

## ループ探索に基づいたアドホックルーティングプロトコルの実装

東京電機大学 工学部 情報システム工学科

神林 洋平 佐川 陽介 桧垣 博章

{kan, sgw, hig}@higlab.k.dendai.ac.jp

ノート型 PC や PDA といった移動コンピュータを IEEE802.11 などの無線 LAN 技術によって接続することで構成されるモバイルネットワークを基礎としたネットワークコンピューティング環境への要求が高まっている。モバイルネットワークのひとつにアドホックネットワークがある。アドホックネットワークでは、専用のルータではなく、移動コンピュータがメッセージをルーティングすることによって、移動性、柔軟性、経済性の高いネットワークを構築できる。コンピュータの移動性を考慮したオンデマンド型のアドホックルーティングプロトコルとして DSR、AODV、DSDV 等が提案、実装されている。本論文では、アドホックネットワークの多様性と不安定性を考慮し、片方向接続を含む場合を想定して設計されたアドホックルーティングプロトコルである LBSR (Loop Based Source Routing) プロトコルを提案し、FreeBSD+PAO への実装について述べる。

## Implementation of Loop Based Source Routing Protocol

Youhei Kanbayashi, Yousuke Sagawa and Hiroaki Higaki

{kan, sgw, hig}@higlab.k.dendai.ac.jp

Department of Computers and Systems Engineering

Tokyo Denki University

In an ad-hoc network where only mobile computers are included, higher mobility and flexibility are supported than an infrastructured network with access points. In this paper, we propose LBSR (Loop-Based Source Routing) which is a novel on-demand routing protocol supporting asymmetric wireless communication links. For achieving a route from a source to a destination, LBSR searches multiple loop routes. Here, smaller number of multicasts in a flooding protocol are required than conventional protocols like DSR. We implement LBSR on the FreeBSD+PAO environment.

## 1 背景と目的

近年、コンピュータの小型化と高性能化により、ノート型パーソナルコンピュータや PDA といった携帯型の移動コンピュータが広く利用されるようになってきた。また、ネットワーク技術の発達と WWW をはじめとするネットワークアプリケーションの普及により、ネットワークに接続されるコンピュータの比率が急速に高まっている。このような背景から、移動コンピュータをネットワークに接続して利用するモバイルコンピューティングへの要求が高まっており、IEEE802.11 [1] や HIPERLAN [2] などの無線 LAN 技術が研究開発されている。

会議場やイベント会場におけるネットワークサービスのような移動コンピュータの移動性を活用したネットワークアプリケーションの実行環境として、アクセスポイントを用いないアドホックネットワークが注目されている。アドホックネットワークは、複数の移動ロボット群からなる協調型地雷除去システムのようなアクセスポイントの設置が困難な状況におけるネットワーク構築手段としても有効である。ここでは、ネットワークに専用のルータは接続されておらず、移動コンピュータがメッセージのルーティングを行なう。これによって、移動コンピュータのみからなる、より柔軟性の高いコンピュータネットワークを構築することが可能となる。アドホックネットワークにおけるルーティングプロトコルとして、AODV [3] や DSDV [4] などがある。これらのプロトコルでは、すべての移動コンピュータの無線信号到達範囲の大きさは同一であると仮定している。すなわち、2 台の移動コンピュータ間の接続はすべて双方向であるという前提に基づいた設計がなされている。しかし、無線通信に用いられる電磁波は、反射、回折、透過を行なうため、無線信号到達範囲の大きさはすべての移動コンピュータで同一であるとは限らない。また、各移動コンピュータの無線信号到達範囲の大きさも一定とは言えず、時間とともに変化する。そのため、2 台の移動コンピュータ間に片方向接続が存在する場合がある。本論文では、片方向接続を含むアドホックネットワークにおけるループ探索に基づいたルーティングプロトコル LBSR [5] の FreeBSD+PAO への実装について述べる。

