A Heuristic Multicast Algorithm to Support QoS Group Communications in Heterogeneous Network

Hui Cheng[1], Jiannong Cao[1], Xingwei Wang[2], Srinivasan Mullai[1]
[1]Dept. of Computing, Hong Kong Polytechnic University, China
[2]Computing Center, Northeastern University, China
E-mail: {cshcheng, csjcao}@comp.polyu.edu.hk

Abstract

In this paper, we study the problem of QoS group communication in a heterogeneous network, which consists of multiple MANETs attached to the backbone Internet. We propose a heuristic multicast algorithm called DDVMA (Delay and Delay Variation Multicast Algorithm). DDVMA is designed for solving the DVBMT (Delay- and delay Variation-Bounded Multicast Tree) problem [8], which has been proved to be NP-complete. It can find a multicast tree satisfying the multicast end-to-end delay constraint and minimizing the multicast delay variation. Two concepts- the proprietary second shortest path and partially proprietary second shortest path are introduced, which can help DDVMA achieve better performance in terms of the multicast delay variation than DDVCA (Delay and Delay Variation Constraint Algorithm) [7] that is known to be the most efficient so far. Theoretical analysis is given to show the correctness of DDVMA and simulations are performed to demonstrate the performance of DDVMA in terms of the multicast delay variation.

1. Introduction

The explosive growth of mobile communications has attracted interests in the integration of wireless networks with wireline ones and the Internet in particular. Providing mobile users the wireless access to the Internet is of major interests in today's research in networking. For MANETs, extensive work has been done on extending IP connectivity of the MHs. An integrated connectivity solution is proposed in [1]. Its prototype is implemented by connecting IP networks and MANETs running the AODV routing protocol, where Mobile IP is used for mobility management. MIPMANET [2] is a solution for connecting a MANET to the Internet. MIPMANET uses on-demand routing and provides Internet access by using Mobile IP with foreign agent care-of addresses and reverse tunnelling. A heterogeneous network architecture is proposed in [3], which extends the typical wireless access points to multiple MANETs, each as a subnet of the Internet, to create an integrated environment that supports both macro and micro IP mobility. The heterogeneous network architecture will facilitate the current trend of moving to an all-IP wireless environment.

In the heterogeneous network consisting of multiple MANETs attached to the backbone Internet, a gateway is a fixed node connecting a MANET to the Internet and each gateway serves one MANET. Gateways forward data packets and relay them between MANETs and the Internet. When a MANET is connected to the Internet, it is important for the MHs to detect available Internet gateways. Therefore, efficient gateway discovery mechanism is required. Lots of efforts have been devoted to the problems of gateway forwarding strategies and Internet gateway discovery [4-6]. These works have provided the foundation for our work.

Such an integrated heterogeneous network environment has brought up many new applications. In particular, there is an increasing demand for enhanced services to help users do mobile collaborations, which require the support for mobile group communications. For example, several MANETs, which are distributed in different remote regions, need to coordinate their works. To the best of our knowledge, no multicast algorithm has been proposed to support the QoS group communication in backbone wireline network attached by MANETs.

End-to-end delay is an important QoS parameter in data communications to guarantee that the messages transmitted by the source can reach the destination within a certain amount of time. Multicast delay variation is defined as the difference between the maximum and the minimum multicast end-to-end delays on the multicast tree. It measures the

Proceedings of the 2nd Int'l Conf. on Quality of Service in Heterogeneous Wired/Wireless Networks (QShine'05)
0-7695-2423-0/05 $20.00 © 2005 IEEE

consistency and fairness of receiving messages among all the destinations.

In this paper, we propose a heuristic hybrid multicast algorithm for QoS group communication in the heterogeneous network. Our algorithm is named DDVMA. Each MANET can be seen as a team. When one team wants to send messages to multiple remote teams, two steps are needed. First, the AODV routing protocol is used to discover routes between team leader and the gateway. The end-to-end delay values of all the wireless routes are collected for DDVMA. Second, DDVMA constructs a multicast tree from the source gateway to all the destination gateways in the backbone network utilizing both the topology of backbone network and the delay values of wireless routes. The main contributions of our paper are: (1) consider the integration of MANETs and the backbone Internet and propose a multicast algorithm to support the QoS group communication among the leader MHs of several MANETs; (2) by using AODV to discover the wireless routes and collect the delay information, the construction of the multicast tree in the backbone network can guarantee the QoS requirements of the group communication involving MHs; (3) under the multicast end-to-end delay constraint, the proposed DDVMA can achieve better performance in terms of multicast delay variation than DDVCA known to be the most efficient so far.

