Server-based Routing for Internet over Satellite Links

Gang Qiang † Feng Guo † Zengji Liu † Susumu Ishihara ‡ Tadanori Mizuno ‡
† State Key Laboratory on Intergrated Service Network, Xidian University, Xi'an, Shannxi, China
‡ Mizuno Laboratory, Department of Computer Science, Shizuoka University, Hamamatsu, Japan

Abstract: Current routing protocols assume that routers are connected by *bi-directional links* (BDLs). Recently, High bandwidth, unidirectional transmission to low cost, receiver-only hardware is becoming an emerging network fabric, e.g. broadcast satellite links or some cable links. Specific protocols are required to support dynamic routing over such networks. In this paper, we present a new solution *SEver-based Routing Protocol* (SERP) to support dynamic routing in networks with *unidirectional links* (UDLs). For a link cannot be used in a routing algorithm until a cycle of the link is found, a *unidirectional link registration server* (*ULRS*) is used to process UDLs information centrally in order to simplify the impact of UDLs on the topology of the BDLs network. By extracting UDLs routing problems from the traditional BDLs routing problems, SERP is optimal for bi-directional islands connected via unidirectional links, especially for Internet over satellite systems.

1. Introduction

Recently, broadcast satellite link is becoming an emerging network fabric [1]. Since satellite systems can offer a high bandwidth wireless communication and a large geographical coverage, it is reasonable to integrate them into the Internet transparently. Furthermore, to support some applications, such as digital battlefield and crisis emergency, which need "any where any time" access to information repositories from the Internet, and to achieve the ultimate goal of "whenever wherever" access to communications and the global Internet, there still needs the help of satellite links.

For the limitation of cost and law, satellite links are mostly deployed to provide receive-only connectivity to the Internet, so that UDLs are introduced. On the other hand, current routing protocols are typically categorized into *Link State* (LS) [2] and *Distance Vector* (DV) [3] that are used in IGPs (intra-domain protocols), and *Path Vector* (PV) that is somewhat similar to DV and is used by BGP [4], the most popular EGP (inter-domain protocol). All these approaches do not support UDLs. In fact, adjacent routers cannot exchange their routing information to create forwarding tables, while UDLs are presented.

In Section 2, we consider the impact of UDLs on current routing protocol, and how it has been solved in some related work. Then, we show the model of SERP in Section 3 and description in Section 4. In Section 5, we present some analysis to show how different in performance between our proposal and Pure Cycle Discover method (PCD). Simulation results are presented in Section 6. Conclusion follows in Section 7.

2. UDLR Overview

Impacts of UDLs on routing protocols are:

a) Reachability information is hard to maintain. Routing protocols provide mechanisms to maintain the knowledge of direct neighbors and their reachability, in particular after a failure or a recovery. In a BDL network, listening to periodic messages from neighbors is enough to determine which are the direct neighbors and maintain the reachability of each neighbor whereas in a UDL network maybe there is only one-way connectivity.

b) Metrics are not symmetric. Many routing protocols assume that links metrics are symmetric. As to this assumption, routers deduce reverse routing metrics information from routing messages they receive. This is no longer true in a UDL network.

The distinct characteristic of UDLs is that communication is impossible transmitted in both directions. So UDLs brings routing issues that comprise two cases:

