

分散コンピュータシステムにおける複数クラスジョブの負荷分散

金 宗根 亀田 壽夫
電気通信大学情報工学科

分散コンピュータシステムにおける静的負荷分散問題を考える。Tantawi と Towsley は単一クラスの場合の全体最適化方式の最適解を求めた。本研究では Tantawi らの枠組を複数クラスの場合に拡張し全体最適化方式と個別最適化方式の最適解を簡単な形で求め、各々の方式での負荷分散のためのアルゴリズムを提案した。またシステムモデルで数値実験を行い両方式で決められる負荷のふるまいを比較検討した。それによると、単一クラスの個別最適化方式で起こる異常現象が複数クラスでも存在するのを確認した。また全体最適化方式の場合に起こるクラス間の不平等などの複数クラスで起こるいくつかの異常現象を発見した。

Load balancing of multi-class jobs in a distributed computer system

Chonggun KIM Hisao KAMEDA
The University of Electro-Communications
1-5-1 Chofugaoka, Chofu-shi, Tokyo 182, Japan.

We consider static load balancing problems for a distributed computer system. Tantawi and Towsley obtained an solution of the overall optimal policy for the case of single-class jobs.

We first obtain simple forms of solutions of the overall optimal policy and the individually optimal policy for the case of multi-class jobs. Furthermore, we propose new efficient algorithms that determine the optimal load under the two optimal load balancing policies. The two policies are also examined numerically in a system model. We compare results of the two optimal load balancing policies, and we observe that the two optimal policies for the case of multi-class jobs may show some anomalous behavior.

1 はじめに

ワークステーション等のコンピュータ(以下ノードと呼ぶ)を bus 型通信回線を通じて接続させた、分散型コンピュータシステムが考えられる。このシステムにより、各ユーザからは、あたかも、身近にあるコンピュータの性能が向上し、処理能力が増大するようにみられる事が期待される。ここで各コンピュータに到着するジョブに対する応答時間の期待値を最小にする負荷分散法が重要となる。

応答時間の期待値を最小にする方法には二つが考えられる。一つはシステム全体にわたる平均応答時間を最小にする問題であり、これを **全体最適化方式**と呼ぶ。この方式によると各ノードに到着したジョブのうち、自ノードで処理されるジョブに対する応答時間の期待値と、他ノードで処理されるジョブに対する応答時間の期待値が必ずしも等しくなく、各ジョブ間に不公平感がある。もう一つは Wardrop[1]などが交通トラフィックの均衡問題 [2] に対して考えたように、各ジョブに対して各々の応答時間の期待値が最小になるようにする方式であり、これを **個別最適化方式**と呼ぶ。この方式によるとシステム全体の平均応答時間は必ずしも最小ではないが、あるノードに到着したジョブはどのノードで処理されても応答時間の期待値が同じになる点で公平である。

Tantawi と Towsley[3] は、システムの各ノードへのジョブの到着率、各ノードの処理能力などの諸パラメータが与えられ、また通信による遅延が発信元、着信先の違いによらないという仮定(この仮定は LAN や衛星通信等の場合に成り立つ)の下で単一ジョブクラスの場合の全体最適化方式の解を示し、最適負荷を決定するためのアルゴリズムを提案した。

本研究の主な目的は Tantawi ら[3]のモデルの枠組を複数ジョブクラスの場合に拡張し、全体最適化方式及び個別最適化方式での両方式の解の満たすべき条件を求め、単純化し、最適解を求めるアルゴリズムを提案することである。また具体的なモデルを考えて、提案するアルゴリズムに基づいて数値実験を行うことにより、二つの最適化方式における応答時間の期待値、最適負荷などのふるまいを比較検討する。

2 モデルの記述

分散コンピュータシステムのモデルを考える(図1)。このモデルでは n 個のノードが Bus 形通信ネットワークに任意の形態で接続されており各ノードでは複数クラスのジョブが処理できる。モデルで使用される記号等を次に示す。

- n : ノード数
- m : クラス数
- $\phi_i^{(k)}$: ノード i におけるクラス k ジョブの到着率
- $\beta_i^{(k)}$: ノード i におけるクラス k ジョブの負荷
- $x_{ij}^{(k)}$: ノード i からノード j へ転送されるクラス k ジョブの割合
- $\lambda^{(k)}$: 通信線におけるクラス k ジョブの通信量
- $\phi^{(k)}$: クラス k ジョブのシステム全体への到着率
- Φ : システム全体の到着率
- ϕ : $[\phi_1, \phi_2, \dots, \phi_n]$
- $\phi^{(k)}$: $[\phi_1^{(k)}, \phi_2^{(k)}, \dots, \phi_n^{(k)}]$
- ϕ_i : $[\phi_i^{(1)}, \phi_i^{(2)}, \dots, \phi_i^{(m)}]$

- β : $[\beta_1, \beta_2, \dots, \beta_n]$
- $\beta^{(k)}$: $[\beta_1^{(k)}, \beta_2^{(k)}, \dots, \beta_n^{(k)}]$
- β_i : $[\beta_i^{(1)}, \beta_i^{(2)}, \dots, \beta_i^{(m)}]$
- λ : $[\lambda^{(1)}, \lambda^{(2)}, \dots, \lambda^{(m)}]$
- $F_i^{(k)}(\beta_i)$: 到着ノード i で処理を受ける場合の平均遅延(微分可能で凸形の増加関数)
- $F_j^{(k)}(\beta_j) + G^{(k)}(\lambda)$: 他のノード j へ伝送され処理を受けるクラス k ジョブの応答時間の期待値
- $G^{(k)}(\lambda)$: 通信によるクラス k の平均遅延(発信元、着信先の違いによらない微分可能で凸形の非減少関数)

各ノード i へのクラス k ジョブはポアソン過程により到着する。ノード i に到着するジョブはノード i で処理されるか、あるいは、他のノード j に転送されて処理されるかになる。クラス k ジョブが一旦他のノードに転送された場合、さらに別のノードに転送されることはないとする。

