Performance Evaluation of PIM-SM Recovery

Tarik Čičić[1], Stein Gjessing[1], and Øivind Kure[2]

[1] University of Oslo, Department of Informatics
P.B. 1080 Blindern, 0316 Oslo, Norway
{tarikc, steing}@ifi.uio.no
[2] Norwegian University of Science and Technology
Center for Technology at Kjeller
P.B. 70, 2027 Kjeller, Norway
okure@unik.no

Abstract. A PIM-SM-built multicast tree must be restructured/recovered when the underlying unicast routing tables change. In this article we describe the PIM-SM recovery mechanisms and evaluate the recovery performance, showing its dependence on a range of network and session parameters. Our results show that a significant recovery performance improvement is possible if the multicast recovery is immediately triggered when the unicast routing state changes. Furthermore, our results show that a substantial packet loss can be caused by non-reductive, "benign" events in the network, such as an addition of a new link.

1 Introduction

Stephen Deerings Ph.D. dissertation and the ensuing work in IETF on multicast protocols were the foundation for IP multicast [1, 2, 3]. The subsequent establishment of Mbone [4] positioned IP multicast as an emerging, powerful IP technology supporting a range of new, primarily multimedia applications. To address the inherent scalability problems of this technology, "Protocol Independent Multicast – Sparse Mode" (PIM-SM, [5]) was developed, and it is the most widely used multicast routing protocol today.

PIM-SM creates and maintains unidirectional multicast trees based on explicit Join/Prune protocol messages. These control messages are sent on a node-to-node basis. PIM is "protocol independent" in the sense that it is independent of the underlying unicast protocol — it can run on top of any unicast routing protocol.

To build a multicast tree, PIM multicast routers use a mechanism called Reverse Path Forwarding [6]. RPF determines the direction to the root of the tree using the unicast routing tables. This information is used to select an interface on which Join/Prune messages are sent, and where the multicast packets originated at the root are expected to arrive. Based on received Join/Prune messages, routers maintain a set of mappings between the input interface and the output interfaces for each known multicast group.

In case of unicast routing change, all multicast routing entries are reexamined using the RPF mechanism in order to determine the (possibly) new input

interface. This process of reestablishing the multicast tree we call *tree recovery*. If the new input interface differs from the old one, the multicast routing entry is updated: the new input interface is set instead of the old one and the new input interface is removed from the output interface list, if it was in it. Finally, control messages are sent to the neighboring routers: Join at the new input interface and Prune at the old input interface, if it is operational. In the transient phase, from the unicast routing change to the stabilization of the new multicast tree, packet loss may occur.

PIM-SM has received substantial attention in the research community [7, 8]. Also, significant research has been done on application-level error recovery for real-time IP multicast [9] and reliable multicast applications [10]. However, there has been less attention on the multicast tree recovery at the network level. Wang et al. [11] focussed on the performance of fault recovery in PIM Dense Mode running over OSPF. In addition, they analyzed the qualitative aspect of fault recovery of PIM running over OSPF. Our work extends these results and focuses on the performance of PIM-SM recovery.

2 Problem Statement

The performance of PIM-SM tree recovery is influenced by a range of factors, including the network topology properties (e.g. average node degree, link delay), multicast session properties (e.g. group size and data flow properties) and routing mechanisms (e.g. unicast routing protocol and multicast recovery initiation method).

In this paper we explore the effect of these parameters on the performance of PIM-SM recovery. In particular, the multicast recovery initiation can be based on periodic polling of the unicast routing tables (*periodic recovery*), or on receiving of an explicit change notification from the unicast routing process (*triggered recovery*). The periodic recovery is more common in practice, since it does not assume that the unicast routing is aware of the multicast routing. In our work we analyze performance and cost aspects of both mechanisms.

