解　説

フォールトトレラント分散システム向けアルゴリズム

2. 放送型通信アルゴリズム†

滝沢　誠††　中村章人††

1. はじめに

グループウェアなどの新しい分散型応用が，今後の重要な計算機応用となるにつれて，新しい通信アルゴリズムが求められてきている．従来の通信網では，OSI TP 4[9] や TCP/IP[5] などのプロトコルにより，二つの通信エンティティ間での高信頼な通信が提供されている．エンティティとは，通信し合う実体であり，たとえばプロセスがこれにあたる．これに対して，新しい分散型システムでは，従来の一対のエンティティ間の通信に加えて，複数のエンティティからなるグループとしての通信が必要となってきている．このような通信は，放送型通信と呼ばれている．放送型通信では，あるエンティティから送信されたメッセージは，全宛先に届けられる．ここで，宛先の一つでも受信に失敗した場合は，全宛先でメッセージを受信しない．これは，受信の原子性と呼ばれている．また，複数のエンティティがメッセージを送信しているとき，各エンティティで，どのような順序でメッセージを受信するかという問題がある．本稿では，放送型通信の論理的な性質を，受信の原子性と受信順序について整理する．また，メッセージの紛失などの障害があるもとで，こうした種々の放送型通信を提供するためのアルゴリズムについて解説し，その比較を行う．

本稿の構成は，以下のようである．まず，**2.** で，分散型システムのモデルを示す．**3.** では，1対1型通信網を用いた放送型通信アルゴリズムを示す．**4.** では，放送型の通信網を用いたアルゴリズムを示す．**5.** では，各アルゴリズムを，**2.** のモデルに基づいて比較検討する．

2. 分散型システムのモデル

本章では，OSI 参照モデルに基づいて，分散型システムのモデルを示す．

2.1 階層モデル

分散型システムを，通信網層，システム層，応用層の3階層により考える（**図-1**）．応用層の $n(\geq 2)$ 個の応用エンティティ $A_1, \cdots, A_n$ は，システム層のサービスアクセス点 (SAP) $S_1, \cdots, S_n$ が提供する通信サービス（システムサービス）を用いて，相互にメッセージの交換を行う．システムサービスは，システムエンティティ $E_1, \cdots, E_n$ の協調動作により実現され，各 S_i は，E_i によって提供される．同一層内のエンティティ間の通信単位（メッセージ）を，プロトコルデータ単位 (PDU) とする．各 E_i は，通信網 SAP N_i を通じて通信網サービスを用いて PDU の送受信を行う．本稿では，通信網層が提供する通信サービスを用いて，応用エンティティに放送型の通信サービスを提供する問題について考える．システムエンティティ間の相互動作を記述するプロトコル（アルゴリズム）の複雑さは，システムサービスの種類と，通信網サービスの種類に依存する．

通信システムでは，コネクション[8] の概念が重要である．コネクションは，二つの SAP 間で信

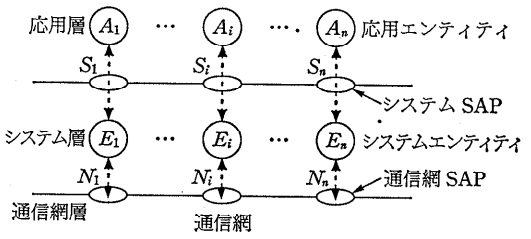


図-1 分散型システムのモデル

† Broadcast Communication Algorithms by Makoto TAKIZAWA and Akihito NAKAMURA (Department of Computers and Systems Engineering, Faculty of Science and Engineering, Tokyo Denki University).

†† 東京電機大学理工学部経営工学科

頼性のある効率的な，送信順序を保存した受信を行わせる論理的な通信路である．これを $n(\geq 2)$ 個に拡張した概念をグループ（または群[21),22)]）とする．グループ G は，n 個の SAP の組 $S_1, \cdots, S_n$ である．各 S_i は，E_i により提供される $(i=1, \cdots, n)$．G は，$E_1, \cdots, E_n$ により提供され，$G=\langle E_1, \cdots, E_n \rangle$ と書くことにする．G が提供するサービスの性質については，次節で論じる．

2.2 サービスモデル

各層のエンティティ T_i が下位層の SAP P_i を通じて利用する通信サービスをモデル化する．ある SAP で送信された PDU が，高々一つの SAP でのみ受信される通信を1対1型とし，複数の SAP で受信されるものを放送型とする．複数の SAP に対する通信サービスを，ログの集合としてモデル化する．ログとは，送受信された PDU の系列である．

各 T_i の動作は，送受信イベントの系列として示せる．T_i による P_i での PDU p の送信を $send_i[p]$，p の受信を $recv_i[p]$ により示す．送受信イベント間の順序関係 $(\rightarrow)$[12)] を定義する．

[イベント間順序関係] 各イベント e_1 と e_2 について，以下の場合に $e_1 \rightarrow e_2$ である．

（1） ある T_i 内で，e_1 が e_2 よりも先に起きる．

（2） ある T_i と T_j について，$e_1=send_i[p]$ で $e_2=recv_j[p]$ である p が存在する．

（3） あるイベント e_3 について，$e_1 \rightarrow e_3$ かつ $e_3 \rightarrow e_2$ である．□

ログ L は，PDU の系列 $<p_1 \cdots p_m]$ である．p_1 は L の先頭，p_m は L の最後の PDU とする．L 内で p_i が p_j に先行する $(i<j)$ ならば，$p_i \rightarrow_L p_j$ と書く．各 SAP P_i に対して，送信ログ SL_i と受信ログ RL_i を定義する．$send_i[p] \rightarrow send_i[q]$ ならば $p \rightarrow_{SL_i} q$，$recv_i[p] \rightarrow recv_i[q]$ ならば $p \rightarrow_{RL_i} q$ である．つまり，SL_i と RL_i はおのおの，P_i での T_i の送信と受信の履歴である（**図-2**）．