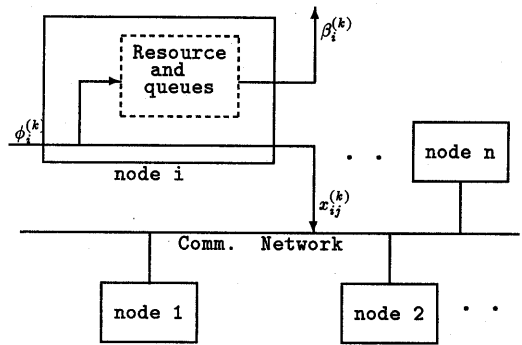


図1. Bus 型分散コンピュータシステム

システム全体の平均応答時間 $(D(\beta))$ は平均のノード遅延と平均の通信遅延の和の形で表される。

$$D(\beta) = \frac{1}{\Phi} \sum_{k=1}^m \left(\sum_{i=1}^n \beta_i^{(k)} F_i^{(k)}(\beta_i) + \lambda^{(k)} G^{(k)}(\lambda) \right) \quad (1)$$

ただし

$$\sum_{i=1}^n \beta_i^{(k)} = \phi^{(k)}, \quad k = 1, 2, \dots, m,$$

$$\beta_i^{(k)} \geq 0, \quad i = 1, 2, \dots, n, k = 1, 2, \dots, m.$$

ここで各クラスの通信量 $\lambda^{(k)}$ は $\beta_i^{(k)}$ により決められる。

$$\lambda^{(k)} = \frac{1}{2} \sum_{i=1}^n |\phi_i^{(k)} - \beta_i^{(k)}|$$

各ノードを、ジョブを処理するやり方により、各クラスごとに次のように呼ぶ。

- (1) idle source($R_d^{(k)}$): 到着したジョブの全部を他ノードに送る。
- (2) active source($R_a^{(k)}$): 到着したジョブの一部を他ノードに送る。
- (3) neutral($N^{(k)}$): 到着したジョブを全部自分で処理する。
- (4) sink($S^{(k)}$): 到着したジョブと他ノードから送ってきたジョブを処理する。

通信ネットワークを通じてノード*i*からノード*j*へ流れるクラス*k*ジョブの割合 $x_{ij}^{(k)}$ は、例えばジョブの需要・供給の制約条件で次のように決められる。

$$x_{ij}^{(k)} = \begin{cases} \frac{(\beta_j^{(k)} - \phi_j^{(k)})}{\lambda^{(k)}} (\phi_i^{(k)} - \beta_i^{(k)}) & i \in R_a^{(k)}, R_s^{(k)} \quad j \in S^{(k)} \\ 0 & \text{otherwise} \end{cases}$$

ノード*j*が sink の場合は $(\beta_j^{(k)} - \phi_j^{(k)})$ がノード*j*へ送られるジョブの割合を表す。一方、ノード*i*が source の場合は $(\phi_i^{(k)} - \beta_i^{(k)})$ がノード*i*から出てくるジョブの割合を表す。

3 アルゴリズム

3.1 全体最適化方式の解

全体最適化方式に置ける最適解を表現するためにノード*i*のクラス*k*の限界ノード遅延 $f_i^{(k)}(\beta_i)$ とクラス*k*の限界通信遅延 $g^{(k)}(\lambda)$ の定義を次のように与える。

$$f_i^{(k)}(\beta_i) = \frac{\partial}{\partial \beta_i^{(k)}} \sum_{l=1}^m \beta_i^{(l)} F_i^{(l)}(\beta_i), \quad (2)$$

$$g^{(k)}(\lambda) = \frac{\partial}{\partial \lambda^{(k)}} \sum_{l=1}^m \lambda^{(l)} G^{(l)}(\lambda). \quad (3)$$

ここで逆関数 $f_i^{(k)-1}$ を次のように定義する。

$$f_i^{(k)-1}(\beta_i | \beta_i^{(k)} = x) = a, \quad f_i^{(k)}(\beta_i | \beta_i^{(k)} = a) = x, \\ = 0, \quad f_i^{(k)}(\beta_i | \beta_i^{(k)} = 0) \geq x.$$

ここで $\beta_i | \beta_i^{(k)} = x$ の意味は、ベクトル β_i の要素 $\beta_i^{(k)}$ のみが x に入れ替わって他の要素はもとの β_i の値と同じベクトルを表す。

[定理1] システム全体の平均応答時間を最小にする全体最適化の解は $k = 1, 2, \dots, m$ に対し次の関係を満たす。

$$f_i^{(k)}(\beta_i | \beta_i^{(k)} = 0) \geq \alpha^{(k)} + g^{(k)}(\lambda), \quad \beta_i^{(k)} = 0 \quad (i \in R_a^{(k)}), \quad (4a) \\ f_i^{(k)}(\beta_i | \beta_i^{(k)} = \phi_i^{(k)}) > \alpha^{(k)} + g^{(k)}(\lambda) > f_i^{(k)}(\beta_i | \beta_i^{(k)} = 0), \quad 0 < \beta_i^{(k)} < \phi_i^{(k)} \quad (i \in R_s^{(k)}), \quad (4b) \\ \alpha^{(k)} \leq f_i^{(k)}(\beta_i | \beta_i^{(k)} = \phi_i^{(k)}) \leq \alpha^{(k)} + g^{(k)}(\lambda), \quad \beta_i^{(k)} = \phi_i^{(k)} \quad (i \in N^{(k)}), \quad (4c) \\ f_i^{(k)}(\beta_i | \beta_i^{(k)} = \phi_i^{(k)}) < \alpha^{(k)}, \quad \beta_i^{(k)} > \phi_i^{(k)} \quad (i \in S^{(k)}). \quad (4d)$$

ただし、 $\alpha^{(k)} = \min f_i^{(k)}(\beta_i)$ とする。さらに次の制約条件が満たさなければならない。

$$\lambda_{sen}^{(k)} = \lambda_{rec}^{(k)}, \quad (5) \\ \lambda_{sen}^{(k)} = \sum_{i \in R_s^{(k)}, R_a^{(k)}} \min\{\phi_i^{(k)}, \phi_i^{(k)} - f_i^{(k)-1}(\beta_i | \beta_i^{(k)} = \alpha^{(k)} + g^{(k)}(\lambda))\}, \\ \lambda_{rec}^{(k)} = \sum_{i \in S^{(k)}} (f_i^{(k)-1}(\beta_i | \beta_i^{(k)} = \alpha^{(k)}) - \phi_i^{(k)}).$$

