極小型ブロッキングをもつモデルに対して、Mitra と Mitrani は一種の分解法による近似方法を提案した [7]。この節では、その方法を前節で定義した通信型ブロッキングをもつ組み立て型の待ち行列ネットワークモデルへ適用を試みる。

簡潔のために、ノード k ($k > K_1$) への入力は2つ以下のノードからとする。組み立てノードでない場合、入力ノードはノード $u(k)$ で表す。組み立てノードの場合、ノード $u_1(k)$ と $u_2(k)$ はそれぞれノード k へ入力ノードの1つを表し、単なる $u(k)$ は $u_1(k)$ と $u_2(k)$ とのどちらかを意味するとする。誤解の生じない場合、 $u(k), u_1(k), u_2(k)$ をそれぞれ u, u_1, u_2 で表す。

さらに、ノード k の下流のノードをノード $d(k)$ で表し、 $u(k)$ 同様、誤解が生じる恐れがなければ d で表す。

3.1 サブシステム

グループ1, 2の各ノードについては、われわれはノード k ($=1, 2, \dots, K-1$) とその下流のノード $d(k)$ を合わせてサブシステム k としてとらえる。ノード K はそれ自身でサブシステム K をなす。このようなサブシステムにおいては、サブシステム k ($> K_1$) への到着は上流ノードおよびノード k の両方の状態に依存しているため、ポアソン到着とはならない。各サブシステムから退去もポアソン過程ではない。

サブシステム k のスループットを T_k とおけば、定

常状態において各サブシステムスループットはシステム全体のスループットと等しくなるので、

$$(1) T_k = T, k=1, 2, \dots, K$$

さらに、サブシステム k において、ノード k の各バッファ $u(k)$ の周辺定常確率を $P_k(X_u = n_u)$ で表せば、

$$(2) P_k(X_u = n_u) = P(X_u = n_u), n_u = 0, 1, \dots, C_u, \\ k = K_1 + 1, K_1 + 2, \dots, K$$

が成り立つ。同様に、バッファ k の周辺定常確率を $P_k(X_k = n_k)$ で表すと、

$$(3) P_k(X_k = n_k) = P(X_k = n_k), \\ n_k = 0, 1, \dots, C_k, k=1, 2, \dots, K-1$$

が成り立つ。

3. 2 近似サブシステム

組み立てノードであるノード $k (K_1 < k < K)$ に対して、サブシステムを図4のような近似サブシステムをもって近似する。

まず、下流のノードについては、ノード $d(k)$ への他の入力ノードに関連する部分を切り捨て、サイズ C_k のバッファで近似する。ノード k でのサービスを終了した客がいったんこのバッファに入る。バッファ内の客数 n_k が正のとき、客が平均 $1/\delta_k$ のポアソン過程に従ってバッファから退去するとする。ここで、

$$\delta_k(n_k) = \begin{cases} 0, & n_k = 0, \\ \delta_k, & n_k = 1, 2, \dots, C_k \end{cases}$$

とおく。

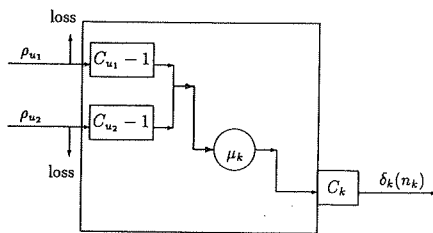


図4：近似サブシステム k

外部から上流のバッファ $u(k)$ への到着流は、到着率 ρ_u のポアソン過程で近似する。バッファ u がいっぱいになれば、到着した客がロスとなる。

この近似サブシステムの状態は (n_{u_1}, n_{u_2}, n_k) をもって表し、定常状態における (n_{u_1}, n_{u_2}, n_k) の確率は、 $P_k(X'_{u_1} = n_{u_1}, X'_{u_2} = n_{u_2}, X'_k = n_k)$ で、周辺確率は $P_k(X'_u = n_u)$ および $P_k(X'_k = n_k)$ で表す。

近似サブシステム k のスループットを T'_k とおくと、 T'_k は単位時間あたりの平均到着客数なので、

$$(4) T'_k = \rho_u [1 - P_k(X'_u = C_u)]$$

である。同時に T'_k は単位時間あたりの平均退去客数でもあるので、

$$(5) T'_k = \delta_k [1 - P_k(X'_k = 0)]$$

で求めることもできる。

他の近似サブシステムは上記の近似サブシステムとは基本構造が同じであるが、以下の点において異なる。

グループ2の組立ノードでないノード k の入力はいくつのノードからであるので、窓口の前にバッファが1つしかない。したがって、近似サブシステム k の状態は (n_u, n_k) で表す。

グループ1の場合、ノード k の窓口の前に常に無限の客がいるため、近似サブシステム k の状態は n_k のみで表すことができる。

同様に、ノード K のサービスを終了した客が直ちにシステムを退去するので、近似サブシステム K において、窓口の下流にバッファをつける必要はない。近似サブシステムの状態は (n_{u_1}, n_{u_2}) または n_u で表す。

