

Mobile Ad Hoc Networks における蟻コロニー最適化と 状態保有フェロモンを用いた省電力を考慮する経路探索手法の提案

An Ant-based Routing Algorithm with Multi-phase Pheromone and Power-saving in Mobile Ad Hoc Networks

宮下 翔平 *

Shohei Miyashita

法政大学大学院 情報科学研究科 情報科学専攻

Email: shohei.miyashita.4j@stu.hosei.ac.jp

Abstract—This paper proposes an ant-based routing algorithm with multi-phase pheromone and power-saving (ARMPP) for enhancing the adaptability and stability of mobile ad hoc networks (MANETs). The routing algorithm is based on the ant colony optimization framework that is a kind of swarm intelligence and inspired from the ant's life. MANETs are dynamic, self-built and infrastructure-less networks. In MANETs, a link disruption resulting from the dynamic topology deteriorates the performance of network, and the increment of energy consumption brings reduction of the network lifetime because almost all nodes operate on a battery. The goal of this paper is to enhance the adaptability and stability in reacting to the mobility of nodes, that is, to variate the network topology timely and to maximize the lifetime of the network. We propose a two-stage construction method of pheromone and a new pheromone update method. In simulation experiments, we compare ARMPP with reference algorithms such as ad hoc on-demand distance vector (AODV) and destination-sequenced distance vector routing (DSDV). The simulation results show that our algorithm can achieve better performance than those algorithms in the measures of the adaptability and stability.

1. 序論

近年、スマートフォンやタブレット PC といった携帯可能な端末が普及し、簡単に所持できる時代にある。また携帯端末は、一般的に無線通信機能が標準搭載されている。このような時代背景もあり、MANETs は注目される研究テーマの一つとなっている。

MANETs とは、マルチホップ無線ネットワークである。携帯端末、以降ノードと呼ぶ、のみで構成可能であり、基地局といったインフラストラクチャが整備されていなくとも通信が可能となる。あるノードがあるノードへ通信をしたいが、電波が直接届かない場合であっても、中間に位置するノードへ転送を委譲し、バケツリレーのように転送することで通信が可能となる。応用先として、被災地や局地的な混雑など基地局を十分に利用できない環境下での利用が考えられる。しかしながら、MANETs は考慮すべき問題をいくつか抱えている。MANETs では携帯端末を取り扱うため、ノードは自由に移動し位置を変化させるため、動的なネットワークポロジとなる。無線通信によるノード間のリンクには有効距離があるため、動的なトポロジではリンク断が発生する。リンク断に起因し経路が消失するため、パケットの損失が多発する。また、一般的に携帯端末はバッテ

リーによって電力が供給されるため、バッテリーに残された電力は重要な資源である。電力を使い果たしたノードは通信が不可能となるため、ネットワークの構成ノード数が減り、リンク断が発生しやすくなる。加えて、被災地などでは電力供給を十分に行えない可能性も考えられるため、不要な電力の消費を避け、ノードの生存期間をなるべく引き伸ばすことが望ましい。以上から、本研究ではリンク断および電力消費について注目する。

トポロジの変異が、ノードの移動やバッテリー状況の変動によって引き起こされるため、MANETs における経路探索は動的最適化問題と捉えることができる。ここでの最適解とは、リンク断の起きにくい安定した、かつ消費電力の小さい経路である。また、より良い経路を発見するために、状況変化に対し順応性の高い手法を利用することが望ましい。本研究では、この問題を解決するために自然界から着想を得た群知能に着目し、動的な問題に有効な手法、蟻コロニー最適化 [1] を利用する。

2. 先行研究

2.1. 蟻コロニー最適化

蟻コロニー最適化 (ACO) は、アリの採餌行動に着想を得たメタヒューリスティクスの一つである。自然界において、生物は一般にフェロモンとして知られる揮発性物質を分泌することができ、アリもその一つである。この場合に分泌されるフェロモンは他個体を寄せ付ける性質を持ち、高濃度であるほどより多くのアリを引きつける。採餌行動の流れを次に述べる。アリは巣から様々な方向へ歩き回り、もし餌場を発見したならば自分が通った道を辿り巣へと戻る。この際に道標として上記のフェロモンを設置する。この道標をたどることによって、他個体が発見した経路から餌を獲得している。加えて、道標を利用する際に新たにフェロモンを追加設置し、フェロモン濃度を上昇させる。一方でフェロモンは揮発性のため時間経過によってその濃度は低くなっていく。もし経路が非効率である場合、フェロモンの増加量より減少量のほうが上回るためフェロモンとともに経路が消失する。逆に効率的である場合は、増加量が大きくなるため濃度が上昇し、良質な経路として保持される。より効率的な経路を発見した場合は上記のことが繰り返される。より良質な経路発見、濃度の上昇、アリの交通量増加、濃度のさらなる上昇という正のフィードバックによって、アリは非常に効率的な経路を発見している。

ACO では上記の行動をモデル化し、人工アリを用いて最適解を探索する。ある解を発見した際には、解の質に応じた人工フェロモンの蓄積が行われる。人工アリは蓄積された人工フェロモンの量を基に確率的に探索方向

