

通信状態を考慮した経路選択を可能にする アドホックネットワークプロトコルの提案

三嶋 勇太

無線 LAN を標準搭載した携帯端末の普及に伴い、無線端末のみでネットワークを構築するモバイルアドホックネットワーク(MANET: Mobile Ad-hoc Network)の研究が期待されている。MANET で提案されている多くのアドホックルーティングプロトコルは、経路生成の際に TCP や UDP の通信状況が考慮されていない。そのため、最短経路が複数存在する場合にはどの経路を選ぶかは実装に依存したものとなっている。本稿では OLSR(Optimized Link State Routing)を拡張することにより、TCP, UDP それぞれの特性を活せる経路選択が可能なアドホックルーティングプロトコルを提案する。

A proposal on an Ad-hoc Network Protocol Enable to Considering Traffic Condition

YUTA MIKAMO

With the spread of mobile nodes, studies on MANET (Mobile Ad-hoc Network) that can build networks solely with mobile nodes are drawing much attention. However, most of the ad-hoc routing protocols have not considered the traffic conditions in the network. Thus, we propose an ad-hoc routing protocol consider traffic conditions by way of extending OLSR (Optimized Link State Routing).

1 はじめに

無線 LAN は配線が不要で端末が自由に移動できるなどの利便性からネットワークへの接続方法として需要が高まってきている。無線 LAN を構築する方法には、端末が必ず AP(Access Point)を介して通信を行うインフラストラクチャーモードによる方法と、端末同士で直接通信を行うアドホックモードによる方法がある。後者は、災害時やイベント会場などで一時的な無線ネットワークを構築できるモバイルアドホックネットワーク (MANET: Mobile Ad-hoc Network) [1]に応用されている。MANET は、あらゆる無線端末が中継端末となり得るため、その場でネットワークを構築することができるという特徴がある。近年では、インフラストラクチャーモードの AP間を MANET の技術で結合する無線メッシュネットワークの研究にも注目が集まっている。MANET を構築するには、各端末がアドホックルーティングプロトコルを用いてルーティングテーブル(以下 RT と記述)を生成する必要がある。アドホックルーティングプロトコルは、IETF(Internet Engineering Task Force)により、現在まで多くの方式が標準化されている[2-8]。

しかし、これらの方式は、経路生成の際に中継ホップ数が最短となる経路(最短経路)を探索することが目的となっており、最短経路が複数存在する場合にどの経路を選択するかは実装に任されている場合が多い。そのため、トラフィックが集中した中継ノードが発生すると、パケットロスが多発し、スループットが低下してしまうという課題がある[9]。複数経路の中から、適切な経路を選択することを目的としたアドホックルーティングプロトコルの研究として、以下のものが挙げられる。ABR (Associativity-Based Long-lived Routing)[10]の経路選択では、リンク切断が長時間起こらない、安定した経路を選択する。各ノードは一定間隔ごとに隣接ノードへビーコンを送信する。より多くのビーコンを受信したノードからなるリンクは持続性が高いと期待されるため、安定した経路により通信を行うことができる。しかし、ノードの移動が少ない環境では、ビーコンの受信回数に差が出ないため、スループットの向上が期待できない経路が選択される可能性がある。

ETR(Estimated-TCP-Throughput Maximization based Routing)[11]はDSR(Dynamic Source Routing Protocol)[3]を拡張することにより、宛先への複数の経路候補に対してTCPスループットを予測し、スループットの高い経路を選択する。TCPスループットは所定のモデル式を使って計算される。モデル式には遅延(RTT:Round-Trip Time)と往復パケット喪失率(RTPL:Round-Trip Packet Loss ratio)の情報が必要であり、これらの情報を収集するために新たな制御メッセージを設け、一定間隔で送信する。しかし、この方式はTCPスループットだけに注目しており、UDPスループットは考慮していない。また、新たな制御メッセージにより、ネットワークのオーバーヘッドが高くなるという課題がある。

