An Efficient Approach for Load-Shared and Fault-Tolerant Multicore Shared Tree Multicasting

A. Alyanbaawi, B. Gupta, S. Rahimi, N.Rahimi, and K. Sinha
Department of Computer Science
Southern Illinois University
Carbondale, IL 62901, USA
ashraf @(siu.edu), bidyut, Rahimi, nick, Koushik.sinha@(cs.siu.edu)

***Abstract*— Multicasting can be done in two different ways: source based tree approach and shared tree approach. Shared tree approach is preferred over source-based tree approach because in the later construction of minimum cost tree per source is needed unlike a single shared tree in the former approach. However, in shared tree approach a single core needs to handle the entire traffic load resulting in degraded multicast performance. Besides, it also suffers from 'single point failure'. In this paper, novel and efficient schemes for load shared multicore multicasting are presented. Multiple cores are selected statically, that is, independent of any existing multicast groups and also the selection process is independent of any underlying unicast protocol. Some of the selected cores can be used for fault-tolerant purpose also to guard against any possible core failures.**

Keywords—core selection; pseudo diameter; distance vector routing; multicore multicasting; load sharing

I. INTRODUCTION

With the growth of internet, multicast routing has gained its importance in typical applications which include numerous forms of audiovisual conferencing and broadcasting, negotiation and e-commerce systems, replicated database querying, online games as well as the trivial resource discovery feature of Internet routers [1], [2], [4], [6], [12] - [15].

Multicast communication protocols can be classified into two categories, namely, source-based trees [1], [15] and core based trees (CBT) [2]. A problem associated with source-based-tree routing is that a router has to keep the pair information (source, group) and it is a one tree per source. In reality the Internet is a complex heterogeneous environment, which potentially has to support many thousands of active groups, each of which may be sparsely distributed; this technique clearly does not scale. Shared tree based approaches like CBT [2], [11] and protocol independent multicasting – sparse mode (PIM-SM) [6] offer an improvement in scalability by a factor of the number of active sources.

A core-based tree / shared tree [2] involves having a single node, known as the core of the tree, from which branches emanate. These branches are made up of other routers, so-called non-core routers, which form a shortest path between a member-host's directly attached router and the core. A core need not be topologically centered, since multicasts vary in nature and therefore, the form of a core-based tree also can vary [2]. CBT is attractive compared to source based tree because of its key architectural features like scaling, tree creation, and unicast routing separation. The major concerns of shared tree approach are; core selection and core as a single point failure. Core selection [2], [7] is the problem of appropriate placement of a core or cores in a network for the purpose of improving the performance of the tree(s) constructed around these core(s).

In static networks core selection depends on knowledge of entire network topology. It involves all routers in the network. There exist several important works [3], [5], [13], [14] which take into account network topology while selecting a core. Maximum Path Count (MPC) core selection method [3] needs to know complete topology to calculate shortest paths for all pairs. The nodes are then sorted in descending order of their path counts. The first nodes are selected to be the candidate cores. In Delay Variant Multicast Algorithm (DVMA) it is assumed that the complete topology is available at each node [5]. It works on the principle of k-shortest paths to the group of destination nodes concerned. If these paths do not satisfy a delay constraint, then it may find a longer path, which is a shortfall of DVMA. Optimal Cost Based Tree (OCBT) [13], [14] approach calculates the cost of the tree rooted at each router in the network and selects the one which gives the lowest maximum delay over all other roots with lowest cost. It needs knowledge of the whole topology.

A. Related Works

In the case of single core-tree based multicast, the core has to handle all traffic load. It degrades the performance. To overcome the problem, shared tree multicast using multiple cores is the only solution. It distributes the total traffic load on the cores resulting in improved load balancing and thereby causing improved multicast performance. There exist in the

978-1-5386-0837-1/17/$31.00 ©2017 IEEE

literature some important contributions in this area of multicore-based multicasting [16] - [18]. The goal of these works is to achieve load balancing. In [16], Jia et al. have presented a Multiple Shared-Trees (MST) based approach to multicast routing. In their approach, the tree roots are formed into an *anycast* group so that the multicast packets can be anycast to the nearest node at one of the shared trees. However, load balancing is at the level of each source choosing the best core closest to it rather than attempting to utilize all the candidate cores simultaneously. This may lead to congestion in a core if multiple sources choose that core based on their shortest delay metric.

