

MANETのための拡張FACEプロトコルを用いた経路決定方法とその評価

東京電機大学 理工学部 情報システム工学科

高橋 秀彰 卯木 宏幸 桧垣 博章

E-mail: {hide, unoki, hig}@higlab.k.dendai.ac.jp

モバイルアドホックネットワークにおけるオンデマンドルーティングプロトコルでは、送信元移動コンピュータから送信先移動コンピュータまで経路探索要求メッセージ *Rreq* を配送することが必要である。多くのルーティングプロトコルにおいて、フラッディングによる実現がなされている。これに対して、各移動コンピュータが隣接移動コンピュータの位置情報を用いることによって *Rreq* のコピーを必要としないFACEプロトコルが提案されているが、検出される経路のホップ数が大きくなる問題がある。本論文では、FACEプロトコルを拡張し、各部分平面の辺を時計回り、反時計回りに配送される *Rreq* のコピーを用いることにより、より短時間により短い経路を検出する。評価実験の結果、FACEプロトコルで検出される経路の約50%の長さの経路を検出できる。また、FACEプロトコルの約2倍のメッセージ数を要するものの、フラッディングと比較して十分に少ないメッセージ数で経路検出できることが明らかとなった。

Extended FACE Routing Protocol for Detection of Shorter Routes

Hideaki Takahashi, Hiroyuki Unoki and Hiroaki Higaki

Department of Computers and Systems Engineering

Tokyo Denki University

E-mail: {hide, unoki, hig}@higlab.k.dendai.ac.jp

In an on-demand routing protocol for mobile ad-hoc networks(MANETs), it is required for a route request message *Rreq* to transmit from a source mobile computer to a destination one. Here, flooding is widely used. In FACE protocol, no flooding is used and each mobile computer depends on location information of neighbour mobile computers to determine a mobile computer to which it forwards an *Rreq*. However, a hop count achieved by FACE protocol is much larger. In order to solve this problem, the authors propose an extended FACE protocol where copies of an *Rreq* are transmitted in clockwise and counterclockwise around each divided surface and a shorter transmission route is detected. In our simulation results, the proposed protocol detects 50% shorter route with twice messages of FACE protocol, however, the required messages are much less than flooding.

1 背景と目的

ノート型PCやPDA、自律移動ロボットなどの移動コンピュータからなるモバイルネットワーク環境が構築されている。ここでは、各移動コンピュータは、無線信号の送受信が可能であるが、互いに無線信号を直接交換できない移動コンピュータ間の通信は、他の移動コン

ピュータを介したマルチホップ配送によって実現する。このように、移動コンピュータ自身がメッセージのルーティングを行なうアドホックネットワークは、災害時の救助活動、会議やイベントのための一時的なネットワーク、原子力発電所の内部や高所、深海、宇宙などの環境での自律移動型ロボットからなるネットワーク、センサ

ネットワークなど、ネットワークトポロジの柔軟な変化が要求される環境、基地局の設置が困難あるいは不可能な環境に適用される。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は、以下の2種類に分類される。

- 各移動コンピュータの位置情報を用いないプロトコル
- 各移動コンピュータの位置情報を用いるプロトコル

前者として、DSR [6]、AODV [10]、TORA [9]、ABR [12]、LBSR [11] 等が提案されている。ここでは、経路探索要求メッセージ $Rreq$ を送信元移動コンピュータ M_s からフラッディング [2] することによって送信先移動コンピュータ M_d まで配送する。 $Rreq$ を隣接移動コンピュータから初めて受信した中継移動コンピュータが $Rreq$ のコピーを自身の無線信号到達範囲内に位置するすべての移動コンピュータへとブロードキャストする。 M_d へと到達した $Rreq$ を転送した移動コンピュータのアドレスを転送した順に列挙すると、 M_s から M_d までの経路となる。これらのプロトコルを用いると、各移動コンピュータの送信タイミングによる影響を受けるものの、原理的には M_s から M_d への最短経路を検出することが可能である。しかし、ネットワーク内のすべてのリンクを通じて $Rreq$ が配送され、すべての移動コンピュータにおいて受信した $Rreq$ の処理が必要となることから、大きな通信オーバーヘッドと処理オーバーヘッドを要するという問題がある。例えば、センサネットワークのように多数の移動コンピュータから構成されるアドホックネットワークでは、フラッディングを用いることによって、最終的に得られる経路上にない多数の移動コンピュータが $Rreq$ メッセージのコピーを配送することになる。

一方、後者においては、各移動コンピュータが GPS や固定基地局からのビーコン等を用いて自身の位置情報を取得可能であることを前提とし、この情報に基づいてメッセージの配送経路を決定する。位置情報を用いないプロトコルが移動コンピュータ間の隣接関係のみに依存する全域的なフラッディングを用いるのに対して、位置情報を用いることによってそのオーバーヘッドを削減することが可能となる。例えば LAR [8] では、送信元移動コンピュータ M_s と送信先移動コンピュータ M_d の座標 $p(M_s)$, $p(M_d)$ を用いることによって、 $Rreq$ のフラッディングによるオーバーヘッドを削減している。ここでは、 M_s が $p(M_d)$ を入手 (推測) することが可能であることを前提として、 $p(M_s)$ と $p(M_d)$ を頂点とする長方形領域内にある移動コンピュータのみを中継コンピュータとする。この領域内に存在する移動コンピュータは、 $Rreq$ を隣接移動コンピュータから初めて受信したならば、この $Rreq$ のコピーを自身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。一方、この領域に含まれていない移動コンピュータが $Rreq$ を隣接移動コンピュータから受信しても $Rreq$ のコピーをブロードキャストすることはせず、この $Rreq$ を破棄する。この方法を用いることによって、 $Rreq$ のブロードキャストメッセージ数を削減することが可能となる。しかし、LAR ではこの領域外を経由する経路を検出することができない。つまり、経路が存在するにも関わらず検出することができない場合がある。そこで、経路検出率を高めるための改良 (例えば [14] など) が提案されている。