* Supervisor: Prof. Yamin Li

を決定し、複数個体による解の同時探索を行う。ここで注目したいのは、上記の行動において探索を行う方向などを制御する指揮者が存在せずとも、アリという分権化されたエージェントが、フェロモンという局地的に付加された情報の交換から優良な経路を取得できる点である。一般的に、このような現象は *stigmergy* と呼ばれ、自己形成される構造は自然界に多く存在する。インフラストラクチャを考慮しない MANETs において、転送の流れを制御する調停役のような存在は期待できない。各ノードは隣接するノードとの交信によって局地的な情報交換を行い、収集した情報から転送先を決定する必要がある。このような状況は、上記のアリが置かれている状況との類似点が見受けられる。また MANETs は自己形成なネットワークであることから、ACO に代表される群知能の適応に優位性があると考え注目した。

2.2. 通信プロトコル

近年、MANETs への利用を考慮した数多くの通信プロトコルが提案されている。本項では既存の通信プロトコルについて触れる。まず通信プロトコルは主に3つの分類が可能であり、それぞれプロアクティブ型、リアクティブ型、ハイブリッド型と呼ばれている。以下にそれぞれの特徴を示す。

プロアクティブ型プロトコル: 各ノードが定期的に経路制御用のパケットを近隣ノードに送信し、ルーティングテーブルの情報を積極的に更新する点が特徴であり、テーブル駆動型であるとも言える。ネットワークのサイズが小さい、ノードの速度が小さい時に優位性がある。一方で、ノード速度が大きいなどテーブルの情報がすぐ古くなる状況下では不利になる。同タイプの通信プロトコルとして、Optimized Link State Routing (OLSR) [2] や DSDV [3] が挙げられる。

リアクティブ型プロトコル: 経路制御用のパケットを利用するのが、主に新規の宛先への通信要求の発生時または、既知の宛先への通信失敗時に限定する点の特徴である。そのため、要求発生時のオーバーヘッドが高くなる傾向にあるが、最新情報による経路の決定ができるほか、消費電力を小さくすることができる。同タイプの通信プロトコルとして、Dynamic Source Routing (DSR) [4] と AODV [5] が挙げられる。

ハイブリッド型プロトコル: ハイブリッド型では、プロアクティブ型とリアクティブ型の特徴を取り入れている。近傍ノードに対してはプロアクティブ型を採用し制御を行い、遠方ノードに対してはリアクティブ型による制御を行う。同タイプの通信プロトコルとして、Zone Routing Protocol [6] が挙げられる。

ACO を利用した既存の通信プロトコルとして、ハイブリッド型として分類できる AntHocNet [7] がある。AntHocNet では、データを送信する際にプロアクティブ型として動作する。送信元が積極的にアリに模した制御用パケットを利用し、フェロモンの更新や新たな経路発見を活性化させる。これは、AntHocNet がデータ送信の際にも確率的に経路を決定するためである。他にも、ARA [8]、AntNet [9] といった通信プロトコルが提案されているが、上記のプロトコルでは電力消費に関するパラメータは考慮されていない。近年では、WPAR [10] のような ACO ベースで、電力消費を考慮したプロトコルが提案されている。WPAR では、バッテリーに残された電力に関して考慮されている。本研究では、これに加え経路で消費される電力を考慮に加えた手法を提案する。

表 1. ノード i におけるフェロモンテーブル

	node ₁	node ₂	...	node _{m}
dest ₁	$\tau_{i11}^{\text{new}}, \tau_{i11}^{\text{old}}$	$\tau_{i21}^{\text{new}}, \tau_{i21}^{\text{old}}$...	$\tau_{im1}^{\text{new}}, \tau_{im1}^{\text{old}}$
dest ₂	$\tau_{i12}^{\text{new}}, \tau_{i12}^{\text{old}}$	$\tau_{i22}^{\text{new}}, \tau_{i22}^{\text{old}}$...	$\tau_{im2}^{\text{new}}, \tau_{im2}^{\text{old}}$
⋮	⋮	⋮	⋮	⋮
dest _{l}	$\tau_{i1l}^{\text{new}}, \tau_{i1l}^{\text{old}}$	$\tau_{i2l}^{\text{new}}, \tau_{i2l}^{\text{old}}$...	$\tau_{iml}^{\text{new}}, \tau_{iml}^{\text{old}}$

