

## 大規模放送網のための送信順序保存放送通信手順

高村 昌興, 中村 幸人, 滝沢 誠

東京電機大学

e-mail {taka, naka, taki}@takilab.k.dendai.ac.jp

グループウェアシステムのような分散型応用では、複数エンティティ間の高信頼放送通信が要求されている。これまでに研究されてきた放送通信プロトコルでは、グループ内の  $n$  個のエンティティに対して、 $O(n)$  から  $O(n^2)$  の通信負荷がある。小規模なシステムでは有効であるが、エンティティを数百程度含んだ大規模なシステムでは、性能面で有効でない。また、通信網の高速化により、各通信エンティティでの処理負荷が問題になり、これを減少させる必要がある。本論文では、大規模なシステムで放送通信を行なわせるために、同一通信チャンネルに接続された複数のエンティティを、複数のグループに分割することにより、各エンティティでの処理負荷を減少させる手続きと、その評価について論じる。

### 1 はじめに

グループウェア [9] のような分散型応用では、OSI [18] といった従来の 1 対 1 通信プロトコルに加えて、複数エンティティ間のグループ通信プロトコル [1, 2, 4, 7, 8, 10, 11, 12, 13, 15, 16, 17] が要求されている。ここで、エンティティのグループを群とする。これらのプロトコルは、エンティティ数が数十程度までの群を対象としており、大規模群では、各エンティティの処理負荷が大きくなり、実用的でない。ここでは、百以上のエンティティの群に対する放送通信プロトコルについて論じる。

放送通信プロトコル [11, 12, 17] で、エンティティ数  $n$  に対して通信される PDU 数と処理負荷は、 $O(n)$  から  $O(n^2)$  になる。超高速網 [3] において、各エンティティの処理速度は、通信速度よりも遅いため、各エンティティの処理負荷を一定値以下に押えることが重要である。このため、本論文では、同一通信チャンネルに接続された複数のエンティティを、複数のグループに分割することにより、処理負荷の減少を試みる。[11, 12, 13, 15, 16, 17] では、群内での送受信の原子性と受信順

序について論じている。これらの中で、送信元の順序を保存する送信順序保存放送通信 (OP) プロトコル (Order-Preserving broadcast communication protocol) [14, 17] が示されている。本論文では、OP を拡張した大規模群送信順序保存放送通信 (LOP) プロトコル (OP protocol for Large cluster) について述べる。

まず、2 章では、OP プロトコルの概要を示す。3 章では、群を複数のグループに分割した複合群のモデルを述べる。4 章では、LOP プロトコルについて述べる。5 章では、LOP プロトコルの処理負荷を評価する。

### 2 OP プロトコル

まず、OP プロトコル [14, 17] の概要を示す。

#### 2.1 サービス

単一チャンネル (1C) サービス [11, 17] は、Ethernet MAC [5] や、無線システムで提供されるサービスを抽象化したものである。ここでは、全エンティティは、同じ順序で全 PDU を受信するが、あるエ

ンティティは、オーバーラン [3] 等により、PDU を受信し損ねる場合がある。OPプロトコルは、ICサービスを利用し、まず、複数エンティティ間に群Cを確立する。群は、二つのエンティティ間のコネクション [6] を、 $n(\geq 2)$  個のエンティティ間に拡張した概念である。これまで、いくつかの高信頼放送通信プロトコル [11, 12, 13, 15, 16, 17] が示されているが、OPプロトコルでは、各PDUは、C内の送信元の送信順序を保存しながら全エンティティに送られる。

[例] 群Cが、エンティティ  $E_1, E_2, E_3$  により提供されるとする。各  $E_j$  は、送受信 PDU の履歴を示す受信ログ  $RL_j$  と送信ログ  $SL_j$  を持つ ( $j = 1, 2, 3$ ) [図 1]。ここで、 $\langle p_1 \dots p_m \rangle$  を、PDU  $p_i$  のログとする ( $i = 1, \dots, m$ )。例えば、 $SL_1$  は、 $E_1$  が a、b、c の順に PDU を送信したことを示す。このとき、 $E_1$  は  $\langle a p x b c y \rangle$ 、 $E_2$  は  $\langle p a b x y c \rangle$ 、 $E_3$  は  $\langle a p b x c y \rangle$  といったように、送信元の送信順序を保存して受信する。各  $RL_{ij}$  は  $RL_i$  の副受信ログで、 $E_j$  から届いた PDU が含まれるものとする。例えば、 $RL_{13} = \langle x y \rangle$  となる。 $RL_{ij}$  は、 $SL_j$  と同じである。これが、OPサービスである。□

