マルチメディア通信と分散処理　45－13
（１９９０．５．２５）

通信システムにおける分散データ配置方式

重田 和弘　　高野 誠　　斎藤 勲
ＮＴＴ通信網総合研究所

多数の小規模なノードで構成された、分散通信システムにおけるデータ配置方法について議論する。分散データベースシステムを対象に行われてきた、これまでのファイル配置問題に関する研究成果が、データ参照時間に対する条件の違いにより、分散通信システムに適用できない点を指摘し、分散通信システムを対象としたデータ配置問題を新たに提案する。さらに、この問題に対して、各ノードの自律的動作によって、データの配置を変更する方式を検討し、そのアルゴリズムの一案を示す。また、ネットワークが格子状網の場合について、この方式の評価も行う。

Decentralized Data Allocation in Communication Systems

Kazuhiro Shigeta　Makoto Takano　Isao Saito
NTT Telecommunication Networks Laboratories

1-2356 Take Yokosuka-shi Kanagawa, 238-03, Japan

A decentralized data allocation method in distributed communication systems has been discussed. We formulate a data allocation problem which is suitable for distributed communication systems and we proposed a heuristic data transfer algorithm to minimize the data access time. Effectiveness of the proposed algorithm was confirmed for mesh networks which have an enough data storage capacity. Some properties on the algorithm such as convergence was studied .

1．まえがき

通信システムに対するニーズの多様化にともない、システム規模やサービスに対して、フレキシビリティの高いシステムの構成が必要になってくる。このうち、規模のフレキシビリティに対応するために、小規模なノードの集合として通信システムを構成する、分散通信システムの研究が行われている。

多数のノードで構成される分散通信システムには、データ更新コストやメモリ量の問題から、従来より集中システムで行われてきたような、各ノードごとに、そこで必要とするデータをすべて保持する方法は適用できない。分散通信システムでは、分散データベースで行われているように、各データを一部の限られたノードにのみ配置し、各ノードが必要に応じて、他ノードにあるデータを参照する方法が適している。

分散データベースにおいて、最適なデータ配置を求める問題は、以前から、ファイル配置問題として研究されている。これらの研究の多くは、各ノードの各データに対する参照頻度が与えられたとき、各ノードのファイル蓄積コストとファイル転送コストの和が最小となる、ファイル配置を求める問題を扱っている[1][2][3]。また、問題を簡単化して、0－1ナップザック問題に帰着したり[4]、データの参照頻度の代わりに、参照の有無だけが与えられたと仮定して議論したものもある[5]。ファイル配置問題はＮＰ－困難であるため、最適解ではなく近似解を求めるアルゴリズムを検討したものが多い。

分散通信システムは、電話のサービスに代表されるように実時間性が厳しく、これまで研究の対象とされてきた分散データベースシステムと比較して、データ参照時間に対する条件がより厳しく規定されるており、従来の研究成果を分散通信システムへ適用するのは困難である。そこで、本稿では、分散通信システムをモデルとした新たなデータ配置問題を提案する。

分散通信システムでは、データの参照時間を常に一定値以内に保つ必要から、各ノードのデータ参照頻度の変化に対応して、すみやかにデータの配置を変更しなければならない。そのためには、各ノードの自律動作により、各ノードがその周辺のデータについてのみ配置先を決定する、分散管理方式が良いと考えられる。しかし、このような場合については、まだ十分な検討はされていない。また、分散通信システムでは、データの配置変更の都度、配置されるデータの量に応じて、各ノードのデータ蓄積容量を増減させることはできない。従って、各ノードのデータ蓄積容量は、コストとしてではなく、制約条件として扱わねばならない。分散通信システムでは、以上の点を考慮して、データ配置問題を検討する必要がある。

そこで、本稿では、この点を考慮したデータ配置問題に対して、データの移動を、各ノードの自律的な動作で行う、自律分散型動的データ配置方式を提案し、その具体的なアルゴリズムの一案も示す。

2．自律分散型動的データ配置方式

2．1　対象システム

本稿では、網構成に関してフレキシビリティのある通信システムについて検討するため、対象とする通信システムの網形態に制限は与えない。また、各ノードには、全データの所在が常に正しく与えられ、各ノードは、自由に他ノードのデータを参照できるものとする。

