衛星回線を含むネットワークにおける動的経路制御に関する研究

慶應義塾大学環境情報学部村井研究室
藤枝 俊輔
sirokuma@sfc.wide.ad.jp

指導教員
村井純
楠本博之
平成12年 2月 28日

Abstract

本研究では、片方向の衛星回線を含むネットワークにおける動的経路制御の問題点を整理し、その解決法を提案し、実装した。

インターネットにおいて片方向の衛星回線を含むネットワークを構築する場合、動的な経路制御プロトコルが正しく動作しないことが問題となる。その原因は、既存の経路制御プロトコルが通信路の双方向性を前提としているため、片方向の通信路を通信に利用できる状態と判断しないからである。

この問題の解決法として仮想ブロードキャストリンクを用いた手法が提案されている。この手法では、衛星回線の受信局から送信局へパケットをトンネリングし、受信局による衛星回線への送信を送信局に代行させることで、衛星回線上に仮想的なブロードキャストリンクを構築する。これにより、既存の経路制御プロトコルをそのまま衛星回線に適用できる。本研究では、この機構を実装し、その有効性を検証した。

しかし、衛星回線には少数の送信局と非常に多くの受信局が接続することが一般的なのに対し、既存の経路制御技術はこのようなネットワークトポロジを想定していない。そのため、多数の受信局が衛星回線に接続した場合、既存の経路生制御プロトコルでは効率や性能が問題になる。

あらゆる規模の受信局が接続する衛星回線をインターネットにおいて利用するためには、衛星回線を含むネットワークトポロジに適応した経路制御プロトコルが必要である。本研究では、Open Shortest Path First(OSPF) を拡張し、衛星回線を含むネットワークにおいて動作するようにした。

本研究で実装した2つの機構を、対応するネットワークトポロジ、多数の受信局が接続した場合の大規模性、経路の収束速度の点から比較評価した。その結果、本研究で開発した新しいOSPFが、既存の提案が持つ、受信局の大規模性に対応できないといった問題点を解決することが示された。

本研究により、あらゆる規模の衛星回線を、大規模ネットワークの一部として利用することができる。これにより、インターネットにおいて、衛星回線の特性をより有効に利用したネットワーク環境が構築できる。g

Abstract

In this research, a dynamic routing architechture for networks that include uni-directional satellite link is designed and implemented. Dynamic routing protocols used in the current Internet, requires bi-directional links. Thus, current dynamic routing protocols can not be used with networks that include uni-directional satellite links.

A technique using pseudo broadcast links, is being propsed to solve this problem. In this technique, pseudo broadcast link is realized by the following steps. A packet from the satellite receiver is tunneled from the satellite receiver to the satellite sender. When the satellite sender receives the tunneled packet, it broadcasts the packet through the satellite link. The broadcast packet is sent as from the satellite receiver. Using this technique, current dynamic routing protocol used in the Internet, can be used within satellite networks. In this research, implementation and evaluation of this technique is done.

However, the satellite link differs from common links used in the Internet. The satellite consists of a few satellite sender and many satellite receivers. Current dynamic routing techniques do not suppose network topology similar to satellite networks. Thus, when dynamic routing is used with a sattelite links with many sattelite receivers, throughput of the dynamic routing system is not good.

To adapt to various scale sattelite links, a new routing mechanism is needed. In this research, extension to OSPF(Open Shortest Path First) is developed. The developed extension to OSPF works with various scale sattelite links.

Evaluation to the implemented system is done with various network topology. Scale and convergence of the implemeted system is evaluated. The evaluation showed that, extending the OSPF for sattelite link, solves the current scaling problem.

By this reseach, using various scale sattelite link over the Internet will be eabled. This research enables construction of Internet using sattelite systems.

目 次

図目次

表目次

第1章

評価

本章では、第5章で述べた仮想ブロードキャストリンクと、第??章で述べた衛星回線を含むネットワークで動作するOSPFの性能を比較する。評価項目は次の3点である。

- 経路制御プロトコルが正しく動作するトポロジ
- 経路収束時間
- Receiverの増加に対する規模性

仮想ブロードキャストリンクは、多くの種類の経路制御プロトコルを動作させせることができる。しかし、本論文で設計したOSPFと比較するため、仮想ブロードキャストリンク上で動作させる経路制御プロトコルにはOSPFを用いた。

1.1 評価環境

各ノードのオペレーションシステムにはFreeBSD3.3-RELEASEを用いた。経路制御デーモンにはzebra-0.82を用いた。UDL環境は、ReceiverのEthernetデバイスドライバを送信できないように変更して実現した。

1.2 OSPFの動作

UDLを含むネットワークにおいて、仮想ブロードキャストリンクを用いてOSPFが正しく動作するか検証した。また、同ネットワークにおいて、本論文で設計した新しいOSPFが正しく動作するか検証した。評価トポロジは、??で述べた4つのトポロジを用いた。

1.2.1 ケース1トポロジでの評価

FeederとReceiverにエリア内で双方向の接続生があるトポロジ上での動作を評価した。図??に示すネットワークにおいて各機構を動作させせた。

図??では、Network AがUDLである。Network B、C、D、EはBDLである。Feeder、Receiver、RouterではOSPFを動作させる。Host Aでは50.0.0.1にデフォルト経路をスタティック

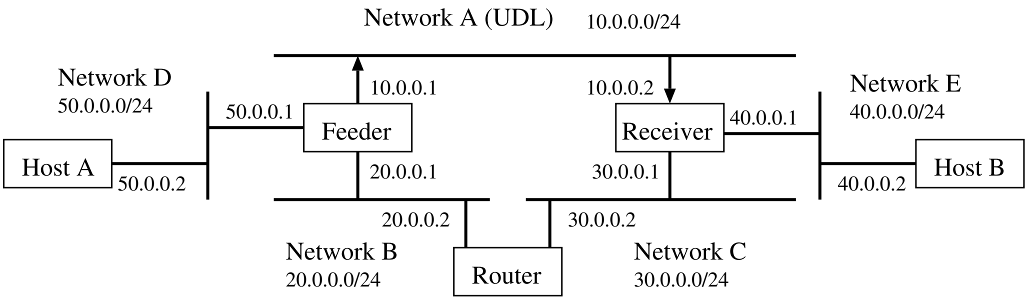

図 1.1: ケース1の評価トポロジ

に設定した。Host Bでは40.0.0.1にデフォルト経路をスタティックに設定した。全てのノードはエリア0(バックボーン)に属する。各ノードがネットワークに接続するコストを表??に示す。

