

群知能を用いたインターネットルーティングの提案

2011SE025 後藤 朝陽 2011SE159 三福 健太

指導教員 後藤 邦夫

1 はじめに

近年、スマートフォンの普及により音楽や動画などのマルチメディアの利用頻度が増加している。増加する通信量の対策として、我々はパケットフロー最適化の観点からインターネットにおける群知能を用いたルーティングを提案する。本稿では蟻の群知能をモデルに考案された蟻コロニー最適化 (Ant Colony Optimization, 以降 ACO) 理論をルーティングの原理として採用している。

群知能を用いたネットワークルーティングシステムとして AntNet がある。AntNet とはネットワーク経路の最適化を目的としたものであり、モデル化したネットワーク問題に対し有効な実験や検証を行う [1]。先行研究では、IP データグラムネットワークを模倣した離散イベントシミュレータ (C++) を実現していた [2]。そこで我々は、実際の IP ネットワーク上で動作する ACO を用いたプログラムを作成した。Linux Ubuntu12.04LTS(32bit) 環境下でネットワークエミュレーションソフト「CORE」[3] を用いて仮想ネットワークを構築し実験した。ACO の既存研究の調査を三福健太が担当し、実現方法の調査とプログラミングを後藤朝陽が担当した。

2 アルゴリズムの概要

本節では群知能、ACO のモデルとなった自然界の蟻の習性と、ネットワークで利用されているアルゴリズムの概要を示す。

2.1 群知能

鳥や魚の群れは全体に指示を出す指導者が存在しなくても、個々が単純な振る舞いをしているだけで、全体を 1 つのシステムとして見たとき高度に最適化された動作をする。これを群知能と呼ぶ。群知能には本研究で用いる蟻をヒントにして考案された ACO だけでなく、蛍の点滅にヒントを得たホタルのアルゴリズムやミツバチの採餌行動に基づく人工蜂アルゴリズムなど存在する。群知能を応用した例として無人機の制御、惑星の地図作成や映像作品での鳥の群れの動きの描写などがある。

2.2 ACO

次に ACO について説明する。自然界に存在する蟻は餌と巣を往復する時に、最短の経路を確立することができる。図 1 で示したように、蟻は餌を巣に持って帰る途中に通った道にフェロモンと呼ばれる体内で生成した揮発性の高い液体を撒く。他の蟻はフェロモンを辿りながら餌と巣を行き来する。しかし、蟻は必ずしもフェロモンがある道を通るのではなく、フェロモンがない道を通る場合がある。このように、蟻は餌と巣を何度も往復することで複数の経路

を形成していく。前述したように、フェロモンは揮発性が高く時間の経過で蒸発していくため、フェロモンを撒く間隔が長くなると蒸発して薄くなる。逆に、餌と巣の距離が短ければフェロモンの濃度は濃くなる。このように、蟻は餌と巣を往復するだけで最短経路を形成できる。ACO では、蟻がフェロモンを辿って最短経路を形成する習性を利用して経路を最適化する。

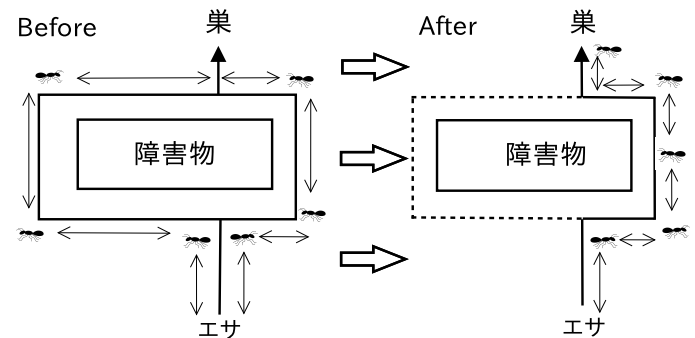


図 1 蟻の挙動

2.3 フェロモンの公式

ACO の中核を担うのがフェロモンであり、フェロモンの更新 (式 1)、追加量 (式 2)、蒸発 (式 3) を下記の式に示す。式 1 での τ_{ijd} はノード i からノード j を経由してノード d へ向かう場合のフェロモン量を表しており、また t は回数を表す。また、右辺の ρ はフェロモンの蒸発係数であり、この値の範囲は $0 \leq \rho \leq 1$ である。式 2 での $Q(k)$ はエージェント k の探索した巡回路の長さの逆数を表し、式 3 での T は時間を意味する。

