

ペトリネットを用いたルーティングプロトコル AODV と 移動端末の統合モデルの開発と評価

大河原 優一 指導教員：辻 孝吉

1 はじめに

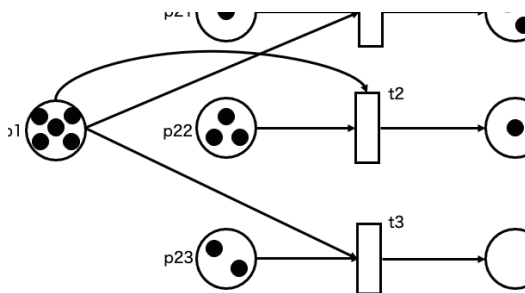
日本のネットワークインフラを支えている通信会社の基地局が災害の影響で故障してしまった場合、被災地や避難先での通信手段が奪われてしまう。これを解決するために基地局を必要とせず、通信機能を有する端末のみによって相互の通信を実現する MANET(Mobile Ad-hoc Network)が注目されている。MANET のルーティングプロトコルの一つである AODV(Ad-hoc On-demand Distance Vector)は通信要求が起こってから経路を確立する Reactive 型のプロトコルで、一対一の通信に優れている[1]。AODV は IETF の"MANET WG"によって仕様書の検討がされているが、実際に運用してみると不具合が生じてしまう場合があるため前段階で動作を検証することが求められる[2]。代表的なネットワークシミュレータ(NS2 など)は性能の評価を適しているのに対して、プロトコルのシステムの動きを可視的にも、また論理的にも検証することには向いていない。そこで筆者の先行研究では、多くの事象が並行して動くシステムのモデル化に適しているペトリネットを用いて AODV のシステムモデルを開発・評価した。しかし、以下に挙げられる課題点が残った。

1. 端末の移動を正確にモデル化できていない
2. 送信元・送信先端末が固定
3. AODV における未実装の機能

そこで、本研究では先行研究で残った課題解決を行うこと、端末の移動モデルと統合して扱うことのできる AODV のシステムモデルの改良を行う。

2 ペトリネット

MANET のような通信プロトコルは離散的な事象が互いに関係しあって動作する並行離散事象システムと考えることができる。このような離散的な事象をモデル化する手法としてペトリネットを用いた解析手法がある。ペトリネットは状態を プレース(○)、事象をトランジション(□)としてシステム構造を 可視的に表現するツールであり、トークン(●)を使用することによりシステムの並行的でダイナミックな事象をシミュレートするとともに論理的に解析することができる[3]。またトークンに色を持たせ識別させたものをカラーペトリネットという。例を図



1に示す。

3 AODV

AODV は Reactive 型のルーティングプロトコルである。送信元のノードが目的地である送信先のノードと通信を行うために直接ないし中継機を用いて通信するためのルートを構築する。経路の構築方法を簡単に説明する。

まず送信元端末から RREQ(Route Request)メッセージがネットワークに拡散する。RREQ メッセージを受け取った端末は自身の経路表を更新しながら送信先端末に届くまで再度拡散される。送信先端末が RREQ メッセージを受信すると、送信元端末へ RREP(Route Reply)メッセージを返信する。RREP メッセージを送信元端末が受信することで経路が構築される仕組みになっている。

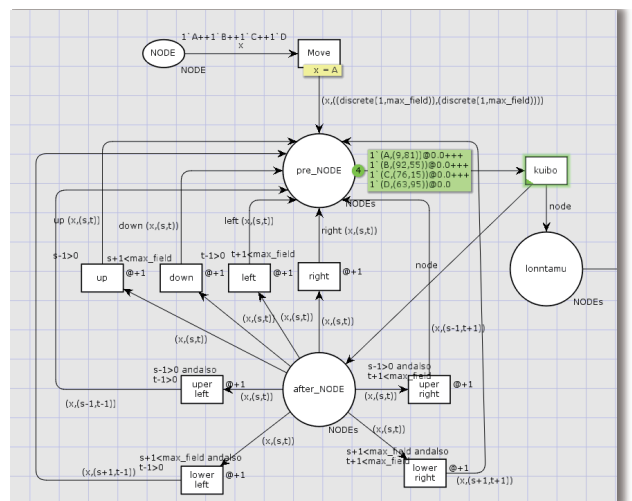
AODV における経路表はデータパケットを送信する際に利用する。また経路表はシーケンス番号を利用することで最新に保たれている。

また、リンク切断などによって経路が切断された際には RERR(Route Error)メッセージが生成される。このメッセージを受け取った場合は経路表が無効化され、再度経路探索をする必要がある。

4 ペトリネットによるモデル化

本研究ではペトリネットを用いたモデル化を行い、シミュレーションを行う際に"CPN Tools"を使用する。端末の移動モデルと AODV の制御メッセージシステムをペトリネットにモデル化する。例として端末の移動モデルの作成手順を説明する。