3. 提案手法

3.1. 状態保有フェロモン

自然界において、フェロモンは一般的に化学物質の混合物である。構成する化学物質および成分比によって種特異性を持たせ、特定の生物を引きつける役割を持つ。また、上記の通りフェロモンは揮発性の物質である。ここで、本研究ではフェロモンの蒸発により成分比にも影響が出ると仮定し、時間経過によって一定の濃度におけるフェロモンの誘発する能力が変化すると考える。この仮定を反映するため、ACO で利用する人工フェロモンに2つの状態を導入する。1つの状態は、アリから分泌されて間もない状態とし、New と呼称する。もう1つの状態は、分泌からある程度時間が経過した状態とし、Old と呼称する。また、フェロモンの量は、それぞれ New を τ^{new} 、Old を τ^{old} で示す。本研究は、New に対して状況変化に対応する速度を上昇させ、より動的問題に適応させることを期待する。また、Old に対して他のより良い経路を探索するための機会を残し、局地解への収束も防ぐことを期待する。 τ^{new} 、 τ^{old} の初期値はともに0であり、New は、パケットの送受信により隣接するノードとのリンクを検知し、かつ $\tau^{\text{new}} = 0$ のとき、定数 $C_1 > 0$ で再設定される。本論では $C_1 = 0.74$ 。Old は、New が時間経過によって遷移したときのみ蓄積され、その際に、 $\tau^{\text{old}} \leftarrow \tau^{\text{old}} + \tau^{\text{new}}$ かつ、 $\tau^{\text{new}} \leftarrow 0$ となる。ここで New が Old へ遷移するまでの時間を T として設定する。本論では $T \in [3.2, 5.5]$ 、単位を秒とし、一様乱数を用いて New 生成ごとに設定する。

フェロモンテーブルは、各隣接ノードとのリンクに蓄積されたフェロモン量を、各宛先ノードごと区別し行列の形式で格納される。本研究では、2つのフェロモン量を扱うため τ^{new} および τ^{old} を組として格納する。ここで表1にノード i におけるフェロモンテーブルの形式を示す。 d 行 j 列のフェロモン量を $\tau_{ijd}^{\text{new}}, \tau_{ijd}^{\text{old}}$ で表し、 j 番目の隣接ノードを node_j 、 d 番目の宛先ノードを dest_d で表している。 j 列すべてのフェロモン量が増えたときは node_j をテーブルから削除する。

次にフェロモン量を基にした確率算出について述べる。本研究では、まず2つのフェロモン量を混合し1つのフェロモン量 τ_{ijd} を以下の式より求める。

$$\tau_{ijd} = \alpha \cdot \tau_{ijd}^{\text{new}} + \tau_{ijd}^{\text{old}}. \quad (1)$$

ここで $\alpha > 0$ は引きつける強さのパラメータであり、本論では $\alpha = 1.75$ 。次に求めた τ_{ijd} を用いて、隣接ノード j を選択する確率 p_{ijd} を算出する式を以下に示す。

$$p_{ijd} = \begin{cases} \frac{(\tau_{ijd})^\beta}{\sum_{k \in \mathcal{N}_i} (\tau_{ikd})^\beta} & \text{if } j \in \mathcal{N}_i, \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

ここで \mathcal{N}_i は、ノード i における隣接ノード群から、人工アリがすでに探索したノードを除外した集合である。また、 β はフェロモン量を強調するためのパラメータであり、蓄積されたフェロモンが多いリンクの選択される確率を上昇させる。経路探索の際には、 $\beta = 1$ とする。

3.2. メトリクス

各ノードは人工アリが通過する際に、各隣接ノードへのリンクに蓄積されるフェロモン量および蒸発するフェロモン量を決定する。そのため、探索した経路の情報を基に各係数として算出する。本項ではフェロモン量の決定に利用するメトリクスについて述べる。

本研究では、4つのメトリクスを考慮する。1つ目について、送信元から宛先までに必要なホップ数を利用する。ホップ数は経路長であり、経路長が長いとリンク断に遭遇する可能性が上がる。ここで評価値を λ_i^{hop} で表し、以下の式から求める。

$$\lambda_i^{\text{hop}} = \frac{1}{J_s^i}. \quad (3)$$

ここで、 J_s^i は送信元ノード s からノード i までアリが移動したノードの数であり、ホップ数と同値である。

2つ目について、ノード i が隣接ノード j への送信を要求し、到着が確認されるまでの平均的な時間を利用し、便宜上 *Delay* と呼ぶ。Delay を評価するにあたり、ノード i における蓄積した Delay の平均値と、隣接ノード j への送信に対した *Delay* の平均値を保持し、前者を \bar{t}_i 、後者を \bar{t}_{ij} で表す。平均値は送信のたびに更新するため、以下の式から逐次的に平均値を求める。

$$\bar{t}_{k+1} = \bar{t}_k + \frac{t - \bar{t}_k}{n}. \quad (4)$$

ここで n は更新した回数、 t は計測された値、 \bar{t}_k は現在の平均値、 \bar{t}_{k+1} は更新された平均値である。 \bar{t}_i は 0、 \bar{t}_{ij} は近隣ノード i を検知した際に \bar{t}_i で初期化し、以降 \bar{t}_i はすべての送信、 \bar{t}_{ij} はノード j への送信時に更新が行われる。送信元ノード s からノード i までに蓄積された Delay を t_{si} とし、評価値 λ_i^{delay} を以下の式から求める。

$$\lambda_i^{\text{delay}} = \exp(-a_{\text{delay}} \cdot t_{si}^2). \quad (5)$$

ここで、 $a_{\text{delay}} > 0$ は制御用のパラメータであり、本論では $a_{\text{delay}} = 0.005$ 、 t_{si} の単位はミリ秒。

