Distributed Core Multicast (DCM): a multicast routing protocol for many groups with few receivers

Ljubica Blazević Jean-Yves Le Boudec

Institute for computer Communications and Applications (ICA)
Swiss Federal Institute of Technology, Lausanne
{Ljubica.Blazevic, Leboudec}@epfl.ch

Abstract

We present a multicast routing protocol called Distributed Core Multicast (DCM). It is intended for use within a large single Internet domain network with a very large number of multicast groups with a small number of receivers. Such a case occurs, for example, when multicast addresses are allocated to mobile hosts, as a mechanism to manage Internet host mobility or in large distributed simulations. For such cases, existing dense or sparse mode multicast routing algorithms do not scale well with the number of multicast groups. DCM is based on an extension of the centre-based tree approach. It uses several core routers, called Distributed Core Routers (DCRs) and a special control protocol among them. DCM aims: (1) avoiding multicast group state information in backbone routers, (2) avoiding triangular routing across expensive backbone links, (3) scaling well with the number of multicast groups. We evaluate the performance of DCM and compare it to an existing sparse mode routing protocol when there is a large number of small multicast groups. We also analyse the behaviour of DCM when the number of receivers per group is not a small number.

1 Introduction

We describe a multicast routing protocol called Distributed Core Multicast (DCM). DCM is designed to provide low overhead delivery of multicast data in a large single domain network for a very large number of small groups. This occurs when the number of multicast groups is very large (for example, greater than a million), the number of receivers per multicast group is very small (for example, less than five) and each host is a potential sender to a multicast group.

DCM is a sparse mode routing protocol, designed to scale better than the existing multicast routing protocols when there are many multicast groups, but each group has in total a few members.

Relevant aspects of existing multicast routing protocols are described in Section 2. Sparse mode multicast routing protocols, such as the protocol independent multicast (PIM-SM) [5] and the core-based trees (CBT) [3], build a single delivery tree per multicast group that is shared by all senders in the group. This tree is rooted at a single centre router called "core" in CBT, and "rendezvous point" (RP) in PIM-SM.

Both centre-based routing protocols have the following potential shortcomings:

- traffic for the multicast group is concentrated on the links along the shared tree, mainly near the core router;
- finding an optimal centre for a group is a NP-complete problem and requires the knowledge of the whole network topology [26]. Current approaches typically use either an administrative selection of centers or a simple heuristic [20]. Data distribution through a single centre router could cause non optimal distribution of traffic in the case of a bad positioning of the centre router, with respect to senders and receivers. This problem is known as a triangular routing problem.

PIM-SM is not only a centre-based routing protocol, but it also uses source-based trees. With PIM-SM, destinations can start building source-specific trees for sources with a high data rate. This partly addresses the shortcomings mentioned above, however, at the expense of having routers on the source-specific tree keep source-specific state. Keeping the state for each sender is undesirable when the number of senders is large.

DCM is based on an extension of the centre-based tree approach and is designed for the efficient and scalable delivery of multicast data under the assumptions that we mention above (a large number of multicast groups, a few receivers per group and a potentially a large number of senders to a multicast group).

As a first simplifying step, we consider a network model where a large single domain network is configured into areas that are organised in a two-level hierarchy. At the top level is a single backbone area. All other areas are connected via the backbone(see Figure 1). This is similar to what exists with OSPF[14].

The issues addressed by DCM are: (1): to avoid multicast group state information in backbone routers, (2): to avoid triangular routing across expensive backbone links and (3) to scale well with the number of multicast groups.

The following is a short DCM overview and it is illustrated in Figure 1. We introduce an architecture based on several core

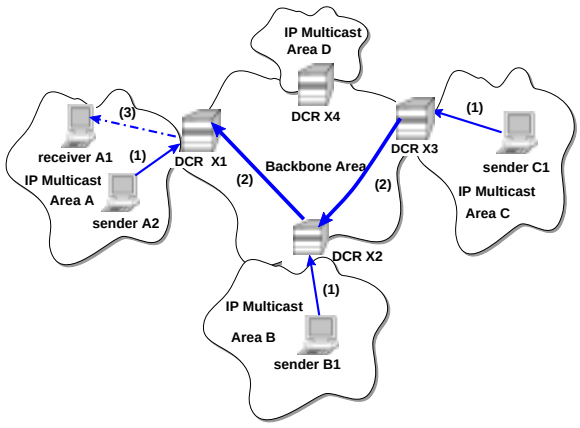


Figure 1: This is a model of a large single domain network and an overview of data distribution with DCM. In this example there are four non-backbone areas that communicate via the backbone. We show one multicast group *M* and DCRs X1, X2, X3 and X4 that serve *M*. Step (1): Senders A2, B1 and C1 send data to the corresponding DCRs inside their areas. Step (2): DCRs distribute the multicast data across the backbone area to DCR X1 that needs it. Step (3): A local DCR sends data to the local receivers in its area.

routers per multicast group, called Distributed Core Routers (DCRs).

- The DCRs in each area are located at the edge of the backbone. The DCRs act as backbone access points for the data sent by senders inside their area to receivers outside this area. A DCR also forwards the multicast data received from the backbone to receivers in the area it belongs to. When a host wants to join the multicast group *M*, it sends a join message. This join message is propagated hop-by-hop to the DCR inside its area that serves the multicast group. Conversely, when a sender has data to send to the multicast group, it will send the data encapsulated to the DCR assigned to the multicast group.
- The Membership Distribution Protocol (MDP) runs between the DCRs serving the same range of multicast addresses. It is fully distributed. MDP enables the DCRs to learn about other DCRs that have group members.
- The distribution of data uses a special mechanism between the DCRs in the backbone area, and the trees rooted at the DCRs towards members of the group in the other areas. We propose a special mechanism for data distribution between the DCRs, which does not require that non-DCR backbone routers perform multicast routing.

With the introduction of the DCRs close to any sender and receivers, converging traffic is not sent to a single centre router in the network. Data sent from a sender to a group within the same area is not forwarded to the backbone. Our approach alleviates the triangular routing problem common to all centre-based trees, and unlike PIM-SM, is suitable for groups with many sporadic senders. Similar to PIM-SM and CBT, DCM is independent of underlying unicast routing protocol.

In this paper we examine the properties of DCM in a large single domain network. However, DCM is not constrained to a single domain network. Interoperability of DCM with other inter-domain routing protocols is the object of ongoing work.

The structure of this paper is as follows. In the next section we give an overview of the existing multicast routing protocols. In Section 3 we present the architecture of DCM. That is followed by the DCM protocol specification in Section 4. In Section 5 we give a preliminary evaluation of DCM. Section 6 presents two examples of the application of DCM: when DCM is used to route packets to the mobile hosts, and when it is used in a large distributed simulation application.

2 Overview of Multicast Routing Protocols

There are two basic families of algorithms that construct multicast trees used for the distribution of IP multicast data: source specific trees and group shared trees. In the former case an implicit spanning tree per source is calculated, which is minimal in terms of transit delay from a source to each of the receivers . In the latter case only one shared tree, which is shared by all sources, is built. There are two types of shared trees. One type is the Steiner minimal tree (SMT)[27]. The main objective is to build a tree that spans the group of members with a minimal cost and thus globally optimise the network resources. Since the Steiner minimal tree problem is NP-complete, numerous heuristics have been proposed [25]. No existing SMT algorithms can be easily applied in practical multicast protocols designed for large scale networks [26]. The other type of shared trees is a centre-based tree that builds the shortest path tree rooted “in the centre” of the networks and spans only receivers of the multicast group.

Below we briefly describe existing dense and sparse mode multicast routing protocols in the Internet.

Dense mode multicast routing protocols

Traditional multicast routing mechanisms, such as DVMRP[24] and MOSPF[13], are intended for use within regions where multicast groups are densely populated or bandwidth is plentiful. Both protocols use source specific shortest path trees. These routing schemes require that each multicast router in the network keeps per source per group information.

DVMRP is based on the Reverse Path Forwarding (RPF) algorithm that builds a shortest path sender-based multicast delivery tree. Several first multicast packets transmitted from a source are broadcasted across the network over links that may not lead to the receivers of the multicast group. Then the tree branches that do not lead to group members are pruned by sending prune messages. After a period of time, the prune state for each (source, group) pair expires and reclaims stale prune state. Subsequent datagrams are flooded again until branches that do not lead to group members are pruned again.

This scheme is currently used for Internet multicasting over the MBONE.

In MOSPF, together with the unicast routing information, group membership information is flooded so that all routers can determine whether they are on the distribution tree for a particular source and group pair. MOSPF is designed atop a link-state unicast routing protocol called OSPF[14]. With MOSPF, in order to scale better, a large routing domain can be configured into areas connected via the backbone area. Multicast routers in non-backbone areas have the complete membership information inside their corresponding areas, while the aggregate membership information of the area is inserted in the backbone. Like DVMRP, MOSPF has a high routing message overhead when groups are sparsely distributed.