アドホックネットワークにおける経路探索の第1ステップは、経路探索要求メッセージ $Rreq$ を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_d まで配送することである。これを全域的あるいは部分的なフラッディング (ブロードキャスト) で実現するのではなくユニキャストで実現する、すなわち、 $Rreq$ のコピーを生成することなく配送する手法として、一連の Greedy プロトコル [4]、GPSR プロトコル [7]、FACE プロトコル [1] が提案されている。ここでは、経路探索にフラッディングを用いない。各移動コンピュータは、自身および自身と直接通信可能な移動コンピュータ (以降、隣接移動コンピュータと呼ぶ) の位置情報を保持するものとする。この位置情報と送信先移動コンピュータの位置情報 (および FACE プロトコルにおいては送信元移動コンピュータの位置情報) を用いて1つの隣接移動コンピュータを選択し、 $Rreq$ をユニキャストで転送する。これを繰り返すことによって $Rreq$ を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_d まで配送する。このようなプロトコルを用いる場合、経路探索にフラッディングを用いず、検出される経路上にない移動コンピュータによる $Rreq$ の送受信とその処理が必ずしも要求されないことから、通信オーバーヘッド、処理オーバーヘッドは前者に比べて小さくなる。ただし、各移動コンピュータが $Rreq$ の転送先として最適な隣接移動コンピュータを選択するとは限らないため、最短経路を検出できるとは限らない。また、Greedy プロトコルや GPSR においては、中継移動コンピュータが $Rreq$ の転送先を決定できないデッドエンドに陥ることもある。

FACE プロトコルは、各移動コンピュータが隣接移動コンピュータ、送信元移動コンピュータ、送信先移動コンピュータの位置情報のみを取得可能という前提にも関わらず、経路が存在する場合には必ず経路が検出できるという点で、LAR、Greedy プロトコル、GPSR プロトコルに比べて優れたプロトコルであるといえる。また、 $Rreq$ をすべてユニキャストで配送することが可能

であり、 $Rreq$ のコピーを生成しないという点でも、通信オーバーヘッドが小さい経路検出を可能としている。しかし、検出した経路が一般的に長くなるという問題がある。これは同時に、 $Creq$ を送信先移動コンピュータまで配送するために要する時間が長い、すなわち、経路探索に要する時間が長く、アプリケーションメッセージの配送開始が遅延することを意味する。論文 [3] のシミュレーションでは、FACEプロトコルによるメッセージの配送経路は最短経路の（ホップ数で）約10倍の長さであるという結果を導いている。検出された経路のホップ数が大きい場合、特にこの経路を用いて多数のアプリケーションメッセージが配送される場合には、全体として大きなオーバーヘッドを要することになる。そこで本論文では、FACEプロトコルを改良し、経路検出率を100%としたまま、より短時間に、より短い経路を用いて $Rreq$ を配送することが可能な拡張FACEプロトコルを提案する。さらに、Gabrielグラフの適用によって拡大した検出経路長を縮小する手法を導入する。

2 従来手法

移動コンピュータ群 $M = \{M_1, \dots, M_m\}$ からなるアドホックネットワークにおいて、送信元移動コンピュータ $M_s \in M$ から送信先移動コンピュータ $M_d \in M$ への配送経路を探索する。各移動コンピュータ M_i は、GPS等によって自身の位置情報を取得している。また、メッセージを直接交換することが可能な移動コンピュータを隣接移動コンピュータと呼び、その集合を $Nei(M_i)$ で表す。このとき、 $M_j \in Nei(M_i)$ の位置情報も M_i は取得しているとする。さらに、 M_d の位置情報も取得、保持しているものとする。このとき、経路探索要求メッセージ $Rreq$ を M_s から M_d まで配送する。 $Rreq$ を配送した各移動コンピュータが自身のアドレスを $Rreq$ に記録することによって、経路 $R_{s \rightarrow d}$ を得ることができる。各移動コンピュータが位置情報を用い、フラッドイングを採用しないルーティングプロトコルは、各移動コンピュータが経路探索要求メッセージ $Rreq$ をどの隣接移動コンピュータにユニキャスト送信するか、その選択の方法により、以下の2種類に分類される。

- 距離等の指標を用い、その指標による評価値が最小あるいは最大となる隣接移動コンピュータを $Rreq$ の転送先として選択する方法。一連の Greedy プロトコル、GPSRプロトコル等がある。
- 経路が存在するならば、その経路が検出されることを保証した方法により、隣接移動コンピュータの1つを転送先として選択する方法。FACEプロトコル等がある。