In [17], a unique tree consisting of multiple cores is maintained with one of the cores being the root. The objective of the work is to develop a loop free multi-core based tree by assigning level numbers to the cores and the nodes to join the tree to help maintain the tree structure. The cores need to coordinate with one another for their operations.

Zappala et.al. [18] have considered two different approaches for multicore load balanced multicasting. The first one is senders-to-many scheme; it partitions the receivers of a group among the trees rooted at different cores so that each receiver is exactly on one core tree at a time. Therefore, one core tree spans some of the group members only. Even though it offers less routing state, yet it has the complex task to take care of newly arriving group members, i.e. partition them appropriately to the different core trees. In their second scheme, each core tree spans over the entire receiver group. To transmit data, different senders in a multicast group can use different trees with respect to the proximity of the source to the tree; it helps in balancing the traffic load and improve performance. The trees are maintained independently unlike the work in [17]. The core distribution method follows the hash based scheme of [19].

B. Our Contribution

In this paper, we have considered shared tree multicast with multiple cores. The motivation of the work is to improve multicast performance via load sharing; we prefer the term 'sharing' to 'balancing' because the objective of the work is to engage all cores whenever possible and more so because it is not possible to know a priori the duration of different multicast sessions running at a given time. A brief sketch of the proposed work is as follows. Our core selection process is different from the ones in [18] and [19] and it is based on the idea of *pseudo diameter* [8] – [10]. The proposed schemes neither partition the receivers among the different core-trees nor they build a unique tree consisting of multiple cores [17], [18]. In our proposed scheme, each core tree spans all members of all existing groups. Cores are determined statically using the routing information of all routers in the underlying unicast domain. It is independent of any underlying unicast protocol active in the domain. The metric used in the determination is the one used in the unicast routing tables of the routers. The best core, i.e. the core with the minimum pseudo diameter, is called the primary core. In our approach, a sender does not decide which core to send packets to unlike in [18]; rather any source must send its first packet to the primary core and the primary core then decides if the sender needs to send the rest of its packets to any other core.

The paper is organized as follows. In Section II, we state briefly the existing concept of pseudo diameter and in Section III we state the multicore selection scheme. In Section IV, we discuss the performance of the core selection scheme. In Section V, we present the proposed multi-core load-sharing multicast schemes. Finally, Section VI draws the conclusion.

II. PSEUDO DIAMETER

Two widely used unicast routing protocols are distance vector routing (DVR) and link state routing (LSR). In the former one, routers do not have the knowledge of network topology, whereas in the later routers have this knowledge. Note that the concept of pseudo diameter is independent of the underlying unicast routing protocol. We denote the unicast routing table by UCT_i for some router r_i; it can be either the DVR table or the LSR table of the router depending on the unicast protocol used. Pseudo diameter of a router r_i denoted as $P_d(r_i)$ is defined as follows [8] - [10].

$$P_d\,(r_i) = \max\,\{c_{i,j}\}, \quad \text{where } c_{i,j} = \text{cost}\,(r_i, r_j), \; [1 \leq j \leq n, j \neq i] \text{ and } c_{i,j} \in UCT_i \text{ and n = number of routers in the network}$$

In words, pseudo diameter of router r_i denoted as $P_d(r_i)$, is the maximum value among the costs (as present in its routing table UCT_i) to reach from r_i all other routers in a network. The implication of pseudo-diameter is that any other router is reachable from router r_i within the distance (i.e. no. of hops/delay etc.) equal to the pseudo diameter P_d (r_i) of router r_i,. It thus directly relates to the physical location of router r_i. Pseudo diameter is not the actual diameter of the network, because it depends on the location of router r_i in the network. So different routers in the network may have different values for their respective pseudo diameters. Therefore, pseudo diameter P_d is always less than or equal to the actual diameter of a network.