[証明] は付録 A 参照。

3.2 全体最適化のアルゴリズム

全体最適化方式で最適負荷を求めるアルゴリズムを次に示す。このアルゴリズムを各クラスに対して行い、クラス*k*の通信量 $\lambda^{(k)}$ を持つ通信量ベクトル $\lambda^{(l)}$ を決めていく。ここで l は全てのクラス*i* 対

して下記のアルゴリズムを実行する回数である。この過程を $|\lambda_{l-1} - \lambda_{(l)}| < \epsilon$ になるまで全てのクラス*i* に対して繰り返す。ここで回目の計算においてクラス*k*の負荷 $\beta_{(l)}^{(k)}$ を計算するためにはクラス $k-1$ までに計算された $\beta_{(l)}^{(1)}, \dots, \beta_{(l)}^{(k-1)}$ も使い、次のようにクラス*k*の負荷を決めて行く。

$$(\beta_{(l)}^{(1)}, \beta_{(l)}^{(2)}, \dots, \beta_{(l)}^{(k)}, \beta_{(l)}^{(k)}, \beta_{(l-1)}^{(k+1)}, \dots, \beta_{(l-1)}^{(n)})$$

l 回目のクラス*k*の負荷 $\beta_{(l)}^{(k)}$ を決める詳細なアルゴリズムは次のようになる。

1. 整理する。

$$f_1^{(k)}(\beta_1 | \beta_1^{(k)} = \phi_1^{(k)}) \leq f_2^{(k)}(\beta_2 | \beta_2^{(k)} = \phi_2^{(k)}) \leq \dots \leq f_n^{(k)}(\beta_n | \beta_n^{(k)} = \phi_n^{(k)})$$

もし $f_1^{(k)}(\beta_1 | \beta_1^{(k)} = \phi_1^{(k)}) + g^{(k)}(\lambda | \lambda^{(k)} = 0) \geq f_n^{(k)}(\beta_n | \beta_n^{(k)} = \phi_n^{(k)})$ ならば

クラス*k*の負荷分散は必要ない。

2. $\alpha^{(k)}$ 、 $\alpha^{(k)} + g^{(k)}(\lambda)$ を決める。

$$\lambda_{rec}^{(k)} = \sum_{i \in S^{(k)}} (f_i^{(k)-1}(\beta_i | \beta_i^{(k)} = \alpha^{(k)}) - \phi_i^{(k)})$$

$$\lambda_{sen}^{(k)} = \sum_{i \in R_a^{(k)}, R_s^{(k)}} (\phi_i^{(k)} - \max\{0, f_i^{(k)-1}(\beta_i | \beta_i^{(k)} = \alpha^{(k)} + g^{(k)}(\lambda_{rec}))\})$$

二分検索により $\lambda_{rec}^{(k)} = \lambda_{sen}^{(k)}$ になる $\alpha^{(k)}$ を決める。

ここで $(f_1^{(k)}(\beta_1 | \beta_1^{(k)} = \phi_1^{(k)}) \leq \alpha^{(k)} \leq f_n^{(k)}(\beta_n | \beta_n^{(k)} = \phi_n^{(k)}))$ である。

$\alpha^{(k)}$ 、 $\alpha^{(k)} + g^{(k)}(\lambda)$ が決まる。

3. 最適負荷を決定する。

(4a) の場合、 $\beta_i^{(k)} = 0$,

(4b) の場合、 $\beta_i^{(k)} = f_i^{(k)-1}(\beta_i | \beta_i^{(k)} = \alpha^{(k)} + g^{(k)}(\lambda))$,

(4c) の場合、 $\beta_i^{(k)} = \phi_i^{(k)}$,

(4d) の場合、 $\beta_i^{(k)} = f_i^{(k)-1}(\beta_i | \beta_i^{(k)} = \alpha^{(k)})$.

3.3 個別最適化方式の解

個別最適化方式は、交通トラフィックの均衡問題において、各個人が目的地までの最短距離を選んで運行する使用者均衡法として、Wardrop[1] 等によって以前から研究されている。本研究では、このような個別最適化方式を、分散コンピュータシステムにおける負荷分散について考える。即ち、本研究の個別最適化方式は、システムに到着する各ジョブの各々の応答時間の期待値が最小となるように各ジョブの処理ノードが決められる負荷分散方式である。

[定理2] 個別最適化方式に対して均衡解が存在する。その均衡状態においてすべてのノードは $k = 1, 2, \dots, m$ に対し次の関係を満たす。

$$F_i^{(k)}(\beta_i | \beta_i^{(k)} = 0) \geq \Theta^{(k)} + G^{(k)}(\lambda), \quad \beta_i^{(k)} = 0 \quad (i \in R_a^{(k)}), \quad (6a) \\ F_i^{(k)}(\beta_i | \beta_i^{(k)} = \phi_i^{(k)}) > \Theta^{(k)} + G^{(k)}(\lambda) > F_i^{(k)}(\beta_i | \beta_i^{(k)} = 0), \quad 0 < \beta_i^{(k)} < \phi_i^{(k)} \quad (i \in R_s^{(k)}), \quad (6b) \\ \Theta^{(k)} \leq F_i^{(k)}(\beta_i | \beta_i^{(k)} = \phi_i^{(k)}) \leq \Theta^{(k)} + G^{(k)}(\lambda), \quad \beta_i^{(k)} = \phi_i^{(k)} \quad (i \in N^{(k)}), \quad (6c) \\ F_i^{(k)}(\beta_i | \beta_i^{(k)} = \phi_i^{(k)}) < \Theta^{(k)}, \quad \beta_i^{(k)} > \phi_i^{(k)} \quad (i \in S^{(k)}). \quad (6d)$$