$$\tau_{ijd}(t+1) \leftarrow (1 - \rho)\tau_{ijd}(t) + \Delta\tau_{ijd}(t) \quad (1)$$

$$\Delta\tau_{ijd}(t) = \sum_{k=1}^n Q(k) \quad (2)$$

$$\tau_{ijd}(T+1) \leftarrow \frac{\tau_{ijd}(T)}{1 + \Delta\tau_{ijd}(T)} \quad (3)$$

3 ACO ルーティングシステムの実現

ACO を応用したインターネットルーティングを実現するために考案したネットワークの情報収集の手法とシステムの概要を説明する。システムの構成を図 2 に示す。

3.1 エージェント

エージェントとは、ノードがネットワーク情報を収集するために送信するデータを指す。ACO ルーティングにおいてエージェントの役割は 2 つある。フェロモンの経路

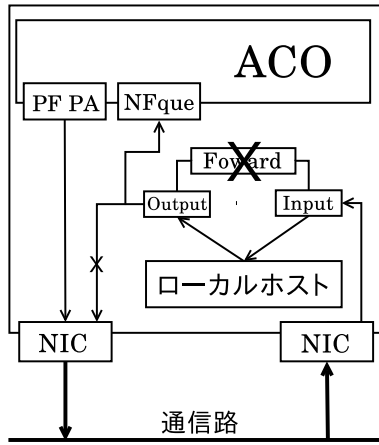


図2 システム構成

の発見とフェロモンの散布である．エージェントを目的地ノードまで進む往路エージェントと目的地ノードから送信元ノードまで遡る復路エージェントの二種類に分ける．以下に各エージェントの振る舞いを示す．

エージェントの転送ルール

● 往路エージェント

- 宛先へ直接送信できる場合を除いて，エージェントをフェロモンに基づいて転送し，中継点に到着する度にネットワーク情報を記録する．（フェロモンの初期値は全て0であり，フェロモンが同値だった場合は均一確率でランダムに経路が選択される）
- 同じ中継点を通った場合は，エージェントを破棄する．
- 目的地ノードに到着した場合，往路エージェントは復路エージェントになる．

● 復路エージェント

- 復路エージェントを受け取った場合，記録されている中継点を遡って転送する．このとき中継ノードのフェロモンテーブルを更新する．
- ネットワーク情報として記録されている時間を読み取り，フェロモンの計算に用いる．（時間を t として式 (1) に代入して計算する）

3.2 フェロモンテーブル

往路エージェント転送の際は，各ルータが保持しているフェロモンテーブルを参照する．フェロモンテーブルは復路エージェントによって書き換えられ，テーブルには任意のエージェントを受け取った場合にどの隣接ノードに転送するか決定する対応表が記述されている．

図3で表したような，ルータ S X Y D が配置されているネットワークモデルを考える．フェロモンテーブルの行には隣接ノードが配置され，列には目的地ノードが配置されている．ノード S からノード D へ向かう往路エージェントを考えた場合，隣接ノード X を選択する確率は 57%，隣

