We are IntechOpen,
the world's leading publisher of Open Access books
Built by scientists, for scientists

4,800
Open access books available

122,000
International authors and editors

135M
Downloads

154
Countries delivered to

Our authors are among the

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

BOOK CITATION INDEX
CLARIVATE ANALYTICS
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Multiple Multicast Tree Construction and Multiple Description Video Assignment Algorithms

Osamah Badarneh[1] and Michel Kadoch[2]
[1]*Yarmouk University*
[2]*Ecole de Technologie Superieure*
[1]*JORDAN*
[2]*CANADA*

1. Introduction

In this chapter, we introduce novel algorithms for constructing multiple multicast tree and assigning multiple description (MD) video to a group of heterogeneous multicast destinations. Our main objective is to increase the number of assigned MD video to each destination node. In order to achieve our objective, we propose to employ the independent-description property of MDC (multiple description coding) along with multiple multicast tree. We mean by independent-description property of MDC the following. If there are three video descriptions, for example, then receiving any subset of video descriptions, i.e., $(\{VD_1, VD_2, VD_3\}, \{VD_1, VD_2\}, \{VD_1, VD_3\}, \{VD_2, VD_3\}, \{VD_1\}, \{VD_2\}, \{VD_3\})$ of the video descriptions will reproduce the original video in different qualities depending on the number of video descriptions received.

A main issue of video multicasting for heterogeneous destinations is the assignment of video descriptions and the construction of multicast trees. However, the assignment of MD video and the construction of multicast tree can greatly affect the user satisfaction (i.e., affect the number of assigned video description to each destination and hence affect the quality of the received video. However, many questions are raised: How multiple multicast tree should be constructed? And how MD video should be assigned? Is it better to construct multiple multicast tree first and then assign the video descriptions? Or is it better to assign the video descriptions first and we then construct multiple multicast tree? Should we perform that in a distributed manner or in a centralized one? Does the independent-description of MDC increase the user satisfaction?

To answer these questions, we propose different algorithms to construct multiple multicast tree and to assign MD video. The proposed algorithms are: Serial MDC, Distributed MDC, Centralized MDC, and sequential MDC. Serial MDC algorithm constructs multiple paths, to each destination, and assigns a different video description to each of them. After that, it constructs multiple multicast tree based on the assignment of MD video. Distributed MDC algorithm assigns MD video and constructs multiple multicast tree in parallel and in distributed fashion. In Centralized MDC, the assignment of MD video and the construction of multiple multicast tree are performed in a centralized manner. However, Centralized MDC first constructs multiple multicast tree and then assigns different video description to each multicast tree. Finally, Sequential MDC sequntially assigns MD video to each multicast tree.This means that all destinations should be assigned the first description. Then,

destinations that need another description should be assigned the second video description, and so on. The main difference between Sequential MDC and Centralized MDC algorithms is that the former does not employ the independent-property of MDC.

We evaluate and compare our proposed algorithms under different network conditions. For example, network size, and multicast group size. Simulation results demonstrate that, indeed, the way of multicast trees construction and the assignment of MD video can greatly affect the user satisfaction. In addition, simulation results show that MDC can achieve higher user satisfaction compared to Layered Coding (LC) with a small cost in terms of number of pure forwarders nodes, bandwidth utilization, and aggregate tree delay. Furthermore, simulation results show that the independent-description property of MDC can increase the user satisfaction.

The rest of this chapter is organized as follows. In the next section, we present the related work. In section 3, we present our network model and problem formulation of video multicasting. In Section 4, we describe our proposed algorithms for constructing multiple node-disjoint multicast trees and assigning MD video. In Section 5, we evaluate our proposed algorithms. The complexity analysis of the protocols is presented in Section 6. Finally, our conclusions are presented in Section 7.

2. Related work

An ad hoc network is a multihop wireless network without a preinstalled infrastructure or centralized administration. It can be deployed in situations where infrastructure is unavailable or where temporary network is needed. In this network, nodes are free to move randomly anytime, anywhere, and arrange themselves as required. Since nodes are often not within the radio transmission range of each other, each node operates not only as a host but also as a router, forwarding packets for other mobile nodes. In a typical ad hoc environment, mobile nodes work as a group to accomplish a certain task. Hence, multicast is very useful and efficient means of supporting group-oriented applications. Multicast is an essential technology for many applications such as video distribution and group video conferencing, and results in bandwidth and power savings as compared to multiple unicast sessions.

Many researches over the last several years have focused on unicast and multicast video transmission over wireless ad hoc networks (Wei & Zakhor, 2007; Mao, Cheng, Hou & Sherali, 2006; Agrawal et al., 2006; Chow & Ishii, 2008; Mao, Hou, Cheng, Sherali, Midkiff & Zhang, 2006; Mao et al., 2003). The main objective of these researches is to improve the quality of the received video by exploiting the error resilience properties of MDC along with multiple paths. In other words, MD video are encoded and transmitted over different paths to each destination node. If only any path is broken, packets corresponding to the other descriptions on the other paths can still arrive at the destination node on time.

MDC has been proposed as an alternative of the LC (Layered Coding) technique. In contrast to LC, MDC is a coding technique which fragments a single media stream into independent bit-streams, where the multiple bit-streams are referred to as multiple descriptions. In order to decode the media stream, any description can be used (we referred to as "independent-description" property (Badarneh et al., 2008)); however, the quality improves with the number of descriptions received in parallel. The idea of MDC is to provide error resilience to media streams. Since an arbitrary subset of descriptions can be used to decode the original stream, network congestion or packet loss, which is common in best-effort networks

such as the Internet, will not interrupt the stream but only cause a temporary loss of quality. The quality of a stream can be expected to be roughly proportional to data rate sustained by the receiver (Goyal, 2001; Puri & Ramchandran, 1999).

Video multicast over wireless ad hoc networks with path diversity has been studied in Wei & Zakhor (2007); Mao, Cheng, Hou & Sherali (2006); Agrawal et al. (2006); Chow & Ishii (2008). Chow and Ishii have proposed a multicast protocol for video transmission called MT-MAODV (Multiple Trees Multicast Ad Hoc On-demand Distance Vector) (Chow & Ishii, 2008). An extension to the well-knownMAODV to construct two optimally disjoint multicast trees in a single routine for video multicast was proposed. MDC scheme is used to split the video into several independent and equally important video descriptions. Each description is transmitted over different tree. In (Mao, Cheng, Hou & Sherali, 2006), the authors introduced a multicast approach for multiple description video over ad hoc networks. An application-centric, cross-layer routing approach with the objective of minimizing the over all video distortion was proposed. In this approach multiple source trees for MD video multicast are used. Furthermore, each description is coded into a base layer and number of enhancement layers. Packets belonging to the same description from both the base layer and enhancement layers are transmitted on the same tree. The authors showed that this approach can effectively deal with frequent link failures and diverse link qualities in wireless ad hoc networks. Agrawal et al. have presented a multiple tree protocol called Robust Demand-driven Video Multicast Routing (RDVMR) (Agrawal et al., 2006). RDVMR explores the path diversity and error resilience properties of MDC. RDVMR deploys a novel path based Steiner tree heuristic to reduce the number of forwarding nodes in each tree, and constructs multiple trees in parallel with a reduced number of common nodes among them to provide robustness against path breaks and to reduces the total data overhead. Two multiple tree multicast routing protocols were presented in (Wei & Zakhor, 2007). Serial MDTMR protocol (Multiple Disjoint Trees Multicast Routing) constructs two disjoint multicast trees in a serial fashion. However, in order to reduce routing overhead and construction delay of serial MDTMR, parallel MNTMR (Multiple Nearly-disjoint Trees Multicast Routing) was suggested. This protocol constructs two nearly-disjoint multicast trees in a single routine by dividing the network virtually into two parts and tree construction is carried out simultaneously at both virtual topologies. Both serial MDTMR and parallel MNTMR protocols explore MDC to provide robustness for video multicast applications. In order to improve the quality of the received video, the video was split into two descriptions and each description was transmitted over a different tree.

3. Network model and problem formulation

3.1 Network model for multicasting

We consider a multi-hop wireless ad hoc network with $\mathcal{V}$ nodes. The nodes communicate with each other via wireless links. Each node in the network can communicate directly with a subset of the other nodes in a network. A node v can transmit directly to node u if the both nodes are within the transmission range of each other. We modeled a wireless ad hoc network as weighted $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is a set of wireless nodes each with random location and $\mathcal{E}$ is a set of wireless communication links between the nodes. A link between node pair $\{v, u\}$ indicates that both nodes v and u are within each other's transmission range. The nodes in set $\mathcal{V}$ can be of the following three types:

– **Multicast source node**: The node that sends out the multicast video packets. We denote it by $\mathcal{S}$.

– **Destination node**: A node that receives the multicast video packets. The set of destination nodes in a multicast tree is denoted by $\mathcal{Y} \subseteq \mathcal{V} - \mathcal{S}$

– **Forwarder node**: A node that is an intermediate hop in the path from the source $\mathcal{S}$ to a destination node in $\mathcal{Y}$. It is denoted by $\mathcal{F}$.

Two positive real-valued functions are defined on a link $e = \{v,u\} \in \mathcal{E}$, namely:

– **Link Delay**: $d(e) \in \Re^+$.

– **Link Bandwidth**: $Bw(e) \in \Re^+$.

In this work, we focus on the network layer, i.e., the construction of multiple multicast trees and the assignment of MD video. We assume that the physical and MAC layers dynamics, such as the link delay and bandwidth, are translated into the network layer parameters. These parameters can be measured at every node and distributed through the network using LSAs (Link State Advertisements) (Clausen & Jacquet, year 2003).

DEFINITION 1: *A path p from the multicast source $\mathcal{S}$ to a destination node in $\mathcal{G}$ is defined as a list of nodes $(v_1, v_2, \cdots, v_k)$ such that $\forall j, 1 \leq j \leq k, e_j = (v_j, v_{j+1}) \in \mathcal{E}$ and no node appears more than once.*

The delay of the path p is the sum of all link delays, that is,

$$d(p) = \sum_{j=1}^{k-1} d\left(e_j\right) \tag{1}$$

The bandwidth of the path p is the minimum available bandwidth of all links, which is defined as

$$Bw(p) = \min_{e_i \in p} \{Bw(e_i)\} \tag{2}$$

In case of $\mathcal{K}$ node-disjoint paths, $P = \{p_1, p_2, \cdots, p_{\mathcal{K}}\}$, then the delay of the $\mathcal{K}$ paths for a destination node is:

$$d(P) = \max_{p_j \in P} \left\{d\left(p_j\right)\right\} \tag{3}$$

Let L be the number of the multicast trees constrcuted to meet the destinations' requirments, then the delay of the tree-aggregate $T = t_1 \cup t_2 \cup \cdots t_L$ is defined as:

$$d(T) = \max_{l \in |1...L|} d(t_l) \tag{4}$$

where $d(t_l)$ is the delay of a multicast tree t_l, which is defined as the longest delay from the source $\mathcal{S}$ to the destinations on t_l, that is:

$$d(t_l) = \max_{p_i \in t_l} \{d(p_i)\}, \quad i = |1...m| \tag{5}$$

where m is the number of destinations on t_l.