ただし、 $\Theta^{(k)} = \min_i F_i^{(k)}(\beta_i)$ とする。さらに次の制約条件が満たさなければならない。

$$\lambda_{sen}^{(k)} = \lambda_{rec}^{(k)}, \quad (7)$$

$$\lambda_{sen}^{(k)} = \sum_{i \in R_s^{(k)}, R_r^{(k)}} \min\{\phi_i^{(k)}, \phi_i^{(k)} - F_i^{(k)-1}(\beta_i |_{\beta_i^{(k)} = \Theta^{(k)} + G^{(k)}(\lambda)})\},$$

$$\lambda_{rec}^{(k)} = \sum_{i \in S^{(k)}} (F_i^{(k)-1}(\beta_i |_{\beta_i^{(k)} = \Theta^{(k)}}) - \phi_i^{(k)}).$$

[証明] は付録 B 参照。

3.4 個別最適化のアルゴリズム

[定理 1] と [定理 2] から分かるように、両方式の解は非常に似ていて使用している関数の形の違い以外はすべて同じである。従って二つのアルゴリズムも全く同じ形になる。全体最適化方式のアルゴリズムから $f_i^{(k)}(\beta_i)$, $g^{(k)}(\lambda)$, $\alpha^{(k)}$ を各々 $F_i^{(k)}(\beta_i)$, $G^{(k)}(\lambda)$, $\Theta^{(k)}$ に置き換えると個別最適化方式のアルゴリズムが得られる。

4 数値実験

4.1 実験モデル

本研究で提案したアルゴリズムにより全体最適化方式による負荷分散と個別最適化方式による負荷分散のふりまいがどのようになるかを調べるためにシステムモデルの具体例を与えて調べる。ノードモデルとしては図 2 のような複数クラスのセントラルサーバモデルを考える。このモデルの CPU を含む各デバイス ($j = 0, 1, \dots, d$) をサーバと呼ぶ。

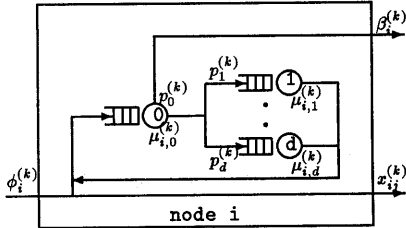


図 2. 複数クラスのセントラルサーバモデル

セントラルサーバモデルのクラス k のノード遅延の期待値は以下の (8) 式のように表される。ここでサーバ 0 は CPU であり、プロセスシャレリング方式で処理する。サーバ 1 から d までは I/O デバイスであり、FIFO 順で処理する。図 2 で $p_0^{(k)}$ と $p_j^{(k)}$ ($j = 1, \dots, d$) は CPU から出て行くクラス別ジョブの確率である。また $q_0^{(k)} = \frac{1}{\mu_{i,0}^{(k)}}$ であり、 $q_j^{(k)} = \frac{p_j^{(k)}}{\mu_{i,j}^{(k)}}$ ($j = 1, \dots, d$) である。通信ネットワークモデルとしてはプロセスシャレリング処理方式の M/G/1 モデルを考える。通信遅延の期待値を表す式は以下の (9) 式のようにになる。ここで $t^{(k)}$ ($k = 1, 2, \dots, m$) は待ちを含まない各クラス別通信所要時間を表している。

$$F_i^{(k)}(\beta_i) = \sum_{j=0}^d \frac{q_j^{(k)} \frac{1}{\mu_{i,j}^{(k)}}}{1 - (q_j^{(1)} \frac{\beta_j^{(1)}}{\mu_{i,j}^{(1)}} + \dots + q_j^{(m)} \frac{\beta_j^{(m)}}{\mu_{i,j}^{(m)}})} \quad (8)$$

$$G^{(k)}(\lambda) = \frac{t^{(k)}}{1 - (t^{(1)}\lambda^{(1)} + \dots + t^{(m)}\lambda^{(m)})} \quad (9)$$

処理能力の面から見て 3 種類に分けられる七つのノードが bus 形通信ネットワークに接続されている分散コンピュータシステムのモデルについて数値実験を行う。本実験では二つのクラスの場合を対象にする。また、クラス別の通信所要時間 $t^{(k)}$ ($k = 1, 2$) は同じにする。表 1. には各ノードの規模 (処理能力) 及び各ノードの各クラスのジョブの到着具合 (率) 等を表している。表 2. では各ノードにおける各サーバの処理率及び CPU から出ていくジョブの確率を記している。

表 1: システムモデルのパラメタ

ノード	規模 (処理能力)		ジョブの到着具合 (率)	
	クラス 1	クラス 2	I/O bound クラス 1	CPU bound クラス 2
1	大	特大	小 (11.0)	大 (9.0)
2	中	中	大 (10.0)	小 (0.5)
3			中 (5.0)	中 (1.0)
4			小 (1.0)	大 (2.0)
5	小	小	大 (5.0)	小 (0.02)
6			中 (2.0)	中 (0.05)
7			小 (0.5)	大 (0.10)

表 2: ノードモデルのパラメタ

ノード	各サーバの処理率				CPU から出るジョブの確率			
	$\mu_{i,0}^{(1)}$	$\mu_{i,1}^{(1)}$	$\mu_{i,2}^{(1)}$	$\mu_{i,3}^{(1)}$	$p_0^{(1)}$	$p_1^{(1)}$	$p_2^{(1)}$	$p_3^{(1)}$
1	1000	200	200	200	0.1	0.3	0.3	0.3
	100	200	200	200	0.2	0.266	0.266	0.266
2	300	100	100	.	0.1	0.45	0.45	.
	15	100	100	.	0.2	0.4	0.4	.
3-4	300	100	.	.	0.1	0.9	.	.
	15	100	.	.	0.2	0.8	.	.
5-7	150	100	.	.	0.1	0.9	.	.
	3	100	.	.	0.2	0.8	.	.