The unicast routing changes are caused by events belonging to three broad classes: *Topology Reduction*, e.g. link failure, removal or node failure, *Topology Enrichment*, e.g. link recovery or adding a new link and *Dynamic Routing Change*, e.g. link metric change. If topology reduction has occurred, the packet loss is often inevitable, since it takes time to reconstruct the multicast tree using alternative links. Intuitively, events belonging to the other two classes, called *benign events* in the rest of this paper, should not cause any packet loss. However, the standard PIM-SM recovery procedure implies that, in the case of a changed input interface, the old input interface is immediately disabled. In other words, events such as enrichment of the network by a new, operational link can also cause multicast packet loss. In this paper we evaluate the PIM-SM recovery performance both in the case of topology reduction (link failure) and a benign event (link recovery).

3 Performance Evaluation

We have developed a simulation model of PIM-SM [12] using the Network Simulator (NS) framework [13]. The model provides a general implementation of PIM-SM (routing based on explicit Join/Prune protocol messages, soft state with periodic refresh etc.) and a detailed implementation of the PIM-SM recovery [5]. The model is parameterized through a range of parameters including the average node degree, link delay, group density, CBR source rate etc. The unicast routing is based on the NS's standard distributed implementation of the Distance Vector protocol. We use random network topologies constructed to reflect real transport networks [14, 15].

In each simulation instance, after the multicast distribution tree has stabilized and the source has started to send data, a randomly chosen link within the multicast tree is taken down. This event we call "link-down" event. After the multicast tree has recovered, the link is reintroduced in the network ("link-up" event). We measure the packet loss in receivers caused by these events.

To evaluate the effect of the different parameters on PIM-SM recovery performance, we conduct a set of simulations where the parameters are varied within anticipated real network values. The following parameter ranges are chosen: recovery mechanism (periodic p=20ms, periodic p=50ms or triggered) average node degree (D={2.5, 3.0, ..., 5.0}) and group density (5, 10, 15, 20 receiver nodes out of 30 in the network). The average link delay in all test networks is 3ms, bandwidth 10Mb/s, CBR rate is 500packets/second and the packet length is 320Bytes.

3.1 Performance Evaluation Basis

In this subsection we first analyze unicast recovery. We find the average packet loss in a unicast data flow when a link goes down, under the same conditions as in the forthcoming multicast study. We will use these results as a comparison for the multicast recovery performance. Furthermore, we present how many multicast receivers are affected by the tree recovery and how often the packet loss occurs. These data are significant for a proper evaluation of the multicast packet loss figures presented later in this section.

We believe that it is only of interest to consider simulation instances where tree recovery after link-down is possible. Hence, we are not considering simulation instances where the link-down event resulted in disconnected topology.

Unicast Loss. Our study is performed in networks using a unicast routing protocol based on the Distance Vector (DV) algorithm. When DV is used, each node has sufficient information to immediately repair the failed route if and only if the alternative route is two hops long. If three or more hops are necessary, the upstream node will discard the packets while the routing updates are exchanged and the routing state converges. This period will be longer in sparse networks due to longer alternative routes.

If a Link State (LS) protocol implementation is used, the unicast packet loss is not dependent on the average node degree. The unicast routing update flooding starts as soon as the link failure is detected. After receiving the update, each node can calculate the alternative route instantly. Therefore the packet loss will occur mainly due to the loss of packets traversing the faulty link, and it will be lower than the corresponding one for a DV routing protocol.

We have measured the DV unicast recovery performance as the packet loss in a unicast flow with same properties as the tested multicast flow (10Mb/s CBR flow, 500packets/second, 320Bytes packets). The unicast flow traverses the same faulty link as the multicast flow presented in the remainder of this section.

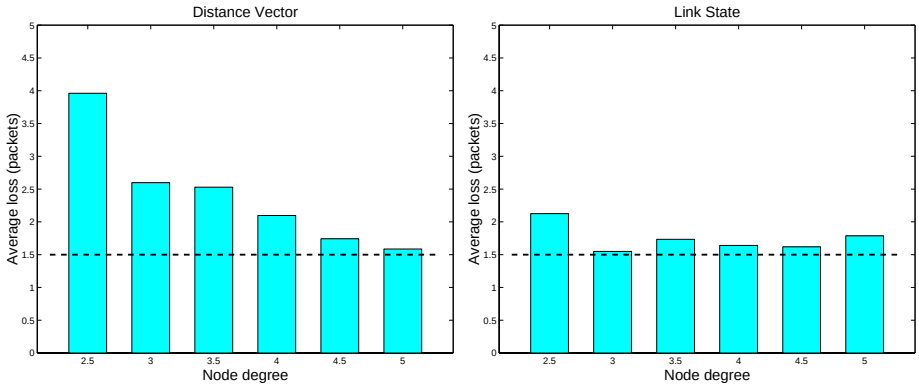

Fig. 1. Unicast packet loss, depending on the average node degree. The expected on-link loss is 1.5 packets because NS excludes the link transmission time in its loss model