$E_1$	$RL_1: \langle a p x b c y \rangle$	$SL_1: \langle a b c \rangle$
$E_2$	$RL_2: \langle p a b x y c \rangle$	$SL_2: \langle p \rangle$
$E_3$	$RL_3: \langle a p b x c y \rangle$	$SL_3: \langle x y \rangle$

図 1: OP サービスの例

## 2.2 三相による正しい受信概念

エンティティ  $E_1, \dots, E_n$  が提供する群Cで、 $E_k$  が放送した PDU  $p$  は、三相手続き [15, 16] により受信される。 $n$  をCの基数とする。

[三相による正しい受信][図 2]

- (1)  $p$  が  $E_j$  で受信される。
  - (2)  $E_j$  は、「全宛先が  $p$  を受信した」ことを知る。
  - (3)  $E_j$  は、「 $p$  の各宛先は、『全宛先が  $p$  を受信した』ことを知っている」ことを知る。□
- (1)、(2)、(3) の条件を満足するとき、各  $\forall p$  は、 $E_j$  で受理、前確認、確認されたとする。これは、複

数エンティティ間での受信の確かさのレベルを示している。(3) がもっとも確かなレベルで、受信の原子性が保証されることになる。

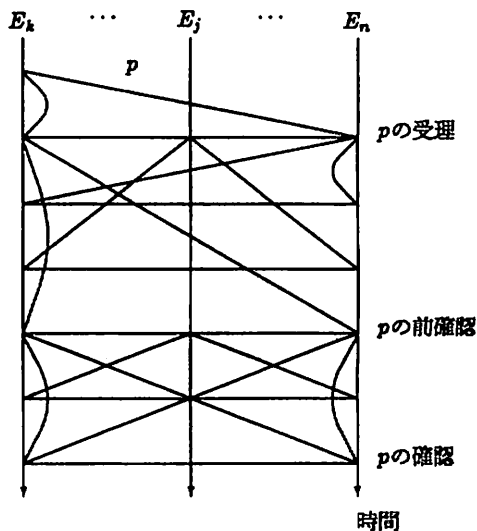


図 2: 三相受信手続き

## 2.3 データ転送手順

各 PDU  $p$  は、図 3 に示す属性から構成される。記法  $p.F$  は、 $p$  の属性  $F$  を示す。 $p.CID$  はCの群識別子、 $p.SRC$  は送信元、 $p.SEQ$  はシーケンス番号である。 $E_k$  が PDU を放送するたびに、 $SEQ$  は 1 つずつ増加する。 $p.ACK_j$  は、 $E_k$  が  $E_j$  から次に受信予定の PDU のシーケンス番号である ( $j = 1, \dots, n$ )。  $p.DATA$  はデータである。

$E_k$  は、各  $E_j$  からの PDU の受信履歴を示すログ  $RL_{kj}$  を持つ。これは、受理、前確認、確認された PDU が記録された副受信ログ  $RRL_{kj}$ 、 $PRL_{kj}$ 、 $ARL_{kj}$  から構成される。 $E_k$  は、送信ログ  $SL_k$  と、送信用の変数  $SEQ_k$  と  $REQ_j$  を持つ。 $E_k$  が、 $p$  を放送するとき、

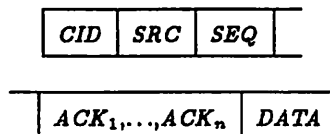


図 3: OP PDU 形式

$p.SEQ := SEQ, p.ACK_j := REQ_j$ とし、 $SEQ := SEQ+1$ とする。

$E_k$ は、各 $E_j$ に対して、受信回数 $REQ_j$ を持つ。 $E_j$ から届いた $p$ が、 $REQ_j = p.SEQ_k$ ならば、 $E_k$ は $p$ を受信し、 $REQ_j := REQ_j+1$ とする。ここで、 $E_j$ が $p$ と $q$ を放送し、 $E_k$ が $p$ を受信し損ねたとする。 $E_k$ が $q$ を受信したとき、 $REQ_j < q.SEQ$ となり、 $p$ の紛失がわかる。また、他の $E_h$ が、 $p$ を受信した後 $r$ を放送するとする。 $E_k$ が $r$ を受信したとき、 $r.ACK_j := p.SEQ+1$ であるので、 $REQ_j < r.ACK_j$ となる。これにより、 $E_k$ は、 $SEQ$ が $REQ_j$ 以上で、 $q.SEQ$ より小さいPDU、即ち $p$ の紛失を検出できる。そこで、 $E_k$ は、 $E_j$ に $p$ の再送要求を行なう。このように、OPプロトコルでは、紛失したPDUだけの選択的再送[17]を行う。