## 2 LBSR プロトコル

双方向接続のみからなるアドホックネットワークでは、送信元  $S$  から送信先  $D$  までの経路  $R_{S \rightarrow D}$  を検出すれば、同時に経路  $R_{D \rightarrow S}$  を検出したことになる。したがって、 $R_{D \rightarrow S}$  を用いることで、 $R_{S \rightarrow D}$  の経路情報を  $S$  が得ることができる。片方向接続を含むアドホックネットワークにおいては、 $S$  から

$D$  までの経路情報を  $S$  が得るためには、 $S$  から  $D$  への経路  $R_{S \rightarrow D}$  と  $D$  から  $S$  への経路  $R_{D \rightarrow S}$  が必要である。DSR [6] では、これらを 2 つの独立なフラッディングによって求めている。フラッディングは、**message diffusion protocol** [7] を無線 LAN 環境に適用したものである。無線 LAN に利用される無線通信メディアの多くはブロードキャストベースであるため、ある移動コンピュータが送信した無線信号は、その到達範囲内にあるすべての移動コンピュータが受信することができる。ある移動コンピュータがメッセージ  $m$  をブロードキャストし、それを受信したすべての移動コンピュータが同様に  $m$  をブロードキャストする。これを繰り返すことによって、マルチホップで到達可能なすべての移動コンピュータに  $m$  を配送することが可能である。これがフラッディングである。これに対して、LBSR では、 $R_{S \rightarrow D}$  と  $R_{D \rightarrow S}$  を連結して得られるループ経路を探索している。特に、求める経路である  $S$  から  $D$  を通り  $S$  に戻るループ経路を探索する途中で検出される  $S$  から  $D$  を通らずに  $S$  に戻るループ経路をユニキャストで利用することにより、マルチキャストの利用を削減し、経路探索に要するプロトコルオーバーヘッドを低減している。

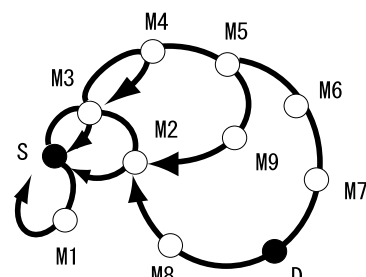


図 1: ループ経路構築

LBSR では、経路探索時に 2 種類のメッセージ RREQ と LOOP を用いる。RREQ は、 $S$  から  $S$  へ戻るループ経路を探索するためのメッセージであり、経路上にある移動コンピュータのアドレスのシーケンスが格納されている。LOOP には、 $S$  から  $S$  に戻るループ経路上のアドレスのシーケンスが含まれている。LOOP は、このループ経路上をユニキャストで配送され、受信した移動コンピュータに  $S$  へ至る経路を通知する。また、各移動コンピュータ  $M_i$  は、2 種類の経路キャッシュを持つ。RREQ キャッシュには、送信元  $S$  から  $M_i$  までの経路 (アドレスのシーケンス) が格納される。また、LOOP キャッシュには、 $M_i$  から  $S$  までの経路が格納される。RREQ キャッシュ、LOOP キャッシュへの経路追加は、それぞれ RREQ メッセージ、LOOP メッセージの受信時になされる。LOOP メッセージは、

LOOP キャッシュの経路情報を用いてユニキャストされる。

[LBSR プロトコル]