Core Based Trees (CBT) sparse mode multicast routing architecture

Unlike DVMRP and MOSPF, a CBT [3] uses centre based shared trees: it builds and maintains a single shared bidirectional multicast distribution tree for every active multicast group in the network. This tree is rooted in a dedicated router for a multicast group that is called the *core* and it spans all group members. Here we give a short description of how a shared tree is built and how a host sends to the group.

A host starts joining a group by multicasting an IGMP[7] host membership report across its attached link. When a local CBT aware router receives this report, it invokes the tree joining process (unless it has already joined the tree) by generating a join message. This message is then sent to the next hop on the path towards the group's core router. This join message must be explicitly acknowledged either by the core router itself or by another router that is on the path between the sending router and the core, which itself has already successfully joined the tree. Once the acknowledgement reaches the router that originated the join message, a new receiver can receive the multicast traffic sent to the group. The state of the shared tree is periodically verified by exchanging of echo messages between neighbouring CBT routers on the shared tree.

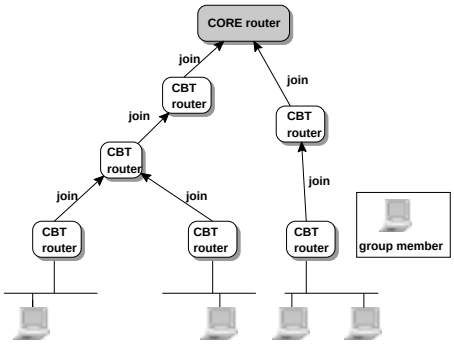


Figure 2: Construction of the shared tree with CBT

Data can be sent to a CBT tree by a sender whose local router is not attached to the group tree. The sender originates native multicast data that is received by a local CBT router. This router finds out the relevant core router for the multicast group, and thus encapsulates the data packet (IP-in-IP) and unicasts it to the core router. After the core router decapsulates the packet it disseminates the multicast data over the group shared tree. When a multicast data packet arrives at the router on the tree, the router uses the group address as an index into the multicast forwarding cache. Then, it sends a copy of the incoming multicast packet over each interface listed in the entry, except the incoming interface.

Data from the sender whose local router is already on the group tree is not sent via the core, but is distributed over the tree from the first-hop on-tree router.

The main advantages of the CBT are that it is independent of the underlying unicast routing protocols and the routers keep forwarding information that correspond only to the multicast group and that do not depend on the source. This makes shared-based trees routing protocols more scalable than source-based trees routing protocols.

The main disadvantages of CBT are: CBT has a potentially higher delay compared with DVMRP because multicast packets do not take the shortest path from the source to the destinations; traffic is concentrated on the links along the shared tree; and the triangular routing problem for non-member senders.

Protocol Independent Multicast Sparse Mode (PIM-SM)

PIM-SM [5] combines the source specific shortest path trees and centre based shared trees. On one hand, PIM-SM is conceptually similar to CBT: it builds a shared directed multicast distribution tree per multicast group centered at a special router called the Rendezvous Point (RP). However, unlike CBT, PIM-SM builds unidirectional trees. The sending of multicast data is similar to CBT. Initially, the sender encapsulates data in register messages and sends them directly to the RP where the data is distributed along the shared tree. If the source continues to send, the RP may send an explicit source-specific join message towards the source. This sets up a path for traffic from the source to the RP.

On the other hand, the unique feature of PIM-SM is that for those sources whose data rate justifies it, forwarding of multicast data from a particular source to the destination group can be shifted from the shared tree onto a source-based tree. However, the result is that the routers on the source-specific tree need to keep a source-specific state.

Multiple centers routing protocols

In order to solve some of the problems inherent to both PIM-SM and CBT due to the existence of the single center router per multicast group, there are several routing protocols that introduce multiple center routers per multicast group. Hierarchical PIM (HPIM)[2] builds on PIM-SM by using the hierarchy of RPs for a group. A receiver joins the lowest level RP, this RP joins a RP at the next level and so on. The number of levels in the hierarchy depends on the scope of a multicast group. For global groups, HPIM does not perform well, because all multicast data is distributed via the RP that is the highest in the hierarchy.

Multicast Source Discovery Protocol (MSDP) [6],[28],[8] allows multiple RPs per multicast group in a single share-tree PIM-SM domain. It can also be used to connect several PIM-SM domains together. Members of a group initiate sending of a join message towards the nearest RP. MSDP enables RPs, which have joined members for a multicast group, to learn about active sources to the group. Such RPs trigger a source specific join towards the source. Multicast data arrives at the RP along the source-tree and then is forwarded along the group shared-tree to the group members. [28] proposes to use the MSDP servers to distribute the knowledge of active multicast sources for a group.

3 Architecture of DCM

In this section we describe the general concepts used by DCM. A detailed description follows in Section 4. We group general concepts into three broad categories: (1) hierarchical network model (2) how membership information is distributed and (3) how user data is forwarded.

3.1 Hierarchical Network Model

We consider a network model where a large single domain network is configured into areas that can be viewed as being organised in a two-level hierarchy. At the top level is a single backbone area to which all other areas connect. This is similar to what exists with OSPF[14]. In DCM we use the area concept of OSPF. However, DCM does not require underlying unicast link state routing.

Our architecture introduces several core routers per multicast group that are called Distributed Core Routers (DCRs). The DCRs are border routers situated at the edge with the backbone. Inside each non-backbone area there can exist several DCRs serving as core routers for the area.

3.2 Distribution of the Membership Information

Regarding the two-level hierarchical network model, we distinguish distribution of the membership information in non-backbone areas and in the backbone area.

Inside non-backbone areas, multicast routers keep group membership information for groups that have members inside the corresponding area. But unlike MOSPF, the group membership information is not flooded inside the area. The state information kept in multicast routers is per group ((*,G) state) and not per source per group (no (S,G) state). If for the multicast group G there are no members inside an area, then no (*,G) state is kept in that area. This is similar to MSDP when it is applied on our network model.

Inside the backbone, non-DCR routers do not keep the membership information for groups that have members in the non-backbone areas. This is different from MSDP where backbone routers can keep (S,G) information when they are on the source specific distribution trees from the senders towards RPs. This is also different from MOSPF where all backbone routers have complete knowledge of all areas' group membership. In DCM, the backbone routers may keep group membership information for a small number of reserved multicast groups that are used for control purposes inside the backbone. We say a DCR is labelled with a multicast group when there are members of the group inside its corresponding area. DCRs in different areas run a special control protocol for distribution of the membership information, e.g information of being labelled with the multicast group.

3.3 Multicast Data Distribution

Multicast packets are distributed natively from the local DCR in the area to members inside the area. Multicast packets from senders inside the area are sent towards the local DCR. This can be done by encapsulation or by source routing. This is similar to what exists in MSDP.

DCRs act as packet exploders, and by using the other areas' membership information attempt to send multicast data across the backbone only to those DCRs that need it (that are labelled with the multicast group). DCRs run a special data distribution protocol that try to optimize the use of backbone bandwidth. The distribution trees in the backbone are source-specific, but unlike MSDP do not keep (S,G) information.

4 The DCM Protocol Specification

In this section we give the specification of DCM by describing the protocol mechanisms for every building block in the DCM architecture.

4.1 Hierarchical Network Model: Addressing Issues

In each area there are several routers that are configured to act as candidate DCRs. The identities of the candidate DCRs are known to all routers within an area by means of an intra-area bootstrap protocol [4]. This is similar to PIM-SM with the difference that the bootstrap protocol is constrained within an area. This entails a periodic distribution of the set of reachable candidate DCRs to all routers within an area.

Routers use a common hash function to map a multicast group address to one router from the set of candidate DCRs. For a particular group address *M*, we use the hash function to determine the DCR that serves[1] *M*.

The used hash function is $h(r(M), DCR_i)$. Function $r(M)$ takes as input a multicast group address and returns the range of the multicast group, while DCR_i is the unicast IP address of the DCR. The target DCR_i is then chosen as the candidate DCR with the highest value of $h(r(M), DCR_j))$ among all j from set $\{1, .., J\}$ where J is the number of candidate DCRs in an area:

$$h(r(M), DCR_i) = \max\{h(r(M), DCR_j), j = 1, .., J\} \quad (1)$$

[1] A DCR is said to serve the multicast group address *M* when it is dynamically elected among all the candidate DCRs in the area to act as an access point for address *M*

One possible example of the function that gives the range[2] of the multicast group address *M* is :

$$r(M) = M \& B \text{ , where } B \text{ is a bit mask.} \quad (2)$$

We do not present here the hash function theory. For more information see [23], [4] and [19]. The benefits of using hashing to map a multicast group to DCR are the following:

- We achieve minimal disruption of groups when there is a change in the candidate DCR set. This means that we have to do a small number of re-mappings of multicast groups when there is a change in the candidate DCR set. See [23] for more explanations.
- We apply the hash function *h(.,.)* as defined by the Highest Random Weight (HRW) [19] algorithm. This function ensures load balancing between candidate DCRs. This is very important, because no single DCR serves more multicast groups than any other DCR inside the same area. By this property, we achieve that when the number of candidate DCRs increases, a decrease of the load on each DCR. Load balancing is more efficient when the number of possible ranges of multicast addresses is larger[19].