Figure 1 (left) shows the Distance Vector unicast loss depending on average node degree. Our results provide a good illustration of the quick recovery in highly connected networks. In 30-nodes networks with the average node degree of 5, the probability of having a two-hop alternative path for a link failure is very high. Hence, the DV packet loss for degree 5 is expected to be just above the minimum, estimated loss of the packets traversing the link:

$$L_{\min} = R \cdot (d_{\mathrm{p}} + d_{\mathrm{t}}) = 500\ \mathrm{s}^{-1} \cdot 0.003256\ \mathrm{s} = 1.628 \qquad (1)$$

where R is the packet rate, d_{p} is the 3 ms link propagation delay and d_{t} is the 0.256 ms packet transmission time. In our simulation environment the minimum loss is even lower than the estimated minimum (1), since the NS loss model implementation excludes the packet transmission time.

As expected, the LS packet loss is independent of the average node degree (Fig. 1, right).

Affected Receivers. A link failure within the multicast tree will always affect at least one receiver. It is important to present how many receivers are affected in order to gain a complete view of the recovery performance.

The multicast trees are higher (more hops on average from the source to each of the receivers) in the networks with low connectivity than in the networks with high connectivity. Therefore, a link failure is more likely to affect a receiver in a network with low connectivity than in a network with high connectivity.

Figure 2 (left) shows the average number of receivers affected by a single failure. For example, 33% receivers are affected in networks with the average node degree D=2.5 and with group size 5, and only 10% in D=5.0 networks with 20 receivers.

Sources for Packet Loss. Packet loss may occur due to both link-down and link-up events. The link-down event causes loss in 95% cases in the triggered recovery and almost always in the periodic recovery, regardless of the network parameters. The link-up event, however, causes packet loss only if the input interface has changed, which is more probable in low connectivity networks. This is the case since the alternative paths in the high connectivity networks will in general be shorter and "closer" to the original path — achievable through change of output interfaces in the transit nodes only.

The link-up event causes loss from ∼50% cases in D=5 networks to 80% cases in D=2.5 networks (Fig. 2, right).

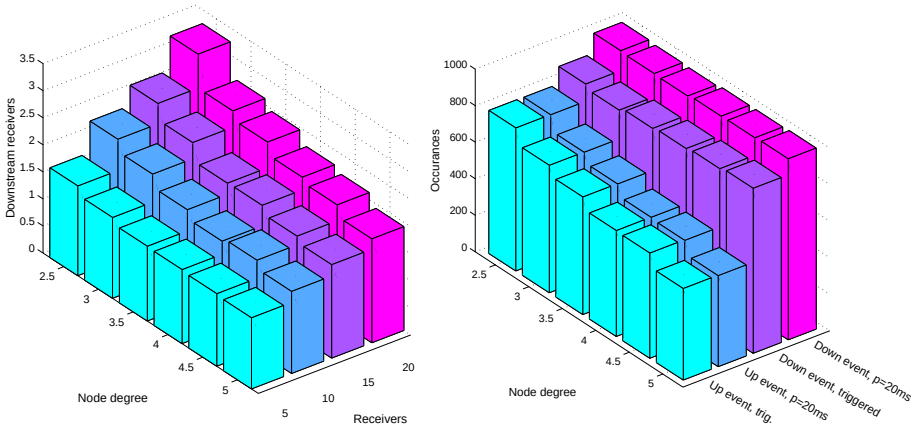

Fig. 2. Consequences of tree recovery: mean number affected receivers (left) and events causing the packet loss out of 1000 simulation instances (right)

3.2 Link-Down Event

When a branch is removed from the multicast tree, the downstream nodes will be cut off until an alternative route is established. The total cutoff time T is bounded:

$$T_{\mathrm{m}} \leq T \leq T_{\mathrm{u}} + p + T_{\mathrm{m}}$$

where T_{u} is the time it takes to recover the unicast routing, p is the unicast routing check period (20ms and 50ms in our simulations, zero for the triggered recovery) and T_{m} is the multicast routing recovery time (upstream propagation of the Join-messages to the closest node in the tree).