表 1.1: 各リンクのコスト

ノード	Network A	Network B	Network C	Network D	Network E
Feeder	1	1	-	1	-
Receiver	65535[1]	-	1	-	1
Router	-	1	100	-	-

仮想ブロードキャストリンクにおける**OSPF**の動作

DTCPを用いて、Receiverから20.0.0.1(Feeder)へのトンネルを設定した。経路が収束すると、OSPFを動作させたルータ全てがエリア内の経路を保持した。この時のFeeder、Receiver、Routerにおける経路表を図??、図??、図??に示す。

```
Destination        Gateway            Flags      Refs      Use      Netif
10/24              link#2             UC           0        0        ed1
20/24              link#3             UC           0        0        vx0
30/24              10.0.0.2           UG12c        0        0        ed1
40/24              10.0.0.2           UG12c        1       32        ed1
50/24              link#4             UC           0        0        ed0
```

図 1.2: Feederにおける経路表

経路が収束した時点で、tracerouteによってHost AとHost Bの間で通信する場合に使用する経路を調べた。Host AからHost Bに向かうパケットは、Feeder、Receiverの順にルータに中

---

[1] 仮想ブロードキャストリンクでOSPFを動作させる時のみ設定

```
Destination         Gateway             Flags      Refs      Use      Netif
10/24               link#1              UC            0        0       ed1
20/24               30.0.0.2            UG12c        42       42       ed2
30/24               link#2              UC            0        0       ed2
40/24               link#3              UC            0        0       ed3
50/24               30.0.0.2            UG12c         0       35       ed2
```

図 1.3: Receiverにおける経路表

```
Destination         Gateway             Flags      Refs      Use      Netif
10/24               20.0.0.1            UG12c         0        0       ed1
20/24               link#1              UC            0        0       ed1
30/24               link#2              UC            0        0       de0
40/24               30.0.0.1            UG12c         0        8       de0
50/24               20.0.0.1            UG12c         0       29       ed1
```

図 1.4: Routerにおける経路表

継され、UDLを用いてHost Bに到達した。Host BからHost Aに向かうパケットとは、Receiver、Router、Feederの順にルータに中継され、BDLだけを用いてHost Aに到達した。Host AからHost Bにtracerouteを行なった結果を図?? に示す。Host BからHost Aにtracerouteを行なった結果を図??に示す。

```
 1  50.0.0.1  0.629 ms  0.522 ms  0.493 ms
 2  10.0.0.2  1.281 ms  1.014 ms  1.027 ms
 3  40.0.0.2  1.255 ms  1.255 ms  1.215 ms
```

図 1.5: Host AからHost Bへのtracerouteの結果

本研究で設計した**OSPF**の動作

Receiver_Type1とReceiver_Type2の両方を動作させた。その結果、各ルータにおいて、仮想ブロードキャストリンク上で既存のOSPFを動作させた場合と同じ経路表が作成された。(図??、図??、図??参照) また、Host A、Host B間におけるtracerouteでも、同じ結果が得られた。(図??、図??参照)

1.2.2　ケース2トポロジ上での評価

FeederとReceiverにエリア内で双方向の接続生がないトポロジ上での動作を評価した。FeederはABRで、UDLが存在するエリアとは別エリアにBDLを持っている。図??に示すネットワー

```
 1  40.0.0.1  0.372 ms  0.277 ms  0.276 ms
 2  30.0.0.2  0.921 ms  0.648 ms  0.646 ms
 3  20.0.0.1  0.823 ms  0.706 ms  0.692 ms
 4  50.0.0.2  1.167 ms  1.145 ms  1.125 ms
```

図 1.6: Host B から Host A への traceroute の結果

クにおいて各機構を動作させた。

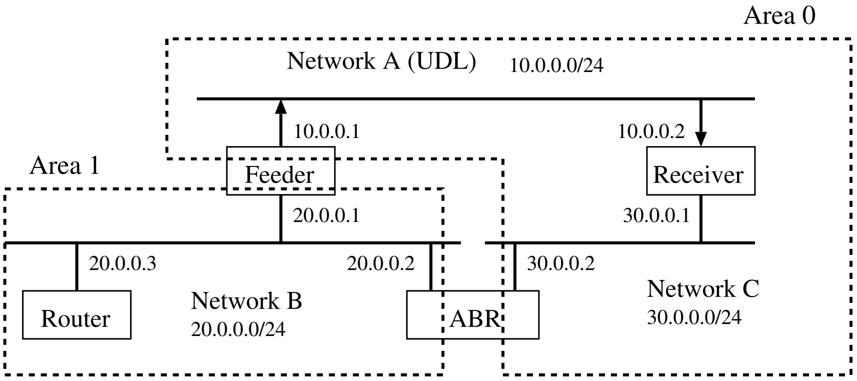

図 1.7: ケース 2 の評価トポロジ

図??では、Network A が UDL である。Network B、C は BDL である。Network A、C はエリア 0 に属し、Netowk B はエリア 1 に属する。エリア 0、エリア 1 間のエリア境界ルータは、Feeder、ABR の 2 つである。各ルータがネットワークに接続するコストを表??に示す。

表 1.2: 各リンクのコスト

ノード	Network A	Network B	Network C
Feeder	1	1	-
Receiver	65535	-	1
ABR	-	1	100
Router	-	1	-

