Extension of RP Relocation to PIM-SM Multicast Routing

Ying-Dar Lin, Nai-Bin Hsu, Chen-Ju Pan
Dept of Computer and Information Science, National Chiao Tung University, Hsinchu, Taiwan.
E-mail:{ydlin, gis84811, gis87550}@cis.nctu.edu.tw

Abstract— **The Protocol Independent Multicast-Sparse Mode (PIM-SM) protocol establishes core-base tree to forward multicast datagrams in a network. In PIM-SM, the core or *Rendezvous Point* (RP) of a group is determined at each multicast router by hashing a group address, i.e., a class-D IP address, to one of the candidate RPs. The hash function is characterized by its ability to evenly and uniquely choose the core for a group and remains insensitive to the geographic distribution of the group members and the sources. However, it may result in a multicast tree with high cost.**

This study presents a relocation mechanism which is extension to PIM-SM, in which RP could be relocated periodically. When a new RP is found, the original RP informs all members to re-join to the new RP. Simulation results indicate that the extended version, *RPIM-SM*, reduces about 20% tree cost than PIM-SM when the group size is medium. Moreover, comparing *RPIM-SM* with the optimal core-based tree reveals that they have less than 5% difference in tree cost. Furthermore, an increase of the number of candidate RPs brings *RPIM-SM* even closer to the optimal core-based tree. Results in this study demonstrate that relocation improves the performance of PIM-SM.

Keywords— **PIM-SM, Rendezvous Point, relocation, RPIM-SM**

I. Introduction

Multicast routing protocols can be categorized as source-based tree and shared-tree protocols. A source-based tree protocol builds separate trees for each (source, group) pair, that is, each source has its own tree that reaches the active group members, such as DVMRP [1], PIM-DM [2], and MOSPF [3]. On the other hand, shared-based protocols such as PIM-SM [4] and CBT [5] build distributed trees having a central point (or core) to whom all receivers attached. Typically, a source sends datagrams to the RP and RP forwards them to all members through the RP-based multicast tree. Therefore, a shared-tree router only needs to maintain state information for each group instead of for each (source, group) pair.

In PIM-SM, new members wanting to join a group send *Join* messages to a core, called *Rendezvous Point* (RP), of the distribution tree. The RP administers the specific multicast group(s), and facilitates the joining/leaving of the group members. The address of the RP is determined in each multicast router by mapping a group to one of the candidate RPs with a hash function. However, the chosen RP may be inappropriate for its group, for example, causes more tree cost. The hash function is characterized by its ability to evenly and uniquely choose a candidate RP to be a core in order to balance the service load of each candidate RP in the network.

This study proposes an RP relocation mechanism which is extension to the PIM-SM multicast routing protocol. The hash function of PIM-SM is initially used to obtain an RP for the multicast group. As the sources come and go, RP relocates its location periodically afterward. An attempt is made to obtain an appropriate core location by using the estimated tree cost function to evaluate the appropriateness of the candidates. When RP relocation is determined, the original RP multicasts NEW_RP message to all members to inform them to re-join to the new RP. Consequently, a new distribution tree is built with the least cost.

The rest of this work is organized as follows. Section II presents the issues of PIM-SM protocol and our motivations. Section III describes the details of the RP relocation mechanism. Section IV presents the control messages in addition to PIM-SM. Next, Section V provides the simulation models and results. Conclusions are finally drawn in Section VI.

II. Issues and Motivations

PIM-SM is a commonly used multicast routing protocol that provides efficient communication for multicast groups with sparsely distributed members. The designers observed that several hosts wanting to participate in a multicast conference do not justify having their group's multicast traffic periodically broadcast across the entire network. To eliminate the scaling problem, PIM-SM is designed to limit multicast traffic so that only routers interested in receiving traffic for a particular group will receive it. By unicast routing, the source router knows how to reach and forward the traffic to the RP. Then, RP distributes the traffic to all the members through the RP-based multicast tree.