1. A unidirectional network overlay on top of bi-directional network, as depicted in Figure 1.

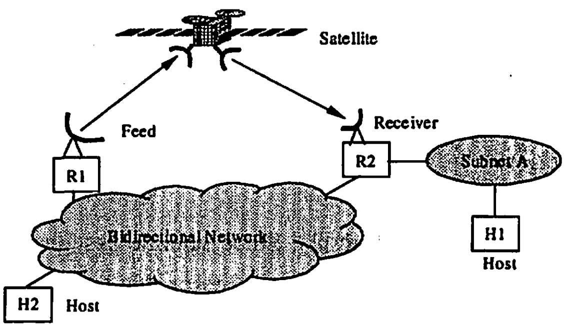

Figure 1 An unidirectional overlays on top of bi-directional network

2. A mixed network, as depicted in Figure 2.

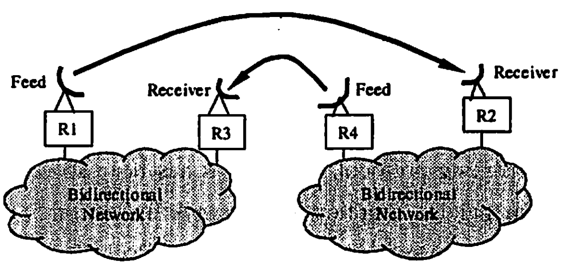

Figure 2 Bidirectional islands connected via unidirectional links

In Figure 1, it is obvious that the route from H2 to H1 via the UDL should be prefered to the route via BDLs as it is the shortest path. In order to do this, R1 must learn that subnet A is reachable via the UDL. While extend Figure1 to Figure 2 that is a case correspond to a more general approach, the topology becomes more complex as UDLs are more and more widespread.

The IETF working group on *unidirectional link routing* (UDLR) is currently working on this problem. In UDLR, the node connected to a UDL with a send-only interface is feed; the node connected to a UDL with a receive-only interface is receiver. UDLR specified the problem of Figure 2 to be two cases: 1) Bi-directional islands connected via UDLs. 2) The general case of asymmetric and possibly UDLs. Our proposal focuses on the Bi-directional islands case.

Former works have been done to exploit that implementing routing protocols in networks that contain UDLs. Robert McCurley and Fred B. Schneider [5] proposed an algorithm using complete topology information, which is obviously not optimal. Other works can be categorized into short term solutions and long term solutions.

2.1 Short term solution

There are two approaches allow dynamic routing in configuration contained UDLs. One is based on routing protocol modification and the other on tunneling, both currently discussed within the IETF UDLR working group.

The representative example of protocol modification is modified RIP which has been exactly described in [6] and [7]. The main scheme of this solution is building up a back channel from a receiver to a feed, so a fatal shortcoming comes out. Consider the configuration depicted in Figure 2. Obviously, the back channel could not be built up while there is UDL among it.

Tunnel solution, precisely described in [8], uses a back channel to build up a tunnel for dynamic routing with UDLs. The routing protocol used here does not need any modification. Besides having the same fault as protocol modification, tunnel solution will encounter another problem that comes out while lots of receivers belong to a feed. It is necessarily to build up too many tunnels that ended at the feed. As satellites provide wide coverage, this problem should be considered.

2.2 Long term solution

Approaches presented by Ernst and Dabbous [9] and Bao and Garcia-Luna-Aceves [10] are long term solutions. Both of them are based on cycle discover method. It is true that a link cannot be used in a routing algorithm until a cycle of the link is found. However, the links of the Internet are mainly BDLs nowadays, and BDLs will play an important role in the Internet. So the algorithm of [9] or [10], which takes all links as UDLs, will not be optimal for large scale deployment. We call such algorithms as PCD.

In this paper, we present a *SEver-based Routing Protocol* (SERP) to support dynamic routing in networks with UDLs, especially for Internet over satellite links. A ULRS is used to process UDLs information centrally within a BDLs island. And its correctness is verified.

3. Network Architecture and Model

Figure 3 shows the architecture of Internet over satellite links. Nodes are connected via BDLs fabric within each *bi-directional zone* (BZ) where there are three kinds of special nodes: feed, receiver and ULRS. UDLs connect BZs through satellite relay fabric. As is typically done for network routing discussions, we model the network as graphs. Excluding UDLs, a BZ is modeled as an undirected graph $G_B = (N, L)$, where N represents all nodes; $L \subseteq N \times N$ is the set of links. By introducing ULRS and only taking UDLs into consideration, a BZ is modeled as $G_U = (V, A)$ $V = \{S, F, R\}$, where S, F, R represents ULRS, feeds and receivers respectively; A is the set of UDLs. The function of ULRS is to extract UDLs routing problems from the traditional BDLs routing problems. Placing several synchronized ULRS in a BZ can make SERP more reliable.