4.2 実験結果

図 3. は全体最適化方式の下で、通信ネットワークの通信所要時間を 0 から次第に大きくした場合の、平均応答時間の変化と最適負荷のふりまいを表している。この図は八つの部分に分かれており、各部分が左側と右側の二つのグラフに分かれている。この図の左側の各グラフは、各ノードのクラス別平均応答時間を表す七つのグラフと、その下のシステム全体の平均応答時間 (M.R.T) と平均通信遅延 (C.D.T) を表すグラフで、計八つのグラフになっている。右側のグラフは、各ノードのクラス別最適負荷を表す七つのグラフと、その下の通信ネットワークのクラス別のトラフィック (通信線を通じて送られるジョブの割合) を表すグラフで、計八つのグラフになっている。各グラフにはクラス 1 とクラス 2 の値を一括に示している。左側の 8 番目のグラフ内には平均通信遅延 (C.D.T) と比較するため通信時間を表している。

信所要時間を細線で示してある。点線がある場合は、その尺度が実線のものとは異なることを示している。図4. は個別最適化方式の下で通信所要時間を変化させた場合である。その配置などは図3. と同じである。

図3. と図4. から、通信所要時間が小さい場合は、処理能力に比べて到着率が小さいノードに、処理能力に比べて到着率が大きいノードから、かなりのジョブが送られているのが分かる。さらに通信所要時間が大きくなるに従って、各ノードはジョブを送らなくなり、結局は全てのジョブを自ノードで処理するようになっていくのが見られる。通信所要時間が大きくなるにつれて自ノードでジョブを処理するようになる傾向は、通信ネットワークのトラフィックが減っていくことにも現れている。

しかし、この実験結果からいくつかの、直観と反するように見える異常現象も見られる。全体最適化方式の場合の図3. で、ノード5のクラス2の平均応答時間と負荷のふるまいを見ると次のような現象が観測できる。即ち、通信所要時間が小さく、到着したジョブを全て他ノードに転送している間よりも、通信所要時間が大きくなり、自ノードで処理しはじめてからの方が、かえってクラス2の平均応答時間が減少している。また図3. のノード2とノード3のクラス2の負荷のふるまいを見ると、通信所要時間が大きくなっていくにもかかわらず、自ノードで処理する負荷が一時的に減少する現象が観測できる。この場合には同時にクラス1の負荷が急激に変化しているのが注目される。

個別最適化方式の場合の図4. では、単一クラスの場合の数値実験結果[5]と同じように、通信所要時間が大きくなるにもかかわらずシステム全体の平均応答時間(M.R.T)が一時的に下がる異常現象が観測される。

5 まとめ

本研究では Tantawi ら [3] のモデルの枠組を複数クラスジョブに拡張し、全体最適化方式と個別最適化方式の解の満たすべき条件を求め、最適解を求めるアルゴリズムを提案した。この条件は Tantawi ら [3] の単一クラスについて示した解の形より簡単化されている。また、新しく求められたこのアルゴリズムも Tantawi ら [3] の単一クラスのためのアルゴリズムより非常に簡単化されている。このアルゴリズムを利用して、全体最適化方式と個別最適化方式で決定される負荷分散のふるまいを比較検討するため、数値実験をシステムモデルで行った。数値実験の結果、単一クラスの両方式に対する数値実験の結果[5]と同じいくつかの性質と異常現象を観測した。それから、複数クラスの場合のみに起こるクラス間の不平等などのいくつかの異常現象が見出された。

付録 A. [定理 1] の証明

Tantawi と Towsley [3] の単一クラスの解から複数クラスの解に拡張することができる。これは、単一クラスの場合と同様に Kuhn-Tucker 定理を用いて得られる。この解は $k = 1, 2, \dots, m$ に関して次の関係を満たす。

$$\begin{aligned} f_i^{(k)}(\beta_i) &\geq \alpha^{(k)} + g^{(k)}(\lambda), & \beta_i^{(k)} &= 0, & (i \in R_d^{(k)}), & (A1a) \\ f_i^{(k)}(\beta_i) &= \alpha^{(k)} + g^{(k)}(\lambda), & 0 < \beta_i^{(k)} < \phi_i^{(k)}, & (i \in R_a^{(k)}), & (A1b) \\ \alpha^{(k)} \leq f_i^{(k)}(\beta_i) &\leq \alpha^{(k)} + g^{(k)}(\lambda), & \beta_i^{(k)} &= \phi_i^{(k)}, & (i \in N^{(k)}), & (A1c) \\ \alpha^{(k)} &= f_i^{(k)}(\beta_i), & \beta_i^{(k)} &> \phi_i^{(k)}, & (i \in S^{(k)}), & (A1d) \end{aligned}$$

ただし、 $\alpha^{(k)} = \min_{i \in S^{(k)}} f_i^{(k)}(\beta_i)$ である。さらに次の制約条件が満たさなければならない。

$$\begin{aligned} \sum_{i \in S^{(k)}} f_i^{(k)-1}(\beta_i | \beta_i^{(k)} = \alpha^{(k)}) + \sum_{i \in N^{(k)}} \phi_i^{(k)} + \\ \sum_{i \in R_a^{(k)}} f_i^{(k)-1}(\beta_i | \beta_i^{(k)} = \alpha^{(k)} + g^{(k)}(\lambda)) = \phi^{(k)} \quad (A2) \end{aligned}$$