In order to broadcast the set of candidate RPs to the network nodes, the *bootstrap router* (BSR) is elected for the domain. BSR originates *Bootstrap* messages to distribute the set of RPs information, which are distributed hop-by-hop throughout the domain. There is only one RP-set per PIM-SM domain. By using a hash function, say $Hash()$, each router can uniquely map a group address G to one of the routers in the RP-set. That is, candidate C_k is chosen as RP if C_k yields highest hash value $Hash(G, M, C_i)$, for all C_i belong to the $RPset$. The hash mark, M, allows a number of consecutive groups to

TABLE I

SYMBOL DEFINITIONS.

V: set of routers	$hashRP$: initial hashed RP of G
E: set of links	$currentRP$: current RP of G
S: set of sources	$newRP$: RP to be migrated
R: set of members	$rFlag$: relocation flag
G: multicast group	P_c, P_e: Probability functions
M: hash mask	$RPset$: set of candidate RPs
m: group member	C_i: ith candidate RP, $C_i \in RPset$
t_r: relocation timer	C_k: min cost candidate, $C_k \in RPset$
TC: cost of multicast tree	k: current # of members in G
u, v: networks nodes, $u, v \in V$	q: cost reduction threshold
n: # of nodes in the network	$dist$: distance or hop count
n_d:# duplicate dist. node in S	deg: node degree or connectivity

resolve to the same RP.

The shared tree, although provides better scalability, does not optimize the delivery path through the network. RP for the group is typically designed without respect to its location. Thus, an RP could be located for away from all group members, resulting in inefficient transmission.

Therefore, in this study, assume RP_i is the current RP of G, we propose an RP relocation mechanism which relocates RP when the tree cost reduction, $TC(RP_i) - TC(RP_{i+1})$, if it is larger than a pre-defined threshold, q. In addition, three control messages are needed to migrate to the new RP and reliably maintain the membership of the group. For convenience of our description, Table I summarizes the symbols used in this study.

III. RP RELOCATION ALGORITHM

Of priority concern in a distribution tree, the lower the tree costs of hop count and delay implies better routing paths. Tree cost is defined as $\sum_{(u,v)\in T} cost(u,v)$, i.e., the sum of the costs of the links in the multicast tree. Our mechanism attempts to reduce the tree cost by relocating the RP for a multicast group. To map an appropriate RP in a group, this study uses an estimated function in [6] to obtain a RP with the minimum tree cost from the $RPset$ of a group. Equation (3) calculates the distribution tree cost if the candidate C_i is taken as the RP, $TC(C_i)$, by taking the average of Eq. (1) and (2), i.e., the maximum and minimum bounds on tree cost. These equations use the distance (i.e. $cost : E \mapsto \mathbf{R}^+$) for each possible destination as the metric. Notably, the distance information is already available to routers.

$$TC_{min}(C_i) = max_{u\in S} dist(C_i, u) + n_d \quad (1)$$

$$TC_{max}(C_i) = \begin{cases} \sum_{u\in S} dist(C_i, u) & \text{if } |S| \le deg(C_i) \\ \sum_{u\in S} dist(C_i, u) - (|S| - deg(C_i)) & \text{otherwise} \end{cases} \quad (2)$$

$$TC(C_i) = \frac{TC_{min}(C_i) + TC_{max}(C_i)}{2} \quad (3)$$

The best-case tree is linear if a lower bound on the cost, TC_{min}, of a tree rooted at some node is obtained. The cost of the tree is simply the maximum distance from RP to any sender. When giving the hop counts (i.e. cost:$E \mapsto$ 1) as the metrics, the function can obtain a tighter bound by adding the number of members that are at an equal distance. This bound is owing to that the distribution tree cannot be completely linear, but must have at least an additional link.

Opposite to the lower bound of the cost, the upper bound on the cost, TC_{max}, of a tree rooted at some node is that no links are shared among the paths to each member. Therefore, the maximum tree cost is the sum of the member distances. Also, if the number of group members is greater than the root degree, the bound is tightened by subtracting the difference to calculate the cost for the sharing links. The estimation function is thus defined as the average of the sum of the minimum cost and maximum cost.