As an example, consider the network shown in Fig. 1. Without any loss of generality, we have considered DVR protocol as the underlying unicast protocol used in the network. The respective DVR tables of the routers are shown in Fig. 2. Note that the diameter of the network is 90. From router A's table, its pseudo diameter is 90, which is equal to the network diameter; whereas for router C it is 70 as is seen from C's DVR table. It means that if C is the source of a communication, the maximum cost to reach any other router will be 70, which is less than the network diameter of 90. In this context, the following observations [10] are worth mentioning.

Lemma 1. Let S_i be the source and r_i, r_j, ….., r_m be the respective reductions in the pseudo-diameter (P_d) of S_i before a data packet arrives at its destination D, then $r_i + r_j + \ldots\ldots\ldots + r_m \leq P_d$.

Lemma 2. Any broadcasting algorithm based on pseudo diameter guarantees that each router in the network receives a copy of the packet sent by the broadcast source.

III. MULTICORE SELECTION SCHEME

We present a systematic approach to select the cores to be used for multicore multicasting. Cores are selected statically using the routing information of all routers in the underlying unicast domain. This core selection is independent of any multicast group. For multicore multicast, each router r_i executes the following steps to select the required number of cores, m.

```
Multicore selection process

1. Router $r_i$ determines its $P_d(r_i)$ from its unicast routing table.
2. It broadcasts $P_d(r_i)$ in the network using pseudo diameter based broadcasting [10].
3. Router $r_i$ receives all $P_d(r_j)$ from every other router $r_j$ , $1 \leq j \leq n, j \neq i$
4. Router $r_i$ creates a list of m routers out of all n routers, sorted in ascending order based on their $P_d$ values.
   // the first one in the m-router list is the primary core.
```

Remark 1. Every router creates the same sorted list of the m cores.

Remark 2. Message complexity of the core selection process is $O(n^2)$.

Remark 3. In the core selection process, a router does not need to know the entire topology.

Observation 1. For fault-tolerant single-core multicast, there are (m-1) number of redundant (stand by) cores.

Observation 2a. To guard against any possible core failures while using the m cores for multicasting, a total of 2m cores can be selected by the *Multicore selection process*.

Observation 2b. According to the positions in the sorted list all odd-numbered cores can be used for multicasting. Every odd-numbered core will have its standby as the even numbered core that immediately follows it in the sorted list of the cores; this standby core selection utilizes the proximity between these two cores from the viewpoint of their pseudo diameter values.

A. An Example

Consider the example network of Fig.1. Primary core, in our approach, is a router that has the least pseudo diameter value compared to all other routers in the network. From the DVR tables of the network, pseudo-diameter of all routers in the network can be obtained. Pseudo diameters of routers A, B, C, D, E, F, G, and H are 90, 90, 70, 80, 60, 90, 80, and 80 respectively.

Each router broadcasts its P_d value to all other routers in the network. At the end of the broadcast, each router has the same sorted list of the routers based on their respective pseudo diameters. It is shown in Fig. 3. Observe in the figure that there is more than one router with same P_d value. Routers D, G, and H have the same P_d value, viz., 80. If this situation arises, these routers are sorted in descending order of their router ids (IP addresses). Without any loss of generality, we assume that router H has the highest router id, followed by routers G and D respectively. So router H has the priority to be selected as a possible candidate core over routers G and D. According to Fig. 3, for the given network in Fig. 1, router E is the primary core as it is the one with least pseudo-diameter value, 60. For multicore multicasting, the first few routers from the list will be selected as needed. To incorporate core redundancy in case of single core multicast, router C can be chosen as the secondary core (standby core) as it has the next least pseudo-diameter. It will make the single-core scheme fault tolerant. More cores can act as standby for larger degree of fault-tolerance.