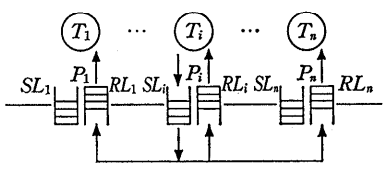


図-2 サービスモデル

送受信された PDU の関係として，以下がある．

● **情報保存**：RL_i が $SL_1, \cdots, SL_n$ 内のすべての PDU を含むならば，RL_i は情報保存する．

● **順序保存**：p と q を RL_i 内の任意の PDU とする．$p, q \in SL_j$ であり，$p \rightarrow_{SL_j} q$ ならば $p \rightarrow_{RL_i} q$ であるとき，RL_i は順序保存する．

● **因果関係保存**：RL_i 内の任意の PDU p と q について，$send_j[p] \rightarrow send_k[q]$ ならば $p \rightarrow_{RL_i} q$ であるとき，RL_i は因果関係保存する．

RL_i が情報保存するとき，T_i は，全 SAP で送信された全 PDU を P_i で受信する．RL_i が順序保存するとき，T_i は，各 SAP で送信された PDU を送信順に受信する．E_j が p を E_i と E_k に送信し，E_k は，p の受信後に q を E_i に送信したとする．このとき，E_i は，p の受信後に q を受信するならば，RL_i は因果関係保存する[2)]．

次に，複数のエンティティで受信された PDU の関係について考える．

● **情報同値**：RL_i と RL_j がともに同じ PDU の集合を含むならば，RL_i と RL_j は情報同値である．

● **順序同値**：RL_i と RL_j に含まれるすべての PDU p と q について，$p \rightarrow_{RL_i} q$ ならば $p \rightarrow_{RL_j} q$ であるとき，RL_i と RL_j は順序同値である．

RL_i と RL_j が情報同値のとき，T_i と T_j は同じ PDU を受信する．順序同値の場合，同じ順序で受信する．

グループ通信とは，n 個の SAP $P_1, \cdots, P_n$ 間のグループ G を確立した後に，G 内で PDU を送受信する通信である．各 PDU は，G 内の全 SAP に送信される．これに対して，選択的グループ通信[15),16)]では，各 PDU が G 内の一部を宛先として，その宛先に送信される．SL_{ij} を，SL_i 内で E_j を宛先にもつ PDU の部分系列とする．

● **選択的情報保存**：RL_i が，$SL_{1i}, \cdots, SL_{ni}$ 内のすべての PDU を含むならば，RL_i は選択的情報保存する．

2.3 放送型通信サービス

前節のサービスモデルに基づいて，通信網サービスとシステムサービスを定義する．システム層は，通信網サービスを用いて，以下に示す高信頼な放送型のシステムサービスを応用層に提供する．

● **送信順序保存（OP）サービス**：各システム

RL_1: <$axbcpydzq$] SL_1: <$abcd$]
RL_2: <$pxabyczdq$] SL_2: <pq]
RL_3: <$axpybqczd$] SL_3: <xyz]

図-3 OP サービスの例

RL_1: <$axbcpydzq$] SL_1: <$abcd$]
RL_2: <$axbcpydzq$] SL_2: <pq]
RL_3: <$axbcpydzq$] SL_3: <xyz]

図-4 TO サービスの例

SAP S_i の受信ログ RL_i が，情報保存でかつ順序保存である．

● **因果関係保存 (CO) サービス**：各 RL_i が，情報保存でかつ因果関係保存である．

● **全順序 (TO) サービス**：各 RL_i が，順序保存で情報保存，かつ互いに順序同値である．

OP サービスでは，各エンティティは，おのおののエンティティから，全 PDU を送信順に受信できる．たとえば，**図-3** で，E_2 は，E_1 が送信した PDU a, b, c をこの順序で受信する．TO サービスでは，各エンティティは，全 PDU を同一の順序で受信する (**図-4**)．

次に，通信網サービスを定義する．

● **単一チャネル (1C) サービス**：各通信網 SAP N_i の受信ログ RL_i が，順序保存でかつ互いに順序同値である．

● **多チャネル (MC) サービス**：各 RL_i が，順序保存である．

1C サービスは，ローカルエリア網や無線網などにより提供されるサービスを抽象化したものである．通信網の高信頼化と超高速化[1]により，伝送速度が各エンティティの処理速度よりも速くなり，バッファのオーバラン，オーバフローにより，送信された PDU を受信できないことがある．1C サービスでは，全エンティティは同じ順序で PDU を受信するが，ある PDU を受信できない場合もある．たとえば，図-4 で，E_2 が c を紛失した場合，RL_2=<$axbpydzq$] となる．MC サービスの例として，ワークステーション間を複数の Ethernet などの 1C サービス網で接続したシステムがある．ワークステーション間に 1 対 1 のコネクションが確立されたシステムもこの例である．たとえば，図-3 で，E_2 と E_3 がそれぞれ c と q を紛失した場合，RL_2=<$pxabyzdq$], RL_3=<$axpybczd$] となる．E_2 は x の後に a を受信し，E_3 は a の後に x を受信しているように，異なるエンティティからの PDU を同一の順序で受信するとは限らない．

