

平成 27 年度

法政大学 修士論文

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

法政大学大学院 情報科学研究科 情報科学専攻

学籍番号 14T0013

宮下 翔平

Shohei Miyashita

E-mail: shohei.miyashita.4j@stu.hosei.ac.jp

指導教員: 李 亜民 教授

目次

図目次	ii
表目次	iii
Abstract	iv
概要	iv
1 まえがき	1
2 先行研究	2
2.1 蟻コロニー最適化	2
2.2 通信プロトコル	2
3 提案手法	3
3.1 状態保有フェロモン	3
3.2 メトリクス	3
3.3 ルーティングステップ	3
3.3.1 経路探索段階	3
3.3.2 データ配送段階	3
3.3.3 エラー対応	3
4 NS-3 へのプロトコル実装に関して	4
5 シミュレーション結果	5
5.1 環境設定	5
5.2 実験-1	5
5.3 実験-2	5
6 結び	6
謝辞	8
参考文献	9
付録	10
A.1 使用したシナリオファイル(一部抜粋)	10

図目次

図1	送信元 S から送信先 D への経路モデル	3
----	-------------------------------------	---

表目次

表1	シミュレーション設定	5
----	------------------	---

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-configured, 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 of 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.

概要

日本語あらすじ

1. まえがき

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デバイスが普及し，簡単に携帯端末を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おり，ユーザは気軽に通信を行うことができる．このような時代背景もあり，無線ネットワークの一種である Mobile Ad hoc Networks (MANETs) [1] は近年の注目される研究テーマの一つとなっている．

以下省略

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として知られている. また, Population ベースに分類されることが特徴である. これは, 複数の解を探索する個体 (エージェント) によって複数の解を同時に探索し, エージェント数の偏りによって解の優劣を判断するものである.

以下省略

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. 本節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つに分類が可能であり, 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.

以下省略

3. 提案手法

本節では、フェロモンの構造、フェロモン量による確率算出、フェロモンを格納する構造、フェロモンの評価関数および経路探索の手順について触れる。

以下省略

3.1. 状態保有フェロモン

以下省略

3.2. メトリクス

以下省略

3.3. ルーティングステップ

本項にて、提案手法における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、既存手法と同様、経路探索およびデータパケットの転送、以上2つの段階で構成される。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. 経路探索の段階では、制御用パケットを ACO における人工アリとして利用し、送信元ノードから宛先ノードまでの良質な経路の取得を目的とする。

中略

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 のモデルを示す。

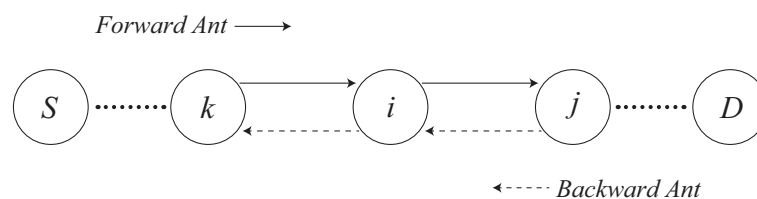


図 1: 送信元 S から送信先 D への経路モデル

3.3.2. データ配送段階. データ配送段階において、送信元ノードは宛先ノードにデータパケットを送信する。

以下省略

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。そのため、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ことが重要である。以下では、本研究におけるエラーの処理について述べる。

以下省略

4. NS-3 へのプロトコル実装に関して

本研究におけるシミュレーションでは、ネットワークシミュレータ ns-3 [2] を利用する。また、利用するバージョンは ns-3.24.1 である。

ns-3 において 32-bit のシリアライズおよびデシリアライズは符号なし整数によって行う関数は用意されているが、浮動小数点から直接実行される方法は用意されていない。ここで、*Sendcost* の合計値および最小の *Battery* には単精度浮動小数点を利用しているので工夫が必要である。本研究では IEEE 754 の単精度浮動小数点フォーマットを利用しているため、以下のリスト 1 を利用する。浮動小数点から整数に変換する際は `TransformCast<float, uint32_t>` と記述する。

リスト 1: 32-bit 浮動小数点と 32-bit 整数の変換

```
1 template<typename T, typename U>
2 inline U
3 TransformCast(T t)
4 {
5     U *up = (U *) &t;
6     return *up;
7 }
```

以下省略

5. シミュレーション結果

以下省略

5.1. 環境設定

まず、本シミュレーションにおける環境設定を以下に示す。

中略

以上のシミュレーション設定を表1にまとめる。移動速度は、歩行者、自転車、車を想定しており、[3]を参考にしている。

表 1: シミュレーション設定

パラメータ名	設定値
移動モデル	Random waypoint
シミュレーション時間	180 minutes
データパケットサイズ	1,024 Byte
無線規格	IEEE 802.11n 2.4GHz
データレート	MCS; index-0
バッテリー容量	21,312 J
パス損失	対数距離伝搬損失モデル