The rest of the paper is organized as follows. Section 2 describes the network model, the problem specification, related work, and introduces several new concepts. DDVMA is proposed in Section 3. Theoretical analysis and simulation results are described in Section 4. Finally, we conclude the paper in Section 5.

2. Preliminaries

2.1. Network Model and Problem Specification

The backbone network can be modeled as a weighted digraph G(V,E), where V represents the set of nodes including gateways, and E represents the set of links between the nodes. For each link $l \in$ E, a *link-delay* function D:$l \rightarrow r^+$ is defined. A nonnegative value D(l) represents the transmission delay on link l.

Multicast messages are sent from the leader MH of the source MANET. Messages are first forwarded to the source gateway $v_s \in V$ through the route discovered by AODV, then arrive at a set of destination gateways M $\subseteq$ V-{v_s} through the multicast tree T in the backbone network, and finally are forwarded to the leader MHs of the destination MANETs through the wireless routes between each destination gateway and each leader MH, respectively. To guarantee the QoS of group communication, the multicast end-to-end delay between the leader MH of the source MANET and the leader MH of each destination MANET should not exceed the multicast end-to-end delay constraint Δ, and the multicast delay variation among the leader MHs of destination MANETs should be minimized.

Let $P_T(v_s,v_w)$ denote the path from the source gateway v_s to a destination gateway $v_w \in M$ on T. Then the transmission delay between v_s and v_w on T is defined as $\sum_{l \in P_T(v_s,v_w)} D(l)$. We define a *gateway-delay* function W:g$\rightarrow r^+$ for each gateway g$\in$ {v_s} $\cup$ M. It assigns gateway g a nonnegative value W(g), which represents the delay of the wireless route discovered between gateway g and the leader MH of the MANET g serves.

In our paper, the problem of QoS group communications in the heterogeneous network model is to find an optimal multicast tree $T^*(V_{T^*}, E_{T^*}), \{v_s\} \cup M \subseteq V_{T^*}, E_{T^*} \subseteq E$, satisfying:

$$(1)\ \Delta_{T^*} = W(v_s) + \max_{v_w \in M}\{\sum_{l \in P_{T^*}(v_s,v_w)} D(l) + W(v_w)\} \le \Delta$$

$$(2)\ \delta_{T^*} = \min_{T}\{\max_{v_u,v_w \in M}\{\left|\begin{matrix}(\sum_{l \in P_T(v_s,v_u)} D(l) + W(v_u)) - \\ (\sum_{l \in P_T(v_s,v_w)} D(l) + W(v_w))\end{matrix}\right|\}\},$$

where T denotes any multicast tree spanning v_s and M in G(V,E).

If we assume W(g)=0 for each g$\in$ {v_s} $\cup$ M, the problem turns to be the DVBMT problem, which has been proved to be an NP-complete problem [8]. Our problem is also NP-complete because it contains, as a special case, the DVBMT problem. Hence, only heuristic algorithms can be developed for our problem.

2.2. Related Work

For the DVBMT problem, several heuristic algorithms have been proposed. DVMA (Delay Variation Multicast Algorithm) [8] is a search algorithm which attempts to construct a multicast tree satisfying both the multicast end-to-end delay constraint and the multicast delay variation constraint. Although DVMA demonstrates good average case behavior in terms of the multicast delay variation, its time complexity is very high. DDVCA is a fast and efficient algorithm, which is meant to search as much as possible for a multicast tree with a small multicast delay variation under the multicast end-to-end delay constraint. DDVCA claims to outperform DVMA slightly in the multicast delay variation. However, in contrast to DVMA, the time complexity of DDVCA is lower.