2.4 選択的放送型通信サービス

分散型システムでは，複数の応用エンティティがグループ G を構成して処理を行うが，各応用エンティティはデータを必ずしも G 内の全エンティティに送信する必要はなく，その一部に送信する．各 PDU p は，宛先 $\{A_{d_1}, \cdots, A_{d_m}\} \subseteq G$ をもつ．これを $p_{\{d_1,\ldots,d_m\}}$ により示す．たとえば，$a_{\{2,3\}}$ は，a の宛先が A_2 と A_3 であることを示す．

● **選択的送信順序保存 (SP) サービス**：各システム SAP S_i の受信ログ RL_i が，選択的情報保存でかつ順序保存である．

● **選択的全順序 (ST) サービス**：各 RL_i が，互いに順序同値で選択的情報保存である．

SP サービスでは，各エンティティは，自分宛の PDU だけを，紛失なく，送信された順に受信する．たとえば，A_1 は，A_2 から p と q をこの順序で受信している (**図-5**)．ST サービスでは，複数のエンティティで受信された任意の二つの PDU は，それらのエンティティで同じ順序で受信される．たとえば，c と p は，A_1 と A_3 で同じ順序で受信される (**図-6**)．

2.5 原子性

あるエンティティ E_i により送信された PDUp が，全 $E_1, \cdots, E_n$ に届けられたとき，p は原子的に受信されたとする．ここで，エンティティが一つでも p を受信できないとき，どのエンティティも p を受信しない．TO, CO, OP の各サービスでは，情報保存性が保障されるため，すべての PDU について原子性が提供される．しかし，1C および MC サービスでは，受信の原子性が保障されない．

放送型通信アルゴリズムでは，複数のエンティティでの受信の原子性の判断を，どのように実現するかが問題となる．また，その判断をどのエンティティが行うかにより，以下の二つの方式が

RL_1: <$cxpydq$] SL_1: <$a_{\{2,3\}}b_{\{3\}}c_{\{1,3\}}d_{\{1,2\}}$]
RL_2: <$adxyp$] SL_2: <$p_{\{1,2,3\}}q_{\{1\}}$]
RL_3: <$aybcpz$] SL_3: <$x_{\{1,2\}}y_{\{1,2,3\}}z_{\{3\}}$]

図-5 SP サービスの例

RL_1: <$xcpydq$] SL_1: <$a_{\{2,3\}}b_{\{3\}}c_{\{1,3\}}d_{\{1,2\}}$]
RL_2: <$axpyd$] SL_2: <$p_{\{1,2,3\}}q_{\{1\}}$]
RL_3: <$abcpyz$] SL_3: <$x_{\{1,2\}}y_{\{1,2,3\}}z_{\{3\}}$]

図-6 ST サービスの例

ある．

● **集中型制御**：PDU の原子的な受信を，常にある一つの制御エンティティが判断し，他がこれに従う方式である．たとえば，2相コミットメント[7]がこれにあたる．

● **分散型制御**：各 PDU の原子的受信の判断を，異なるエンティティが行う方式である．一般に，PDU の送信元がこの判断を行う．これを弱分散型とする．さらに，この判断を，送信元を含めて各エンティティが自分自身で行う方式を，強分散型方式という．

集中型制御は，アルゴリズムが単純な利点がある．しかし，制御エンティティへの負荷の集中，制御エンティティからのメッセージ待ちによる遅延時間の増加の問題がある．一方，分散型制御では，制御負荷が複数のエンティティに分散される．さらに，下位層に放送型通信サービスを利用した場合には，互いに他のエンティティの送信した PDU を受信できるので，あるエンティティの決定を待たずに自分自身で判断を行える．

強分散型制御では，各エンティティは，PDU の送受信を行いながら，$E_1, \cdots, E_n$ 間で p の原子的受信の決定を行う必要がある．p が原子的に受信されるための条件を示す[21),22)]．

（1） **受理**：E_i は p を受信する．このとき，E_i は p を受理しているという．

（2） **前確認**：E_i は，「全エンティティが p を受信している」ことを知っている．このとき，E_i は p を前確認しているという．

（3） **確認**：E_i は，「全エンティティが p を前確認している」ことを知っている．このとき，E_i は p を確認しているという．

(2)では，E_i は，p を全エンティティが受信したと考えるが，E_j は，p がある E_k で受信されていないと考える．このことは，E_i は p に対する受信確認を全エンティティから受信したが，E_j は E_k から受信できなかった場合に起こる．このために，(3)が必要となる．E_i は，『全エンティティが「全エンティティで p が受信されている」ことを知っている』ことが分かり，E_i は，p が原子的に受信されたと判断できる．

2.6 復 旧

通信網層では，送信された PDU が紛失する場合があり，情報保存性が保障されない．つまり，1C または MC サービスを用いて，高信頼なシステムサービスを提供するには，PDU の紛失検出とその復旧が必要になる．また，PDU の紛失をどのように復旧するかが問題となる．復旧方法として，以下の二つがある．

● **後退復旧**：紛失した PDU 以降に送信されたすべての PDU を再送する．

● **選択的復旧**：紛失した PDU だけを再送する．

後退復旧では，紛失した PDU 以降の送受信を改めて行う．選択的復旧は，後退復旧よりも再送される PDU 数が少ない利点がある．一方で，複数のエンティティ間で PDU の全順序性などの受信順序を保つための同期手順が複雑になる．