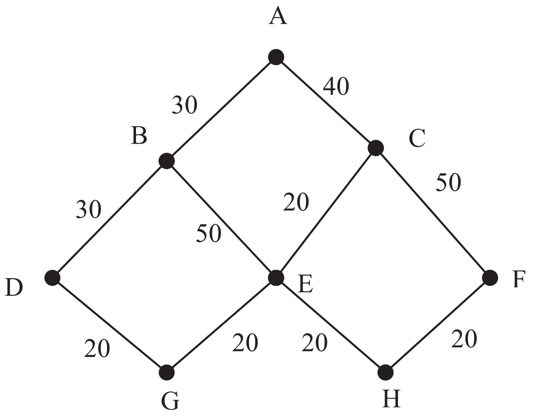


Fig. 1 An 8 router network

A		
Dest.	Next	Delay
A	A	0
B	B	30
C	C	40
D	B	60
E	C	60
F	C	90
G	C	80
H	C	80

B		
Dest.	Next	Delay
A	A	30
B	B	0
C	A	70
D	D	30
E	E	50
F	E	90
G	D	50
H	E	70

C		
Dest.	Next	Delay
A	A	40
B	A	70
C	C	0
D	E	60
E	E	20
F	F	50
G	E	40
H	E	40

D		
Dest.	Next	Delay
A	B	60
B	B	30
C	G	60
D	D	0
E	G	40
F	G	80
G	G	20
H	G	60

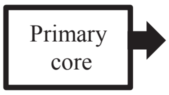
E		
Dest.	Next	Delay
A	C	60
B	B	50
C	C	20
D	G	40
E	E	0
F	H	40
G	G	20
H	H	20

F		
Dest.	Next	Delay
A	C	90
B	H	90
C	C	50
D	H	80
E	H	40
F	F	0
G	H	60
H	H	20

G		
Dest.	Next	Delay
A	E	80
B	D	50
C	E	40
D	D	20
E	E	20
F	E	60
G	G	0
H	E	40

H		
Dest.	Next	Delay
A	E	80
B	E	70
C	E	40
D	E	60
E	E	20
F	F	20
G	E	40
H	H	0

Fig. 2. DVR tables of the routers



r_i	$P_d(r_i)$
E	60
C	70
H	80
G	80
D	80
F	90
B	90
A	90

Fig. 3. Sorted list based on the pseudo diameters

IV. PERFORMANCE

The presented multicore selection scheme needs only one broadcast independent of the number of cores to be used for multicasting. Therefore, the message complexity is $O(n^2)$, where n is the total number of nodes in the network. However, in our approach a router does not need to have the complete topological information. Note that to incorporate the effect of any changes in the network, e.g., router failure, this core selection process can run periodically. Besides, the core selection scheme is independent of any multicast groups unlike most existing works because it is a static core selection approach. The performance of the presented core selection method is compared with some important existing core selection approaches. Complexities of these methods are briefly discussed below.

Maximum path count (MPC) core selection method [3] finds the shortest paths for all pairs of nodes in the given network. Complexity of this approach is $O(n^2)$ where n is the number of nodes in the network. Minimum average distance (MAD) method [3] finds the average distance along the shortest paths from each node to all other nodes in the network. The nodes are sorted in ascending order of their average distance. The first few nodes are selected to be the candidate cores. Complexity of this approach is $O(n^2)$, where n is the total number of nodes in the network. However, in both MPC and MAD a router needs to have the complete topological information unlike our approach. In Delay Variant Multicast Algorithm (DVMA) [5] the worst case complexity of DVMA is $O(klmn^4)$, where k, l are the numbers of path generated, m is the size of the multicast group, and n is the number of nodes in the network. OptTree [3] method suggests an optimization criterion whose complexity is O $(|M|^3|C|)$, where M is the no. of multicast group members and C is the no. of candidate cores.

Topology Generation: In our experimental setup we have used the NS2 simulator. BRITE topology generator is used to create flat random graphs. In BRITE topology, we have chosen Waxman model for topology generation which uses Waxman's probability model for interconnecting the nodes of the topology. From the provided heavy-tailed and random approach for node placement, we have used random placement approach in which each node is placed in a randomly selected location of the plane.