3.2 Problem formulation

Our problem of MD video assignment can be formulated as follows: Given a wireless ad hoc network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with $\mathcal{N}$ number of MD video, a link delay, a link bandwidth, a source $\mathcal{S}$, and a set of destinations $\mathcal{Y} = \{R_1, R_2, \cdots, R_m\}$ such that each destination node $R_i \in \mathcal{Y}$ requires a preference number of MD video, then construct multiple node-disjoint multicast tree spanning $\mathcal{Y} \cup \mathcal{S}$ such that the total number of the assigned video descriptions to each destination is maximized. That is:

$$maximize \left\{ \mathcal{N}_{asg}(R_i) \right\} \tag{6}$$

where $\mathcal{N}_{asg}(R_i)$ is the number of the assigned video descriptions to the destination R_i.

To minimize the delay of every path from the source $\mathcal{S}$ to each destination $R_i \in \mathcal{Y}$, the shortest path tree algorithm is deployed.

4. Multiple multicast tree construction and multiple description video assignment algorithms

4.1 Serial MDC algorithm

The MD video assignment and multiple multicast trees construction algorithms are shown in algorithms $1 - 4$. At the beginning, let the multicast source has a partial topology that contains multiple paths to each destination, as shown in Fig. 1(a). Following, it arranges the destinations that require one and two video descriptions in a descending order according to their number of node-disjoint paths in the sets x and y, respectively. After that, it checks the destinations in the set y if any of them has only one path, if yes, it adds it to the set x. At the end of these steps, the sets x and y contain the destinations arranged in a descending order according to their number of paths. After that, the source node runs the algorithms $1 - 4$. We use the two colors: red and green to refer to the first and second descriptions, respectively. The multicast source starts with the set y and constructs its red (R) and green (G) paths for each destination if possible. To find the R-path, the green nodes (G-nodes) should be removed because they already have been assigned a description and they cannot be on another tree. However, to find the G-path, the red nodes (R-nodes) should be removed. The R and G paths are constructed using shortest path algorithm (in terms of delay).

When the set y is empty, the source node starts with the set x. Since any description can reproduce the original video signal, this, what we referred to as independent-description property of MDC, therefore the multicast source will assign any color (R or G) to each destination in the set x.

Based on the sets of multiple paths $\mathcal{K}_{R_i}$ (the R and G paths) for every destination R_i, then the multicast source S constructs multiple multicast trees for the video transmission according to algorithm 4. That is, all nodes that have been assigned the same color are attached to the same tree. For example, the nodes that have been assigned the R-color are attached to the first tree (R-tree) and the nodes that have been assigned the G-color are attached to the second tree (G-tree). Fig. 1 is an illustrative example.

4.2 Distributed MDC algorithm

In this algorithm the assignment of MD video and the construction of multiple multicast trees are performed in a distributed manner. Each node in the network will only select one video description to transmit it to its neighbor nodes. This condition is to ensure disjointness between multicast trees. Destination nodes are responsible to construct multiple node-disjoint paths to the multicast source, node S . Each destination node will select a number of disjoint
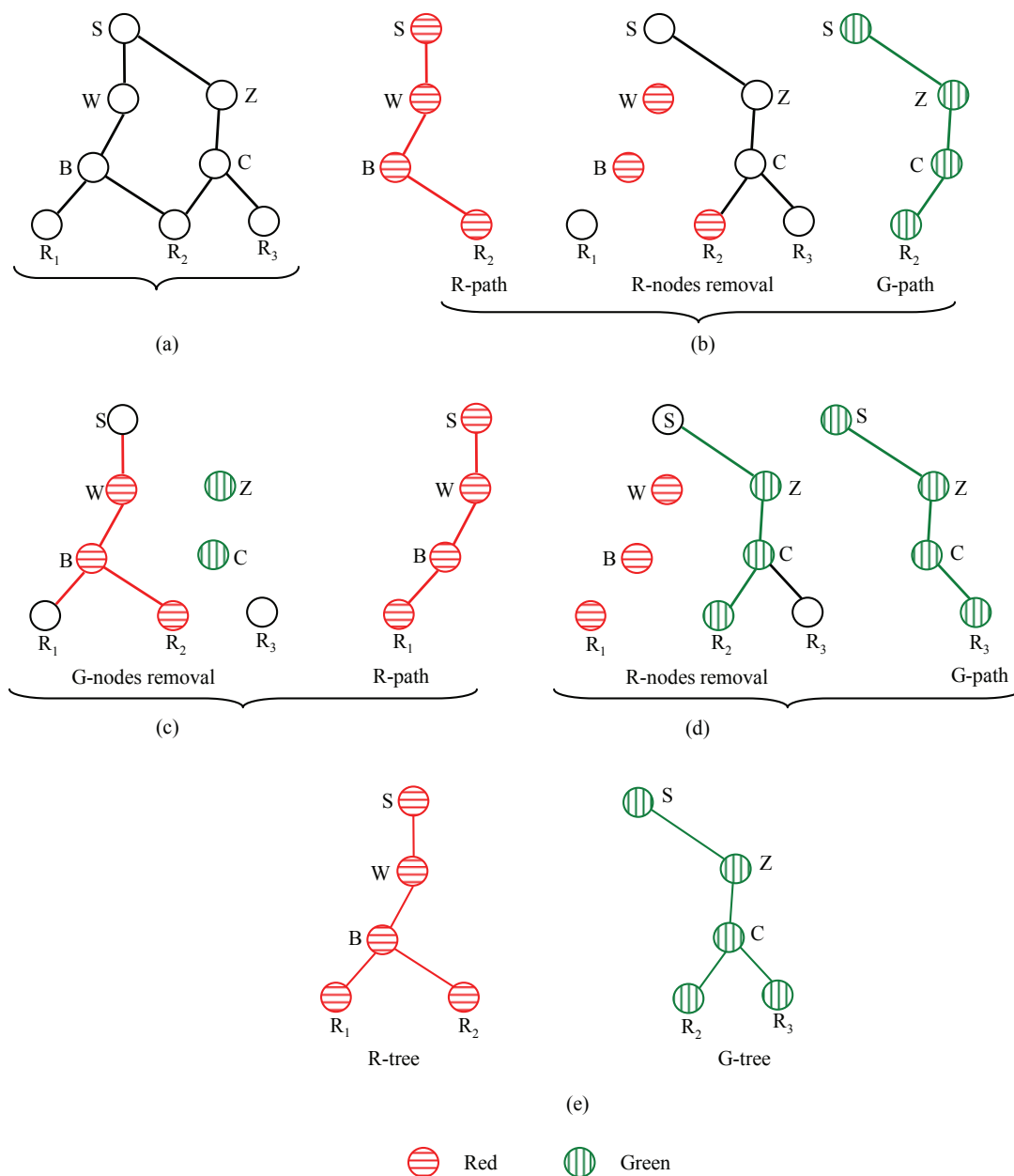
Fig. 1. Serial MDC: An illustrative example: (a) Partial topology. (b) Multiple paths construction and nodes removal for destination R_2. (c) Multiple paths construction and nodes removal for destination R_1. (d) Multiple paths construction and nodes removal for destination R_3. (e) Multiple multicast trees construction.

paths equal to its preference number of MD video. If there are two paths have the same video description, the one with shortest delay will be chosen.

The source node $\mathcal{S}$ will broadcasts the information of the available MD video and the bandwidth requirements for each description to its neighbor nodes. Neighbor nodes that have enough bandwidth will randomly choose one description and rebroadcasts it along with its bandwidth requirement to its neighbor nodes. As we mentioned previously, each node will only choose one description to transmit it to its neighbor nodes to maintain disjointness between multicast trees. This process will continue to reach a destination node.

Algorithm 1 Serial MDC

1: **Given**: $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, set x, and set y
2: **for** $\forall i \in$ set y **do**
3:     $Z =$ set y
4:     Construct a R_{path} using algorithm 2
5:     Construct a G_{path} using algorithm 3
6: **end for**
7: **for** $\forall k \in$ set x **do**
8:     $Z =$ set $x \cup$ set y
9:     Construct a R_{path} using algorithm 2
10:     **if** the $R_{path} = \phi$ **then**
11:         Construct a G_{path} using algorithm 3
12:     **end if**
13: **end for**

When a destination node receives information about a video description, it will rebroadcast this information to its neighbor nodes. This means also that a destination node could be a forwarder node. If this destination node has enough bandwidth it will select another description to receive. After a destination node selects its proper paths it will send this information to the source node.

After the multicast source $\mathcal{S}$ receives the paths for each destination node, it constructs multiple node-disjoint multicast trees. To do so, nodes that have the same video description should be added to the same tree. Algorithm 5 describes the construction of multiple multicast trees.

Fig. 2 shows an example of MD video assignment and construction of multiple multicast trees. The multicast source S broadcasts information about two video descriptions (VD_1, and VD_2) to its neighbor nodes, nodes W, and Z. Each node will randomly select one video description to rebroadcast. Therefore, node W selects VD_1 and node Z selects VD_1. After that, nodes W and Z will rebroadcast this information to their neighbors nodes, nodes B, and C. This process will continue until this information reached the destination nodes, nodes R_1, R_2, and R_3. Destination nodes R_1, and R_3 will select the paths $S \rightarrow W \rightarrow B \rightarrow R_1$, and $S \rightarrow Z \rightarrow C \rightarrow R_3$, respectively, to receive VD_1. The destination node R_2 has two paths with the same description, description VD_1. Therefore, it will select the path with minimum delay. Assume the path $S \rightarrow W \rightarrow B \rightarrow R_2$ is selected. Note that destination node R_2 receives the same video description through different paths. This can be related to the randomness of choosing a video description. Finally, the multicast source S will construct only one multicast tree using algorithm 5. Fig. 2(c) shows multicast tree t_1.

Algorithm 2 R_{path} Construction

1: **for** $\forall j \; G_{path} \in Z$ **do**
2:     $\mathcal{P} =$ Parents of G_{nodes}
3:     $\mathcal{V} \leftarrow \mathcal{V} - \mathcal{P}$
4: **end for**
5: Construct a R_{path} using the shortest path (in terms of delay) algorithm

---

Algorithm 3 G_{path} Construction

1: **for** $\forall j\ R_{path} \in Z$ **do**
2:     $\mathcal{P}$ = Parents of R_{nodes}
3:     $\mathcal{V} \leftarrow \mathcal{V} - \mathcal{P}$
4: **end for**
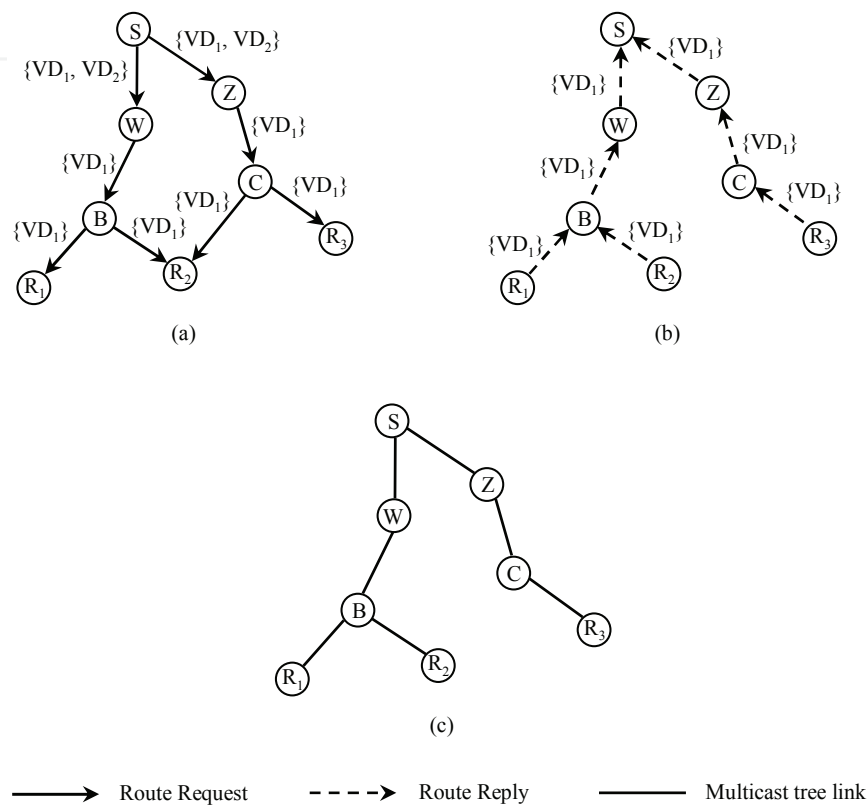5: Construct a G_{path} using the shortest path (in terms of delay) algorithm

Fig. 2. Distributed MDC algorithm: (a) Route Request broadcasts. (b) Route Reply unicast. (c) Multicast tree construction.

