オペレーティング・システム 53－5
（1991. 12. 6）

分散システムにおける静的負荷分散方針

小澤 孝之　　亀田 壽夫

電気通信大学 情報工学科

分散型コンピュータシステムの静的負荷分散方式の一つとして、各ノード(＝ホストコンピュータ)に到着するジョブの平均応答時間が最小になるようにするノード別最適化方式を考えた。これと、システム全体の平均応答時間が最小になるようにする全体最適化方式、各ジョブ毎の応答時間の期待値が最小になるようにする個別最適化方式との性能比較をした。結果、ノード別最適化方針においても、個別最適化方針に見られるような次のような異常現象が見られた。すなわち、通信所要時間が増加するにもかかわらず、システム全体の平均応答時間が減少する異常現象が見られた。

Static Load Balancing Policies in Distributed Computer Systems

Takayuki Kozawa　　Hisao Kameda

Department of Computer Science and Information Mathematics,
The University of Electro-Communications

1-5-1 Chofugaoka, Chofu-shi, Tokyo 182, Japan

As one of static load balancing policies in distributed computer system, we introduced an intra-node optimal policy whereby the mean response time of jobs arriving at each node(=host computer) is minimized for each node. We compare the performance of this policy with that of the overall and individually optimal policies. We observed such an anomalous phenomenon that the overall mean response time may sometimes increase even though the communication time decreases, in the intra-node optimal policy, similarly as in the individually optimal policy.

1 はじめに

通信ネットワークによって、複数のコンピュータ（以下ノードと呼ぶ）を結んだ分散型コンピュータシステムの利点としては、ソフトウェア及びハードウェア資源の共有による資源の節約及び有効利用、同一のファイルを複数のノードに持たせることによる信頼性の向上、システムの負荷を分散させることによるシステムの性能の向上等が挙げられる。ここでは、バス型のネットワークシステムの静的負荷分散について、全体最適化方式 [1][3] ほど集中的でなく、個別最適化方式 [2] ほど分散的でないノード毎の最適化方式を提案し、全体最適化方式、個別最適化方式との性能比較を行なう。性能指標には、平均応答時間を用いる。

2 負荷分散について

2.1 静的負荷分散と動的負荷分散

負荷分散には

- 静的負荷分散
- 動的負荷分散

がある。静的負荷分散は、システムの統計的な情報のみによりジョブの移送を決定し、刻々と変化するシステムの状態には依存しない。これに対して動的負荷分散では、その時々のシステムの状態の情報を得る事によりジョブの移送を決定する。その時々の状態を考慮した動的負荷分散の方がより良い性能を得られると考えられている。一方で、静的負荷分散は、情報交換のオーバーヘッドがなく、また、解析的に解きやすいという利点もある。

2.2 静的負荷分散方式

静的負荷分散は、その方式により以下の 3 つが考えられる。

- 全体最適化方式
- ノード別最適化方式
- 個別最適化方式

全体最適化方式は各々のノードに到着したジョブに対して、システム全体の平均応答時間が最小になるようにジョブを分散する方式である。ノード別最適化方式は各々のノードに到着したジョブに対して、それぞれのノードに到着したジョブの平均応答時間が最小になるように負荷を分散する方式である。個別最適化方式は、各々のノードに到着したジョブに対して、全てのジョブが、他のジョブの処理ノードが決められたとして、自分の応答時間の期待値が最小になっていると思うように負荷を分散する方式である。

全体最適化方式では、システム全体の平均応答時間が最小になるようにジョブの移送を決定するために、各々ののジョブの応答時間の期待値に関しては、必ずしも最適になっているとは限らない。

逆に個別最適化方式は、どのジョブについても、他のジョブの移送先は固定しておいて、そのジョブだけを決定されたノード以外で処理しても、そのジョブの応答時間の期待値は改善されないように負荷分散される。よって、各ジョブについては応答時間の期待値は最小になるが、システム全体の平均応答時間は必ずしも最適になっているとは限らない。

ノード別最適化方式は、各々のノードに到着したジョブに関して、他のノードに到着したジョブの移送先は固定しておいて、そのノードに到着したジョブを決定されたノード以外で処理しても、そのノードに到着したジョブの平均応答時間は改善されないというように負荷分散される。よって、システム全体の平均応答時間、各ジョブの応答時間の期待値は必ずしも最適になっているとは限らない。

3 背景

Tantawi ら [1] によって、バス型ネットワークの全体最適化方式が研究され、最適解が得られた。その後、Kameda ら [2] によって個別最適化方式が検討され、最適解が得られた。さらに、Kim ら [3] によって、複数ジョブクラスにおける全体最適化方式の最適解を求める優れたアルゴリズムが提案された。

4 モデル

ここでは以下のパラメータを用いる。

n : システム内のノード数.