この結果から $k = 1, 2, \dots, m$ に関して (4), (5) の関係が (A1), (A2) の関係の必要条件であることを示す。

$$(A1a) \quad f_i^{(k)}(\beta_i) \geq \alpha^{(k)} + g^{(k)}(\lambda) \text{ の場合, } \beta_i^{(k)} = 0 \text{ である} \\ \text{これから } f_i^{(k)}(\beta_i | \beta_i^{(k)} = 0) \geq \alpha^{(k)} + g^{(k)}(\lambda), \text{ であるので} \\ f_i^{(k)}(\beta_i | \beta_i^{(k)} = 0) \geq \alpha^{(k)} + g^{(k)}(\lambda), \beta_i^{(k)} = 0 \text{ の関係が} \\ \text{成り立つ。}$$

$$(A1b) \quad f_i^{(k)}(\beta_i) = \alpha^{(k)} + g^{(k)}(\lambda) \text{ の場合, } 0 < \beta_i^{(k)} < \phi_i^{(k)} \text{ である} \\ \text{これから } f_i^{(k)}(\beta_i) > f_i^{(k)}(\beta_i | \beta_i^{(k)} = 0) \text{ と} \\ f_i^{(k)}(\beta_i) < f_i^{(k)}(\beta_i | \beta_i^{(k)} = \phi_i^{(k)}) \text{ が成り立つ。} \\ \text{この二つの関係から} \\ f_i^{(k)}(\beta_i | \beta_i^{(k)} = \phi_i^{(k)}) > \alpha^{(k)} + g^{(k)}(\lambda) > f_i^{(k)}(\beta_i | \beta_i^{(k)} = 0), \\ 0 < \beta_i^{(k)} < \phi_i^{(k)} \text{ の関係が成り立つ。}$$

$$(A1c) \quad \alpha^{(k)} \leq f_i^{(k)}(\beta_i) \leq \alpha^{(k)} + g^{(k)}(\lambda) \text{ の場合, } \beta_i^{(k)} = \phi_i^{(k)} \\ \text{である} \\ \text{これから } f_i^{(k)}(\beta_i) = f_i^{(k)}(\beta_i | \beta_i^{(k)} = \phi_i^{(k)}) \text{ になるので} \\ \alpha^{(k)} \leq f_i^{(k)}(\beta_i | \beta_i^{(k)} = \phi_i^{(k)}) \leq \alpha^{(k)} + g^{(k)}(\lambda), \beta_i^{(k)} = \phi_i^{(k)} \\ \text{の関係が成り立つ。}$$

$$(A1d) \quad \alpha^{(k)} = f_i^{(k)}(\beta_i) \text{ の場合, } \beta_i^{(k)} > \phi_i^{(k)} \text{ である} \\ \text{これから } f_i^{(k)}(\beta_i) > f_i^{(k)}(\beta_i | \beta_i^{(k)} = \phi_i^{(k)}) \text{ になるので} \\ f_i^{(k)}(\beta_i | \beta_i^{(k)} = \phi_i^{(k)}) < \alpha^{(k)}, \beta_i^{(k)} > \phi_i^{(k)} \text{ の関係が成り立つ。}$$

ここで β_i に関する次の制約条件から

$$\sum_{i=1}^n \beta_i^{(k)} = \phi^{(k)}, \quad k = 1, 2, \dots, m,$$

$\lambda_{en}^{(k)} = \lambda_{rec}^{(k)}$ の制約条件 (5) が満たされる。

$$\lambda_{en}^{(k)} = \sum_{i \in R_d^{(k)}, R_a^{(k)}} \min\{\phi_i^{(k)}, \phi_i^{(k)} - f_i^{(k)-1}(\beta_i | \beta_i^{(k)} = \alpha^{(k)} + g^{(k)}(\lambda))\},$$

$$\lambda_{rec}^{(k)} = \sum_{i \in S^{(k)}} (f_i^{(k)-1}(\beta_i | \beta_i^{(k)} = \alpha^{(k)}) - \phi_i^{(k)}).$$

このように [定理 1] が証明される。(4a) から (4d) までの $f_i^{(k)}(\beta_i)$ に関する条件はお互いに disjoint であり、かつ全てをつくしている (それ以外の場合は存在しない)。また (A1a) から (A1d) までの $f_i^{(k)}(\beta_i)$ に関する条件もお互いに disjoint であり、全てをつくしている。従って、(4), (5) の関係が (A1), (A2) の関係の充分条件になっていることが容易に分かる。

付録 B. [定理 2] の証明

亀田と植山[5]は Tantawi らと同じ枠組において単一クラスの場合の個別最適化方式の解を求めて、その均衡解が存在するのを証明した。この単一クラス解は次のようにに複数クラス解に拡張することができる。

スケジューリング方式が与えられると $\beta_i^{(k)}$, $i = 1, 2, \dots, n$, $k = 1, 2, \dots, m$ が決定される。

$$\text{ここで } \sum_{i=1}^n \beta_i^{(k)} = \phi^{(k)}, \beta_i^{(k)} \geq 0 \text{ である。}$$

次のように C を定義する。

$$C^{(k)} = \{\beta^{(k)} \mid \sum_{i=1}^n \beta_i^{(k)} = \phi^{(k)}, \beta_i^{(k)} \geq 0, i = 1, 2, \dots, n\}$$

$$C = \prod_{k=1}^m C^{(k)}$$

ここで C は β が取り得る値の集合で、 mn -次元ユークリッド空間の、閉じた、凸形で、有界の部分集合である。 β が決まると

$\lambda^{(k)} = \sum_{i=1}^n \frac{1}{2} |\phi_i^{(k)} - \beta_i^{(k)}|$ はすぐ決定される。それからノード遅延の期待値 $F_i^{(k)}(\beta_i)$ と通信遅延の期待値 $G^{(k)}(\lambda)$ が決定される。ここで

$$\Theta^{(k)} = \min F_i^{(k)}(\beta_i),$$

$$J^{(k)}(\beta) = \{j \mid F_j^{(k)}(\beta_j) = \Theta^{(k)}\}$$

と定義するとノード i の $F_i^{(k)}(\beta_i)$ の値に関して次の四つの場合に分けられる。

- (1). $F_i^{(k)}(\beta_i) > \Theta^{(k)} + G^{(k)}(\lambda)$. この場合ノード i のジョブはノード i 以外のどこかに送られる。
- (2). $F_i^{(k)}(\beta_i) = \Theta^{(k)} + G^{(k)}(\lambda)$. この場合ノード i のジョブはノード i 自分で処理するか、あるいはノード i 以外のノードに送られる。
- (3). $\Theta^{(k)} < F_i^{(k)}(\beta_i) < \Theta^{(k)} + G^{(k)}(\lambda)$. この場合はノード i のジョブはノード i 自分で処理する。
- (4). $\Theta^{(k)} = F_i^{(k)}(\beta_i)$. この場合はノード i のジョブはノード i 自分で処理する (他のノードからもクラス k のジョブが送ってくる。) 従って、スケジューリング後の負荷 β を表す $\beta_i^{(k)}$ が取り得る $\psi_i^{(k)}(\beta)$ 領域は次のようになる。