The algorithm in Fig. 1, a new member m that joins a group sends a $Join$ message to the hashed RP. The $hashRP$ verifies whether if RP of this group has been migrated (i.e. the $rFlag$ is $true$). If so, $hashRP$ redirects m to the new RP by sending the NEW_RP message. Thus, m sends a $Join$ message and established (*, G) state along the path towards the new RP. In addition,

RPIM-SM(G)
set_of_sources S;
set_of_members R;
member $m \in R$;
relocation_flag $rFlag$;
relocation_timer t_r;
tree_cost_function TC;
Begin
  while (1) **do**
    if (m wants to join G) **then**
      $hashRP \leftarrow HashFind(G, RPset)$
      m send $Join$ to $hashRP$
      $hashRP$ check **if** ($rFlag == true$) **then**
        $hashRP$ unicasts $NEW\_RP(currentRP)$ to m
        m send $Join$ to $currentRP$
          and send $Prune$ to $hashRP$
      endif
    endif
    $currentRP$ check **if** (t_r expired and S changed) **then**
      $newRP \leftarrow$ Select C_k from $RPset$,
          where $TC(C_k)$ is minimum
      if reduction of $TC(C_k) > q$ **then**
        $ChangeRP(currentRP, newRP, G)$
      endif
    endif
  endwhile
End

Fig. 1. The RP Relocation.

```
ChangeRP(currentRP, newRP, G)
set_of_members R;
member m ∈ R;
relocation_flag rFlag;
Begin
  currentRP check if (newRP == hashRP) then
    rFlag ← false /* return back to initial RP */
  else
    rFlag ← true
  endif
  currentRP multicasts NEW_RP(G, newRP)
    to {hashRP, sources, members} of G
  Upon receiving NEW_RP message
    begin
    newRP send I_AM_RP message to hashRP
    each source re-register to the newRP
    member m send Join to newRP
      and send Prune to currentRP
    end
  currentRP ← newRP
End
```

Fig. 2. The RP Migration.

m sends a $Prune$ message to the $hashRP$ to discard the previous path. The $currentRP$ also confirms whether if the relocation timer t_r has expired and the set of sources of the group has been changed. If it has, a new RP with significant cost reduction is computed by the above cost functions TC for each candidate RP.

Once the RP with least cost for a group is found, the function **ChangeRP** is invoked, as shown in Fig. 2. The $currentRP$ first checks whether the new RP is the $hashRP$. If it is, the $rFlag$ is reset to $false$ since the $currentRP$ returns back to the original hashed RP. To migrate the distribution tree, message NEW_RP is advertised by the $currentRP$ to inform the $newRP$, $hashRP$, sources, and all members of the group.

For encoding this message, each node with (*,G) state maintains a relocation table to record the state of RP relocation, as well as the addresses of the new RP and old RP, as shown in Table II, $newRPAddr$ and $oldRPAddr$, respectively. The $rFlag$ indicates whether if RP of this group has been relocated, i.e., not the original hashed RP currently. When a new RP is obtained, the previous $newRP$ becomes the $oldRP$ and its address is moved to the $oldRPAddr$; the new RP address is then saved at the $newRPAddr$. Notably, the corresponding incoming interface (iif) and the Reverse Path Forwarding (RPF) neighbor ($nbrRPF$) of the $newRPAddr$/$oldRPAddr$ are also saved. State information of the relocating process prevents the chaos caused by RPF checking during the transition between the two consecutive multicast trees.

TABLE II
Relocation State Table.