仮想ブロードキャストリンクにおける **OSPF** の動作

DTCP を用いて、Receiver から 20.0.0.1(Feeder) へのトンネルを設定した。経路が収束すると、各ルータは以下の経路を保持した。

---

[2] 仮想ブロードキャストリンクで OSPF を動作させる時のみ設定

- Feeder
  - Network A、Cをエリア0のエリア内経路として保持
  - Network Bをエリア1のエリア内経路として保持。
- Receiver
  - NetworkA、Cをエリア0のエリア内経路として保持
  - Network CをABRを経由したエリア外経路として保持
- ABR
  - Network A、Cをエリア0のエリア内経路として保持
  - Network Bをエリア1のエリア内経路として保持。
- Router
  - Network Bをエリア1のエリア内経路として保持
  - Network A、CをFeederを経由したエリア外経路として保持

Routerから10.0.0.2のアドレスへtracerouteした結果を図??に示す。また、Routerから30.0.0.2のアドレスへtracerouteした結果を図??に示す。どちらの場合も、Feederをエリアの出口とし、UDLを経由して中継され、各IPアドレスに到達している。

```
1  20.0.0.1  0.299 ms  0.234 ms  0.231 ms
2  10.0.0.2  0.916 ms  0.565 ms  0.549 ms
```

図 1.8: Routerから10.0.0.2へのtraceroute

```
1  20.0.0.1  0.306 ms  0.234 ms  0.231 ms
2  10.0.0.2  0.926 ms  0.562 ms  0.552 ms
3  30.0.0.2  0.731 ms  0.617 ms  0.691 ms
```

図 1.9: Routerから30.0.0.2へのtraceroute

経路が収束した時のFeeder、Receiver、ABR、Routerにおける経路表を図??、図??、図??に示す。また、Feeder、Receiver、ABR、Routerにおいてospfdが持ったエリア内経路とエリア外経路を図??、図??、図??に示す。

```
Destination        Gateway            Flags     Refs     Use     Netif
10/24              link#2             UC           0        0      ed1
20/24              link#3             UC           0        0      vx0
30/24              10.0.0.2           UG12c        1        6      ed1
```

図 1.10: Feeder における経路表

```
========== OSPF network routing table ==========
N    10.0.0.0/24        [1] area: 0.0.0.0
                        directly attached to ed1
N    20.0.0.0/24        [1] area: 0.0.0.1
                        directly attached to vx0
N    30.0.0.0/24        [2] area: 0.0.0.0
                        via 10.0.0.2, ed1
========== OSPF router routing table ==========
R    20.0.0.1      [1] area: 0.0.0.1, ABR
                           via 20.0.0.2, vx0
```

図 1.11: Feeder の ospfd が持つエリア内経路、エリア外経路

```
Destination        Gateway            Flags     Refs     Use     Netif
10/24              link#1             UC           0       0      ed1
20/24              30.0.0.2           UG12c      103     118      ed2
30/24              link#2             UC           0       0      ed2
40/24              30.0.0.2           UG12c        0       0      ed2
40.0.0.2           link#3             UHLW         1     182      ed3
```

図 1.12: Receiver における経路表

```
========== OSPF network routing table ==========
N    10.0.0.0/24        [65535] area: 0.0.0.0
                        directly attached to ed1
N IA 20.0.0.0/24        [2] area: 0.0.0.0
                        via 30.0.0.2, ed2
N    30.0.0.0/24        [1] area: 0.0.0.0
                        directly attached to ed2
========== OSPF router routing table ==========
R    10.0.0.1      [65535] area: 0.0.0.0, ABR
                           via 10.0.0.1, ed1
R    20.0.0.1      [1] area: 0.0.0.0, ABR
                           via 30.0.0.2, ed2
```

図 1.13: Receiver の ospfd が持つエリア内経路

```
Destination        Gateway            Flags     Refs     Use     Netif
10/24              30.0.0.1           UG12c       0        0      de0
20/24              link#1             UC          0        0      ed1
30/24              link#2             UC          0        0      de0
```

図 1.14: ABR における経路表

```
========== OSPF network routing table ==========
N    10.0.0.0/24        [99] area: 0.0.0.0
                        via 30.0.0.1, de0
N    20.0.0.0/24        [1] area: 0.0.0.1
                        directly attached to ed1
N    30.0.0.0/24        [100] area: 0.0.0.0
                        directly attached to de0
========== OSPF router routing table ==========
R    10.0.0.1      [1] area: 0.0.0.1, ABR
                           via 20.0.0.1, ed1
                           [99] area: 0.0.0.0, ABR
                           via 30.0.0.1, de0
```

図 1.15: ABR の ospfd が持つエリア内経路

```
Destination         Gateway             Flags      Refs      Use      Netif
10/24               20.0.0.1            UG12c         0        6      fxp1
20/24               link#2              UC            0        0      fxp1
30/24               20.0.0.1            UG12c         0        6      fxp1
```

図 1.16: Router が持つ経路表

```
========== OSPF network routing table ==========
N IA 10.0.0.0/24        [2] area: 0.0.0.1
                        via 20.0.0.1, fxp1
N    20.0.0.0/24        [1] area: 0.0.0.1
                        directly attached to fxp1
N IA 30.0.0.0/24        [3] area: 0.0.0.1
                        via 20.0.0.1, fxp1
========== OSPF router routing table ==========
R    10.0.0.1      [1] area: 0.0.0.1, ABR
                           via 20.0.0.1, fxp1
R    20.0.0.1      [1] area: 0.0.0.1, ABR
                           via 20.0.0.2, fxp1
```

図 1.17: Router の ospfd が持つエリア内経路、エリア外経路

本研究で設計した**OSPF**の動作

Feeder と Receiver の間にエリア内で双方向の接続性がないトポロジでは、Receiver_Type2 は動作しない。そのため、Receiver_Type1 だけを動作させた。その結果、各ルータにおいて、仮想ブロードキャストリンク上で既存のOSPF を動作させた場合と同じ経路表が作成された。(図??、図??、図??参照) また、Feeder、Receiver、ABR、Router において、ospfd が持ったエリア内経路とエリア外経路も同一だった。(図??、図??、図??参照。) また、Router から 10.0.0.2、30.0.0.2 への traceroute も同じ結果だった。(図??、??参照。)