3つ目について、送信開始から到着が確認されるまでに消費された電力を利用する。これを評価するにあたり、Delay と同様に平均値を保持する。ノード i における、蓄積した電力消費の平均値と、隣接ノード j への電力消費の平均値をそれぞれ \overline{de}_i 、 \overline{de}_{ij} で表し、平均値の更新には式 4 を利用する。送信元ノード s からノード i に至るまでに蓄積された電力消費を de_{si} 、評価値を λ_i^{tx} で表し、以下に評価する式を示す。

$$\lambda_i^{\text{tx}} = 1 - \exp(-a_{\text{tx}} \cdot de_{si}). \quad (6)$$

ここで、 $a_{\text{tx}} > 0$ は制御用のパラメータであり、本論では $a_{\text{tx}} = 0.05$ 、 de_{si} の単位はミリジュール。

4つ目について、バッテリー残量に注目し、特定のノードにパケットの転送が集中し消費電力の偏りが発生することを軽減する。ここで現実世界において、各ノードは異なるバッテリー容量を持つことが予想されるため、バッテリー残量の評価は絶対値では行わずバッテリー残量とバッテリーの最大容量との比率を利用する。

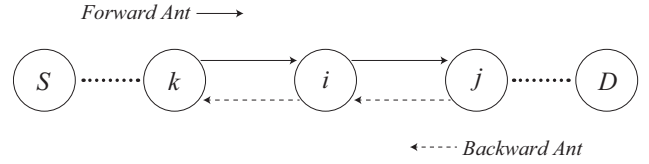


図 1. 送信元 S から送信先 D への経路モデル

ノード i のバッテリー残量を c_i 、バッテリーの最大容量を c_i^{max} とし、評価値 $\lambda_i^{\text{battery}}$ を以下の式より求める。

$$\lambda_i^{\text{battery}} = 1 - \left(\frac{c_i}{c_i^{\text{max}}}\right)^2. \quad (7)$$

ここで着目するのは、その経路における最小なバッテリー残量であり評価値を $\lambda_{\min}^{\text{battery}}$ で表す。

以上 4つのメトリクスから蓄積量および蒸発量に対する係数 Δ 、 η を以下の式から算出する。

$$\Delta = \sqrt{\lambda_i^{\text{hop}} \cdot \lambda_i^{\text{delay}}}, \quad (8)$$

$$\eta = \sqrt{\lambda_i^{\text{tx}} \cdot \lambda_{\min}^{\text{battery}}}. \quad (9)$$

求めた Δ 、 η から New を次式で更新する。

$$\tau^{\text{new}} \leftarrow \tau^{\text{new}} + \Delta \cdot \tau^{\text{new}} - \eta \cdot \tau^{\text{new}}. \quad (10)$$

次に、人工フェロモンの値をある一定期間ごとに小さくすることで実際のフェロモンの蒸発は再現される。本研究では時間経過による蒸発を Old に対し行う。以下に蒸発による Old の更新式を示す。

$$\tau^{\text{old}} \leftarrow (1 - \rho) \cdot \tau^{\text{old}}. \quad (11)$$

ここで $\rho \in (0, 1]$ は蒸発率であり、本論では $\rho = 0.3$ 、0.8 秒ごとに Old を更新する。

3.3. ルーティングステップ

本項にて、通信要求発生から終了までの手順について触れる。基本的な手順は経路探索およびデータパケット転送の 2 段階で構成される。以下に経路探索、データ配送およびエラー発生への対応について示す。

3.3.1. 経路探索段階. 経路探索において、制御用パケットを ACO における人工アリとして利用する。通信要求が発生し、宛先ノードまでの有効な経路がないと送信元ノードが判断するとき、送信元ノードは制御用パケットを宛先ノードに向けて送信する。ここで送信元ノードから宛先ノードに向かう制御用パケットを Forward Ant (FA) と呼称する。逆に、宛先ノード到着後に送信元ノードに向かう制御用パケットを Backward Ant (BA) と呼称する。ここで、図 1 に FA および BA が通過する経路および、送信元ノード S 、宛先ノード D 、中間ノード k 、 i 、 j のモデルを示し、以降このモデルを例にする。

FA はネットワーク内をランダムに歩き回り、リンクやノードを評価するための情報を収集する。その際に経路のループを防ぐため、通過したノードの識別子を FA のスタックに格納する。FA が宛先ノードに到着した場合、BA が生成され FA は探索を終了する。BA は生成時に FA の保持するスタックを引き継ぎ、FA が探索した経路を逆方向にトレースし送信元ノードへと転送される。BA においても FA と同様に中間ノード通過時に情報の更新を行う。FA は送信元ノードから、BA は宛先ノードから現在の中間ノードまでのホップ数を記憶する。同

Algorithm 1 Roulette wheel selection

Require: 対象ノード i , 宛先ノード d
式 2 から, 各近隣ノードの確率を算出.
一様乱数 $p \in (0, 1]$ を生成.
for all $j \in \mathcal{N}_i$ **do**
 $p \leftarrow p - p_{ijd}$
 if $p \leq 0$ **then**
 近隣ノード j を転送先に決定. 選択終了.
 end if
end for

