2Q-04

異種ワークステーションネットワークにおけるマルチキャストルーティングアルゴリズム*

澤畠 真也　渋沢 進†

茨城大学工学部情報工学科‡

1 はじめに

これまで、異種ワークステーションネットワーク (HNOW) システムにおけるブロードキャストやマルチキャストのための発見的アルゴリズムがいくつか提案されてきた [1,2,3]。文献 [2] は、FNF という発見的アルゴリズムを提案し、10 個のノードまでの HNOW システムにおいて、最適に近い解を達成したことを示した。文献 [3] では、処理ノードとネットワークの両方を異種とする分散異種ネットワークのコミュニケーション構成を示し、FEF、ECEF という 2 つの発見的アルゴリズムを提案した。その実験結果では、FEF と ECEF が分散異種ネットワークにおいて FNF より優れていると示した。しかし、これらのアルゴリズムはブロッキングモデルを想定している。実際には、多くのネットワークやオペレーティングシステムは、ノンブロッキングモデルを使用することができる。そこで、本論文ではノンブロッキングモデルの使用の有効性を示し、その有効性を考慮したマルチキャストのためのアルゴリズムを提案する。

2 メッセージ通信モデル

本論文では、以下の 3 つのパラメータを用いて、送信ノード P_i と受信ノード P_j の間のメッセージ通信コストを表す。

- $O_{send}(i)$: 送信ノード P_i でかかる送信オーバーヘッド
- $X(i,j)$: P_i,P_j 間ネットワークでかかる伝達時間
- $O_{recv}(j)$: 受信ノード P_j でかかる受信オーバーヘッド

これらのパラメータは、それぞれ以下のように定義できる。

$$O_{send}(i) = S_c(i) + S_m(i) \cdot m \quad (1)$$

$$X(i,j) = X_c(i,j) + X_m(i,j) \cdot m \quad (2)$$

$$O_{recv}(j) = R_c(j) + R_m(j) \cdot m \quad (3)$$

$S_c(i)$, $X_c(i,j)$, $R_c(j)$ は、それぞれメッセージサイズに依存しないコスト、$S_m(i) \cdot m$, $X_m(i,j) \cdot m$, $R_m(j) \cdot m$ は、それぞれメッセージサイズに依存するコストである。

これらのパラメータを利用し、送信ノード P_i から受信ノード P_j への伝達時間 $T(i,j,m)$ は式 (4) のように表すことができる。

$$T(i,j,m) = O_{send}(i) + X(i,j) + O_{recv}(j) \quad (4)$$

3 既存のマルチキャストアルゴリズム

本論文では、ノードを 2 つの集合に分ける。集合 A は既にメッセージを受信しているノード、集合 B は目的ノードからなる。目的ノードとは、メッセージを受信すべきノードのことである。また、パラメータとして、$Available(i)$ を用意する。$Available(i)$, $Available(j)$ は、それぞれ送信ノード P_i, 受信ノード P_j の送信可能時間であり、次の送信操作ができるようになるまでの時間を示す。ブロッキングモデルにおける送信可能時間は式 (5),(6) によって求められ、各繰り返しの際に更新される。

$$Available(i) \leftarrow Available(i) + T(i,j,m) \quad (5)$$

$$Available(j) \leftarrow Available(i) + T(i,j,m) \quad (6)$$

- **FEF アルゴリズム**： FEF アルゴリズムでは、集合 A から送信ノードが、集合 B から受信ノードが選ばれる。このとき選ばれるノードは、$T(i,j,m)$ が最小となる送信ノード P_i と受信ノード P_j である。選ばれた受信ノード P_j は、マルチキャスト木に付け加えられ、集合 B から集合 A に移す。これらの操作を集合 B が空になるまで繰り返す。
- **ECEF アルゴリズム**： ECEF アルゴリズムも FEF アルゴリズムと同様に、集合 A から送信ノードが、集合 B から受信ノードが選ばれる。このとき選ばれるノードは、$Available(i) + T(i,j,m)$ が最小となる送信ノード P_i と受信ノード P_j である。選ばれた受信ノード P_j は、マルチキャスト木に付け加えられ、集合 B から集合 A に移す。これらの操作を集合 B が空になるまで繰り返す。