3. 3 到着率と退去率の近似

まず、スループットについて次の仮定をおく。

仮定1 定常状態において、各近似サブシステムのスループットがもとのサブシステムのスループットに等しい。すなわち、

$$T'_k = T_k, k=1, 2, \dots, K-1$$

この仮定と(1)より、

$$T'_{u(k)} = T_{u(k)} = T = T_k = T'_k$$

となり、(4)式より、

$$(6) T'_{u(k)} = \rho_u [1 - P_k(X'_u = C_u)]$$

よって、

$$(7) \rho_u = T'_{u(k)} / [1 - P_k(X'_u = C_u)]$$

で ρ_u を推定することができる。

同様に、 $T'_k = T'_{d(k)}$ および(5)式より、

$$(8) \delta_k = T'_{d(k)} / [1 - P_k(X'_k = 0)]$$

他方、周辺定常確率については、次の仮定をおく。

仮定2 定常状態において

$$P_k(X'_k = n_k) = P_k(X_k = n_k)$$

かつ

$$P_k(X'_u = n_u) = P_k(X_u = n_u)$$

仮定および(2), (3)より、

$$P_{u(k)}(X'_{u(k)} = C_{u(k)}) \\ = P_{u(k)}(X_{u(k)} = C_{u(k)})$$

$$= P(X_{u(k)} = C_{u(k)}) = P_k(X_u = C_u)$$

$$(9) \quad \rho_u = T'_u / [1 - P_{u(k)}(X'_u = C_u)]$$

よって, (4)より,

$$(10) \quad \rho_u = T'_u / [1 - P_{u(k)}(X'_u = C_u)]$$

$T'_{u(k)} = T'_u$ だから

$$\rho_u = T'_{u(k)} / [1 - P_{u(k)}(X'_{u(k)} = C_{u(k)})]$$

が成り立つ。添え字を入れ替えれば,

$$(11) \quad \rho_k = T'_k / [1 - P_k(X'_k = C_k)]$$

と書くことができる。

(9)式と同様に

$$(12) \quad P_{d(k)}(X'_k = 0) = P_k(X'_k = 0)$$

が成り立つので, (8)より

$$(13) \quad \delta_k = T'_{d(k)} / [1 - P_{d(k)}(X'_k = 0)]$$

が得られる。

上記の式の中から, ρ_u と δ_k に関するものをそれぞれ一本使って, 近似解を求めることができる。[7]では, 数値実験の結果, 極小型ブロッキングをもつ直列型待ち行列システムに関しては(8)と(11)(組み合わせ1)を組み合わせることによって得られる近似解の精度がもっとも高いことが認められた。しかし, 本論文で扱われるモデルについては, パラメータによっては(7)と(13)(組み合わせ2)の方がよりいい近似解が得られる可能性もあることが分かった。

4 数値結果

前節で提案された近似解法について, 我々は2つのモデルで数値計算を行ない, 近似解法の有効性を検証した。

モデル1 (図5) は10個のノードからなるシステムである。そのうち, ノード7, ノード8, および最終ノードのノード10が組み立て型ノードである。

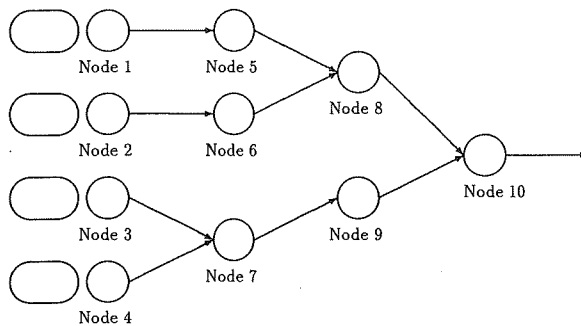


図5：モデル1

表1：モデル1のスループット

($\mu_1 = \dots = \mu_{10} = 4$)

かんばん	シミュレーション	組み合わせ1	組み合わせ2
$C_1 = \dots = C_9 = 1$	1.135±.01	1.084	1.084
$C_1 = \dots = C_9 = 2$	1.828±.01	1.780	1.770
$C_1 = \dots = C_9 = 3$	2.266±.01	2.226	2.218
$C_1 = \dots = C_9 = 4$	2.555±.01	2.538	2.524
$C_1 = \dots = C_9 = 5$	2.764±.01	2.744	2.743
$C_1 = \dots = C_9 = 6$	2.919±.01	2.905	2.908
$C_1 = \dots = C_9 = 7$	3.038±.01	3.030	3.033

表2：モデル2のスループット

($\mu_1 = \dots = \mu_{11} = 4$)

かんばん	シミュレーション	組み合わせ1	組み合わせ2
$C_1 = \dots = C_{10} = 1$	1.106±.01	1.050	1.050
$C_1 = \dots = C_{10} = 2$	1.797±.01	1.739	1.728
$C_1 = \dots = C_{10} = 3$	2.229±.01	2.187	2.176
$C_1 = \dots = C_{10} = 4$	2.523±.01	2.493	2.485
$C_1 = \dots = C_{10} = 5$	2.737±.01	2.712	2.707