In DDVCA, the minimum delay path algorithm and the SPT (Shortest Path Tree) is used. A SPT is constructed by combining all the shortest (i.e., minimum delay) paths from the source node to each destination node. The fundamental strategy of DDVCA comes from CBT (Core Based Tree)'s Core Router concept and the minimum delay path algorithm. The basic idea is described as follows. In DDVCA, for each network node, the SPT from it to all the destination nodes is constructed. The node whose SPT has the minimum multicast delay variation is selected as the central node. Then a checking process is done to examine whether the sum of the minimum delay between the source node and the current central node and the maximum multicast end-to-end delay of the SPT rooted at the central node satisfies the multicast end-to-end delay constraint. If the central node violates the constraint, it will be abandoned. In this case, the algorithm will go on to pick the node whose SPT has the next minimum multicast delay variation as the next possible central node and apply the same checking process until a central node which satisfies the constraint is found.

2.3. Notations and Definitions

A SPT has very good performance in terms of the multicast end-to-end delay. But selecting the shortest paths may lead to a violation of the delay variation constraint among nodes that are close to the source and nodes that are far away from it. Consequently, it may be necessary to select longer paths for some destination nodes in order to further reduce the multicast delay variation of the SPT. Therefore if we introduce higher delay paths to replace the minimum delay paths from the source to some destinations on the SPT, more trees with small multicast delay variations will be searched intuitively comparing to DDVCA.

We define the concepts of *proprietary second shortest path* and *partially proprietary second shortest path*. The proprietary second shortest path and partially proprietary second shortest path will be used as the higher delay path.

We denote the central node being checked as v_c. Let $T(v_c)$ represent the SPT rooted at v_c. For one destination node v_j, we define:

Proprietary links: links which are not shared by other destination nodes on $T(v_c)$.

Proprietary link Set (PS): all the proprietary links of v_j.

In Fig. 1, suppose Vc is the central node, and V2, V4, V5, V6 are destination nodes. For V6, its proprietary links are (V2, V3) and (V3, V6). So its proprietary link set is {(V2, V3), (V3, V6)}.

Proprietary second shortest path: the second shortest path from v_c to v_j, which is obtained by computing the shortest path from v_c to v_j after deleting l from the network topology G, $l \in$ PS. So the number of proprietary second shortest paths equals to the number of proprietary links for v_j. The proprietary second shortest path is actually the shortest path on the network topology G-$\{l\}$.

Partially proprietary links: links which are only shared by all its child destination nodes on $T(v_c)$.

Partially Proprietary link Set (PPS): all the partially proprietary links of v_j.

Also in Fig. 1, for V2, its partial proprietary link is (V1, V2). So its partially proprietary link set is {(V1, V2)}.

Partially proprietary second shortest path: the second shortest path from v_c to either v_j or a child destination node of v_j, which is obtained by computing the shortest path from v_c to either v_j or the child destination node of v_j after deleting l from the network topology G, $l \in$ PPS. The partially proprietary second shortest path is actually the shortest path on the network topology G-$\{l\}$.

For a multicast tree, it is easy to determine the proprietary link set or partially proprietary link set for a destination node. Then we can compute the proprietary second shortest paths or partially proprietary second shortest paths for a destination node using Dijkstra's Algorithm conveniently and quickly.

The characteristics of proprietary second shortest paths and partially proprietary second shortest paths guarantee that adding them to the SPT will not create a cycle, which is proved in Theorem 1. Thus other multicast paths on the SPT will not be interfered with.

Theorem 1: Let $T(v_c)$ be the SPT rooted at v_c. For any destination node $v_j \in M$ with $PS(v_j) \neq \emptyset$, use a

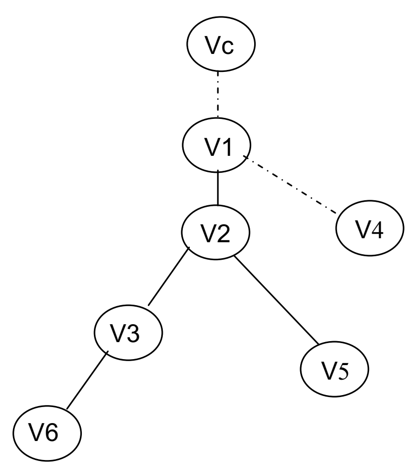


Figure 1. Illustration of proprietary links and partially proprietary links

0-7695-2423-0/05 $20.00 © 2005 IEEE

proprietary second shortest path to replace the shortest path will not create a cycle on $T(v_c)$.