All routers in all non-backbone areas should apply the same functions $h(.,.), r(.)$.

By applying the hash function, a candidate DCR is aware of all the ranges of multicast addresses for which it is elected to be a DCR in its area. DCRs in different areas, that serve the same range of addresses, exchange control information (see Section 4.2.2). There is one reserved multicast group that corresponds to every range of multicast addresses. In order to exchange control information with other DCRs, a DCR joins a reserved multicast group that corresponds to a range of multicast addresses that the DCR serves. Packets destined to a reserved multicast address are routed by using another multicast routing protocol, other than DCM (see Section 4.2.2). Maintaining the reserved multicast groups is overhead, and we want to keep it as low as possible. So we want to keep the number of reserved multicast groups small. Obviously there is a tradeoff between the number of reserved groups and the efficiency of load balancing among DCRs inside an area.

4.2 Distribution of membership information

4.2.1 Distribution of membership information inside non-backbone areas

When a host is interested in joining the multicast group *M*, it issues an IGMP join message. A multicast router on its LAN, known as the designated router (DR), receives the IGMP join message. The DR determines the DCR inside its area that serves *M* by means of a hash function, as described in the Section 4.1.

[2] A range is the partition of the set of multicast addresses into group of addresses. A range to which a multicast group address belongs to is defined by Equation (2). e.g if the bit mask is (hex) 000000FF we get 256 possible ranges of IPv4 class-D addresses.

The process of establishing the group shared tree is similar to PIM-SM [5]. The DR sends a join message towards the determined DCR. Sending a join message forces any off-tree routers on the path to the DCR to forward a join message and join the tree. Each router on the way to the DCR keeps a forwarding state for *M*. When a join message reaches the DCR, this DCR becomes labelled with the multicast group *M*. In this way, the delivery subtree , for the receivers of the multicast group *M* in an area, is established. The subtree is maintained by periodically refreshing the state information for *M* in the routers on the subtree (this is done by periodically sending join messages).

Similar to PIM-SM, when the DR discovers that there are no longer any receivers for *M*, it sends a prune message towards the nearest DCR to disconnect from the shared distribution tree. Figure 3 shows an example of joining the multicast group.

4.2.2 Distribution of membership information inside the backbone

The Membership Distribution Protocol (MDP) is used by DCRs in different areas to exchange control information. The following is the short description of MDP.

As said above, within each non-backbone area, for each range of multicast addresses (as defined by Equation (2)) there is one DCR serving that range. DCRs in different areas that serve the same range of multicast addresses are members of the same MDP control multicast group that is used for exchanging control messages. This group is defined by a MDP control multicast address as described in Section 4.1. There are as many MDP control multicast groups as there are possible ranges of multicast addresses. A DCR joins as many MDP control multicast groups as the number of ranges of multicast addresses it serves in its area.

Each MDP multicast group has as many members as there are non-backbone areas since there is one member DCR per area. In practice, this is usually a small number. For example, in the network in Figure 3, X1, X2, X3 and X4 are members of the same MDP control multicast group, while Y1, Y2, Y3 and Y4 are members of the another MDP control multicast group.

We do not propose a specific protocol for maintaining the multicast tree for the MDP control multicast group. This can be done by means of an existing multicast routing protocol. For example, CBT[3] can be used.

DCRs that are members of the same MDP control multicast group exchange the following control information:

- **periodical keep-alive message.** A DCR sends periodically the keep-alive control message informing DCRs in other areas that it is alive. In this way a DCR in one area has the accurate list of DCRs in the other areas that are responsible for the same multicast groups.
- **unicast distance information.** Each DCR sends, to the corresponding MDP control multicast group, information about the unicast distance from itself to other DCRs that it has learned to serve the same range of multicast addresses. This information comes from existing unicast routing tables.

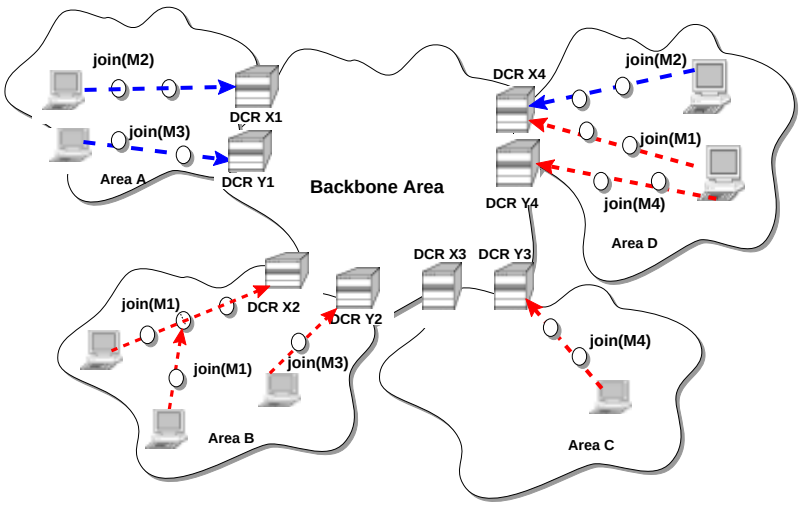


Figure 3: The figure shows hosts in four areas that join four multicast groups with addresses*M1*,*M2*, *M3* and *M4*. Assume that *M1* and *M2* belong to the same range of multicast addresses, while*M3* and *M4* belong to the another range of multicast addresses. Inside every area there are two DCRs. Assume that DCRs (X1,X2,X3 and X4) serve the range of multicast addresses where group addresses*M1* and *M2* belong to. DCRs (Y1,Y2,Y3 and Y4) serve the range of multicast addresses where group addresses*M3* and *M4* belong to. A circle on the figure represents a multicast router in non-backbone areas that are involved in the construction of the DCR rooted subtree. These subtrees are showed with the dash lines. X2 and X4 are now labelled with *M1*, X1 and X4 are labelled with *M2*, Y1 and Y2 are labelled with *M3*, while Y3 and Y4 are labelled with *M4*.

- **multicast group information** A DCR periodically notifies DCRs in other areas about multicast groups for which it is labelled. In this way, every DCR keeps a record of every other DCR that has at least one member for a multicast group from the range that the DCR serves. In example in Figure 3 routers X1,X2,X3 and X4 have a list of labelled DCRs for groups *M1* and *M2*, while Y1,Y2,Y3 and Y4 keeps a list of labelled DCRs for groups *M3* and *M4*.

Section 4.3.2 explains how the control information exchanged with MDP is used for data distribution among DCRs.

MDP uses its MDP control multicast addresses and performs flooding inside the groups defined by those addresses. An alternative approach would be to use MDP servers. This approach leads to a more scalable, but also a more complex solution. This approach is not studied in detail in this paper.

Here we compare DCM to MOSPF[13],[12] in the backbone. In MOSPF, all backbone routers have complete knowledge of all areas' group membership. Using this information together with the backbone topology information, backbone routers calculate the multicast data distribution trees. With MOSPF, complexity in all backbone routers increases with the number of multicast groups. With DCM, DCRs are the only backbone routers that need to keep state information for the groups that they serve. In addition, as described in Section 4.1, the number of multicast groups that a DCR serves decreases as the number of candidate DCRs increases inside an area. Therefore, DCM is more scalable than MOSPF.

With DCM the areas' membership information is distributed among DCRs. An alternative approach, similar to what exists with MSDP, would be to distribute among DCRs the information about active sources in the domain. Then the (S,G) distribution path is built from the DCRs with the members of group G towards the source S. Under our assumptions (a large number of small multicast groups, and many senders) there would be a large number of (S,G) pairs to be maintained in backbone routers. The consequence is that backbone routers would suffer from scalability problems.

4.3 Multicast data distribution

4.3.1 How senders send to a multicast group

The sending host originates native multicast data, for the multicast group *M*, that is received by the designated router (DR) on its LAN. The DR determines the DCR within its area that serves *M*. We call this DCR the source DCR. The DR encapsulates the multicast data packet (IP-in-IP) and sends it with a destination address equal to the address of the source DCR. The source DCR receives the encapsulated multicast data.

4.3.2 Data distribution in the backbone

The multicast data for the group *M* is distributed from a source DCR to all DCRs that are labelled with *M*. Since we assume that the number of receivers per multicast group is not large, there are only a few labelled routers per multicast group. Our goal is to perform multicast data distribution in the backbone in such a way that backbone routers keep a minimal state information while at the same time backbone bandwidth is used efficiently. We propose a solution that can be applied in the

Internet today. It uses point-to-point tunnels to perform data distribution among DCRs. With this solution, non-DCR backbone routers do not keep any state information related to the distribution of the multicast data in the backbone. The drawback is that is that backbone bandwidth is not optimally used because using tunnels may cause possible packet duplications along backbone links. In the Appendix at the end of the paper we propose two alternative solutions for data distribution in the backbone. With those solutions backbone bandwidth is used more efficiently, but at the expense of having the new routing mechanism that needs to be performed by backbone routers.