1.2.3 ケース3トポロジ上での評価

FeederとReceiverにエリア内で双方向の接続生がないトポロジ上での動作を評価した。ReceiverはABRで、UDLが存在するとは別のエリアにBDLを持っている。図??に示すネットワークにおいて各機構を動作させた。

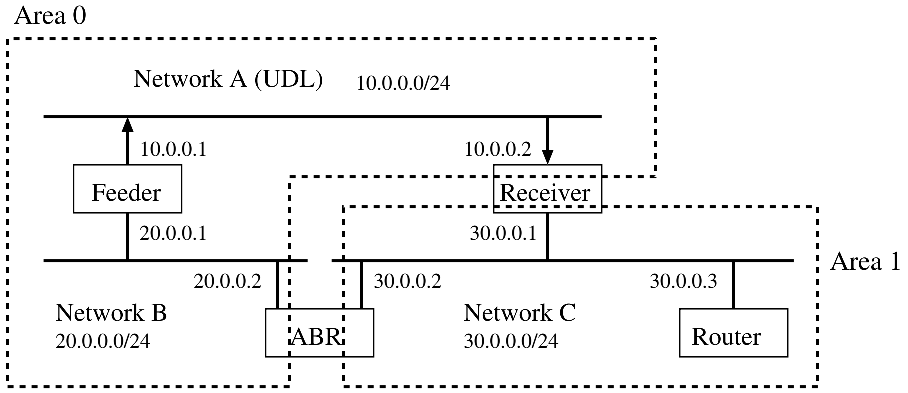

図 1.18: ケース3の評価トポロジ

図??では、Network AがUDLである。Network B、CはBDLである。Network A、Bはエリア0に属し、Netowk Cはエリア1に属する。エリア0、エリア1間のエリア境界ルータは、Receiver、ABRの2つである。各ルータがネットワークに接続するコストを表??に示す。

表 1.3: 各リンクのコスト

ノード	Network A	Network B	Network C
Feeder	1	1	-
Receiver	65535	-	1
ABR	-	1	100
Router	-	-	1

仮想ブロードキャストリンクにおけるOSPFの動作

DTCPを用いて、Receiverから20.0.0.1(Feeder)へのトンネルを設定した。しかし、一旦経路が収束した後、UDLが経路として利用できなくなり、Receiverの経路制御デーモンが動作しないなどの障害が現れた。これは、次の問題が生じたためである。

1. Receiverにおいて、20.0.0.1(Feeder)へのトンネルが設定される。この時、Receiverから20.0.0.1への経路がまだ存在していない場合、2に進む。Receiverから20.0.0.1への経路が存在した場合、3に進む。

---

[3] 仮想ブロードキャストリンクでOSPFを動作させる時のみ設定

2. Network B、Cが各ルータに広告され、ReceiverとFeeder間において、ABRを経由した経路が作成される。
3. Receiverから20.0.0.1へABRを経由して到達する。
4. ReceiverとFeederが隣接ルータとして互いを認識し、Network A(UDL)の存在が各ルータに広告される。
5. Receiverは、20.0.0.1への経路としてUDLを利用する設定に変わる。その理由は、OSPFではコストにかかわらずエリア内の経路がエリア外の経路に優先するからである。この場合、ReceiverはABRを経由するエリア外の経路よりも、Network A(UDL)を用いたエリア内の経路を優先してしまう。
6. Receiverはカプセル化したパケットをUDLに送信しようとし、失敗する。
7. FeederとReceiverの間の経路が失われる。

本研究で設計した**OSPF**の動作

FeederとReceiverの間にエリア内で双方向の接続性がないトポロジでは、Receiver_Type2は動作しない。そのため、Receiver_Type1を動作させた。この機構では、仮想ブロードキャストリンク上でOSPFを動作させた時に生じた問題は起こらなかった。経路が収束すると、各ルータは次の経路を保持した。

- Feeder
  - Network A、Bをエリア0のエリア内経路として保持
  - Network CをReceiverを経由したエリア外経路として保持
- Receiver
  - Network A、Bをエリア0のエリア内経路として保持
  - Network Cをエリア1のエリア内経路として保持
- ABR
  - Network A、Bをエリア0のエリア内経路として保持
  - Network Cをエリア1のエリア内経路として保持
- Router
  - Network Cをエリア1のエリア内経路として保持
  - Network A、BをABRを経由したエリア外経路として保持

```
 1  10.0.0.2  0.447 ms  0.337 ms  0.316 ms
 2  30.0.0.3  0.572 ms  0.550 ms  0.549 ms
```

図 1.19: Feederから30.0.0.3へのtraceroute

```
 1  30.0.0.2  0.650 ms  0.369 ms  0.369 ms
 2  10.0.0.1  0.760 ms  0.593 ms  0.561 ms
```

図 1.20: Routerから10.0.0.1へのtraceroute

Feederから30.0.0.3のIPアドレスへtracerouteした結果を図??に示す。Feederから送信されたパケットは、UDLを経由し、Receiverに中継され、30.0.0.3に到達している。Routerから10.0.0.1のIPアドレスへtracerouteした結果を図??に示す。Routerから送信されたパケットは、ABRを経由してエリア0に中継され、Feeder(10.0.0.1)に到達している。

経路が収束した時のFeeder、Receiver、ABR、Routerにおける経路表を図??、図??、図??、図??に示す。Feeder、Receiver、ABR、Routerにおいてospfdが持ったエリア内経路とエリア外経路を図??、図??、図??、図??に示す。

```
Destination        Gateway            Flags     Refs     Use     Netif
10/24              link#2             UC           0        0     ed1
20/24              link#3             UC           0        0     vx0
30/24              10.0.0.2           UG12c        0        0     ed1
```

図 1.21: Feeder における経路表

```
========== OSPF network routing table ==========
N    10.0.0.0/24        [10] area: 0.0.0.0
                        directly attached to ed1
N    20.0.0.0/24        [1] area: 0.0.0.0
                        directly attached to vx0
N IA 30.0.0.0/24        [11] area: 0.0.0.0
                        via 10.0.0.2, ed1
========== OSPF router routing table ==========
R    10.0.0.2      [10] area: 0.0.0.0, ABR
                           via 10.0.0.2, ed1
```