データの所在を正確に管理し、他ノードへのデータ参照を常時可能とするためのデータ所在管理法としては、同報により、データの移動を全ノードに逐次知らせる方法など、種々のものが考えられ、研究課題の１つであるが、本稿ではこの具体的な実現方法については議論しない。

2．2　問題の定式化

本稿では、問題を定式化するにあたって、（1）式で表される、総データ転送量Zを目的関数とする。この値は、データの参照時間がデータの転送距離に比例するとき、システム全体の平均データ参照時間と比例関係にある。

データのコピーに関しては、各ノードが同じ内容の複数のデータを参照しない環境を想定すれば、データの配置において、コピーの存在を意識する必要がないものも考えられる。従って、本稿では、データのコピーは作成しないとして議論する。

以下に本稿で取り扱った問題を定式化する。

$$\text{目的}\quad \min\quad Z=\sum_{i=1}^{M}\sum_{j=1}^{N} a_{ij}x_{ij} \qquad \cdots\cdots(1)$$

$$\text{ただし、}\quad a_{ij}=\sum_{k=1}^{N} b_{ik}\cdot d_{jk}$$

$$\text{制約}\quad \sum_{i=1}^{M} x_{ij}s_i \leqq c_j\ ,\ \forall j=1,\cdots,N \qquad \cdots\cdots(2)$$

$$\sum_{j=1}^{N} x_{ij}=1\ ,\ \forall i=1,\cdots,N \qquad \cdots\cdots(3)$$

$$x_{ij}=1\ \text{or}\ 0 \qquad \cdots\cdots(4)$$

ここに、

N：ノード数

ノードには、1、2、…、Nの番号が順に与えられているとする。

M：データ数

データには、1、2、…、Mの番号が順に与えられているとする。

b_{ik}：ノードkがデータiを参照する頻度

d_{jk}：ノードjとノードkの距離

c_j：ノードjのデータ蓄積容量

s_i：データiのサイズ

$$x_{ij}=\begin{cases}0：\text{データiがノードjに配置されていない}\\ 1：\text{データiがノードjに配置されている}\end{cases}$$

この問題は、0－1整数計画問題の一種で、NP－困難であることがわかっており、多項式時間で解くアルゴリズムは存在しないと考えられている。

本稿では、この問題の近似解を、システム内の各ノードが自律的に動作することによって求める方法を検討する。

検討にあたっては、議論を簡単にするため、次の仮定をおく。

「各ノードは、自ノードに配置されているデータに関して、自ノードおよび隣接ノードのうち、そのデータに関するデータ転送量を最も小さくできる場所が、どのノードであるか判定できる。」

この仮定は、以下の2つが保証されれば、実現可能である。

①各ノードに、自ノードおよびその隣接ノードからシステム全体の各ノードへの距離が与えられる。

②各ノードは、自ノードに配置されているデータに関して、データ毎に参照ノードとその参照頻度がわかる。

2．3　データ配置方式

図1に、自律分散型動的データ配置方式の処理手順を示す。

最初は、各ノードのデータ蓄積容量を越えない

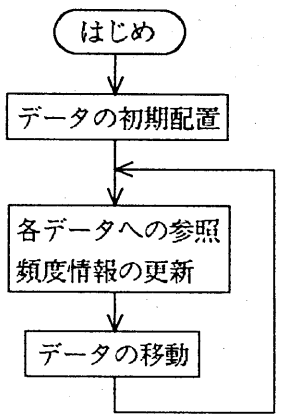


図1　自律分散型動的データ配置方式

ように、適当にデータを配置する。次に、各データが参照される頻度を、ノード対応に定期的に調べ、それをもとに、各ノードが、自ノードに配置されているデータを、データ転送量ができるだけ小さくなるノードへ移動させる。データへの参照頻度情報の更新とデータの移動を繰り返し実行し、データをデータ転送量のより小さい場所へ移動させる。この方法は、データへの参照頻度が途中で変化する場合にも対応できる。

