Journal of Network and Computer Applications 35 (2012) 20–28

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

An overlay multicast protocol for live streaming and delay-guaranteed interactive media

Weizhan Zhang [a], Qinghua Zheng [a], Haifei Li [b], Feng Tian [a,*]

[a] MOE KLINNS Lab and SKLMS Lab, School of Electronics and Information Engineering, Xi'an Jiaotong University, 710049 Xi'an, China
[b] Department of Computer Science, Union University Jackson, TN 38305, USA

ARTICLE INFO

Article history:
Received 31 August 2010
Received in revised form
9 January 2011
Accepted 24 February 2011
Available online 3 March 2011

Keywords:
Overlay multicast
Layered degree constraint
Interactive media
Live streaming
Collaborative learning

ABSTRACT

In many collaborative multimedia applications, there is often a requirement for simultaneously supporting live streaming and shareable interaction. A major challenge in designing such an application by overlay multicast is how to simultaneously provide scalable live streaming and delay-guaranteed interactive media. Live streaming by overlay multicast incurs additional application-layer latency, which conflicts with the delay-sensitive property of interactive media. To handle this dilemma, in this paper, we propose a layered degree-constrained overlay multicast protocol, which organizes the overlay multicast tree as a layered degree-constrained core tree and an extended tree. The core tree maintains available resources in its top layers for subsequent two-way interaction, whereas the extended tree expands the core tree for one-way live streaming. Our simulation and experimental results show that the proposed overlay multicast protocol can simultaneously provide delay-guaranteed interactive media as well as scalable live streaming.

1. Introduction

In many group-oriented media streaming applications, overlay multicast is utilized as an alternative technology for IP multicast, which suffers from various issues such as group management, congestion and flow control, and security (Diot et al., 2000). Using overlay multicast for media streaming over Internet has been intensively studied. Many approaches have been designed and implemented (Hosseini et al., 2007). The overlay multicast can be classified into two broad categories: peer-to-peer (P2P) overlay multicast and infrastructure-based overlay multicast (Li and Yin, 2007). When compared with the P2P solution by end users, the infrastructure-based overlay multicast by proxies is more costly, but is more reliable and easier to control. Therefore, the infrastructure-based overlay multicast can complement the popular P2P streaming in many fields, providing more robust media streaming for many commercial services (Akamai, 2010). In this paper, we focus on utilizing the advantages of the infrastructure-based overlay multicast for quality of service (QoS)-guaranteed collaborative multimedia applications. The similar choices are widely adopted by the researches for the targeted applications. For example, both of the researches adopt the infrastructure scheme for the synchronous e-learning platforms (Maraviglia et al., 2008; Granda et al., 2009).

According to the delivery mode, the legacy overlay multicast researches can also be categorized into one-way live/pre-recorded streaming in large scale and two-way interaction in small group. However, there are many multimedia applications which need to simultaneously provide two-way interactive media as well as one-way media streaming, where the interactive media refers to the bidirectional real-time video or audio interaction in this paper. Examples for such applications are distance learning/training, online auction, online gaming, and other collaborative applications. Furthermore, in many of these applications, the interaction scenario between the presenter and a fraction of participants needs to be shared by all the other participants. We call it sharable interaction in this paper. In particular, the motivation for this research comes from integrating sharable interaction in a collaborative live teaching system (the open source version Bluesky is under incubation (Bluesky, 2010)). Besides the live broadcasting of teaching scenes, interactions between the tutor and the students are enabled, and what is more, these interaction scenarios can be shared by all the participants in the live teaching class, which imitates a tradition-like face-to-face classroom. Therefore, in this paper, we aims at building a new overlay multicast protocol that can simultaneously provide the sharable interaction as well as the scalable live streaming.

Since both the live streaming and the shared interactivities need to be delivered to all participants by overlay multicast, there is a requirement for simultaneously satisfying scalable one-way live streaming and delay-sensitive interaction in a shared overlay multicast structure. A scalable live streaming solution incurs a

* Corresponding author. Tel.: +86 13474672214.
E-mail addresses: ftian@sei.xjtu.edu.cn, phdfengtian@yahoo.com.cn (F. Tian).

1084-8045/$ - see front matter
doi:10.1016/j.jnca.2011.02.013

deep tree structure. The additional application-layer processing delay and access delay crossing an overlay node are aggregated when a media stream is forwarded along the deep tree path. This redundant delay is acceptable to the applications like one-way pre-recorded streaming or one-way live streaming, but it may not be tolerable for an interactive media, which usually has stringent latency requirement. Therefore, in this paper, we propose a layered degree-constrained overlay multicast (LDCOM) protocol to handle the confliction between the dynamic interaction requirement and the scalable live streaming requirement. LDCOM seamlessly integrates a two-tier tree construction, including a core tree and an extended tree. The core tree identifies the original interaction-enabled nodes by a delay threshold, and reserves resources at its top layers in a layered manner. The extended tree meets the needs of the scalability, extending the core tree without the delay constraint. For the interactive nodes out of the core tree scope, LDCOM adopts a dynamic interaction adjustment. The interactive nodes in extended tree are adjusted to the reserved positions in the core tree so as to achieve guaranteed delay performance. Meanwhile, the interaction scenarios are shared by all the participants by the overlay multicast tree. In this way, LDCOM supports the two types of the media simultaneously from the view of tree construction, with a guaranteed maximum delay for the shareable interaction.

The remainder of this paper is organized as follows. Section 2 summarizes related work. Section 3 presents the proposed LDCOM algorithm. In Section 4, we devises the dynamic optimization of the algorithm. We then evaluate the performance of our proposal through simulation and experimentation in Sections 5 and 6, respectively. Finally, Section 7 concludes the paper and discusses potential future works.

2. Related work

There has been a great amount of work in the area of overlay multicasts. The P2P overlay multicast provides flexible and scalable solution, utilizing client resources to forward the media data (Wang et al., 2008; Lua et al., 2008; Do et al., 2008; Guo et al., 2008b,a; Vik et al., 2008; Cai and Zhou, 2006; Magharei et al., 2005; Ren et al., 2008; Zhang et al., 2005; Xu and Li, 2009). However, it suffers from the problem of peer churn and complicates the system management. By contrast, the infrastructure-based overlay multicast deploys dedicated proxies to the edge of the Internet by the service provider, and constructs a collaborative service overlay network (SON) (Yang et al., 2007; Guo and Jha, 2007; Capone et al., 2009; Lao et al., 2007; Pompili et al., 2008). As mentioned earlier, in this paper, we focus our work on an infrastructure-based overlay multicast. Therefore, we will move our concentration on the media delivery mode of overlay multicast in the rest of this section.