1.  $S$  は、 $S$  の無線信号到達範囲内にあるすべての移動コンピュータ  $M_i$  に RREQ をブロードキャストする。
2.  $M_i$  または  $D$  が、RREQ を受信したならば、以下のいずれかの処理を行ってから、RREQ に含まれる経路を RREQ キャッシュに保存する。
  - 2-1. RREQ キャッシュが空であるならば、ただちに、RREQ を自身の無線信号到達範囲内にあるすべての移動コンピュータにブロードキャストする。ただし、RREQ をブロードキャストする前に自身のアドレスを RREQ に含まれるアドレスシーケンスの末尾に加える。
  - 2-2.  $M_i$  の RREQ キャッシュが空でないならば、LOOP キャッシュに含まれる経路に沿ってユニキャストで RREQ を送信する。 $M_i$  は RREQ をユニキャストする前に自身のアドレスを RREQ に含まれるアドレスシーケンスの末尾に加える。LOOP キャッシュが空である場合には、LOOP を受信して LOOP キャッシュに経路が格納されるまで待つ。このとき、RREQ は LOOP にピギーバックされる。
  - 2-3.  $D$  の RREQ キャッシュが空でないならば、この RREQ を破棄する。
3.  $S$  が  $D$  をアドレスシーケンスに含まない RREQ を受信したならば、RREQ に含まれるアドレスシーケンスを用いてソースルーティングされる LOOP をユニキャストで送信する。
4.  $M_i$  が LOOP を受信したならば、これに含まれるアドレスシーケンスから  $S$  への経路を取得し、これを LOOP キャッシュに保存する。そして、LOOP に含まれるアドレスシーケンスを用いて、LOOP をユニキャストで送信する。
5.  $S$  が  $D$  のアドレスをシーケンスに含む RREQ を受信することで  $R_{S \rightarrow D} + R_{D \rightarrow S}$  が得られる。以降、 $S$  は RREQ を受信しても LOOP を送信しない。□

### 3 LBSR の実装

オンデマンド型ソースルーティングプロトコルである DSR は、FreeBSD3.3+PAO 上で実装されている [8]。ここで、DSR は、ネットワーク層の IP とトランスポート層の TCP、UDP の中間に位置し

ている。受信したメッセージの種類は、IP ヘッダのプロトコルフィールドの値により判別可能であり、これを処理するための関数が呼び出される。また、すべてのメッセージの送信時には、送信先をブロードキャストアドレスとした IP ヘッダと IEEE802.11 ヘッダが DSR によって作成される。つまり、この実装においては、すべての DSR メッセージ (RREQ と RREP に加えて DATA も) はブロードキャストによって配送される。

本論文では、FreeBSD3.3+PAO を対象として、[8] の方法に基づき、IP と TCP、UDP の中間に LBSR を実装した。メッセージの種類は、IP ヘッダのプロトコルフィールドの値によって行なっている。ただし、ブロードキャストとユニキャストの両方を利用している点が [8] の実装とは異なっている。LBSR の構成を図 2 に示す。

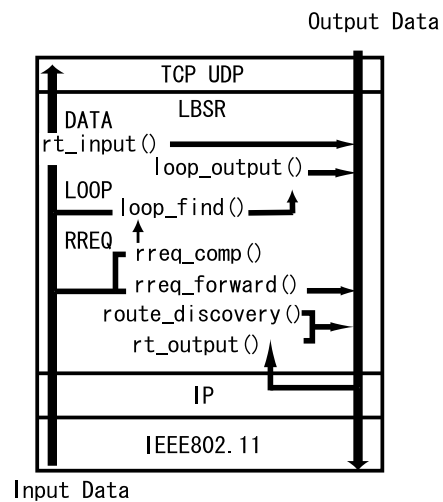


図 2: LBSR の構成

受信した LBSR メッセージの IP ヘッダに含まれるプロトコルフィールドの値 (RREQ:55、LOOP:57、DATA:60) に対応して、それぞれのメッセージを処理する関数が実行される。RREQ を受信したならば、RREQ キャッシュを参照する。RREQ キャッシュが空であるならば、RREQ のブロードキャスト (フラッディング) を行なう関数 `rreq_forward` が呼び出される。RREQ キャッシュが空でないならば、`loop_output` 関数により、ループ経路に沿ったユニキャストによる RREQ の配送を行なう。いずれの場合も、RREQ に格納された送信元からの経路を RREQ キャッシュに格納する。LOOP を受信したならば、LOOP に格納された経路情報を LOOP キャッシュに格納し、`loop_out` 関数で LOOP をループ経路に沿ってユニキャストで配送する。DATA を受信したならば、`rt_input` 関数でメッセージの宛先アドレスを調べる。自身のアドレスであれば、上位