接ノード Y を選択する確率は 43% とする．またノード S からノード X へ向かう場合には隣接ノード X を選択する確率は 92%，隣接ノード Y を選択する確率は 8% とする．目的ノードを D とした場合，隣接ノード X を選択する確率が高いため，X が最もネクスト・ホップとして選ばれやすい．転送先選択は確率的に決定するので必ずしもフェロモン濃度が濃い方が選ばれるということはない．フェロモン付き経路からエージェントの送信先を選ぶ計算を式 4 に示す．左辺 $p_{ij}^k(t)$ はノード i に存在するエージェント k がノード j を選択する確率を表し，右辺の ω は重みを表し範囲は $0 \leq \omega \leq 1$ である．また， N はノード数， $\eta_{ij}(t)$ はノード i からノード j までの経過時間の逆数を意味する．

$$p_{ij}^k(t) = \frac{\omega * \tau_{ija}(t) + (1 - \omega)\eta_{ij}(t)}{\omega + (1 - \omega)(N - 1)} \quad (4)$$

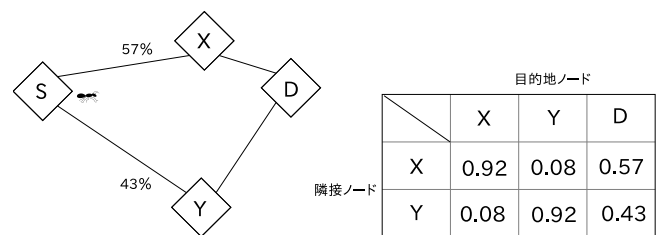


図3 ノード S でのルーティングテーブル例 (単位:%)

4 プログラムの概要

我々がプログラムに組み込んだ内容を示す．我々が実現した機能は，エージェント送受信転送機能，フェロモンテーブルの保持，追加，更新，フェロモンの計算，通常パケットの横取りである．

データリンク層で ACO プログラムを実現した．データリンク層で実現するメリットとしては，IP ヘッダを省くことでフレームに余裕ができ，IP ヘッダに関する計算 (ttl, checksum) をしなくてよい．そして，IP ヘッダに関する処理をなくすことで，全体的にコードを減らすことができた．経路の発見とフェロモン散布に応じてエージェントは，探索エージェントとメンテナンスエージェントの二種類を用意した．両方共タイムスタンプを格納するフィールドを確保したので，これを基に復路でフェロモンテーブルを更新する．フェロモンの計算式やタイミングは前節で説明した．ACO システムの各処理部分を図 4 に示す．

まず周辺情報を収集し，RIP のように隣から遠くまで順番に通信できるようにした．隣接ノードの情報は ping をブロードキャストして収集し，ARP キャッシュを蓄積する．ARP コマンドの出力結果を MACtable というデータテーブルに格納し，フレームリレーで通信する際に利用する．エージェント送受信転送機能はスレッドで別々に処理させる．Eagent S, Eagent F, Magent S, Magent F はそれぞれ探索エージェントの送信，転送，メンテナンスエージェントの送信，転送を担当する．evap はフェロモンの蒸発を定期的に計算するスレッドである．横取りしたパケッ

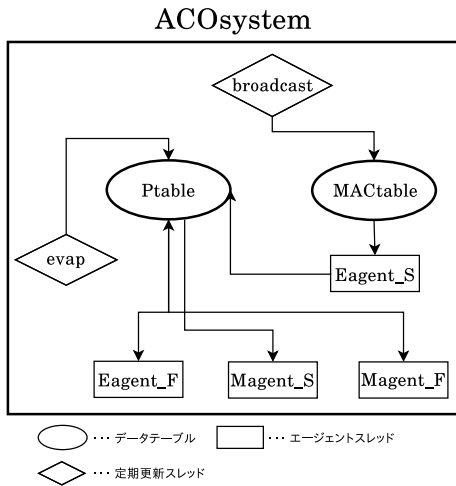


図 4 ACO システムの全体図

トは Ptable を参照して、従来のルーティングテーブルのように経路を決定する。

4.1 探索エージェント処理

エージェントを送信または転送する際に隣接ノードをランダムに抽選し、宛先を決めずに探索エージェントを送信することで未発見ノードを発見する。イーサネットのタイプ番号は 0x0801 にした。探索エージェントの構造体を 5 に示す。