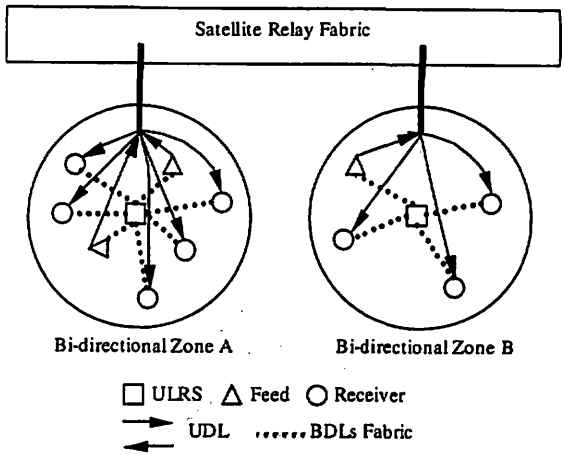

Figure 3 Architecture of Internet over Satellite links

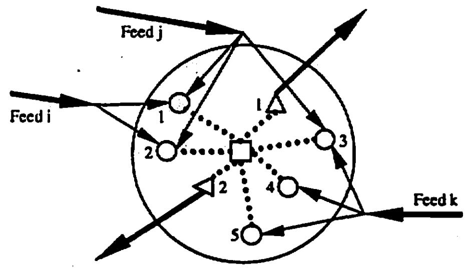

Figure 4 UDLs of a Bi-directional Zone

As satellite systems vary from coverage, all receivers within a BZ are categorized according to which feed they can hear from. Figure 4 shows an example of UDLs in a BZ, where *receiver 1, 2* can hear from *feed i*, *receiver 1, 2, 3* can hear from *feed j* and *receiver 3, 4, 5* can hear from *feed k*. *Feed i, j, k* may be *feed 1* or *feed 2* or the feed in another BZ. All receivers within the BZ are then divided into three categories, $\{R\}_1 = \{1 \quad 2\}$, $\{R\}_2 = \{1 \quad 2 \quad 3\}$ and $\{R\}_3 = \{3 \quad 4 \quad 5\}$. The downlinks from $\{R\}_1$, $\{R\}_2$, $\{R\}_3$ are regarded as *input links* (ILs) of the BZ, say, *IL1*, *IL2*, *IL3* respectively. The uplinks from *feed 1, 2* are regarded as *output links* (OLs) of the BZ, say, *OL1* and *OL2* respectively. Provided the UDL Registration Protocol (describes below) works correctly, ULRS is able to manage the information of *IL* and *OL*. As a result, the model of equation (1) is gotten, where m represents the number of *OL*, n represents the number of *IL*.

$$G_U = (\{S, F_m, \{R\}_n\},\ \{OL_m, IL_n\}) \qquad (1)$$

And the network shown in Figure 4 is simplified as that in Figure 5.

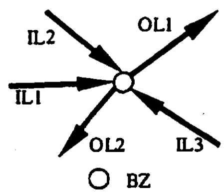

Figure 5 Simplification of Figure 4

Now, model of a network is divided into two planes: one is G_B within a BZ; the other is G_U between BZs. For G_B, we can deploy DV or LS algorithm; for G_U, we can deploy routing algorithm based on cycle discover method, such as [10]. The protocol coupling G_B and G_U is SERP.

4. Description of SERP

SERP comprises of two parts: *UDL Registration Protocol* (URP) and *Unidirectional Routes Forwarding Protocol* (URFP).

4.1 UDL Registration Protocol

URP is responsible for finding and maintaining status of OLs and ILs in a BZ. Supposed that each feed or receiver can detect its UDL interface by lower layer notification or other methods and can be aware of *the Network Identifier (generally is IP address) of its ULRS* (SNI) by some method. URP works as following:

Step 1. A feed, say F_i $1 \le i \le m$, registers *its Network Identifier of UDL interface* (F_iUNI) to ULRS and then broadcasts HELLO message that contains information of F_iUNI and *SNI* via its UDL periodically. *HELLO* message works with cycle discover procedure (URFP). While a cycle of the feed is discovered in G_U, *HELLO* message will floods over the cycle.