G	$rFlag$	$newRP$	iif	$nbrRPF$	$oldRP$	iif	$nbrRPF$
G_1	$true$	RP_1	i_1	v_1	RP'_1	i'_1	v'_1
G_2	$true$	RP_2	i_2	v_2	RP'_2	i'_2	v'_2
	...						
G_n	$false$	RP_n	i_n	v_n			

For example, a node with (*, G_1) state continues to distribute packets from the old RP (RP'_1) at the interface i'_1, until this node receives packets from the new RP (RP_1) at the interface i_1. Furthermore, in order to migrate the distribution tree of the group G_1, according to Table II, the current RP (RP'_1) of G_1 extracts the $newRPAddr$ RP_1 from the table and multicasts the encoded NEW_RP message towards the $newRP$, all sources and members of group G_1. This message redirects all sources to re-register, all members to re-join to the new RP. Note that in case of G_n, since the $rFlag$ is $false$, this entry records $hashRP$ of G_n at the $newRPAddr$ field.

Fig. 3 shows the RP state transition of a specific group G. At the system start up, the RP is in the $Hashing$ state and enters the $Selection$ state as timer t_r expires. If the selected candidate (C_k) can significantly reduce the current tree cost, **ChangeRP** is invoked and the $Transition$ state is entered, and thereafter transits further to the $Relocation$ state or $Hashing$ state, depending on whether the newly selected RP is the original hashed RP. Returning back to the $Hashing$ state means that the original hashed RP turns out to be the best RP for G again. On the other hand, if the reduction in tree cost does not exceed the threshold, q, the RP enters the $Relocation$ state or $Hashing$ state, depending on the value of $rFlag$ where $true$ means this RP is a relocated, instead of hashed, one. Note that the *RPIM-SM* verifies the source list, S, and the tree cost of G every t_r time unless the system re-startup.

IV. Control Messages

A. NEW_RP

This message is used to advertise to the group that who is the new RP from now on. When the computation of tree cost in Fig. 1 is done and the decision of the RP relocation is positive, the current RP sends the message $NEW\_RP(G, newRP)$ to the $newRP$, $hashRP$, and sources, and multicasts it to all members of the group. Then, the migration process is triggered, i.e., the sources re-register to the $newRP$ and members re-join to the $newRP$.

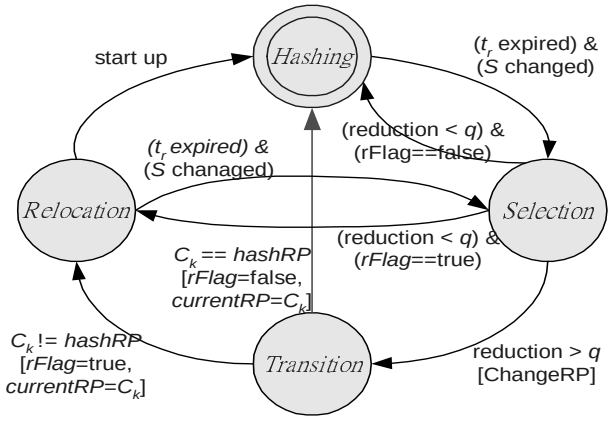

Fig. 3. State transition diagram of the RP.

B. *I_AM_RP and RP_CONFIRM*

Using this message, *newRP* informs the *hashRP* that "I am the RP of the group at this time." Without this information at the *hashRP*, the incoming receivers could not find the current RP to whom the *Join* message send. Upon receiving *I_AM_RP* message, *hashRP* updates the corresponding relocation state of G, and acknowledges the *newRP* with the *RP_CONFIRM* message.

V. Simulation Model And Results

A. Network Model

Simulations are performed to evaluate the performance of the proposed *RPIM-SM* extension to the PIM-SM multicast routing. Random graphs [7], [8] are used to simulate network models in order to ensure that the effects of the different routing algorithms are independent of any specific network. Graphs are generated with an average node degree of 4. Unless otherwise specified, the number of nodes in networks is 100. After each graph has been generated, Prim's or Kruskal's algorithm is used to ensure that the random graph comprises of only one component.

The sequence of events of *join/leave* is generated by a simple probability model [7]. The probability that a adding a node to the multicast group is determined by the probability function is defined as follows:

$$P_c(k) = \frac{\gamma(n-k)}{\gamma(n-k) - (1-\gamma)k} \tag{4}$$

where n is the number of nodes in the network, k is the current number of group members, and $\gamma = k/n$ is a parameter in the range (0, 1) which determines the fraction of nodes in the connection at equilibrium [9]. For example, $P_c(k) = 1/2$ when $\gamma = k/n$. The value of γ that determines the number of nodes in the multicast group is set to 0.5; that is, half of the nodes in the graph are in the multicast group in equilibrium.