どちらの方法も、 $Rreq$ のコピーを生成しないという点で経路探索に要する通信オーバーヘッドを縮小している。前者では、経路検出が成功する確率は必ずしも高くないが、検出された経路のホップ数は最小値に近い値となる。後者では、経路検出の成功が保証されているものの、その経路のホップ数は前者に比べて大きな値となることが多い。

2.1 Greedy プロトコル

論文 [4] で用いられている Greedy プロトコルでは、各移動コンピュータは、最も M_d に近い隣接移動コンピュータに $Rreq$ をユニキャストで転送する。このプロトコルを用いると、ホップ数が最小値に近い経路を検出することができる。ただし、図1に示すように、ある移動コンピュータ $M_i \neq M_d$ で、 M_i がすべての隣接移動コンピュータよりも M_d に近い場合、すなわち $\forall M_j \in Nei(M_i)$ について $|M_i M_d| < |M_j M_d|$ であるならば、 M_i は $Rreq$ の転送先移動コンピュータを選択することができず、 $Rreq$ の M_d への配送を実現することができない。つまり、Greedy プロトコルでは、経路検出確率（経路が存在するという条件のもとで経路が検出される確率）が100%とはならない。

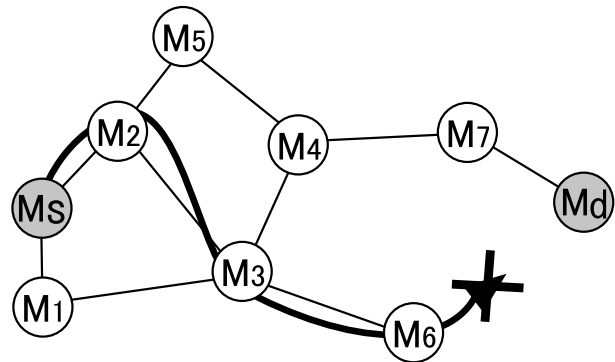


図1: Greedy プロトコル

[Greedy プロトコル]

(送信元コンピュータ M_s)

- (1) $Rreq.destination \leftarrow M_d$ とし、 $M_d \in Nei(M_s)$ であるならば、 $Rreq$ を M_d へ直接配送して終了する。
- (2) $\forall M_j \in Nei(M_s) \cup \{M_s\}$ について、 $|M_j M_d|$ が最小となる M_j に $Rreq$ をユニキャスト送信する。ただし、 $M_j = M_s$ であるならば、 M_s は $Rreq$ を破棄して終了する。

(移動コンピュータ M_i)

- (1) $Rreq.destination \neq M_i$ である $Rreq$ を受信する。
- (2) $\forall M_j \in Nei(M_i) \cup \{M_i\}$ について、 $|M_j M_d|$ が最小となる M_j に $Rreq$ をユニキャスト送信する。ただ

し、 $M_j = M_i$ であるならば M_i は $Rreq$ を破棄して終了する。

(送信先コンピュータ M_d)

- (1) $Rreq.destination = M_d$ である $Rreq$ を受信する。
□

2.2 FACE プロトコル

Greedy プロトコルでは、 M_d へ到達するための転送先移動コンピュータを決定できないデッドエンドが発生することがある。各移動コンピュータが自身と自身の隣接移動コンピュータの位置情報を用いて転送先移動コンピュータを決定する分散的手法であるにも関わらず、デッドエンドを発生しないプロトコルに FACE プロトコル [1] がある。ここで、各移動コンピュータ M_i を頂点、 $M_j \in Nei(M_i)$ について $M_i M_j$ を辺とする平面図形を考える。この図形によって平面は、複数の辺で囲まれた有限個の部分平面 F_1, \dots, F_f に分割される。ただし、この分割においては、2 辺の交わりが必ずしも頂点になっていない。あるいは、2 つの部分平面が頂点、辺以外の共通部分を持つ。2 辺の共通部分には必ず頂点となっている (移動コンピュータが存在する)、または、2 つの部分平面の共通部分は辺または頂点である、という条件を満たすために、以下の条件を満たす Gabriel グラフ [5] の辺のみを用いることとする。

[Gabriel グラフ]

頂点の集合を $M = \{M_1, \dots, M_m\}$ とするとき、以下の条件を満たす線分 $M_i M_j$ を辺とする図形を Gabriel グラフという。

- (1) $|M_i M_j| \leq h$ を満たす。
(2) $\forall M' \in M$ について、 M' は線分 $M_i M_j$ を直径とする円の外部にある。□

FACE プロトコルでは、 h は無線信号の到達距離である。また (2) の条件により、頂点ではない辺と辺との交わりは存在しなくなる。つまり、頂点を結ぶ 2 つの線分が交わる時、条件 (2) により、これらの線分の少なくとも 1 つは、辺としては含まない (ルーティングに使われない) ことになる。Gabriel グラフに含まれる辺のみを用いるために、各移動コンピュータ M_i は、隣接移動コンピュータ $Nei(M_i)$ の部分集合 $Sel_Nei(M_i) \subset Nei(M_i)$ を $Rreq$ の転送先移動コンピュータの候補とする。ここで、Gabriel グラフの条件 (2) により、「 $\forall M_j, \forall M_k \in Sel_Nei(M_i), M_k$ は $M_i M_j$ を直径とする円の内部に存在しない」という条件を満たす必要があるが、これは $Nei(M_i)$ に含まれるすべての移動コンピュータの位置情報を保持している M_i が計算可能である。

