An Adaptive Load Balancing Strategy of Application Layer Multicasts Based on Score of Customer Satisfaction

Jianqun Cui[1], Liang Ma[2], Kuan Gao[1], Libing Wu[3], Yi Yang[4]

[1] School of Computer Central China Normal University, China
[2] NO. 1 Middle School affiliated to Central China Normal University, China
[3] School of Computer Wuhan University, China
[4] Department of Math and Computer Science, Fontbonne University, USA
jqcui@126.com, mail.qqservces@foxmail.com, kuanzhs@163.com, wu@whu.edu.cn, yyang@fontbonne.edu

Abstract

In P2P networks, due to the diversity of population density, there is an imbalance on users' demands towards the internet resources. This paper proposes a method called "An Adaptive Load Balancing Strategy of Application Layer Based on Score of Customer Satisfaction", ALBS-SCS for short. The method gives a score on the QoS of current networks periodically, and then gives feedback to the system, and eventually the system adjusts its services accordingly. This kind of method can truly improve user experience by providing better Multicast User Satisfaction (MUS) for the end systems of the hot spots and congestion areas, at the same time it also promotes the resource utilization rate of the non-hot spots, increases the network capacity and improves the performance of the system. The simulation experiments prove that the adaptive load balancing mechanism ALBS-SCS is better at reducing the average transmission delay and the average link pressure, thus increasing the system average MUS, and improving the overall performance and the user experience.

Keywords: MUS, ALBS-SCS, Multicast network, Hot spot, Load balancing

1 Introduction

Peer to Peer (P2P) networks [1-2] were once appraised by the Fortune magazine as one of the four new technologies that changed the development of the Internet technology. P2P networks, in which all nodes are in the same status, is different from the traditional client server model. That is to say, every node is both the client and the server at the same time. By sharing resources with each other, nodes in the system accomplish the task together. But in reality, because of different intensity of regional population, users' demand for resources is not balanced; different computer has heterogeneity in processing power. Thus, in hot communication region, terminal business may be busy; some of the terminal parent nodes may carry too many children, which leads to the poor quality of streaming media services. However, in other regions, the terminal business may be sparse, when the amount of resources provided is the same, some terminal parent nodes carry a few child nodes, which leads to a situation that business flow is very little and resources are excess. Therefore, the rate of overall resource utilization of system is low, the quality of the service cannot be guaranteed, which affects users experience seriously.

In reality, situation as shown in Figure 1 may appear. Many users in the region where v_3 is the parent node, while a few users in the region where v_4 and v_5 are the parent nodes. In this case, most algorithms simply think that the service in the region where v_3 is the parent node is poor, while the resources of the region where v_4 and v_5 exist are relatively surplus, so that load balance is adjusted according to the number of users or other indicators. But actually, due to clusters where v_4 and v_5 exist are not very well in terms of processing power of nodes and regional power of the Internet. If we adjust service only according to the indicators above, the network condition may not in accordance with the actual user evaluation.

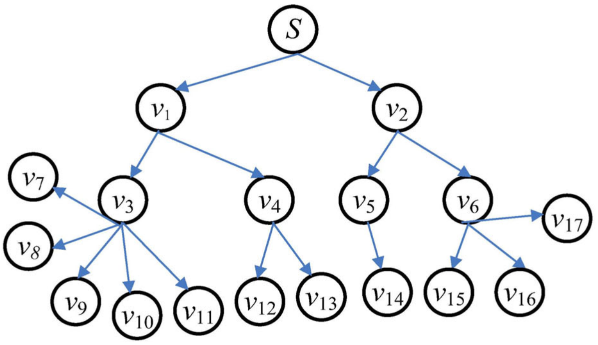

Figure 1. The diagram of a multicast tree

This paper proposes an ALBS-SCS mechanism and

---

*Corresponding Author: Jianqun Cui; E-mail: jqcui@126.com

DOI: 10.3966/160792642018091905014

it not only provides better multicast user satisfaction (MUS) for the hot spots and congestion areas, but also improves resource utilization of non-hotspot areas, to increase network capacity and improve the overall performance of the system.

The outline of this paper is organized as following: section II analyzes the category and reasons of present P2P internet load balance. Section III discusses ALBS-SCS multicast model in detail. Section IV shows the results of experiments on ALBS-SCS multicast model, and it also analyzes the data of the experiment. Section V summarizes the paper and indicates the future work.