データ移動の一手法を3章に示す。

3．データ移動アルゴリズム

3．1　アルゴリズム

自律分散型動的データ配置方式における、データ移動アルゴリズムを以下に示す。

以下の処理を各ノードでそれぞれ並列に実行する。

1.　各データの移動先を求めるノードを決定する。

1.1　自ノードに配置されているデータについて、自ノードおよび隣接ノードのうち、そのデータに関するデータ転送量が最小になるノードを求める。

1.2　自ノードに配置されているデータについて、1.1で求めたノードへ、データ名とそのデータへの参照頻度情報を、移動先決定要求とともに送る。ただし、1.1の結果が自ノードだったデータは除く。

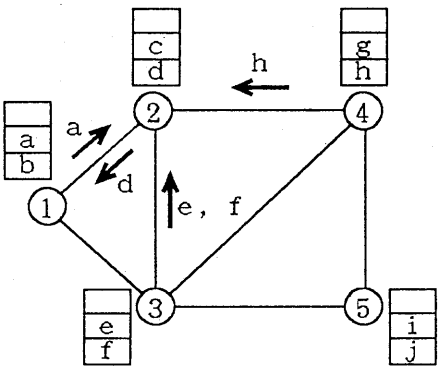
2.　データの移動先を求める。

2.1　隣接ノードから移動先決定要求のあったデータ、および1.1の結果が自ノードだったデータの移動先を、データの配置場所を自ノードおよび隣接ノードに限定した、部分データ配置問題を解いて求める。

ただし、隣接ノードのデータ蓄積容量は、そのノードから自ノードへ、移動先決定要求のあったデータの総量と同じとする。また、自ノードについては、自ノードのデータ蓄積容量のうち、他ノードへ 移動先決定要求を送った、データの総量を除いた量とする。

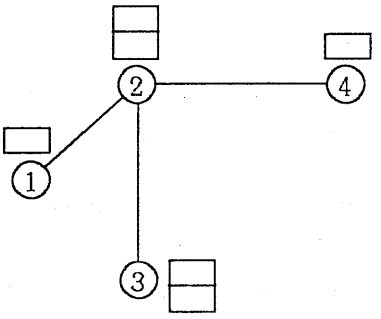
2.2　2.1で求めた配置を、データ移動先として、関係ノードに知らせる。

3.　自ノードのデータについて、2.で求めた場所へ、データを移動させる。



（a）

矢印は各データのデータ移動先決定要求が出された方向を示す。



（b）

ノード2は、データa，c，e，f，hを、自ノードおよび隣接ノード（ノード1、2、3、4）に配置する。

a～j：データ　　：データ蓄積スペース

図2　部分ファイル配置問題の例

各ノードから（a）に示されるように、データ移動先決定要求が出されたなら、ノード2が解く部分データ配置問題は、（b）のようになる。

1.で、各データの移動先を決定するノードを、それぞれ定める。次に、2.において、各ノードは、自ノードで移動先を決定することになったデータについて、配置先を自ノード及び隣接ノードに限定した、部分データ配置問題を解き、データの移動先を求める。最後に、3.で2.の結果に基づいてデータを移動させる。

2.1で、各ノードがそれぞれ解く部分データ配置問題は、対象データやデータ蓄積容量に関して重複がない。従って、各ノードの処理が厳密には同期していない場合や、一部のノードの処理が停止している場合でも、他のノードは処理を継続できる。ただし、このような場合は、データ配置を求める能力は、低下すると考えられる。

2.1における、部分データ配置問題の例を、図2に示す。また、この問題の近似解を求めるアルゴリズムを次に示す。

3．2　部分データ配置問題の解法

以下に、データ移動アルゴリズムの2.1で、部分データ配置問題の近似解を求めるアルゴリズムを示す。

1.　各ノードのデータ蓄積容量を無視して、データ転送量が最小になるノードへ、データを割り当てる。
2.　すべてのノードにおいて、割り当てたデータの量が、そのノードのデータ蓄積容量を越えていなければ、その割当を解として処理を終了し、そうでないときは、3.の処理を行う。
3.　データ蓄積容量を越えて割り当てられたノードのデータについて、以下の処理を行う。
3.1　現在割り当てられているノードの次に、データ転送量が小さくて済むノードを求める。
3.2　3.1で求めたノードへ、新たにデータを割り当てたときの、データ転送量の増加量を各データ毎に求める。
3.3　3.2で求めた増加量が大きいものから順に、データ蓄積容量を越えない量のデータを、今のノードにそのまま割当て、残りのデータは、3.1で求めたノードへ、新たに割り当てる。
3.3　2.の処理に戻る。