ここで、線分 $M_s M_d$ と交わり、以下の条件を満たす

部分平面列 $\langle F_1^{sd}, \dots, F_t^{sd} \rangle$ (ただし、 $M_s \in F_1^{sd}$ かつ $M_d \in F_t^{sd}$) を一意に定めることができる。

[部分平面列の満たす条件]

- (1) 点 $P_i \in M_s M_d \cap F_i^{sd}$ (線分 $M_s M_d$ と部分平面 F_i^{sd} との共通部分)、点 $P_j \in M_s M_d \cap F_j^{sd}$ ($1 \leq i, j \leq t$) について $i < j$ ならば $|M_s P_i| \leq |M_s P_j|$ 。
(2) $1 \leq i \leq t-1$ について、 $F_i^{sd} \cap F_{i+1}^{sd} \neq \emptyset$ 。□

頂点と辺の定義から、メッセージをこの図形の辺に沿って配送することが可能である。上記の条件により、以下の手順によって、経路探索要求メッセージ $Rreq$ を M_s から M_d へ配送することができる。

[FACE プロトコル (概略)]

- (1) M_s から F_1^{sd} の辺に沿って $Rreq$ を配送する。
(2) F_i^{sd} の辺に沿って $Rreq$ を配送しているとき、移動コンピュータ $\bar{M}_i \in F_i^{sd} \cap F_{i+1}^{sd}$ が $Rreq$ を受信したならば、以降 F_{i+1}^{sd} の辺に沿って $Rreq$ を配送する。
□

図 2 では、 $M = \{M_s, M_1, \dots, M_7, M_d\}$ からなるモバイルネットワークにおいて、移動コンピュータを頂点、直接通信可能な移動コンピュータ対 $\langle M_i, M_j \rangle$ に辺 $M_i M_j$ を対応させた図形から生成された Gabriel グラフにより、平面が 3 つの部分平面 F_1, F_2, F_3 に分割されている。ここで線分 $M_s M_d$ は M_s に近い方から F_1, F_2, F_3 の順に交わっている。そこで F_1 の辺、 F_2 の辺、 F_3 の辺の順に $Rreq$ を転送すれば、 $Rreq$ を M_s から M_d へ配送することができる。

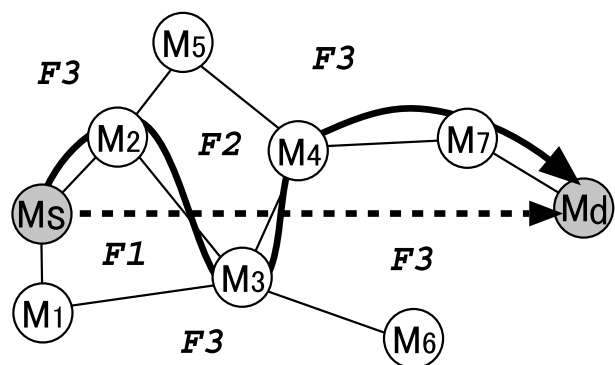


図 2: FACE プロトコル

線分 $M_s M_d$ は、 $F_i^{sd} \cap F_{i+1}^{sd}$ と 1 つの頂点 M_i を共有するか 1 つの辺 $M_i M_{i+1} \in F_i^{sd} \cap F_{i+1}^{sd}$ と交わるかのいずれかである。ただし、 M_s と M_d の座標を (x_s, y_s) 、 (x_d, y_d) とすると、 M_s と M_d を通る直線の式は $(x_d - x_s)(y - y_s) = (y_d - y_s)(x - x_s)$ である。

[Rreq を配送する部分平面の切替条件]

- (1) $M_i(x_i, y_i)$ が $Rreq$ を受信し、 $(x_d - x_s)(y_i - y_s) = (y_d - y_s)(x_i - x_s)$ 、 $(x_i - x_s)(x_i - x_d) \leq 0$ 、 $(y_i - y_s)(y_i - y_d) \leq 0$ を満たすとき、 M_i は線分 $M_s M_d$ 上にあり、 $Rreq$ をその辺に沿って配送する部分平面を切り替える。
- (2) $M_i(x_i, y_i)$ から $M_{i+1}(x_{i+1}, y_{i+1})$ に $Rreq$ が転送され、 $\{(x_d - x_s)(y_i - y_s) - (y_d - y_s)(x_i - x_s)\} \{(x_d - x_s)(y_{i+1} - y_s) - (y_d - y_s)(x_{i+1} - x_s)\} < 0$ および $x' = \{(x_d - x_s)(y_{i+1}x_i - y_i x_{i+1}) - (x_{i+1} - x_i)(y_d x_s - y_s x_d)\} / \{(x_d - x_s)(y_{i+1} - y_i) - (x_{i+1} - x_i)(y_d - y_s)\}$ 、 $y' = \{(y_d - y_s)(y_{i+1}x_i - y_i x_{i+1}) - (y_{i+1} - y_i)(y_d x_s - y_s x_d)\} / \{(x_d - x_s)(y_{i+1} - y_i) - (x_{i+1} - x_i)(y_d - y_s)\}$ として $(x' - x_d)(x' - x_s) \leq 0$ 、 $(y' - y_d)(y' - y_s) \leq 0$ を満たすとき、辺 $M_i M_{i+1}$ と線分 $M_s M_d$ は点 (x', y') で交わり、 M_{i+1} で $Rreq$ をその辺に沿って配送する部分平面を切り替える。
□