2 Related Work

A peer-to-peer [3] (P2P) computer network is one in which each computer in the network can act as a client or server for other computers. For the next generation Internet applications, P2P is considered as the most important idea. To improve the efficiency of the system, load balancing mechanism among peers is critical. The main idea [4] of load balancing is to transfer part of the traffic from the heavily loaded paths to some lightly loaded paths to avoid congestion in the shortest path route and increase the network utilization and network throughput. The basic approach to load balancing is to find a pair of nodes—one is heavily loaded and the other is lightly loaded—and redistribute the load across these two nodes [5].

However, it is far from trivial to (globally) balance the load in a P2P system. There are two main issues in load balancing for P2P: 1) how to determine if a node is overloaded or underloaded, and 2) if so, how to find a suitable partner node with which to redistribute the load. A popular solution in [6-8] is to let each node in the system query for the load of an arbitrary number of other nodes periodically. If the number of queried nodes is large enough, the node can reach the average load of the system. Therefore, the system can determine whether the node is overloaded or not. If the node is overloaded (or underloaded), the system will redistribute its load with the queried node which has the lightest (or heaviest) load since that node should be a lightly (or heavily) loaded node. The main problem with this method is that it can only guarantee the global load balance of the system with some probability. On the other hand, [9] suggests the use of a separate DHT (Distributed Hash Table) such as Skip Graph to maintain the nodes' load distribution. Nevertheless, this solution still has a problem: it incurs a substantial cost for maintaining complete information about the load at every node in the system [10].

In this paper, we propose a new framework, called "A Adaptive Load Balancing Strategy of Application Layer Based on Score of Customer Satisfaction" in structured P2P systems. It not only provides better multicast user satisfaction (MUS) for the hot spots and congestion areas, but also improves resource utilization of non-hotspot areas, to increase network capacity and improve the overall performance of the system. The basic idea of ALBS-SCS: First of all users for a variety of service quality in current multicast network periodically score, and then users take the initiative to feedback satisfaction to the multicast network, finally the system is adjusted periodically according to the user's feedback, really improving the user experience.

3 Adaptive Load Balancing Strategy

Initially, we assume that the processing capacity of each user is the same, and the number of users in each area is roughly the same; of course, the actual environment is more complex. According to the basic idea of ALBS-SCS, the system periodically adjusted the current multicast network to improve the user experience and improve the overall performance of multicast network.

3.1 MUS and Related Performance

The key to establish a good ALBS-SCS is to select the appropriate performance indicators of application layer multicast network, and identify relevant performance for the impact of the current network quality. In addition to planning MUS grade-level indicators and the associated lag, threshold parameters, only in this way, a clear, specific and accurate reflection of the current real user experience can be gotten to make adjustment mechanism ALBS-SCS more perfect, truly improve the user experience of the application layer multicast. Firstly, the relevant performance indicator of ALBS-SCS used is defined as follows.

3.1.1 Basic Performance of MUS

In this paper, the multicast users load availability, quality of service and channel capacity are taken as the basis for the MUS performance. These performance indicators are defined as follows:

Definition 1. Load availability (LAR, Load Available Rate), refers to the percentage of the downlink system bandwidth of the current node has been used. Its expression is as follows:

$$LAR = \frac{DD_{\text{current}}}{DD_{\max}} \times 100\% \tag{1}$$

In this formula, $DD_{current}$ is the downstream bandwidth of the current node and DD_{max} is the maximum bandwidth of the system requirements.

Definition 2. Quality of Service (QoS) is the key indicators, including: availability, bandwidth, throughput, delay, delay variation (including jitter and wander) and packet loss rate. In this paper, we use throughput as the QoS performance indicators.

Definition 3. Channel Capacity (CC), is one of the most important parameters of a communication system, which represents the maximum information transmission rate of the communication system. According to Shannon information theory with the limited input power P and noise variance N0 discrete-time plus Gaussian white noise channel, the channel capacity (the unit is bit / s) is expressed as follows:

$$CC = \frac{1}{2}\log_2(1+\frac{p}{N_0}) \quad \textbf{(2)}$$

Definition 4. multicast user i satisfaction scores (MUS_i) is expressed as follows:

$$MUS_i = \alpha W_{LAR} + \beta W_{Qos} + \lambda W_{CC} \quad \textbf{(3)}$$

In this formula, i is an arbitrary integer that represents any one of a multicast user node. W_{LAR} is the weighting function of the load availability. W_{Qos} is weighting function of the quality of service. W_{CC} is the weighting function of the channel capacity. α, β, γ are the corresponding weighting coefficients and α, β, γ satisfies the following expression:

$$\alpha + \beta + \lambda = 1 \quad \textbf{(4)}$$

In this formula, α, β, γ value will be set at a specific part of the experiment described later, according to the current network conditions.