For one-way media streaming, many P2P overlay multicasts and infrastructure-based overlay multicasts have been designed and implemented. The existing P2P overlay multicast approaches for one-way media streaming can be categorized as either tree-based P2P multicasts (Wang et al., 2008; Lua et al., 2008; Do et al., 2008; Guo et al., 2008b,a; Vik et al., 2008; Cai and Zhou, 2006) or mesh-based P2P overlays (Magharei et al., 2005; Ren et al., 2008; Zhang et al., 2005; Xu and Li, 2009). The tree-based P2P approach organizes peers into an explicit data delivery tree. Some P2P approaches also utilize the multi-tree structure for issues like heterogeneity and redundancy (Castro et al., 2003; Li et al., 2009; Venkataraman et al., 2006; Dan et al., 2007). By contrast, the mesh-based P2P approach maintains a set of partners for each peer, and exchanges the data availability information without an explicit structure support. Generally speaking, mesh-based P2P overlays outperform tree-based P2P approaches in many ways, such as reliability and maintenance overhead (Seibert et al., 2008; Magharei et al., 2007). On the other hand, due to the relative stability of proxy nodes, in infrastructure-based overlay multicasts, explicit tree-based structures are widely utilized (Yang et al., 2007; Guo and Jha, 2007; Capone et al., 2009; Lao et al., 2007; Pompili et al., 2008). Therefore, in this paper, we focus on using tree-based structure under the context of infrastructure-based overlay multicast. However, in the above solutions for one-way live streaming, two-way interactive media delivery is not of concern.

For interactive media streaming, the researches in overlay multicast mainly focus on small group video conference, collaborative simulation, and online games, etc. (Luo et al., 2004; Gu et al., 2005; Chu et al., 2002; Liu and Zimmermann, 2005). These schemes organize the data delivery routes as source-specific trees, and support multicast groups with many to many scenarios. DigiMetro (Luo et al., 2004) is an overlay multicast system tailored to small and impromptu video conference. It organizes the data delivery routes as source-specific trees, which are constructed by a local greedy algorithm. Peertalk (Gu et al., 2005) provides resource-efficient and failure-resilient multi-party voice-over-IP services over overlay multicast. Vik et al. (2008) focuses on reducing the multicast tree diameter for dynamic interactive applications. The research deals with many spanning tree problems. Some spanning tree structures are elected with small diameter attribute for dynamic interactive applications. Active (Liu and Zimmermann, 2005) aims to the solution for the interaction problem in large-scale P2P streaming. The research based on the observation that only a fraction of the users are active at a given time in a large interactive group. It employs a tree optimization algorithm after tree construction to gradually decrease the delay among the interactive nodes. However, reconstructing the multicast tree architecture might be invalid for dynamic interaction adjustments. Further, it does not provide certain delay assurance for the interactions.

In brief, most research works about overlay multicast protocols are designed for one-way media streaming. For interactive media applications, solutions are mainly designed for small groups, or a small active part in a larger group. These approaches provide best-effort strategies to obtain minimal path or pair-wise latency. However, the delay assurance for interactive media is not provided. Furthermore, the requirements for the shareable interaction in many collaborative virtue reality applications are not mentioned, and the conflict between the shareable interaction and the scalable live streaming is not addressed by the existing approaches in literature.

3. LDCOM: a layered degree-constrained overlay multicast protocol

3.1. Overview of the LDCOM

LDCOM adopts a two-tier architecture. A simplified example of LDCOM architecture is shown in Fig. 1. The scope of the core tree is constrained by a delay tolerance threshold. A node belonging to the core tree indicates that its path delay performance is within the tolerance. Otherwise, the node is included in the extended tree. Specially, LDCOM adopts a layered out-degree distribution. The tree construction complies with an additional set of layered degree constraint (LDC) values. It means that the out degree of the overlay nodes in the same layer is constrained by a shared LDC value. In this way, the resources of upper layers in a tree are reserved. Since the upper layer nodes suffer less from the application-layer latency, the reserved resources can be utilized for interactive communications in later occasions. The reservation of resources at top layers of multicast tree may increase the delay of one-way live media, because the tree may have more layers in

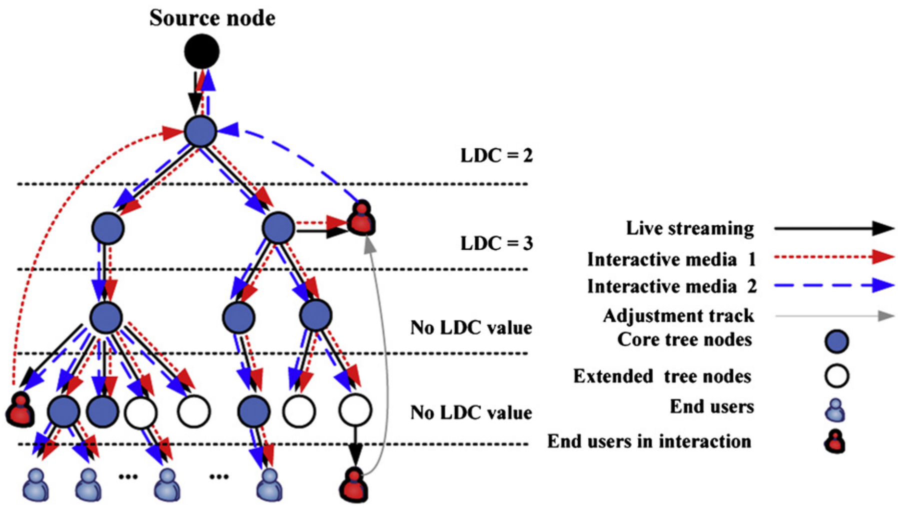

Fig. 1. An example of LDCOM architecture.

order to accommodate all participants. Therefore, only necessary reservation is enforced in the top layers of the core tree. Notice that this redundant delay cost is acceptable because it is invisible to most one-way live media applications. As shown in Fig. 1, the given example only utilizes the top two layers in the core tree for reservation. For the scenario of shareable interactions, if the interactive nodes are belong to the extend tree, LDCOM dynamically adjusts the interactive nodes to the reserved positions in the core tree so that they can achieve guaranteed delay performance from the source. Meanwhile, the interaction scenarios are delivered to all the participants by overlay multicast at the same time. The data flow for LDCOM is also presented in Fig. 1.