Point-to-Point Tunnels

The DCR that serves the multicast group *M* keeps the following information: (1) a set *V* of DCRs that serve the same range to which *M* belongs; (2) information about unicast distances between each pair of DCRs from *V*; (3) the set *L* of labelled DCRs for *M*. The DCR obtains this information by exchanging the MDP control messages with DCRs in other areas. In this way, we present the virtual network of DCRs that serve the same range of multicast group addresses by means of an undirected complete graph $G = (V, E)$. *V* is defined above, while the set of edges *E* are tunnels between each pair of DCRs in *V*. Each edge is associated with a cost value that is equal to an inter-DCR unicast distance.

The source DCR, called *S*, calculates the optimal tree that spans the labelled DCRs. In other words, *S* finds the subtree $T = (V_T, E_T)$ of *G* that spans the set of nodes *L* such that $cost(T) = \sum_{e \in E_T} cost(e)$ is minimised. We recognise this problem as the Steiner tree problem. Instead of finding the exact solution, that is a NP-complete problem, we introduce a simple heuristic called Shortest Tunnel Heuristic (STH). STH consists of two phases. In the first phase a greedy tree is built, by adding one by one, the nodes that are closest to the tree under construction, and then removing unnecessary nodes. The second phase is further improving the tree established so far.

Phase 1: Build a greedy tree

- **Step 1**: Begin with a subtree *T* of *G* consisting of the singe node *S*. $k = 1$.
- **Step 2**: if $k = n$ then goto **Step 4**. n is the number of nodes in set V.
- **Step 3**: Determine a node $z_{k+1} \in V$, $z_{k+1} \notin T$ closest[3] to *T* (ties are broken arbitrarily). Add the node z_{k+1} to *T*. $k = k + 1$. Goto **Step 2**.
- **Step 4**: Remove from *T* non-labelled DCRs of degree[4] 1. Also remove non-labelled DCRs of degree 2 if the triangular inequality[5] holds.

[3] with the smallest cost needed to connect to some node that is already on *T*

[4] A degree of a node in a graph is the number of edges incident with a node

[5] The triangular inequality holds if the cost of a single edge that connects the nodes adjacent to the node-to-be removed is less, or equal, to the sum of the costs of the two edges that connect the node-to-be removed with the two adjacent nodes. A node of degree 2 is removed by its two edges being replaced by a single edge (tunnel) connecting the two nodes adjacent to the node-to-be removed. The source DCR is never removed from a graph

Phase 2: Improve a greedy tree

STH can be further improved by two additional steps:

- **Step 5**: Determine a minimum spanning tree for the subnetwork of *G* induced by the nodes in *T* (after the step 4).
- **Step 6**: Remove from the minimum spanning tree non-labelled DCRs of degree 1. Also remove non-labelled DCRs of degree 2 if the triangular inequality holds. The resulting tree is the (suboptimal) solution.

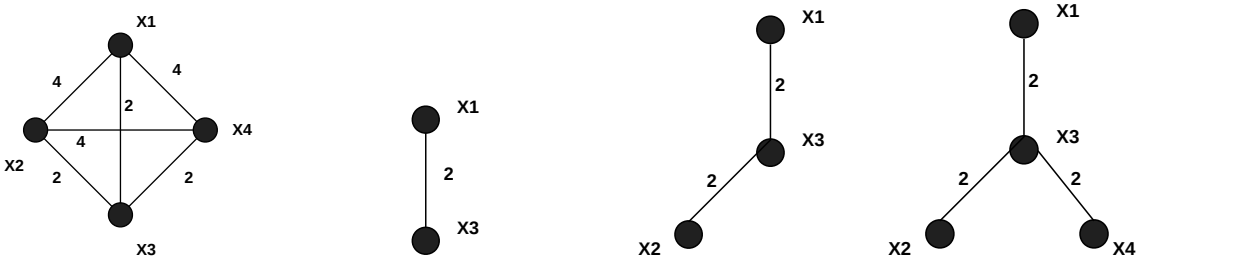
Figures 4, 5 and 6 illustrate three examples of the usage of STH in Figure 3. Nodes X1, X2, X3 and X4 present four DCRs that serve the multicast group *M1*. In the three examples the inter-DCRs unicast distances are different. In all these examples, the source DCR for *M1* is X1 and the labelled DCRs for *M1* are X2 and X4. For the first two examples, the tree that is obtained by the first phase of STH cannot be further improved by steps 5 and 6. In the third example, steps 5 and 6 give improvements in terms of cost of the resulting tree.

The source DCR applies STH to determine the distribution tunnel tree from itself to the list of labelled DCRs for the multicast group. The source DCR puts inter-DCR distribution information in the form of an explicit distribution list in the end-to-end option field of the packet header. Under the assumption that there is a small number of receivers per multicast group, the number of labelled DCRs for a group is also small. Thus, an explicit distribution list that completely describes the distribution tunnel tree is not expected to be long.

When a DCR receives a packet from another DCR, it reads from the distribution list whether it should make a copy of the multicast data and of the identities of the DCRs where it should send multicast data by tunneling. Labelled DCRs deliver data to local receivers in the corresponding area. An example that shows how multicast data is distributed among DCRs is presented in Figure 7. This is a simple example when the resulting distribution tunnel tree is of height 2. Our approach works also for more complex trees, when the height of the tree is more than 2. For such cases, the distribution list that describes the tree is expected to be longer (see Figure 13 in the Appendix for an example of a distribution list that describes a more complex tree).

The source DCR applies STH every time it learns that there is a change of any of the following information: list of DCRs, unicast distances between DCRs, or list of labelled DCRs for a multicast group. If the application of STH results in a new distribution tunnel tree in the backbone, subsequent data is sent along the new tree. Therefore, even though the distribution tree has changed, this does not result in data losses. Analysis of how many data packets are lost, before the source DCR learns about membership change, is to be done.

In order to minimize the encapsulation overhead while sending the multicast data in the backbone, we can use, instead of IP-in-IP tunneling, the encapsulation technique called Minimal Encapsulation within IP[17],[21]. This technique compresses the inner IP header by removing the duplicated fields



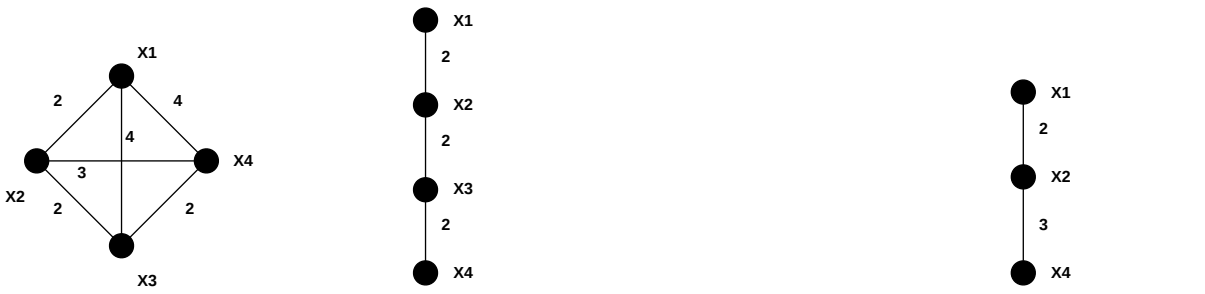
(a) the four node complete graph

(b) start with X1; add X3 to the tree because it is closer to X1 than X2 and X4

(c) X2 is added to the tree

(d) result of STH up to step 4; since X3 is of degree 3 it is not removed from the tree; this is STH solution since the tree cannot be improved by the steps 5 and 6.

Figure 4: First example of application of STH on the complete graph

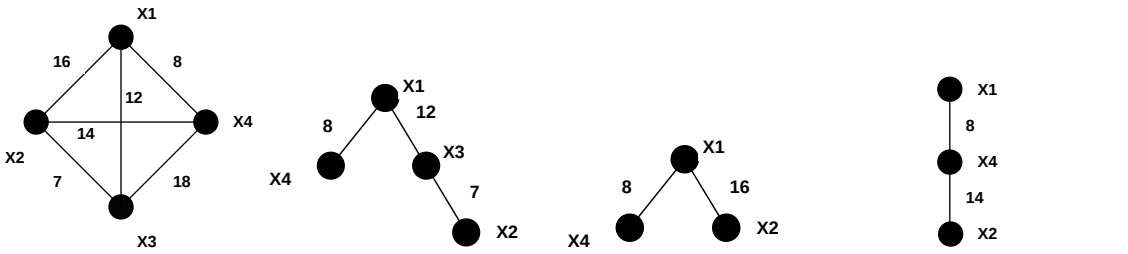


(a) the four node complete graph

(b) result of STH up to step 4

(c) X3 is of degree 2 and it is not the labelled DCR; it is removed from the tree; X2 and X4 are connected with the edge (a tunnel between them); this is STH solution since the tree cannot be improved by the steps 5 and 6.