マルチキャストの完了時間 $CompleteTime$ は、式 (7) によって表すことができる。

$$CompleteTime = \max_{P_i \in A} Available(i) \quad (7)$$

4 ノンブロッキングモデルの導入

文献 [2,3] におけるアルゴリズムは、全てブロッキングモデルを想定している。これは、送信ノードがメッセージを送信する際、受信ノードが完全に受信し終えるまで、他のノードに送信することができないというものである。しかし、本研究ではノンブロッキングモデルを想定する。ノンブロッキングモデルとは、送信ノードがメッセージを送信する際、送信するメッセージは送信ノードの介入なしで伝達される。つまり、送信オーバーヘッド後に送信ノードは他の受信ノードへメッセージを送信することができる。

ノンブロッキングモデルの有効性を図 1 に示す。(a) はブロッキングモデルを、(b) はノンブロッキングモデルを表している。なお、マルチキャスト木の分岐点は、左側の辺から順に送信すると仮定する。つまり、図 1 の場合は $(P_0,P_1),(P_0,P_2),(P_0,P_3)$ の順で送信される。辺の隣の数値は、送信ノード P_0 から各受信ノードまでの通信時間を示し、T_i は (P_0,P_3) 後の各ノードの送信可能時間を示す。(a) の $CompleteTime$ が 340 であるのに対し、(b) の $CompleteTime$ は 160 である。このことからノンブロッキングモデルを想定することが、極めて有効であることが分かる。

ノンブロッキングモデルの場合、各ノードの送信可能時間は、式 (8),(9) で求めることができる。

$$Available(i) \leftarrow Available(i) + O_{send}(i,m) \quad (8)$$

$$Available(j) \leftarrow Available(i) + T(i,j,m) \quad (9)$$

ノンブロッキングモデルの場合、マルチキャスト木を作り、実際にメッセージを送信する時に、どの順序で送信するかが重要になる。図 2 は、その送信順の重要性を示している。(a),(b) はブロッキングモデルを、(c),(d) はノンブロッキングモデルを、それぞれ示している。送信ノードは、左側のノードから先

*Multicast Routing Algorithm for Heterogeneous Network of Workstations

†Shinya Sawahata, Susumu Shibusawa

‡Department of Computer and Information Sciences, Faculty of Engineering, Ibaraki University

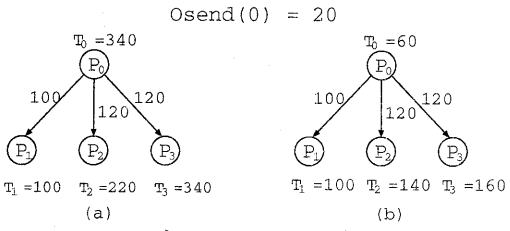

図 1: ノンブロッキングモデルの有効性

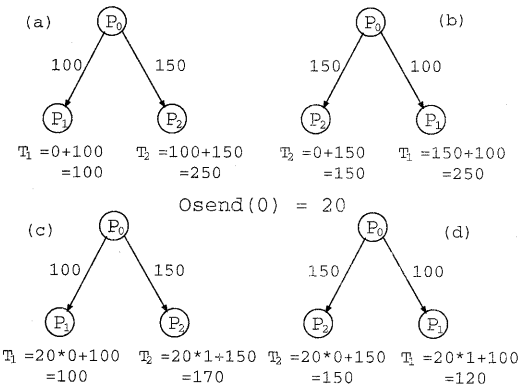

図 2: ノンブロッキングモデルにおける送信順の重要性

に送信すると仮定すると、(a),(b) では、最終完了時間はその送信順に依存していないが、(c),(d) では、その送信順に依存している。このことより、マルチキャスト木の各分岐点での送信順が重要になる。

5 マルチキャスト木修正アルゴリズム

4節で述べたように、ノンブロッキングモデルは、その送信順の重要性を考慮する必要がある。これは、送信オーバーヘッド後に送信ノードは、他のノードへメッセージを送信することができるため、ある分岐点において後の方に送信されるノードやその子ノードは、必然的に受信が遅れてしまう。つまり、分岐点において、先の長くなるであろう受信ノードへはできるだけ早く送信されるべきである。

そこで、送信順を考慮に入れたアルゴリズム (以下、アルゴリズム 1 とする) を提案する。これは、FEF や ECEF によって一旦作成されたマルチキャスト木の分岐点の各辺をソートしつなぎ変えることで送信順を適当なものにする。マルチキャスト木の各分岐点は、それらの各辺の末端の完全な完了時間を把握しなくてはならないため、このアルゴリズムは一旦作成されたマルチキャスト木に修正を加えるといった手法をとる。