Algorithm 4 Serial MDC: Multiple Multicast Tree Construction

1: **Given**: Z = set $x \cup$ set y
2: **for** $\forall i \in Z$ **do**
3:     **if** i has R_{color} **then**
4:         Add i to R_{tree}
5:     **else if** i has G_{color} **then**
6:         Add i to G_{tree}
7:     **end if**
8: **end for**

Algorithm 5 Distributed MDC: Multiple Multicast Tree Construction

---

1: **for** $i = 1$ *to* $\mathcal{V}$ **do**
2:    **if** node i has the 1st video description **then**
3:       Add node i to tree t_1
4:    **else**
5:       Add node i to tree t_2
6:    **end if**
7: **end for**

4.3 Centralized MDC algorithm

Before the construction of multiple node-disjoint multicast trees and the assignment of MD video, the multicast source S starts with constructing individual Multiple Node-Disjoint Paths (MNDP), with minimum delay, to each destination in the multicast group to meet the number of video descriptions required.

DEFINITION 1: MNDP problem: *consider a network represented by a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and a bandwidth constraint $\mathcal{W}$, find a MNDP, set P_i, from the multicast source node $\mathcal{S}$ to the destination node R_i such that:*

1. $d\left(p_{ij}\right)$ is minimized, $\quad \forall\, p_{ij} \in P_i$

2. $Bw\left(p_{ij}\right) \geq \mathcal{W}, \quad \forall\, p_{ij} \in P_i$

Algorithm 6 describes how MNDP are constructed. Before constructing multiple nodedisjoint paths to each destination, we first remove all links with capacity less than the bandwidth requirement, and then we construct multiple shortest paths (in terms of delay) on the residual network. Based on the sets of MNDP constructed, then multicast heuristic algorithm constructs Multiple Node-Disjoint Multicast Trees (MNDMT) for the video transmission, as shown in Algorithm 7.

Algorithm 6 Multiple Node-Disjoint Paths

1: $P_i = \phi$ /* MNDP set */
2: For each destination R_i do
3: Let $\mathcal{G}^*$ be equal to $\mathcal{G}$
4: **repeat**
5:    Find a shortest path p_{ij} to R_i (in terms of delay) in $\mathcal{G}^*$ such that $Bw\left(p_{ij}\right) \geq \mathcal{W}$
6:    Add p_{ij} to P_i
7:    Remove all forwarding nodes of p_{ij} in $\mathcal{G}^*$
8: **until**
   The number of paths in P_i equal to the number of video descriptions required

---

As a simple example, we consider the partial network topology in Fig. 3(a), with a requirement of two descriptions for destination R_2 and one description for both destinations R_1 and R_3, to demonstrate the construction of multiple multicast trees. According to Algorithm 6, there are three path sets (Fig. 3(b)) P_1, P_2, and P_3 from the source S to the destinations R_1, R_2, and R_3, where $P_1 = \{p_{11}\} = \{S \to W \to B \to R_1\}$, $P_2 = \{p_{21}, p_{22}\} = \{S \to W \to B \to R_2, S \to Z \to C \to R_2\}$, and $P_3 = \{p_{31}\} = \{S \to Z \to C \to R_3\}$.

In Fig. 3(c)-(e), we show an example of multiple multicast trees construction using MNDMT. According to Algorithm 7, Step 4, the destination R_2 has the maximum number of paths, which is set P_2, (two paths); therefore we have two multicast trees according to step 5, namely, $t_1 = p_{21}$ and $t_2 = p_{22}$ as seen in Fig. 3(c). The path p_{11} of the destination R_1 will be added to t_1 (Fig. 3(d)), according to Step 8, since it intersects t_1 with the most links. Because $P_1 = \phi$, then the algorithm picks up the next destination, R_3, and adds its path p_{31} to tree t_2 (Fig. 3(e)) according to Step 8. Since all the paths of each destination have been added, then the algorithm ends.

After constructing multiple multicast trees, Algorithm 8 assigns different video description to each tree. Therefore, trees t_1 and t_2 are assigned the first and second descriptions, respectively. Since any description can reproduce the original video signal, this we referred to as independent-description property of MDC, therefore the destination R_3 will be able to reproduce the original video signal. It is worth noting that if LC technique is used instead of MDC and according to Chen-LC algorithm, only one multicast tree will be constructed. Thus, they will be only assigned the basic layer.

4.4 Sequential MDC algorithm

Sequential algorithm constructs multiple disjoint multicast trees and assigns MD video to the destination nodes in a centralized fashion. However, the main difference between sequential MDC and centralized MDC algorithms is that the assignment of MD video is executed in a sequential way. This means that all the destination nodes should be first assigned the first video description (VD_1), then the destination nodes that require a second description they will be assigned the second video description (VD_2) and the destination nodes that require a third description they be assigned the third video description (VD_2) and so on. Therefore, to perform the assignment of MD video in a sequential way, the destination nodes on each multicast tree should be superset of the later, i.e., $t_L \subseteq t_{L-1} \cdots \subseteq t_2 \subseteq t_1$. Algorithm 7 is deployed to construct multiple disjoint multicast trees, and then algorithm 9 is executed to form the final version of the multiple multicast trees. After that, the trees $t_1, t_2, \cdots, t_L$ will be assigned the first, the second and the Lth description, respectively. It is worth pointing out that Sequential MDC algorithm does not employ the independent-property of MDC.

Algorithm 7 Multiple Node-Disjoint Multicast Trees

1: **for** $i = 1$ *to* m **do**
2:     Find the set of MNDP P_i by algorithm 6
3: **end for**
4: Find a set P_i that has the maximum number of paths
5: initially, Let $T = P_i$, i.e., $t_1 = p_{i1}, t_2 = p_{i2}, \ldots, t_L = p_{iL}$
6: **for** $i = 2$ *to* m **do**
7:     Add each path in P_i to T as follows:
8:     Find a path $p_{ij} \in P_i$ such that it intersects a tree $t_k \subset T$ not covering R_i with the most links, and add p_{ij} to t_k
$$t_k \leftarrow t_k + p_{ij}$$
9:     Remove p_{ij} from P_i
$$P_i \leftarrow P_i - p_{ij}$$
10:     Repeat Steps (8) and (9) until $P_i = \phi$
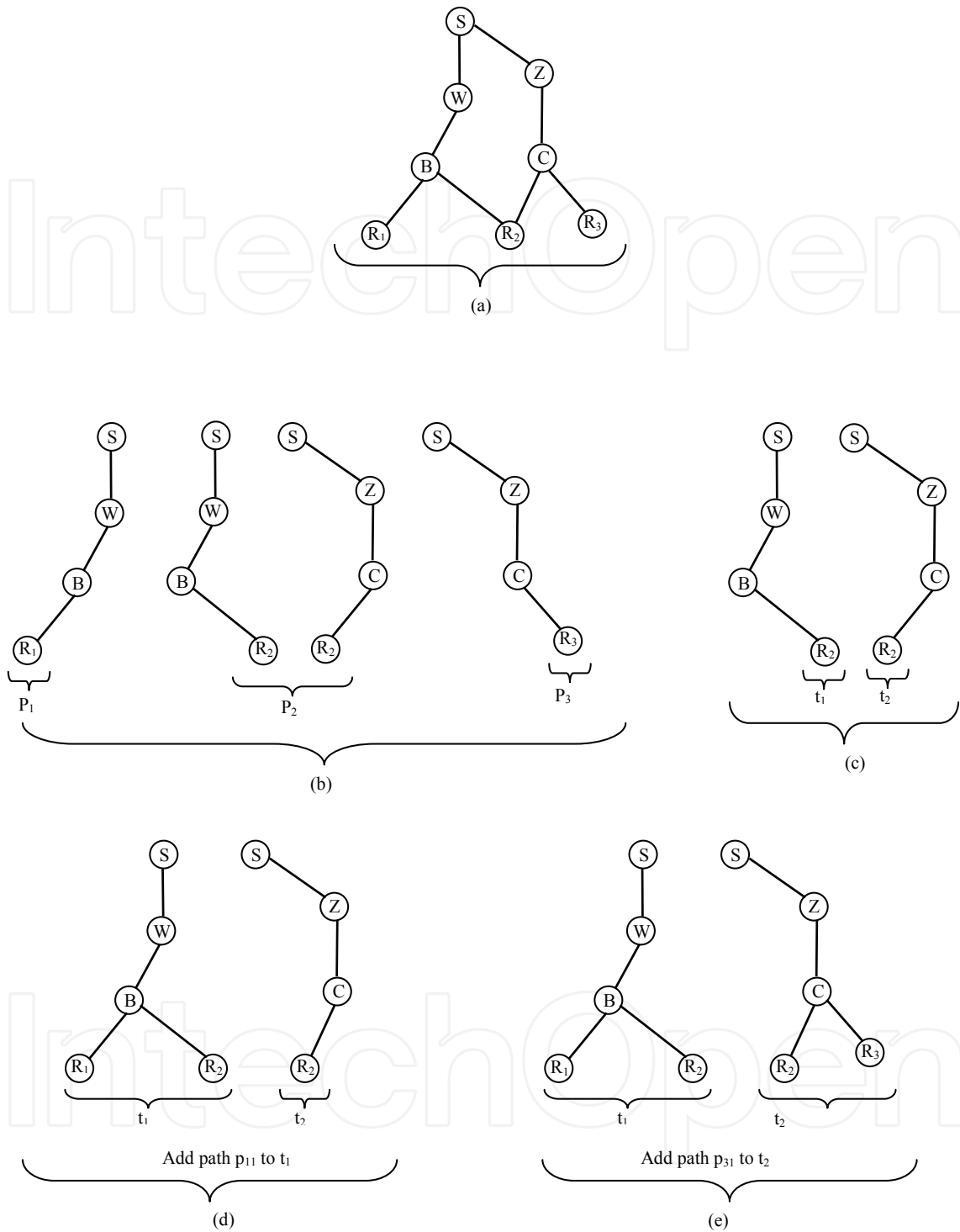11: **end for**