Proof: Let $T^*(v_c)$ represent the SPT from v_c to all nodes in M-$\{v_j\}$. $T^*(v_c)$ is one part of $T(v_c)$. According to the above definition, all links in $PS(v_j)$ are not shared by any destination node in M-$\{v_j\}$ on $T(v_c)$. So no link in $PS(v_j)$ belongs to $T^*(v_c)$. The proprietary second shortest path is obtained by computing the shortest path after deleting the selected proprietary link from the associated network topology. All the shortest paths on $T^*(v_c)$ will still be the shortest paths for the updated network topology. Hence $T^*(v_c)$ is still one part of the improved SPT. The improved SPT will be constructed by combining the proprietary second shortest path with $T^*(v_c)$. If the replacement creates a cycle, the proprietary second shortest path must pass at least one node belonging to $T^*(v_c)$ except v_c. But for the proprietary second shortest path, the subpath from v_c to any node belonging to $T^*(v_c)$ is still the shortest path which coincides with the corresponding path on $T^*(v_c)$. It contradicts with the assumption of a cycle being created. Hence the replacement of a proprietary second shortest path will not create a cycle.

Similarly, we can prove that using a partially proprietary second shortest path to replace the shortest path will not create a cycle on $T(v_c)$.

3. A Heuristic Multicast Algorithm

3.1 Overview of DDVMA

DDVMA constructs a QoS multicast tree in the backbone network to transmit multicast messages from the source gateway to all the destination gateways. The wireless routes between each leader MH and its gateway are discovered and optimized by AODV routing protocol. The delay values of the wireless routes are collected for computation in DDVMA.

Comparing to DDVCA, the improvement of DDVMA is realized by using the proprietary second shortest path or partially proprietary second shortest path to replace the multicast path with the minimum end-to-end delay on the SPT. The improvement procedure can be seen as an optimization procedure, i.e., using a better path to optimize the QoS of the SPT. The optimization objective is to achieve smaller multicast delay variation under multicast end-to-end delay constraint.

The optimization procedure will stop when one of the following two cases occurs:

(1) The multicast delay variation has been decreased to a specified tolerance range or can not be decreased further, whichever occurs first.

(2) The maximum multicast end-to-end delay of the SPT exceeds the given upper bound.

During the optimization procedure, the tree should always keep a SPT structure for the associated network topology. At the beginning, the associated network topology is just the network topology G. After each replacement, the selected proprietary link or partially proprietary link will be excluded from the associated network topology.

For one destination node on the SPT, if its proprietary link set is not NULL, its partially proprietary link set will be NULL, and vice versa. Assume that we are checking the destination node with the minimum multicast end-to-end delay on the SPT. If its proprietary link set is not NULL which means it is a leaf node, we will check whether a proprietary second shortest path can be found to optimize the tree; if its partially proprietary link set is not NULL which means it is a non-leaf node, we will check whether partially proprietary second shortest paths can be found to optimize the tree.

3.2 A Formal Description of DDVMA

In this section, we will present a formal description of DDVMA as showed in Fig. 2. Two procedures are used, one is to deal with the destination node with at least one proprietary link and the other is to deal with the destination node with at least one partially proprietary link. The former is named as procedure P (Proprietary), the latter procedure PP (Partially Proprietary). They are described in subsection 3.3.

As mentioned before, the wireless routing delay between each gateway and the leader MH is used to compute the multicast end-to-end delay and the multicast delay variation in DDVMA. If a path ends at

```
begin
  for each network node v∈V do
    Construct the SPT T(v) from v to all the destinations
    If T(v) satisfies the multicast end-to-end delay constraint, optimize T(v) using procedure P or PP until the multicast delay variation cannot be improved under the multicast end-to-end delay constraint
  end of for v loop
  Choose the node with the smallest value of multicast delay variation under the multicast end-to-end delay constraint as the central node
  Construct the multicast tree by connecting the central node with both the source and all destinations
end
```

Figure 2. A formal description of DDVMA

a destination gateway, the wireless routing delay recorded at the destination gateway needs to be added to the path delay. Thus we can guarantee that the constructed multicast tree satisfies the QoS requirement of the multicast transmission among leader MHs in multiple MANETs attached to the backbone Internet.

3.3 Procedure P and PP

Procedure P starts out with a SPT and decreases the multicast delay variation by replacing the minimum delay multicast path with the appropriate proprietary second shortest path. If procedure P returns False, it means the SPT remains unchanged.