$E_k$ は、 $n \times n$ 行列 $AL$ と $PAL$ を持つ。 $AL$ は、 $E_k$ が「 $E_i$ が、 $AL_{ij}$ 以下の $SEQ$ を持つ全PDUを $E_j$ から受信した」ことを知っていることを示す。 $E_i$ からの $p$ が、 $E_k$ で受信されたとき、 $AL_{ij} := p.ACK_j$ とする( $j = 1, \dots, n$ )。 $E_j$ からのPDUの中で、 $SEQ$ が $\min(AL_{1j}, \dots, AL_{nj})$ 以下のものは、全エンティティで受信された、即ち、前確認されたことが、 $E_k$ はわかる。 $PAL$ は、 $E_k$ が「 $E_i$ が、 $E_i$ からのPDUの中で、 $SEQ$ が、 $PAL_{ij}$ 以下であるものを前確認した」ことを知っていることを示す。 $E_i$ からの $p$ が、 $E_k$ で前確認されたとき、 $PAL_{ij} := p.ACK_j$ とする( $j = 1, \dots, n$ )。 $E_j$ からのPDUの中で、 $SEQ$ が $\min(PAL_{1j}, \dots, PAL_{nj})$ 以下のものは全エンティティで前確認された、即ち、確認されたことが、 $E_k$ はわかる。

ICサービス上のOPプロトコルでは、 $O(n)$ から $O(n^2)$ のPDUが、三相手続きを行うために放送される[13, 14]。PDUヘッダは、 $ACK_j(j = 1, \dots, n)$ を含むので、 $O(n)$ の大きさである[図3]。各エンティティは、 $n \times n$ 行列 $AL$ と $PAL$ を操作するので、 $O(n^2)$ の処理負荷と記憶量が必要となる。

### 3 複合群

従来の群[11, 12, 13, 14, 15, 16]を、平群とする。平群のOPプロトコルでは、処理負荷が $O(n^2)$ であ

ることから、大規模システムでの適用は、性能的に困難である。このために、ICサービス上のエンティティを、複数の要素群というグループに分割する。要素群から構成される群を、複合群とする。

[例] 群サービスを提供する六つのエンティティから成る群 $C$ を考える。平群のOPプロトコルでは、各エンティティからのPDUが全エンティティに放送される。これに対し、 $E_{11}, E_{12}, E_{13}$ の要素群 $C_1$ と、 $E_{21}, E_{22}, E_{23}$ の要素群 $C_2$ に分ける。 $C_1$ と $C_2$ 間でのPDUの交換を行なうエンティティが必要となり、これをゲートウェイ $GE$ とする。ここで、 $E_{11}$ が、PDU  $p$ を放送するとする。 $p$ は、 $E_{11}, E_{12}, E_{13}$ で受信されるとともに、 $GE$ でも受信される。 $GE$ は、 $p$ を $C_2$ 内に放送する。 $GE$ は、 $C_1$ と $C_2$ 内の全エンティティが、 $p$ を受信したことがわかったときに、 $p$ が受信されたことの情報を $ACK$ に含めたPDU  $q_1$ と $q_2$ を各々 $C_1$ と $C_2$ で放送する。 $q_1$ と $q_2$ が受信されたとき、 $C_1$ と $C_2$ 内のエンティティで、 $p$ が前確認される。同様にして、前確認を行なった $q_1$ と $q_2$ が前確認されたとき、 $p$ が確認される。このように、 $C_1$ からみて、 $C_2$ 内での通信は見え、 $GE$ が $C_2$ を代表している。各エンティティでは、基数4の群としての処理負荷を持つことになる。□

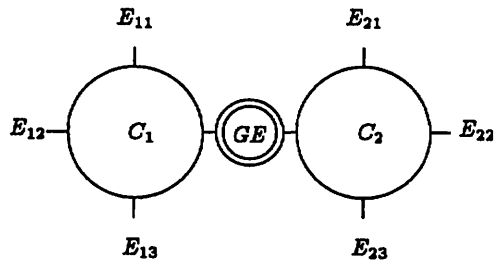


図4: 複合群C

複合群 $C$ は、ゲートウェイ $GE_1, \dots, GE_g$ により接続される要素群 $C_1, \dots, C_h$ から構成される( $C = (C_1, \dots, C_h, GE_1, \dots, GE_g)$ )。全エンティティは、ICサービスを用いるとする。各要素群は、エンティティから構成される。本論文で、複合群 $C$ は、以下の性質を持つとする。

[性質]  $C$ 内の任意の二つの要素群 $C_i$ と $C_j$ 間には、必

ずただ一つのゲートウェイが存在する。□  
 即ち、各エンティティが、他の任意のエンティティに最大一つのゲートウェイを経由することで、PDUを届けられる。 $h = 4$ の複合群としては、図5に示す3つの場合が可能である。