- (1) の場合は $\beta_i^{(k)} = 0$, $\psi_i^{(k)}(\beta) = \{0\}$
- (2) の場合は $0 \leq \beta_i^{(k)} \leq \phi_i^{(k)}$, $\psi_i^{(k)}(\beta) = \{0, \phi_i^{(k)}\}$
- (3) の場合は $\beta_i^{(k)} = \phi_i^{(k)}$, $\psi_i^{(k)}(\beta) = \{\phi_i^{(k)}\}$
- (4) の場合は $\beta_i^{(k)} \geq \phi_i^{(k)}$, $\psi_i^{(k)}(\beta) = \{\phi_i^{(k)}, \phi^{(k)}\}$

この様にして決まる β の領域を $\psi(\beta)$ と表すと

$$\psi(\beta) = \prod_{k=1}^m \left(\prod_{i=1}^n \psi_i^{(k)}(\beta) \cap C^{(k)} \right)$$

である。 $\psi(\beta)$ は明らかに領域 C から C の対応と見なすことができる。亀田と植山[5]と同様にして対応 ψ が上半連続であることを示すことができる。これから Kakutani[4] の不動点定理を用いて均衡解 β が存在する事を示すことができる。この結果を亀田と植山 [5] のように四つの種類に分類して整理してみると次のようになる。

この解は $k = 1, 2, \dots, m$ に関して次の関係を満たす。

$$\begin{aligned} F_i^{(k)}(\beta_i) &\geq \Theta^{(k)} + G^{(k)}(\lambda), & \beta_i^{(k)} &= 0, & (i \in R_d^{(k)}), \\ F_i^{(k)}(\beta_i) &= \Theta^{(k)} + G^{(k)}(\lambda), & 0 < \beta_i^{(k)} < \phi_i^{(k)}, & (i \in R_a^{(k)}), \\ \Theta^{(k)} &\leq F_i^{(k)}(\beta_i) \leq \Theta^{(k)} + G^{(k)}(\lambda), & \beta_i^{(k)} &= \phi_i^{(k)}, & (i \in N^{(k)}), \\ \Theta^{(k)} &= F_i^{(k)}(\beta_i), & \beta_i^{(k)} &> \phi_i^{(k)}, & (i \in S^{(k)}), \end{aligned}$$

ただし、 $\Theta^{(k)} = \min F_i^{(k)}(\beta_i)$ である。さらに次の制約条件が満たさなければならない。

$$\begin{aligned} \sum_{i \in S^{(k)}} F_i^{(k)-1}(\beta_i |_{\beta_i^{(k)} = \Theta^{(k)}}) + \sum_{i \in N^{(k)}} \phi_i^{(k)} + \\ \sum_{i \in R_d^{(k)}} F_i^{(k)-1}(\beta_i |_{\beta_i^{(k)} = \Theta^{(k)} + G^{(k)}(\lambda)}) = \phi^{(k)} \end{aligned}$$

この解から [定理 1] の証明と全く同じ方法で [定理 2] を導くことが出来る。

参考文献

- [1] Wardrop, J. Some theoretical aspects of road traffic research, *Proceedings of the Institute of Civil Engineers, Part II, Vol. I* (1952) pp.325-378.
- [2] Magnanti, Thomas L. Models and algorithms for predicting urban traffic equilibria, *Transportation Planning Models*, Florian, M. (Editor) Elsevier Science Publishers B.V. (North-Holland), (1984) pp.153-185.
- [3] Tantawi, A.N. and Towsley, D. Optimal static load balancing in distributed computer systems, *J.ACM* 32,2 (April, 1985) pp.445-465.
- [4] Kakutani S. A generalization of Brouwer's fixed point theorem, *Duke Mathematical Journal*, 8, (1941) pp.457-459.
- [5] 亀田 壽夫, 植山 淳雄. コンピュータネットワークに置ける負荷分散について. 情報処理学会オペレーティングシステム研究会研究報告, 34-7(1987).