ϕ_i : ノードiへのジョブの外部到着率.

Φ : システム全体のジョブの外部到着率.
$\Phi = \sum_{i=1}^{n} \phi_i$

β_i : ノードiの負荷. ノードiでのジョブの処理率に等しい.

β : 負荷ベクトル. $\beta = [\beta_1, ..., \beta_n]$

x_{ij} : ノードiに到着し、ノードjに移送して処理されるジョブの割合.

x : ジョブ移送率行列. $x = [x_{ij}]$

λ_i : ノードiに到着したジョブのトラフィック量.

λ : ネットワークの全トラフィック量.

$F_i(\beta_i)$: ノードiの処理遅延. ノードiで処理されるジョブが、ノードiに到着してから処理が終了するまでの時間の期待値. β_i に関して微分可能で、凸型増加関数と仮定.

$G(\lambda)$: 通信遅延. ノードiからノードjにジョブを移送するのに必要な時間の期待値. ただし、$G(\lambda)$ はi,jの組合せによらず総トラフィック量のみに依存し、λ に関して微分可能で、凸型非減少関数と仮定.

$T(x)$: システム全体の平均応答時間.

$T_i(x)$: ノードiに到着したジョブの平均応答時間.

ここでは、図1の様なモデルを用いる。ノードiへの外部到着率はϕ_iで、その内ノードjに移送され処理されるのはx_{ij}である。逆に、ノードjに到着したジョブの内ノードiで処理されるのはx_{ji}である。ノードiの負荷はβ_iである。
また、ノードiに到着したジョブの期待値は、

$$F_j(\beta_j) + (1 - \delta_{ij})G(\lambda),$$

$$j = 1, ..., n.$$

ノードiに到着したジョブの平均応答時間$T_i(x)$は、

$$T_i(x) = \frac{1}{\phi_i} \sum_{j=1}^{n} x_{ij} \left(F_j(\beta_j) + (1 - \delta_{ij})G(\lambda) \right),$$

$$j = 1, ..., n, i = 1, ..., n.$$

システム全体の平均応答時間$T(\beta)$は、

$$T(x) = \frac{1}{\Phi} \sum_{i=1}^{n} \beta_i F_i(\beta_i) + \lambda G(\lambda).$$

となる。ここで、δ_{ij}は、$i=j$の時$\delta_{ij}=1$となり、$i \neq j$の時$\delta_{ij}=0$となる。

5 数値実験

5.1 モデル

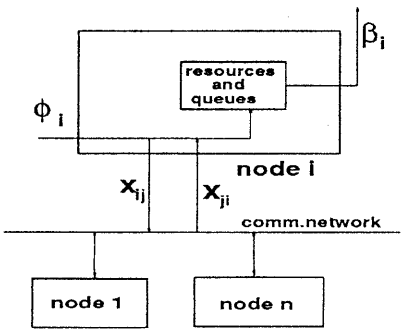

図1. バス型ネットワークモデル.

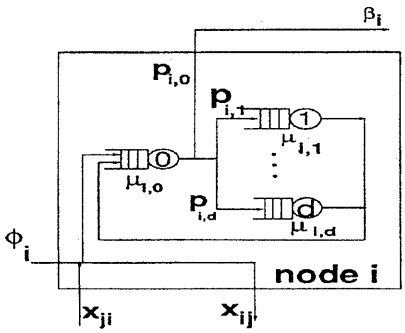

図2. セントラルサーバモデル.