部分データ配置問題の解法については、扱うデータ数が少ないときは、ここに示した近似アルゴリズムではなく、最適解を求めるアルゴリズムを利用することにより、より精度の高い解を求めることができる。なお、このアルゴリズムは、各ノードのデータ蓄積容量に制限がないときに限り、常に最適解が求まる。

4．数値例による評価

ノード数が１６，２５，３６，４９，６４，８１，および１００の７通りの格子状網について、３章で提案した自律分散型動的データ配置方式を対象として、図3に示す条件で、シミュレーションを行った。データの移動は、各ノードで一斉に行われるとし、これを1回のデータ移動とした。

シミュレーション結果は、データの初期配置に影響されるが、ほぼ同様の傾向を示すことから、以下の考察では1例のみを示す。

4．1　総データ転送量の改善量に関する考察

4．1．1　データ蓄積容量に制限がない場合

各ノードのデータ蓄積容量に制限がないときの、総データ転送量とデータ移動回数の関係を、ノード数が４９，１００の場合について、図4に示す。ただし、総データ転送量は、初期配置時の値で正規化した。また、参考値として、3章で述べた部分データ配置問題のアルゴリズムを用いて得られる、最適配置時の総データ転送量を図中に記した。

図4より、ノード数が４９，１００のいずれの場合も、本方式で最適なデータ配置が得られている。

（1）データ数（M）
ノード数の10倍
（2）各ノードのデータ蓄積容量（c_j）
すべてのノードで等しく、10，12，15，20，30，50，100，∞
［c_j（$1 \leqq j \leqq N$）$= c$］
（3）データのサイズ（s_i）
すべてのデータが等しく1
［s_i（$1 \leqq i \leqq M$）$= s = 1$］
（4）各リンクの長さ
すべてのリンクが等しく1
（5）データアクセスの経路
常に最短経路をとる
（6）アクセス条件
各データをシステム全体の20％のノードがアクセスするとし、アクセス回数は、1つのノードにつき、0から20回の範囲の値を等確率でとる。
この値は、乱数により決定する。
（7）データの初期配置
各ノードに10個づつデータを配置する。ただし、どのノードにどのデータを割り当てるかは、乱数により決定する。

図3　シミュレーション条件

4．1．2　データ蓄積容量に制限がある場合

（1）総データ転送量とデータ移動回数の関係

各ノードのデータ蓄積容量に制限がある場合の一例として、データ蓄積容量が12のときの、総データ転送量とデータ移動回数の関係を、ノード数が49，100の場合について、図5に示す。ただし、総データ転送量は、初期配置時の値で正規化した。参考値として、3章で述べた、部分データ配置問題の近似解を求めるアルゴリズムで得られたデータ配置の、総データ転送量を図中に記した。

図5より、各ノードのデータ蓄積容量に制限がある場合は、本方式による総データ転送量の改善量は、参考値と比較して、ノード数49の場合で7．8％、ノード数100の場合で31％の誤差がある。参考値は、近似解であるため、実際には、さらに大きな誤差があると予想される。従って、データ蓄積容量の余裕が少ない場合は、ノード数が大きくなるにつれ、最適解からの誤差も大きくなってゆくものと推測される。

（2）改善率とデータ蓄積容量の関係

本方式の評価尺度として、（5）式で定義する改善率を導入する。

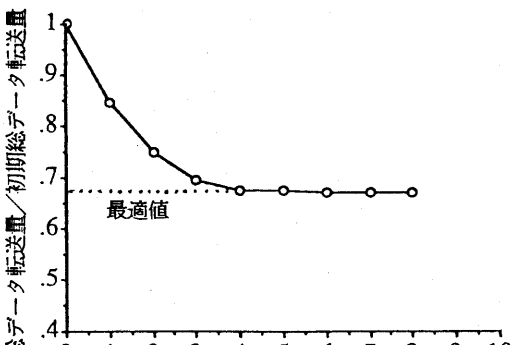
$$\text{改善率} = \frac{\text{データ移動停止時の総データ転送量}}{\text{初期配置時の総データ転送量}} \quad \cdots\cdots \quad (5)$$