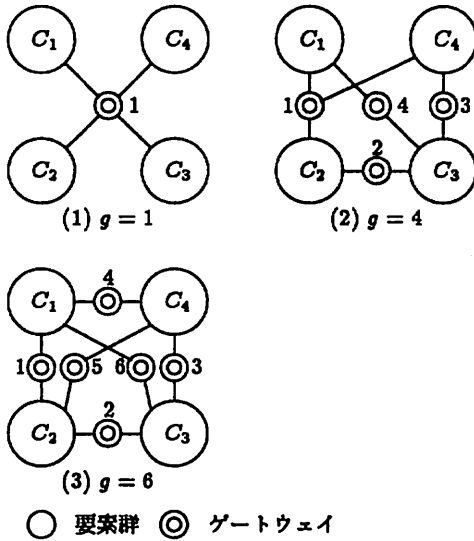


図5: 複合群C

#### 4 複合群でのOPプロトコル

複合群でのOPプロトコル (LOPプロトコル) について論じる。

##### 4.1 PDU形式

複合群  $C = \langle C_1, \dots, C_h, GE_1, \dots, GE_g \rangle$  を考える。要素群  $C_i$  内のエンティティを  $E_{ij}$  とする ( $j = 1, \dots, m_i (\geq 2)$ )。各  $C_i$  は、 $g_i (\leq g)$  個のゲートウェイにより、全ての要素群と接続されているとする ( $C_i = \langle E_{i1}, \dots, E_{im_i}, GE_{i1}, \dots, GE_{ig_i} \rangle, GE_{ij} \in \{GE_1, \dots, GE_g\}$ ) と、 $c_i = m_i + g_i$  個のエンティティから成る。記法  $E_{im_i+j}$  は、 $GE_{ij}$  を示すとする。 $E_{ij}$  の要素番号を  $j$  とする。 $C_i$  内でゲートウェイでないエンティティを、通信エンティティとする。 $E_{ij}$  が放送する各 PDU  $p$  は、以下の属性を持つ。

$p.CID$  PDU が放送されている要素群  $C_i$  の群識別子。

- $p.SRC$  送信元エンティティ ( $E_{ij}$ )。
- $p.LSRC$  PDU を放送した送信元エンティティ。  
 $E_{ij}$  が通信エンティティのときは、 $LSRC = SRC$  である。そうでないとき、 $p$  を最初に送信したエンティティである。
- $p.SEQ$   $E_{ij}$  のシーケンス番号。
- $p.ACK_k$   $C_i$  内の  $E_{ik}$  から、 $E_{ij}$  が次に受信予定の PDU のシーケンス番号。
- $p.DATA$  データ。

##### 4.2 データ転送手順

通信エンティティは、平群のOPプロトコルと同じ動作をするので、 $C$  内の各  $GE_i$  の動作について考える。 $GE_i$  は、要素群  $C_{i1}, \dots, C_{ik}$  を相互接続する ( $i = 1, \dots, g$ )。  $k_i$  を  $GE_i$  の結合度とする。 $GE_i$  は、各  $C_{ij}$  に対して送信処理を行なう副ゲートウェイ  $SGE_{ij}$  から構成される ( $j = 1, \dots, k_i$ ) [図6]。  $SGE_{ij}$  は、 $C_{ij}$  内の通信エンティティが放送した PDU を全  $SGE_{it}$  に送る。また、 $SGE_{ij}$  は、各  $SGE_{it}$  からの PDU を、 $C_{ij}$  内で放送する。 $SGE_{ij}$  の動作について考える。

$SGE_{ij}$  は、 $C_{ij}$  の要素としてOPプロトコルを実行するので、前述したものと同一  $SEQ$ 、 $REQ_s$ 、 $AL$ 、 $PAL$ 、及び、ログ  $SL_i$ 、 $RRL_{is}$ 、 $PRL_{is}$ 、 $ARL_{is}$  を持つ ( $s = 1, \dots, c_i$ )。ここで、 $SGE_{ij}$  の  $C_{ij}$  内での要素番号を  $w_{ij}$  とする ( $E_{ijw_{ij}} = SGE_{ij}$ )。  $minCAL_s := min(AL_{1s}, \dots, AL_{m_s, s})$  とする。 $SGE_{ij}$  が、「 $C_{ij}$  内の全通信エンティティは、 $E_{is}$  が放送した PDU の中で、 $minCAL_s$  よりシーケンス番号の小さい PDU を受信している」ことを知っている。 $SGE_{ij}$  は、 $GE_i$  内で他の  $SGE_{ih}$  との間で通信を行なうために以下のデータ構造を持つ。