Fig. 3. Centralized MDC algorithm

We use Fig. 3(a), to explain how sequential MDC algorithm constructs multiple disjoint multicast trees. At the end of algorithm 7, two disjoint multicast trees are constructed, namely, t_1 and t_2 as seen in Fig. 3(e). However, in order to perform sequential assignment of MD video, R_3 should be connected to t_1. And because Sequential MDC algorithm maintains totally

---

Algorithm 8 MD video assignment

1: For $i = 1$ *to L*   /*L is the number of the multicast trees constructed*/
2: For $j = 1$ *to n*   /*n is the number of MD video, $(L \leq n)$*/
3: If $Bw(t_i) \geq Bw(VD_i)$ then
   $VD_i \rightarrow t_i$

---

disjoint multicast trees, therefore, only one multicast tree, t_1, is constructed as shown in Fig. 4.

t_1
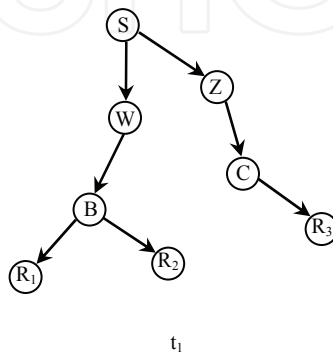
Fig. 4. Demonstration of sequential MDC algorithm

5. Performance evaluation

This section deals with the performance evaluation of our developed algorithms. In particular, we evaluate the performance of our proposed algorithms, namely, Serial MDC, Distributed MDC, Centralized MDC, and Sequential MDC algorithms and compare them with the algorithm proposed by Chen et al. in (Chen et al., 2004). Chen et al. proposed this algorithm for assigning a number of video layers that are encoded using LC technique; we referred to as Chen-LC. To make a fair comparison, we modified Chen-LC algorithm to construct node disjoint multicast trees. Moreover, in order to take the bandwidth requirements for MDC and LC into consideration, we consider the video sequences reported in (Gogate et al., 2002). Since all the video sequences have roughly the same bit rate, we consider the video sequence of "Football". The average video source rate is 1.5 Mbps for each description, whereas the average video source rate for the layered coder is 1.57 Mbps for the base layer and 1.45 Mbps for the enhancement layer.
We generate a wireless ad hoc network by placing a number of nodes at random locations in a square area of $1000 \times 1000 \ m^2$. The radio transmission range is 250 m and the number of video descriptions required by each destination is uniformly distributed to be $\in \{1,2\}$. The residual bandwidth of each link is randomly chosen from $[2,10]$ Mbps. The delay in each link is randomly chosen from $[1,20]$ ms. Moreover, the multicast source S and a set of destinations $\mathcal{Y}$ are randomly chosen from the network graph to form a multicast session. Any

---

Algorithm 9 Sequential algorithm

1: **for** $i = 1$ *to L* **do**
2:     $t_L \subseteq t_{L-1} \cdots \subseteq t_2 \subseteq t_1$
3: **end for**

---

destination node is at least 2-hop away from the multicast source. For each simulation, several experiments have been run to ensure 95% confidence interval. The 95% confidence intervals are always plotted, when they are not visible it means that they are smaller than the curve markers.

To show the significance of our developed algorithms, we evaluate and compare its performance with the well-known multicast algorithm, Chen-LC, using the following metrics:

– **User satisfaction:** This metric is defined as the total number of the assigned video descriptions to all destinations divided by the total number of requested video descriptions by all destinations. This metric presents the effectiveness of a protocol.

$$User\ satisfaction = \frac{\sum_{i=1}^{n} \mathcal{N}_{asg}(R_i)}{\sum_{i=1}^{n} \mathcal{N}_{req}(R_i)} \qquad (7)$$

where $\mathcal{N}_{asg}(R_i)$, and $\mathcal{N}_{req}(R_i)$, are the number of the assigned and requested video descriptions of the destination R_i respectively, and n is the number of destinations.

– **Number of pure forwarders (PF):** It is defined as the number of pure forwarders nodes on the aggregate multicast tree T that are not destinations. This measures the efficiency in terms of minimizing the number of pure forwarding nodes.

$$PF = \sum_{i=1}^{\mathcal{V}} I(v_i) \qquad (8)$$

where $\mathcal{V}$ is the network size and $I(v_i)$ is defined as:

$$I(v_i) = \begin{cases} 1 & \text{for } v_i \in T - \{S, \mathcal{Y}\} \\ 0 & \text{otherwise} \end{cases}$$

– **Bandwidth utilization:** This metrics defined as the total used bandwidth for the video distribution tree(s).

$$Bandwidth\ utilization = \sum_{e \in u} Bw\,(e) \qquad (9)$$

where e denotes a link, u is the set of used links, and $Bw\,(e)$ denotes the bandwidth devoted to video distribution in link e.

– **Aggregate tree delay:** It represents the longest delay from the multicast source s to a destination node R_i on the aggregate tree T, as seen in Equ.(4).

5.1 Varying number of multicast destinations

Fig. 5 to Fig. 8 illustrate Serial MDC, Distributed MDC, Centralized MDC, Sequential MDC and Chen-LC algorithms performance with varying number of multicast destination nodes while the network size is set to 50 nodes. Fig. 5 shows that Serial MDC, Distributed MDC, and Centralized MDC algorithms achieve higher user satisfaction compared to Chen-LC algorithm. This can be related to the independent-description property of MDC. Sequential MDC and Chen-LC algorithms have the same user satisfaction. This is because Sequential MDC algorithm does not employ the independent-property of MDC. In this case it is similar to Chen-LC. In other words, VD_1 is equivalent to the basic layer and VD_2 is equivalent to the enhancement layer. Serial MDC, Distributed MDC, and Centralized MDC algorithms are

well scalable in term of number of destinations. Centralized MDC algorithm achieves a higher user satisfaction compared to Serial MDC and Distributed MDC algorithms. As the number of destinations increases, user satisfaction decreases gradually in Serial MDC, Distributed MDC, and Centralized MDC algorithms. However, the user satisfaction of Chen-LC and Sequential algorithms decreases sharply as the number of destinations increases. That is, as a result of the dependent-layer property of LC (for Chen-LC algorithm) and because Sequential MDC algorithm does not employ the independent-description property of MDC.

Fig. 6 depicts the number of pure forwarders nodes as a function of number of destination nodes. It can be seen that Centralized MDC has slightly higher number of pure forwarders nodes compared to the other algorithms. However, Distributed MDC has a lowest number of pure forwarders nodes. In Fig. 7, we plot the average bandwidth utilization. Clearly, the bandwidth utilization of Centralized MDC is slightly higher than the bandwidth utilization of the other algorithms. This is because Centralized MDC requires more number of pure forwarders, compared to the other algorithms; to constructs multiple node-disjoint trees (see Fig. 6). However, Distributed MDC requires a minimum bandwidth for video distribution trees. This is because Distributed MDC has a minimum number of pure forwarders nodes. We show in Fig. 8 the aggregate tree delay as a function of number of destinations. All algorithms achieve a comparable delay as compared to each other. As the number of destinations increases the aggregate tree delay increases. This is because more paths are constructed to build multiple multicast tree.
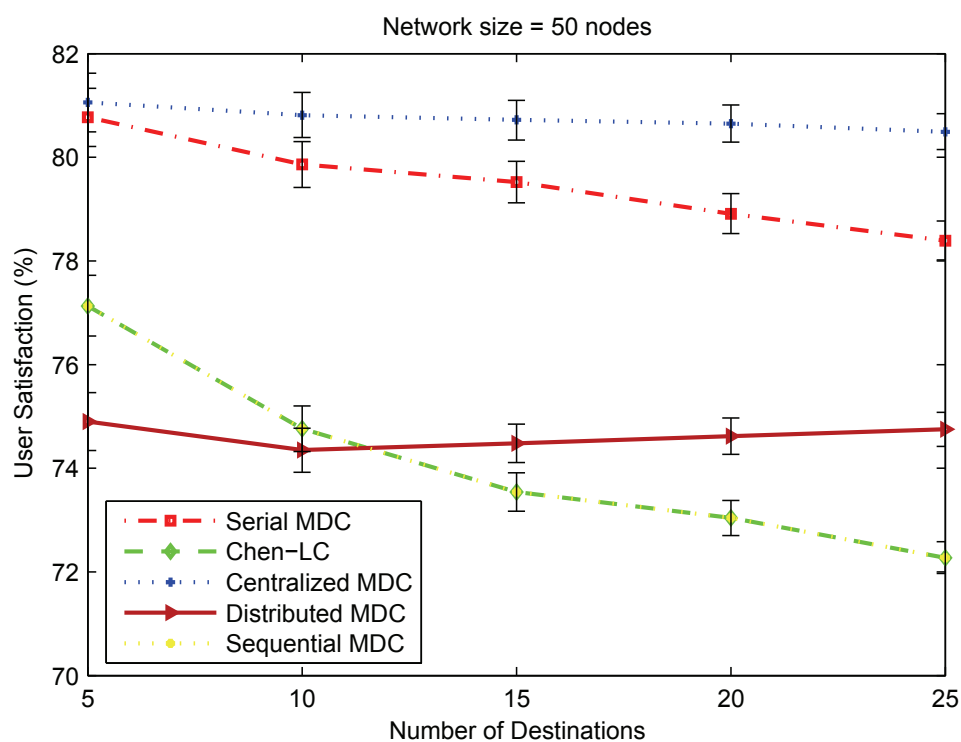
Fig. 5. User satisfaction versus number of destinations. Network size = 50 nodes