このプロトコルでは、線分 $M_s M_d$ と交わる部分平面を順にたどるという手法をとることから、経路が存在するならば、必ずそれを検出することが可能である。s ただし、経路長は Greedy プロトコルに比べて大きくなる。例えば、図 2 では、最初に M_s から M_2 に $Rreq$ が送信されているが、最初に M_1 に送信されているならば、 $M_1 \rightarrow M_3 \rightarrow M_2 \rightarrow M_5 \rightarrow M_4 \rightarrow M_3 \rightarrow M_6 \rightarrow M_3 \rightarrow M_1 \rightarrow M_s \rightarrow M_2 \rightarrow M_5 \rightarrow M_4 \rightarrow M_7 \rightarrow M_d$ という順序で $Rreq$ が配送される。

[FACE プロトコル]

(送信元コンピュータ M_s)

- (1) $Rreq.source \leftarrow M_s, Rreq.destination \leftarrow M_d$ とする。
- (2) $M_d \in Nei(M_s)$ であるならば、 $Rreq$ を M_d へ直接配送して終了する。
- (3) $\forall M_i \in Sel_Nei(M_s)$ について、 $0 \leq \angle M_d M_s M_i < 2\pi$ が最小 (または最大) となる M_i に $Rreq$ をユニキャスト送信する。ただし、 $Rreq.direction \leftarrow clockwise (Req.direction \leftarrow counterclockwise)$ とする。□

(隣接コンピュータ M_i)

- (1) M_j から $Rreq.destination \neq M_i$ である $Rreq$ を受信する。
- (2) $M_d \in Nei(M_i)$ であるならば、 $Rreq$ を M_d へ直接配送して終了する。
- (3) M_j から M_i への $Rreq$ のユニキャスト配送が部分

平面の切替条件を満たし、

- (3-1) $Rreq.direction = clockwise$ であるならば $Rreq.direction \leftarrow counterclockwise$ とする。
- (3-2) $Rreq.direction = counterclockwise$ であるならば $Rreq.direction \leftarrow clockwise$ とする。
- (4) 以下によって定まる M_k に $Rreq$ をユニキャスト送信する。
 - (4-1) $Rreq.destination = clockwise$ であるならば、 $0 \leq \angle M_j M_i M_k < 2\pi$ が最小となる $M_k \in Sel_Nei(M_i) - \{M_j\}$ を定める。
 - (4-2) $Rreq.destination = counterclockwise$ であるならば、 $0 \leq \angle M_j M_i M_k < 2\pi$ が最大となる $M_k \in Sel_Nei(M_i) - \{M_j\}$ を定める。□

(送信先コンピュータ M_d)

- (1) $Rreq.destination = M_d$ である $Rreq$ を受信する。
□

3 提案手法

FACE プロトコルでは、線分 $M_s M_d$ が貫く部分平面列を順に F_1, \dots, F_t とするとき、 F_{2i+1} の辺を反時計回りに、 F_{2i} の辺を時計回りに (あるいはその逆に) 巡るようにメッセージを配送することによって 100% の到達性を実現している。ところが、この方法を用いると、同一のトポロジに対して得られる配送経路の長さが M_s において選択した配送方向によって異なるという問題点がある。例えば、図 3 上側に示すように、最初に時計回

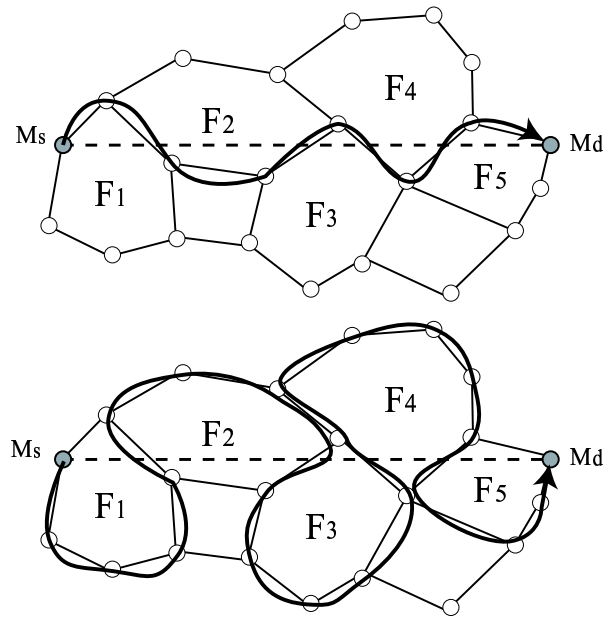


図 3: FACE プロトコルによる経路検出 (1)

りで始めた場合、FACE プロトコルを用いて検出され

る経路の長さは7ホップである。しかし、図3下側に示すように、最初に反時計回りで始めた場合、検出される経路の長さは23ホップである。このように M_s における配送方向の選択によって得られる経路のホップ数が大きく異なる結果となるが、隣接コンピュータの座標のみが取得可能であるという前提においては、経路長が小さくなる配送方向を M_s が決定することは不可能である。一方、図4では、 M_s において時計回りの配送方