5.2. 実験-1

ここでは、ノード数 50, ノードの移動範囲を 2,000 m × 2,000 m としたシミュレーションを考える。比較的ノード密度が小さい状況である。

以下省略

5.3. 実験-2

ここでは、ノード数 30, ノードの移動範囲を 1,500 m × 1,000 m としたシミュレーションを考える。

以下省略

6. 結び

本研究では、時間経過によるバッテリー残量の変動や、ノードの移動といった状況変化への対応速度を向上を狙った MANETs の経路探索手法について言及した。2つのシミュレーション結果から、本研究で提案する手法において、電力の消費量を抑えること、またデータパケットを宛先に届ける確率の向上が可能であることがわかった。また、ノードの速度が上昇し、ネットワークトポロジの変化が多く発生する状況において、上記のデータパケットが宛先に到着する確率や電力消費量の減少がより向上できることがわかった。このことから、本研究の目標である、ノードやネットワークの状況変化への即応性向上が達成できた。

一方で、いくつかの課題がシミュレーション結果から判明した。まず、消費電力量の偏りに改善の余地があることである。これは、シミュレーション結果から既存手法の OLSR よりも消費電力量の偏りが大きいことが判明したためである。この消費電力量の偏りは、ある特定のノードに転送が偏っているとも言えロードバランスの働き具合にも関係している。パケットの転送が特定のノードに集中することで、ネットワークを構成するノード数の早期減少につながり、ネットワークの構成可能時間減少といった性能低下を招きかねないことが課題として挙げる理由である。加えて、ノードの速度が小さく、ノードの密度が大きい場合に性能の低下が見られることが2つ目のシミュレーションから判明した。5.3節でも述べたように、これは経路への収束が遅くなっているためだと考えられる。これを回避するために、パラメータの調整やそれに伴うアルゴリズムの改良が必要であり、課題である。

次に、本研究のシミュレーションにおいて比較のために用いた既存のプロトコルが ns-3 に標準で実装されている手法のみである点である。AODV など ns-3 に標準で実装されているプロトコルは非常に有名であるが、MANETs に向けたプロトコルは先行研究にもあるように、多く提案されている。他にも、AODV をベースに電力消費を考慮するプロトコルなどが提案されている。そのため、より多くのプロトコルとの比較が望まれる。特に本研究の提案手法は ACO ベース、また省電力指向であるため、先行研究の ACO ベース・プロトコル、また省電力指向で設計されたプロトコルとの比較を実施する必要がある。一方で、ns-3 は比較的新しいシミュレーションであることもあり、ns-3 へのプロトコル実装で公開されているものも、ns-2 と比べて少ない。例えば、先行研究として挙げた AntHocNet は ns-2 で実装されており、公開もされている。しかしながら、ns3 と ns2 との API は大きく異なるためそのままの適応は難しい。そのため、比較のためにプロトコルを実装するにはコストがかかる。比較対象としてプロトコルを選択し、ns-3 へ実装することが課題であり、展望でもある。

最後に、本シミュレーションにて本研究に与えたパラメータ、例えば New の強さを決定する α など、は複数回のシミュレーション結果から経験的に求めたものである。そのため、パラメータ決定が難しく、また決定するまでに時間がかかる場合がある。このパラメータ決定にかかるコストを小さくすることもまた課題であると考ええる。

以上の本研究、シミュレーションから得た課題を基に、今後の展望を最後に述べる。

1つ目は、各ノードが転送する制御用のパケットを減少させることである。これは、帯域の消耗、干渉によるノイズの影響を抑え、より効率的に電波を利用したいためである。ただし、FA といった探索のための人工アリを減少させてしまうと、より良い解の探索に支障をきたす可能性がある。ここで考えるのは、エラーの伝達である。本研究での提案手法では、EA は必ず送信元ノードもしくは宛先ノードまでトレースを行っていた。ここで、エラー発生場所から離れるにつれペナルティは小さくなるが、ここで EA のホップ数がある一定数を超えると、ほとんど影響がない程度までペナルティが小さくなってしまふ。そのため、EA を送信するだけのコストに見合わず、むしろ他のノードの通信を阻害するなど、悪

影響のほうが強くなると考えられる。そのため、EAのトレースを途中で破棄するような機構を実装することが展望である。

2つ目は、DSRで実装されているパケット救出のような機構を実装することである。本研究での提案手法では、経路探索の再実行は行わすが、データパケット転送時にエラー発生した場合、その場所からのデータパケット再送信は行っていない。これは、データパケットの宛先に到着する確率に影響するため、例えば、2番目のシミュレーションにて発生したデータパケットの到達確率の低下を抑止することが期待される。パケット救出のような機構を実装した場合、次のような流れがルーティング・ステップに組み込まれる。データパケット転送時にリンク断に遭遇する。リンク断を検知したノードはデータパケットを自分のパケットを格納するキューに挿入し、FAによる経路探索段階を開始する。