Fig. 9 to Fig.12 illustrate Serial MDC, Distributed MDC, Centralized MDC, Sequential MDC, and Chen-LC performance with varying number of multicast destination nodes while the network size is set to 100 nodes. As the network size increase from 50 nodes (Fig. 5) to 100 nodes (Fig. 9), the users satisfaction for all algorithms increases. This is because the number of nodes in the network increases. As a result, the number of paths to each destination is
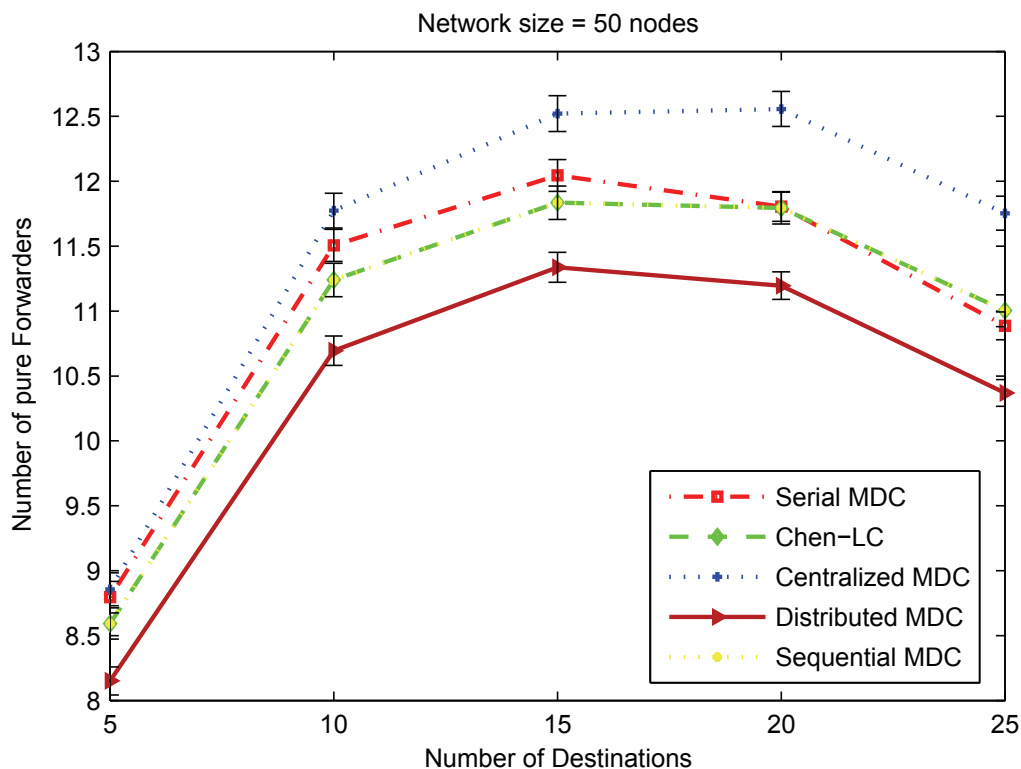
Fig. 6. Number of pure forwarders versus number of destinations. Network size = 50 nodes

increased. Again, Serial MDC, Distributed MDC, and Centralized MDC show good scalability as the number of destinations increases. As the number of nodes in the network increases from 50 to 100 nodes, the number of pure forwarders nodes, the bandwidth utilization, and the aggregate tree delay (in Fig. 10, Fig. 11, and Fig. 12) are increased compared to Fig. 6, Fig. 7, and Fig. 8, respectively.

5.2 Varying network size

Fig. 13 to Fig. 20 compare Serial MDC, Distributed MDC, Centralized MDC, Sequential MDC, and Chen-LC performance, with varying number of nodes in the network (network size) from 50 to 100 nodes, in terms of user satisfaction, number of pure forwarders, bandwidth utilization, and aggregate tree delay. The number of destinations is set to 10 and 30 nodes. Centralized MDC achieve a higher user satisfaction (see Fig. 13 and Fig. 17) compared to the other algorithms. The cost of that is the increase in the number of pure forwarders nodes (see Fig. 14 and Fig. 18), the bandwidth utilization (see Fig. 15 and Fig. 19), and the aggregate tree delay (Fig. 16 and Fig. 20). However this cost is still comparable. As the network size increases, the user satisfaction for all algorithms increases. We related that to the increase in the number of resources in the network, i.e., number of nodes, and bandwidth. Fig. 16 and Fig. 20 show that as the network size increases the aggregate tree delay decreases. This is because more alternate paths (with minimum delay) may exist.