また、これらのプロトコルはTCP通信とUDP通信で共通のRTを用いている。TCPとUDPでは性質が異なるため共通のRTを用いる場合はそれぞれの特性を十分生かすことができない。

本論文では、MANETのアドホックルーティングプロトコルの中でプロアクティブ型の代表的プロトコルOLSR(Optimized Link State Routing)を拡張することで、RTをTCP通信用とUDP通信用で別々に生成し、経路上のTCP/UDPの各プロトコルのトラフィック状態を考慮し、TCPとUDPの特性を生かした最適な経路選択を可能とするアドホックルーティングプロトコルPD-OLSR(Protocol Dependent-OLSR)を提案する。

以下、2章でMANETルーティングプロトコルの分類を示し、3章でOLSRの概要について説明する。4章ではPD-OLSRの概要を説明する。5章で経路生成方法を示し、6章でまとめを行う。

2 OLSR

2.1 OLSRの概要

OLSR(Optimized Link State Routing)はINRIAのプロジェクトHipercom[12]で提案されたMANETを構築するProactive型のルーティングプロトコルである。OLSRでは各ノードが隣接ノードへ定期的にブロードキャストするHELLOとネットワーク全体へ定期的にフラッディングするTCという制御メッセージを送受信することにより、自身の存在をネットワークの全ノードに把握させる。HELLOとTCで送信される情報は、各メッセージの送信元ノードのアドレス、送信元ノードが把握している自身の隣接ノードのアドレス、情報の新しさを識別するシーケンス番号などである。これらの情報はRTを生成するために必要な情報であり、各ノード内の情報リ

ポジトリーに登録される。OLSR の RT 生成プロセスは HELLO と TC の送受信が行われ、情報リポジトリーが更新されることによって進行する。

2.2 OLSR の RT 生成

OLSR の RT は宛先ノード(Dest), Dest への次ホップノード(Next), Dest までのホップ数(hop)から構成され、各 Dest に対して 1 つの経路を保持する。図 2.1 に、OLSR の RT 生成プロセスを示す。簡単のためノードは規則的に配置されており、電波到達範囲は隣接ノードまでとする。図 2.1 に示す RT は、ノード b を対象としてノード m および n への経路が新たに生成される様子を示している。制御メッセージの送受信によってノード m および n へのそれぞれ 1 つの最短経路が生成される。図ではノード b からノード m への最短経路 [b→d→g→k→m] を示している。OLSR では、複数の最短経路が存在する場合、どの経路を選択するかという手順は定義されておらず、実際の経路は、実装に依存したものとなり、多くの場合、最初に発見された最短経路が選択される。ここで、ノード g から f への通信が行われているものとする。ノード g から送信されるキャリアは隣接ノード c, d, f, h, j, k にも届く、そのため図 2.1 左のような経路が生成されると、ノード g の周辺はトラフィックが増加し、スループットが低下する可能性がある。このように OLSR ではネットワークのトラフィックが経路生成時に考慮されていない。