送信

- ・ MACTable からランダムに送信先を決定する
- ・ 上記で決定したエントリを用いて、イーサネットヘッダを入力したのち送信する

転送

- ・ ホップ数を読み込み、規定数に到達していないことを確認する (規定数に到達していた場合、中継 IP アドレスを調べ返送する)
- ・ MACTable からランダムに送信先を決定する
- ・ 上記で決定したエントリを用いて、イーサネットヘッダを入力したのち送信する

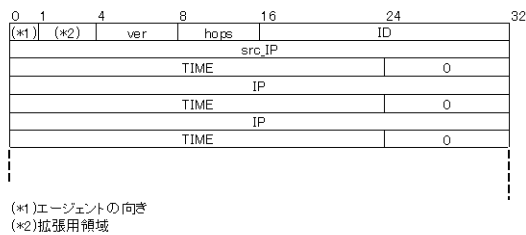


図 5 探索エージェントの構造体

4.2 メンテナンスエージェント処理

フェロモンテーブルに存在するエントリ全てにエージェントを送信する。中継点はフェロモンを元に決定し、何度も送信して最適経路を形成する。イーサネットのタイプ番号は 0x0802 にした。メンテナンスエージェントの構造体を 6 に示す。

送信

- ・ フェロモンが 0 のエントリが存在する場合 (該当エントリが存在する限り即座に実行する)
- ・ 送信先 IP アドレスをキーとして、MACTable からエントリを検索する
- ・ 上記で発見したエントリを用いて、イーサネットヘッダを入力したのち送信する
- ・ すべてのエントリが 0 以上のフェロモンを保持する場合 (3 秒に 1 度)
- ・ フェロモンテーブルに存在するエントリから、送信先 IP アドレスをキーとしてイーサネットヘッダを入力したのち送信する

転送

- ・ 送信先 IP アドレス (隣接アドレス) をキーとして、フェロモンテーブルから一致項目をリストアップする
- ・ 上記で発見したエントリのすべてのフェロモンを用いて、それぞれの選択確率を計算する
- ・ 乱数を生成し、選択確率を用いて宛先を決定する
- ・ 送信先 IP アドレスをキーとして、MACTable からエントリを検索する
- ・ 上記で発見したエントリを用いて、イーサネットヘッダを入力したのち送信する (宛先が自分だった場合、返送あるいは破棄する)

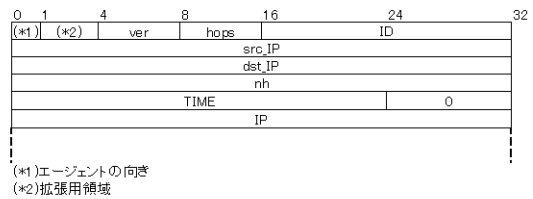


図 6 メンテナンスエージェントの構造体

4.3 ルーティング処理

通常パケットのルーティング手順を説明する。IPv4 のタイプ番号は 0x0800 である。

転送

- ・ nqueue で output チェインのパケットを横取りする
- ・ 送信先 IP アドレス (隣接アドレス) をキーとして、フェロモンテーブルから一致項目をリストアップする
- ・ 上記で発見したエントリの中から、フェロモン濃度

が一番高いエントリを宛先とする

- ・送信先 IP アドレスをキーとして, MACtable からエントリを検索する
- ・上記で発見したエントリを用いて, イーサネットヘッダを入力したのち送信する

フェロモンテーブル

フェロモンテーブルを構築する上で, エージェントの働きは重要な役割を持つ. エージェントがフェロモンテーブルに与える情報は, 時間とどのノードを見つけたのかの 2 点である. エージェントがフェロモンの公式に与える情報と, フェロモンが更新されるタイミングを表 1 に示す. NFqueue でパケットを横取りし, ユーザ空間でルーティングと送信パケットの生成を実現する.

	与える情報	更新するタイミング
追加量	時間	復路エージェント到着時
更新	追加量, フェロモン濃度	追加量計算後
蒸発	フェロモン濃度	10 秒間隔

表 1 復路エージェントがフェロモンに与える情報とタイミング

5 実験と評価

インターネット上のノードがエージェントを相互にやりとりできるか確認する為に, 図 7 のような簡単なネットワークで実験した. ノード n1 のフェロモンテーブルを以下に示す.