Comparing Fig. 13 with Fig. 17, we can note that the user satisfaction, for all algorithms, decreases as the number of destinations increases from 10 to 30 nodes. This because the number of node-disjoint paths to the destination nodes decreases. Thus, the number of assigned MD video to each destination decreases.
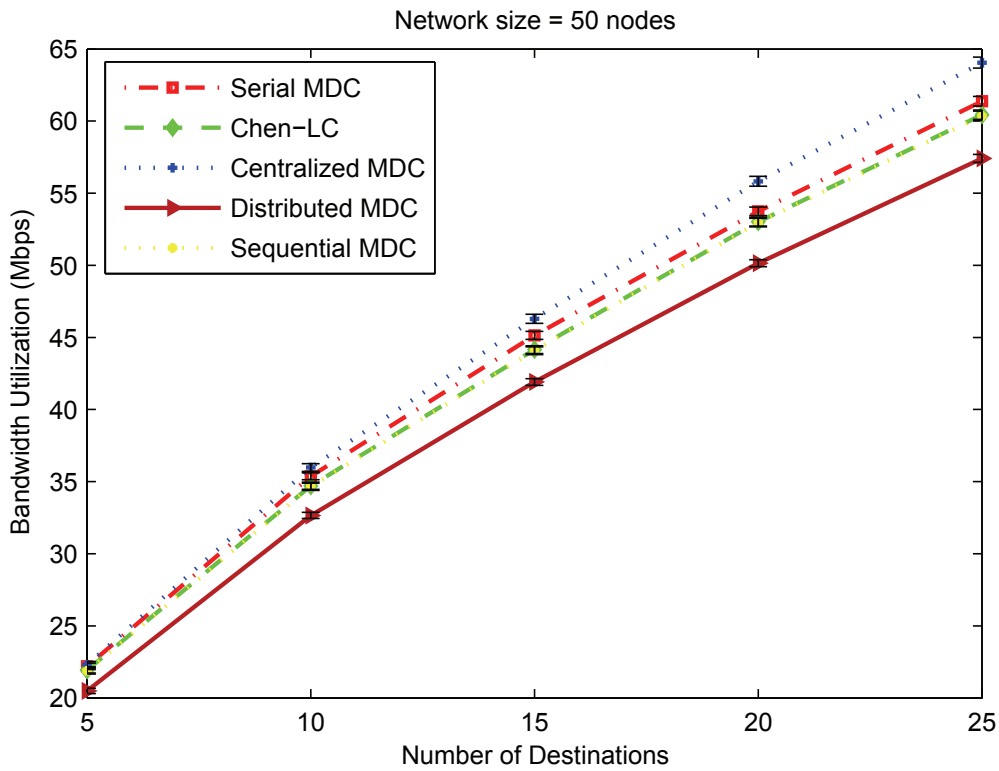
Fig. 7. Bandwidth utilization versus number of destinations. Network size = 50 nodes
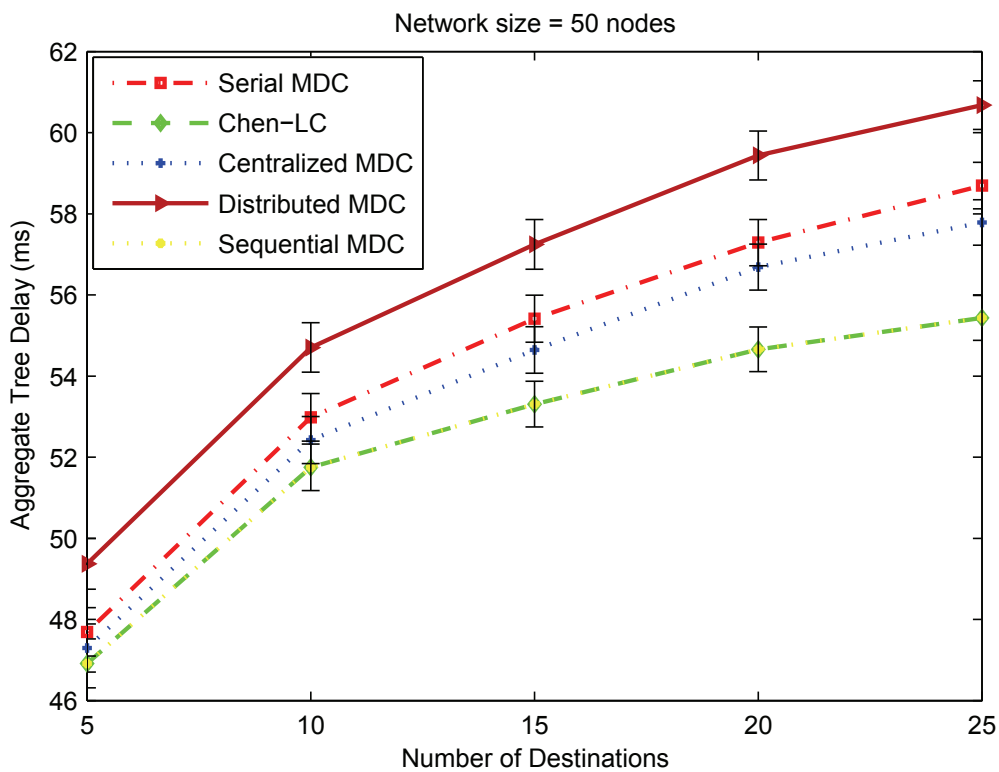
Fig. 8. Aggregate tree delay versus number of destinations. Network size = 50 nodes
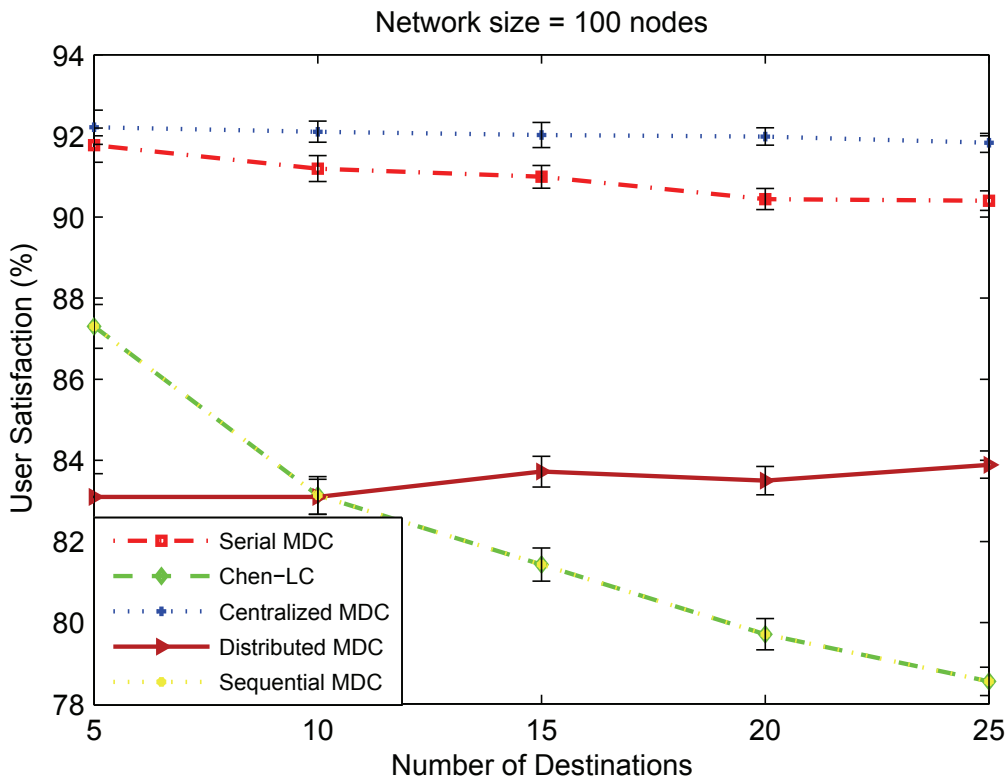
Fig. 9. User satisfaction versus number of destinations. Network size = 100 nodes
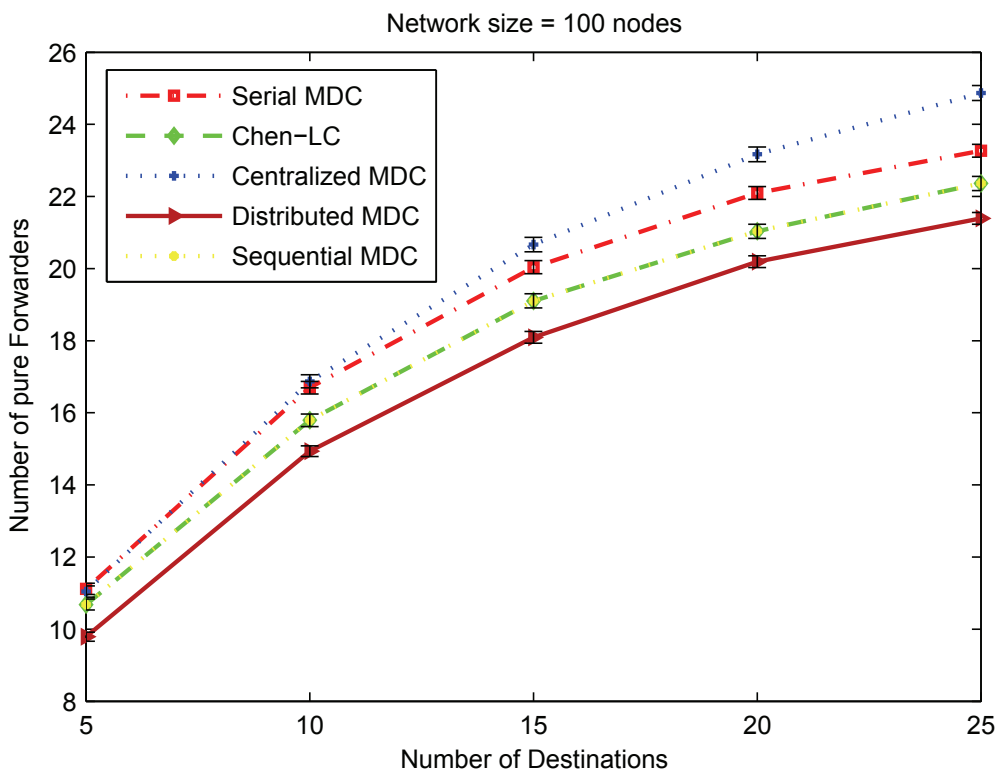
Fig. 10. Number of pure forwarders versus number of destinations. Network size = 100 nodes
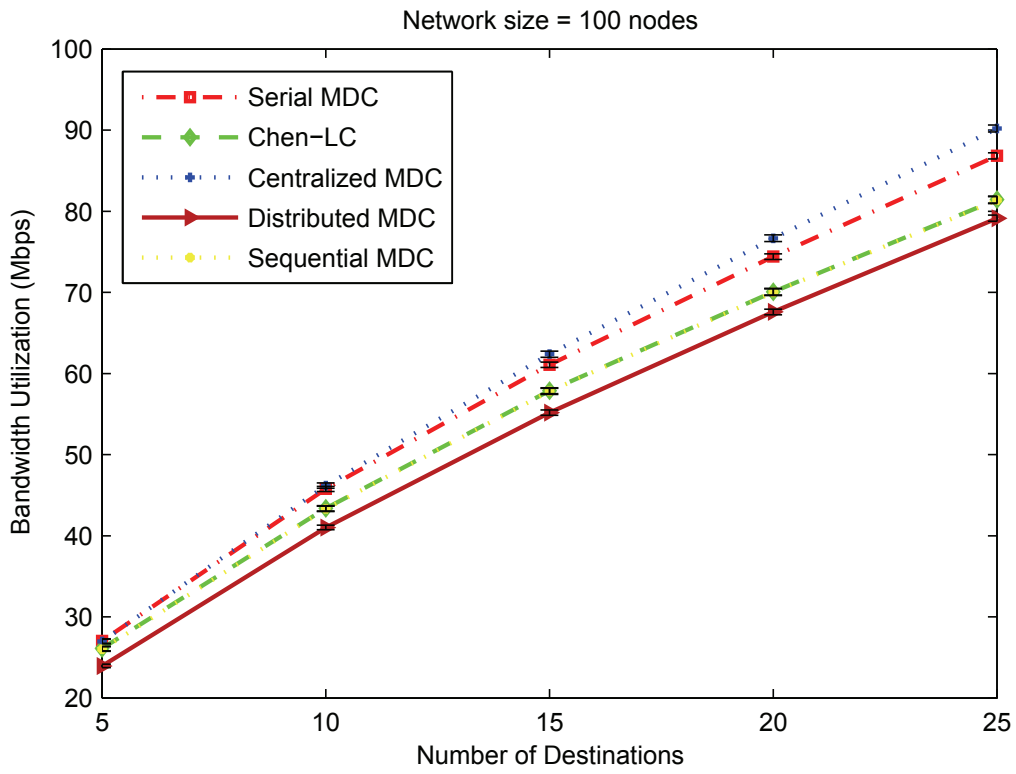
Fig. 11. Bandwidth utilization versus number of destinations. Network size = 100 nodes
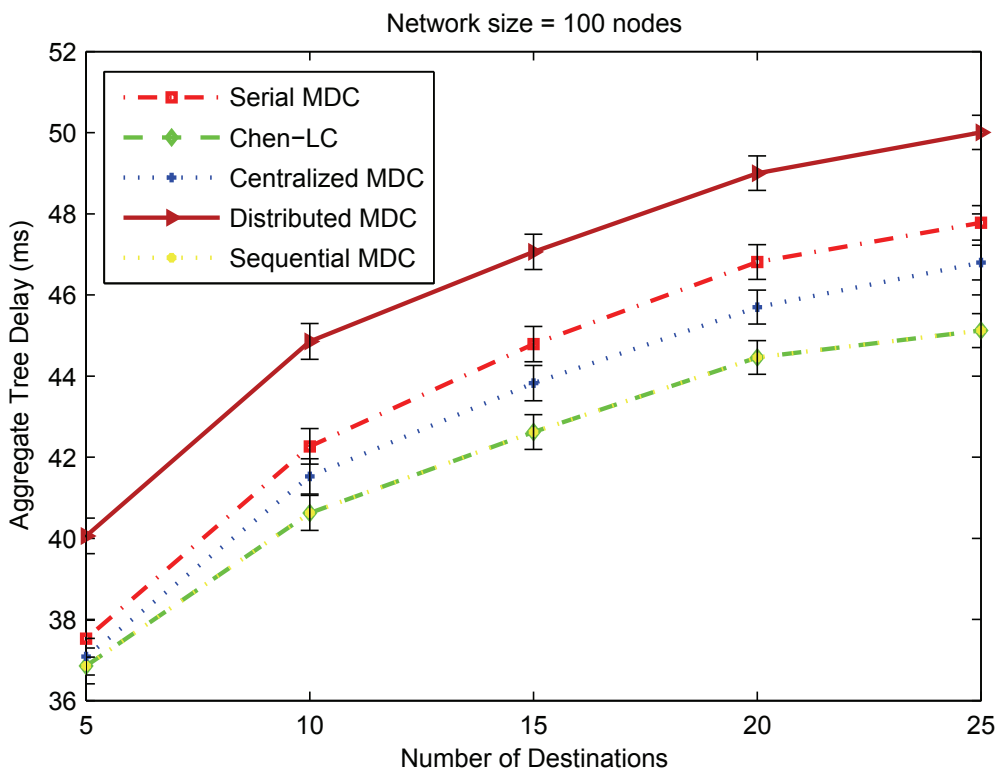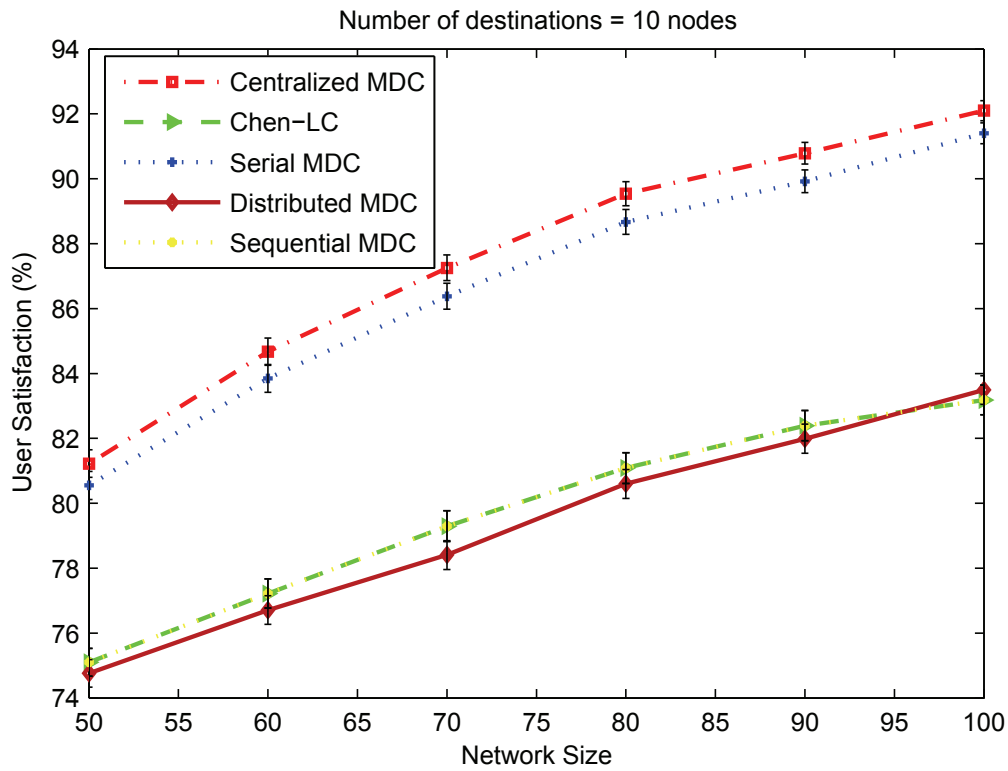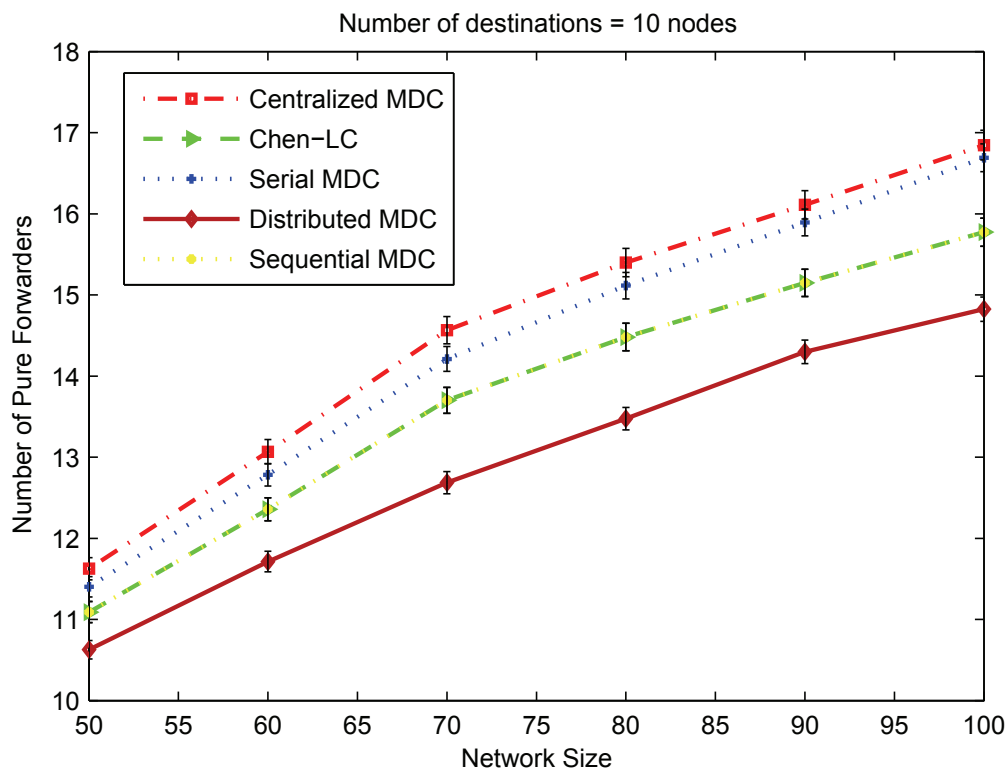