ノードモデルはセントラルサーバモデルとする。セントラルサーバモデルを図2に示す。サーバ0はCPUで、サーバ1からサーバdはI/O装置である。CPUはプロセッサシェアリングで、I/O装置はFCFSとする。ノードiの外部到着率はϕ_iで、そのうちx_{ij} $(j \neq i)$がノードjに移送され処理される。また、ノードj $(j \neq i)$からx_{ji}のジョブが移送されてくる。それらのジョブは、まずサーバ0のCPUでサービスを受ける。CPUの平均サービス率は$\mu_{i,0}$である。CPUを出た後、ジョブは終了するかあるいはI/O装置のサービスを要求する。ジョブが終了する確率は$p_{i,0}$、I/O装置k (k=1,...,d)を要求する確率は、$p_{i,k}$である。I/O装置kの平均サービス率は$\mu_{i,k}$で、I/O装置のサービスを受けたジョブは再びCPUに戻る。ノードiの処理遅延$F_i(\beta_i)$は以下の変数を用いると、ノードiの処理遅延は負荷β_iにのみ依存し、

d : I/O装置の数.(CPUはd=0).

$\mu_{i,j}$: 資源(CPUまたはI/O装置)jの平均サービス率.(j=0,...,d).

$p_{i,j}$: CPUを出た後のジョブの移送の割合.

$q_{i,j}$: $q_{i,0} = \frac{1}{p_{i,j}}$,
: $q_{i,j} = \frac{p_{i,j}}{p_{i,0}}$ $\quad (j = 1, ..., d)$.

$$F_i(\beta_i) = \sum_{j=0}^{d} \frac{q_{i,j}\frac{1}{\mu_{i,j}}}{1 - q_{i,j}\frac{\beta_i}{\mu_{i,j}}}.$$

となる。jはCPUあるいはI/O装置を表し、j=0がCPU、j=1,...,dがI/O装置に対応する。

ネットワークモデルはバス型で、単一チャネル通信ネットワークのプロセッサシェアリングのM/G/1モデルとする。ここで、通信所要時間をtとすると(通信所要時間とは、ジョブの到着ノードから処理ノードへ移送される際に、ネットワークを通るのに要する時間で、待ちの時間は含まない)、通信遅延はネットワークの総トラフィック量λのみに依存して、

$$G(\lambda) = \frac{t}{1 - t\lambda}$$

となる。

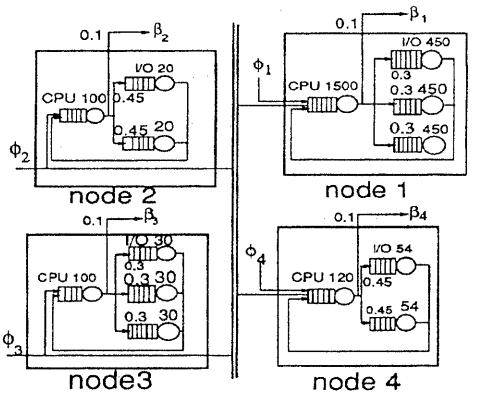

図3. 数値実験に用いたモデル.

実際に数値実験に用いたモデルは図3に示されている。ノード数は4で、それぞれ1つのCPUと複数のI/O装置を持っている。ここでは、全てのジョブは任意のノードで処理が出来るものとする。

ノード1のパラメータについて説明すると、ノード1は1つのCPUと3つのI/O装置を持っていて、ジョブの外部到着率はϕ_1(jobs/sec)、処理率はβ_1(jobs/sec)、CPUの平均サービス率は$\mu_{1,0}$=1500(jobs/sec)、各I/O装置のサービス率は$\mu_{1,1}$=$\mu_{1,2}$=$\mu_{1,3}$=450(jobs/sec)、CUPを出た後、ジョブが終了する確率は$p_{1,0}$=0.1、I/O装置1,2,3を要求する確率はそれぞれ$p_{1,1}$=$p_{1,2}$=$p_{1,3}$=0.3である。ノード2,3,4についても同様である。