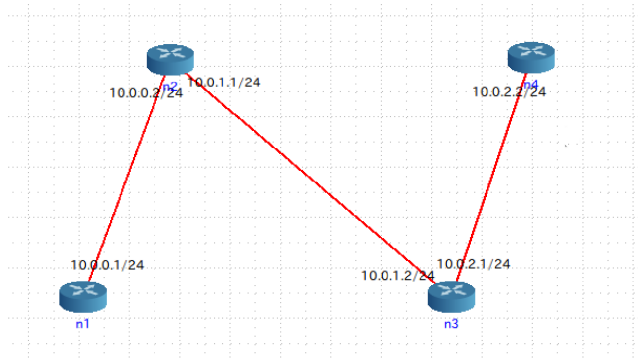


図 7 実験用ネットワーク

実行結果

```
[Eforward Thread]: returning agent dir[1] hops[3]
key not found, new entry will be added
Entry#1: flag[1], prev[0], ipaddr[10.0.0.2],
         pheromon[0], NH[10.0.0.2], IF[#9]
Entry#2: flag[0], prev[0], ipaddr[10.0.1.2],
         pheromon[0], NH[10.0.0.2], IF[#9]
Entry#3: flag[0], prev[0], ipaddr[10.0.2.2],
         pheromon[0], NH[10.0.0.2], IF[#9]
```

実行結果を確認してみるとネットワーク上に存在する 3 つのノードが発見されている. 戻ってきたエージェントの dir が 1 となっているが, これは復路エージェントを示す数値である. ノードは 3 つあるので, ホップ数が 3 になっているのは設計通りに動作している証拠である. flag は宛先が隣接ノードか否かを表している. prev は選択確率で計算に用いる前回コストの逆数である. 図を見て分かる通り, 経路は一本道なので, ノード n1 はインターフェースを 1 つしか持っていない. すなわち, ノード n1 が保持する最大エントリは NH とノード数の積で 3 となる. 末端以外のノードを宛先としたフェロモンは 0 になっているが, メンテナンスエージェントを送信してフェロモンを付加すれば, フェロモンによる確率的な経路決定が可能となる. 任意のネットワークで ACO が動作したとき, フェロモン濃度は経路のパラメータと相関性があることを確認した. (実験結果は本稿に示した)

6 おわりに

おわりに我々の研究活動の成果について言及する. 我々の研究目的は新しいインターネットルーティングの提案であり, ACO 理論に注目しルーティングとして機能させるために各情報を収集, 整理した. そして, フェロモンのデータ収集を目的として ACO システムのプログラムを作成した. 上記の過程で発生した課題を下記に示す.

- システムが不安定
- フェロモンの精度が低い

1 つ目の「システムが不安定」とは, ノード数が多くなるとシステムが停止してしまうことがあり, 実際にルーティングシステムとして採用するには安定性に欠けるということである. 2 つ目の「フェロモンの精度が低い」とは, ネットワーク回線の性能差が, フェロモン濃度あまり反映されないということである. つまり, ネットワーク回線の微小な差をフェロモン濃度に強く反映させるには, 公式の重みを調整しなければならない.

参考文献

- [1] Baras, J. and Mehta, H.: A Probabilistic Emergent Routing Algorithm for Mobile Ad Hoc Networks, *Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, pp. 20–24 (2003).
- [2] Di Caro, G. A. and Dorigo, M.: Two Ant Colony Algorithms for Best-Effort Routing in Datagram Networks, *Proceedings of the Tenth IASTED International Conference on Parallel and Distributed Computing and Systems (PDCS'98)*, IASTED/ACTA Press, pp. 541–546.
- [3] U.S. Naval Research Laboratory Networks and Communication Systems Branch: Common Open Research Emulator web page (accessed: Jan. 2015). <http://www.nrl.navy.mil/itd/ncs/products/core>.