様に FA は送信元ノードから, BA は宛先ノードからメトリクスの Delay および電力消費を累積し, 最小のバッテリー残量比を保持する. これらの情報は取得した経路およびフェロモンの評価に利用する. また, 本研究では経路探索時に宛先ノードへの経路と同時に送信元ノードへの経路の評価ならびにフェロモンの更新を行う. ノード i が BA を受信した場合, ノード j を経由するノード D までの経路が取得できる. よって, 中間ノードおよび送信元ノードが BA を受信したならば, 宛先ノードへの経路の評価が行える. ここでノード i が FA を受信した場合を考えると, ノード k を経由するノード S までの最新の経路が取得できる. よって BA と同様に, 中間ノードおよび宛先ノードが FA を受信した場合は, 送信元ノードへの経路の評価を行う. これは, 他ノードがノード S への通信を要求する場合を考慮するためである. 送信元ノードに至る経路の評価を同時に実行することで, フェロモンの更新を活性化し良質な経路への収束を早めることが狙いである.

MANETs において, 送信元ノードから宛先ノードまで複数の経路を構築できることが望ましい. これは, リンク断の発生による経路崩壊が発生しても他の経路で代用できる可能性があることに加え, 電力消費といった負担の分散を期待できるためである. このため, 送信元ノードは FA をブロードキャストし周囲のノードすべてに経路探索の開始を通知する. 中間ノードにおいては, FA の転送はブロードキャストまたはユニキャストにて行う. 中間ノードにおける既知の近隣ノード数が 2 未満の場合, FA をブロードキャストする. それ以外の場合ユニキャストを利用するが, 次の転送先は擬似コード 1 に示すルーレット方式にて決定される.

送信元ノードが通信要求してから初めて BA を取得した際, 他の BA を収集するため一定期間待機し, データ配送を開始する. これは, 後続の BA が電力面で優れた経路を保持する可能性を考慮し, 先行する BA が必ずしも最適な経路を保持すると断定できないためである. 以上の経路探索の流れを擬似コード 2 に示す.

3.3.2. データ配送段階. 本研究では, 経路探索にて取得した経路を利用しデータを配送する. ここで人工アリから取得した経路を一時的に格納するキャッシュを用意し, 経路を格納すると同時に経路の評価を行う. 経路の評価値 Fit は式 10 と類似の次式で求める. $Fit = (1 + \Delta - \tau)/2$. ここで Δ , τ は式 8, 9 から得る係数である. 送信元ノードは, 取得した Fit が最大の経路を利用してトレースにより宛先ノードまで配送する. 一方で, 以前に取得した経路は時間経過による劣化を考慮する必要がある. 本論では取得してから 4.5 秒経過した経路は削除する.

3.3.3. エラー対応. ノードがモビリティを持つ状況下で, リンク断に起因するエラーは頻繁に発生する. 発生したエラーを処理し, データ配送への影響を小さくする

Algorithm 2 Route Discovery

Require: 人工アリ ant , 対象ノード i , 転送元ノード k
if $ant = \text{Forward Ant}$ **then**
 if ノード $i = \text{送信元ノード}$ **then**
 ant へ情報付加, メトリクス初期化.
 FA をブロードキャストする.
 else if ノード $i = \text{宛先ノード}$ **then**
 ant のメトリクス更新. 式 10 より τ_{iks}^{new} 更新.
 ant より BA を生成, ant を捨てる.
 経路トレース開始, ノード k へ BA を転送.
 else if ノード $i = ant$ が未探索の中間ノード **then**
 ant のメトリクス更新. 式 10 より τ_{iks}^{new} 更新.
 if 既知の近隣ノード数が 2 つ未満 **then**
 ant をブロードキャストする.
 else
 ルーレット方式より, 次の転送先を決定.
 ant を決定した転送先へユニキャスト.
 end if
 end if
else if $ant = \text{Backward Ant}$ **then**
 if ノード $i = \text{送信元ノード}$ **then**
 ant のメトリクス更新. 式 10 より τ_{ijd}^{new} 更新.
 他の BA 収集のため, 一定期間だけ待機.
 待機時間終了後, データ配送開始.
 else if ノード $i = \text{中間ノード}$ **then**
 ant のメトリクス更新. 式 10 より τ_{ijd}^{new} 更新.
 ant のスタックから次の転送先を取得し転送.
 end if
end if

ことが重要である. そのためエラー発生を検知した際には, 制御用パケットにて他のノードへエラーを伝達する. またそれを Error Ant(EA) と呼称する. EA は生成元となる FA および BA が経由したノードをトレースしてエラーを伝達する. FA がエラーを検知した場合 EA は送信元ノードに向けてトレースを開始し, BA から生成された場合は宛先ノードに向けてエラーを伝達する. 本研究でのエラー検知は 2 段階で行われ, 1 段階目はリンクの崩壊, 2 段階目は経路の崩壊を考慮する.