2.7 同 期 性

各 E_i は，PDU を他のエンティティに同期的または非同期的に送信する．同期通信では，PDU の原子的な受信が確認されるまで，次の PDU を送信しない．一方，非同期通信では，PDU の原子的な受信が確認される前に，次の PDU を送信できる．

3. 1対1型通信網を用いた放送型通信アルゴリズム

1対1型通信網サービスを用いたアルゴリズムについて述べる．各エンティティがもつローカル変数には大文字，PDU 内の各情報には小文字の記号を用いる．各アルゴリズムで，各 PDU p に含まれる情報として，シーケンス番号 $p.seq$ がある．また，各 E_i は，次に E_j から受信予定の PDU のシーケンス番号を示す変数 REQ_j をもつ $(j=1, \cdots, n)$．

3.1 Garcia-Molina らのアルゴリズム

Garcia-Molina と Spauster によるアルゴリズム[6]（GS と呼ぶ）では，節点をエンティティ，辺を PDU の送信経路とした木構造の伝播グラフが用いられる．通信網上の複数のグループから伝播グラフが構成される．たとえば，4つのグループ $G_1=\langle E_1, E_2, E_3, E_4\rangle$, $G_2=\langle E_1, E_2, E_6\rangle$, $G_3=\langle E_1, E_4, E_5\rangle$, $G_4=\langle E_7, E_8\rangle$ から，**図-7** に示す伝播グラフが生成される．ここで，各 G_i 内の全エンティティは，この中の一つを根節点（$root(G_i)$ と書く）とする部分木内に含まれる．各 PDU は，宛先グループ G_i の $root(G_i)$ に送信され，G_i に属するエンティティを含む部分木へ再帰的に送信

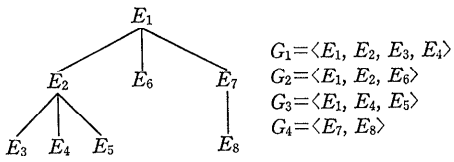


図-7 伝播グラフ

される．本アルゴリズムは，MC サービスを用いて TO サービスを提供する．

各 E_i について，$children_i(G)$ を，G 内のエンティティを子孫としてもつ E_i の子節点の集合とする．各 E_i は，あるエンティティから受信した PDU をその子節点に送信するときに用いるシーケンス番号 SEQ_C と，各グループ G に対してシーケンス番号 SEQ_G をもつ．REQ_P は，次に親節点から受信予定の PDU のシーケンス番号である．PDUp は，シーケンス番号 seq と seq_C をもつ．

（1） 送信元 (E_i):

$p.seq:=SEQ_G$, $SEQ_G:=SEQ_G+1$ とし，$p=\langle E_i, G\rangle$ を $root(G)$ に送信する．

（2） 根節点 (E_j): p を受信したとする．

$p.seq=REQ_i$ ならば，$REQ_i:=REQ_i+1$ とし，S_j を通じて p を A_j に渡す．$p.seq_C:=SEQ_C$, $SEQ_C:=SEQ_C+1$. 各 $E_k\in children_j(G)$ に $p=\langle E_i, G\rangle$ を送信する．

（3） 根節点以外 (E_k): p を親節点 E_h から受信したとする．

$p.seq_C=REQ_P$ ならば，$REQ_P:=REQ_P+1$ とし，S_k を通じて p を A_k に渡す．$p.seq_C:=SEQ_C$, $SEQ_C:=SEQ_C+1$. 各 $E_m\in children_k(G)$ に $p=\langle E_i, G\rangle$ を送信する． □

図-7 の例で，E_8 が G_1 に p を放送する場合を考える．E_8 は，まず $E_1(=root(G_1))$ に p を送信する．E_1 は，E_3 と E_4 を子孫にもつ子節点，すなわち E_2 に送信する．次に，E_2 が，E_3 と E_4 に p を送信する．

E_i は，q の後に p を送信し，E_j は，q の受信に失敗したとする．p を受信したとき，$p.seq>REQ_i$ なので，E_j は q の紛失を検出する．根節点 E_j は，送信元 E_i からの p の受信により q の紛失を検出したならば，つまり $p.seq>REQ_i$ ならば，p をバッファに記憶し，$REQ_i\le q.seq<p.seq$ である q の再送を E_i に要求する．同様に，根節点以外の E_k は，親節点 E_h からの p の受信により q の紛失を検出したならば，つまり $p.seq_C>REQ_P$ ならば，p をバッファに記憶し，$REQ_P\le q.seq_C<p.seq_C$ である q の再送を E_h に要求する．

あるグループの根節点から全メンバに PDU を送信するまでに，グループのメンバ以外の節点を通過し得る点が問題となる．たとえば，図-7 では，G_3 に PDU を送信する場合の E_2 がこれにあたる．伝播グラフの構成時に，このような節点を減少させることが問題となる．また，エンティティの障害時には，伝播グラフを再構成しなければならない．

3.2 Birman らのアルゴリズム

Birman らのアルゴリズム[2] (BSS と呼ぶ) は，各 PDU ごとに，その送信元が 2 相コミットメント[7] に基づいた制御を行う弱分散型である．本アルゴリズムは，ISIS[2] の ABCAST プリミティブとして応用エンティティに提供されている．本アルゴリズムは，高信頼な 1 対 1 型通信網サービス (OP サービス) の利用を仮定しているため，PDU の紛失検出および再送はアルゴリズムに含まれない．本アルゴリズムでは，仮想時計[12]の値を時刻印として PDU に与え，時刻印の順序で PDU を配送することにより TO と CO サービスを提供する．各 E_i の仮想時計の値を VT_i とする．E_i は，待ち行列 Q_i をもつ．