After $T(v_c)$ is modified, the network graph G' associated with it needs to exclude the selected proprietary link l (i.e., $G' = G' - \{l\}$) in order to keep the shortest path tree structure of $T(v_c)$. On the updated network topology G', the proprietary second shortest path keeps to be the shortest path and the improved SPT keeps to be the shortest path tree. The associated network topology will be updated each time the SPT is modified by procedure P.

Different from procedure P, the partially proprietary second shortest paths are used in procedure PP. If non-leaf node v_j is the destination node with the minimum multicast end-to-end delay on $T(v_c)$, some child nodes of v_j will also be destination nodes. Node v_j with all its child destination nodes forms a subset M' of M. $P(v_c,v_j)$ represents the multicast path from v_c to v_j on $T(v_c)$, and $T(v_j)$ represents the sub-SPT rooted at node v_j on $T(v_c)$. $P(v_c,v_j)$ is the common part of each multicast path $P(v_c,j)$, $j \in M'-\{v_j\}$. For each $j \in M'-\{v_j\}$, $P(v_c,j)$ will also be changed when $P(v_c,v_j)$ is replaced by the corresponding partially proprietary second shortest path $P'(v_c,v_j)$.

We propose the following strategy to handle the changes of $P(v_c,j)$ ($j \in M'-\{v_j\}$) caused by the change of $P(v_c,v_j)$ in procedure PP: compute the partially proprietary second shortest path $P'(v_c,j)$ as the new multicast path from v_c to j for each $j \in M'-\{v_j\}$. If procedure PP returns False, it means the SPT remains unchanged. Similar to procedure P, the associated network topology will be updated each time the SPT is modified in procedure PP.

3.4 An Illustrative Example of DDVMA

In the following, we will illustrate the operation of DDVMA with an example. We will contrast it with DDVCA, so we also use the given computer network topology in [7]. The network topology is showed in Fig. 3. For a group communication scenario, we denote Vs as the source gateway, V2, V5 and V9 as the destination gateways, i.e. M={V2, V5, V9}. The number in the parentheses near gateway g (including the source gateway and all the destination gateways) represents the corresponding wireless route delay W(g). Suppose the multicast end-to-end delay constraint Δ is 60. Because the wireless route delay between the source leader MH and the source gateway is 1, the multicast end-to-end delay constraint used in DDVMA will be 59 (i.e., 60-1). Table 1 shows the procedure of selecting a central node in DDVCA. Table 2 shows the corresponding procedure in DDVMA.

In Table 1, for each network node Vi, the minimum path delay between it and each destination gateway (i.e., the wireline transmission path delay in the backbone network, plus the corresponding wireless route delay recorded at the destination gateway) is listed in each column. Then dv(Vi), the multicast delay variation of the SPT rooted at Vi, is computed and listed in the bottom line of each column. From Table 1, we know the multicast delay variation of the SPT rooted at Vs, V7 and V8 are the minimum. Assume Vs is selected, we get the multicast tree which satisfies the multicast end-to-end delay constraint and achieves the multicast delay variation 8.

In Table 2, for each network node Vi, we also list the path delay between it and each destination gateway on the improved SPT rooted at Vi in each column. A * next to a delay value indicates that it is the delay of proprietary second shortest path or partially proprietary second shortest path. It means that the corresponding minimum delay paths on the SPT have been replaced by the proprietary or partially proprietary second shortest paths. Then dv(Vi), the multicast delay variation of the improved SPT, is computed and listed in the bottom line of each column. From Table 2, we know the multicast delay variation of the improved

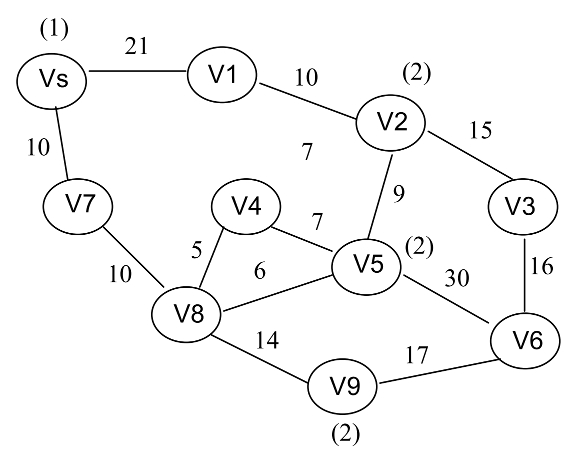


Figure 3. A given network topology G=(V, E)

