動的グループ通信プロトコル

1F－2

鈴木　等,　中村　章人,　滝沢　誠
東京電機大学

1　はじめに

グループウェア等の応用では、複数の通信エンティティ間での協調動作が必要とされる。このためには、複数エンティティ間のグループ通信が必要となる。複数エンティティのグループを群[?]とする。群通信のためのプロトコルが[?, ?, ?]等で議論されている。ここでは、群を利用する応用エンティティが送信するプロトコルデータ単位(PDU)を、各エンティティ毎の送信順に受信できる送信順序保存(OP)サービス[?, ?]を考える。本論文では、エンティティの停止、復旧によりインスタンスが変化しても、データ転送を停止せずに、動作中エンティティにより完全分散環境下でOPサービスを提供するプロトコルを示す。

2　システムモデル

本システムは、網、システム、応用の3層から構成される[図??]。網SAP(NSAP) $S_1,\ldots,S_n$の集合を網群Cとする。Cは、システム層のエンティティ$E_1,\ldots,E_n$に、各々$S_1,\ldots,S_n$を通して、OPサービスを提供する。各E_iは、応用エンティティA_iに、システムSAP(SSAP) D_iを通して、ある通信サービスを提供する($i = 1, \ldots, n$)。

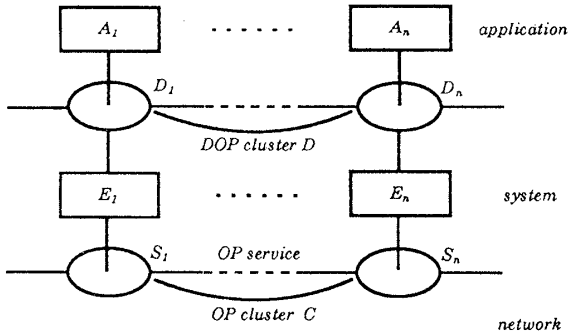


図1: システム階層

システム群Dは、SSAP $D_1, \ldots, D_n$間に存在する。各D_iは、システムエンティティE_iにより提供される。Dには、**スキーマ**と**インスタンス**との2つの側面がある。Dのスキーマは、エンティティの構成を示し、$D = \langle E_1, \ldots, E_n\rangle$と書く。$D$のインスタンスは、群が開設された状態である。$E_i$の状態が動作中から停止中に変化することを$E_i$の**停止**、停止中から動作中となることを**復旧**とする。E_iが復旧したとき、停止前の状態を何も記憶していないとき、**完全停止**とする。少なくともDのスキーマについて知っているとき、**部分停止**とする。各E_iの状態を$state(E_i)$と書く。$state(E_i) \in \{$動作中(A), 停止中$(S)\}$である。Dのインスタンス状態を$state(D)$とし、動作中のエンティティの組$\langle E_{a1}, \ldots, E_{ah}\rangle (\subseteq D)$により示す。

群内のあるE_iが停止した場合に、群インスタンスが消滅する群[?, ?]を**静的群**とする。これに対して、E_iが停止しても、他の動作中エンティティ間で放送通信サービスを提供し続ける群を**動的インスタンス群**とする。一方、新たにエンティティが群に加入、離脱することにより、スキーマが変化する群を**動的スキーマ群**とする。本論文では、OPサービスを提供する動的インスタンス群について論じる。

3　動的群$OP(DOP)$プロトコル

DOP群$D = \langle E_1, \ldots, E_n\rangle$を提供するプロトコルについて述べる。

3.1　動的群$OP(DOP)$サービス

動的$OP(DOP)$**群**$D = \langle E_1,\ldots,E_n\rangle$では、$D$内のある$E_i$が停止しても、残りの動作中エンティティにより、$OP$サービスが提供される。各$E_i$からみて動作中であるエンティティの組を、$E_i$の視野とし、$state(D)_i$と書く。$E_i$からみた、$E_j$の状態を$state(E_j)_i$と書く。動作中の各エンティティは、送信元の送信順にPDUを受信する。

3.2　エンティティ状態

E_iは、E_jから一定時間何のPDUも受信しないとき、E_iはE_jの**停止を認識する**とする。このとき、$state(E_j)_i$を**停止合意中**(AS)とし、E_jの停止を他のエンティティに通知する。$state(E_j)_i = A$のときに、E_iが他のE_kより、E_jの停止通知を受ける場合がある。この状態$state(E_j)_i$を**停止認識中**(AAS)とする。以上から、E_iがE_jの停止について合意するための条件は以下である。