（1） 送信元 E_i は，p を各エンティティに送信する．

（2） 各 E_j は，受信した p に時刻印 $\langle VT_j, j\rangle$ と「配送不可」マークを付与し，待ち行列 Q_j に追加する．$VT_j:=VT_j+1$. $\langle VT_j, j\rangle$ を送信元 E_i に送信する．

（3） E_i は，各 E_j からの p に対する時刻印をすべて受信したならば，その中の最大値 $\langle VT_k, k\rangle$ を選択し，これを各 E_j に送信する．

（4） 各 E_j は，p の時刻印を，E_i から受信した新しい時刻印に変更し，「配送不可」マークを「配送可」に変更する．Q_j 内の PDU を時刻印順に整列し，先頭の PDU が「配送可」マークの付いたものであれば，これを配送する．

$G=\langle E_1, E_2, E_3\rangle$ に対する本アルゴリズムの例を，**図-8** に示す．各 PDUp は，待ち行列内に $\langle p$, 時刻印, マーク$\rangle$ として記憶される．各 E_i

(1) Q_1: ⟨p_3,⟨15,1⟩,不可⟩ ⟨p_1,⟨16,1⟩,不可⟩ ⟨p_2,⟨17,1⟩,不可⟩
Q_2: ⟨p_2,⟨16,2⟩,不可⟩ ⟨p_1,⟨17,2⟩,不可⟩ ⟨p_3,⟨18,2⟩,不可⟩
Q_3: ⟨p_1,⟨17,3⟩,不可⟩ ⟨p_3,⟨18,3⟩,不可⟩ ⟨p_2,⟨19,3⟩,不可⟩
(2) Q_1: ⟨p_3,⟨15,1⟩,不可⟩ ⟨p_2,⟨17,1⟩,不可⟩ ⟨p_1,⟨17,3⟩,可⟩
Q_2: ⟨p_2,⟨16,2⟩,不可⟩ ⟨p_1,⟨17,3⟩,可⟩ ⟨p_3,⟨18,2⟩,不可⟩
Q_3: ⟨p_1,⟨17,3⟩,可⟩ ⟨p_3,⟨18,3⟩,不可⟩ ⟨p_2,⟨19,3⟩,不可⟩
(3) Q_1: ⟨p_3,⟨15,1⟩,不可⟩ ⟨p_1,⟨17,1⟩,可⟩ ⟨p_2,⟨19,3⟩,可⟩
Q_2: ⟨p_1,⟨17,3⟩,可⟩ ⟨p_3,⟨18,2⟩,不可⟩ ⟨p_2,⟨19,3⟩,可⟩
Q_3: ⟨p_3,⟨18,3⟩,不可⟩ ⟨p_2,⟨19,3⟩,可⟩
(4) Q_1: ⟨p_1,⟨17,3⟩,可⟩ ⟨p_3,⟨18,3⟩,可⟩ ⟨p_2,⟨19,3⟩,可⟩
Q_2: ⟨p_3,⟨18,3⟩,可⟩ ⟨p_2,⟨19,3⟩,可⟩
Q_3: ⟨p_3,⟨18,3⟩,可⟩ ⟨p_2,⟨19,3⟩,可⟩

注)「配送可」マークを「可」,「配送不可」マークを「不可」と略す.

図-8 BSS アルゴリズムの実行例

が PDU p_i を送信したとする ($i=1,2,3$). (1)で, 各 E_i は p_1 に対する時刻印を E_1 に送信する. E_1 は, この中の最大値 ⟨17,3⟩ を各 E_i に送信する. (2)では, 各 E_i が p_1 の時刻印を ⟨17,3⟩ に変更し,「配送可」マークを付与する. E_3 では, Q_3 の先頭が p_1 なので, これが配送される. 同様にして, (3) では, p_2 の時刻印が ⟨19,3⟩ に, マークが「配送可」に変更される. E_2 が p_1 を配送する. (4)では, p_3 の時刻印が変更されるとともに, 全エンティティで各 PDU が「配送可」となる. 全エンティティで, p_1, p_3, p_2 の順に PDU が配送される.

4. 放送型通信網を用いた放送型通信アルゴリズム

放送型通信網サービスを用いた場合のアルゴリズムについて考える.

4.1 Kaashoek らのアルゴリズム

Kaashoek らのアルゴリズム[11] (KTHB と呼ぶ) は, Ethernet などの放送型通信網を用いた集中型制御方式のアルゴリズムである. 1C サービスを用いて, TO サービスを提供する. 本アルゴリズムは, 分散オペレーティングシステム Amoeba[28] 用のプロトコルとして用いられている. データ転送は次の2段階で行われる.

(1) 送信元エンティティ E_i は, PDU p を1対1型通信サービスを用いてシーケンサと呼ばれる制御エンティティに送信する.

(2) シーケンサは, p に一意なシーケンス番号を付与し, これを放送する. p を受信した各エンティティは, 紛失がないならば, これを応用エンティティに渡す.