3.1.2 Intended User's MUS Level of Geographical Boundary

ALBS-SCS cluster is divided into three levels of MUS. Level 1 is representative of areas of the best MUS, level 2 is representative of areas of balance MUS and level 3 represents the location of poor MUS. The specific evaluation mechanism is as follows:

Level 1 cluster average MUS judgment mechanism is as follows:

$$\frac{1}{K}\sum_{i=1}^{k} MUS_i - HYS \geq Threshold_1 \quad \textbf{(5)}$$

In this formula, MUS_i is the parent node of the i-th child node of MUS, K is the total number of the child nodes which the parent node has, HYS is delay parameters of the system requirements, and $Threshold_1$ is level 1 MUS threshold value.

Level 2 cluster average MUS judgment mechanism is as follows:

$$\frac{1}{K}\sum_{i=1}^{k} MUS_i + HYS < Threshold_1 \quad \textbf{(6)}$$

$$\frac{1}{K}\sum_{i=1}^{k} MUS_i - HYS \geq Threshold_2 \quad \textbf{(7)}$$

MUS_i, K, HYS in Formula 6, 7 are the same as the Formula 5, $Threshold_2$ is level 2 MUS threshold value. ($Threshold_2$ is set in the specific part of the experiment).

Level 3 cluster average MUS judgment mechanism is as follows:

$$\frac{1}{K}\sum_{i=1}^{k} MUS_i + HYS < Threshold_2 \quad \textbf{(8)}$$

MUSi, *K*, *HYS* in Formula 8 are the same as Formula 5.

3.2 ALBS-SCS Update Mechanism

ALBS-SCS update mechanism use the periodic active feedback information [19] update mechanism, and its specific implementation steps are as follows [11-12].

(1) Every certain period of time T, the bottom of the parent node sends a MUS query message to the RP nodes;

(2) After RP node receives a MUS query message which the lowest level parent node sends, it will return a MUS query response message to the entire cluster parent node uniformly. (Due to a number of times connection cannot guarantee a successful connection in the specific circumstances, so we believe that parent node which can correctly receive confirmation message sent by RP node is a normal connection node at this time.)

(3) When the parent node of each cluster receive confirmation query message which RP node sends, the parent node is sending all MUS scoring of the current network to the cluster children nodes immediately;

(4) After each cluster node receives the satisfaction of scoring of its parent node which sends a request, they are immediately scoring three performance indicators such as the current network load availability, quality of service (QoS), the channel capacity, and then scoring results feedback to the parent node;

(5) Based on their own children nodes which received for the network score, each parent node weights each network indicators. Then according to the Formula 3, it can be derived from the current user MUS. Finally, based on the number of children nodes in the cluster and MUS of each child node, the parent node calculates the current average MUS of cluster, meanwhile this result is fed back to RP node. According to the feedback of each cluster parent, RP nodes adjust the current multicast network.

All messages are inherited from Base Overlay Message. The format of ALBS_SOCRE_RESPONSE (Score confirmation message) is shown in Table 1.

Table 1. Score confirmation message

int command enum (ALBSCommand);	Inheritance source message
TransportAddress srcNode	Source Address
TransportAddress destNode	Destination address
int QoS;	Quality of Service scoring
int LAR	Load availability
int CC	Channel capacity

Update mechanism pseudo-code is as follows:

```
new AlbsMusQueryMessage("ALBS_MUS_QUERY");
sendMessageToUDP (serverNode, query);
if (dynamic_cast<AlbsMusQueryMessage*> (msg) != NULL)
 {
   new AlbsMusResponseMSG
   ("ALBS_MUS_RESPONSE";
   sendMessageToUDP(srcNode, response);
 }
if (dynamic_cast<AlbsMusResponseMessage*> (msg) != NULL)
 {
   for (TaSetIt itn = node->begin(); itn != node->end(); ++itn)
 {
   new TestDropMessage("ALBS_SCORE_QUERY";
   sendMessageToUDP(*itn,score);
 }
 }
if (dynamic_cast<AlbsScoreQueryMessage*> (msg) != NULL)
 {
   new AlbsMusRespMSG("ALBS_MUS_RESPONSE";
   sendMessageToUDP(srcNode, response);
 }
```