5.2 実験項目

全体最適化方式、ノード別最適化方式、個別最適化方式各々について、通信所要時間に対するシステム全体の平均応答時間を計算し、比較する。

各々の方針について、通信所要時間に対する、各々のノード毎に到着したジョブの平均応答時間を比較する。

ジョブの外部到着率を変えて同様の実験を行なう。

6 結果

図4-図11に計算結果を示す。

図4、図5は、全体最適化方式、ノード別最適化方式、個別最適化方式の通信所要時間に対するシステム全体の平均応答時間を示したものである。

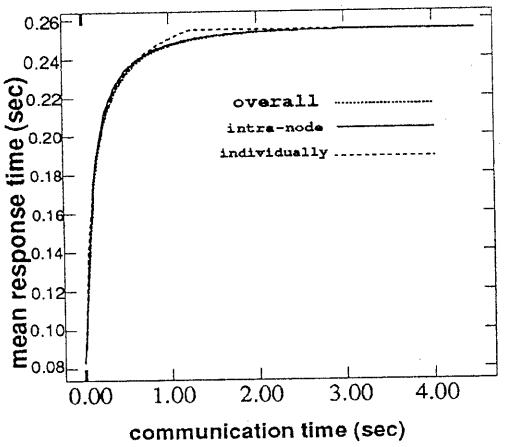

図 4. 各最適化方式のシステム全体の平均応答時間
$\phi_1 = 80.0, \phi_2 = 7.0, \phi_3 = 7.0, \phi_4 = 7.5(jobs/sec)$

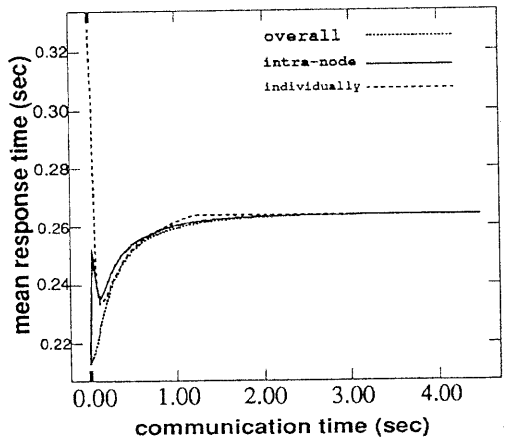

図 5. 各最適化方式のシステム全体の平均応答時間
$\phi_1 = 120.0, \phi_2 = 7.0, \phi_3 = 7.0,$
$\phi_4 = 7.5(jobs/sec)$

図 4 では、各ノードへのジョブの外部到着率は、ϕ_1=80.0, ϕ_2=7.0,ϕ_3=7.0,ϕ_4=7.5(jobs/sec) である。3 つの方式はほとんど同じ性能を示している。通信所要時間の増加に対して個別最適化方式は他の 2 つの方式に比べて比較的早く負荷の分散を止めている。

図 5 は、ノード 1 へのジョブの到着率を図 4 の場合の 80.0(jobs/sec) から 120.0(jobs/sec) に変えた時の、通信所要時間に対するシステム全体の平均応

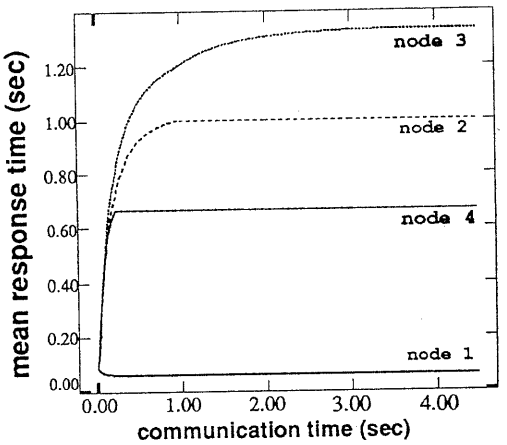

図 6. 全体最適化方式における各ノードに到着したジョブの平均応答時間
$\phi_1 = 80.0, \phi_2 = 7.0, \phi_3 = 7.0, \phi_4 = 7.5(jobs/sec)$

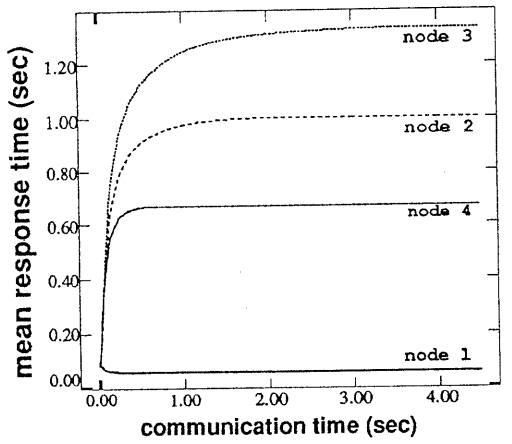

図 7. ノード別最適化方式における各ノードに到着したジョブの平均応答時間
$\phi_1 = 80.0, \phi_2 = 7.0, \phi_3 = 7.0, \phi_4 = 7.5(jobs/sec)$

答時間を示したものである。ここでは、ノード別最適化方式、個別最適化方式共に、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が減少するという異常現象が見られる。全体最適化方式では見られない。また、より分散的な個別最適化方式の方がノード別最適化方式よりもシステム全体の平均応答時間が小さくなる事がある。