Proceedings of the 2nd Int'l Conf. on Quality of Service in Heterogeneous Wired/Wireless Networks (QShine'05)
0-7695-2423-0/05 $20.00 © 2005 IEEE

Table 1. The manner by which DDVCA selects a central node

Network node Vi		Vs	V1	V2	V3	V4	V5	V6	V7	V8	V9
The minimum delay	Destination node Vj										
The multicast path delay	V2	33	12	2	17	9	11	33	24	14	28
between Vi and Vj	V5	28	21	11	26	9	2	32	18	8	22
	V9	36	38	28	35	21	22	19	26	16	2
The maximum delay from Vi to each Vj		36	38	28	35	21	22	33	26	16	28
The minimum delay from Vi to each Vj		28	12	2	17	9	2	19	18	8	2
dv(Vi)		8	26	26	18	12	20	14	8	8	26

Table 2. The manner by which DDVMA selects a central node

Network node Vi		Vs	V1	V2	V3	V4	V5	V6	V7	V8	V9
The minimum delay	Destination node Vj										
The multicast path delay	V2	33	12	2	17	22*	11	33	24	14	28
between Vi and Vj	V5	34*	21	11	26	13*	2	32	24*	14*	22
	V9	36	38	28	35	21	22	19	26	16	2
The maximum delay from Vi to each Vj		36	38	28	35	22	22	33	26	16	28
The minimum delay from Vi to each Vj		33	12	2	17	13	2	19	24	14	2
dv(Vi)		3	26	26	18	9	20	14	2	2	26

SPT rooted at V7 and V8 are both the minimum. Assume V7 is selected, we get the multicast tree which also satisfies the multicast end-to-end delay constraint and achieves the multicast delay variation 2.

So DDVMA can achieve the multicast tree with smaller multicast delay variation than DDVCA.

4. Performance Evaluation

4.1 Theoretical Analysis

We will show that DDVMA can achieve better efficiency in terms of the multicast delay variation than DDVCA by theoretical analysis.

In DDVMA, since each network node is checked, the source gateway v_s is also likely to be selected as the central node. The multicast tree will be constructed by connecting v_s to each destination gateway through the minimum delay path. Such a multicast tree is the SPT from v_s to all destination gateways. If it does not satisfy the multicast end-to-end delay constraint, obviously there does not exist any multicast tree, which will satisfy the multicast end-to-end delay constraint regulated by the input. This characteristic has also been stated in [7].

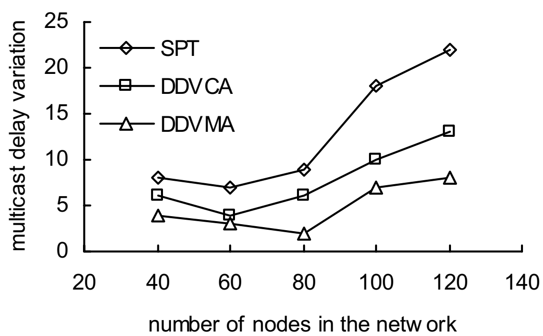
For each network node being checked, when the SPT is constructed, the multicast delay variation of the SPT will be used in DDVCA. But in DDVMA, we will execute procedure P and PP to further reduce the multicast delay variation of the SPT. Procedure P and PP can keep the SPT unchanged or return an improved SPT with smaller multicast delay variation. Procedure P or PP will be called repeatedly until the SPT cannot be improved. So DDVMA will search more possible multicast trees and achieve better efficiency in terms of the multicast delay variation than DDVCA. The illustration in subsection 3.4 just shows the characteristic.

The time complexity of DDVCA is $O(mn^2)$, m is the number of destination nodes, n is the number of network nodes. Since the time complexity of procedure P and PP is $O(n^2)$ as Dijkstra's Algorithm, the time complexity of DDVMA is the same as DDVCA.