1 段階目について, ノード i が FA をブロードキャストまたはノード j に対して行った転送の成功が確認できない場合, リンク断発生と判断しエラー対処を開始する. ノード i がノード k に BA を転送する際も同様である. ブロードキャストの成否は, 近隣ノードが送信する人工アリを取得し, 保有する情報の一致とスタックの 2 番目の要素が受信したノードかどうかで判断する.

2 段階目について, ノード i がノード j へ転送した FA が経路を形成しリンク断に遭遇しなかった場合, ノード i を BA が通過する. 一方で, 経路の形成に失敗した際は 2 つの状況が考えられる. 1 つは EA が通過する場合であり, もう 1 つは BA, EA が通過しない場合である. 例えばノード j への転送後, BA 到着前にノード i, j 間のリンクが崩壊した場合に発生する. よって 2 段階目では, 転送成功後ある一定期間 BA または EA が検知できない時エラー発生と判断する. ここで 1 つの経路において複数の EA 発生を防ぐため, EA がノード i を通過する際に, 当該ノードが EA を生成してないと判断できるときのみトレースを継続する.

ある宛先への経路は各ノードの各リンクに付加された人工フェロモンによって制御しているため, エラーが発生した場合は関連するフェロモン量を減少するペナル

表 2. シミュレーション設定

パラメータ名	設定値
実験回数	20
ノード数	50
ノード移動範囲	2,000 m × 2,000 m
ノード移動モデル	Random waypoint
シミュレーション時間	180 minutes
無線規格	IEEE 802.11n 2.4 GHz
データレート	MCS; index-0
バッテリー容量	21,312 J
総パケット数	10,000
パケットサイズ	1,024 Byte

ティを課すことによりエラーを反映させる。まずリンク断により通信不能となったノードに関するフェロモンをテーブルから削除する。次に EA を生成しエラーを伝達する。EA を受信したノードは、最近の EA 転送元である近隣ノードと当該の宛先ノードからフェロモンテーブルを参照し、フェロモン量を減少させる。例えば、ノード i からノード j へ FA の転送が失敗した場合、ノード i はフェロモンテーブルの j 列を破棄する。ノード k は、フェロモンテーブルの d 行 i 列のフェロモンを減少する。EA は送信元ノード S までトレースされる。

本研究ではフェロモンに与えるペナルティを、エラーを検知したノードから離れるにつれて小さくする。これは迂回路の構築を期待し、エラー発生場所から離れた地点での解の多様性を維持する狙いがある。ペナルティを変化させるにあたり、2段階目のエラー検知において BA を待機する時間に差をつける必要があるため、FA のホップ数が増加するごとに待機時間をある程度小さくする。EA 転送元ノード j 、宛先ノード d 、EA の生成元ノードからノード i までのホップ数 h_i 、ペナルティの初期値 $\epsilon_0 = 0.25$ とし、ノード i におけるペナルティ ϵ_i および ϵ_i によるフェロモンの更新を以下の式に示す。

$$\epsilon_i = (1 - \epsilon_0)^{h_i}, \quad (12)$$

$$(\tau_{ijd}^{\text{new}}, \tau_{ijd}^{\text{old}}) \leftarrow (1 - \epsilon_i) \cdot (\tau_{ijd}^{\text{new}}, \tau_{ijd}^{\text{old}}). \quad (13)$$

送信元ノードは BA を一切検知できない場合、現在経路探索可能な状況でない判断し、一定期間待機した上で再度経路探索を行う。

4. 実験結果

本研究は、ネットワークシミュレータ ns-3 [11] により評価し、ns-3.24.1 を使用する。主なシミュレーションの設定、条件を表 2 に示し、以下で詳しく触れる。

ns-3 ではバッテリー容量をジュールで与える必要がある。そのため 5.92Whr、3.8V の性能をもつバッテリーを想定し 21,312J を与える。無線通信による電力消費は ns-3.21 より導入された [12] の参照値を利用するが、ns-3 のデフォルト値は供給電力 3.0V で計算されているため、本研究では 3.8V にて再計算した値を利用する。

無線規格は IEEE 802.11n、周波数 2.4 GHz を利用し、Modulation and Coding Scheme (MCS) の 0 番目、帯域幅 20 MHz である。またノードの最大送信出力は $A_{tx} = 16.0206$ dBm、送信アンテナの利得 $G_t = 2.5$ dB、受信アンテナの利得 $G_r = 2.5$ dB とし、Nist Error Rate Model [13] を利用する。また、パス損失には対数距離

表 3. ノード速度および停止時間

ID	速度タイプ	最小速度	最大速度	停止時間
1	Walker	3 km/h	5 km/h	1 sec
2	Bicycle	12 km/h	18 km/h	1 sec
3	Car	30 km/h	50 km/h	1 sec