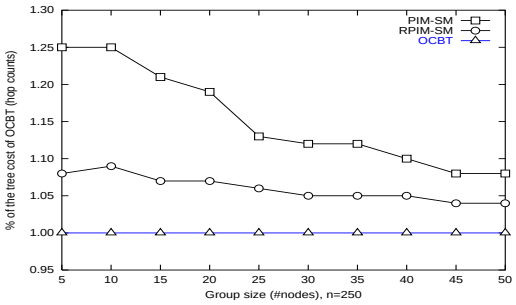

Fig. 4. Tree cost comparison (hop counts).

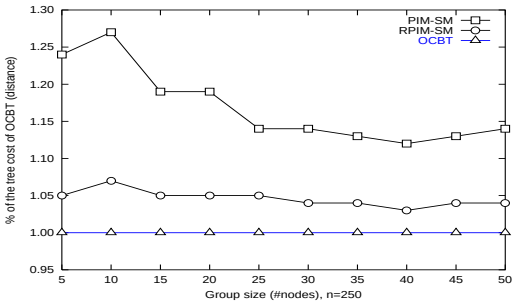

Fig. 5. Tree cost comparison (distance).

B. Simulation Results

B.1 Tree cost comparison

Fig. 4 and 5 compare the tree costs of PIM-SM and *RPIM-SM* with the cost of the optimal center-based tree (OCBT). The tree cost of the OCBT calculates the actual cost of the tree rooted at each node in the network and the one with the lowest maximum length among all of those with the lowest cost is selected. Fig. 4 shows the average results of 100 simulations in a 100-node network. Notably, the Y-axis plots the ratio between tree cost of the PIM-SM or *RPIM-SM* to the tree cost of the OCBT. The hash function in PIM-SM does not consider the geographic distribution of members, accounting for why the *RPIM-SM* performs better than PIM-SM, i.e., closer to the cost of OCBT, particularly in a small group size. With a larger group size, there is a higher likelihood of the hashed RP near to the center of the group, thereby decreasing the tree costs of both mechanisms. Hence, for various group sizes, *RPIM-SM* performs better than PIM-SM.

Similar to Fig. 4, Fig. 5 compares the tree costs of PIM-SM and *RPIM-SM* using the distance metric. The tree cost direction of the *RPIM-SM* is more stable than the cost of PIM-SM. The behavior resembles that found in Fig. 4. Our results indicate that when the group size is under 10, the relocating RP renders about 20% reduction in tree cost. The curve clearly reveals the merits of *RPIM-SM*.

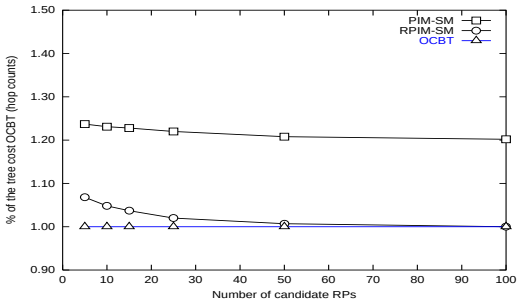

Fig. 6. Tree cost vs. number of candidate RPs.

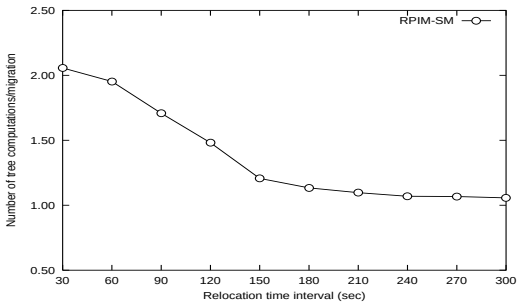

Fig. 7. Effects of relocation time interval.