(1) E_iがE_jの停止を認識する。
(2) 動作中の全エンティティから、E_jの停止認識通知を受信する。

次に、部分停止していたE_jが復旧する場合を考える。E_jは復旧すると、RCV PDUを放送する。E_iが、E_jからRCVを受信したとき、E_iはE_jの**復旧を認識する**とする。E_jの復旧通知を全動作中エンティティに放送する。このとき、$state(E_j)_i$を**復旧合意中**(SA)とする。他のE_kから、E_jの復旧通知を受けたとき、E_iは、$state(E_j)_i$を**復旧認識中**(SSA)とする。E_iは以下が成り立つとき、E_jの復旧に合意したとする。

(1) E_iは、E_jの復旧を認識する。
(2) E_iは、全動作中エンティティから、E_jの復旧認識通知を受信する。

3.3　停止、復旧手続き

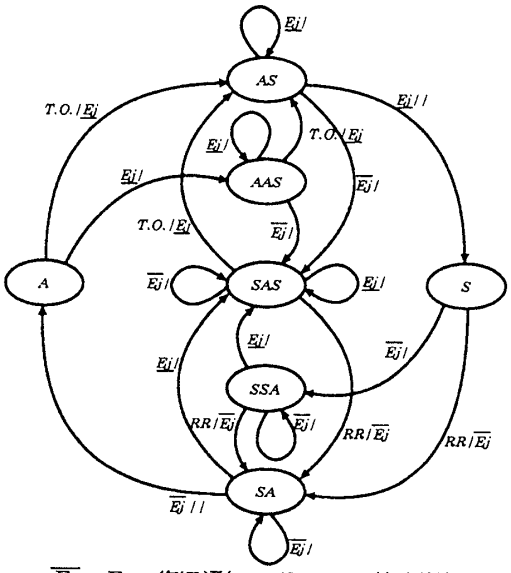
各E_iからみた動作中エンティティの集合を$A_i (\in D)$とする。S_i、AS_i、AAS_iを各々、停止合意、停止合意中、停止認識中のエンティティ集合とする。SA_i、SSA_iを各々、復旧合意中、復旧認識中のエンティティ集合とする。各E_iが送信するPDU pは、以下の項目を含む。

$p.STATE = state(D)_i$を示すビットマップ。
$p.TRANS =$状態の変化を示すビットマップ。

ここで、ビットマップBに対して、$B[i]$はi番目のビットを示す。各E_iは、他のE_jから次に受信予定のPDUのシーケンス番号REQ_j $(j = 1, \ldots, n)$を持つ。ビットマップ$STATE$は、E_iが動作中と認識しているエンティティの集合AA_iを示す($E_j \in AA_i$ならば$STATE[j] = 0$)。$TRANS$は、E_iが停止又は復旧合意待ちのエンティティ集合$AS_i \cup SA_i$を示す($E_j \in AS_i \cup SA_i$ならば$TRANS[j] = 1$)。

E_jが停止したとする。E_iは以下の手順によりE_jの停止に合意する。

Dynamic Group Communication Protocol
Hitoshi Suzuki, Akihito Nakamura, and Makoto Takizawa
Tokyo Denki University



$\overline{E_j}$: E_jの復旧通知、 E_j :E_jの停止通知
図 2: $state(E_j)_i$ の遷移

[停止合意手続き]

(1) $state(E_j)_i = A$のとき、
  (a) タイムアウトにより、 E_jの停止を認識:
  $STATE[j] := TRANS[j] := 1$;
  $A_i := A_i - \{E_j\}$; $AS_i := AS_i \cup \{E_j\}$;
  $s.STATE := STATE$; $s.TRANS := TRANS$;
  $broadcast(\ s\ )$;
  (b) E_jの停止通知の受信:
  $state(E_j)_i := AAS$;
  $A_i := A_i - \{E_i\}$; $AAS_i := AAS_i \cup \{E_j\}$;

(2) $state(E_j)_i = AAS$で、 E_jについてタイムアウト:
  $STATE[j] := TRANS[j] := 1$;
  $AS_i := AS_i \cup \{E_j\}$; $AAS_i := AAS_i - \{E_j\}$;
  $sa.STATE := STATE$; $sa.TRANS := TRANS$;
  $broadcast(\ sa\ )$;

(3) $state(E_j)_i = AS$のとき: A_i内の全E_kについて、 $state(E_j)_i = state(E_j)_i$のとき、 E_jの停止についての合意する: $TRANS[j] := 0$; $state(E_j)_i := S$;
  $S_i := S_i \cup \{E_j\}$; $AS_i := AS_i - \{E_j\}$;□