We now describe the implementation of the protocol briefly in five steps.

Step 1. *Delay metric selection*: LDCOM is designed to support shareable interactions between the sponsor and the participants, as well as large-scale live streaming. Therefore, the path delay between the source and the participants is considered as the metric for tree construction.

Step 2. *Layer degree constraint allocation*: Before the tree construction, the protocol allocates a set of LDC values for all of the layers. The number of the layers needed for reservation is calculated in advance by considering the delay tolerance threshold and the average delay of an overlay hop as well as the average access delay of an end user. The allocation of layer degree constraints aims to maximize the profitability of resource reservation, utilizing an integer programming method.

Step 3. *Tree construction*: Based on the preparation of the previous steps, LDCOM builds a two-tier tree to handle the overlay nodes joining and leaving. The newly joined nodes are attached either to the core tree or to the extended tree.

Step 4. *Join of end users*: Based on the established overlay multicast tree, the newly joined participants are attached to the tree in a clustered manner, which makes the user congruent to the network topology.

Step 5. *Dynamic interaction adjustment*: In the final step, the protocol implements a dynamic interaction adjustment to the scene of interaction. The interactive nodes in extended tree are adjusted to the reserved positions in upper layers of the core tree with a top-down manner.

3.2. Delay metric selection

The overlay network can be modeled as a complete undirected graph, denoted by $G(V,E)$. V denotes the set of all peer nodes, and $E=V\times V$ refers to the set of connections. $T(V_T, E_T)$ is subgraph of G representing the multicast tree, and $T_{core}(V_{core}, E_{core})$ is subgraph of T representing the core tree. A delay cost function $c(e)$ is defined for each overlay link $e\in E$, and a node degree bound $D(v)$ is allocated for each overlay node $v\in V$.

In a shareable interaction scenario considered by the paper, the source node is the sponsor of an interactive session, such as a tutor node in an e-learning system. We call it a sponsor-based situation. Under this sponsor-based situation, the other interactive nodes communicate with the sponsor primarily. The interactive behaviors among other nodes are less important. For this situation, a node belongs to the core tree if the delay between itself and the root node is below the threshold T_{tol}. That is, for a node $v\in V_T$, let $L(P)=\sum_{e_i\in P}c(e_i)$ where the path $P=(v,\ldots,s)$ denotes the overlay path between source node s and node v in the multicast tree, the node v belongs to the core tree if $L(P)\le T_{tol}$. The complexity of the judgment is $o(n)$.

In particular, if we consider the access delay incurred by end users, the delay constraint of the core tree is adjusted to T'_{tol} as presented below:

$$T'_{tol}=T_{tol}-T_{access}. \tag{1}$$

In the formula, T_{access} denotes the average access delay of an end user to overlay tree. Since this additional access delay is considered at the moment of core tree construction in advance, the end users belonging to the core tree and the newly adjusted interactive nodes to the core tree will roughly satisfy the T_{tol} threshold. In the following parts of the paper, T_{tol} will indicates this T'_{tol} value, which simplifies the illustration.

3.3. Layer degree constraint allocation

Before tree construction, we illustrate the layered degree distribution in advance. Special LDC values are allocated in view of different layers. The out degree of the tree nodes in the same layer is constrained by a shared LDC value. Further, the strategy for allocating LDC values among different layers is to maximize the efficiency of the reserved resources. The set of LDC values is calculated as follows.

First, the number of layers needed for reservation is calculated in advance by considering the delay constraint T_{tol}, the average delay cost of an overlay hop, and the requirement of the interactive scale. The protocol evaluates in advance the maximum number of interactive nodes allowed, and defines the delay tolerance threshold for the interaction. For the average delay values, they can be obtained by some traditional methods, such as RTT detection carried out at random and limited to a few instances, or utilizing DNS queries by an existing system of DNS servers. Suppose the delay constraint is 400 ms, the average delay cost of edges is 180 ms. For the interaction scenario, due to the delay limitation, only three layers including the root can be used for reservation. And only two layers are needed for the scope of 10 interactive nodes with five nodes node degree bound.

Second, the allocation of the LDC values is implemented. Suppose $V_i \subset V_T$ indicates the tree nodes in the ith layer, and $D(V_i)=\sum_{v\in V_i} D(v)$ denotes the summarized degree bounds of the node set V_i. Then, the reserved resource in the layer can be denoted as $RSV_i = D(V_i)-N_i$, N_i is the number of nodes in the ith layer of the tree, that is, $N_i=|V_i|$. Suppose n layers is needed for reservation. If ω_i indicates the profitability of the ith layer for interaction, the entire profitability of the reserved resources can be denoted as R, which is formulated as follows:

$$R=\omega_1 \cdot RSV_1+\omega_2 \cdot RSV_2+\cdots+\omega_n \cdot RSV_n. \tag{2}$$

The layered degree distribution aims to maximize the entire profitability R, namely, the n-tuple $(N_1,\ldots,N_n)$ should satisfy the following optimization:

$$\max \quad (\omega_1 \cdot RSV_1+\omega_2 \cdot RSV_2+\cdots+\omega_n \cdot RSV_n)$$
$$\text{s.t.} \quad \begin{cases} 1 \le N_1 \le D(V_1) \\ N_1 \le N_2 \le D(V_2) \\ \cdots \\ N_{n-1} \le N_n \le D(V_n) \\ RSV_1+RSV_2+\cdots+RSV_n=N_{interactive} \end{cases} \tag{3}$$

In this way, the layer degree constraints (LDC_1, LDC_2, …, LDC_n) can be defined as the optimized result of (N_1, …, N_n), which maximizes the entire reservation profitability R. In formula (3), $N_{interactive}$ is the upper bound of the interactive scale, which is pre-decided before the tree construction. The clause $N_{n-1} \le N_n$ avoids unexpected bottleneck in the core tree during the optimization. The n-tuple $(\omega_1,\ldots,\omega_n)$ indicates the profitability of the reserved resources. In this paper, it refers to the quality of the interaction if the interactive nodes are adjusted to the reserved layers. Since upper layer an adjusted node is located, better delay performance of the interaction is obtained. For simplicity, the algorithm utilizes the reverse order of the layer ID (n, n-1, …, 1) to act as $(\omega_1,\omega_2,\ldots,\omega_n)$.