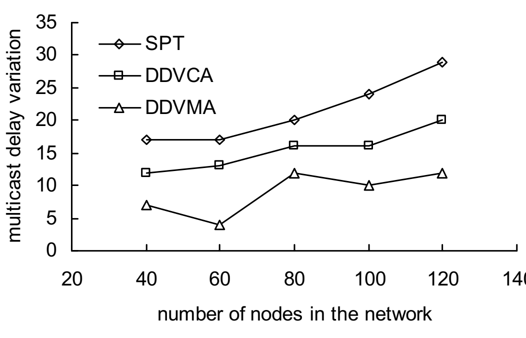
4.2 Simulation

Simulation experiments are conducted to examine the efficiency of DDVMA. Given two integers n and m ($n-1 \leq m \leq n(n-1)/2$), an interval [LD, UD], and an integer d, our random graph generator will generate a connected network topology graph with n nodes and m links. The delay on each link are an integer value in [LD, UD], which is in direct proportion to the length of the link. The degree of each node does not exceed d. The random graph generator first generates the n nodes. It then picks out two different nodes randomly. For the two nodes, if no direct link connects them and both of their node degrees are less than d, a new link between them will be added to the graph. This process is continued until m links are added to the graph. Similar random graph generation approach is introduced in [9].

In our simulation experiments, we generate five different network topology graphs. Their sizes begin from ones with 40, 60, 80, and up to 120 nodes. The delay on each link is drawn from the interval [1, 10]. Referring to Ref. [7], for a specified multicast group, the upper bound on the maximum multicast end-to-end delay, Δ, is set to be 1.5 times the minimum delay between the source node and the farthest destination node. In the simulation, we compare the proposed algorithm DDVMA with DDVCA and the SPT Algorithm produced from Dijkstra's Algorithm. We will evaluate the multicast delay variations and multicast end-to-end delays of the three algorithms. For each network, we investigate two cases, one is that the destination nodes in the multicast group will occupy 5% of the total nodes on the network and the other is 20%. For each case, we generate twenty different multicast groups randomly. Then twenty multicast trees will be obtained by each algorithm. We calculate the average over the multicast delay variations of the twenty multicast trees for each algorithm. The average value will be used to evaluate the efficiency of the algorithm in terms of the multicast delay variation.



(a)



(b)

Figure 4. A comparison on the multicast delay variations of the three different algorithms: (a) multicast group sizes equal to 5% of the number of network nodes, (b) multicast group sizes equal to 20% of the number of network nodes

Fig. 4 shows the simulation results of multicast delay variations. Fig. 4(a) corresponds to the multicast groups of sizes equal to 5% of the number of network nodes. It can be regarded as the scenario that multicast nodes are distributed sparsely on the network. Fig. 4(b) corresponds to the multicast groups of sizes equal to 20% of the number of network nodes. It represents the scenario that multicast nodes are distributed densely on the network. We observe that the trees constructed by DDVMA have an average multicast delay variation that is always smaller than that of the SPT and DDVCA trees. With the ratio of the multicast group size to the number of network nodes increasing from 5% to 20%, it is apparent that the multicast delay variation of DDVMA performs much better than that of DDVCA. The performance of the SPT Algorithm is the worst in terms of the multicast delay variation among the three algorithms.

To evaluate the performance of each algorithm, we also calculate the average over the maximum multicast end-to-end delays of the obtained multicast trees for each algorithm. Fig. 5 shows the simulation results on the multicast end-to-end delays of different algorithms. It corresponds to the case in which the destination nodes in a multicast group occupy 5% of the total network nodes. The simulation result of the 20% case is similar. We only present and discuss the 5% case. We observe that the multicast end-to-end delay of DDVCA performs better than that of DDVMA, but not much. It can be explained by the design of DDVMA. In DDVMA, we improve the multicast delay variations of the SPTs by introducing higher delay paths. If the delay of the accepted new path exceeds the maximum multicast end-to-end delay of the SPT, the maximum multicast end-to-end delay of the multicast tree will increase. But if the delay of the new path is so high that the multicast delay variation of the SPT is increased, the path will not be accepted. So in average,

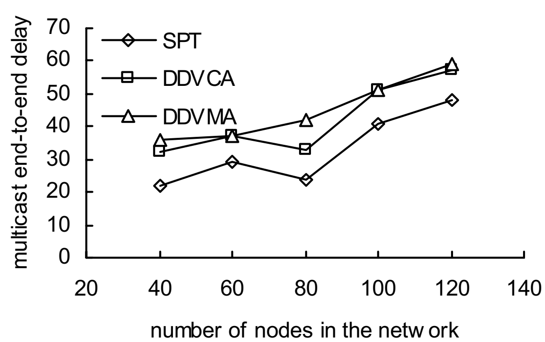


Figure 5. A comparison on the multicast end-to-end delays of the three algorithms

we can see that DDVMA and DDVCA have competing performance on multicast end-to-end delays. The SPT Algorithm has the best performance in terms of the multicast end-to-end delay inherently.