通信網上には, 一時に一つのシーケンサが存在する. 各エンティティは, シーケンサから PDU を同一の順序で受信する. シーケンス番号により PDU の紛失を検出したエンティティは, シーケンサに対して再送要求を行う. シーケンサは, 再送要求を受信したならば, 要求された PDU を再送する. 再送要求と再送 PDU の送信には, 1対1型通信サービスを用いる.

各 PDU は, 原子的受信が確認されるまで, シーケンサのバッファ内に保持されている. シーケンサと各エンティティ間の通信は, 同期的である. 各 E_i は, 最後に受信した PDU のシーケンス番号 (受信確認) *ack* を, 送信する PDU に含める. シーケンサは, 行列 $T=[T_1,\cdots,T_n]$ をもち, E_i から受信した *ack* を T_i に記録する. T 内の最小値を $minT$ とすると, $p.seq \leq minT$ である PDU p は全エンティティで受信されていることが分かり, p を記憶しているバッファを解放する.

本アルゴリズムは, シーケンサによる集中型制御を行うため, 負荷の集中が問題である. また, エンティティの障害は考えない. シーケンス番号が全 PDU で一意であるため, 紛失した PDU だけを選択的に再送できる.

4.2 滝沢と中村のアルゴリズム

滝沢と中村は,強分散型制御により,TO[21)~23),25)] と OP[24)], SP[15),16)] の各サービスを提供する一連のアルゴリズムを示している. 通信網サービスとしては, 1C または MC サービスを用いる. これらのアルゴリズムは, 強分散型制御を実現するために, **2.5** で示した3相の手順からなる. 各エンティティは, PDU を非同期的に送信できる.

A. 全順序アルゴリズム

滝沢と中村のアルゴリズムの中で, 1C サービスを用いて TO サービスを提供するアルゴリズム[21)~23),25)] (TNTO と呼ぶ) を示す. 各 PDU は, シーケンス番号 *seq* と, 各 E_k から受信予定の PDU のシーケンス番号 (受信確認) ack_k ($k=1,\cdots,n$) をもつ. また, 各 E_i は, 次に送信予定の PDU のシーケンス番号を示す変数 SEQ と, 受信した PDU を記憶するための待ち行列 RRL_i と PRL_i, および $n\times n$ 行列 AL と PAL をもつ. 以下に, アルゴリズムを示す.

(1) **送信と受理**: (1-1) E_j は, $p.seq:=SEQ$,

$SEQ:=SEQ+1$, $p.ack_k=REQ_k$ $(k=1,\cdots,n)$ とし，p を放送する．(1-2) p が到着したとき，E_i は，$REQ_j=p.seq$ (ただし，$p.src=E_j$) ならば p を受理し，RRL_i に追加する．このとき，$AL_{kj}:=p.ack_k$ $(k=1,\cdots,n)$，$REQ_j:=REQ_j+1$ とする．

(2) **前確認**：RRL_i の先頭の p について，$p.seq<min\{AL_{jk}|k=1,\cdots,n\}$ ならば，p は前確認される．p を PRL_i に移動し，$PAL_{kj}:=p.ack_k$ $(k=1,\cdots,n)$ とする．

(3) **確認**：PRL_i の先頭の p について，$p.seq<min\{PAL_{jk}|k=1,\cdots,n\}$ ならば，p は確認される．p を A_i に渡す．

PDU の紛失は，シーケンス番号 seq と受信確認 $ack_1,\cdots,ack_n$ により検出される．紛失した PDU 以降に受信した PDU を各 E_i が RRL_i から削除した後，これらの PDU を再送する後退復旧方法が用いられる．紛失した PDU だけを再送するだけでは，順序同値性を保障できない場合がある．たとえば，異なったエンティティから送信された二つの PDU a と b について，各エンティティがいずれか一方しか受信していない場合がこれにあたる．選択的復旧を行うためには，再送された PDU の受信順序についての合意を行うための付加的な同期手順が必要となる．手順の簡単さから，後退復旧が用いられている．

強分散型制御で非同期的通信を行うために，他のエンティティのバッファ量を各 E_i が知る必要がある．PDU に残りバッファ量を含ませて，他に通知する．BUF_j を，E_i が知っている E_j のバッファ量とする．$minBUF$ を，$BUF_1,\cdots,BUF_n$ の最小値とする．このとき，$minBUF/n^2$ をウィンドウ幅として，PDU を非同期的に送信する．

B. 送信順序保存アルゴリズム

文献 19), 24), 25), のアルゴリズム (TNOP と呼ぶ) は，MC サービスを用いて OP サービスを提供する．基本的なアルゴリズムは TNTO と同じであるが，各 E_j から受信した PDU は，E_j 用の受信ログ RL_{ij} に記憶する．PDU の紛失に対しては，選択的復旧を行う．E_j から再送された PDU は，そのシーケンス番号によって，RL_{ij} に挿入できるからである．

C. 選択的送信順序保存アルゴリズム

TNOP アルゴリズムを改良し，MC サービス上で SP サービスを提供するアルゴリズム[15)~17)] (TNSP と呼ぶ) が示されている．TNTO と TNOP では，グループ G 内の全エンティティに PDU が送信されるが，本アルゴリズムでは，各 PDU p が G 内の一部のエンティティを宛先として送信される．p は，seq に加えて，各 E_j に対する副シーケンス番号 $pseq_j$ $(j=1,\cdots,n)$ をもつ．$pseq_j$ は，p が E_j 宛であるとき，1 加算される．各 E_i は，E_j から受信予定の副シーケンス番号を示す変数 $PREQ_j$ をもつ．E_j からの p の受信に失敗しても，p が E_j 宛でないならば，受信する必要がない．E_i が p の紛失を検出しても，ただちに E_j に p の再送を要求せずに，E_j から次の PDU q が到着するのを待ち，q が E_i 宛で $q.pseq_i=PREQ_j$ ならば，p が E_i 宛でなかったことが分かる．このときは，E_i は p の紛失を無視できる．