図 1.22: Feeder の ospfd が持ったエリア内経路とエリア外経路

```
Destination        Gateway            Flags     Refs     Use     Netif
10/24              link#1             UC           0        0     ed1
30/24              link#2             UC           0        0     ed2
20/24              30.0.0.2           UG12c        0        0     ed2
```

図 1.23: Receiver における経路表

```
========== OSPF network routing table ==========
N    30.0.0.0/24        [1] area: 0.0.0.1
                        directly attached to ed2
N IA 10.0.0.0/24        [12] area: 0.0.0.1
                        via 30.0.0.2, fxp0
N IA 20.0.0.0/24        [2] area: 0.0.0.1
                        via 30.0.0.2, fxp0
========== OSPF router routing table ==========
R    20.0.0.1      [1] area: 0.0.0.1, ABR
                           via 30.0.0.2, ed2
```

図 1.24: Receiver の ospfd が持ったエリア内経路とエリア外経路

```
Destination          Gateway              Flags      Refs       Use       Netif
10/24                20.0.0.1             UG12c        0        156        ed1
20/24                link#1               UC           0          0        ed1
30/24                link#2               UC           0          0        de0
```

図 1.25: ABR における経路表

```
========== OSPF network routing table ==========
N    10.0.0.0/24        [11] area: 0.0.0.0
                        via 20.0.0.1, ed1
N    20.0.0.0/24        [1] area: 0.0.0.0
                        directly attached to ed1
N    30.0.0.0/24        [100] area: 0.0.0.1
                        directly attached to de0
========== OSPF router routing table ==========
R    10.0.0.2      [100] area: 0.0.0.1, ABR
                           via 30.0.0.1, de0
```

図 1.26: ABR の ospfd が持ったエリア内部経路

```
Destination        Gateway            Flags     Refs     Use     Netif
10/24              30.0.0.2           UG12c        0       0     fxp0
20/24              30.0.0.2           UG12c        0       0     fxp0
30/24              link#1             UC           0       0     fxp0
```

図 1.27: Router における経路表

```
========== OSPF network routing table ==========
N IA 10.0.0.0/24        [12] area: 0.0.0.1
                        via 30.0.0.2, fxp0
N IA 20.0.0.0/24        [2] area: 0.0.0.1
                        via 30.0.0.2, fxp0
N    30.0.0.0/24        [1] area: 0.0.0.1
                        directly attached to fxp0
========== OSPF router routing table ==========
R    10.0.0.2      [1] area: 0.0.0.1, ABR
                           via 30.0.0.1, fxp0
R    20.0.0.1      [1] area: 0.0.0.1, ABR
                           via 30.0.0.2, fxp0
```

図 1.28: Router の ospfd が持ったエリア内経路とエリア外経路

1.2.4 ケース4

UDLによってエリアが分割されているトポロジ。Feeder、Receiver共にABRではないが、エリア内にFeederからReceiverへの接続性だけが存在している。Feeder側の区分とReceiver側の区分には、それぞれABRが存在する。図??に示すネットワークにおいて、各機構を動作させた。

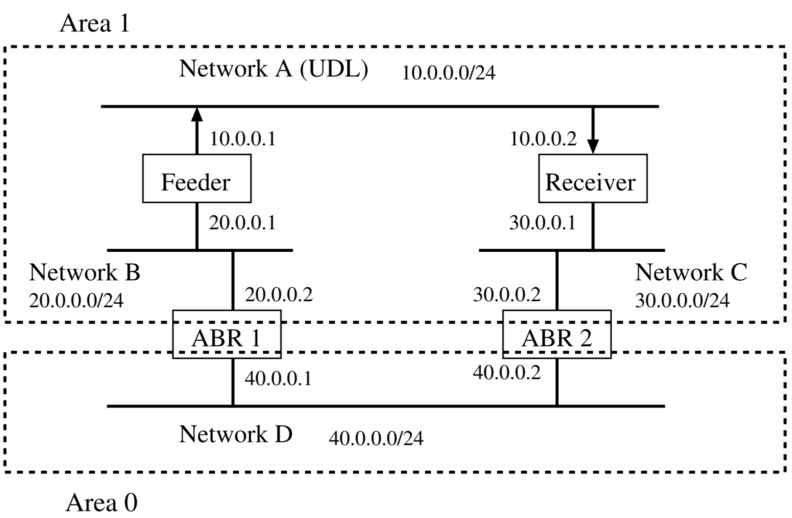

図 1.29: ケース4の評価トポロジ

図??では、Network AがUDLである。Network B、C、D、はBDLである。Network A、B、Cはエリア1に属し、Network Dはエリア0(バックボーン)に属している。エリア0、エリア1間には、ABR1、ABR2という2つのエリア境界ルータが存在している。各ルータがネットワークに接続するコストを表??に示す。

表 1.4: 各リンクのコスト

ノード	Network A	Network B	Network C	Network D
Feeder	1	1	-	-
Receiver	65535	-	1	-
ABR1	-	1	-	1
ABR2	-	-	1	1

仮想ブロードキャストリンクにおける**OSPF**の動作

DTCPを用いて、Receiverから20.0.0.1(Feeder)へのトンネルを設定した。しかし、UDLを利用しない形で経路が収束した後、UDLを利用したFeederからReceiverへの経路とUDLを逆流

---

[3] 仮想ブロードキャストリンクでOSPFを動作させる時のみ設定

しようとするReceiverからFeederへの経路が設定された。その後、再びUDLを利用しない形で経路が設定されるといった繰り返しが生じた。これは、次の問題が生じたためである。

1. Receiverにおいて、20.0.0.1(Feeder) へのトンネルが設定される。この時、Receiverから20.0.0.1への経路が存在していない場合、2に進む。Receiverから20.0.0.1への経路が存在した場合、4に進む。

2. エリア1はFeeder側の区分(Feeder、ABR1、Network B)とReceiver側の区分(Receiver、ABR2、Network C)に別れ、独立したエリアとしてエリア内の経路が計算される。また、エリア0でもエリア内の経路が計算される。