モデル2はモデル1のノード10の下流にもう1つノードを付け加えたものである。したがって, 最終ノードは組み立て型ノードではない。

表1はモデル1の数値実験の結果を示している。表から分かるように, すべての近似値がシミュレーションで得られたスループットの値より小さい。しかし, 各ノードのかんばん数が1のときの近似誤差が約-4.5%であるに対して, かんばん数が増加するにつれ, 近似値がシミュレーション値に近づいていく。各ノードのかんばん数が7の時は誤差は-0.3%前後しかなく, 非常に精度の高い近似値が得られたといえよう。

かんばん数が少ない時, 組み合わせ1による近似は組み合わせ2による近似より精度が少し高いが, かん

ばん数が多くなると、逆に組み合わせ2による近似の方がわずかに精度が高くなる。

表2はモデル2の数値実験の結果を示している。モデル1と同様、近似値がシミュレーション値より小さいが、かんばん数が増えるにつれて精度が高くなる。組み合わせ2による近似の方が精度が少し劣るが、かんばん数が増えるにつれ組み合わせ1による近似に近づくことがわかる。かんばん数が増えれば、モデル1同様に逆転するであろう。

5 む す び

本論文は、組み立てノードを複数もつ待ち行列ネットワークに対して近似解析法を提案した。この近似法は[7]の近似法を拡張したものであるが、ブロッキングルールおよびノード構造の違いにより、[7]と違って、パラメータの近似式が2通りの組み合わせが提案された。

本論文で扱われているモデルは単一窓口で、指数サービスであるが、近似法は複数窓口、相型サービス時間分布に関するても有効であると思われる。

複数組み立てノードをもつネットワークモデルについてはほとんど研究されていない。バッファ配置などについての検討が今後の課題である。

謝 辞

本研究に際し、貴重な助言を下された東京工業大学情報理工学研究所の高橋幸雄先生と、参考文献を提供して下さいた電気通信大学電子情報学科の高橋美佐さんに、心より感謝の意を表したい。

参 考 文 献

- [1] Altiok, T., "Approximate Analysis of Exponential Tandem Queues with Blocking", *European Journal of Operational Research* 11 (1982) 390-398.
- [2] Bhat, U.N., "Finite Capacity Assembly-like Queues", *Queueing Systems* 1 (1986) 85-101.
- [3] Brandwajn, A. and Jow, Y.L., "Tandem exponential queues with finite buffers", in: Hasegawa, T., Takagi, H. and Takahashi, Y. (Eds.), *Computers Networking and Performance Evaluation*, ELSEVIER, (North Holland) 1986, 245-257.
- [4] Harrison, J.M., "Assembly-like Queues", *Journal of Applied Probability* 18 (1981) 354.
- [5] Hopp, W.J. and Simon, J.T., "Bounds and Heuristics for Assembly-like Queues" *Queueing Systems* 4 (1989) 137-156.
- [6] Lipper, E.H. and Sengupta, B., "Assembly-like Queues with Finite Capacity: Bounds, Asymptotics, and Approximations", *Queueing Systems* 1 (1986) 67-83.
- [7] Mitra, D. and Mitrani, I., "Analysis of a Novel Discipline for Cell Coordination in Production Lines, I", *Management Science* 36 (1990) 1548-1566.
- [8] Mitra, D. and Mitrani, I., "Analysis of a Novel Discipline for Cell Coordination in Production Lines, II", *Operations Research* 39 (1991) 807-823.
- [9] Perros, H.G. and Altiok, T., "Approximate analysis of open networks of queues with blocking: Tandem configurations", *IEEE Transactions on Software Engineering* SE-12 (1986) 450-461.
- [10] Schweitzer, P.J. and Altiok, T., "Aggregate modeling of tandem queues with blocking", in: Iazeolla, G., Courtois, P.J. and Boxma, O.J. (Eds.), *Computer Performance and Reliability*, Elsevier (North Holland), 1988, 135-149.
- [11] Schweitzer, P.J. and Altiok, T., "Aggregate modeling of tandem queues without intermediate buffers", in: Perros, H.G. and Altiok T. (Eds.), *North-Holland Queueing Networks with Blocking*, 1989, 47-72.
- [12] Song, Y. and Takahashi, Y., "Aggregate Approximation for Tandem Queueing Systems with Production Blocking", *Journal of Operations Research Society of Japan* 34 (1991) 329-354.
- [13] Song, Y. and Takahashi, Y., "Approximation for Kanban Production Systems", to appear in *International Journal of Systems Science*.
- [14] Takahashi, Y., "A new type aggregation

- method for large Markov chains and its application to queueing networks”, in: Akiyama, M. (Ed.), *Proceedings of ITC 11* (Kyoto, Japan, Sept, 1985) Elsevier (North Holland), IAC 1985, pp. 1, 4 A.1.1-4.
- [15] Takahashi, Y., “Aggregate approximation for acyclic queueing networks with communication blocking”, in: Perros, H.G. and Altiook T. (Eds.), *Queueing Networks with Blocking*, North-Holland, 1989, 33-46.
- [16] Takahashi, Yutaka, Miyahara, H. and Hasegawa, T., “An approximation method for open restricted queueing networks”, *Operations Research* 28 (1980) 597-602.