5. 評 価

各アルゴリズムの性能を，PDU 数と遅延時間により比較する．PDU 数は，ある PDU がグループ G 内の全応用エンティティに配送されるまでに必要な PDU 数である．これらの PDU の転送に必要な時間を遅延時間とする．各エンティティ間の伝送遅延時間を T とする．G 内のエンティティ数を n とする．**表-1** に，各アルゴリズムの比較を示す．1 対 1 型通信網を前提とするアルゴリズムで，放送型通信網も利用できる場合には，放送型通信網を利用した場合の性能も示す．

アルゴリズム GS では，グループ G 宛の PDU は，まず G の根節点に送信され，続いて部分木内の各節点に送信される．1 対 1 型通信網を用いるので，n 個の PDU の送信が必要である．しかし，部分木内に G 以外のエンティティが含まれる可能性があるので，そのエンティティ数を ε とすると，$n+\varepsilon$ 個の PDU の送信が必要である．放送型通信網を用いると，あるエンティティからその子節点の全エンティティへの送信を一度で行える．伝播グラフの高さを h とすると，$h+1$ 個の PDU の送信が必要となる．

アルゴリズム BSS では，送信元から各エンティティへの送信に n 個，n 個の各エンティティから送信元への VT の送信に n 個，送信元から各エンティティへの確認の PDU が n 個で，合計

表-1 各アルゴリズムの比較

アルゴリズム	放送型サービス	通信網サービス	制御方式	宛先	同期性	復旧方法	性能		特徴
							PDU 数	遅延時間	
GS	TO	1対 1/MC	弱分散	グループ	非同期	選択的	1対1網：$n+\varepsilon$ 放送網：$h+1$	$(h+1)T$	木構造
BSS	TO/CO	1対 1/OP	弱分散	グループ	同期	—	1対1網：$3n$ 放送網：$n+2$	$3T$	2相
KTHB	TO	放送/1C	集中	グループ	同期	選択的	2	$2T$	2相
TNTO	TO/CO	放送/1C	強分散	グループ	非同期	後退	$2n+1$	$3T$	3相
TNOP	OP	放送/MC	強分散	グループ	非同期	選択的	$2n+1$	$3T$	3相
TNSP	SP	放送/MC	強分散	選択的	非同期	選択的	$2m+1$	$3T$	3相

$3n$ 個の PDU の送信が必要である．

アルゴリズム KTHB では，シーケンサへの送信に1個と，シーケンサから各エンティティへの送信に放送型通信を用いるので1個，合計2個の PDU の送信が必要である．

アルゴリズム TNTO では，送信元から各エンティティへの送信に1個，前確認のために n 個，確認のために n 個で，合計 $2n+1$ 個の PDU の送信が必要である．PDU に対する受信確認は，次に送信する PDU にピギーバックされるため，一つの PDU ごとに $2n+1$ 個の PDU が送信されるわけではない．TNSP では，各 PDU の宛先数 $m(\leq n)$ により異なる．

6. おわりに

本稿では，放送型通信アルゴリズムの論理的な性質を述べ，これに基づいて代表的なアルゴリズムを整理した．このほかにも，文献 3), 13), 14) などのアルゴリズムがあるが，基本的な性質は，ここで述べたアルゴリズムに基づいている．放送型通信アルゴリズムを利用する応用グループの性質をまとめたものとして，文献4）がある．また，ISO でも，文献 10) により放送型通信プロトコルの標準化が進められている．

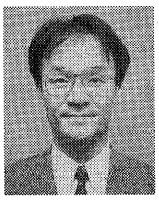
現在，プロトタイプがおのおの実現されてきた段階である．今後，実際の応用のもとでの性能面の評価が重要となってくる．今後の課題として，ST サービスの実現，優先度を考慮した PDU の順序付け[18]，放送型通信における安全保護の問題[26]，超高速網上での放送型通信アルゴリズム[20]，大規模グループでの放送型通信を効率良く行うための放送型通信アルゴリズム[27]などがある．

参考文献

1) Abeysundara, B. W. and Kamal, A. E.: High-Speed Local Area Networks and Their Performance: A Survey, *ACM Computing Surveys*, Vol. 23, No. 2, pp. 221-264 (1991).
2) Birman, K., Schiper, A. and Stephenson, P.: Lightweight Causal and Atomic Group Multicast, *ACM Trans. Computer Systems*, Vol. 9, No. 3, pp. 272-314 (1991).
3) Chang, J. M. and Maxemchuk, N. F.: Reliable Broadcast Protocols, *ACM Trans. Computer Systems*, Vol. 2, No. 3, pp. 251-273 (1984).
4) Liang, L., Chanson, S. T. and Neufeld, G. W.: Process Groups and Group Communications: Classifications and Requirements, *IEEE Computer*, Vol. 23, No. 2, pp. 56-66 (1990).
5) Defense Communications Agency: DDN Protocol Handbook, Vol. 1-3, NIC 50004-50005 (1985).
6) Garcia-Molina, H. and Spauster, A.: Ordered and Reliable Multicast Communication, *ACM Trans. Computer Systems*, Vol. 9, No. 3, pp. 242-271 (1991).
7) Gray, J.: Notes on Database Operating Systems, *Operating Systems: An Advanced Course*, *Lecture Notes in Computer Science*, Springer-Verlag (1978).
8) ISO: OSI—Basic Reference Model, ISO 7498 (1984).
9) ISO: OSI—Connection Oriented Transport Protocol Specification, ISO 8073 (1986).
10) ISO/IEC JTC 1/SC 6 N 7788: Second Working Draft of a Technical Report Type 3—Guidelines for Enhanced Transport Mechanisms (1992).
11) 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).
12) Lamport, L.: Time, Clocks, and the Ordering of Events in a Distributed System, *Comm. ACM*, Vol. 21, No. 7, pp. 558-565 (1978).