Optimization results of resource reservation for 10 interactive nodes are shown in Table 1. In the table, the 2-tuple results in the first row denote the LDCs for two layers, which are presented with different node degree bounds. The denotation $\times$ indicates there is no LDC constraint. And the 2-tuple results in the second row present the resource reservation results in the top two layers correspondingly.

For example, the tree in Fig. 1 corresponds to the first column of the table, which is constructed with reservation for 10 interactive nodes. In Fig. 1, the node degrees are set to 5 for all nodes. The layer degree constraint result is (2, 3), and the resource reservation result is (3, 7). That is, three positions are reserved for the node in layer 1 with LDC_1 setting to two, and seven positions are reserved for the two nodes in layer 2 with LDC_2 setting to 3.

Table 1
Resource reservation results for 10 interactive nodes.

	$D(v)=5$	$D(v)=10$	$D(v)=20$
(LDC_1, LDC_2)	(2,3)	(1,9)	(10, $\times$)
(RSV_1, RSV_2)	(3,7)	(9,1)	(10,0)

3.4. Construction of the core tree

Vik et al. (2008) gives an overview of some pros and cons of some spanning tree algorithms for interactive applications. It indicates that the degree-limited shortest path tree (dc-SPT) is one of the appropriate candidates for dynamic interactive applications, considering the diameter, reconfiguration time, and maximum degree. At the same time, the dc-SPT is suitable for one-way live streaming in nature, which is also a part in our proposal. Therefore, the construction of the core tree is deduced from the dc-SPT algorithm in this paper.

For the construction of the core tree, the layered degree-constrained delay-limited shortest path tree is introduced.

Definition 1. The layered degree-constrained delay-limited shortest path tree (LDC-DLSPT): Given an undirected graph $G(V,E)$, a node degree constraint $D(v)$ for each vertex $v \in V$, a set of layer degree constraints (LDC_1, LDC_2,…,LDC_n) for needed n layers, a delay cost function $c(e)$ for each overlay link $e \in E$, find a spanning tree T, starting from a root node $s \in V$, for each $v \in V$, the path $P = (v,\ldots,s)$ minimizes $\sum_{e_i \in P} c(e_i)$, subject to the constraints that the path delay does not exceed the bound T_{tol}, that is $L(P) \le T_{tol}$, $L(P)=\sum_{e_i \in P} c(e_i)$, the degree of the node $d(v) \le D(v)$, and the sum degree of the nodes in the ith layer do not exceed its layer degree constraint, that is, $d(V_i) \le LDC_i$.

LDCOM uses a heuristic SPT algorithm to construct the core tree, which is a Dijkstra-like algorithm in view of the layered structure. The core tree is constructed as a breadth-first searching (BFS) tree. Each round of tree construction begins when a newly initialized node contacts the root of the multicast tree. In each round, the new node considers its delay back to the root through the current parents in the current layer. The candidate parent node that satisfies the node degree bound and the layer degree constraint with minimal delay becomes the parent, while the minimal delay should subject to the constraint T_{tol}. Otherwise, the next layer nodes of the core tree are selected as the candidate parents and a new round begins. The successfully joined node identifies itself as a core tree node. Further, if all nodes in the core tree fail to satisfy the delay bound or two kinds of degree constraints, the node is excluded from the core tree and tries to join the extended tree in the latter occasion.

3.5. Construction of the extended tree

In the construction of the core tree, the precondition of the constraint threshold T_{tol} must be satisfied in order to meet the requirement of an interactive media. Due to the enlargement of the multicast tree, the newly joined nodes are difficult to satisfy the core tree constraint. Therefore, the extended tree is used to adopt the rest of the participants. In the extended tree, the stringent delay constraint for interactive media in the core tree construction is removed.

Definition 2. The layered degree-constrained shortest path tree (LDC-SPT): Given an undirected graph $G(V,E)$, a node degree constraint $D(v)$ for each vertex $v \in V$, a set of layer degree constraints (LDC_1, LDC_2,…,LDC_n) for needed n layers, a delay cost function $c(e)$ for each overlay link, find a spanning tree T, starting from a root node $s \in V$, for each $v \in V$, the path $P = (v,\ldots,s)$ minimizes $\sum_{e_i \in P} c(e_i)$, the degree of the node $d(v) \le D(v)$, and the sum degree of the nodes in the ith layer does not exceed the layer degree, that is, $d(V_i) \le LDC_i$.

The LDC-SPT problem is exactly the same as the traditional dc-SPT problem. The only difference is that the degrees of overlay nodes obey to a set of extra LDC values and the tree can be treated as an extension of the core tree. The layered degree-constrained manner in extended tree makes sure that the construction of the extend tree does not occupy the vacancies reserved by the core tree. LDCOM employs a similar heuristic SPT algorithm to construct the extended tree.

The joining process to the extended tree begins when a joining node fails to be adopted as a core tree node. Then, the joining node contacts the root of the tree again to become an extended tree node by a recursive process. In each round, the new node considers its delay back to the root through the candidate parents in the current layer. Different from the construction of the core tree, only a list of the candidate parents in the same layer are obtained, limited by a slip window W. The approach avoids searching in the whole layer when a large overlay occurs, and provides a scalable solution for one-way live streaming. The candidate node satisfies the two degree constraints with minimal delay becomes the parent, without considering the constraint T_{tol}. Otherwise, if all of the candidates exceed the degree constraints, another set of W candidate nodes in the layer is chosen and a new round starts. Further, the set of candidate nodes in the next layer is selected as the candidates when all nodes of the precious layer are unemployable. Finally, the joined node is marked as a node of the extended tree.