3つ目は、新たなパラメータ導入しメトリクスとして扱う、また動的に各パラメータの値を再設定するような機構を実装することである。まず新たなパラメータとは、各隣接ノードがそれぞれの程度安定しているか、および、近隣ノードの数を総合的に判別するパラメータである。例えば、近隣ノードとのリンク生成から経過した時間や、当該近隣ノードへパケットを送信した回数、パケット送受信によって最後に検知してからの経過時間を指標とする。ノードは移動しているが、ある隣接ノードと移動の方向が似通っている場合、リンクが比較的長く接続されることが考えられる。また、ただ利用しなかったため経過時間が長いという可能性を考慮し、近隣ノードを検知してからの利用回数と最後に当該近隣ノードを利用した時間から、信用性を判断する。また、近隣ノードが多ければ、リンク断に対応しやすいとして近隣ノードの数も考慮する。この指標によって、より安定している経路選択のために利用し、トポロジ変化より不安定な状況での性能向上を狙うことがひとつの展望である。次の動的に各パラメータの値を再設定するような機構とは、ノードの状況、例えば、周囲のノードのモビリティなど、によって設定したパラメータを再設定、更新してより適応度を向上させることを目指すためである。この機構によるパラメータ更新には周囲のノード状況を指標として扱う上記の新たなパラメータによって行えると考えている。本研究でのシミュレーションは、Newの強さを決定する α などのパラメータはノードの速度に関わらず同じ値にて行っている。しかしながら、パラメータ決定をする際に行った複数回のシミュレーションから、性能向上する際のパラメータにばらつきがあった。例えば、BAを収集するための待機時間が長くなると、消費電力の減少や、最小バッテリー残量の平均バッテリー残量との差の縮小といった向上が見られた点である。また、ノードの速度によってのばらつきも見られた。そのため、設定されたパラメータ値を、近隣ノードとのモビリティを評価できるようなパラメータで制御し、動的に再設定するような機構を考案することで性能向上が望める。これは、今後の研究対象である。

最後は、セキュリティに関する問題を考慮することである。本研究では、リンク断および、電力消費について着目したが、セキュリティも重要であることに変わりはない。人工アリのもつスタック改ざんによる経路の破壊や変更などの攻撃を受けることがあるほか、本研究で用いているプロミスキャス・モードによる転送の成否判定を利用しないほうが好ましい場合の対処法もまた考えられる。またセキュリティを考慮すべき理由として、将来的に、ウェアラブル・デバイス、センサーといった機器でアドホックネットワークを構築するサービスなどが登場する可能性もある。IoTといった今後の動向を考え、セキュリティを考慮させることもまた今後の課題である。

謝辞

本研究を進めるにあたり，ご指導を頂いた指導教員の李亜民教授に感謝致します。また，日常の議論を通じて多くの指摘を頂いた李研究室の同期ならび後輩に感謝します。

参考文献

- [1] E. M. Royer and C.-K. Toh, "A review of current routing protocols for ad hoc mobile wireless networks," *Personal Communications, IEEE*, vol. 6, no. 2, pp. 46–55, Apr 1999.
- [2] G. F. Riley and T. R. Henderson, "The ns-3 network simulator," in *Modeling and Tools for Network Simulation*. Springer, 2010, pp. 15–34.
- [3] 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.

付録

A.1. 使用したシナリオファイル (一部抜粋)

```
1 // Core
2 #include "ns3/core-module.h"
3 #include "ns3/network-module.h"
4 #include "ns3/applications-module.h"
5 #include "ns3/internet-module.h"
6 #include "ns3/wifi-module.h"
7 #include "ns3/mobility-module.h"
8 #include "ns3/energy-module.h"
9 #include "ns3/config-store-module.h"
10 #include "ns3/flow-monitor-module.h"
11 // Routing
12 #include "ns3/aodv-module.h"
13 #include "ns3/olsr-helper.h"
14 #include "ns3/dsdv-module.h"
15 #include "ns3/dsr-module.h"
16 #include "armpp/armpp-helper.h"
17 // Animation
18 #include "ns3/netanim-module.h"
19 // standard
20 #include <iostream>
21 #include <cmath>
22 #include <string>
23 #include <sys/stat.h>
24 #include <limits>
25 #include <stdlib.h>
26 #include <algorithm>
27
28 using namespace ns3;
29
30 NS_LOG_COMPONENT_DEFINE ("test-suite");
31
32 /*****
33 /////////////// scenario class interface
34 *****/
```

以下省略