アルゴリズム 1 作成後の木において、最も深い位置にある分岐点から順に、以下を繰り返す。

子ノード $C_1, C_2, C_3, \cdots$ について $T_i = Available(C_i) - O_{send}(Parent) * (i-1)$ の値について降順にソートし、つなぎ変える。このとき、$Available(C_i)$ は、各子ノードの送信可能時間を示し、$O_{send}(Parent)$ は親ノードの送信オーバヘッドを示す。ソート完了後、親ノードはその子ノードの中で最大の送信可能時間を保持する。

6 シュミレーション結果

シュミレーションにおける入力情報は、送信オーバーヘッド、通信帯域、メッセージサイズ、総ノード数、目的ノード数である。送受信オーバーヘッドについて、メッセージサイズに依存しない部分 S_c, R_c は $100\mu sec$ から $500\mu sec$、メッセージサイズに依存する部分 S_m, R_m は $0.0001\mu sec/byte$ から $0.01\mu sec/byte$ の範囲でランダムに生成する。また、ルートノードと目的ノードはランダムに決定した。

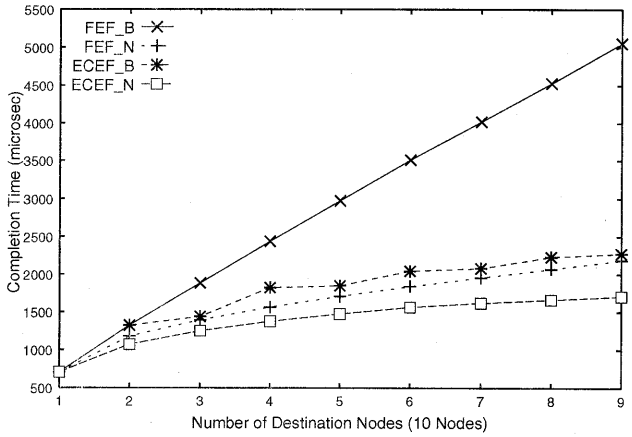

図 3: ノンブロッキングのマルチキャスト完了時間

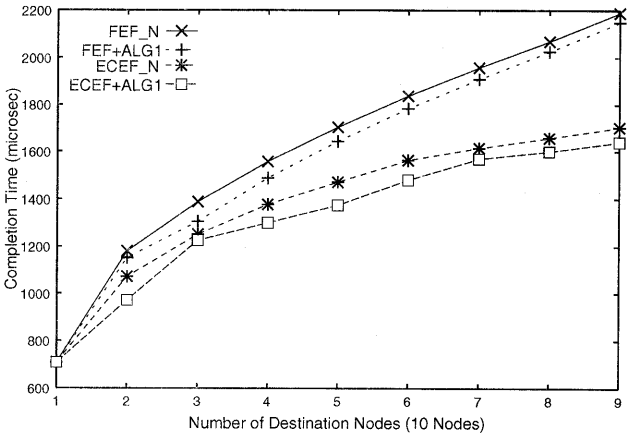

図 4: アルゴリズム 1 の使用時のマルチキャスト完了時間

図 3, 4 は通信帯域を $10MBps$、メッセージサイズを $1kbyte$ とし、マルチキャストの完了時間はそれぞれの入力情報において各アルゴリズムを 1000 回実行し、その平均を採ったものである。図 3 は、FEF と ECEF のブロッキングモデルとノンブロッキングモデルのマルチキャスト完了時間を示しており、FEF と ECEF の両方に関して、ノード数が多くなるほど、その減少の度合が大きくなっている。これより、ノード数が増加するほど、ノンブロッキングモデルの重要性が増すと考えられる。図 4 は、FEF と ECEF の完了時間と、それらにアルゴリズム 1 を使用した場合のマルチキャスト完了時間を示しており、FEF と ECEF ともにノード数の増減に関わらず、ある値 (約 $50\mu sec$ から 100 μsec) だけ減少している。これより、大幅な減少は望めないが、アルゴリズム 1 を使用したほうがマルチキャスト完了時間を減らせることが分かる。

参考文献

[1] M.Banikazemi, et al., "Communication Modeling of Heterogeneous Networks of Workstations for Performance Characterization of Collective Operations," Proc. of the Heterogeneous Computing Workshops, 1999.

[2] M.Banikazemi, et al., "Effcient Collective Communication on Heterogeneous Networks of Workstations," Proc. of Int'l Conf. on Parallel Processing, 1998.

[3] P.B.Bhat, et al., "Efficient Collective Communication in Distributed Heterogeneous Systems," Proc. of Int'l Conf. on Distributed Computing Systems, 1999.