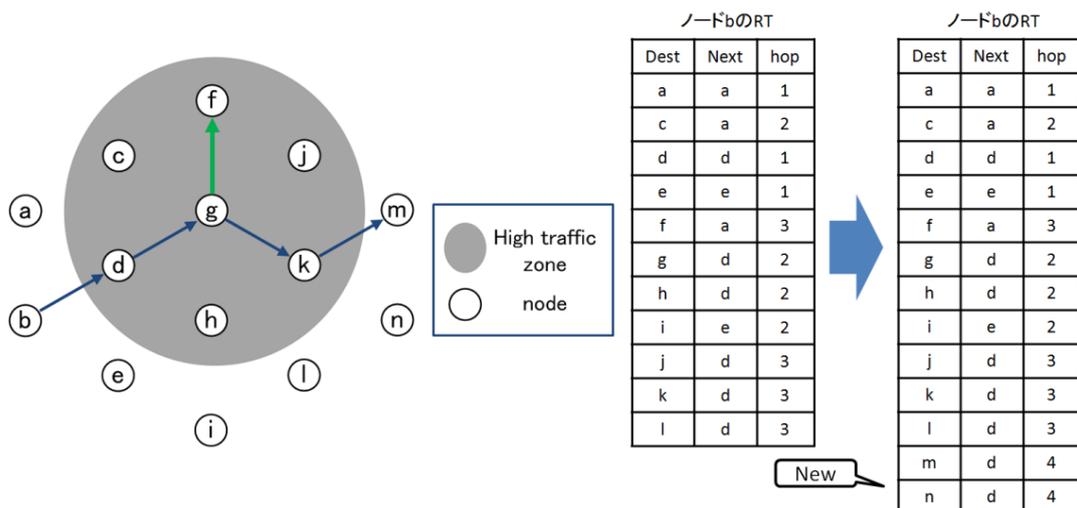


図 2.1 OLSR の RT 生成

3 提案方式 PD-OLSR

3.1 TCP と UDP の違い

TCP/IP ネットワークでは UDP と TCP という特性の異なる通信が存在する。UDP では端末が意図した流量のトラフィックがそのままネットワークへ送出され、ネットワーク内のパケットロスには考慮していない。これに対し TCP では輻輳制御の機能により ACK が順調に返ってくるとウィンドウサイズを拡大し、帯域を有効に使おうとする。パケットロスが発生すると輻輳が発生したと判断し、ウィンドウサイズを縮小する。このようにウィンドウサイズが適切に調整され、ネットワークの更なる輻輳を防止する。ネットワーク上のトラフィックはまず送出された UDP パケットの合計により UDP が占めるトラフィック量が定まり、残りの帯域を複数の TCP セッションが分け合う。

3.2 PD-OLSR の経路選択指標

PD-OLSR では OLSR の基本部分そのまま用い、制御メッセージに自身の通信状態を表す情報を追加して送受信する。受信したノードはその情報を元に UDP 通信と TCP 通信それぞれの専用の RT を生成する。3.1 の考えに基づき、UDP 通信と TCP 通信の経路選択に用いる指標を別々に考える。UDP 通信の経路選択指標は UDP トラフィック(UDP traffic)、TCP 通信の経路選択指標は TCP セッション数(TCP session)とする。UDP トラフィックとは各ノードが検出するネットワーク上のキャリアの総量で、TCP セッション数は各ノードが検出する TCP セッション数の合計である。

3.3 ダイクストラ法

PD-OLSR では経路探索を行う際に、ダイクストラ法[13]を用いる。ダイクストラ法は、グラフ上の 2 頂点間の最短経路を効率的に求めるアルゴリズムである。経路の合計コスト基準に経路を求めることができ、カーナビゲーションシステムの経路探索や、鉄道の経路案内においても用いられている。

既存の OLSR では、ネットワークトポロジーの情報からホップ数を基準に最短経路を得ているが、ダイクストラ法を用い、経路コストとして各ノードが測定した通信状態を用いることで、通信状態を考慮し、さらにホップ数を増やし、混雑している部分を迂回した冗長経路を選択できる。

しかし、ダイクストラ法における経路コストは、リンクに対して 1 つのコストを設定する必要がある。ノードごとの通信状態、UDP 通信であれば UDP トラフィックは、各ノードが測定するものであるため、その情報からリンクに対するメトリックに変換する必要がある。

PD-OLSR では、ノードの通信状態からリンクメトリックに変換する際、双方向の通信経路で異なるメトリックに変換する。

ネットワーク内の各ノードは、一定時間ごとに制御メッセージを送信する。隣接ノードから制御メッセージを受信すると、自ノードが持っているネットワーク内の情報を更新する。また、ネットワーク内の各ノードの制御メッセージを送信するタイミングはノードごとに異なる。各ノードのトラフィック情報は、自ノードの検出するトラフィックを常に計測し、制御メッセージに載せて広告する。自ノード以外のノードのトラフィック情報は、制御メッセージを受信するたびに更新する。