B.2 Number of Candidate RPs

Fig. 6 describes the tree costs versus the number of candidate RPs in the group and also contains the experimental results with a fixed group size of about fifteen and 100 nodes. Both PIM-SM and *RPIM-SM* reduce the trees cost with increase of the number of candidate RPs. However, *RPIM-SM* is more effective on tree cost reduction. In particular, as all network nodes are candidates, tree cost of the *RPIM-SM* is the same as that of OCBT.

B.3 Time Interval of RP Relocation

Obviously, in *RPIM-SM*, computing the RP relocation causes additional overhead. To minimize the additional consumption, how long is the relocation interval t_r should be set must be determined. Fig. 7 shows the ratio of number of relocation computations to the number of RP migrations on average. These statistics depend on the cost reduction threshold, q, which is set to 10% in our experiments. According to this figure, as the time interval t_r is small, more than one computation is required per each RP migration. When the time interval exceeds 150 seconds, the curve approaches to 1.0. This finding suggests that the time interval of relocation was sufficient so that the migration almost took place when relocation computation is executed. Based on our simulation model, the time period t_r can be set at every 150 seconds to verify whether if the RP of each group needs to be relocated.

VI. Conclusions

This paper proposes a dynamic mechanism, *RPIM-SM*, to extend the PIM-SM multicast routing protocol to RP relocation. The original hashed *Rendezvous Point* (RP) location may be inappropriate for the group. Therefore, according to the estimated tree cost, the PIM-SM is extended by relocating the RP periodically. In addition, the candidate RP with minimum tree cost is selected to be the new RP of the group. To migrate to the new RP, a additional message, *NEW_RP*, is needed to notify sources to re-register, and members to re-join the new RP.

Simulation results indicate that *RPIM-SM* reduces about 20% in tree cost of PIM-SM when group size is around 10. Moreover, when we compare *RPIM-SM* with the optimal core-based tree, *RPIM-SM* has less than 5% difference in tree cost. When the number of candidate RPs increases, *RPIM-SM* is even closer to the optimal core-based tree. In addition, under the network size of 100 nodes and the dynamic group membership, simulation results further indicate that 120–150 seconds would be appropriate as the time interval of possible RP relocation.

Acknowledgments

The authors would like to acknowledge the suggestions of the anonymous reviewers.

References

[1] D. Waitzman, C. Partridge, and S.E. Deering, "Distance vector multicast routing protocol," RFC 1075, IETF, Nov. 1988.

[2] S. Deering, D. Estrin, D. Farinacci, V. Jacobson, A. Helmy, and L. Wei, "Protocol indepenent multicast version 2, dense mode specification," *Internet Draft, draft-ietf-idmr-pim-dm-05.txt*, May 1995.

[3] J. Moy, "Multicast extensions to OSPF," RFC 1584, IETF, Mar. 1994.

[4] D. Estrin, D. Farinacci, A. Helmy, D. Thaler, S. Deering, M. Handley, V. Jacobson, C. Liu, P. Sharma, and L. Wei, "Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol specification," RFC 2362, IETF, June 1998.

[5] A. Ballardie, "Core Based Trees (CBT) multicast routing architecture," RFC 2201, IETF, Sept. 1997.

[6] David G. Thaler and Chinya V. Ravishankar, "Distributed center-location algorithms," *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 3, pp. 291–303, Apr. 1997.

[7] Bernard M. Waxman, "Routing of multipoint connections," *IEEE Journal on Selected Areas in Communications*, vol. 6, no. 9, pp. 1617–1622, Dec. 1988.

[8] Bernard M. Waxman, "Performance evaluation of multipoint routing algorithms," in *Proc. Infocom'93*. IEEE, Mar. 1993, vol. 3, pp. 980–986.

[9] D. Estrin, Mark Handley, Ahmed Helmy, and Polly Huang, "A dynamic bootstrap mechanism for rendezvous-based multicast routing," in *Proc. Infocom'99*. IEEE, 1999, vol. 3, pp. 1090–1098.