- $GSEQ$   $SGE_{ij}$  でのシーケンス番号。
- $GREQ_h$   $SGE_{ih}$  から次に受信予定の PDU シーケンス番号 ( $h = 1, \dots, k_i$ )。
- $GAL_{tu}$   $SGE_{ij}$  が、「 $SGE_{it}$  が  $SGE_{iu}$  から次に受信予定である」ことを知っている PDU シーケンス番号。
- $GSL$  他の副ゲートウェイへの送信ログ。
- $GRL$   $SGE_{i1}, \dots, SGE_{ik_i}$  からの受信ログ。

$minGAL_i := min(GAL_{i1}, \dots, GAL_{ik_i})$  である。□

グ  $L$  と PDU  $p$  に対して、 $enqueue(L, p)$  は、 $L$  の最後尾に  $p$  を記録することを示す。また、 $top(L)$  は、 $L$  内の先頭の PDU を示し、 $p := deque(L)$  は、 $top(L)$  を  $L$  から除き、 $p$  とすることを示す。

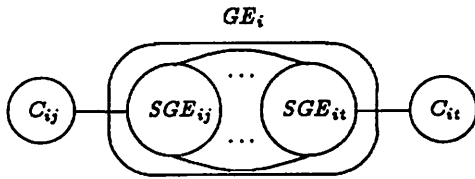


図 6: 副ゲートウェイ

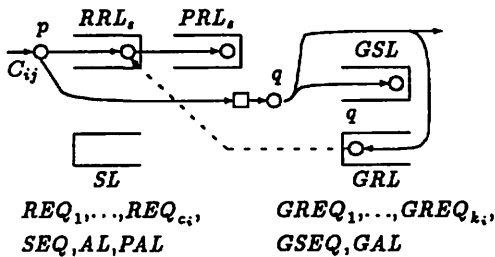


図 7:  $SGE_{ij}$  の動作

はじめに、 $SGE_{ij}$  が、 $C_{ij}$  内の  $E_{ij_s}$  により送信された PDU  $p$  を受信した場合の手順を考える。

[ $C_{ij}$  から  $p$  の受信]

- (1) OP プロトコルに従って、 $p$  の受信処理を行なう。ここで、 $p$  は受理されて、 $enqueue(RRL_s, p)$ 。
- (2)  $p.SRC \neq p.LSRC$  ならば、終了する。 $E_{ij_s}$  は、ゲートウェイであり、 $p$  は  $C_{ij}$  以外の要素群で送信されたものであるため、他の群に放送しない。
- (3)  $p.SRC = p.LSRC$  (即ち、 $E_{ij_s}$  は通信エンティティ) ならば、以下の手順により、 $p$  を全  $SEQ_{it}$  に送信する ( $t = 1, \dots, k_i$ )。

(3-1) 以下の PDU  $q$  を作り、 $enqueue(GRL, q)$ 。

$q.SRC := SGE_{ij}; q.LSRC := p.SRC;$   
 $q.SEQ := GSEQ; GSEQ := GSEQ + 1;$   
 $q.PTR := p; q.DATA := p.DATA;$

(3-2)  $r := top(GSL)$  (ここで、 $r.SRC = E_{ij_s}$  で  $p = r.PTR$  とする);

if ( $r.SEQ < minCAL_s$ ) {

$r := deque(GSL);$

$r.ACK_t := GREQ_t (t = 1, \dots, k_i);$

$r$  を全  $SGE_{it}$  に送信し、(3-2) を繰り返す;

}□

$C_{ij}$  から届いた  $p$  は、各  $SGE_{ih} (h = 1, \dots, k_i)$  に送信するための PDU  $q$  に交換される。各  $q$  は、 $GE_i$  内で通信するため、 $SGE_{ij}$  のシーケンス番号を持ち、これにより、受理、前確認が行なわれる。(2) により、各 PDU が、要素群間で冗長に送信されることを防いでいる。

次に、 $SGE_{ij}$  が、 $SGE_{it}$  より、PDU  $q$  を受信した場合の手順を以下に示す。

[ $SGE_{it}$  より  $q$  の受信]

(1)  $q.SEQ \neq GREQ_t$  ならば、 $SGE_{it}$  に、シーケンス番号  $GREQ_t$  からの PDU の再送を要求する。

