情報処理学会第39回(平成元年後期)全国大会

2T-3 固定重み付き分散マッチメイキングのプロセスグループ構成法

中島　周

日本アイ・ビー・エム株式会社　東京基礎研究所

1．はじめに

分散マッチメイキングは，ネットワーク内のネームサービスや排他制御を分散的に実行するための方法である[1]．その効率は実行時に必要となる通信量で表され，その値を決めるのはある性質を持ったプロセスグループの要素数である．[1] では重みなしの分散マッチメイキングの効率の下限が示され，プロセス数nが自然数の2乗に等しい場合についていくつかのプロセスグループの構成例が示された．本稿では，各プロセスが同じ負荷を分担する場合について，一般の自然数nに対して最適なプロセスグループ構成法を与え，その構成法の応用が固定重み付きの場合にも効率の良いプロセスグループを作ることを示す．

2．分散マッチメイキング

ネットワーク内にn個のプロセスが存在し，それらには1からnまでの識別子がついているとする．このとき各プロセスにプロセスの集合を対応させる関数P(i)，Q(i) を考える，このP(i)，Q(i) が $1 \leqq i, j \leqq n$ であるすべてのi，jについて

$$|P(i) \cap Q(j)| \neq \phi \qquad (1)$$

を満足すれば，この性質を利用して分散ネームサービス[1] や分散排他制御[2] を実現することができる．たとえば，プロセスiがプロセスグループP(i) にそのアドレスを教えているとき，プロセスjがプロセスグループQ(j) にプロセスiのアドレスを尋ねればそのうちの1つ以上のプロセスがプロセスiのアドレスを返すことができる（図1）．この定式化では，ただ1つのネームサーバが存在する場合，アドレス検索要求をブロードキャストする場合，自分のアドレスをすべてのプロセスにブロードキャストして教える場合など，種々のネームサービスの形態を取り扱うことができる．

分散マッチメイキングの実行時の効率は

$$C = |P(i)| + |Q(j)| \qquad (2)$$

によって決められる．すべてのプロセスが平等に負荷を分担する場合にはCの下限値は $2\sqrt{n}$ になる[1]．

3．重みなしの場合のプロセスグループ構成法

本稿では，以下，すべてのプロセスが同じアルゴリズムを実行し，すべてのプロセスが等しく負荷を分担する場合，つまりプロセス間に対称性のある場合について考える．

[1] ではnが自然数の2乗である場合に，プロセス識別子を $\sqrt{n} \times \sqrt{n}$ に並べ，iが属する行をP(i)，iが属する列をQ(i) として，すべてのプロセスが等しく処理を負担する場合の最適なプロセスグループの構成法を与えた．しかしこの方法はnが自然数の2乗でない場合には使えない．以下では，各プロセスの対称性を最小限度くずすことを許して，一般の自然数nに対して有効なP(i)，Q(i) の構成法を示す．

まず，列の数を $\lceil\sqrt{n}\rceil$ にして，プロセス識別子で行列を作る．最後の行のあいているところにはさらに1から順にプロセス識別子をうめていく．そして，iが属する列をP(i)，iが属する行をQ(i) とする．ただし行列中に2度出現する識別子については，最初に出現する場所でP(i)，Q(i) を構成する（図2）．この構成法はCの下限値を与え，その値は

$n > \lceil\sqrt{n}\rceil \cdot (\lceil\sqrt{n}\rceil - 1)$ のとき $2\lceil\sqrt{n}\rceil$，

$n \leqq \lceil\sqrt{n}\rceil \cdot (\lceil\sqrt{n}\rceil - 1)$ のとき $2\lceil\sqrt{n}\rceil - 1$