Figure 5: Second example of application of STH on the complete graph



(a) the four node complete graph

(b) result of STH up to step 4

(c) X3 is removed from the tree; result of STH up to step 5

(d) Steps 5 and 6: minimum spanning tree of X1, X2 and X4; this is STH solution

Figure 6: Third example of application of STH on the complete graph

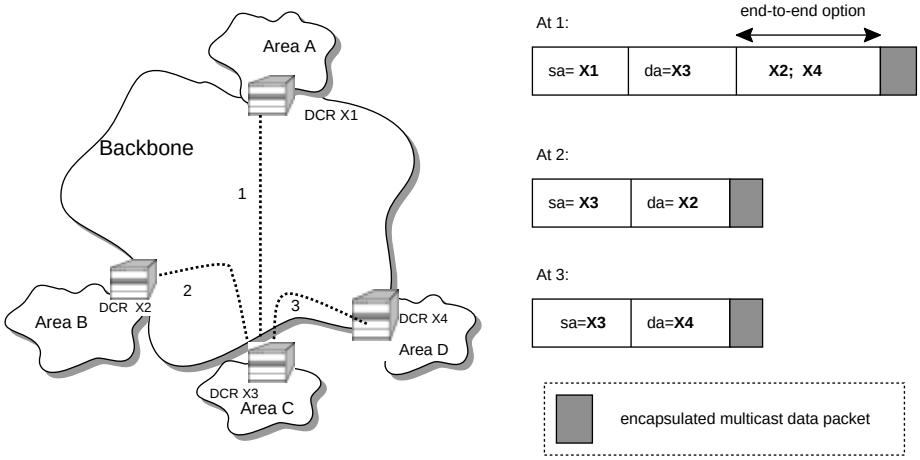


Figure 7: The figure presents an example of inter-DCR multicast data distribution by using point-to-point tunnels. The source DCR is X1 and labelled DCRs are X2 and X4. X1 calculates the distribution tunnel tree to X2 and X4 by applying STH. Assume that the result of STH gives the distribution tunnel tree consisting of edges X1-X3, X3-X2 and X3-X4 (as in Figure 4). Then X1 sends the encapsulated multicast data packet to X3. In the end-to-end option field of the packet, a distribution list is contained. X3 sends two copies of multicast data: one to X2 and the other to X4. On this figure are also presented packet formats at various points (points 1, 2 and 3) on the way from X1 to X2 and X4. A tunnel between the two DCRs is shown with the dash line.

that are in both inner and outer header. In this way we have less header length overhead and less MTU problems than in the case of IP-in-IP encapsulation.

4.3.3 Data distribution inside non-backbone area

A DCR receives encapsulated multicast data packets either from a source that is within its area, or from a DCR in another area. A DCR checks if it is labelled with the multicast group that corresponds to the received packet, i.e whether there are members of the multicast group in its area. If this is the case, a DCR forwards the multicast packet along the distribution subtree that is already established for the multicast group (as is described in Section 4.2.1).

5 Preliminary Evaluation of DCM

We have implemented DCM using the Network Simulator (NS) tool [1]. To examine the performance of DCM, we performed simulations on a single-domain network model consisting of four areas connected via the backbone area. Figure 8 illustrates the network model used in simulations where areas A,B, C and D are connected via the backbone. The whole network contains 128 nodes. We examined the performance under realistic conditions: the links on the network were configured to run at 1.5Mb/s with a 10ms delay between hops. The link costs in the backbone area are higher than the costs in other areas. We evaluate DCM and compare it with the shared-tree case of PIM-SM. Our assumptions are given in the introductory part of the paper: there is a large number of small multicast groups and a large number of potential senders that sporadically send to a group. The most interesting example where such assumptions are satisfied is when one multicast address is assigned to a mobile host. We do not consider the case of PIM-SM when it builds source-specific trees because this introduces a high degree of complexity to PIM-SM when the number of senders is large. We analyse the following characteristics: size of the routing table, traffic concentration in the network and control traffic overhead.

We also discuss how DCM performs when there are some groups that have many members that are sparsely distributed in a large single domain network.

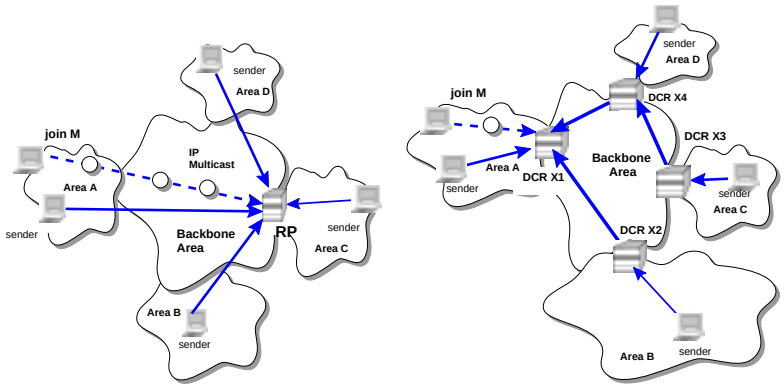
This is the preliminary evaluation of DCM. The evaluation of DCM with some real world network model is yet to be done.

5.1 Amount of multicast router state information and CPU Usage

DCM requires that each multicast router maintains a table of multicast routing information. In our simulations, we want to check the size of multicast router routing table. This is the number of (*,G) multicast forwarding entries. The routing table size becomes an especially important issue when the number of senders and groups grows, because router speed and memory requirements are affected.

We performed a number of simulations. In all the simulations, we used the same network model presented in Figure 8, but with different numbers of multicast groups. For each group there are 2 receivers and 20 senders.

Within each area, there is more than one candidate DCR. The hash function is used by routers within the network to map a multicast group to one DCR in the corresponding area. We randomly distributed membership among a number of active



(a) PIM-SM Shared-tree (b) DCM

Figure 8: The figure presents one member of the multicast group *M* in area A and four senders in areas A, B, C and D. Two different approaches for data distribution are illustrated: the shared-tree case of PIM-SM and DCM. In the case of DCM within each area there is one DCR that serves *M*. In PIM-SM one of the DCRs is chosen to be the centre router (RP) . With PIM-SM, all senders send encapsulated multicast data to the RP. In DCM each sender sends encapsulated multicast data to the DCR inside their area. With PIM-SM, multicast data is distributed from the RP along established distribution tree to the receiver (dash line). With DCM, data is distributed from source DCRs (X1, X2, X3 and X4) to a receiver by means of point-to-point tunnels (full lines in the backbone) and the established subtree in Area A (a dash line)

groups. For every multicast group, receivers are chosen randomly. In the same way, senders are chosen.

The same scenarios were simulated with PIM-SM applied as the multicast routing protocol. In PIM-SM, candidate RP routers are placed at the same location as candidate DCRs in the DCM simulation.

We verified that among all routers in the network, routers with the largest routing table size are DRCs in the case of DCM. In the case of PIM-SM they are RPs and backbone routers. We define the most loaded router as the router with the largest routing table size. Figure 9 shows the routing table size in the most loaded router for the two different approaches. Figure 9 illustrates that the size of the routing table of the most loaded DCR is increasing linear with the number of multicast groups. The most loaded router in PIM-SM is in the backbone. As the number of multicast groups increases, the size of the routing table in the most loaded DCR becomes considerably smaller than the size in the most loaded PIM-SM backbone router.

As it is expected, routing table size in RPs is larger than in DCRs. This can be explained by the fact that the RP router in the case of PIM-SM is responsible for the receivers and senders in the whole domain, while DCRs are responsible for receivers and senders in the area where the DCR belongs.

For non-backbone routers, simulation results show that with the placement of RPs at the edges of the backbone, there is not a big difference in their routing table sizes for DCM and PIM-SM.

Figure 10 illustrates the average routing table size in backbone routers for the two routing protocols. In the case of PIM-SM, this size is increasing linear with the number of multi-

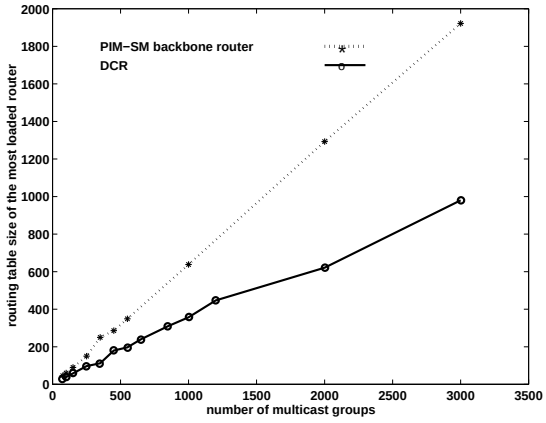


Figure 9: Routing table size for the most loaded routers

cast groups. With DCM all join/prune messages from the receivers in non-backbone areas are terminated at the corresponding DCRs situated at the edge with the backbone. Thus, in DCM non-DCR backbone routers need not keep multicast group state information for groups with receivers inside non-backbone areas. Backbone routers may keep group membership information only for a small number of the MDP control multicast groups.