T_{u} is shorter for larger average node degrees. T_{m} decreases as the probability of a nearby in-tree node increases, influenced by the number of receivers and the average node degree.

We expect T to perform much better on average than the worst case. First, the expected time before the multicast recovery starts is $p/2$. Also, the unicast recovery time is often included in this period. Furthermore, the multicast recovery may start before the unicast is completely recovered in the triggered recovery. This happens because the unicast routing recovery takes several routing message exchanges to stabilize, and that the multicast recovery succeeds quickly since the neighboring node may be a member of the same multicast group. In the process of unicast routing, the multicast input interface may temporarily point in wrong direction. This has no effect on the final multicast routing entry, since it is always coherent with the unicast routing.

Our performance evaluation results are shown in Fig. 3. For the same recovery mechanism and number of receivers, each sextuple of adjacent bars represents the six average network degrees (2.5 to 5) we have tested. The standard deviation in these measurements ranges from ∼2.5 packets for the triggered recovery with 20 group members to ∼7.5 packets for the periodic recovery with 5 group members.

The unicast loss pattern (Fig. 1, left) is recognizable in the charts for low group sizes. For higher group sizes, the multicast recovery often suceeds before the unicast is completely recovered due to the high probability of the neighbor node beeing a member of the multicast group, thereby obscuring the unicast loss pattern.

The effect of the node degree and the group size is shown in the characteristic pattern where the performance increases by ∼3 packets from D=2.5 networks with 5 receivers to D=5.0 networks with 20 receivers, for both periodic and triggered recovery.

We can observe that the loss performance is dominated by the unicast routing check period p: the mean loss value for the triggered, periodic p=20ms and periodic p=50ms recovery is 4.5, 8.8 and 15.8 packets, respectively. The difference between the first two is 4.3 packets. The expected time between the link down event and the recovery procedure initiation is $p/2$=10ms or 5 packets. The 0.7 packet difference is caused by the overlap between the unicast and multicast recovery.

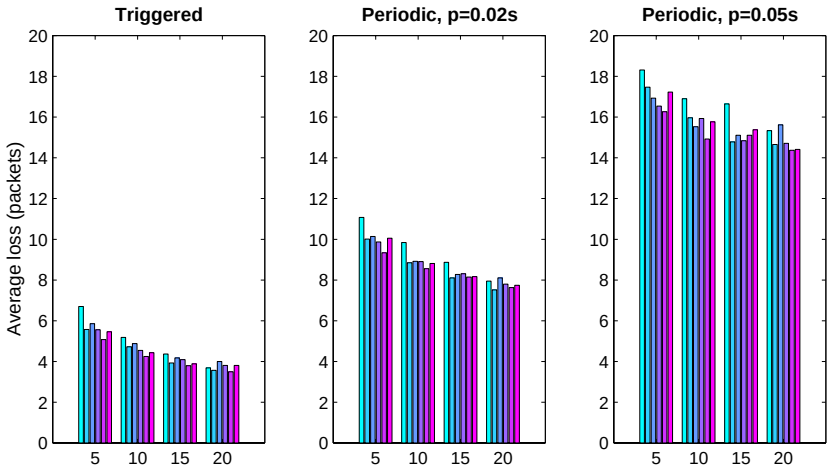

Fig. 3. Mean packet loss per affected receiver, link-down event. Each sextuple of adjacent bars represents the six average node degrees: 2.5 (leftmost) to 5 (rightmost) links per node, for group sizes 5, 10, 15 and 20 receivers