We have experimented with 10 different 30-router topologies, 40-router topologies, and 50-router topologies. In each topology group sizes vary from 5 to 25.

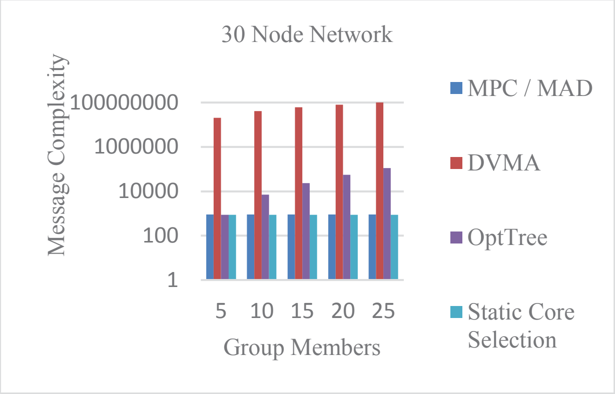


Fig. 4. 30-router network

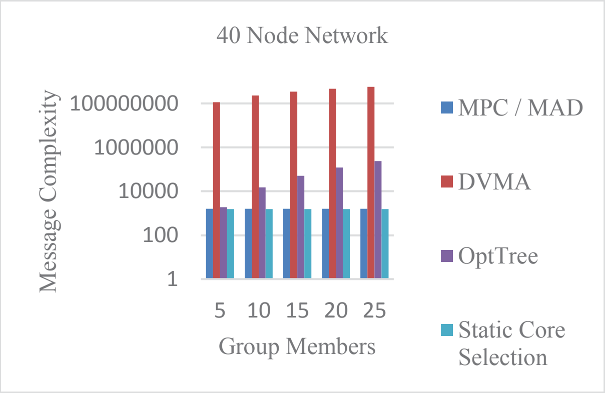


Fig. 5. 40-router network

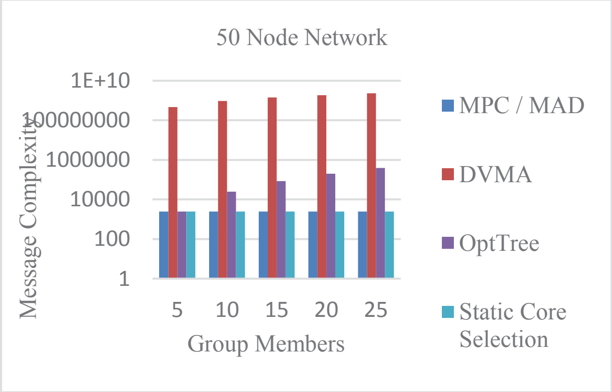


Fig. 6. 50-router network

For each 30-router topology multiple routers are chosen as sources for each group. We have measured the total number of packets generated for core selection in our approach. We have then compared it with some existing important related approaches. Fig. 4 shows the average number of packets generated on all 30-router topologies for the core selection approaches discussed in this paper. In a similar way average number of packets generated for 40 and 50-router network-topologies are shown in Fig. 5 and Fig. 6 respectively. Simulation results have shown superiority of our approach to these other approaches.