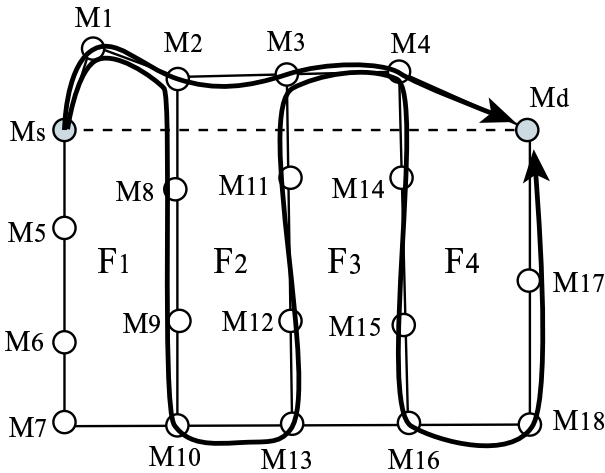


図4: FACEプロトコルによる経路検出(2)

向を選択した場合に検出される経路の長さも、反時計回りの配送方向を選択した場合に検出される経路の長さも16ホップであり、その差はない。しかし、 $M_s M_d$ が貫く F_1, F_2, F_3, F_4 の辺を順に巡る経路としては、 $M_s \rightarrow M_1 \rightarrow M_2 \rightarrow M_3 \rightarrow M_4 \rightarrow M_d$ という5ホップの経路が存在する。しかし、この経路が、 F_1, F_2, F_3, F_4 のすべての部分平面において時計回りの経路をとることによって得られることを、隣接移動コンピュータの位置情報を持つことのみが許されている各移動コンピュータが知ることはできない。

そこで本論文では、図5に示すように、送信元移動コンピュータ M_s およびFACEプロトコルに従って送信したメッセージがユニキャスト配送される辺と $M_s M_d$ とが交わる移動コンピュータ M_i が、次に辺に沿ってメッセージが配送される部分平面の辺上を時計回り、反時計回り、両方向にメッセージを送信することによって $Rreq$ のコピーがより短い経路を通して M_d に配送される拡張FACEプロトコルを提案する。ここで、 M_i は自身およびすべての隣接移動コンピュータの位置情報を保持していることから、メッセージに M_s と M_d の位置情報をピギーバックすることによって、「 $\exists M_j \in Sel_Nei(M_i), M_i M_j$ と $M_s M_d$ が交わる」となる M_j を決定することが可能である。そこで、 M_i の送信先移動コンピュータ M_j が

この条件を満たすとき、 M_i は M_j に $Rreq$ のコピーをユニキャスト送信するとともに、次に $M_s M_d$ が貫く部分平面の M_i の隣接コンピュータのうち M_j でないもの $\overline{M_j}$ にも $Rreq$ のコピーをユニキャスト送信する。 M_j と $\overline{M_j}$ にユニキャスト送信した $Rreq$ の配送方向を異なるものとすることによって、部分平面の辺上を時計回り、反時計回りに同時配送する。その結果、次の部分平面との共有点に先に到達した $Rreq$ のコピーがその部分平面上を配送されることになる。これによって、 M_s から M_d に至る線分 $M_s M_d$ が貫く各部分平面の各辺をそれぞれ1つ以上含む最短の経路に沿って $Rreq$ メッセージを配送することが可能となる。

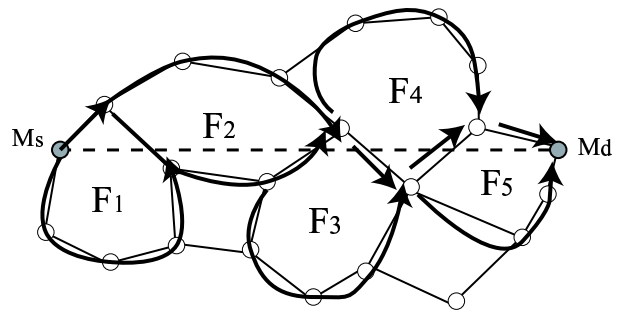


図5: 拡張FACEプロトコルによる経路検出

[拡張FACEプロトコル]

(送信元コンピュータ M_s)

- (1) $Rreq.source \leftarrow M_s, Rreq.destination \leftarrow M_d$ とする。
- (2) $M_d \in Nei(M_s)$ であるならば、 $Rreq$ を M_d へ直接配送して終了する。
- (3) $\forall M_i \in Sel_Nei(M_s)$ について、 $0 \leq \angle M_d M_s M_i < 2\pi$ が最小となる M_i に $Rreq$ をユニキャスト送信する。ただし、 $Rreq.direction \leftarrow clockwise$ とする。
- (4) $\forall M_i \in Sel_Nei(M_s)$ について、 $0 \leq \angle M_d M_s M_i < 2\pi$ が最大となる M_i に $Rreq$ をユニキャスト送信する。ただし、 $Rreq.destination \leftarrow counterclockwise$ とする。□

(隣接コンピュータ M_i)

- (1) M_j から $Rreq.destination \neq M_i$ である $Rreq$ を受信する。
- (2) $M_d \in Nei(M_i)$ であるならば、 $Rreq$ を M_d へ直接配送して終了する。
- (3) 同一の $Rreq$ をすでに送信済みであるならば、受信した $Rreq$ を破棄して終了する。
- (4) 部分平面の切り替え条件(1)を満足するならば、送信元コンピュータ M_s の処理(2)以降を行なって終了する。