Step 2. A receiver, say R_i, registers to ULRS while received a *HELLO* message from a feed F_j directly or by flooding. The content of registration includes R_iUNI, F_jUNI and SNI extracted from HELLO message.

Step 3. By collecting all registrations, ULRS is aware of the network model in the form of equation (1). For each $\{R\}_k$ $1 \le k \le n$, ULRS sets $MIN\{R\}_k = R_i$, to which the cost is lowest. As all nodes in $\{R\}_k$ receive the same UDL routing information from IL_k, only forwarding the information at $MIN\{R\}_k$ is to avoid sending redundant packets. To achieve this goal, ULRS sends command *STOP_FORWARD* to all non-optimal receivers. If a feed or receiver is disconnected from G_B or reports its UDL failure, ULRS deletes the registration associated with it and chooses another one if $MIN\{R\}_k$ was deleted.

4.2 Unidirectional Routes Forwarding Protocol

Based on the results of [9] or [10], the routing of G_U can be established correctly by cycle discover procedure. We simply describe the difference of our proposal.

While found a cycle in G_U, URFP is responsible for forwarding unidirectional routes and introducing them to G_B. From the model of Figure 5, a node on a circle is the equivalent of a BZ. For each BZ on a cycle, there is a BZ being its predecessor and a BZ being its successor. The basic knowledge of unidirectional routes is that a feed in predecessor knows the reachability of all receivers that can hear from it. In URFP, a ULRS on a cycle floods such basic knowledge to the ULRS of its predecessor through its successors on the cycle path. Then, the ULRS of the predecessor introduces the reachability information to the associated feed. According to the routing algorithm deployed in G_B, SERP needs to do some auxiliary jobs while introducing, such as recalculates the cost to each destination network through UDL for DV algorithm or exchanges the UDL state of link state databases with its neighbors for LS algorithm. These jobs are easy to practice and their details are out of this paper.

Figure 6 shows an example network that contains three BZs. For a cycle, say $ABCA$, was established, C is predecessor of A and B is successor of A. S_A knows

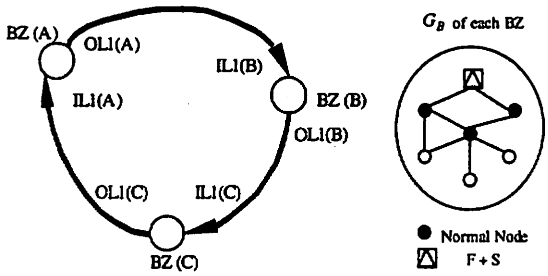

Figure 6 A 3-BZ Sample Network Topology

the reachability information $F_{1_C} \to \{R\}_{1_A}$ from URP and floods it to S_C through S_B in URFP. S_C introduces $F_{1_C} \to \{R\}_{1_A}$ to F_{1C} so that all nodes of BZ(C) know $F_{1_C} \to \{R\}_{1_A}$ with the help of the algorithm deployed in G_{B_C} and the auxiliary jobs done by S_C. To make use of $F_{1_A} \to \{R\}_{1_B}$ or $F_{1_B} \to \{R\}_{1_C}$ is the same way as above.

5. Performance Evaluation

To design a protocol for dynamic routing, the most important element is overload introduced by the protocol. The result of fewer overloads is better performance. In this section, we analyze the number of packets need to maintain correct routing in SERP.

The complexity of URP is based on the number of OL and IL. Packets need in URP are feed register packets, $MIN\{R\}_k$ chosen packets and receiver register packets. The sum of them is:

$$N_{URP} = \sum_{1\le i\le m} L(F_i) + \sum_{1\le i\le n} L(MIN\{R\}_i) + \sum_{1\le i\le n} \sum_{R_j \in \{R\}_i} L(R_j)$$

$L(V)$ is the number of links from V to each S

(2)

The performance of URFP is the same as [10] except for using BZ instead of node while cycle finding and maintaining. The complexity of URFP is based on cycle size. Packets need for each cycle is:

$$N_{URFP} = \sum_{BZ_i \text{ on the cycle}} L(F_{\Delta i}) + L(MIN(\{R\}_{\Delta i})$$

$V_{\Delta i}$ means F or R used by the cycle in BZ_i (3)

Compared with the complexity of [9] and [10], the searching procedure of cycle and flooding link state occurs in G_U rather than the whole network.