Here we also investigate how DCM compares to PIM-SM in terms of CPU. In non-backbone areas, the forwarding mechanism of multicast data in routers, other than DCRs and RPs, is

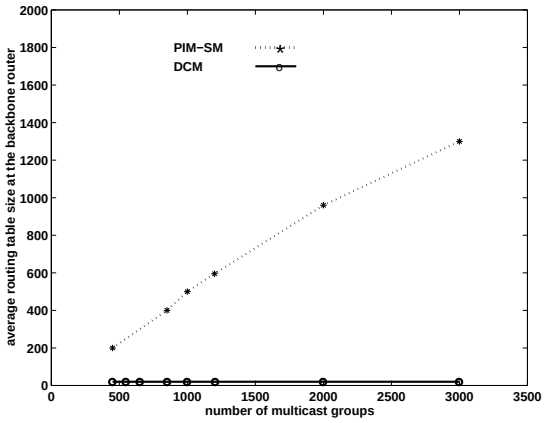


Figure 10: Average routing table size at backbone router

the same for the two approaches. Thus, the forwarding engine in such routers costs the same in terms of CPU for the two routing protocols. However, in the case of DCM, DCRs have more complex forwarding engines than RPs in the case of PIM-SM. The reasons are that DCRs run MDP, and the special packet forwarding mechanism in the backbone. Consequently, DCRs use more CPU than RPs in the case of PIM-SM. The detailed analysis and numerical results for CPU usage for the two approaches is yet to be done.

5.2 Traffic concentration

In the shared-tree case of PIM-SM, every sender to a multicast group initially encapsulates data in register messages and sends them directly to the RP router uniquely assigned to that group within the whole domain. In Figure 8(a) all four senders to a multicast group send data towards a single point in the network. This increases traffic concentration on the links leading to the RP.

Unlike PIM-SM, CBT builds bidirectional shared trees. With CBT, data from a sender whose local router is already on the shared tree is not sent via the core, as is the case with unidirectional shared trees, but is distributed over the tree from its local on-tree router. However, in the case that we consider, when there are only a few receivers per multicast group, many senders' local routers are not on the shared tree. With CBT, in this case, data should be distributed via the core, which becomes similar to data distribution with PIM-SM.

With DCM, converging traffic is not sent to a single point in the network because each sender sends data to the DCR assigned to a multicast group within the corresponding area (as presented in Figure 8(b)).

In DCM, if all senders and all receivers are in the same area, data is not forwarded to the backbone. In that way, backbone routers don't forward the local traffic generated inside an area. Consequently, triangular routing across expensive backbone links is avoided.

5.3 Control traffic overhead

Join/prune messages are overhead messages that are used for setting up, maintaining and tearing down the multicast data delivery subtrees. In our simulations we wanted to measure the number of these messages that are exchanged in the cases when DCM and PIM-SM are used as the multicast routing protocols. Simulations are performed with the same simulation parameters as in the previous subsection (2 receivers per multicast group). They have shown that in DCM the number of join/prune messages is around 20% smaller than in PIM-SM. This result can be explained by the fact that in DCM all join/prune messages from the receivers in the non-backbone areas are terminated at the corresponding DCRs inside the same area, close to the destinations. In PIM-SM join/prune messages must reach the RP that may be far away from the destinations.

In DCM, for every MDP control multicast group, DCRs exchange the MDP control messages. As it is explained in Section 4.2.2, the number of the MDP control multicast groups is equal to the number of possible ranges of multicast group addresses. This number is set independently of the number of multicast groups in the areas. The number of members per MDP control multicast group is equal to the number of non-backbone areas. This is usually a small number. Since the senders to the MDP control multicast group are DCRs, which are the MDP control group members, the number of senders is also a small number. The overhead of the MDP keep-alive control messages depends on the time period that they are sent. DCRs also exchange the MDP control messages that notify the multicast groups for which they are labelled. Instead of sending periodically the MDP control message for every single multicast group that is serves, a DCR can send an aggregate control information for a list of multicast groups, thus reducing the MDP control traffic overhead.

5.4 Behaviour of DCM when the number of receivers per multicast group is not a small number

DCM is a sparse mode routing protocol, designed to be optimal when there are many groups with a few members. Below we investigate how DCM performs when there are some groups that have many members that are sparsely distributed in a large single domain network.

- In the case of PIM-SM, when there are more receivers per multicast group, more control join/prune messages are sent towards the RP for the multicast group. This router is probably far away from many receivers.

  In the case of DCM, join/prune messages are sent from receivers towards the nearest DCR. This entails that the number of join/prune messages in the case of DCM becomes considerably smaller than in the case of PIM-SM when the number of receivers increases. The number of the MDP control multicast groups and the MDP control traffic overhead are independent of the number of receivers per multicast group.

- DCM alleviates the triangular routing problem that is common to the shared tree case of PIM-SM. When the

number of receivers increases, the triangular routing problem with PIM-SM is more important.

- With DCM, when the number of receivers per group increases, we can expect that there are more labelled DCRs per group (but this number is always less that the number of areas). The time to compute the distribution tunnel tree in the backbone is equal to the time to perform STH. The required time is dependent of the number of DCRs that serve the multicast group (equal to the number of non-backbone areas) and is independent of the number of receivers per group. However, we can expect that the distribution tunnel tree in the backbone after applying STH is more complex, and that it contains more tunnel edges, since the number of labelled routers is larger and more nodes need to be spanned.

  With DCM, data distribution in the backbone uses point-to-point tunnels between DCRs. With this approach backbone routers other than DCRs need not be multicast able, but the cost is that it does not completely optimize the use of backbone bandwidth. In order to make the data distribution more optimal, backbone routers should also be included in the forwarding of multicast data. In the Appendix, we give a short outline of our ongoing work on the new mechanisms for distributing the multicast data in the backbone.

6 Examples of application of DCM

6.1 Example of application of DCM in distributed simulations

Distributed simulations and distributed games are applications where scalable multicast communication is needed to support a large number of participants. [11] describes a network architecture for solving the problem of scaling very large distributed simulations. A large-scale virtual environment is spatially partitioned into appropriately sized hexagonal cells. Each cell is mapped to a multicast group. For a large virtual environment there exists a large number of multicast groups. Each participant is associated with a number of cells according to its area of interest, and it joins the corresponding multicast groups. Every participant have the view of all other participants that are members of the same multicast group. Participants can move and dynamically change their cells of interest. We can assume that in a large virtual environment the number of participants per cell is not a large number. In this case DCM can be applied to route packets to a multicast group inside the cell.

6.2 Example of application of DCM: supporting host mobility

Another application of DCM is to use it to route packets to the mobile hosts. We start this subsection with a short description of the certain existing proposals for providing host mobility in the Internet and then illustrate how DCM can support mobility.

Overview of proposals for providing host mobility in the Internet

In the IETF Mobile IP proposal [16] each host has a permanent home IP address that does not change regardless of the mobile host's current location. When the mobile host visits a foreign network, it is associated with a care-of-address, that is IP address related with the mobile host current position in the Internet. When a host moves to visited network it registers its new location with its home agent. The home agent is a machine that acts as a proxy on behalf of the mobile host when it is absent. When some stationary host sends packets for the mobile host it addresses them to the mobile host's home address. When packets arrive on the mobile host's home network, the home agent intercepts them and sends by encapsulation packets towards the mobile host's current location. With this approach all datagrams addressed to a mobile host are always routed via its home agent. This causes the so-called triangle routing problem.

In IPv6 mobility proposal[18] when a handover is performed, the mobile host is responsible for informing its home agent and correspondent hosts about its new location. In order to reduce packet losses during handover, [18] proposes a router-assisted smooth handover.

The Columbia approach [10] was designed to support intracampus mobility. Each mobile host always retains one IP home address, regardless of where it is on the network. There is a number of dedicated Mobile Support Stations (MSSs) that are used to assure the mobile host's reachability. Each mobile host is always reachable via one of the MSSs. When a mobile host changes its location it has to register with a new MSS. A MSS is thus aware of all registered mobile hosts in its wireless cell. A source that wants to send a packet to a mobile host sends it to the MSS that is closest to the source host. This MSS is responsible for learning about the MSS that is closest to the mobile host and to deliver the packet. A special protocol is used to exchange information among MSSs.

MSM-IP (Mobility support using Multicasting in IP) [15] proposes a generic architecture to support host mobility in the Internet by using multicasting as a mechanism to route packets to the mobile hosts. The routing protocol used in this architecture is out of the scope of MSM-IP.