- (5) M_j から M_i への $Rreq$ のユニキャスト配送が部分平面の切替条件を満たし、
- (5-1) $Rreq.direction = clockwise$ であるならば $Rreq.direction \leftarrow counterclockwise$ とする。
- (5-2) $Rreq.direction = counterclockwise$ であるならば $Rreq.direction \leftarrow clockwise$ とする。
- (6) 以下によって定まる M_k に $Rreq$ をユニキャスト送信する。
- (6-1) $Rreq.destination = clockwise$ であるならば、 $0 \leq \angle M_j M_i M_k < 2\pi$ が最小となる $M_k \in Sel_Nei(M_i) - \{M_j\}$ を定める。
- (6-2) $Rreq.destination = counterclockwise$ であるならば、 $0 \leq \angle M_j M_i M_k < 2\pi$ が最大となる $M_k \in Sel_Nei(M_i) - \{M_j\}$ を定める。
- (7) M_i から M_k への $Rreq$ のユニキャスト配送が部分平面の切替条件 (2) を満足するならば、
- (7-1) $Rreq.direction = clockwise$ であるならば $0 \leq \angle M_k M_i M_l < 2\pi$ が最小となる $M_l \in Sel_Nei(M_i) - \{M_k\}$ に $Rreq$ をユニキャストする。
- (7-2) $Rreq.direction = counterclockwise$ であるならば $0 \leq \angle M_k M_i M_l < 2\pi$ が最大となる $M_l \in Sel_Nei(M_i) - \{M_k\}$ を定める。 □

(送信先コンピュータ M_d)

- (1) $Rreq.destination = M_d$ である $Rreq$ を受信する。 □

拡張FACEプロトコルを用いることによって、 $Rreq$ は $M_s M_d$ が貫く部分平面を順に辿るという条件のもとで最短経路を得ることができる。このときに得られた経路上の移動コンピュータの列を $R = \langle M_0 = M_s, M_1, \dots, M_{l-1}, M_l = M_d \rangle$ とする。 R の検出において Gabriel グラフを用いていることから、 $\exists M_i \in R$ について $M_j (j > i + 1)$ が M_i の信号到達範囲内に存在することがある。これは、Gabriel グラフ作成時に線分 $M_i M_j$ が取り除かれたためである。そこで、 R を得た M_d では、このような M_i, M_j に対して、 $M_k (i < k < j)$ を R から取り除くことによって、 $R' \subset R$ なる短縮経路を構成することが可能である。 M_i と M_j の対の選択方法は複数存在するが、 M_d ではそのすべてを計算し、経路長が最短となる短縮経路 R^{min} を決定することができる。この R^{min} を $Rreq$ メッセージにピギーバックし、 R^{min} に従って $Rreq$ を配送することで、 M_s から M_d へのメッセージ配送をソースルーティングやネクストホップによって実現できる。

4 評価

提案プロトコルの有効性をシミュレーションによって評価する。評価環境は1辺200mの正方形領域に20台~100台の移動コンピュータを一樣分布に基づいてランダムに配置したものであり、すべての移動コンピュータの無線信号到達範囲は半径50mの円であるとする。送信元移動コンピュータ、送信先移動コンピュータの対をランダムに選択し、DSR、FACE、拡張FACEの3つのプロトコルを用いる場合の $Rreq$ メッセージ配送に要するメッセージ数と検出される経路の長さ(ホップ数)を測定した。また、拡張FACEについては、経路短縮によって得られた経路長(アプリケーションメッセージは、この経路を用いて配送される)についても示している。メッセージ数の測定結果を図6、図7に示す。FACE、拡張FACEにおいては、送信元移動コンピュータと送信先移動コンピュータを結ぶ線分が貫く部分平面群の辺上のみを $Rreq$ が配送されるため、すべてのリンク上を配送されるフラッディングに比べて格段に少ないメッセージ数しか必要としない。また、拡張FACEは、FACEの約2倍のメッセージを要するが、これは各部分平面の辺上を時計回り、反時計回りに同時に $Rreq$ を配送するためである。この方法を用いることにより、図8に示すように、拡張FACE($Rreq$)はFACEと比べて検出経路のホップ数を53.6%縮小している。すなわち、46.4%の時間で $Rreq$ の配送を終了している。また、経路短縮の手続きによって、37.5%の短縮が実現されている。この結果、アプリケーションメッセージの配送に用いられる経路の長さは、DSRによって検出される経路の長さの1.8%増にとどまっている。提案手法は、フラッディングを用いないことから、DSRに対してメッセージ数を大幅に削減していることを考慮すると、有効な手法であると結論づけることができる。