5. Conclusions

In this paper, a heuristic multicast algorithm DDVMA is developed for constructing the multicast tree spanning the source gateway and all the destination gateways in the backbone network. Two procedures- procedure P and procedure PP are proposed to generate higher delay paths which are used to replace the corresponding shortest paths on the SPT. Procedure P and PP are integrated into it to further reduce the multicast delay variation of the SPT rooted at the stand-by central node.

We use existing AODV protocol to discover the routes between the leader MH and its gateway. Each gateway records and updates the delay information of the wireless route. Combined with wireless routes between each leader MH and its gateway, the QoS multicast tree obtained by DDVMA can support the communications among leader MHs in multiple MANETs attached to the backbone Internet. Furthermore, the multicast tree can satisfy the multicast end-to-end delay constraint and achieve smaller multicast delay variation than the multicast tree obtained by DDVCA known to be the best algorithm for the DVBMT problem. Also DDVMA can be implemented in an IP routing protocol. This makes our solution simple and feasible to QoS group communications in the heterogeneous network.

Theoretical analysis is made on DDVMA and shows its correctness. Computer simulations also compare our algorithm with DDVCA and the SPT Algorithm in terms of multicast delay variations and multicast end-to-end delays. The simulation results show that DDVMA can achieve the smallest multicast delay variation with a litter higher multicast end-to-end delay than DDVCA.

Acknowledgement

This work is supported in part by the University Grant Council of Hong Kong under the CERG grant PolyU 5170/03E, by the National Natural Science Foundation of China under Grant No. 60473089, No. 60003006 (jointly supported by Bell Lab Research China) and No. 70101006, by the National 863 Plan of China under Grant No. 2001AA121064.

References

[1] C. Åhlund and A. Zaslavsky, "Extending global IP connectivity for ad hoc networks," Telecommunication Systems, vol. 24, no. 2-4, pp. 221-250, 2003.

[2] U. Jönsson, F. Alriksson, T. Larsson, P. Johansson and G. Q. M. Jr, "MIPMANET- mobile IP for mobile ad hoc networks," in Proc. IEEE MobiHoc, pp. 75-85, 2000.

[3] Y.-C. Tseng, C.-C. Shen and W.-T. Chen, "Integrating mobile IP with ad hoc networks," IEEE Computer, vol. 36, no. 5, pp. 48-55, 2003.

[4] M. Ghassemian, P. Hofmann, C. Prehofer, V. Friderikos and H. Aghvami, "Performance analysis of Internet gateway discovery protocols in ad hoc networks," in Proc. IEEE WCNC, pp. 120-125, 2004.

[5] P. M. Ruiz and A. F. Gómez-Skarmeta, "Enhanced Internet connectivity for hybrid ad hoc networks through adaptive gateway discovery," in Proc. IEEE LCN, pp. 370-377, 2004.

[6] E. Nordström, P. Gunningberg and C. Tschudin, "Comparison of forwarding strategies in Internet connected MANETs," Mobile Computing and Communications Review, vol. 8, no. 4, pp. 72-76, 2004.

[7] P.-R. Sheu and S.-T. Chen, "A fast and efficient heuristic algorithm for the delay- and delay variation-bounded multicast tree problem," Computer Communications, vol. 25, no. 8, pp. 825-833, 2002.

[8] G. N. Rouskas and I. Baldine, "Multicast routing with end-to-end delay and delay variation constraints," IEEE Journal on Selected Areas in Communications, vol. 15, no. 3, pp. 346-356, 1997.

[9] B. M. Waxman, "Routing of multipoint connections," IEEE Journal on Selected Areas in Communications, vol. 6, no. 9, pp. 1617-1622, 1988.