Cellular IP [22] architecture relies on the separation of local mobility from wide area mobility. Cellular IP is applied in a wireless access network and it can interwork with Mobile IP to provide wide area mobility support, that is mobility between Cellular IP networks. With Cellular IP network nodes maintain distributed caches for location management and routing purposes. Distributed paging cache coarsely maintains the position of 'idle' mobile hosts in a cellular IP network. Distributed routing cache maintains the position of active mobile hosts in a Cellular IP network and is updated more frequently that a paging cache. In a Cellular IP network there exists one gateway node (GW). A mobile host entering a Cellular IP network communicates the local GW's address to its home agent as care-of-address. All packets for the mobile host enter a Cellular IP network via the GW. From GW, packets addressed to a mobile host are routed to its current base station on a hop-by-

hop basis according to routing caches in the network nodes.

Application of DCM to host mobility

In this section we show how DCM can be applied in the mobility management approach based on multicasting. This approach is not an alternative to Mobile IP [16] since DCM is not a solution to wide-area mobility. In contrast, this approach can be used as an alternative to Cellular IP[22] within a single domain network.

We consider the network environment composed of wireless cells. Mobile hosts communicate with base stations over wireless links, while the base stations have the fixed connection to the Internet.

When a visiting mobile host arrives into the new domain it is assigned a temporary multicast address[6]. This is the care-of address that the mobile keeps as long it stays in the same domain. This is unlike Mobile IP [16] where the mobile host does a location update after each migration and informs about this to its possible distant home agent.

We propose to use DCM as the mechanism to route packets to the mobile hosts. As explained in Section 4.1, for the mobile host's assigned multicast address, within each area, there exists a DCR that serves that multicast address. These DCRs are responsible for forwarding packets to the mobile host. As said before, the DCRs run the MDP control protocol and are members of a MDP control multicast group for exchanging MDP control information.

A multicast router in the mobile host's cell initiates a joining the multicast group assigned to the mobile host. Typically this router coexists with the base station in the cell. As described in Section 4.2.1 the join message is propagated to the DCR inside the area that serves the mobile host's multicast address. Then, the DCR sends to the MDP control multicast group a MDP control message when the mobile host is registered.

In order to reduce packet latency and losses during a handover, advance registration can be performed. The goal is that when a mobile host moves to a new cell, the base station in the new cell should already start receiving data for the mobile host. The mobile host continues to receive the data without disruption. There are several ways to perform this:

- A base station that anticipates[7] the arrival of a mobile host initiates joining the multicast address assigned to the mobile host. This is illustrated in one example in Figure 11.
- In the case where a bandwidth is not expensive on the wired network, all neighbouring base stations can start receiving data destined to a mobile host. This guarantees that there would be no latency and packet losses during a handover.

A packet for the mobile host reaches all base stations that joined the multicast group assigned to the mobile host. At the same time the mobile host receives data only from a base station in its current cell. A base station that receives a packet on behalf of the mobile host that is not present in its cell can either discard a packet or buffer it for a certain interval of time (e.g. 10ms). Further research is needed to determine what is the best approach.

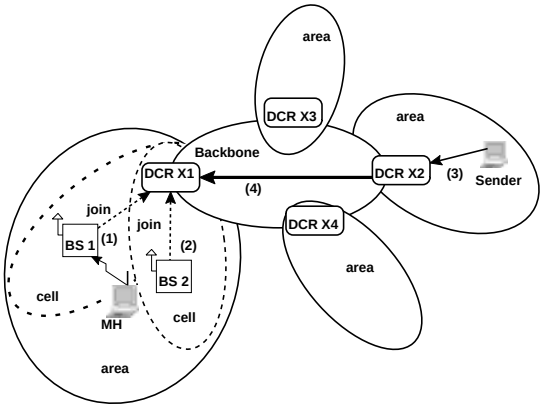


Figure 11: The mobile host (MH) is assigned the multicast address *M*. Four DCRs, X1, X2, X3 and X4 serve *M*. Step (1): Base station BS1 sends a join message for *M* towards X1. X1 informs X2, X3 and X4 that it has a member for *M*. Step (2): Advance registration for *M* in a neighbouring cell is done by BS2. Step (3): The sender sends a packet to multicast group *M*. Step (4): The packet gets delivered through the backbone to X1. Step (5): X1 receives encapsulated multicast data packet. From X1 data is forwarded to BS1 and BS2. MH receives data from BS1.

Here we describe in more details how advance registration is performed. At its current cell, the mobile host receives data along the distribution subtree that is established for the mobile host's multicast address. This tree is rooted at the DCR and maintained with a periodical sending of the join messages. Now, suppose that the base station in the neighbouring cell anticipates arrival of the mobile host. It begins a joining process for the multicast group assigned to the mobile host. This process is terminated when a join message reaches a router that is already on the distribution tree. When the cells are close to each other, joining is terminated at the lowest branching point in the distribution tree. This ensures that the neighbouring base station quickly becomes a part of the multicast distribution tree with low overhead. The neighbouring base station can start joining the multicast group assigned to the mobile host after the mobile host leaves its previous cell. Routers on the distribution tree keep forwarding information for a given time, even if the previous base station stops refreshing the tree because the mobile host leaves its cell. As before, if the base stations are close to each other, the multicast distribution tree for the new base station can be established in a short period of time thus making handover efficient. One example that illustrates advance registration is presented in Figure 12.

[6] In this paper we do not discuss how a multicast address is assigned to a mobile host

[7] The mechanism by which the base station anticipates the arrival of the mobile host is out of the scope of this paper

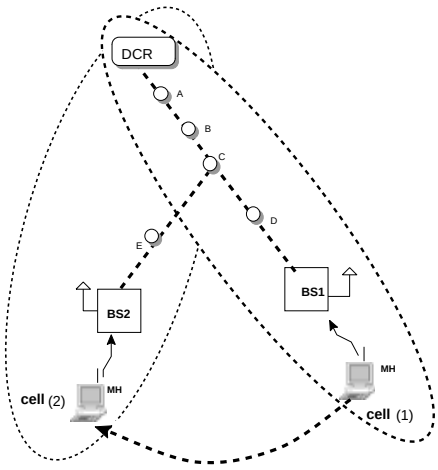


Figure 12: This figure presents an example of advance registration. At first, the mobile host (MH) is in cell 1. MH is assigned a multicast address *M*. Base station BS1 receives data for MH along the distribution subtree rooted at the DCR. On this subtree are routers A, B, C and D. Before host moves from cell 1 to cell 2, neighbouring base station BS2 initiates an advance joining for *M*. Joining at position 2 is terminated at router C.

Comparison of the mobility management based on DCM and the Cellular IP approach

In Cellular IP, the process of establishing the distribution tree from the gateway node to the mobile host is similar to what exists in DCM for establishing the distribution subtree from a DCR to the mobile host in its current area. With DCM, maintenance of the distribution tree is performed by sending of periodic join messages and is initiated by the base stations in the vicinity of the mobile host. With Cellular IP, this is done on the packet basis sent from the active mobile.

We see the scalability problem with Cellular IP when there is a large number of mobile hosts inside the Cellular IP network. The single gateway node is the centre of all distribution trees that are built for mobile hosts within a network. All the traffic for mobile hosts inside the Cellular IP networks goes via the gateway node that presents a 'hot spot' in the network.

With DCM we avoid existence of the center router, potential 'host spot' in the network. DCM builds distribution subtrees for mobile hosts that are rooted at a number of DCRs. We believe DCM to scale better than Cellular IP when there is a large number of mobile hosts.

Open Issues

In this paper we do not address the problems of using multicast routing to support end-to-end unicast communication. These problems are related to protocols such as: TCP, ICMP, IGMP, ARP. A simple solution to this problem could be to have a special range of unicast addresses that are routed as multicast addresses. In this way, packets destined to the mobile host are routed by using a multicast mechanism. Conversely, at the end systems, these packets are considered as unicast packets and standard unicast mechanisms are applied.

7 Conclusions

We have considered the problem of multicast routing in a large single domain network with a very large number of multicast groups with a small number of receivers. Our proposal, called Distributed Core Multicast (DCM) is based on an extension of the centre-based tree approach. DCM uses several core routers, called Distributed Core Routers (DCRs) and a special control protocol among them. The objectives achieved with DCM are: (1) avoiding state information in backbone routers, (2) avoiding triangular routing across expensive backbone links, (3) scaling well with the number of multicast groups. Our initial results tend to indicate that DCM performs better than the existing sparse mode routing protocols in terms of multicast forwarding table size. We have presented an example of the application of DCM where it is used to route packets to the mobile hosts.

Appendix

In section Section 4.3.2 we presented one solution called point-to-point tunnels for the distribution of multicast data between DCRs. Point-to-point tunnels avoid triangular routing across expensive links in the backbone, but does not completely optimise the use of backbone bandwidth. Here we present two alternative solutions called tree-based source routing and list-based source routing that use backbone bandwidth more optimally than point-to-point approach.