3.6. Joining of end users

LDCOM uses a similar manner as the construction of the extended tree for the joining of end users. The difference is that LDCOM simply employs the end-to-end delay between the newly joined user and the candidate overlay node as the delay metric, rather than the path delay from the source. This approach makes nearby users cluster together, and makes it possible for end users to find nearby overlay proxies. Hence, this clustered manner is in favor of making the users congruent to the network topology (Zhang et al., 2002).

Each round of joining of end user begins when a newly initialized end user contacts the root of the multicast tree. In each round, the new user considers its end-to-end delay to the current parents in the same layer, limited by a slip window W. The candidate parent node that satisfies the node degree bound and the layer degree constraint with minimal delay becomes the parent. Otherwise, if all the candidates exceed the degree constraints, another set of W candidate nodes in the layer is chosen and a new round starts. Further, the set of candidate nodes in the next layer is selected as the candidates when all nodes of the precious layer are unemployable. As mentioned earlier, the layered degree-constrained manner makes sure that the joined end users do not occupy the vacancies reserved for the interactions.

3.7. Dynamic interaction adjustment

On the scene of interaction, a dynamic interaction adjustment algorithm is introduced. If an interactive node v belongs to the core tree, it means the path delay is under the threshold T_{tol}, namely, $L(P) \leq T_{tol}$, $P = (v,\ldots,s)$. Therefore, the interaction is permitted. the stream of the interaction from the source node to the node is delivered along the tree path. Meanwhile, the stream of the interaction from the interactive node to the source are directly sent back to the root node, and delivery to all participants by the multicast tree. If an interactive node is located beyond the scope of the core tree, the adjustment algorithm is carried out as follows. the interactive node separates from its parent and jumps to the reserved position in the core tree from the root node to the lower layers nodes. During the process, the degrees of all overlay nodes are constrained by the node degree constraint $D(v)$. When the connected linkages of upper layer nodes reach the threshold $D(v)$, the newly adjusted node turns to join the position in lower layers in the reserved core tree. The adjustment process is succeeded when the node found an appropriate parent within the degree bound. Otherwise, the interaction behavior is refused.

In LDCOM, when interaction session is concluded, interactive users can simply rejoin the multicast tree. In this way, the reserved resources are released, which makes the proposal flexible to dynamic interactive groups.

4. Dynamic optimization

The dynamic attribute of infrastructure-based overlay multicast lies in two aspects. First, the status of overlay nodes changes according to the variation of network condition, such as the delay and bandwidth performance. Second, the dynamic lies in the unstable of the overlay nodes. To guarantee the efficiency and resilience of LDCOM, dynamic optimization of the tree construction is introduced. The protocol treats the two situations respectively.

4.1. Core tree dynamic optimization

Due to the variation of network condition, LDCOM periodically updates the scope of the core tree according to the delay constraint, so as to insure that the nodes in the core tree satisfy the requirement of the T_{tol} correctly. The frequent update of the core tree is a huge burden to a large-scale multicast tree. LDCOM fulfills the update of the core tree by maintaining and refreshing a list of the core tree edge at certain interval, that is, the leaf nodes of the core tree only. For node v in the list, LDCOM utilizes the delay performance of node v to update the scope of the core tree recursively. If the delay result cannot satisfy the constraint, the process traces up to the father of node v. Otherwise, the children of node v in extended tree are selected as the substitute candidates for node v in the list. The recursive process is terminated until an appropriate ancestor or a set of offsprings of node v is found, then the substitute is inserted in the list, and the next candidate leaf node in the list is picked. In this way, the deteriorated nodes in the core tree are eliminated, and at the same time, some extended tree nodes with ameliorative performance are included into the core tree.

4.2. Resilience for the churn

The failure, or leaving of nodes, is the main issue in P2P systems. However, it is less important in an infrastructure-based overlay multicast with dedicated, stable proxies. LDCOM employs a reactive mechanism to provide data recovery. When leaving or failure of a nonleaf overlay node occurs, an additional simple process is carried out to select an alternative parent for the subtree of the node.

If the leaving node belongs to the extended tree, for simplicity, the first found leaf node in the subtree of the leaving node is selected as the new parent. In another occasion, if a failure node belongs to the core tree, the above process is still carried out to select an alternative parent for the subtree of the failure node. Differently, a further verification of the core tree's identity is preformed for the whole subtree within the core tree scope. The verification follows the method in the precious part of the core tree optimization.

Finally, for both the dynamic interaction adjustment and the resilience for the churn, since there is no process for the delay detection, these on-the-fly modifications are instantaneous and the temporary intervals are imperceptible.

5. Simulation

5.1. Simulation setup

The performance of LDCOM is demonstrated in simulation by network simulator (NS2, 2010). All the overlay nodes are attached to a set of stub nodes (at the verge) in GT-ITM topology (Zegura et al., 1996). The overlay topologies are distributed in the size of 80 nodes, 320 nodes, and 2560 nodes, respectively. The node degrees of overlay nodes are set to 5, 10, and 20, respectively. The number of concurrent interactive clients starts from 1 to 10 increasingly. In the simulation, the threshold T_{tol} is mainly set to 400 ms according to ITU-T standard (ITU-T, 2006), and is also set to 600 ms and 800 ms with less stringent manner. The overlay processing delay and the access delay of an end user are set to 60 and 20 ms, respectively. Finally, in the simulation and the following experimental study, all the delay pairs are obtained by averaging the delay results of several ICMP pings.

Uniformly, the simulation results in figures by the proposal before employing the interactive adjustment algorithm and after performing the adjustment algorithm are named LDCOM and LDCOM*, respectively. Notice that the simulation results before interactive adjustment can also be seen as the average delay results of one-way live streaming. Moreover, to reduce the randomness of the simulation and experimental results, the evaluation results presented in this paper are the average results over 10 runs at least. In the simulation, we show only the sets of means.

Finally, the dc-SPT is used as a benchmark for quantitative comparisons with the proposed approach in the simulation evaluation as well as the experimental study. In the evaluations, the average delay results by standard dc-SPT method are provided. Notice that the dc-SPT is a global optimization method, which means that the dc-SPT tree is built by considering the path delay between the joining node and the source node through all the tree nodes under the degree constraints. Therefore, it is a convincing comparative study benchmark.