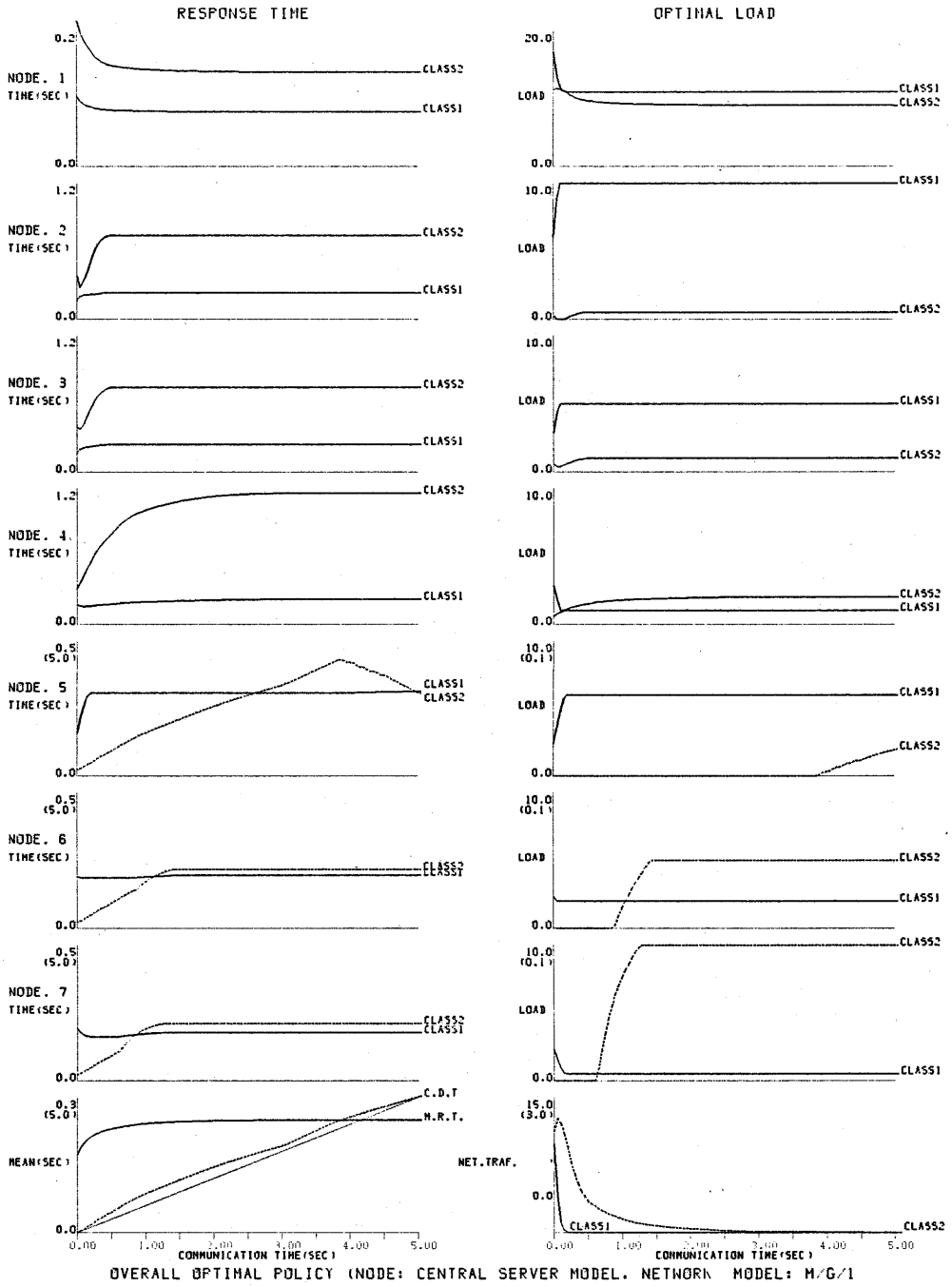


図3. 全体最適化方式での数値実験結果

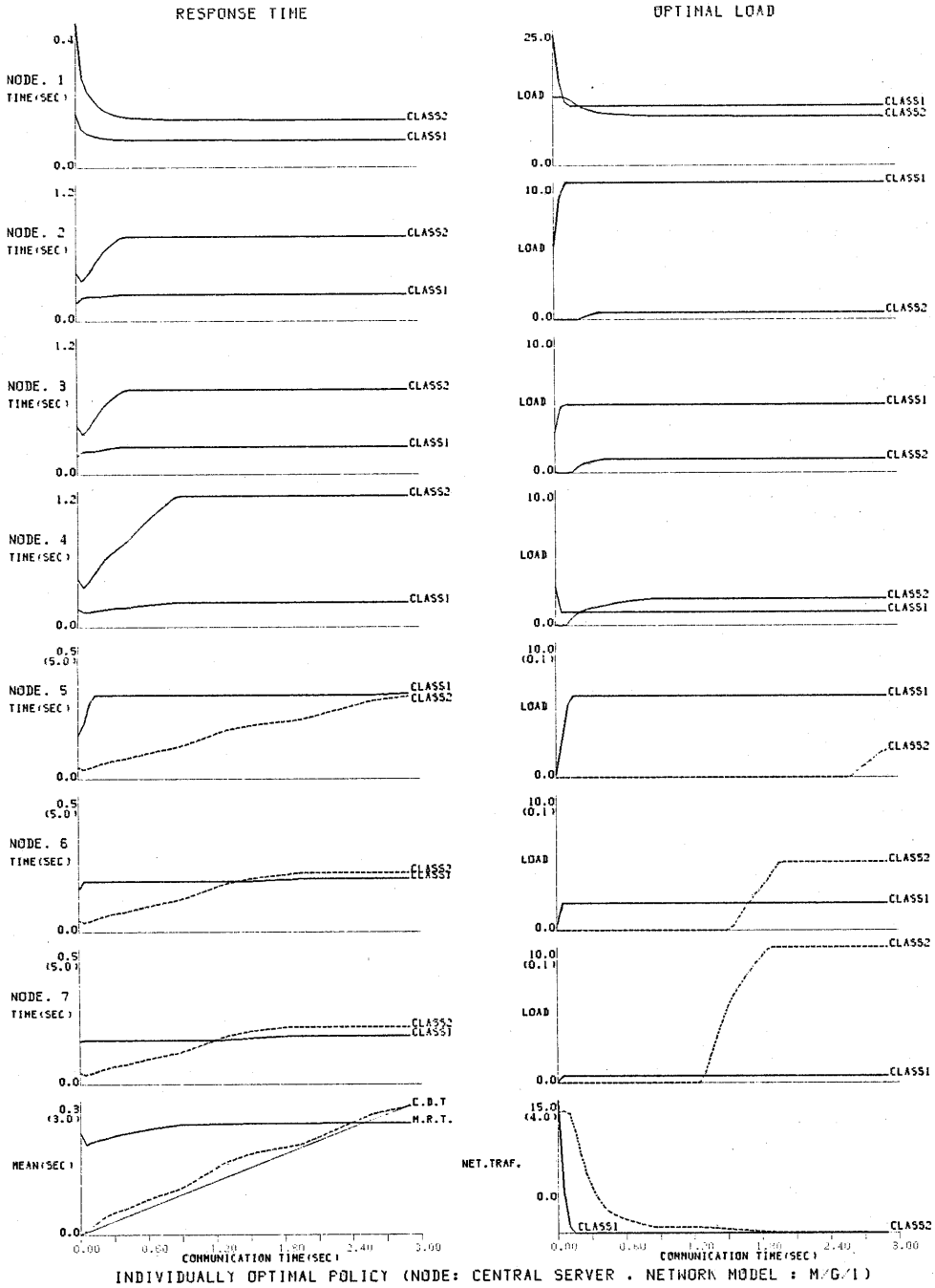


図 4. 個別最適化方式での数値実験結果