V. LOAD SHARED MULTICORE MULTICAST

In this section, we present two simple and yet very effective load-shared multicast schemes, LSM-cast1 and LSM-cast2; we have considered distribution of senders to different cores irrespective of which group a sender sends to. Each core tree spans all members of all existing groups. Cores are determined statically using the routing information of all routers in the underlying unicast domain. It is independent of any underlying unicast protocol active in the domain. The best core, i.e. the core with the minimum pseudo diameter, is called the primary core. In our approach, a sender does not decide which core to send packets to unlike in [18]; rather any source must send its first packet to the primary core and the primary core then decides if the sender needs to send the rest of its packets to any other core. Below we present the schemes. We assume that there are m cores present in the system, C_0 denotes the primary core, and C_i denotes the ith core. S_i^{new} denotes the ith source; N denotes the current number of unique senders initially set at 0.

```
LSM-cast1

Sender $S_i^{new}$ (executed by each new sender)
    1. Sends the 1st multicast packet to $C_0$
    2. Receives the message 'continue with core $C_j$'
    3. Sends rest of the packets to $C_j$

Primary core $C_0$ (executed by the primary core)
    1. Receives the 1st multicast packet from $S_i^{new}$
    2. Multicasts the packet in the $C_0$-based tree
    3. Selects the core $C_j = N \bmod m$
    4. Unicasts 'continue with core $C_j$' to $S_i^{new}$
    5. $N = N+1$

Core $C_j$
    1. Multicasts the packets received from $S_i^{new}$
```

Let us now estimate the extra traffic load on the primary core. Primary core always has to send the message '*continue* *with core C_j'*. So there are N unicasts to N sources from the primary core. In addition, we need to consider the following different cases as well. Let N = km + n', k is an integer and $0 \leq k \leq m$ and n' < m

Case 1: $N \leq m$

Number of extra single-packet multicasts by C_0: (N-1)

Case 2: $N > m$

Number of extra single-packet multicasts by C_0: k(m-1) + (n'-1)

We see that this extra load is a function of both N and m. However, networks have no control over the number of multicast sources; therefore, we infer that the extra traffic load on the primary core can reduce as we select more cores.

We now state the second scheme.

```
LSM-cast2

Sender $S_i^{new}$ (executed by each new sender)
    1. Sends the 1st multicast packet to $C_0$
    2. Receives the message 'continue with core $C_j$'
    3. Sends rest of the packets to $C_j$

Primary core $C_0$ (executed by the primary core)
    1. Receives the 1st multicast packet from $S_i^{new}$
    2. Selects the core $C_j = N \bmod m$
    3. Unicasts the packet to $C_j$
    4. Unicasts 'continue with core $C_j$' to $S_i^{new}$
    5. $N = N + 1$

Core $C_j$
    1. Multicasts the packet received from $C_o$
    2. Multicasts the packets received from $S_i^{new}$
```

Let us now estimate the extra traffic load on the primary core. As in LSM-cast1 there are N unicasts to N sources from the primary core. In addition, we need to consider the following.

Case 1: $N \leq m$

Number of unicasts by C_0 to other cores: (N-1)

Case 2: $N > m$

Number of unicasts by C_0 to other cores: k(m-1) + (n'-1)

Thus, we infer that the extra traffic load on the primary core is less in the second approach than in the first one because it is the extra unicasts in the second approach, not the extra multicasts. However, when N is small the first approach may become almost comparable with the second approach from the viewpoint of this extra traffic load.

VI. CONCLUSION

In this paper we have presented a multicore selection scheme, which is independent of any existing multicast groups and independent of any underlying unicast protocol. It uses the concept of pseudo diameter. The metric used in the determination is the one used in the unicast routing tables of the routers. The core selection is unanimous because all routers have the same information needed for the selection. The proposed load sharing schemes neither partition the receivers among the different core-trees nor they build a unique tree consisting of multiple cores [17], [18]. In our approaches, a sender does not decide which core to send packets to unlike in [18]; rather any source must send its first packet to the primary core and the primary core then decides if the sender needs to send the rest of its packets to any other core. Therefore, which core to use by a sender is not the sender's responsibility – this appears to be logical. We have shown that the degree of fault-tolerance can be enhanced from covering single point-failure to any number of core failures and the presented idea as appeared in the observations to achieve this is very simple. .

REFERENCES

[1] Andrew Adams, Jonathan Nicholas, and William Siadak, "Protocol independent multicast - dense mode (PIM-DM)," Internet Engineering Task Force (IETF), RFC-3973, January 2005.

[2] Tony A. Ballardie, "Core based tree multicast routing architecture," Internet Engineering Task Force (IETF), RFC 2201, September 1997.