[停止合意手続きの例] 図?? に、$D = \langle E_1, E_2, E_3, E_4\rangle$の停止合意手続きの例を示す。

(1) E_3からのPDUを一定時間受信しないことで、E_1はE_3の停止を検出する。E_1は、$STATE[3] := TRANS[3] := 1 (state(E_3)_1 := AS)$とし、$s.STATE := STATE$、$s.TRANS := TRANS$なる$STOP$ PDU sを放送する。

(2) E_1からの$STOP$ sを受信した動作中の各$E_k (k = 1,2,4)$は、$(s.STATE\ \&\ s.TRANS)[3] = 1$であるので、$state(E_3)_k = AAS$とする。各$E_k$は、タイムアウトにより$E_3$の停止を認識した後、$state(E_3)_k = AS$、即ち$STATE := TRANS := 0100$とする。$sa.STATE := STATE$、$sa.TRANS := TRANS$なる$STOP_ACK$ saを放送する。

(3) E_kは、A_k内の全エンティティからsaを受信したとき、全動作中エンティティが同一の$STATE$を持つことがわかり、$state(E_3)_k = S$とする ($TRANS[3] := 0$)。各E_kは、E_3から受信したPDUの内、$p.DSEQ \geq REQ_3$なるpを受信ログ内から廃棄する。□

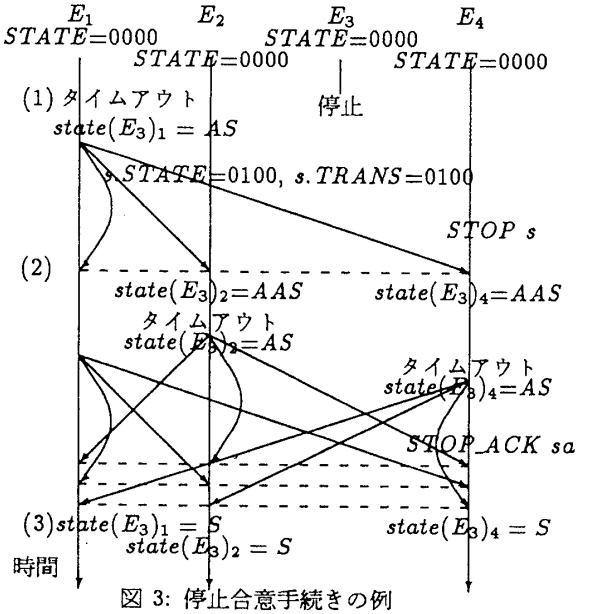


図 3: 停止合意手続きの例

次にE_jが停止から復旧したとする。E_iは以下の手順によりE_jの復旧に合意する。

[復旧合意手続き]

(1) $state(E_j)_i = S$のとき、
  (a) E_iが、E_jからRCVを受信 (E_jの復旧認識):
  $STATE[j] := 0$; $TRANS[j] := 1$;
  $S_i := S_i - \{E_j\}$; $SA_i := SA_i \cup \{E_j\}$;
  $ra.STATE := STATE$; $ra.TRANS := TRANS$;
  $broadcast(\ ra\ )$;
  (b) E_iが、E_jの復旧通知の受信:
  $state(E_j)_i := SSA$;
  $S_i := S_i - \{E_j\}$; $SSA_i := SSA_i \cup \{E_j\}$;

(2) $state(E_j)_i = SSA$で、 E_jからRCVの受信:
  $STATE[j] := 0$; $TRANS[j] := 1$;
  $SA_i := SA_i \cup \{E_j\}$; $SSA_i := SSA_i - \{E_j\}$;

(3) $state(E_j)_i = SA$で、 A_i内の全E_kについて、$state(E_j)_k = state(E_j)_i$ならば、 E_jの復旧について合意する:
  $TRANS[j] := 0$; $state(E_j)_i := A$;
  $A_i := A_i \cup \{E_j\}$; $SA_i := SA_i - \{E_j\}$;□

4 まとめ

本論文では、群内のエンティティの状態が変化する動的インスタンス群に対して、動作中の全エンティティ間にOPサービスを提供するDOPプロトコルを示した。本プロトコルにより、インスタンス状態が同時に変化する場合に、動作中のエンティティ間のデータ転送を停止せずに、各々のインスタンス状態の変化について、動作中のエンティティ間で合意することができる。

参考文献

[1] 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).

[2] Nakamura, A. and Takizawa, M., "Reliable Broadcast Protocol for Selectively Ordering PDUs," *Proc. of the IEEE ICDCS-11*, 1991, pp.239-246.

[3] Takizawa, M. and Nakamura, A., "Partially Ordering Broadcast (PO) Protocol," *Proc. of the IEEE INFOCOM90*, 1990, pp.357-364.