

## 5ZA-7 アドホックネットワークのためのチェックポイントプロトコル\*

東京電機大学 工学部 情報システム工学科†  
小野 真和 梶垣 博章‡ §

## 1 はじめに

近年、ノート型 PC などの移動コンピュータや IEEE802.11 などの無線 LAN プロトコル技術が広く普及し、利用されている。また、移動コンピュータのみで構成され、イベント会場や災害現場などで一時的に利用されるアドホックネットワークへの要求が高まっている。ネットワーク環境におけるミッションクリティカルアプリケーションの実現手法として、チェックポイントリカバリプロトコルがある。しかし、従来のプロトコルは有線ネットワークを対象としている。そのため、状態情報を格納するための安定記憶が存在することと、メッセージの送信元コンピュータと送信先コンピュータの同期によって一貫性のないメッセージが検出、回避できる程度に十分な通信帯域幅が存在することが前提となっている。本論文では、これらの前提が成立しないアドホックネットワークにおけるチェックポイントプロトコルを提案する。

## 2 従来手法

## 2.1 チェックポイントプロトコル

アドホックネットワーク  $\mathcal{N} = (\mathcal{V}, \mathcal{E})$  とは、移動コンピュータの集合  $\mathcal{V}$  と互いに直接メッセージを交換することが可能な移動コンピュータ  $M_i, M_j \in \mathcal{V}$  の間の双方向リンク  $\langle M_i, M_j \rangle$  の集合  $\mathcal{E}$  で定まるネットワークである。一般に、ネットワーク環境において、チェックポイントプロトコルによって各移動コンピュータ  $M_i \in \mathcal{V}$  が状態情報  $S_i$  を獲得し、安定記憶に保存することによって設定されるローカルチェックポイント  $c_i$  の集合であるグローバルチェックポイント  $C_V$  が一貫性を持つとは、次の性質を満たすことをいう [2]。

[定義]

- 1) 送信元移動コンピュータ  $M_s$  から送信先移動コンピュータ  $M_r$  へ配送されるメッセージ  $m$  が紛失メッセージであるとは、グローバルチェックポイント  $C_V$  に対して、 $Send(m)$  が  $c_s$  に先行し、 $c_r$  が  $Receive(m)$  に先行することである。なお、 $Send()$  と  $Receive()$  は、アプリケーション層におけるメッセージ送受信イベントである。
- 2) メッセージ  $m$  が孤児メッセージであるとは、 $C_V$  に対して、 $c_s$  が  $Send(m)$  に先行し、 $Receive(m)$  が  $c_r$  に先行することである。
- 3) 一貫性のあるグローバルチェックポイントとは、孤児メッセージを含まず、すべての紛失メッセージをリカバリ時に再送信可能であるものである。□

従来のチェックポイントプロトコルにおいては、 $m$  が  $C_V$  に対する紛失メッセージや孤児メッセージとなることを  $M_r$  でのみ判定することを前提としている。そのため、これらの発生を回避するために、アプリケー

ションメッセージの送信を禁止し、システム全体での同期を行なうことを必要としていた。しかし、アドホックネットワークにおいては、無線通信が低信頼であることや、競合や伝送遅延などにより、システム全体の同期に要する通信オーバーヘッドが大きくなる。また、ネットワークの構成要素が移動コンピュータのみであることから、安定記憶の実現が困難である。

## 3 提案プロトコル

以下の条件のもとでプロトコルを構成する。

[前提条件]

- 1) アドホックネットワークに含まれるすべての移動コンピュータは、チェックポイントプロトコルの実行中、マルチホップ配送により互いにメッセージ交換が可能である。
- 2) 各移動コンピュータは、隣接する移動コンピュータのリストを保持している。
- 3) 隣接する移動コンピュータ間の通信リンクは双方向であり、半二重通信が行なわれる。また、ユニキャスト通信は、受信確認と再送機構により、メッセージの紛失なく実現されているものとする。□