層にデータ部を渡す。それ以外の場合、DATAの経路情報にしたがってユニキャストでメッセージを送信する。

上位層からの送信要求を処理する場合は、IPによってrt\_output関数、route\_discovery関数が起動される。キャッシュに経路が保持されているならば、これを用いてソースルーティングを行なう。保持されていないならば、RREQのフラッディングによるLBSR経路探索を開始する。

### 3.1 メッセージフォーマット

LBSRで用いる3種類のメッセージ、RREQ、LOOP、DATAのフォーマットについて説明する。

#### [RREQメッセージ(図3)]

RREQは、送信元*S*から送信先*D*へのループ経路を探索するためにフラッディングされるメッセージである。それぞれの経路探索に用いられるメッセージ群を区別するために、*S*によってID(識別子)が与えられる。ネットワーク内では、*S*のIPアドレスとこのIDとの2項組によって、経路探索に関連するメッセージ群を識別している。本実装では、8ビットの整数値をインクリメンタルかつサイクリックに使用している。また、RREQを送信する際、IPヘッダに含まれるプロトコルフィールドには57の値が設定され、送信元アドレスには、RREQを送信する各移動コンピュータ自身のIPアドレス、送信先アドレスには、ブロードキャストアドレスが設定される。RREQはブロードキャストで送信される。

**ID:** 経路探索ID。同一送信元*S*からのRREQを一意に識別するためのID。

**Length:** RREQメッセージ全体の長さ。byte単位で表す。

**Option:** オプション。0が格納される。

**Padding:** パディング。常に0の値である。

**Source address:** 送信元*S*のIPアドレス。

**Target address:** 送信先*D*のIPアドレス。

**Addr\_num:** このRREQのアドレスシーケンスに格納されているIPアドレスの数。

**Option1,2:** オプション。どちらにも0が格納される。

**Address[Addr\_num]:** アドレスシーケンス。RREQが経由した各移動コンピュータのIPアドレスが順に格納される。

#### [LOOPメッセージ(図3)]

LOOPは、送信元*S*が発見したループ経路を各移動コンピュータのLOOPキャッシュに設定するためのメッセージである。LOOPに含まれるIDは、対応するRREQと同じ値であり、RREQと同様に、

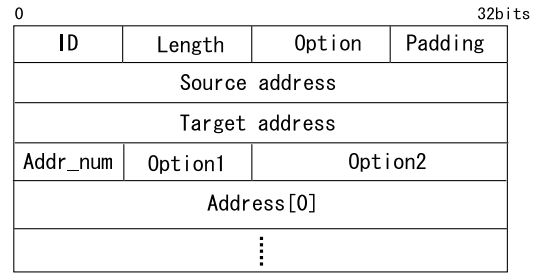


図3: RREQとLOOP

送信元*S*のアドレスとの2項組によって、関連する一群のメッセージを識別する。LOOPを送信する際、IPヘッダに含まれるプロトコルフィールドには57の値が格納され、送信元アドレスには、LOOPを送信する移動コンピュータのIPアドレス、送信先アドレスには、LOOP内のアドレスシーケンスから取り出された次ホップの移動コンピュータのIPアドレスが設定される。LOOPはユニキャストで送信される。

**ID:** 経路探索ID。同一送信元*S*からのLOOPを一意に識別するためのID。対応するRREQメッセージのIDと同じ値である。

**Length:** LOOPメッセージ全体の長さ。byte単位で表す。

**Option:** 通常は0である。送信元*S*が、送信先*D*を含んだループ経路 $R_{S \rightarrow D} + R_{D \rightarrow S}$ を検出した場合に、*S*がこのフィールドに1を格納して、LOOPを送信する。このLOOPを受信した各移動コンピュータは、優先的なLOOPとして扱う。