である．前者の例としてn＝47のときのC＝14，，後者の例としてn＝41のときのC＝13があげられる．

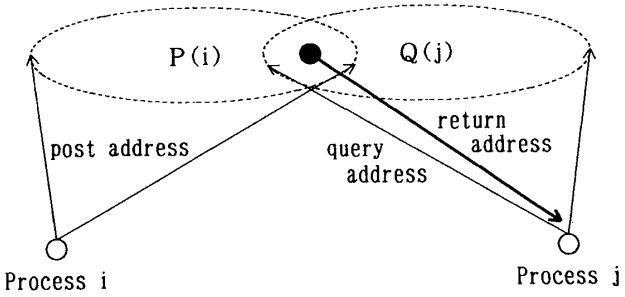

図1　ネームサービスの場合の分散マッチメイキング

A Process Grouping Method for Distributed Match-Making with Fixed Weight

Amane NAKAJIMA

IBM Research, Tokyo Research Laboratory

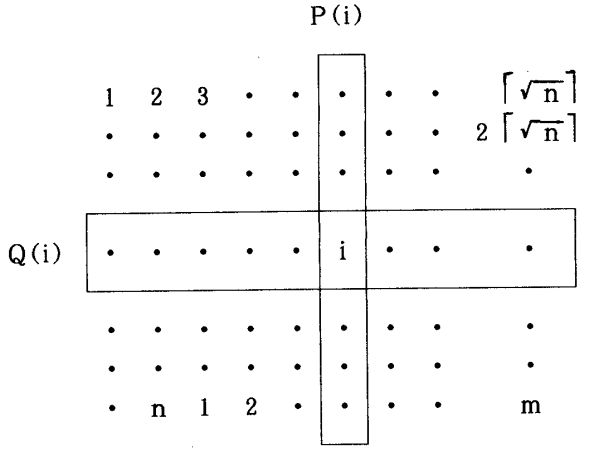

図2　重みなし分散マッチメイキングのグループ構成法

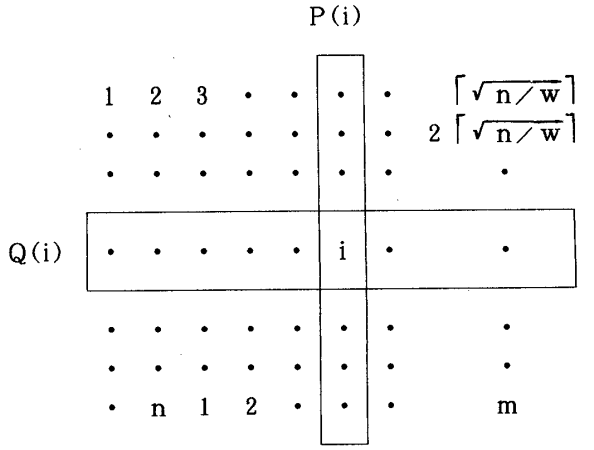

図3　固定重み付き分散マッチメイキングのグループ構成法

4．固定重み付きの場合のプロセスグループ構成法

最初にネームサービスを例にとり，固定重み付きの持つ意味を考える．2．で示したような分散ネームサービスを行う場合，プロセスiが生成されたとき，またはプロセスiが存在するプロセサを移動したときは，プロセスは｜P(i)｜個のプロセスに新しい自分のアドレスを知らせる．また，プロセスiが他のプロセスのアドレスを知りたいときは，｜Q(i)｜個のプロセスにそれを尋ねる．よって，前者と後者の比が1：wのときの実行時のコストは(2) ではなく

$$C_w = |P(i)| + w|Q(j)| \qquad (3)$$

で表される．そこで，この値を小さくするプロセスグループの一般的な構成法が必要となる．これを以下で述べる．

まず，$\sqrt{n/w}$が自然数の場合を考える．3．のようにa×bの行列を作り，P(i)，Q(i) を構成したとする．このとき

$$a \cdot b \geqq n \qquad (4)$$

$$C_w = a + bw \qquad (5)$$

である．(4) の条件下でのC_wの最小値は，

$$a = \sqrt{nw},\quad b = \sqrt{n/w} \qquad (6)$$

のときの$2\sqrt{nw}$となる．つまり列の数が$\sqrt{n/w}$，行の数が$\sqrt{nw}$の行列を作り，3．のようにしてP(i)，Q(i) を決めればよい

nが一般の自然数の場合は，列の数が$\lceil\sqrt{n/w}\rceil$になるようにして3．と同じ方法で行列を作る（図3）．このときC_wの値は

$$\left\lceil \frac{n}{\lceil\sqrt{n/w}\rceil} \right\rceil + w\lceil\sqrt{n/w}\rceil \qquad (7)$$

になる．n＝10，w＝2の場合の構成例を表1に示す．

表1　固定重み付きグループの例（n＝10，w＝2）

i	P(i)	Q(i)
1	1 4 7 10	1 2 3
2	1 2 5 8	1 2 3
3	2 3 6 9	1 2 3
4	1 4 7 10	4 5 6
5	1 2 5 8	4 5 6
6	2 3 6 9	4 5 6
7	1 4 7 10	7 8 9
8	1 2 5 8	7 8 9
9	2 3 6 9	7 8 9
10	1 4 7 10	1 2 10

5．おわりに

固定重み付き分散マッチメイキングの効率の良いプロセスグループ構成法を述べた．この構成法は任意の自然数のプロセス数に対して有効である．本方法は固定重みに対するものであるが，実際への応用では固定重みだけで十分のものが多いと考えられる．

参考文献

[1] Mullender, S. J. and Vitányi, P. M. B.: Distributed Match-Making, *Algorithmica*, Vol. 3, No. 3, pp. 367-391 (1988).

[2] Maekawa, M.: A $\sqrt{N}$ Algorithm for Mutual Exclusion in Decentralized Systems, *ACM Trans. on Computer Systems*, Vol. 3, No. 2, pp. 145-159 (1985).