図 6,7,8 は各々全体最適化方式、ノード別最適化

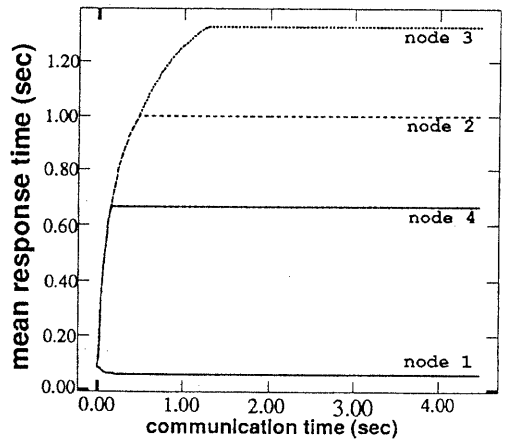

図 8. 個別最適化方式における各ノードに到着したジョブの平均応答時間
$\phi_1 = 80.0, \phi_2 = 7.0, \phi_3 = 7.0, \phi_4 = 7.5(jobs/sec)$

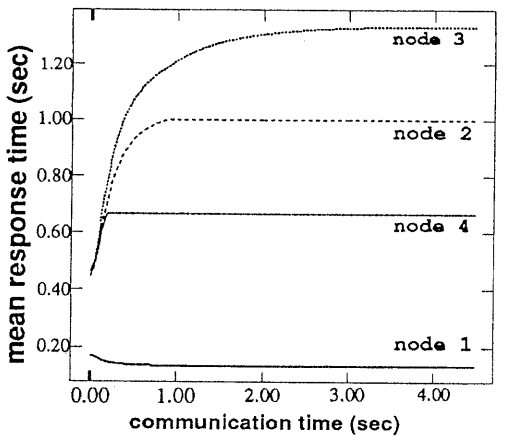

図 9. 全体最適化方式における各ノードに到着したジョブの平均応答時間
$\phi_1 = 120.0, \phi_2 = 7.0, \phi_3 = 7.0,$
$\phi_4 = 7.5(jobs/sec)$

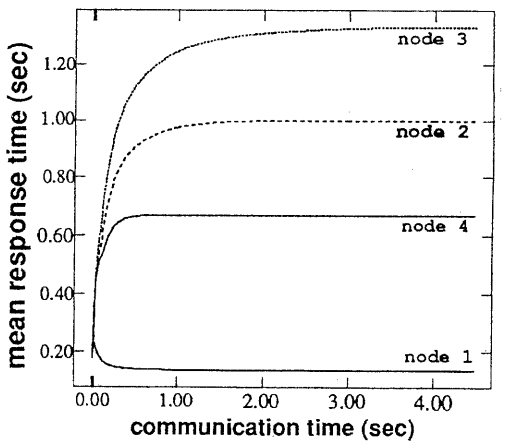

図 10. ノード別最適化方式における各ノードに到着したジョブの平均応答時間
$\phi_1 = 120.0, \phi_2 = 7.0, \phi_3 = 7.0,$
$\phi_4 = 7.5(jobs/sec)$

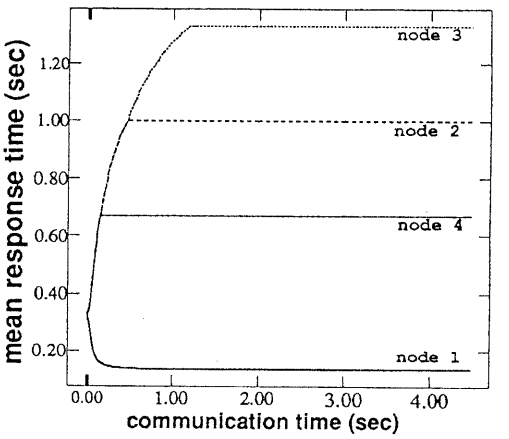

図 11. 個別最適化方式における各ノードに到着したジョブの平均応答時間
$\phi_1 = 120.0, \phi_2 = 7.0, \phi_3 = 7.0,$
$\phi_4 = 7.5(jobs/sec)$