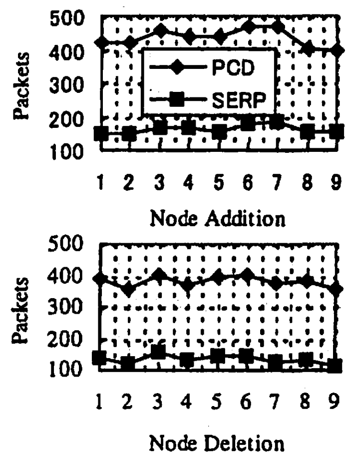

Figure 7 Simulation Results

6. Simulation

We simulated SERP and PCD [10] in *Network Simulator* version 2 to show how different in overload between them. Simulation ran for each protocol on the same topology of Figure 6. As there is only one feed in each BZ, nine nodes except feeds were chosen to do node addition and deletion events.

[10] is based on LS algorithm, in accordance with it, we chose LS as the routing algorithm deployed in BZs. Figure 7 shows the simulation results of summarized routing update packets after each single change, it indicates that SERP introduces fewer overloads than PCD.

7. Conclusion

This paper introduces a server (ULRS) to process UDLs information centrally in order to simplify the topology of the mixed network. While BDLs and UDLs are separated by ULRS, routing algorithm is able to be optimal by processing them differently. Moreover, the routing protocols running on the node that only has BDLs need not be changed. It is an advantage for easy deployment. SERP can be taken as a supplementation of current routing protocols. The limitation of SERP is that it aims at exploiting bi-directional islands connected via UDLs, especially for Internet over satellite systems, where the network is relative stable. For the scenarios where network topology changes continuously and UDL exists arbitrarily, such as ad-hoc network, it's better to use Demand-based protocol [11].

References

[1] Yongguang Zhang, "Integrating Satellite Networks with Internet Multicast Backbone (Mbone over Satellite)", *NASA Workshop on Satellite Networks: Architectures, Applications, and Technologies*, Cleveland, OH, June 1998.

[2] J. Moy, "Open Shortest Path First – OSPF version 2", Request For Comments 1583, Proteon, Inc, March 1994.

[3] G. Malkin, "RIP version 2 – Carrying Additional Information", Request For Comments 1723, Xylogics, Inc, November 1994.

[4] Y. Rekhter, T. Li, "A Border Gateway Protocol 4". Request For Comments 1771, IBM Corp. & Cisco Systems, March 1995.

[5] Robert McCurley, Fred B. Schneider, "Derivation of a distributed algorithm for finding paths in directed networks", Science of Computer Programming, 6(1): 1–9, 1986.

[6] Emmanuel Duros and Walid Dabbous, "Supporting Unidirectional Links in the Internet", In Proceedings of the 1st International workshop on Satellite-based Services, Rye, New-York, November 1996 INRIA Sophia-Antipolis, France.

[7] Emmanuel Duros and Christian Huitema, "Handling of Unidirectional Links with RIP", Internet draft, INRIA Sophia-Antipolis, March 1996.

[8] S. Hanks, "Generic Routing Encapsulation", Request For Comments 1701, NetSmiths Ltd, October 1994.

[9] Thierry Ernst and Walid Dabbous, "A circuit-based approach for routing in unidirectional links networks", Technical report, Inria, Institut National de Recherche en Informatique et en Automatique, 1997.

[10] L. Bao and J.J. Garcia-Luna-Aceves, "Link State Routing in Networks with Unidirectional Links," Proc. IEEE IC3N 99, Boston, Massachusetts, October 11--13, 1999.

[11] Joseph Macker and Scott Corson, http://www.ietf.org/html.charters/manet-charter.html.