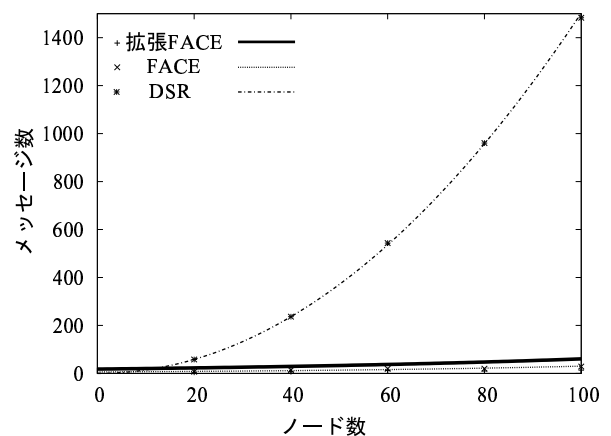


図6: メッセージ数の測定結果

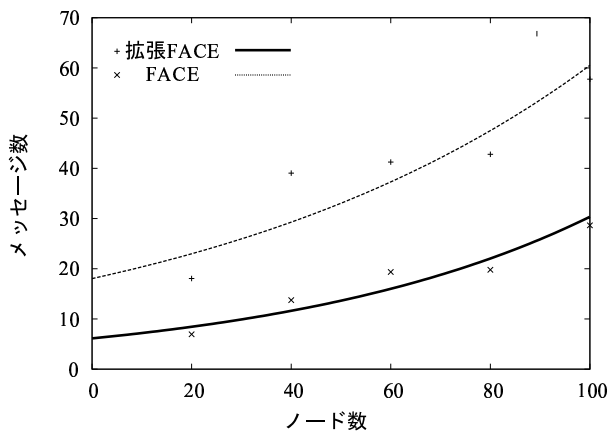


図 7: メッセージ数の測定結果 (DSR を除く)

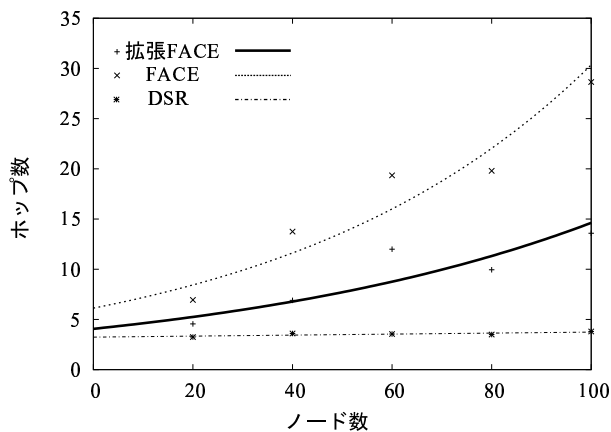


図 8: ホップ数の測定結果

5 まとめ

本論文では、FACEプロトコルを拡張し、100%の到達性を保ちつつ、より短い配送経路と検出できるプロトコルを設計し、その有効性をシミュレーション実験によって明らかにした。本プロトコルで用いた部分平面の辺上に限定した *Rreq* のコピーの配送は、FACEプロトコルの3次元空間への拡張であるCUBEプロトコル [13] における配送手法としても有効であることが見込まれる。この適用方法の検討が今後の課題である。

参考文献

- [1] Bose, P., Morin, P., Stojmenovic, I. and Urrutia, J., "Routing with Guaranteed Delivery in Ad Hoc Wireless Networks," Proceedings of the 3rd ACM International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications, pp. 48-55 (1999).
- [2] 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).

- [3] Datta, S., Stojmenovic, I. and Wu, J., "Internal node and shortcut based routing with guaranteed delivery in wireless networks," Proc. of IEEE International Conference on Distributed Computing and Systems, pp. 461-466 (2001).
- [4] Finn, G.G., "Routing and addressing problems in large metropolitan-scale internetworks," ISI Research Report ISU/RR, pp. 87-180 (1987).
- [5] Gabriel, K.R. and Sokal, R.R., "A new statistical approach to geographic variation analysis," Systematic Zoology, Vol.18, pp. 259-278 (1969).
- [6] Johnson, D.B., Maltz, D.A., Hu, Y.C., and Jetcheva, J.G., "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks," Internet Draft, draft-ietf-manet-dsr-04.txt (2000).
- [7] Karp, B. and Kung, H.T., "GPSR: Greedy Perimeter Stateless Routing for Wireless," Proceedings of The 6th Annual International Conference on Mobile Computing and Networking, pp. 243-254 (2000).
- [8] Ko, Y.B. and Vaidya, N.H., "Location-Aided Routing (LAR) in Mobile Ad Hoc Networks," Proceedings of the 4th International Conference on Mobile Computing and Networking, pp. 66-75 (1998).
- [9] Park, V. and Corson, S., "Temporally-Ordered Routing Algorithm (TORA) Version 1 Functional Specification," Internet Draft, draft-ietf-manet-tora-spec-04.txt (2001).
- [10] Perkins, C.E. and Royer, E.M., "Ad-hoc On-Demand Distance Vector Routing," Proceedings of IEEE Workshop on Mobile Computing Systems and Applications, pp. 99-100 (1999).
- [11] Sagawa, Y., Asano, T. and Higaki, H., "Loop-Based Source Routing Protocol for Mobile Ad-hoc Networks," Proceedings of the IASTED International Conference on Communications and Computer Networks (CCN2002), pp. 19-23 (2002).
- [12] Toh, C.K., "Associativity-based routing for ad-hoc mobile networks," Wireless Personal Communication Journal, vol.4, no. 2, pp. 103-139 (1997).
- [13] 高橋, 桧垣, "空間内に分布する移動コンピュータのためのルーティングプロトコル," 情処研報, vol.2002, No.49, pp. 19-24 (2002).
- [14] 水本, 山口, 廣森, 谷口, "MANETにおける位置情報マルチキャストルーティング MgCast の性能評価と応用に関する検討," 情処研報, Vol.2003, No.144, pp. 15-22 (2003).