3. ABR1、ABR2はsummary LSAによってそれぞれのエリアの経路を別エリアに広告する。
   - ABR1はFeeder側の区分の経路をエリア0に広告する。
   - また、ABR1はエリア0の経路をFeeder側の区分に広告する。
   - ABR2はReceiver側の区分の経路をエリア0に広告する。
   - また、ABR2はエリア0の経路をReceiver側の区分に広告する。
   - その後、ABR1はABR2が広告したReceiver側の区分の経路をFeeder側の区分に広告する。
   - ABR2はABR1が広告したFeeder側の区分の経路をReceiver側の区分に広告する。

   以上によって、Feeder側の区分とReceiver側の区分はバックボーンを経由し、独立したエリア同士として通信ができる状態になる。

4. Receiverかわ20.0.0.1(Feeder)へバックボーンを経由して到達する。

5. ReceiverとFeederが隣接ルータとして互いを認識し、Network A(UDL)の存在が各ルータに広告される。

6. Receiverは、20.0.0.1への経路としてUDLを利用する設定に変わる。また、Receiver側の区分のルータはFeeder側の区分へ到達するためにUDLを逆流する経路を利用しようとする。その理由は、ケース3と同様に、OSPFではコストにかかわらずエリア内の経路がエリア外の経路に優先するからである。Network A(UDL)の存在が広告された時点で、Feeder側の区分とReceiver側の区分は1つのエリアとして経路計算される。Receiver側の全てのルータはFeeder側の区分へ到達するため、UDLを逆流する経路を利用しようとする。

7. Receiver側の区分からFeeder側の区分への到達生が失われる。ReceiverはBDLを利用してトンネリングするはずのパケットを再びUDLに送信しようとし失敗する。FeederとReceiver間ではHelloが片方向にしか流れなくなるので、RouterDeadIntervalが経過するとFeederとReceiverは通信の障害を検知する。

8. FeederとReceiverは、UDLをリンク情報に含まない新しいRouter LSAを作成し、それぞれの区分に広告する。

9. エリア1内の全てのルータが、UDLを利用しない形で経路を再計算する。(2に戻る。)

つまり、Receiver側がバックボーンを経由して20.0.0.1(Feeder)に到達している間はFeederとReceiverが経路制御メッセージを交換できる。しかし、FeedderとReceiverが必要な情報を交換し終え新しい経路が計算されると、Receiverは20.0.0.1(Feeder)に到達できなくなる。そのため、一定時間後に元の経路に戻る。

本研究で設計した**OSPF**の動作

FeederとReceiverの間にエリア内で双方向の接続性がないトポロジでは、Receiver_Type2は動作しない。そのため、Receiver_Type1を動作させた。この場合、仮想ブロードキャストリンク上でOSPFを動作させた時に生じた問題は起こらなかった。経路が収束すると、各ルータは次の経路を保持した。

- Feeder
    - Network A、B、Cをエリア1のエリア内部経路として保持
    - Network DをABR1を経由したエリア外経路として保持
- Receiver
    - Network Cをエリア1のエリア内経路として保持
    - Network A、B、DをABR2を経由したエリア外経路として保持
- ABR1
    - Network A、B、Cをエリア1のエリア内経路として保持
    - Network Dをエリア0のエリア内経路として保持
- ABR2
    - Network Cをエリア1のエリア内経路として保持
    - Network Dをエリア0のエリア内経路として保持
    - Network A、BをABR1を経由したエリア外経路として保持

ABR1から30.0.0.2のIPアドレスへtracerouteした結果を図?? に示す。ABR1から送信されたパケットは、UDLを経由しReceiverに中継され30.0.0.2に到達する。

```
1  20.0.0.1  0.479 ms  0.520 ms  0.225 ms
2  10.0.0.2  0.917 ms  0.589 ms  0.561 ms
3  30.0.0.2  1.078 ms  0.798 ms  0.921 ms
```

図 1.30: ABR1から30.0.0.2へのtraceroute

Receiverから20.0.0.1のIPアドレスへtracerouteした結果を図?? に示す。Receiverから送信されたパケットは、ABR2、ABR1の順に中継され20.0.0.1に到達する。

```
1  30.0.0.2 (30.0.0.2)  0.635 ms  0.451 ms  0.413 ms
2  40.0.0.1 (40.0.0.1)  0.959 ms  0.600 ms  0.573 ms
3  20.0.0.1 (20.0.0.1)  0.585 ms  0.584 ms  1.029 ms
```

図 1.31: Receiverから20.0.0.1へのtraceroute

経路が収束した時のFeeder、Receiver、ABR1、ABR2における経路表を図??、図??、図??、図??、に示す。また、Feeder、Receiver、ABR1、ABR2においてospfdが持ったエリア内経路とエリア外経路を図??、図??、図??、図??、に示す。

```
Destination        Gateway            Flags     Refs     Use     Netif
10/24              link#2             UC          0        0      ed1
20/24              link#3             UC          0        0      vx0
30/24              10.0.0.2           UG12c       0        0      ed1
40/24              20.0.0.2           UG12c       0        0      vx0
```

図 1.32: Feederにおける経路表

```
========== OSPF network routing table ==========
N    10.0.0.0/24        [1] area: 0.0.0.1
                        directly attached to ed1
N    20.0.0.0/24        [1] area: 0.0.0.1
                        directly attached to vx0
N    30.0.0.0/24        [2] area: 0.0.0.1
                        via 10.0.0.2, ed1
N IA 40.0.0.0/24        [2] area: 0.0.0.1
                        via 20.0.0.2, vx0
========== OSPF router routing table ==========
R    20.0.0.2      [1] area: 0.0.0.1, ABR
                           via 20.0.0.2, vx0
R    30.0.0.1      [2] area: 0.0.0.1, ABR
                           via 10.0.0.2, ed1
```