5.2. Average path delay of interactive nodes

The core idea of LDCOM is to decrease the average path delay between interactive nodes and the source node for the sponsor-based interaction. In the simulation, for each step of increasing the interactive nodes scale, the average path delay for all interactive nodes is obtained.

Suppose the set of the interactive nodes is denoted as $V_{interactive} \subset V_T$. The average path delay of $L(P)$ for all of $v \in V_{interactive}$ is calculated as follows:

$$L_{avg} = \frac{\sum L(P)}{|V_{interactive}|}, \quad P=(v,\ldots,s), \ \ v \in V_{interactive}. \tag{4}$$

The L_{avg} results before and after adjustment by LDCOM are shown in Fig. 2. The number of interactive clients starts from 1 to 10 increasingly. The $D(v)$ is set to 5, and the T_{tol} is set to 400 ms. In the simulation, with the adjustment introduced by the proposal, the L_{avg} values after the adjustment remain stable within the 400 ms delay constraint. The dynamic interaction adjustment algorithm restrains the increasing tendency of the delay between the source and the interactive users. In contrast, the path delay result of dc-SPT increases along with the enlarging topology, since both of the physical link delay and the application-layer relay delay increase in the same time. In Fig. 2(c), the average delay by dc-SPT is already larger than the 400 ms delay threshold.

To be more general, the L_{avg} results with $D(v)$ setting to 10 and 20 are also obtained for 320 nodes topology in Fig. 3. In Fig. 3, the same as in Fig. 2, all the average delay values after adjustment by LDCOM remain stable within the latency requirement. In comparison with LDCOM, the results of dc-SPT in Fig. 3(b) have similar performances. It is because that many interaction requirements are directly allowed by LDCOM without adjustment, with the node initially belonging to the core tree.

In brief, LDCOM can stably provide path delay guarantee for the interactive media with different network conditions, which achieves our designing goal. Furthermore, LDCOM outperforms dc-SPT with stringent constraints. However, without the adjustment, it incurs some delay for one-way live streaming compared with the dc-SPT. It can be explained as the compound result of two factors: One is the resource reservation in top layers, and more significantly, the other is the heuristic tree construction method in view of only a layer of the tree. Notice that the dc-SPT is a global optimization method.

5.3. Acceptance ratio of the interaction

LDCOM fully guarantees the delay requirement for the interaction, which is the key purpose of the proposal. To achieve this delay assurance, there must be enough vacations reserved in the core tree. Therefore, the scope of the core tree and the scope of the interactive nodes determine the acceptance ratio of the interaction. Under stringent network conditions, such as rigorous delay and degree constraints with large amount of interactive participants, LDCOM may decrease the acceptance ratio of the interactive requirements.

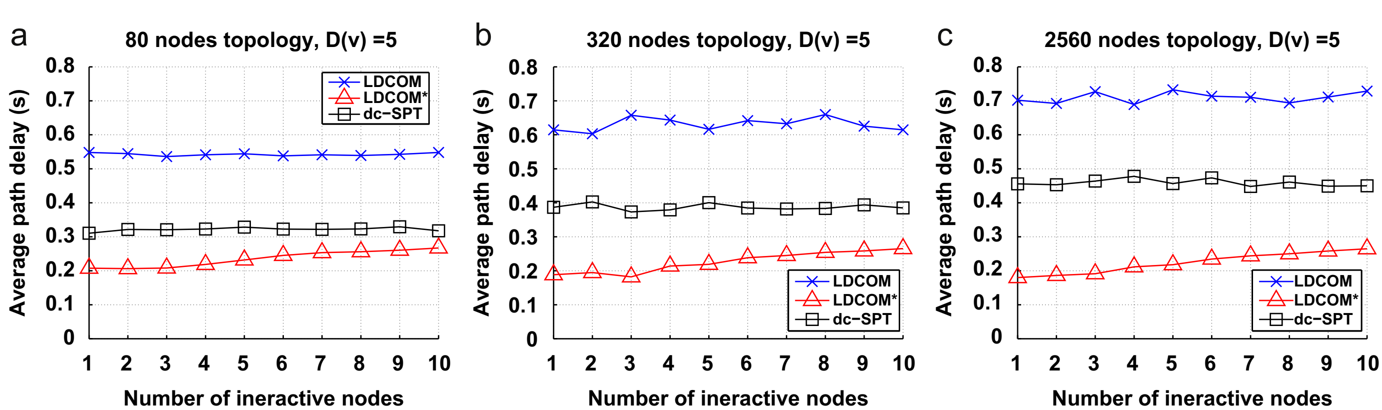

Fig. 2. Average path delay results of interactive nodes with different topologies: (a) 80 nodes topology, $D(v)=5$; (b) 320 nodes topology, $D(v)=5$; (c) 2560 nodes topology, $D(v)=5$.

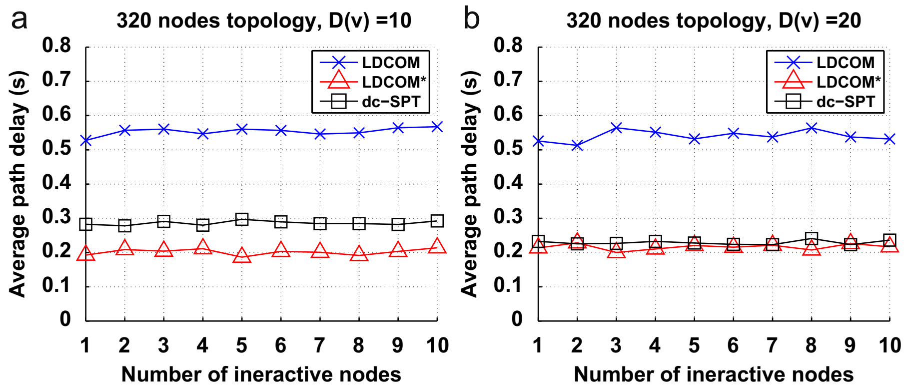

Fig. 3. Average path delay results of interactive nodes with different degree constraints: (a) 320 nodes topology, $D(v)=10$ and (b) 320 nodes topology, $D(v)=20$.