チェックポイントプロトコルの基本形を示す。チェックポイントプロトコルの開始は、任意の移動コンピュータが任意のタイミングで行なうことができる。チェックポイント設定要求の伝達と、チェックポイント間の同期は、チェックポイント設定要求メッセージ  $CReq$  のブロードキャスト [1] によって実現される (図 1)。

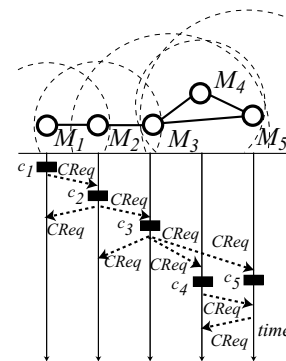


図 1: チェックポイントプロトコル

$CReq$  を受信した移動コンピュータ  $M_i$  は、現在の状態情報  $S_i$  を獲得することによってローカルチェックポイント  $c_i$  を設定するとともに、 $CReq$  を隣接する移動コンピュータ群、すなわち、 $M_i$  の無線信号到達範囲内に存在するすべての移動コンピュータにブロードキャストする。これを繰り返すことによって、前提条件 1) により、すべての移動コンピュータがローカルチェックポイントを設定することができる。また、移動コンピュータ  $M_i$  の状態情報  $S_i$  を保存する安定記憶を実現するために、 $S_i$  を複数の隣接移動コンピュータに記憶させる手

\*Checkpoint Protocol for Mobile Ad hoc Networks

†Tokyo Denki University

‡Masakazu Ono and Hiroaki Higaki

§{masa, hig}@higlab.k.dendai.ac.jp

法を用いる。各移動コンピュータは、ローカルチェックポイント  $c_i$  における状態情報  $S_i$  を獲得してから  $CReq$  のブロードキャストを行なうことから、 $S_i$  を  $CReq$  によって配送することにより、追加のメッセージを要することなく  $S_i$  の配送が実現される。

ここで、チェックポイントプロトコルの実行と並行に送受信されたメッセージは、紛失メッセージや孤児メッセージとなる可能性がある。紛失メッセージは、いずれかの移動コンピュータに保存し、リカバリ回復時に、保存されたメッセージを再送信することによって、システム状態の一貫性を維持することができる。一方、孤児メッセージは、リカバリ再実行時に送信元移動コンピュータが同一のメッセージを再度送信する保障がないことから、その発生を回避しなければならない。

メッセージの伝達とローカルチェックポイントの関係から、紛失メッセージとなる可能性があるメッセージ  $m_l$  を中継移動コンピュータ、すなわち  $m_l$  の送信元でも送信先でもない移動コンピュータが検出することができる。もし、この検出を送信先移動コンピュータ  $M_r$  で行なったとするならば図2に示すように、 $M_r$  が  $m_l$  を受信した時点、すなわち  $Receive(m_l)$  においては、 $M_r$  がすでに  $CReq$  を隣接移動コンピュータへブロードキャスト送信済みであることが考えられる。この場合、 $m_l$  を隣接移動コンピュータに記憶させるために、 $m_l$  を含む制御メッセージをブロードキャストする必要がある。しかし、 $M_r$  は、 $M_r$  がブロードキャストすべき紛失メッセージの数を決めることができないため、 $M_r$  においてチェックポイントプロトコルの終了ができなくなってしまう問題が発生する。