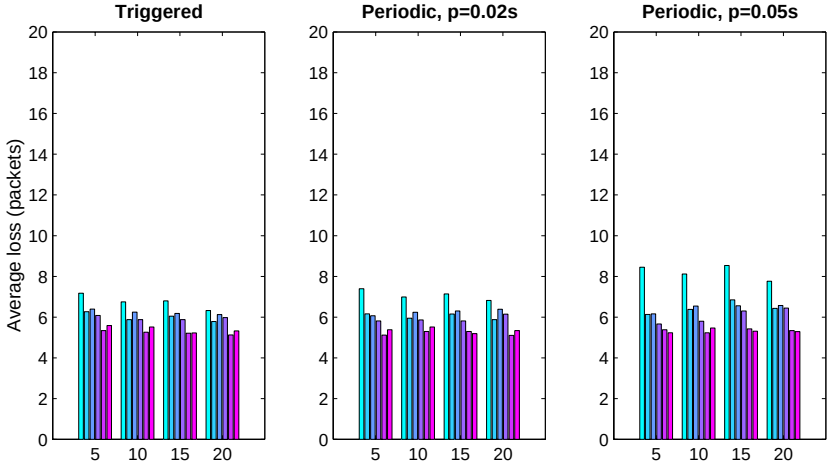

Fig. 4. Mean packet loss per affected receiver, link-up event. Each sextuple of adjacent bars represents the six average node degrees: 2.5 (leftmost) to 5 (rightmost) links per node, for group sizes 5, 10, 15 and 20 receivers

3.3 Link-Up Event

When a network link recovers, the unicast routing tables are updated for routers that have the link in a shortest path route. In our scenario, the unicast routing tables become the same as before the link-down event. The multicast routing process notices this benign event, and starts the recovery procedure in order to reestablish the better multicast tree.

The PIM-SM recovery procedure implies that the old input interface on a router is closed instantaneously when the new input interface is chosen. It takes time for the multicast flow to propagate over the new branch. The packet loss in this case is dependent on the branch propagation delay, which has fewer hops and a shorter delay in networks with high node degree.

The mean packet loss caused by the link-up event is shown in Fig. 4. The packet loss is largely independent of the recovery period, since the old input interfaces are operational and unchanged even though the unicast routing changes in this period.

4 Overhead Comparison

PIM-SM tree recovery includes the multicast routing table recalculation and the exchange of Join/Prune control messages on the new links. These actions will respectively cause additional router CPU consumption and the network load increase. We provide an estimate of how often the overhead is incurred for the the two recovery types (periodic/triggered).

Computational Overhead. Each time the unicast routing state has changed, an RPF check has to be done for each multicast routing entry. The triggered recovery will be invoked only when the changes have occurred, justifying the routing table processing. If Distance Vector unicast routing is used, the procedure can be invoked repeatedly as the unicast routing stabilizes. This will additionally stress the system in the transient phase.

The periodic recovery needs to know if there has been any changes in unicast routing tables since the last invocation of the recovery procedure. If this is impossible, because e.g. the unicast routing is unaware of the coexisting multicast routing and does not release the last update information, the only implementable solution is the least effective, periodic recovery with unicast routing table processing in each invocation.

Communication Overhead. In PIM-SM recovery the communication overhead consists of two parts, the transmission of packets that are rejected due to wrong input interface and the Join/Prune messages triggered by multicast routing changes. The PIM-SM standard specifies sending the prune message on the old input interface and merging Join/Prune messages for many multicast groups, thereby minimizing the communication overhead.

The periodic recovery will send at most one Join/Prune message, and only if the input interface has changed. The triggered recovery may send several Join/Prune messages, as the DV routing stabilizes.

Repeated Recovery Invocations in Triggered Recovery. To understand the amount of the additional overhead in transient period in the triggered recovery, we have counted the number of calls and the number of input interface changes in our simulator. We have tested D=3.0 topologies with a single, five-member group.

The link-down event caused an average of 1.75 recovery procedure invocations per multicast node. A maximum of 13 invocations was registered, however, 4 or fewer invocations were registered in more than 95% cases. A maximum of 5 Join/Prune messages per node was sent. In 75% cases the input interface remained the same (zero Join/Prune messages sent), and in additional 20% cases a single Join/Prune was sent.