(2)  $q.SEQ = GREQ_t$  ならば、以下を行なう。

(2-1)  $GAL_{tr} := q.ACK_r (r = 1, \dots, k_i);$

$enqueue(GRL, q); GREQ_t := GREQ_t + 1;$

(2-2)  $p := top(GRL)$  (ここで、 $p.SRC = SGE_{iu}$ );

while ( $p.SEQ < minGAL_u$ ) {

$p := deque(GRL);$

if ( $h = j$ ) {

$r := p.PTR; r.MARK := ACCEPT;$

}

$p := top(GRL);$

}

(2-3) for ( $l = 1, \dots, m_i$ ) {

$p := top(RL_l)$  (ここで、 $p.SRC = E_{ij_s}$ );

while ( $p.MARK = ACCEPT$ ) {

/\*  $p$  の前確認処理を行なう。\*/

$p := deque(RL_l);$

$PAL_{su} := p.ACK_u (u = 1, \dots, k_i);$

$enqueue(PRL_s, p); REQ_s := p.SEQ + 1;$

}

}□

$E_{ij_s}$  より  $p$  が  $SGE_{ij}$  に届いても、 $REQ_s$  は変更されない。従って、 $p$  の受信後、 $SGE_{ij}$  が  $q$  を送信しても、 $q$  は  $p$  の受信確認を運ばない。 $p$  が、 $GE_{ij}$  と接続する全要素群の通信エンティティで受信されると、即ち、(2-2) で  $p$  が前確認されると、 $REQ_s := p.SEQ + 1$  となる。この後に、 $SGE_{ij}$  より送信される PDU が、 $p$  の受信確認を運ぶ。

[例] 図4で、 $C_1$ 内の $E_{11}$ が、 $p$ を放送したとする。 $GE$ 内の $SGE_1(=E_{14})$ は、 $p$ を受信して $SGE_2(=E_{24})$ に渡す。 $SGE_2$ は、 $p$ から $q$ を作り、 $q$ を $C_2$ 内に放送する。 $SGE_1$ は、 $C_1$ 内の $E_{11}$ 、 $E_{12}$ 、 $E_{13}$ から、 $p$ についての受理情報を含んだPDUを受信したならば、 $SGE_2$ にこれを放送する。また、 $SGE_1$ は、 $GE$ 内の全副ゲートウェイ、即ち、 $SGE_2$ から、 $p$ についての受理情報を受けたとき、 $p$ の受理情報( $r.ACK_1 := p.SEQ+1(=REQ_1)$ )を含んだPDU  $r$ を $C_1$ 内に放送する。[図8] □

PDUの紛失は、OPプロトコルと同様に、タイムアウトにより検出され、再送が要求される。通信エンティティでは、受信したPDUが前確認されないと、タイムアウトにより、再送の要求がなされる。LOPプロトコルでは、ゲートウェイを通して、他の要素群へ放送されるので、OPプロトコルの場合よりも、遅延時間は二倍となり、タイムアウト間隔を二倍以上大きくとる必要がある。

## 5 評価

複合群 $C$ 内の全通信エンティティ数を $n$ 、全ゲートウェイ数を $g$ とする。各要素群 $C_i$ には、 $m_i$ 個の通信エンティティと $g_i$ 個のゲートウェイを含む( $i = 1, \dots, h$ )。  $C_i$ の全エンティティ数 $c_i$ は、 $m_i + g_i$ である。このとき、 $C_i$ の各通信エンティティ $E_{ij}$ の処理負荷は、 $O(c_i^2)$ なので $c_i^2 = (m_i + g_i)^2$ とする。従って、全通信エンティティの処理負荷は、 $CEP = \sum_{i=1}^h m_i(m_i + g_i)^2$ となる。また、各ゲートウェイ $GE_j$ は、 $k_j$ 個の要素群と相互接続している( $j = 1, \dots, g$ )。そのため、 $C_j$ に接続している $SGE_{ij}$ は、 $c_{ij}^2 = (m_j + g_j)^2$ の処理負荷を持つ。また、 $k_j$ 個の要素群と相互接続していることより、 $k_j$ 個の副ゲートウェイ間の通信のために $k_j^2$ の処理負荷を持つ。これより、全ゲートウェイの処理負荷は、 $GEP = \sum_{i=1}^g \sum_{j=1}^{k_i} (c_{ij}^2 + k_j^2) = \sum_{i=1}^g g_i(m_i + g_i)^2 + \sum_{i=1}^g k_i^2$ となる。 $C$ での全処理負荷 $TEP$ は、 $CEP + GEP = \sum_{i=1}^h m_i(m_i + g_i)^2 +$

	$CEP/n^3$	$GEP/gn^2$
(1) $g=1$	0.067600	0.2720
(2) $g=4$	0.074275	0.1687
(3) $g=6$	0.078400	0.1572

表 1: 複合群の処理負荷

$$\sum_{i=1}^g \sum_{j=1}^{k_i} (c_{ij}^2 + k_j^2) = \sum_{i=1}^h (m_i + g_i)^3 + \sum_{i=1}^g k_i^2 \text{ である。}$$

ここで、 $n = 100$ 、 $h = 4$ で、各 $C_j$ について

$m_j = m = 100/4 = 25$ とする例を考える。図5に示す複合群が可能である。このとき、平群における各エンティティの処理負荷 $n^2$ に対する各通信エンティティとゲートウェイの処理負荷の比率を表1に示す。図からわかるように、複合群では処理負荷を減少できる。