改善率とデータ蓄積容量の関係を、ノード数が25，49，および100の場合について、図6に示す。

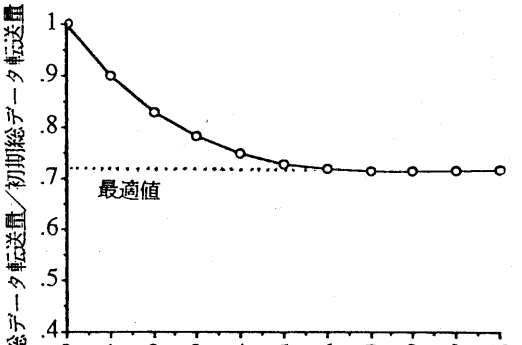
図6より、データ蓄積容量が30以上の、データ蓄積容量に十分余裕ある場合は、データ蓄積容量に制限がない場合と改善率にほとんど差がないことから、ノード数に関係なく、ほぼ最適な配置が得られていると推定できる。

一方、データ蓄積容量が10から30の、データ蓄積容量に余裕がない場合は、データ蓄積容量に十分余裕ある場合と比較して、改善率が著しく低下している。この理由として、次の2つが考えられる。一つは、データ蓄積容量が小さくなることで、データの移動がスムーズに行われなくなり、良い場所へデータを移動出来なくなるというものである。もう一つは、データ蓄積容量が小さくなるのに伴い、問題の最適解自体が悪くなっているというものである。これらの2つのうち、どちらが支配的かは、まだ明らかにしておらず、データ蓄積容量に余裕が少ない場合の本アルゴリズムの評価については、今後の検討課題である。