図 1.33: Feederのospfdが持ったエリア内経路とエリア外経路

```
Destination         Gateway             Flags      Refs      Use      Netif
10/24               link#1              UC           0         0       ed1
20/24               30.0.0.2            UG12c        4       156       ed2
30/24               link#2              UC           0         0       ed2
40/24               30.0.0.2            UG12c        0         0       ed2
```

図 1.34: Receiver における経路表

```
========== OSPF network routing table ==========
N IA 10.0.0.0/24        [4] area: 0.0.0.1
                        via 30.0.0.2, ed2
N IA 20.0.0.0/24        [3] area: 0.0.0.1
                        via 30.0.0.2, ed2
N    30.0.0.0/24        [1] area: 0.0.0.1
                        directly attached to ed2
N IA 40.0.0.0/24        [2] area: 0.0.0.1
                        via 30.0.0.2, ed2
========== OSPF router routing table ==========
R    30.0.0.1      [1] area: 0.0.0.1, ABR
                           via 30.0.0.2, ed2
```

図 1.35: Receiver の ospfd が持ったエリア内経路とエリア外経路

```
Destination        Gateway            Flags     Refs     Use     Netif
10/24              20.0.0.1           UG12c       0        0     fxp0
20/24              link#1             UC          0        0     fxp0
30/24              20.0.0.1           UG12c       0        0     fxp0
40/24              link#2             UC          0        0      xl0
```

図 1.36: ABR1 における経路表

```
========== OSPF network routing table ==========
N    10.0.0.0/24        [2] area: 0.0.0.1
                        via 20.0.0.1, fxp0
N    20.0.0.0/24        [1] area: 0.0.0.1
                        directly attached to fxp0
N    30.0.0.0/24        [3] area: 0.0.0.1
                        via 20.0.0.1, fxp0
N    40.0.0.0/24        [1] area: 0.0.0.0
                        directly attached to xl0
========== OSPF router routing table ==========
R    30.0.0.1      [3] area: 0.0.0.1, ABR
                           via 20.0.0.1, fxp0
```

図 1.37: ABR1 の ospfd が持ったエリア内経路とエリア外経路

```
Destination          Gateway              Flags     Refs      Use      Netif
10/24                30.0.0.1             UG12c       0         0       de0
20/24                40.0.0.1             UG12c       1       151       ed1
30/24                link#2               UC          0         0       de0
40/24                link#1               UC          0         0       ed1
```

図 1.38: ABR2 における経路表

```
========== OSPF network routing table ==========
N IA 10.0.0.0/24        [3] area: 0.0.0.0
                        via 40.0.0.1, ed1
N IA 20.0.0.0/24        [2] area: 0.0.0.0
                        via 40.0.0.1, ed1
N    30.0.0.0/24        [100] area: 0.0.0.1
                        directly attached to de0
N    40.0.0.0/24        [1] area: 0.0.0.0
                        directly attached to ed1
========== OSPF router routing table ==========
R    20.0.0.2      [1] area: 0.0.0.0, ABR
                           via 40.0.0.1, ed1
```

図 1.39: ABR2 の ospfd が持ったエリア内経路とエリア外経路

1.3 経路収束時間の比較

仮想ブロードキャストリンク上で既存のOSPFを動作させた場合と、本研究で設計したOSPFを動作させた場合について、ネットワークトポロジに変化が生じた場合に経路が収束するまでの時間を比較した。

1.3.1 収束時間の定義

経路の収束とは、全てのルータが新しい経路表を作成し、経路の変更がなくなることである。経路の収束時間とは、新しいリンクが追加された場合または既存のリンクが使用できなくなった場合に、そのリンクを持つルータがそれを検知するまでの時間と、AS内の全てのルータが新しい経路表を作成し終えるまでの時間の合計である。

1.3.2 LSAの伝搬速度

OSPFでは、全てのルータがLSAのデータベースを保持し、その情報を元に経路表を作成することを第??で述べた。各ルータは新しいLSAを受けとる度に新しい経路を計算する。そのため、LSAの伝搬速度が経路の収束に大きく影響する。しかし、OSPFはLSAの伝搬速度を速めるためにFloodingを利用する。そのため、LSAの伝搬に必要な時間は僅かである。本設計では、衛星回線においてDRとReceiver_Type2はデータベースの交換を行なわない。そのため、Floodingに使用される経路が減り、LSAの伝搬速度は遅くなる可能性がある。しかし、LSAの伝搬に必要な時間は僅かなので、これによって経路の収束時間が受ける影響は少ない。

1.3.3 LSAの作成速度

OSPFでは、隣接ルータと接続性をHelloプロトコル(以下Hello)によって検知する。隣接ルータと接続性が失われた場合、ルータは即座に新しいLSAを作成し広告する。このために必要な時間は、既存のOSPFと本研究で設計したOSPFにおいて違いはない。

隣接ルータと接続が回復した場合、または新しいルータがネットワークに加わった場合、各ルータはHelloによって新しい隣接ルータを発見し、必要ならばトポロジデータベースの交換を行なう。データベースの交換が終った後、各ルータは新しいLSAを作成する。ここで、UDLへの接続性を一度失ったFeederが復帰した場合を考える。他にDRが存在していた場合、FeederはDRとデータベースを交換する。しかし、他にDRが存在していなかった場合、既存のOSPFでは、Feederが新しいDRになり全てのReceiverとデータベースを交換する必要がある。この時、Feeder(DR)が交換するデータベースの量はReceiverの数にほぼ比例する。本研究で設計したOSPFでは、FeederはReceiver_Type1とだけデータベースを交換すればよい。そのため、既存のOSPFに比べデータベースを同期させるノード数が大きく減少する。データベースの交換にかかる時間が減少するため、その分速く新しいLSAを作成し広告できると考えられる。

1.4 規模性の比較

UDLにおいて、Receiverの数が増えた場合のDR(Feeder)の負荷について考える。既存のOSPFと本研究で設計したOSPFが最も異なる点は、本研究で設計したOSPFが一部のReceiver