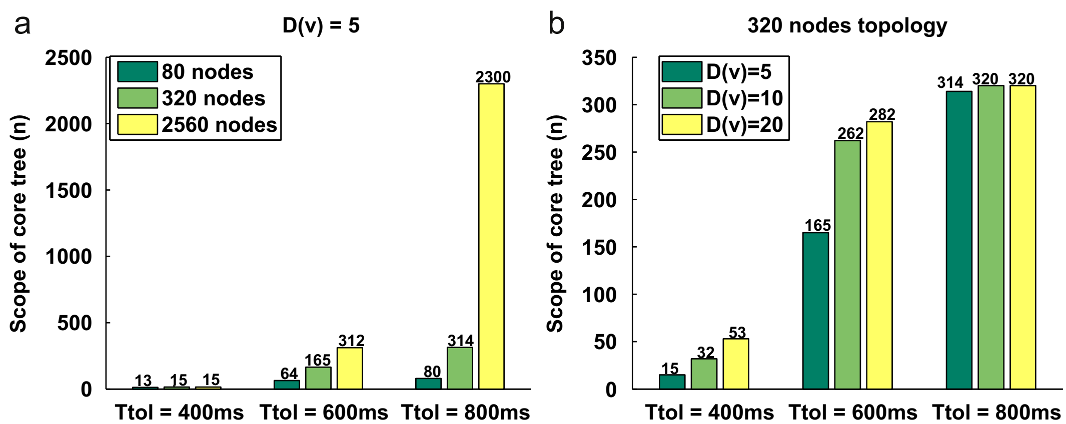

Fig. 4. Scope of the core tree: (a) different topologies, $D(v)=5$ and (b) different degree constraints, 320 nodes topology.

Fig. 4 presents the core tree scope by LDCOM for different topologies with different degrees and delay constraints. As shown in Fig. 4(a), a smaller core tree scope is obtained under a set of stringent conditions. It indicates that smaller reservation can be made accordingly. However, since the scope of the shareable interaction usually limits within a small group due to the bandwidth problem, the proposal works well in most cases. Under the settings in the simulation, the acceptance ratio is about 100 presents within ten interactive nodes.

6. Experimental results

As mentioned earlier, this research is inspired by a collaborative live teaching system built by our research group. We have actually realized a synchronous live teaching prototype in which the proposed algorithms have been implemented. The prototype consists of four instances besides the database. They are the centralized control server (CCS), the teacher end, the student end, and the data transmit unit (DTU). Dedicated DTU nodes are used as the overlay proxies to construct the multicast trees. Under the TCP session control of the CCS, multi-channel live teaching media is delivered along the multiple multicast trees by RTP/RTCP over UDP datagram. For each channel individually, the two-tier tree construction method of LDCOM is adopted. In this manner, the prototype can simultaneously support large-scale one-way live streaming and two-way shareable interaction between tutor and participants. Fig. 5 is snapshots of a student node in three scenarios. Figure 5(a) shows the live teaching scenario, Fig. 5(b) presents the interaction scenario where the node is one of the interactive nodes, and Fig. 5(c) illustrates the situation that the node shares the interaction scenario while it does not belong to the interaction group.

To further demonstrate the efficiency of LDCOM in achieving our objectives, we perform an experiments by deploying the algorithms on PlanetLab (2006). The experiment measures the average path delay reducing for the interaction. The same as the simulation, the experimental results are average results over 10 runs. For each time, the 280 nodes are random selected from 306 nodes distributed on the PlanetLab sites. The pair-wise delay between two planetlab nodes is also an average value of ICMP Pings. The scope of interactive nodes is set to 10. The obtained result of the average round trip time is 250 ms, therefore, the average delay among planetlab nodes is about 125 ms. Furthermore, the tested processing delay of one node is about 60 ms, and the threshold T_{tol} is set to 400 ms according to the ITU-T standard.

Table 2 shows the experimental result. The path delay between the root and the interactive nodes by LDCOM is within the delay threshold after the adjustment. LDCOM fully guarantees the delay requirement for interactive nodes in the interactive scenario. Further, it outperforms the dc-SPT after the dynamic adjustment.

7. Conclusion and future work

This paper addresses the issue to simultaneously provide shareable interaction as well as live streaming by overlay multicast, which is a commonly requirement in many collaborative multimedia applications. LDCOM utilizes a layered degree-constrained strategy to reserve resources at top layers of media delivery tree, and supports subsequent interactive media. Besides the one-way live streaming, a

b

c

Fig. 5. Snapshots of a student node: (a) live teaching mode; (b) interaction scenario that the node belongs to the interaction group; (c) interaction scenario that the node does not belong to the interaction group.

Table 2
Experimental result.

Tree algorithm	Core tree scope	Acceptance ratio	Path delay (s)	
			Initial	Adjusted
LDCOM	98	100%	0.4053	0.2488
dc-SPT	×	×	0.3011	

our simulation and experiment results show that the approach stably provides delay guarantee for interactive media. With undesirable network situation or stringent application requirements, LDCOM outperforms the dc-SPT by adopting a dynamic adjustment for the interactive nodes.

In addition to the delay-sensitive interactive media, more benefits may be gained from the reserved resources in top layers of media delivery tree. The future work will try to expand the utilization of LDCOM structure by importing other QoS metrics. Furthermore, besides the simulations and the experiments of the proposal, the future work will include evaluating the efficiency of the algorithm in real collaborative learning environment, obtaining subjective judgments by users.

Acknowledgements

Funding for this work was provided by China NSF Grant (60825202, 61070072, 60921003), National Science and Technology Major Project Grant (2010ZX01045-001-005), Key Projects in the National Science & Technology Pillar Program Grant (2009BAH51B02), the Fundamental Research Funds for the Central Universities under Grant No. xjj20100057, xjj20100052 and China CNGI Project Grant (CNGI-09-01-13).