ノードのトラフィックからリンクメトリックに変換する双方向で異なるメトリックとは、図 4.1 において、A、B をノードとし、a をノード A の、b をノード B の各ノードが計算した UDP トラフィックとすると、A→B 方向のリンクメトリックとして b を、B→A 方向のリンクメトリックとして a を用いる。

一方で、双方向で同じメトリックを用いる場合、両端ノードのトラフィックの合計値(または平均値)をリンクメトリックとして用いることが考えられる。図 3.1 の例であれば A→B、B→A 方向の両方で a+b となる。しかし、合計値を用いる場合、メトリックの更新には、リンクの両端ノードのトラフィック、図 3.1 の例であれば a と b 両方の情報が更新される必要がある。

双方向で異なるメトリックを用いる場合、リンクの両端ノードの内、宛先方向のノードのトラフィック、A→B 方向であれば b、B→A 方向であれば a の情報が更新されることで、メトリック

の更新が可能となる。つまり双方向で同じメトリックを用いる場合と比較して、更新に必要な情報を少なくすることができ、トラフィックの状況を経路探索により反映することができる。

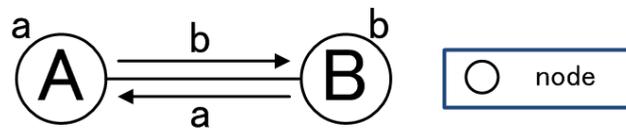


図 3.1 双方向メトリック

表 3.1 メトリック更新に必要な情報

メトリック	通信方向	リンクメトリック	更新に用いるノードのトラフィック	トラフィック状況の経路探索への反映しやすさ	経路探索能力
双方向で異なる	A→B	b	b	○	○
	B→A	a	a		
双方向で同じ	A→B, B→A	a+b	aおよびb	△	○

4 PD-OLSR の RT 生成

4.1 RT 生成手順

PD-OLSR では UDP 通信用と TCP 通信用の RT を別々に生成する。以下、図 5.1 および図 5.2 を用いて UDP 通信用の RT 生成を例にして、PD-OLSR の経路生成手順を示す。TCP 通信用の RT 生成についても、UDP 通信用 RT の生成手順とほぼ同じである。トラフィックの条件は図 3.1 の場合と同じである。図 4.2 のテーブルは、各ノードが計算した UDP traffic の情報である。ノード g からノード f への通信が行われているため、隣接ノードであるノード c, d, h, j, k では UDP traffic が検出されている。ここでは仮に検出されるトラフィック量を 4 としている。

4.2 経路探索用テーブル RMT

図 4.2 左のテーブルは UDP 通信用の RT を生成するためにノード b が持つ、新たに定義した経路計算用テーブル Route Metric Table(RMT)である。経路中のトラフィック量の合計をコストと呼ぶこととする。RMT は宛先ノード(Dest), 宛先へ経路のコスト(Cost), 経路のホップ数(hop), 経由するノード(hop1,hop2,...)の情報から構成される。

4.3 動作

PD-OLSR では、各ノードが計算した自身の通信状態を表す UDP traffic 情報を HELLO メッセージと TC メッセージに載せて隣接ノードへ広告する。各ノードは制御メッセージによって共有したネットワーク内のノードの隣接ノード情報と UDP traffic 情報から双方向メトリックに変換し、その情報を元にダイクストラ法によって経路探索を行う。その経路探索の結果を、RMT に保存する。RMT の情報から Dest, hop1, hop の情報が RT に保存される。

生成される経路は、各 Dest に対して経路の合計コストが最小のものとなる。例えば、図 4.2 でノード b からノード m へ UDP 通信が行われると、高トラフィックゾーンを避けた経路[b→e→i→l→n→m]が生成される。