伝搬損失モデルを利用する。そのため参考距離 $d_0 = 1$ m におけるパス損失 L_0 dB を次式より求める。 $L_0 = 20 \log_{10}(4\pi d_0/\lambda) \approx 40.052$ dB。ただし周波数 $f = 2.4 \times 10^9$ Hz、真空中における光速 $c = 299,792,458$ m/s、波長 $\lambda = c/f$ 。ここでパス損失 L dB は次式より与えられる。 $L = L_0 + 10n \log_{10}(d/d_0)$ 。ただし d m はパス損失を考える距離、 n はパス損失指数であり本論では $n = 3$ 。ここで、本シミュレーションにおけるノードの通信可能距離を考える。熱雑音による入力雑音電力 N_t W は次式で与えられる。 $N_t = k_B T \Delta f \approx 8.00574 \times 10^{-14}$ W。ただしボルツマン定数 $k_B = 1.3803 \times 10^{-23}$ J/K、導体における絶対温度 $T = 290$ K、帯域幅 $\Delta f = 20 \times 10^6$ Hz であり、 T は ns-3 で設定されている。雑音指数 NF はデフォルト値の $NF = 7$ dB を利用し、ノイズ・フロア N_f は $N_f = 10 \log_{10}(N_t \times 10^3) + 7 \approx -93.966$ dBm で求まる。実際は、干渉によるノイズから最終的なノイズ N dBm を求めるが、ここでは考慮せず $N = N_f$ とする。最後に、信号対雑音比 SNR dB は次式より求められる。 $SNR = (A_{tx} + G_{tx} + G_{rx} - L) - N$ 。ここで [13] を参考にすると、MCS-0 では SNR がおよそ 3 dB を下回るとエラーが多発し通信が不可能となる。以上で求めた値および $SNR = 3$ を利用すると $d \approx 250$ m が求まるため、本シミュレーションでのノード間の通信可能距離は 250m 程度である。

ノードの移動には Random waypoint を利用する。これはノードの目的地座標、速度および到着後の停止時間が毎回ランダムで選択され移動するモデルである。本研究ではノードの速度及び停止時間を表 3 に示す範囲に限定し、移動速度ごとに評価を行う。速度は歩行者、自転車、車を想定しており [14] を参考にしている。

最後にトラフィックについて、2つのノード n_1 、 n_2 を設定し、ひとつの組として考える。 n_1 は n_2 へ、 n_2 は n_1 へとそれぞれリクエストを生成する。本論ではリクエストは固定レートで 2 秒毎に生成され、各ノードは 5,000 データパケット生成する。

提案手法は ns-3 に標準実装されている既存手法の OLSR、DSDV、DSR および AODV と比較する。シミュレーション結果を以下の図 2、3、4 に示す。図 2 ではデータパケットが宛先に到達する確率を示し、値が 1 に近いほど多くのパケットが届けられることを表す。ここで、特に速度タイプ 2、3 といったノードの速度が大きい場合に本研究での向上が見られた。この結果から、トポロジの変化に対する即応性の向上が達成できたと考えられる。大きい速度において性能向上できたひとつの要因に、経路探索の再実行が挙げられる。速度が大きい場合、1 回目段階で経路が発見できなくとも、2 回目実行の際にはトポロジが大きく変わっており、経路が発見できる可能性が高まるためである。図 3 では平均的な電力消費量を示している。ここで、平均的な電力消費量は次のように求めている。シミュレーション終了時の全バッテリー残量から平均的なバッテリー消費量を算出し、宛先に到達したデータパケット数で除算する。これにより、1 つのリクエストを成功させるのにかかる平均的な電力消費量として表している。これは、リクエスト

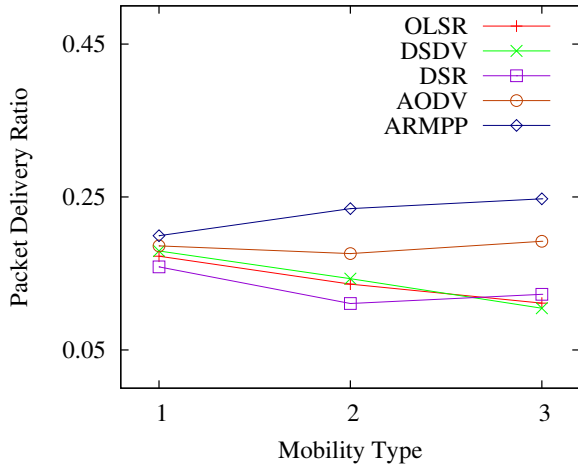


図 2. データパケットが宛先ノードへ到着する確率

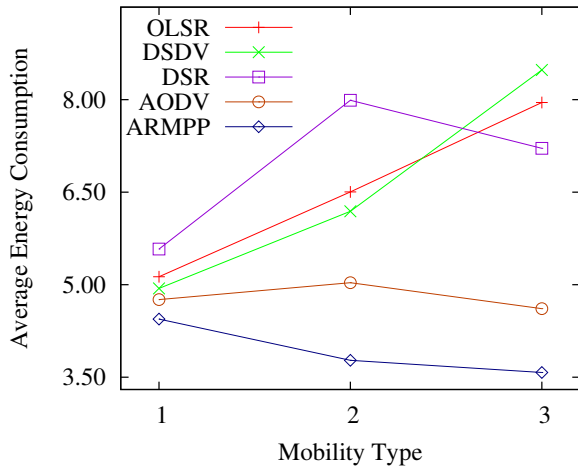


図 3. 1 回のリクエスト成功に対する平均電力消費量 [J]