Tree-Based Source Routing

This solution assumes that the DCRs are aware of backbone topology (e.g the backbone is one OSPF area) and backbone routers implement a special packet forwarding mechanism called tree source routing. This approach consists in that a source DCR for the multicast group computes a shortest path tree rooted at itself to a list of labelled DCRs for the multicast group. On a shortest path can be included DCRs in other areas that serve the multicast address, as well as non-DCR backbone routers. A description of a shortest path tree with destinations and branching points is included in the tree source routing header by the source DCR. Figure 13 shows one example of tree source routing approach.

This approach ensures that backbone bandwidth is used more optimally than if the "point-to-point tunnels" approach is used. This is achieved at the expense of introducing the new tree source routing mechanism that needs to be performed by backbone routers.

List-Based Source Routing

This solution proposes a new list-based multicast data distribution in backbone. Here we give an initial description of

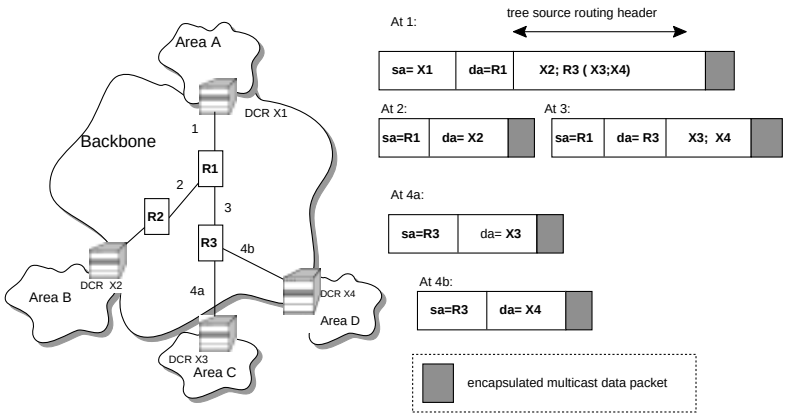


Figure 13: This figure shows how multicast data is distributed from source DCR X1 to labelled DCRs X2, X3 and X4 by using tree-based source routing approach. X1 puts distribution information in tree source routing header after computing a shortest-path tree to routers X2, X3 and X4. At first, the data should be delivered to backbone router R1 where two copies of the multicast data are made. One copy is sent encapsulated to X2, while the other is sent encapsulated to backbone router R3. As soon as router R3 receives a packet it reads from the tree source routing header that it should send two copies of the multicast data: one to X3 and the other to X4.

this mechanism. The final solution is the object of ongoing research.

As in the previous approach we assume that the DCRs are aware of the backbone topology. A special list-based source routing protocol is performed by the DCRs and backbone routers. This works as follows: as soon as a source DCR determines that it must forward a packet to a list of DCRs, it determines the next backbone router(s) to which it should send a copy of the packet to reach every listed DCR. The source DCR sends a copy of the packet to each determined router together with a sublist of the DCRs that should be reached from this router. This sublist is contained in a list source routing header. This is similar to the IP option field that is described in [9]. Unlike a tree-based source routing header, where in a tree source routing header can be included also non-DCR backbone routers, the list source routing header contains only the final DCR destinations.

Each backbone router performs the same steps until multicast data has reached every labelled DCR. Note that a DCR can also send a copy directly to another DCR.

On Figure 14 is presented one example of list-based source routing approach.

References

[1] Network Simulator. Available from http://www-mash.cs.berkeley.edu/ns.

[2] Hierarchical Protocol Independent Multicast (HPIM). Available from www.cs.ucl.ac.uk/staff/jon/mmbook/book/book.html, 1997.

[3] A. Ballardie. Core Based Trees (CBT) Multicast Routing Architecture. RFC 2201, September 1997.

[4] Deborah Estrin, Mark Handley, Ahmed Helmy, Polly Huang, and David Thaler. A Dynamic Mechanism for Rendezvous-based Multicast Routing. In *Proc. of IEEE INFOCOM'99*, New York, USA, March 1999.

[5] D. Estrin et.al. Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification. RFC 2117, June 1997.

[6] D. Farinacci et. al. Multicast Source Discovery Protocol (MSDP). Internet Draft(work in Progress), June 1998.

[7] W. Fenner. Internet Group Management Protocol, Version 2. RFC 2236, November 1997.

[8] B. Fenner et. al. Multicast Source Discovery protocol MIB. Internet Draft(work in Progress), May 1999.

[9] C. Graff. IPv4 Option for Sender Directed Multi-Destination Delivery. RFC 1770, 1995.

[10] John Ioannidis, Dan Duchamp, and Gerald Q. Maguire Jr. IP-based Protocols for Mobile Internetworking. In *Proc. of SIGCOMM'91*, Zurich, Switzerland, September 1991.

[11] Michael Macedonia and Michael Zyda et.al. Exploiting Reality with Multicast Groups: A Network Architecture for Large-Scale Virtual Environments. In *IEEE Computer Graphics and Applications*, September 1995.

[12] J. Moy. MOSPF: Analysis and Experience. RFC 1585 (Informational), 1994.

[13] J. Moy. Multicast Extensions to OSPF. RFC 1584, 1994.

[14] J. Moy. OSPF version 2. RFC 1583, 1994.

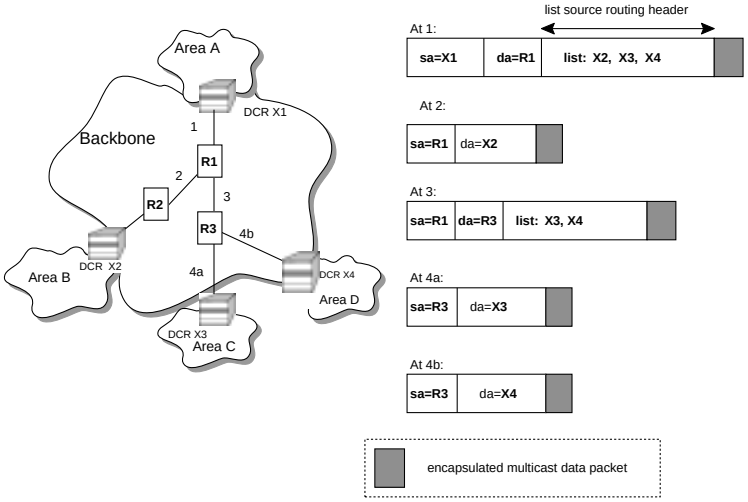


Figure 14: This figure shows how multicast data is distributed from the source DCR X1 to labelled DCRs X2, X3 and X4 by using the list-based source routing approach. X1 determines that it should send a copy of multicast data to backbone router R1. Router X1 puts in the list source routing header information that X2, X3 and X4 should be reached from R1. As soon as R1 receives a packet it makes two copies of the multicast data. One copy of the multicast data is encapsulated is a packet that is sent to X2. Another copy of the multicast data is sent to R3. This packet contains in the list source routing header a list of DCRs that should be reached from R3. The list contains X3 and X4. As soon as R3 receives a packet from R1 it makes two copies of the multicast data. One copy is sent encapsulated to X3. Another copy is sent encapsulated to X4.

[15] Jayanth Mysore and Vaduvur Bharghavan. A New Multicasting-based Architecture for Internet Host Mobility. In *The Third Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'97)*.

[16] C. Perkins. IP Mobility Support, Network Working Group. RFC 2002, October 1996.

[17] C. Perkins. Minimal Encapsulation within IP. RFC 2004, 1996.

[18] Charles E. Perkins and David B. Johnson. Mobility Support in IPv6. In *Proc. of the Second Annual International Conference on Mobile Computing and Networking (MobiCom'96)*.

[19] D. G. Thaler and C. V. Ravishankar. Using Name-Based Mappings to Increase Hit Rates. *IEEE/ACM Transactions on Networking*, 6(1), February 1998.

[20] David G. Thaler and Chinya V. Ravishankar. Distributed Center-Location Algorithms. *IEEE JSAC*, 15(3), April 1997.

[21] J. Tian and G. Neufeld. Forwarding State Reduction for Sparse Mode Multicast Communication. In *Proc. of IEEE INFOCOM'98*, March/April 1998.

[22] Andras G. Valko. Cellular IP: A New Approach to Internet Host Mobility. *ACM SIGCOMM Computer Communicastion Review*, January 1999.

[23] Vinod Valloppillil and Keith W. Ross. Cache Array Routing Protocol v1.0. Internet Draft(work in Progress), 1998.

[24] D. Waitzman, S. Deering, and C. Partridge. Distance vector multicast routing protocol. RFC 1075, 1988.

[25] Bernard M. Waxman. Routing of Multipoint connections. *IEEE JSAC*, 6(9), December 1988.

[26] Liming Wei and Deborah Estrin. The Trade-offs of Multicast Trees and Algorithms. In *Proc.of the 1994 International Conference on Computer Communications and Networks*, San Francisco, CA, USA, September 1994.

[27] Pawei Winter. Steiner problem in networks: A survey. *Networks*, 17(2), 1987.

[28] Li. Yunzhou. Group Specific MSDP Peering. Internet Draft(work in Progress), June 1999.