TCP 通信用の RT 生成については、図 4.1 および図 4.2 と上記の説明において、UDP traffic

を TCP session に置き換えることで生成することができる。TCP 通信用の RT 生成の場合、RMT から TCP 通信用 RT に選ぶ経路は、各 Dest に対して最小 TCP Session となる経路が選択される。もし、TCP Session が同じであった場合は、UDP traffic の少ない経路が選択される。

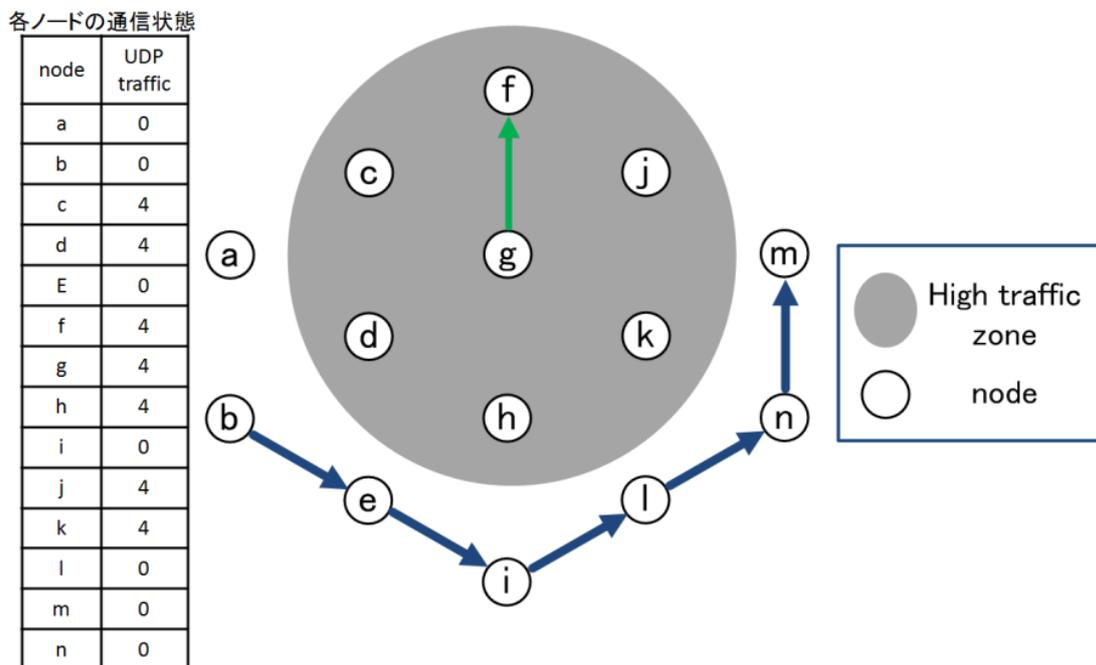


図 4.1 PD-OLSR で生成される経路例

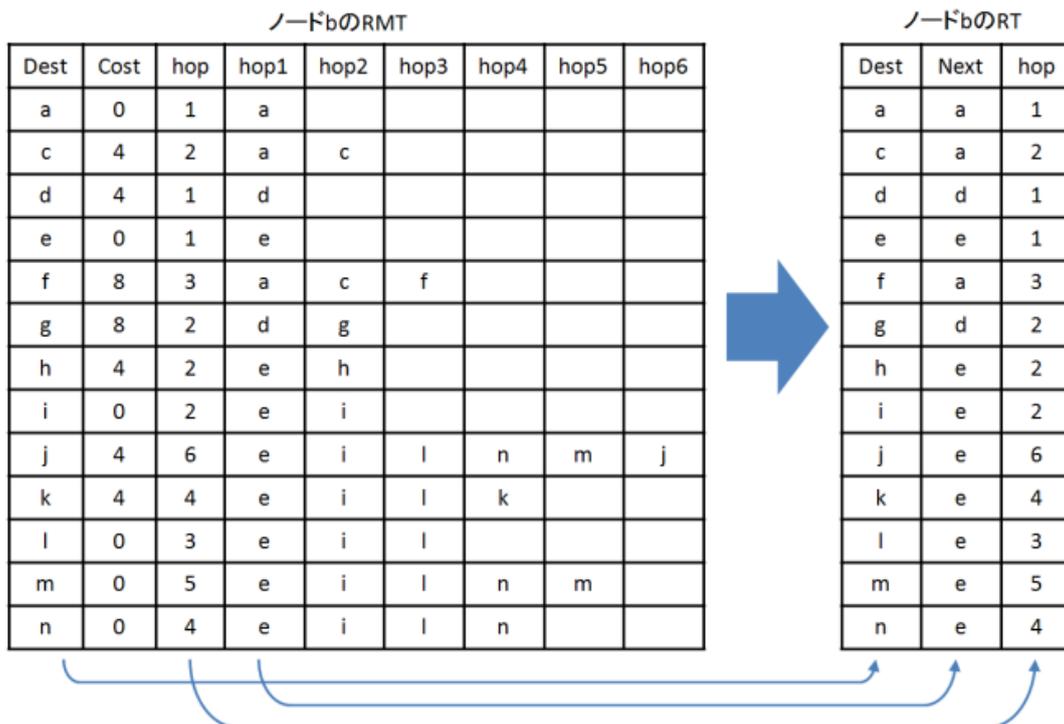


図 4.2 PD-OLSR の RT 生成

5 むすび

OLSR を拡張することで、TCP 用と UDP 用の RT を別々に生成し、ノードのトラフィック状況を考慮した経路選択が可能となるプロトコル PD-OLSR を提案した。RT を分けることで、TCP と UDP で通信の特性が異なるという点を経路探索に反映することができる。さらに、ダイクストラ法を用いることで、OLSR で選択される、ホップ数を基準にした最短経路に限らず、高いトラフィックのノードを回避した最適な経路選択が可能となる。ダイクストラ法で用いるリンクメトリックを双方向で異なるメトリックとすることで、メトリック更新に必要な情報を少なくし、トラフィック状況を経路選択により反映しやすくする拡張を検討した。

今後は、シミュレータに PD-OLSR を実装し、UDP 通信、TCP 通信、および UDP 通信と TCP 通信が混在する環境においてのシミュレーションを行い、PD-OLSR の効果を確認する。

参考文献

- [1] S. Corson: “Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations”, RFC 2501 (1999)
- [2] T. Clausen, Ed. : “Optimized Link State Routing Protocol (OLSR)”, RFC 3626 (2003)
- [3] D. Johnson : “The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4”, RFC 4728 (2007)
- [4] C. Perkins : “Ad hoc On-Demand Distance Vector (AODV) Routing”, RFC 3561 (2003)
- [5] R. Ogier: Topology Dissemination Based on Reverse-Path Forwarding (TBRPF), RFC3684, IETF (2004).