図9に、 $n = 100$ で各要素群間に1つのゲートウェイがある場合に、 $h$ に対する以下の値を、平群での処理負荷に対する割合で示す。

- (1)  $CEP/n^3$  = 各エンティティの処理負荷
- (2)  $GEP/(gn^2)$  = 各ゲートウェイの処理負荷
- (3)  $GEP/n^3$  = ゲートウェイ全体の処理負荷
- (4)  $TEP/n^3$  = 各エンティティの処理負荷

図からわかるように、各要素群は基数をおなじとするときには、 $h = 10$ のときに、各通信エンティティとゲートウェイの処理負荷を最小とできる。従って、10個の基数10の要素群に分割することが望ましい。どのように要素群を接続するかにより、(1)~(4)の値は変わってくる。与えられたエンティティ数に対して、最適な複合群の構成は、今後の課題である。

## 6 おわりに

本論文では、大規模なシステムの複数エンティティ間での高信頼放送通信を有効に行なうための手順について述べた。本プロトコルでは、ICサービス網に接続されたエンティティを、複数の要素群に分割することで、各エンティティの処理負荷を減少できる。この最良の複合群の構成方法は、今後の課題である。

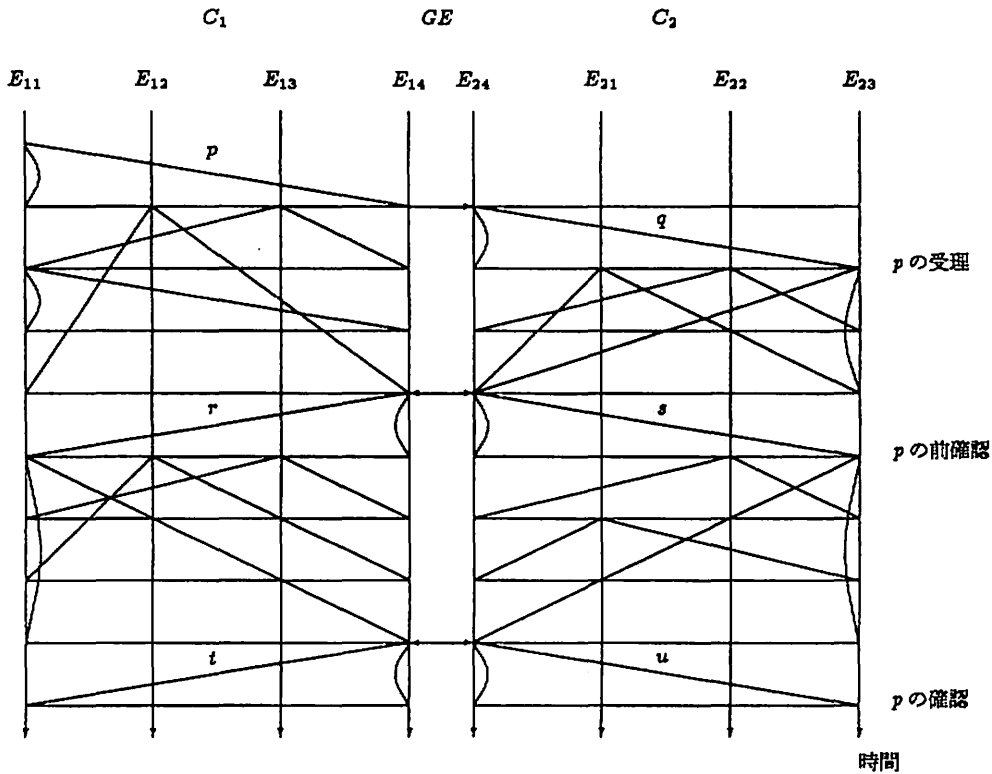


図 8: 要素群 $C_1$ と $C_2$ 間での三相手続き

### 参考文献

- [1] Birman, K., Schiper, A., and Stephenson, P.: *Lightweight Causal and Atomic Group Multicast*, *ACM Trans. on Computer Systems*, Vol.9, No.3, pp.272-314 (1991).
- [2] Chang, J. M. and Maxemchuk, N. F.: *Reliable Broadcast Protocols*, *ACM Trans. on Computer Systems*, Vol.2, No.3, 1984, pp.251-273 (1984).
- [3] Doeringer, W. A., Dykeman, D., Kaiser-swerth, M., Meister, B. W., Rudin, H., and Williamson, R.: *A Survey of Light-Weight Transport Protocols for High-Speed Networks*, *IEEE Trans. on Communications*, Vol.38, No.11, pp.2025-2039 (1990)
- [4] Garcia-Molina, H. and Spauster, A.: *Message Ordering in a Multicast Environment*, *Proc. of the 9th IEEE Int'l. Conf. on Distributed Computing Systems*, pp.354-361 (1989).
- [5] IEEE: *IEEE Project 802 Local Network Standards-Draft* (1982).
- [6] International Standards Organization: *OSI - Connection Oriented Transport Protocol Specification*, *ISO 8079* (1986).
- [7] Kaashoek, M. F., Tanenbaum, A. S., Hummel, S. F., and Bal, H. E.: *An Efficient Reliable Broadcast Protocol*, *ACM Operating Systems Review*, Vol.23, No.4, pp.5-19 (1989).