This result shows that, in our DV simulation environment, the triggered recovery induced a 75% higher computational overhead and a slightly higher control message overhead, as compared to the periodic recovery.

Link State Routing. A Link State unicast routing protocol will receive at most one unicast routing update for each event (e.g. single link removal). This implies that at most one multicast recovery procedure invocation will occur, even in the triggered recovery. In other words, in LS-based networks, the triggered and the periodic recovery will have similar computational and communication overhead.

5 Conclusion

We have evaluated the PIM-SM recovery performance depending on the recovery mechanism and various topology and session parameters. Packet loss occurs due to both reductive and benign events. We simulated a reductive event as a link failure (link-down event) and a benign event as the link recovery (link-up event).

The link-down event causes packet loss in at least 95% cases in our test environment, regardless of the other parameter settings. The triggered recovery has superior performance as compared to the periodic recovery. The triggered recovery will in general have computational and communication overhead of the same order as the periodic recovery, but may not be implementable on some systems. Other factors (e.g. average node degree) have a moderate effect on the performance. In general, PIM-SM recovers quickly, showing performance close to the underlying unicast recovery.

The packet loss caused by the link-up event is unnecessary high, and can be decreased using an improved recovery algorithm. Detailed specification and analyze of this algorithm is the topic of our current research.

References

[1] Steve Deering. *Multicast Routing in a Datagram Internetwork*. PhD thesis, Stanford University, 1991.

[2] David Waitzman, Craig Partridge, and Steve Deering. Distance vector multicast routing protocol. RFC 1075, November 1988.

[3] Steve Deering. Host extensions for IP multicasting. RFC 1112, August 1989.

[4] Hans Eriksson. MBONE: The multicast backbone. *Communications of the ACM*, 37(8), August 1994.

[5] Deborah Estrin, Dino Farinacci, Ahmed Helmy, David Thaler, Stephen Deering, Mark Handley, Van Jacobson, Ching gung Liu, Puneet Sharma, and Liming Wei. Protocol independent multicast – sparse mode (PIM-SM): Protocol specification. RFC 2362, June 1998.

[6] Yogen K. Dalal and Robert M. Metcalfe. Reverse path forwarding of broadcast packets. *Communications ACM*, 21:1040–1048, December 1978.

[7] Liming Wei and Deborah Estrin. Multicast routing in dense and sparse modes: simulation study of tradeoffs and dynamics. In *Computer Communications and Networks, Fourth International Conference on*, pages 150–157. IEEE, September 1995.

[8] Tom Billhartz, J. Bibb Cain, Ellen Farrey-Goudreau, Doug Fieg, and Stephen Gordon Batsell. Performance and resource cost comparison for the CBT and PIM multicast routing protocols. *IEEE J. Selec. Areas Commun.*, 15(3):304–315, April 1997.

[9] Georg Carle and Ernst W. Biersack. Survey of error recovery techniques for IP-based audio-visual multicast applications. *IEEE Network*, pages 24–36, November 1997.

[10] Christophe Diot, Walid Dabbous, and Jon Crowcroft. Multipoint communication: A survey of protocols, functions, and mechanisms. *IEEE J. Selec. Areas Commun.*, 15(3):277–290, April 1997.

[11] Xin Wang, C. Yu, Henning Schulzrinne, Paul Stirpe, and Wei Wu. IP multicast fault recovery in PIM over OSPF. In *Proceedings ACM SIGMETRICS*, June 2000.

[12] Tarik Čičić, Stein Gjessing, and Øivind Kure. Tree recovery in PIM sparse mode. Research Report 293, University of Oslo, Department of Informatics, March 2001. ISBN 82-7368-243-9.

[13] UCB/LBNL/VINT. Network simulator - ns (version 2). WWW. http://www.isi.edu/nsnam/ns/.

[14] Bernard M. Waxman. Routing of multipoint connections. *IEEE J. Selec. Areas Commun.*, 6(9):1617–1622, December 1988.

[15] Ellen W. Zegura. Georgia tech internetwork topology models. WWW. http://www.cc.gatech.edu/fac/ Ellen.Zegura/graphs.html.