Fig. 12. Aggregate tree delay versus number of destinations. Network size = 100 nodes

Fig. 13. User satisfaction versus network size. Number of destinations = 10 nodes

Fig. 14. Number of pure forwarders versus network size. Number of destinations = 10 nodes
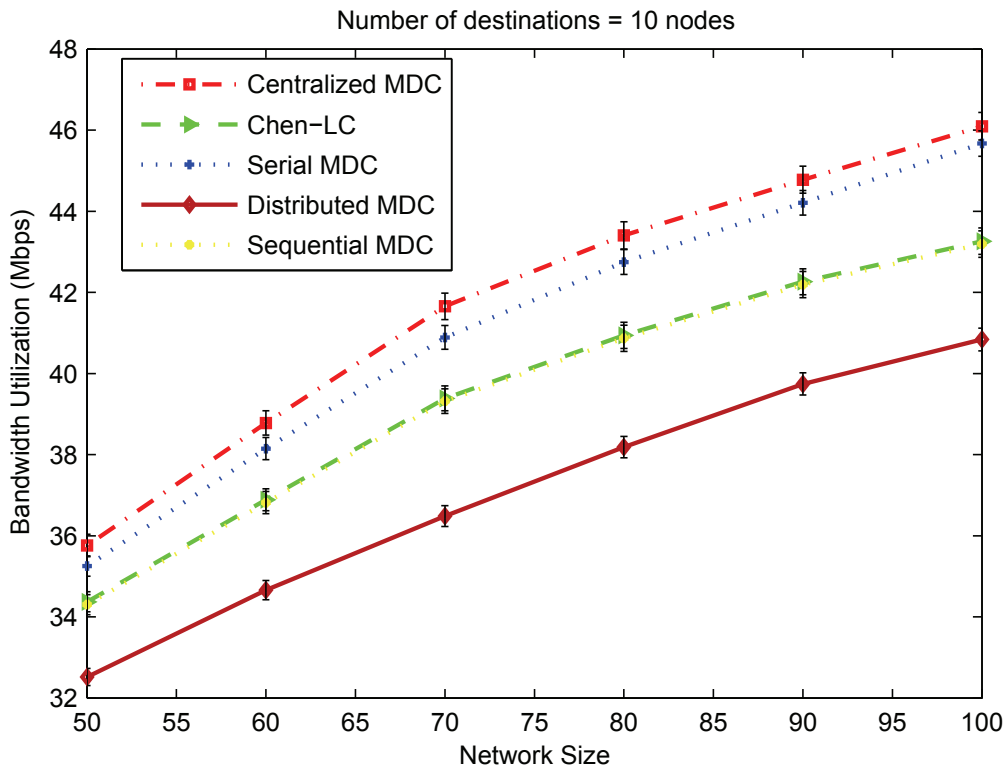
Fig. 15. Bandwidth utilization versus network size. Number of destinations = 10 nodes
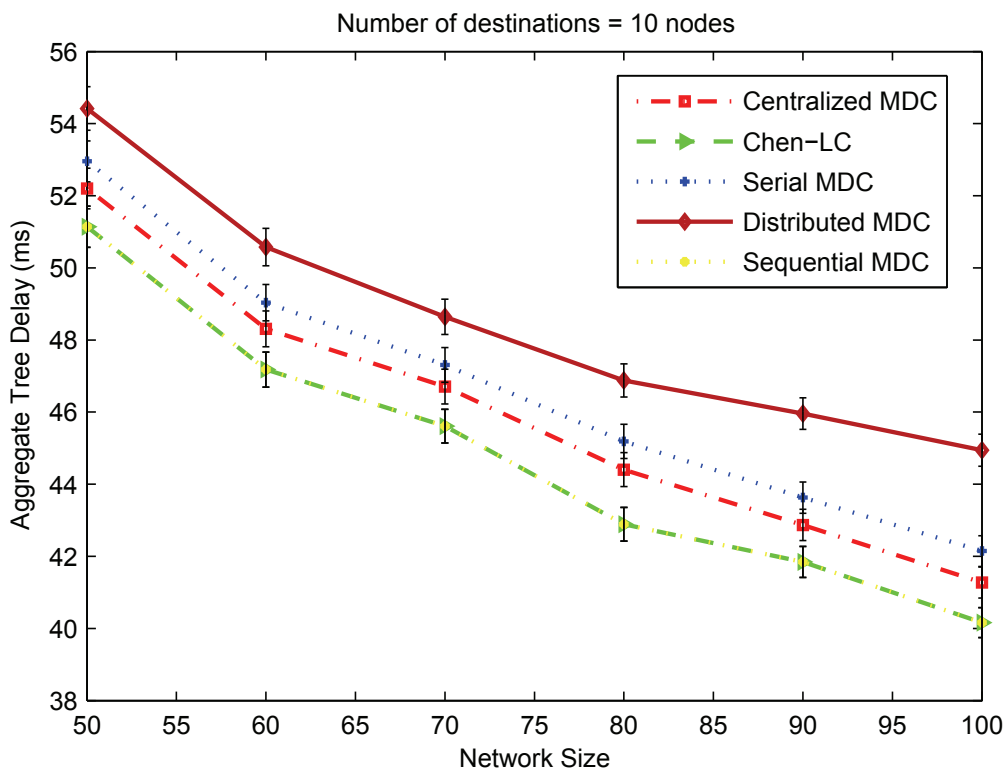
Fig. 16. Aggregate tree delay versus network size. Number of destinations = 10 nodes
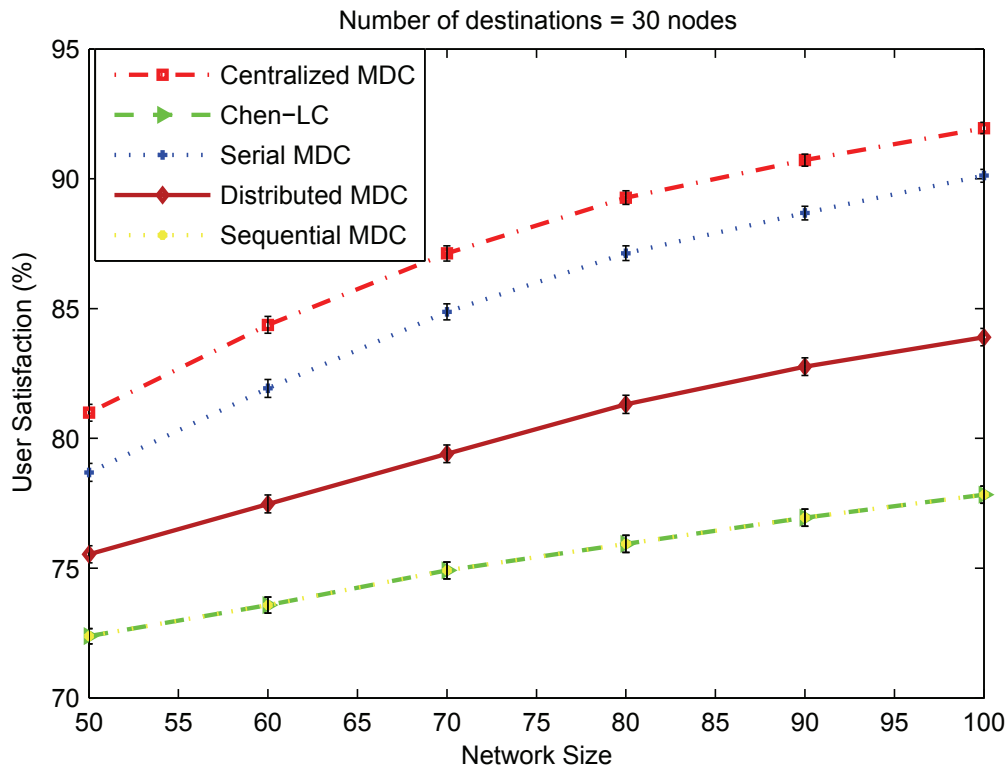
Fig. 17. User satisfaction versus network size. Number of destinations = 10 nodes
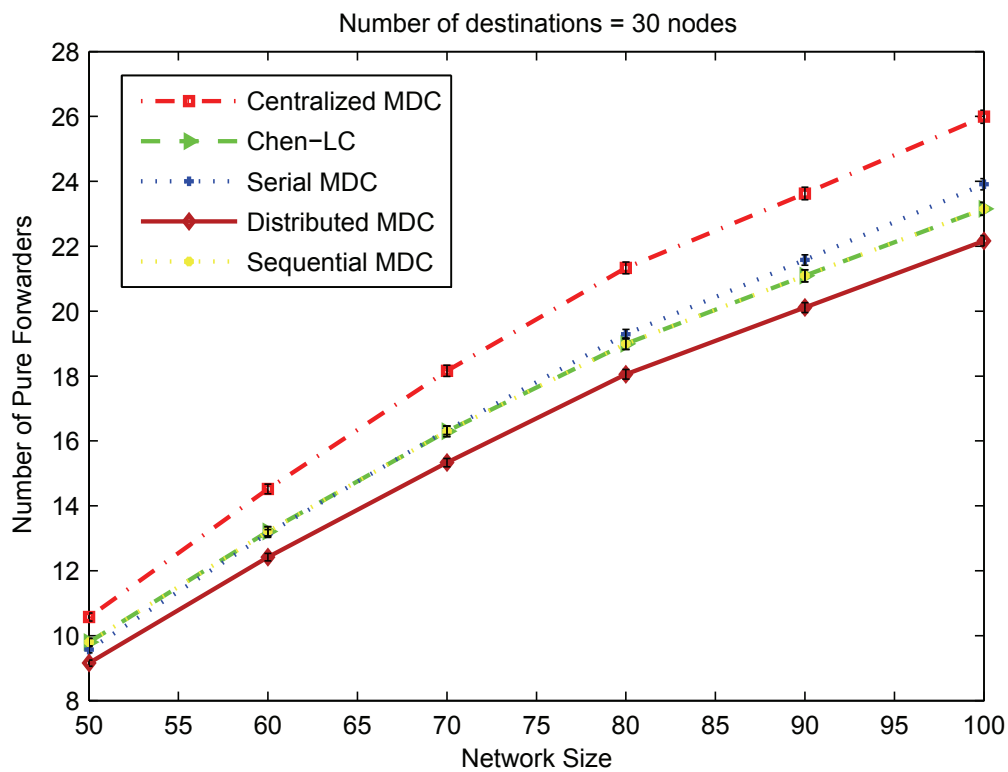
Fig. 18. Number of pure forwarders versus network size. Number of destinations = 10 nodes
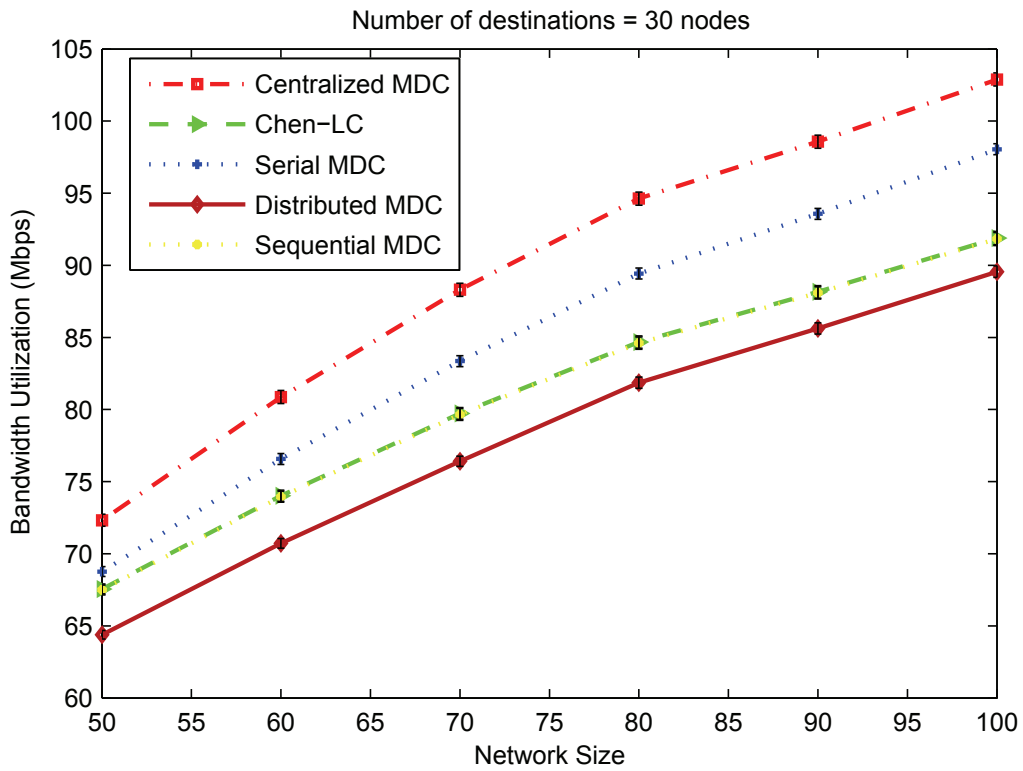
Fig. 19. Bandwidth utilization versus network size. Number of destinations = 10 nodes
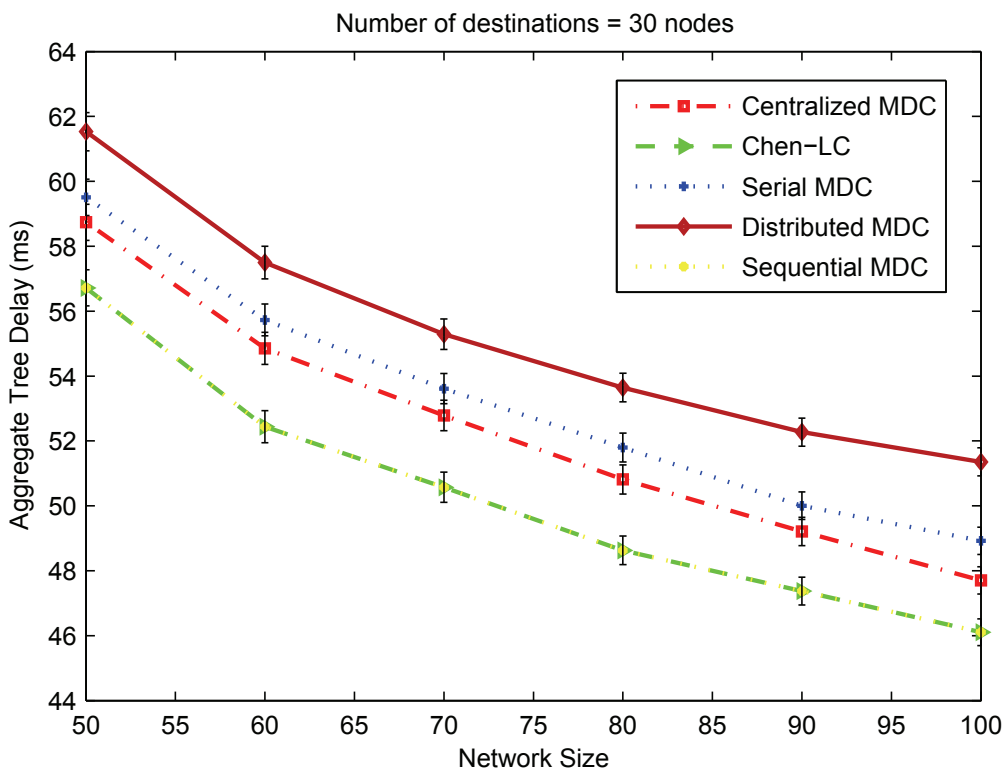
Fig. 20. Aggregate tree delay versus network size. Number of destinations = 10 nodes