References

Akamai, 〈http://www.akamai.com/ 〉; 2010.
Bluesky, 〈http://incubator.apache.org/bluesky/ 〉; 2010.
Cai Y, Zhou J. An overlay subscription network for live internet tv broadcast. IEEE Transactions on Knowledge and Data Engineering 2006;1:1711–20.
Capone A, Elias J, Martignon F. Routing and resource optimization in service overlay networks. Computer Networks 2009;53(2):180–90.
Castro M, Druschel P, Kermarrec A, Nandi A, Rowstron A, Singh A. SplitStream: high-bandwidth multicast in cooperative environments. In: Proceedings of the nineteenth ACM symposium on operating systems principles; 2003. p. 298–313.
Chu Y, Rao S, Seshan S, Zhang H. A case for end system multicast. IEEE Journal on Selected Areas in Communications 2002;20(8):1456–71.
Dan G, Fodor V, Chatzidrossos I. On the performance of multiple-tree-based peer-to-peer live streaming. In: Proceedings of IEEE INFOCOM'07; 2007. p. 2556–60.
Diot C, Levine B, lyles B, Kassem H, Balensiefen D. Deployment issues for the ip multicast service and architecture. IEEE Network Magazine 2000;14(1):78–88.
Do T, Hua K, Tantaoui M. Robust video-on-demand streaming in peer-to-peer environments. Computer Communications 2008;31(3):506–19.
Granda J, Garcia D, Nuo P, Suarez F. An efficient networking technique for synchronous e-learning platforms in corporate environments. In: International symposium on performance evaluation of computer & telecommunication systems, 2009. SPECTS 2009, vol. 41. IEEE; 2009. p. 96–103.
Gu X, Wen Z, Yu P, Shae Z. Supporting multi-party voice-over-IP services with peer-to-peer stream processing. In: Proceedings of the 13th annual ACM international conference on multimedia; 2005. p. 303–6.
Guo D, Liu Y, Li X. Bake: a balanced kautz tree structure for peer-to-peer networks. In: Proceedings of IEEE INFOCOM'08; 2008. p. 2450–7.
Guo J, Jha S. Pricing multicast proxies for internet live media streaming dimensioning server access bandwidth and multicast routing in overlay networks. In: Proceedings of IEEE LCN; 2007. p. 149–56.
Guo Y, Suh K, Kurose J, Towsley D. DirectStream: a directory-based peer-to-peer video streaming service. Computer Communications 2008b;31(3):520–36.
Hosseini M, Ahmed D, Shirmohammadi S, Georganas N. A survey of application-layer multicast protocols. IEEE Communications Surveys & Tutorials 2007;9(3): 58–74.
ITU-T. Y.1541: Network Performance Objectives for IP-Based Services. International Telecommunication Union; 2006.
Lao L, Cui J, Gerla M, Chen S. A scalable overlay multicast architecture for large-scale applications. IEEE Transactions on Parallel and Distributed Systems 2007:449–59.
Li B, Yin H. Peer-to-peer live video streaming on the internet: issues, existing approaches, and challenges. IEEE Communications Magazine 2007;45(6):94–9.
Li Z, Zhu Z, Xie G, Li Z. Fast and proximity-aware multi-source overlay multicast under heterogeneous environment. Computer Communications 2009;32(2): 257–67.
Liu L, Zimmermann R. A low latency P2P live streaming architecture. In: Proceedings of SPIE/ACM conference on multimedia computing and networking (MMCN); 2005. p. 26–37.
Lua E, Zhou X, Crowcroft J, Mieghem P. Scalable multicasting with network-aware geometric overlay. Computer Communications 2008;31(3):464–88.

Luo C, Li J, Li S. DigiMetro-an application-level multicast system for multi-party video conferencing. In: Proceedings of IEEE GLOBECOM'04; 2004. p. 982–7.
Magharei N, Rasti A, Stutzbach D, Rejaie R. Peer-to-peer receiver-driven mesh-based streaming. In: Proceedings of ACM SIGCOMM'05; 2005. p. 1415–23.
Magharei N, Rejaie R, Guo Y. Mesh or multiple-tree: a comparative study of live p2p streaming approaches. In: Proceedings of IEEE INFOCOM'07; 2007. p. 1424–32.
Maraviglia G, Masi M, Merlo V, Licandro F, Russo A, Schembra G. Synchronous multipoint e-learning realized on an intelligent software-router platform over unicast networks: design and performance issues. In: Emerging technologies and factory automation, 2007. ETFA, IEEE Conference on: IEEE, 2008. p. 1172–9.
NS2, 〈http://www.isi.edu/nsnam/ns/〉; 2010.
PlanetLab, 〈http://www.planet-lab.org〉; 2006.
Pompili D, Scoglio C, Lopez L. Multicast algorithms in service overlay networks. Computer Communications 2008;31(3):489–505.
Ren D, Li Y, Chan S. On reducing mesh delay for peer-to-peer live streaming; 2008. p. 1732–40.
Seibert J, Zage D, Fahmy S, Nita-Rotaru C. Experimental comparison of peer-to-peer streaming overlays: an application perspective. In: Proceedings of IEEE LCN; 2008. p. 20–7.
Venkataraman V, Yoshida K, Francis P. Chunkyspread: heterogeneous unstructured tree-based peer-to-peer multicast. In: Proceedings of ICNP'06; 2006. p. 2–11.
Vik K, Halvorsen P, Griwodz C. Multicast tree diameter for dynamic distributed interactive applications. In: Proceedings of IEEE INFOCOM'08; 2008. p. 1597–605.
Wang F, Liu J, Xiong Y. Stable peers: existence, importance, and application in peer-to-peer live video streaming. In: Proceedings of IEEE INFOCOM'08; 2008. p. 2038–46.
Xu J, Li V. Request-driven swarming scheme for P2P data streaming. Computer Communications 2009;32(12):1410–7.
Yang S, Kim Y, Wang B. Designing infrastructure-based overlay networks for delay-sensitive group communications. In: Proceedings of IEEE GLOBECOM'07; 2007. p. 565–70.
Zegura E, Calvert K, Bhattacharjee S, et al. How to model an internetwork. In: Proceedings of IEEE INFOCOM'96. vol. 2; 1996. p. 594–602.
Zhang B, Jamin S, Zhang L. Host multicast: a framework for delivering multicast to end users. In: Proceedings of IEEE INFOCOM'02; 2002. p. 1366–75.
Zhang X, Liu J, Li B, Yum T. CoolStreaming/DONet: a data-driven overlay network for peer-to-peer live media streaming. In: Proceedings of INFOCOM'05; 2005. p. 2102–11.