13) Luan, S. W. and Gligor, V. D.: A Fault-Tolerant Protocol for Atomic Broadcast, *IEEE Trans. Parallel and Distributed Systems*, Vol. 1, No. 3, pp. 271-285 (1990).
14) Melliar-Smith, P. M., Moser, L. E., and Agrawala, V.: Broadcast Protocols for Distributed Systems, *IEEE Trans. Parallel andDistributed Systems*, Vol. 1, No. 1, pp. 17-25 (1990).
15) Nakamura, A. and Takizawa, M.: Reliable Broadcast Protocol for Selectively Ordering PDUs, *Proc. of the 11th IEEE ICDCS*, pp. 239-246 (1991).
16) Nakamura, A. and Takizawa, M.: Design of Reliable Broadcast Communication Protocol for Selectively Partially Ordered PDUs, *Proc. of the IEEE COMPSAC '91*, pp. 673-679 (1991).
17) 中村章人，滝沢 誠：多チャネル上の選択的放送通信プロトコルのデータ転送手続き，情報処理学会論文誌，Vol. 33, No. 2, pp. 223-233 (Feb. 1992).
18) Nakamura, A. and Takizawa, M.: Priority-Based Total and Semi-Total Ordering Broadcast Protocols, *Proc. of the 12th IEEE ICDCS*, pp. 178-185 (1992).
19) 中村章人，滝沢 誠：多チャネル上の送信順序保存通信プロトコル，情報処理学会論文誌，Vol. 34, No. 1, pp. 135-143 (Jan. 1993).
20) 中村章人，滝沢 誠：グループ通信プロトコルにおけるレート制御方式，情報処理学会マルチメディア通信と分散処理研究会，60-4, pp. 27-34 (1993).
21) Takizawa, M.: Cluster Control Protocol for Highly Reliable Broadcast Communication, *Proc. of the IFIP Conf. on Distributed Processing*, pp. 431-445 (1987).
22) Takizawa, M.: Design of Highly Reliable Broadcast Communication Protocol, *Proc. of IEEE COMPSAC* 87, pp. 731-740 (1987).
23) 滝沢 誠，中村章人：1チャネル上の全順序放送通信プロトコルにおけるデータ転送手続き，情報処理学会論文誌，Vol. 31, No. 4, pp. 609-617 (Apr. 1990).
24) Takizawa, M. and Nakamura, A.: Partially Ordering Broadcast (PO) Protocol, *Proc. of the 9th IEEE INFOCOM*, pp. 357-364 (1990).
25) Takizawa, M. and Nakamura, A.: Reliable Broadcast Communication, *Proc. of IPSJ Int'l Conf. on Information Technology* (*InfoJapan*), pp. 325-332 (1990).
26) Takizawa, M. and Mita, H.: Secure Group Communication Protocol for Distributed Systems, *Proc. of the IEEE COMPSAC 93* (1993).
27) Takizawa, M., Takamura, M. and Nakamura, A.: Group Communication Protocol for Large Group, *Proc. of the 18th IEEE Annual Conf. on Local Computer Networks* (1993).
28) Tanenbaum, A. S., Renesse, R. V., Staveren, H. V., Sharp, G. J., Mullender, S. J., Jansen, J. and Rossum, G. V.: Experiences with the Amoeba Distributed Operating System, *Comm. ACM*, Vol. 33, No. 12, pp. 46-63 (1990).

(平成5年5月31日受付)



滝沢 誠（正会員）

1950年生．1973年東北大学工学部応用物理学科卒業．1975年同大学院工学研究科応用物理学専攻修士課程修了．同年(財)日本情報処理開発協会入社．1986年東京電機大学理工学部経営工学科講師，1987年より同助教授．工学博士．1989年9月より1年間ドイツ国立情報処理研究所 (GMD) 客員教授．1990年7月より Keele 大学（英国）客員教授．分散型データベースシステム，通信網，分散型システム，知識ベースシステム等の研究に従事．著書「知識工学基礎論」(共著，オーム社)，「データベースシステム入門技術解説」(ソフト・リサーチ・センター)，「分散システム入門」(共著，近代科学社)，「OSI プロトコル技術解説」(共著，ソフト・リサーチ・センター)，電子情報通信学会，人工知能学会，ACM，IEEE 各会員．

中村 章人（正会員）

1966年生．1989年東京電機大学理工学部経営工学科卒業．1991年同大学院理工学研究科修士課程修了．現在同大学院理工学研究科博士後期課程在学中．分散型システム，通信プロトコル等に興味をもつ．著書「OSI プロトコル技術解説」(共著，ソフト・リサーチ・センター)．1992年情報処理学会奨励賞受賞．