6. Complexity analysis of the algorithms

We analyze the complexity of our proposed algorithms as follows. For Serial MDC, the shortest path algorithm (Dijkstra's algorithm) is of complexity $O(|\mathcal{V}|\log(|\mathcal{V}|) + |\mathcal{E}|) \leq O(\mathcal{V}^2)$ where $|\mathcal{V}|$ and $|\mathcal{E}|$ are the number of nodes and number of wireless communication links in the partial topology, respectively. Since it iterates $\mathcal{Y}$ times, where $\mathcal{Y}$ is the number of destination nodes. Therefore the complexity is $O(\mathcal{V}^2 \times \mathcal{Y})$ and finally the algorithm iterates $|\mathcal{N}_{req}(VD)|$ times, where $|\mathcal{N}_{req}(VD)|$ is the total number of required video descriptions for all destinations. As a result, the complexity of Serial MDC is given by $O(\mathcal{V}^2 \times \mathcal{Y} \times \mathcal{N}_{req}(VD))$. For Distributed MDC the complexity is given by $O(\mathcal{V})$. Finally, Centralized MDC, Sequential MDC, and Chen-LC algorithms have the same complexity of Serial MDC algorithm.

7. Conclusion

In this chapter we study the problem of multiple multicast trees construction and the assignment of MD video. Different algorithms are proposed for that purpose. These algorithms are: Serial MDC, Distributed MDC, Centralized MDC, and Sequential algorithms. Serial MDC, Distributed MDC, and Centralized MDC algorithms deploy the independent-description property of MDC, whereas Sequential MDC algorithm does not take this property into consideration. Simulation results demonstrate that deploying this property of MDC along with multiple multicast tree can greatly improve the user satisfaction. Furthermore, simulation results demonstrate that the way of multiple multicast tree construction and the assignment of MD video can affect the user satisfaction. In addition, simulation results show that MDC can achieve higher user satisfaction compared to Layered Coding (LC) with a small cost in terms of number of pure forwarders nodes, bandwidth utilization, and aggregate tree delay.

8. References

Agrawal, D., Reddy, T. B. & Murthy, C. S. R. (2006). Robust demand-driven video multicast over ad hoc wireless networks, *3rd International Conference on Broadband Communications, Networks and Systems* pp. 1–10.

Badarneh, O., Kadoch, M. & Elhakeem, A. (2008). A new approach for the construction of multiple multicast trees using multiple description video for wireless ad hoc networks, *IEEE Conference on Local Computer Networks* pp. 152–159.

Chen, J., Chan, S.-H. G. & Li, V. O. (2004). Multipath routing for video delivery over bandwidth-limited networks, *IEEE Journal on Selected Areas in Communications* **22**(10): 1920–1932.

Chow, C.-O. & Ishii, H. (2008). Multiple tree multicast ad hoc on-demand distance vector (mt-maodv) routing protocol for video multicast over mobile ad hoc networks, *IEICE-Transactions on Communications* **E91-B**: 428–436.

Clausen, T. & Jacquet, P. (year 2003). Optimized link state routing protocol (olsr). www.ietf.org/rfc/rfc3626.txt.

Gogate, N., Chung, D.-M., Panwar, S. & Wang, Y. (2002). Supporting image and video applications in a multihop radio environment using path diversity and multiple description coding, *IEEE transactions on circuits and systems for video technology* **12**(9): 777–792.

Goyal, V. K. (2001). Multiple description coding: Compression meets the network, *IEEE Signal Processing Magazine* **18**: 74–94.

Mao, S., Cheng, X., Hou, Y. T. & Sherali, H. D. (2006). Multiple description video multicast in wireless ad hoc networks, *Mobile Networks and Applications* **11**: 63–73.

Mao, S., Hou, Y., Cheng, X., Sherali, H., Midkiff, S. & Zhang, Y.-Q. (2006). On routing for multiple description video over wireless ad hoc networks, *IEEE Transactions on Multimedia* **8**(5): 1063–1074.

Mao, S., Lin, S., Panwar, S., Wang, Y. & Celebi, E. (2003). Video transport over ad hoc networks: multistream coding with multipath transport, *IEEE Journal on Selected Areas in Communications* **21**(10): 1721–1737.

Puri, R. & Ramchandran, K. (1999). Multiple description source coding through forward error correction codes, *IEEE Proceedings Asilomar Conference on Signals, Systems, and Computers* pp. 342–346.

Wei, W. & Zakhor, A. (2007). Multiple tree video multicast over wireless ad hoc networks, *IEEE Transactions on Circuits and Systems for Video Technology* **17**: 2–15.
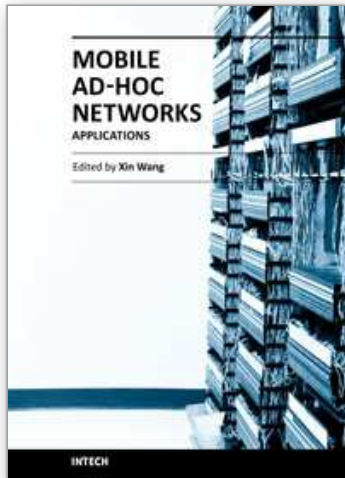
Mobile Ad-Hoc Networks: Applications

Edited by Prof. Xin Wang

Being infrastructure-less and without central administration control, wireless ad-hoc networking is playing a more and more important role in extending the coverage of traditional wireless infrastructure (cellular networks, wireless LAN, etc). This book includes state-of the-art techniques and solutions for wireless ad-hoc networks. It focuses on the following topics in ad-hoc networks: vehicular ad-hoc networks, security and caching, TCP in ad-hoc networks and emerging applications. It is targeted to provide network engineers and researchers with design guidelines for large scale wireless ad hoc networks.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

INTECH
open science | open minds