以上より、各ノードのデータ蓄積容量に十分余裕がある場合は、本方式で、ほぼ最適配置が得られることが明らかになった。

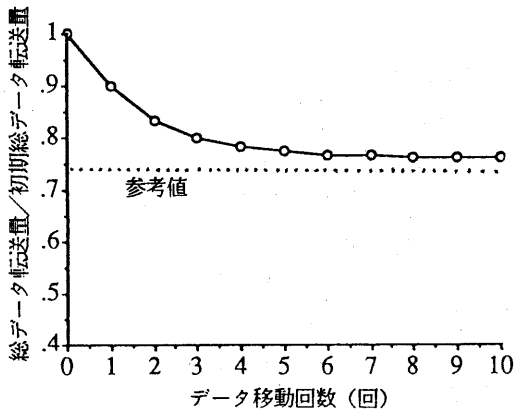


（ａ）ノード数が４９の場合

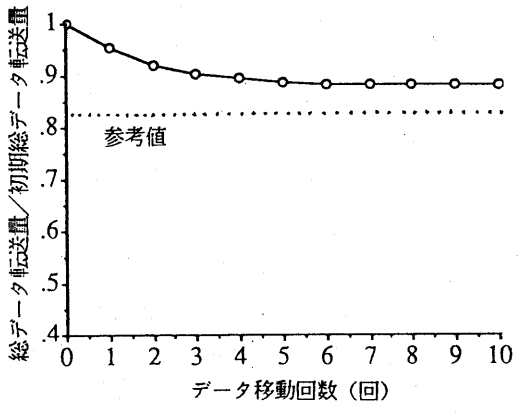


（ｂ）ノード数が１００の場合

図４　総データ転送量とデータ移動回数の関係
（データ蓄積容量に制限のない場合）



（ａ）ノード数が４９の場合



（ｂ）ノード数が１００の場合

図５　総データ転送量とデータ移動回数の関係
（データ蓄積容量が１２の場合）

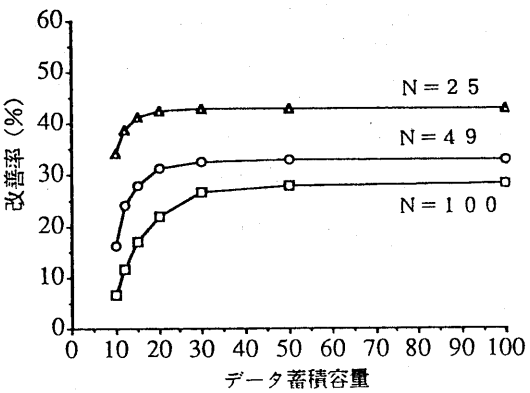


図６　改善率とデータ蓄積容量の関係

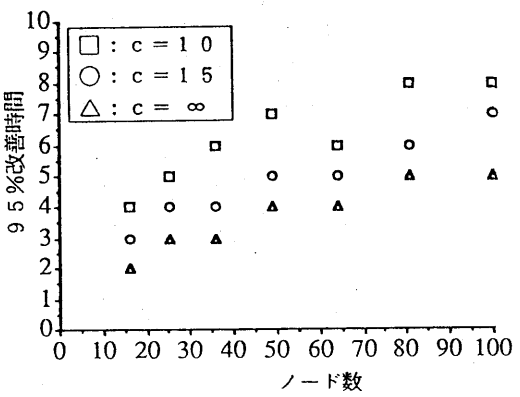


図７　９５%改善時間とノード数の関係

４．２　改善速度に関する考察

図４、図５より、本方式は、データ移動のごく初期の段階で、最終的に得られる改善量とほぼ等しい改善がなされている。そこで、改善速度の評価尺度として、９５％改善時間を導入する。ただし、９５％改善時間は、改善量がデータ移動停止時の値の９５％を越えるまでに要する、データ移動の回数で与える。９５％改善時間が小さいということは、改善速度が早いことを意味する。

９５％改善時間とノード数の関係を、データ蓄積容量が１０，１５，および∞の場合について、図7に示す。

図7より、ノード数をＮとすると、９５％改善時間は、ほぼ$\sqrt{N}$に比例している。格子状網では、ノード間の平均距離は、$\sqrt{N}$のオーダーであることを考慮すると、本方式の改善速度は十分速いといえる。

５．あとがき

分散通信システムにおけるデータ配置方法について議論した。分散データベースシステムを対象に行われてきた、これまでのファイル配置問題に関する研究成果が、データ参照時間に対する条件の違いにより、分散通信システムに適用できない点を指摘し、分散通信システムを対象としたデータ配置問題を、新たに提案した。さらに、分散通信システムにおけるデータ配置方式の１手法を示し、格子状網に対して、その方式の評価を行った。その結果、格子状網については、各ノードのデータ蓄積容量に十分余裕がある場合は、今回提案した方法で、ほぼ最適なデータ配置が得られることが明らかになった。また、このデータ配置を得るために要するデータの移動回数が、ノード数をＮとするとき、$\sqrt{N}$に比例することも確認した。

今後は、まだ明らかにされていない、データ蓄積容量の余裕が少ない場合について、およびネットワークモデルが格子状網以外の場合について、今回提案した方式の評価を行い、さらに、データ移動アルゴリズムの改良等も行う予定である。また、本研究では、データの所在管理方式については検討しなかったが、これも重要な問題であり、今後の課題とする。

謝辞

本検討を進めるにあたり、有益な御助言、御討論を頂いた、ＮＴＴ通信網総合研究所斎藤孝文グループリーダ、ＮＴＴソフトウエア研究所今瀬真主幹研究員、ならびにＮＴＴ交換システム研究所久保田稔主任研究員に深謝致します。

参考文献

[1] Wesley W. Chu, “Optimal file allocation in a multiple computer sysytem”, IEEE Trans. Comput., C-18, pp. 885-889(1969)

[2] Casey. R. G., “Allocation of copies of a file in a information network”, proc. SJCC, AFIPS, pp. 617-625(1972)

[3] James F. Kurose, “A microeconomic approach to optimal file allocation”, IEEE Trans. Comput., C-38, pp705-717(1989)

[4] S. Ceri, G. Martella, and G. Pelagatti, “Optimal file allocation in a computer network :a solution method based on the knapsack problem”, Computer Network 6, pp. 345-357(1982)

[5] 菊野、吉田、杉原、角田, “ユーザの要求に基づくファイル配置問題の計算複雑さ”, 信学論（D）, J-65D No. 4, pp. 458-463(1982)