**Padding:** パディング。常に0である。

**Source address:** 送信元*S*のIPアドレス。

**Target address:** 送信先*D*のIPアドレス。

**Addr\_num:** ループ経路に属する移動コンピュータの数。

**Option1,2:** オプション。どちらも0が格納される。

**Address[Addr\_num]:** アドレスシーケンス。検出されたループ経路に属する移動コンピュータのIPアドレスが順に格納される。

#### [DATAメッセージ(図4)]

DATAは、アプリケーションデータを配送するためのメッセージであり、送信元*S*から送信先*D*へソースルーティングされる。IPヘッダに含まれるプロトコルフィールドの値は60である。また、IPヘッダの送信元アドレスはDATAを送信する移動コンピュータのIPアドレス、送信先アドレスには、DATA内のアドレスシーケンスから取り出された次ホップの移動コンピュータのIPアドレスが

設定される。DATA はユニキャストで送信される。

**Protocol:** プロトコル番号。上位層に渡すアプリケーションデータのプロトコル番号の値が入る。

**Length:** DATA メッセージ全体の長さ。byte 単位で表す。

**Segment Left:** ルーティングする経路の残りの移動コンピュータのアドレス数。経路上にある各移動コンピュータは、受信したメッセージを転送する際にこの値を 1 減らす。

**Padding:** パディング。常に 0 である。

**Source address:** 送信元 *S* の IP アドレス。

**Target address:** 送信先 *D* の IP アドレス。

**Address[Addr\_num]:** アドレスシーケンス。*S* から *D* までの経路上にある移動コンピュータの IP アドレスが順に格納されている。

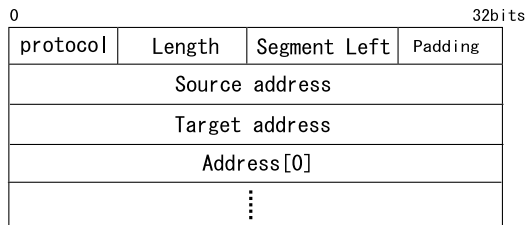


図 4: DATA

### 3.2 キャッシュ

各移動コンピュータは、メッセージの送信元 *S* と経路探索 ID の 2 項組のそれぞれに対して、RREQ キャッシュと LOOP キャッシュを持つ。

[RREQ キャッシュ(図 5)]

RREQ を受信した場合に送信元 *S* からこの移動コンピュータまで RREQ が配送された経路が格納されるキャッシュである。ループ経路は LOOP キャッシュと RREQ キャッシュによって検出されるが、新たなループ経路が発見された場合や、一定時間使用されなかった経路は RREQ キャッシュから削除される。RREQ キャッシュの構造は以下の通りである。

**Flag:** 送信先 *D* へのループ経路が検出された場合に設定されるフラグ。フラグが設定された後に受信した送信元 *S* と経路探索 ID の等しい RREQ パケットは破棄される。

**ID:** 経路探索 ID。

**Src:** 送信元 *S* の IP アドレス。

**Dst:** 送信先 *D* の IP アドレス。

**Rt\_list:** RREQ に含まれる経路情報が格納される。後から LOOP を受信した場合、LOOP メッセージの経路情報と Rt\_list に格納された経

路情報とを用いて新規ループ経路の検出を行なう。その後、Rt\_list に格納されている経路情報は、すべて破棄される。

**addr\_num:** 受信した RREQ メッセージが経由した移動コンピュータの数に 1 を加えたもの。

**address[addr\_num]:** 受信した RREQ が経由した移動コンピュータの IP アドレスに自身の IP アドレスを追加したアドレスシーケンス。

**Next\_rt\_list:** LOOP を受信していない場合、送信元 *S* と経路探索 ID の 2 項組が等しい RREQ を複数受信したならば、複数の経路情報を格納するための Rt\_list を連結リストで管理するために用いるポインタ。