方式、個別最適化方式の場合の各ノードへのジョブの到着率が $\phi_1 = 80.0$, $\phi_2 = 7.0$,$\phi_3 = 7.0$,$\phi_4 = 7.5$(jobs/sec) の場合の通信所要時間に対する各ノードに到着したジョブの平均応答時間を示している。全ての方式について、通信所要時間が短い時には、ノード 2,3,4 は到着したジョブの一部をノード 1 に移送している。そして、通信所要時間が増加するに従い、ジョブの移送の割合が減少している。それに従いノード 2,3,4 に到着したジョブの平均応答時間は増加し、ノード 1 に到着したジョブの平均応答時間は減少している。ノード 1 の処理能力がノード 2,3,4 の処理能力に比べて非常に大きいので、ノード 2,3,4 に到着したジョブがノード 1 で処理される割合が、ノー

ド2,3,4に到着したジョブの平均応答時間に大きな影響を及ぼす。ノード1に到着したジョブの平均応答時間の減少率に比べて、ノード2,3,4に到着したジョブの平均応答時間の増加率は大きい。

図8の個別最適化方式では、他の最適化方式に比べて通信所要時間が短いうちに負荷の分散を止めている。図6の全体最適化方式と図7のノード別最適化方式は性能がよく似ているが、ノード2とノード4についてはノード別最適化方式の方が全体最適化方式よりも通信所要時間が長くなるまで負荷の分散をしている。

図9,10,11は、$\phi_1 = 120.0$(jobs/sec)の時の全体最適化方式、ノード別最適化方式、個別最適化方式での各ノードに到着したジョブの平均応答時間を示している。$\phi_1 = 80.0$(jobs/sec)の時には、通信所要時間が0の場合には、各々の方式に対して、全ての各ノードに到着したジョブの平均応答時間がほとんど変わらなかった。$\phi_1 = 120.0$(jobs/sec)の時には、通信所要時間が0の場合には、図11の個別最適化方式のみにおいて、各ノードに到着したジョブの平均応答時間がほとんど同じ値をとる。図9の全体最適化方式では、通信所要時間が0の時でもノード2,3,4に到着したジョブの平均応答時間はノード1に到着したジョブの平均応答時間の2倍以上もある。ノード1へのジョブの到着率を80.0(jobs/ sec)から120.0(jobs/sec)に増加したにもかかわらず、ノード1に到着したジョブの平均応答時間の増加よりもノード2,3,4に到着したジョブの平均応答時間の増加の方が大きい。

一方、図10のノード別最適化方式では、通信所要時間が0付近では、処理能力の小さいノード2,3,4に到着したジョブの平均応答時間が、ノード1に到着したジョブの平均応答時間よりも小さくなっている。よって、各ノードを比較すれば、ノード別最適化方式が全体最適化方針、個別最適化方式よりも性能は良いが、図6に見られるようにシステム全体では、異常現象も見られ、3つの負荷分散方式の中では個別最適化方式よりもシステム全体の平均応答時間が大きくなることがある。

7 まとめ

本研究では、ノード別最適化方式について、より集中的な全体最適化方式、より分散的な個別最適化方式と比較を行なった。数値実験の結果より、ノード別最適化方式はジョブの到着率が比較的小さい時には、システム全体の平均応答時間は、全体最適化方式、個別最適化方式とあまり差がない事がわかる。しかし、処理能力の大きなノードへのジョブの到着率を増加すると、通信所要時間が小さい時には、平均応答時間が全体最適化方式に比べて悪化する。そして、個別最適化方式に見られるような異常現象も見られる。また、ノード別最適化方式は、より分散的な個別最適化方式よりもシステム全体の平均応答時間が大きくこともある。

ノード別最適化方式は、各ノードに到着したジョブの平均応答時間を小さくするために、システム全体の平均応答時間を犠牲にする度合がより大きいことがある。時にはその度合が個別最適化方式よりも大きくなる。

参考文献

[1] A.N.Tantawi and D.Towsley. Optimal Static Load balancing in Distributed Computer Systems, *J.ACM32*,2(April 1985), 445-465.

[2] H.Kameda and A.Hazeyama. Individual vs. Overall Optimization for Static Load Balancing in Distributed Computer Systems, *Computer Science Report, The University of Electro-Communications* (1988).

[3] C.Kim and H.Kameda. Optimal Static Load Balancing of Multi-Class Jobs in a Distributed Computer System, *Proc. 10th Intl. Conf. Distributed Comput. Syst.*, IEEE, 1990, pp.562-569.