- [8] Luan, S. W. and Gligor, V. D.: *A Fault-Tolerant Protocol for Atomic Broadcast*,

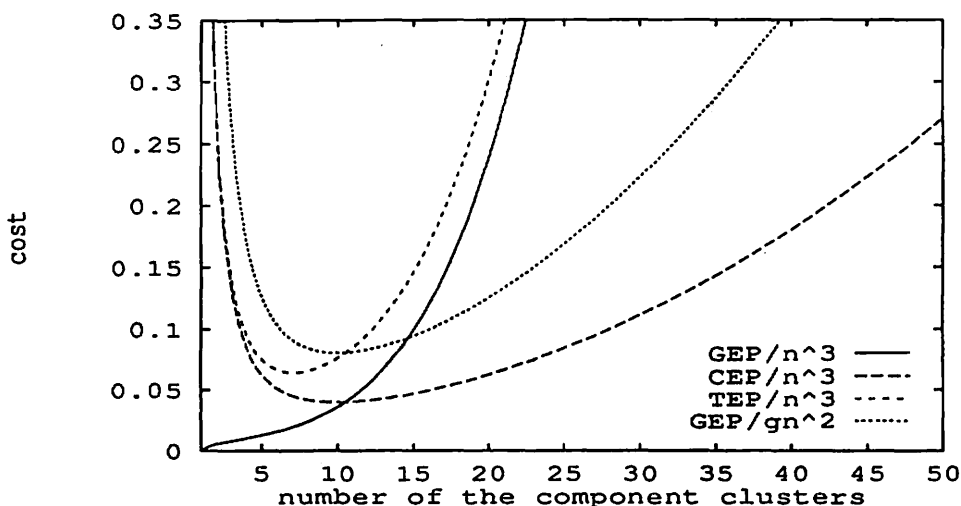


図9: 処理負荷 ( $n = 100, g = h * (h - 1) / 2$ )

*IEEE Trans. on Parallel and Distributed Systems*, Vol.1, No.3, pp.271-285 (1990).

- [9] 松下 温: 図解 グループウェア入門, オーム社 (1991).
- [10] Melliar-Smith, P. M., Moser, L. E., and Agrawal, V.: *Broadcast Protocols for Distributed Systems*, *IEEE Trans. on Parallel and Distributed Systems*, Vol.1, No.1, pp.17-25 (1990).
- [11] Nakamura, A. and Takizawa, M.: *Reliable Broadcast Protocol for Selectively Ordering PDUs*, *Proc. of the 11th IEEE Int'l Conf. on Distributed Computing Systems (ICDCS-11)*, pp.239-246 (1991).
- [12] Nakamura, A. and Takizawa, M.: *Design of Reliable Broadcast Communication Protocol for Selectively Partially Ordering PDUs*, *Proc. of the IEEE COMPSAC'91*, pp.673-679 (1991).
- [13] Nakamura, A. and Takizawa, M.: *Priority-Based Total and Semi-Total Ordering Broadcast Protocols*, *Proc. of the 12th IEEE Int'l Conf. on Distributed Computing Systems (ICDCS-12)*, pp.178-185 (1992).
- [14] 中村 章人, 滝沢 誠: 多チャネルシステム上の送信順序保存放送通信プロトコル, 情報処理学会論文誌, Vol.34, No.1, pp.135-143 (1993).
- [15] Takizawa, M.: *Cluster Control Protocol for Highly Reliable Broadcast Communication*, *Proc. of the IFIP Conf. on Distributed Processing*, pp.431-445 (1987).
- [16] Takizawa, M.: *Design of Highly Reliable Broadcast Communication Protocol*, *Proc. of IEEE COMPSAC87*, pp.731-740 (1987).
- [17] Takizawa, M. and Nakamura, A.: *Partially Ordering Broadcast (PO) Protocol*, *Proc. of the 9th IEEE Conf. on Computer Communications (INFOCOM)*, pp.357-364 (1990).
- [18] Tanenbaum, A. S.: *Computer Networks*, Prentice-Hall Int'l, Inc (1989).