を成功させるのにどの程度効率的に電力を利用しているかを比較するためである。図 3 から本研究にてより効率的に電力を利用をでき、目標とした電力消費の抑制が達成できたと考える。しかしながら、図 4 より本研究では電力消費量の偏りが OLSR より大きく電力消費の負担分散が達成できたとはいえにくい。これは経路の収束がある程度進むと、経路探索でのユニキャストの回数による差が現れてくるのがひとつの要因として考える。

5. 結び

本論にて電力消費の効率化、状況変化への即応性向上をする経路探索手法の提案を行った。実験結果から、本手法にて電力消費の効率化およびパケット到達率向上が可能であることを示した。しかしながら、電力消費の偏りが既存手法より大きいこと、ノードの速度が小さい時の優位性のさらなる向上が課題として残った。

展望として、New の強さを決定する α といった各パラメータをノードの状況に応じて動的に再設定するような機構を提案および実装し、より性能向上を狙うこと、また先行研究の ACO ベース・プロトコル、省電力指向で設計されたプロトコルを ns-3 へ実装し本研究と比較を行うことが挙げられる。

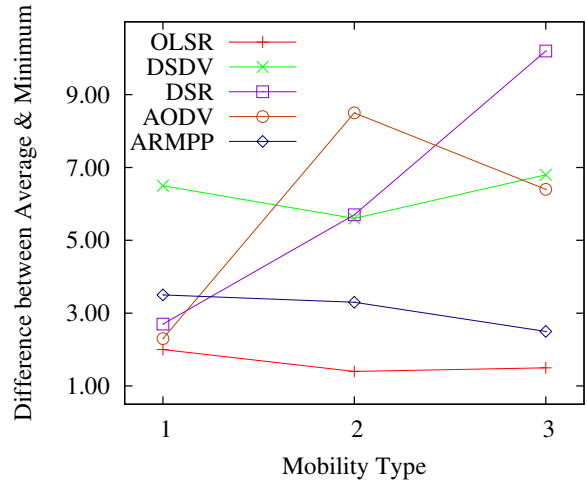


図 4. 平均バッテリー残量と最小バッテリー残量の差 [J]

参考文献

- [1] M. Dorigo and G. Di Caro, "Ant colony optimization: a new metaheuristic," in *Proceedings of the 1999 Congress on Evolutionary Computation*, vol. 2, 1999, pp. 1470–1477.
- [2] P. Jacquet, P. Mühlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot, "Optimized link state routing protocol for ad hoc networks," in *Multi Topic Conference, 2001. IEEE INMIC 2001. Technology for the 21st Century. Proceedings. IEEE International*, 2001, pp. 62–68.
- [3] C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (dsv) for mobile computers," *ACM SIGCOMM computer communication review*, vol. 24, no. 4, pp. 234–244, 1994.
- [4] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile computing*. Springer, 1996, pp. 153–181.
- [5] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *Second IEEE Workshop on Mobile Computing Systems and Applications*, Feb 1999, pp. 90–100.
- [6] Z. J. Haas, "A new routing protocol for the reconfigurable wireless networks," in *IEEE 6th International Conference on Universal Personal Communications Record*, vol. 2, 1997, pp. 562–566.
- [7] G. Di Caro, F. Ducatelle, L. M. Gambardella, and M. Dorigo, "Antnet: an adaptive nature-inspired algorithm for routing in mobile ad hoc networks," *European Transactions on Telecommunications*, vol. 16, no. 5, pp. 443–455, 2005.
- [8] M. Günes, U. Sorges, and I. Bouazizi, "Ara - the ant-colony based routing algorithm for manets," in *International Conference on Parallel Processing Workshops*, 2002, pp. 79–85.
- [9] G. Di Caro and M. Dorigo, "Antnet: A mobile agents approach to adaptive routing," Technical Report IRIDIA/97-12, IRIDIA, Université Libre de Bruxelles, Belgium, Tech. Rep., 1997.
- [10] G. S. Abkenar, A. Dana, and M. Shokouhifar, "Weighted probability ant-based routing (wpar) in mobile ad hoc networks," in *Electrical and Computer Engineering (CCECE), 24th Canadian Conference on*, 2011, pp. 826–831.
- [11] G. F. Riley and T. R. Henderson, "The ns-3 network simulator," in *Modeling and Tools for Network Simulation*. Springer, 2010, pp. 15–34.
- [12] D. Halperin, B. Greenstein, A. Sheth, and D. Wetherall, "Demystifying 802.11n power consumption," in *Proceedings of the 2010 international conference on Power aware computing and systems*. USENIX Association, 2010, pp. 1–5.
- [13] G. Pei and T. R. Henderson, "Validation of ofdm error rate model in ns-3," *Boeing Research Technology*, pp. 1–15, 2010.
- [14] S. Miyashita and Y. Li, "Armpp: An ant-based routing algorithm with multi-phase pheromone and power-saving in mobile ad hoc networks," in *Computing and Networking (CANDAR), 2015 Third International Symposium on*. IEEE, 2015, pp. 154–160.