- [6] Zygmunt J. Haas, Marc R. Pearlman, Prince Samar: The Zone Routing Protocol (ZRP) for Ad Hoc Networks, Internet draft, IETF MANET Working Group (2002). Expiration: January, 2003.
- [7] Charles E. Perkins, Pravin Bhagwat: Highly Dynamic Destination-Sequenced Distance- Vector Routing (DSDV) for Mobile Computers, ACM SIGCOMM , Vol. 24, No. 4 (1994).
- [8] V.Park, S.Corson: Temporally-Ordered Routing Algorithm (TORA) Version 1 Functional Specification, Internet draft, IETF MANET Working Group (2001).
- [9] Douglas S. J. De Couto, Daniel Aguayo, Benjamin A. Chambers, Robert Morris: Performance of Multihop Wireless Networks: Shortest Path is Not Enough, ACM SIGCOMM Computer Communication Review, pp. 83–88 (2003).
- [10] Toh, C.-K.: Associativity-Based Routing for Ad-Hoc Mobile Networks, Wireless Personal Communications , Vol. 4, No. 2, pp. 103–139 (1997).
- [11] 高橋ひとみ, 斉藤匡人, 間 博人, 戸辺義人, 徳田英幸 : MANET における TCP スループット推定による経路選択機構の実環境評価, 情報処理学会論文誌, Vol. 46, No. 12, pp. 2857–2870 (2005).
- [12] Hipercom : <http://www.lix.polytechnique.fr/hipercom/>
- [13] A.V. Goldberg, Craing Silverstein: Inmpem- taintions of Dijkstra’s Algorithm Based on Muli-Level Buckets, (1995).
- [14] The Network Simulator - ns-2: <http://www.isi.edu/nsnam/ns/>.