- ① 初期値として N 台の端末を用意する。
 - ② N 台にそれぞれランダムな座標を与える。
 - ③ 現在時刻(内部時間)の座標を記録する。すでに記録されている場合は新しい座標に更新する。
 - ④ フィールドの範囲内で上下左右斜め方向に移動して、内部時間を 1 進める。
 - ⑤ シミュレーション終了時刻まで③～④を繰り返す
- 作成されたモデルを図 2 に示す。



AODV の制御メッセージシステムモデルは仕様書を参考にして、端末の移動モデルと統合して扱えるように作成する。

5 モデルの検証

モデルが正常に動作しているかを検証するために、以下の内容で作成したモデルの検証を行なった

- 送信元ノードと宛先ノードはランダムで決定され、内部時間1秒毎にデータパケットの送信要求が行われる。
- ノードは範囲内のフィールドを動き続ける
- データパケットの送信成功回数と失敗回数を記録して、その和がデータパケットの送信要求回数と一致しているか検証する

表1はシミュレーション環境の具体的な設定である。

端末台数	10台
------	-----

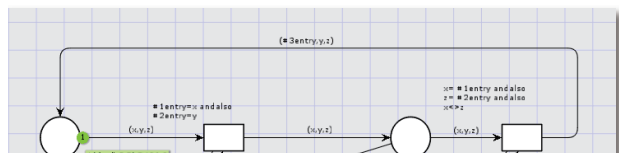
図3は上記設定でモデルの検証を行なった際のデータパケットの送信成功回数(“Send DATA”)と失敗回数(“N Send DATA”)である。

Name	Count	Avg	Min	Max
Marking_size_New_Page'N_Send_DATA_1	48	9.797200	0	28
Marking_size_New_Page'Send_DATA_1	92	39.425213	0	72

Simulation steps executed: 42301
Model time: 100.0

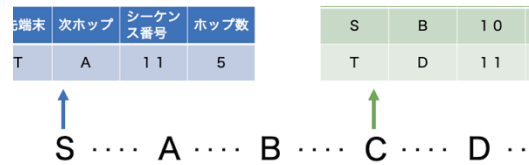
シミュレーション時間は100秒であるので、データパケットの送信要求は100回発生したこととなる。図3を確認するとデータパケットの送信成功回数が72回、失敗回数28回であるので、その和は100回になる。よってデータパケットの送信要求回数と等値であるので、モデルが正常に動作していることがわかる。

しかし、あるパターンにおいて部分的にモデルがデッドロックを起こす現象が確認された。図4はその現象におけるモデルの状態である



このモデルは経路表内に宛先端末がある場合に双方向の通信経路の有効期限を更新するモデルである。トランジションの条件を満たせず、状態が遷移できていない。

デッドロックが起こった原因を考察する。図5は端末S-T間の通信経路が有効な時における端末SとCの経路表である。経路表はRREQメッセージ(もしくはRREPメッセージ)を受け取った端末が更新する仕組みになっている。



よって端末の途中で中継していた端末Cは両端の端末の宛先を持つことができるが、両端の端末は中間の端末の宛先を持つことができない。この状態で端末Cから端末Sに経路要求が起こった場合、端末Cは端末Sの宛先を知っているためC→S間の端末(C,B,A)は正常に経路表を更新することができるが、S→C間の端末(S,A,B)はCの宛先を持っていないので更新を正常に行えずデッドロックを起こしたのではないかと考えられる。この状態では端末の移動モデル以外のモデルがデッドロックを起こしたままになってしまい、モデルの正常な動きを妨げてしまう。そこで今回のモデルではデッドロックが起こった場合に送信失敗として扱うといった条件を付けた。この条件の元で再度検証を行なった結果、正常に動作した結果と同様の結果を得ることができた。

おわりに

本論文では先行研究で挙げた課題点を解決したAODVのモデル化と移動端末の統合モデルの開発を行なった。検証の結果、提案手法を使うことにより、仕様書では発見できなかったデッドロック等を発見することが示すことができた。また、提案モデルを使うことで正常に動作するプロトコルに改良することが示した。

今回のモデルでは通信の衝突を無視したかたちでモデルの作成を行なった。また、規格化されているAODVの機能もまだ全てを実装できていない。今後はこれらも考慮したモデルを開発することで、より一層現実に近いシミュレーションを行うことが今後の課題である。

参考文献

- [1] 山中 祐樹, 諏訪 敬祐: “MANET”を利用した災害時における情報配信システムのシミュレーション評価”, 情報処理学会, FIT2015(第14回情報科学技術フォーラム), https://ipsj.ixsq.nii.ac.jp/ej/?action=pages_view_main&active_action=repository_view_main_item_detail&item_id=154143&item_no=1&page_id=13&block_id=8 (最終閲覧日:2022/11/27)
- [2] RFC 3561 - Ad hoc On-Demand Distance Vector (AODV) Routing - IETF Tools, July 2003,
- [3] 村田忠夫: “ペトリネットの解析と応用”, 近代化学者, 東京都, 1992