[3] A. Karaman and H. Hassanein, "Core selection algorithms in multicast routing – comparative and complexity analysis," J. Computer Communications, Vol. 29, No. 8, pp. 998-1014, May 2006.

[4] Stephen E. Deering and David R. Cheriton, "Multicast routing in datagram internetworks and extended LANs," ACM Transactions on Computer Systems (TOCS), Vol. 8, No. 2, pages. 85-110, May 1990.

[5] George N. Rouskas and Ilia Baldine, "Multicast routing with end-to-end delay and delay variation constraints," IEEE Journal on Selected Areas in Communications, Vol. 15, No. 3, April 1997.

[6] Bill Fenner, Mark Handley, Hugh Holbrook, and Isidor Kouvelas, "Protocol independent multicast - sparse mode (PIM-SM)," Internet Engineering Task Force (IETF), RFC-4601, August 2006.

[7] Weijia Jia, Wei Zhao, Dong Xuan, and Gaochao Xu, "An efficient fault-tolerant multicast routing protocol with core- based tree techniques," IEEE Trans. on Parallel and Distributed Systems, Vol.10, No.10, pages.984-1000, October1999.

[8] Sindoora Koneru, Bidyut Gupta, Shahram Rahimi, and Ziping Liu, "Hierarchical pruning to improve bandwidth utilization of rpf-based broadcasting," IEEE Symposium on Computers and Communications (ISCC), Split, Croatia, pp. 96-100, July 2013.

[9] Sindoora Koneru, Bidyut Gupta, and Narayan Debnath, "A novel DVR based multicast routing protocol with hierarchical pruning," International Journal of Computers and Their Applications (IJCA), Vol. 20, No. 3, pages 184 – 191, September 2013.

[10] Sindoora Koneru, Bidyut Gupta, Shahram Rahimi, Ziping Liu, and Narayan Debnath, "A highly efficient rpf-based broadcast protocol using a new two-level pruning mechanism," Journal of Computational Science (JOCS), Vol. 5, No. 3, March 2014.

SpringerLink, Vol. 2345, pages.1045-1056, Berlin, Heidelberg, 2002.

[11] Hwa-Chun Lin and Shou-Chuan Lai, “A simple and effective core placement method for the core based tree multicast routing architecture,” Proc. IEEE Int. Conf. Performance, Computing, and Communications, pages. 215-219, February 2000.

[12] Tom Pusateri, “Distance vector multicast routing protocol,” Juniper Networks, Internet Engineering Task Force (IETF), draft-ietf-idmr-dvmrp-v3-
11.txt, October 2003.

[13] Young-Chul Shim and Shin-Kyu Kang, “New center location algorithms for shared multicast trees,” Lecture Notes in Computer Science, SpringerLink, Vol. 2345, pages.1045-1056, Berlin, Heidelberg, 2002.

[14] David G. Thaler and Chinya V. Ravishankar. “Distributed center-location algorithms,” IEEE Journal on Selected Areas in Communication, vol. 15, no. 3, pages. 291- 303, April 1997.

[15] David Waitzman, Craig Partridge and Stephen E. Deering “Distance vector multicast routing protocol (DVMRP),” Internet Engineering Task Force (IETF), RFC 1075, November 1988.

[16] W. Jia, W. Tu, W. Zhao and G. Xu, “Multi-shared-trees based multicast routing control protocol using anycast selection,” The International Journal of Parallel, Emergent and Distributed Systems, Taylor & Francis, vol. 20(4), pp. 69–84, March 2005.

[17] C. Shields, J.J. Garcia-Luna-Acevez, “The ordered core based tree protocol”, Proc. IEEE INFOCOM’97.

[18] D. Zappala, A. Fabbri, and V. Lo, “An evaluation of shared multicast trees with multiple active cores,” Journal of Telecommunication Systems, March 2002.

[19] Deborah Estrin, Mark Handley, Ahmed Helmy, and Polly Huang, “A dynamic bootstrap mechanism for rendezvous-based multicast routing,” Proc. IEEE INFOCOM, 1999