としかデータベースを同期させない点である。

Receiver_Type2 が増えた場合、Feeder にかかる新たな負荷を次に示す。

- 送受信する経路制御パケットのトラフィック量
    - Receiver から受信する UDL_Hello パケット (Hello Interval 間隔に 1 パケット)
    - UDL に送信する UDL_Hello が 4 バイト大きくなる
- メモリ
    - 新しい Receiver が作成する Router LSA を格納するメモリ
    - 新しい隣接ルータ (Receiver) の状態を保持するメモリ
- 処理
    - Hello の処理 (隣接ルータの状態を維持する処理を含む)

これらの負荷はどれも小さいため、Receiver_Type2 が増加した場合も、DR の負荷は緩やかに上昇する。Receiver_Type1 が増えた場合、DR には上に述べた負荷に加え次の負荷がかかる。

- 送受信する経路制御パケットのトラフィック量
    - 新しい Receiver_Type1 とデータベース情報を交換するパケット
      (Database Description Packet)
    - どちらかに欠けている LSA があった場合、それを送り合うパケット
      (Link State Request Packet、Link State Update Packet)
- メモリ
    - ほげ
- 処理
    - データベース情報の交換に必要な処理
    - データベースの同期に必要な処理
- データベースの更新

    新しい LSA が広告された場合、DR(Feeder) は Link State Update Packet によって、マルチキャストで UDL に広告する。各 Receiver は、Link State Acknowledgememt Packet により、無事新しい LSA を受信したことを DR に知らせる。この時 DR は、UDL に接続する全てのノードにおいて、Link State Update Packet が無事に受信されたことを確認する必要がある。また、Link State Acknowledgement Packet を受信していないノードに対し再度 Link State Update Packet を送信する必要がある。

これらの負荷は、Receiver_Type2 が増えた時に加わる負荷に比べトラフィック、メモリ、処理の点から大きい。そのため、Receiver_Type1 が増加した場合、DR に負荷が集中し経路制御プロトコルが効率的に動作しないと考えられる。Receiver_Type1 が増えるたびに DR にかかる負荷は、既存の OSPF において隣接ルータが増えた場合に DR にかかる負荷とほぼ同じである。本研究で設計した OSPF では、一部の Receiver だけが Receiver_Type1 として動作するのに比べ、既存の OSPF では全ての Receiver が Receiver_Type1 と同じ動作をする。したがって、本研究で設計した OSPF は既存の OSPF に比べより多数の Receiver が接続するトポロジで動作すると考えられる。Receiver の数と DR の負荷との関係について、シミュレータなどを用いて今後更に調査を行なう予定である。

第2章

結論と今後の課題

2.1 結論

本論文では、インターネットにおいて片方向の衛星回線を含むネットワークを構築する際に生じる問題を整理し、動的な経路制御プロトコルが正しく動作しない問題を解決した。

衛星回線は、既存の通信回線にない様々な特徴を持っている。そのため、インターネットにおいて衛星回線を用いる場合、衛星回線の特性を活かすために、既存の通信回線とは異なる方法で用いる必要がある。その代表的な例として、衛星回線を片方向に設定し送受信を行うノードと受信だけを行うノードを接続する方法を述べた。しかし、このような片方向の衛星回線を含むネットワークでは、RIPやOSPFなどの経路制御プロトコルが正しく動作しない。その理由を第2章で述べた。

この問題を解決するため、現在IETFでは”仮想ブロードキャストリンクを用いる手法”が議論され、その技術の標準化が進められている。本研究では、この既存の提案による機構を実装し評価した。その結果、仮想ブロードキャストリンク上でOSPFを動作させた場合、エリアの分け方によってOSPFが正しく動作しない場合があると分かった。仮想ブロードキャストリンクを用いる手法では、ReceiverはUDLに送信するパケットをFeederのBDLインタフェース宛にカプセル化し、BDLインタフェースから送信する。しかし、OSPFはエリア内の経路をエリア外の経路よりも優先させるため、ReceiverからFeederのBDLインタフェースまでの経路がエリア内にUDLしか存在しない場合、カプセル化したパケットを再びUDLインタフェースから送信しようとし失敗する。また、衛星回線には少数の送信局と非常に多くの受信局が接続することが一般的なのに対し、既存の経路制御技術はこのようなネットワークトポロジを想定していない。そのため、多数の受信局が衛星回線に接続した場合、既存の経路生制御プロトコルでは効率や性能が問題になる。

OSPFを改変し、衛星回線を含むネットワークにおいて動作する新しい経路制御プロトコルを開発した。この機構では、FeederとReceiver間の接続性を検知するため、UDL Helloプロトコルを設計し、ReceiverからFeederに双方向回線を経由してHelloメッセージを送信する。また、Receiverが2種類の手法でFeederと経路制御メッセージを交換し、指定されたReceiverだけがFeederとデータベースを同期させることで、Feederの負荷を下げた。

本研究で改変したOSPFの有効性を検証するため、LAN上に構築した様々なネットワークトポロジで評価を行なった。その結果から、本機構があらゆる種類のネットワークトポロジで動作

することを述べた。また、仮想ブロードキャストリンクを構築する機構との比較により、改変したOSPFがより大規模な衛星回線上で動作することを述べた。

本研究により、あらゆる規模の衛星回線を、大規模ネットワークの一部として利用することができるようになった。これにより、インターネットにおいて、衛星回線の特性をより有効に利用するネットワークが構築できるようになった。

2.2 今後の課題

本研究で開発した新しいOSPFでは、Receiverが2種類の動作をする。本設計では多数のReceiverがReceiver_Type2として動作し、少数のReceiverがReceiver_Type1として動作することを想定している。どのReceiverがどの種類の動作をするかは、管理者の設定で行なう。しかし、トポロジの変化によってReceiver_Type2がReceiver_Type1として動作する必要が起こる状況も考えられる。そのため、Receiverの動作を動的に変更する機能が必要である。

また、本研究によるOSPFでは、Receiverの大規模性を考慮した設計を行なった。しかし、どの程度のReceiver数までこのプロトコルが動作するのか、具体的な数字は計測していない。今後、ネットワークシミュレータなどを用いて、衛星回線に多数のReceiverが接続した場合の動作を調査する予定である。