謝辞

本研究を行うに当たり，多大なるご指導，ご鞭撻を頂きました渡邊晃教授に心より感謝いたします。また，有益なご助言，及びご検討を頂いた渡邊研究室の皆様にも深く感謝いたします。

6 附録

アドホックルーティングプロトコルの分類

MANET ではノードの移動によるリンク接続状態の急激・頻繁な変化への対応や、各ノードがその無線通信範囲外のノードと通信するために中継機能を持たせる必要がある。そのためには、MANET に特化したアドホックルーティングプロトコルが必要である。現在でも様々なアドホックルーティングプロトコルが開発されているが、全ての環境に適するプロトコルはいまだに開発されていない。今後もこのルーティングプロトコルの開発は大きな課題である。これまでに開発されたアドホックルーティングプロトコルは、表 2.1 に示すように 3つの型に分類することができる。これらは、その特徴が活かせる環境によって使い分けられる。

表 2.1 MANET のルーティングプロトコルの分類

分類	プロトコル例
Proactive 型	OLSR, DSDV, TBRPF
Reactive 型	AODV, DSR, TORA, ABR
Hybrid 型	ZRP

6.1 Proactive 型

Proactive 型のルーティングプロトコルでは、通信の要求が発生する前からルーティングテーブルを生成しておく方式で、通信の要求が発生すると即座に通信を開始できる。各ノードはルーティング情報を格納するためのテーブルを 1 つ以上持ち、ネットワークトポロジーの変化に応じてネットワーク全体に経路の更新情報を配送する。ルーティングに必要なテーブル数と、ネットワークの構造の変化を知らせるブロードキャスト方式の違いにより、いくつかのプロトコルが存在する。Proactive 型のルーティングプロトコルの特徴として、無通信時にも制御パケットが流れるため、消費電力は大きくなるが、通信を開始する際に遅延が発生しないことから、通信頻度の高いネットワークに適することが挙げられる。

6.2 Reactive 型

Reactive 型のルーティングプロトコルは、オンデマンド型のプロトコルである。すなわち、あるノードにおいて宛先ノードへの経路が必要になった時点で、ネットワーク内で経路探索プロセスを始動する。このプロセスは経路が見つかるか、利用可能なすべての経路パターンを試し終わると終了する。いったん経路が発見され、確立すると宛先へのアクセスができなくなるか経路が不要になるまでは、その経路が維持される。Reactive 型のルーティングプロトコルの特徴として、通信時に経路を決定するまでに遅延が発生してしまうが、オンデマンドで経路を構築するために、ノードの移動が頻繁なネットワークに適することが挙げられる。

6.3 Hybrid 型

Hybrid 型のルーティングプロトコルは、Proactive 型と Reactive 型の両方の長所を取り入れた複合プロトコルである。ネットワーク内を複数のゾーンに分割し、ゾーン内では Proactive 型のプロトコルを使用し、定期的な経路情報の更新はゾーン内のノードについてのみ行う。宛先ノードが送信元のゾーン外にある場合は Reactive 型のプロトコルを用いて経路を構築する。

Hybrid 型ではこのように両方の特徴を活かすことができるが、ノードが密集するような場合においてはゾーン内の管理すべきノードが多くなり、トポロジー管理が難しいという課題がある。