**Next\_cache:** 送信元 *S* と経路探索 ID の 2 項組が異なる、すなわち異なる経路探索の情報をリストで管理するために用いるポインタ。

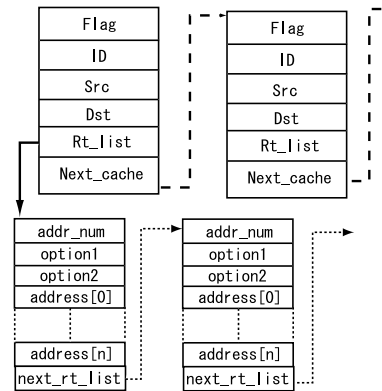


図 5: RREQ キャッシュ

[LOOP キャッシュ(図 6)]

送信元 *S* 以外の各移動コンピュータは、IP ヘッダの送信先アドレスが自身の IP アドレスに等しい LOOP を受信したならば、LOOP キャッシュが有効になり、LOOP の送信が可能となる。LOOP キャッシュの構造は以下の通りである。

**Flag:** 送信先 *D* へのループ経路が検出された場合に設定されるフラグ。フラグが設定された後に受信した送信元 *S* と経路探索 ID の等しい LOOP パケットは破棄される。

**ID:** 経路探索 ID。

**Src:** 送信元 *S* の IP アドレス。

**Dst:** 送信先 *D* の IP アドレス。

**Rt\_list:** 最初に受信した LOOP メッセージに含まれる経路情報のみが格納される。2 回目以降に受信した LOOP は、ここに格納された経

路に沿って、 $S$  まで配送される。

**addr\_num**: ループ経路に属する移動コンピュータの数。

**address[addr\_num]**: ループ経路に属する移動コンピュータのアドレスシーケンス。

**Next\_cache**: 送信元  $S$  と経路探索 ID の 2 項組が異なる、すなわち異なる経路探索の情報をリストで管理するために用いるポインタ。

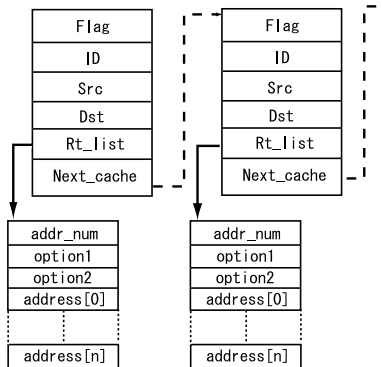


図 6: LOOP キャッシュ

これまでに主要機能の実装を終了し、4 台の移動コンピュータによる経路探索、データ配送の正常動作を確認した。なお、片方向接続を含む場合の動作についても確認済みである。

#### 4 まとめと今後の課題

本論文では、LBSR を FreeBSD3.3+PAO 上に実装する際のプロトコルスタック、動作システムについて述べた。今後は性能評価とキャッシュ管理機構、再経路構築機構についての拡張を行なう。

#### 参考文献

- [1] “Wireless LAN Medium Access control(MAC) and Physical Layer(PHY) Specifications,” Standard IEEE 802.11 (1997).
- [2] “Radio Equipment and Systems(RES); HIPER-LAN,” ETSI Functional Specifications (1995).
- [3] Perkins, C.E. and Royer, E.M., “Ad-hoc On-Demand Distance Vector Routing,” Proc. of IEEE Workshop on Mobile Computing Systems and Applications, pp. 99–100 (1999).
- [4] Perkins, C.E. and Bhagwat, P., “Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers,” ACM SIGCOMM’94, pp. 234–244 (1994).
- [5] 佐川, 松垣, “アドホックネットワークにおけるループ型ルーティングプロトコル,” 情報処理学会第 62 回全国大会, No. 3, pp. 359–360 (2001).
- [6] David, B., David, A., Hu, Y.C., Jorjeta, G., Jetcheva, “The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks,” Internet Draft, draft-ietf-manet-dsr-01.txt (1998).
- [7] Moses, Y. and Roth, G., “On reliable message diffusion.” Proc. of the 8th ACM Symposium on Principles of Distributed Computing, pp. 119–128 (1989).
- [8] <http://www.monarch.cs.cmu.edu/dsr-impl.html>.