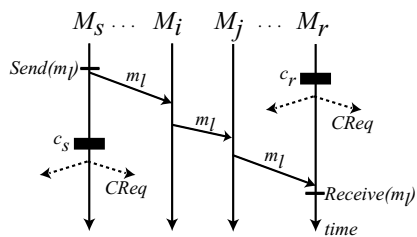


図 2: 紛失メッセージの検出とチェックポイント

そこで、紛失メッセージとなる可能性のあるメッセージ  $m_l$  を  $CReq$  を送信する前に検出し、保存することが可能な移動コンピュータの存在が不可欠である。各  $m_l$  に対して、図3(a)(b)のいずれかの事象が配送経路上のいずれかの移動コンピュータで発生する。図3(a)の場合は、 $M_j$  において  $m_l$  を検出し、 $CReq$  に含めてブロードキャストできる。一方、図3(b)の場合は、 $m_l$  が紛失メッセージとなる可能性があるメッセージであることを検出できるのは  $M_j$  であるが、 $M_j$  はすでに  $CReq$  メッセージを送信済みである。そこで、 $m_l$  の検出を  $M_j$  が行ない、受信確認メッセージを利用して  $M_i$  に通知する。この手法を実現するためには、 $M_i$  が  $CReq$  を送信する前にその結果を通知する必要があるため、メッセージ転送時から受信確認応答メッセージの受信までの間は  $CReq$  の送信を禁止する必要がある。

このとき、図4(a)のように、送信元  $M_1$  から送信されたメッセージ  $m$  について、 $M_2$  において紛失メッセージとなる可能性が検出され、保存されたにもかかわらず、送信先  $M_3$  において紛失メッセージとならない場合がある。このような場合にリカバリが発生すると、 $M_3$  は  $m$  を複数回受信してしまうこととなり、一貫性を保つことができない。そこで、図4(b)のように  $M_2$  にお

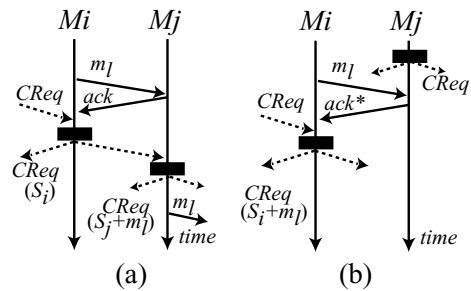


図 3: 中継移動コンピュータでの紛失の可能性のあるメッセージの検出とチェックポイント

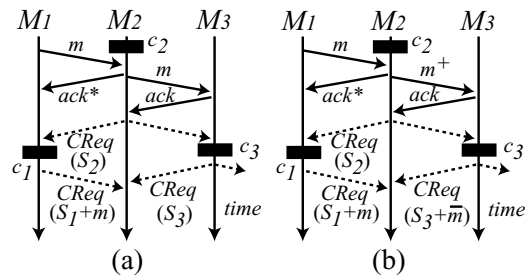


図 4: 誤検出された紛失メッセージの多重受取りの回避

いて紛失メッセージの可能性を検出したメッセージ  $m$  を次の転送先に送信する場合、転送するメッセージに再送信されることを示す情報を付与し、 $m^+$  として転送する。 $M_3$  において  $m^+$  が紛失メッセージとならなかった場合、 $M_3$  がリカバリ時に再送信された  $m$  を破棄する情報  $\bar{m}$  を  $CReq$  に付与する。これによって、一貫性を保つことができる。

なお、孤児メッセージとなる可能性のあるメッセージは、中継移動コンピュータでは対処を行わない。送信先移動コンピュータで受信を遅延させることによるのみ対処すれば十分である。

#### 4 まとめと今後の課題

本論文では、アドホックネットワークにおけるチェックポイントプロトコルを示した。ここでは、紛失メッセージとなる可能性のあるメッセージを中継移動コンピュータで検出し、保存する機構を導入することにより、各移動コンピュータが隣接移動コンピュータに対して状態情報とメッセージログを含むチェックポイント要求メッセージの送信を一度で行なうことができ、同期オーバーヘッドを削減することができる。耐  $k$  故障のための状態情報記憶手法 [3] と組み合わせることが今後の課題である。

#### 参考文献

- [1] Corson, M.S. and Ephremides, A., "A Distributed Routing Algorithm for Mobile Wireless Networks," ACM Journal of Wireless Networks, vol. 1, No. 1, pp. 61-81 (1995).
- [2] Elnozahy, E. N. and Zwaenepoel, W., "On the Use and Implementation of Message Logging," Proc. of the Fault-Tolerant Computing Symposium, pp. 298-307(1994).
- [3] 平川, 小野, 松垣, "無線マルチホップ型ネットワークのためのチェックポイントプロトコル," 情報研報, Vol. 2003, No. 118, pp. 79-84(2003).