3.3 ALBS-SCS Adjustment Mechanism [17-18]

In the communication of application layer multicast, if there is too much load balancing adjustment, it will not only increase the cost of the system, but also lead disconnect and reconnect frequently. This kind of situation seriously affects the quality of communication. Conversely, if the load balancing mechanism works too slowly, it will lead to hot regions overload for a long time, which will reduce the quality of service and affect the user experience. At the same time, an inefficient load balancing mechanism simply settings from the overloaded area to light load transfer, which may cause the overloading region into a light load region, and in turn the previous light load region becomes a new overloaded regional. So the arrival of the next scanning period T, the inefficient load balancing mechanism needs to be re-adjusted, so that the weight load region is swapping now. It is commonly known as ping-pong effect. Such adjustment process not only does not improve MUS, but also wastes system resources. Therefore, it is crucial for a good load balancing adjustment mechanism. First of all, based on the characteristics of the largest number of leaf nodes of the multicast tree, ALBS-SCS selects adjustment methods to accelerate adjustment cycle, reduce blocked communications. Secondly, the update cycle time is set so that the node status of the adjustment time can be relatively consistent, and to avoid the contradiction between the node regulation time and the update mechanism cycle time. Finally, according to various possible scenarios, nodes are provided with a corresponding fast adjustment mechanism [13]. ALBS-SCS specific adjustment mechanism is as follows:

According to each cluster parent node feedback average cluster MUS, the RP node uses the Formula 5, 6, 7, 8 to calculate the average value of MUS within the entire cluster. If this time there is no average MUS for third level cluster, multicast network does not adjust and wait for the next feedback again. On the other hand, if this time there is an average MUS of third level cluster, the RP node sends load balancing instructions and all parent node information of first level average MUS to those parents in cluster. Since the average MUS of second level cluster is a balanced cluster, it does not need to adjust. Therefore, the number of the cluster second level average MUS has no effect on the ALBS-SCS adjustment mechanisms. We do not analyze the different number of second level average MUS cluster. However, the average MUS of third level cluster needs to be adjusted and the average MUS of first level can be accepted of adjustment. The number of clusters has an impact on the ALBS-SCS adjustment mechanism, according to the number of the average MUS of first level and third level which we describe separately. Because of that not every average MUS of first level cluster can accept the average MUS of third level cluster (presence of ping-pong effect), ALBS-SCS adjustment mechanism divides the average MUS of first level cluster into admissible switching cluster and non-admissible switching cluster, which is described as follows.

Definition 5. admissibility switch cluster is that the average MUS of first level accepts switching children nodes of the average MUS of third level cluster, the cluster can keep at least for the second level.

Definition 6. non- admissibility switch cluster is the cluster that the average MUS of first level may become the third level after the cluster accepts switching children nodes of the cluster whose average MUS is in third level.

In Figure 2, we specifically analyze an admissibility switch cluster and a non- admissibility switch cluster. As shown in Figure. 2, Firstly, v_3, v_4, v_5 and v_6 gather statistics of the score of its corresponding children nodes on the cluster load for multicast network availability, quality of service (QoS), channel capacity index weighted score of MUS (assuming the MUS of v_7, v_8, v_9, v_{10}, v_{11}, v_{12}, v_{13} and v_{14} is 30, 60, 40, 80, 80, 85, 90, 90 respectively, Threshold1 is 80, Threshold2 is 60.). Secondly, send the statistical results back to RP node. RP node calculated the average MUS of the cluster which is under the leadership by v_3, which is less than 60, and belongs to the third level. The average MUS of the cluster led by v_4 is equal to 80, which belongs to the second level. The average MUS of cluster led by v_5, v_6 parent node is greater than 80, which belong to the first level. RP node transmits load balancing adjustment instructions and the information of v_5, v_6 whose average MUS belongs to first level to v_3. v_3 node sends a handover request command and the evaluation MUS of the children nodes of v_3 to v_5 and v_6. v_5 and v_6 consider the worst MUS of the child node of v_3 as first choice to switch a single step after receive handover request message from v_3 (avoiding ping-pong effect, reducing the number of switching cycles). Meanwhile ALBS-SCS adjustment mechanism calculates the smallest possible system after switching MUS (Because the load availability, quality of service (QoS), the channel capacity of these indicators of the first level is better than the three levels. Switching node causes MUS increased, so the minimum MUS is original MUS in the cluster). After switching, the smallest average MUS of v_5 is less than 60, and the level of the average MUS of this cluster may become the third level, therefore the cluster led by v_5 is non-admissible switching node cluster of v_3. However, after switching the smallest average MUS of the cluster led by v_6 is still greater than 60, so the parent node of the cluster can accept the children nodes of v_3 as switching nodes. Therefore, the cluster led by v_6 can be admissibility switch cluster of v_3.

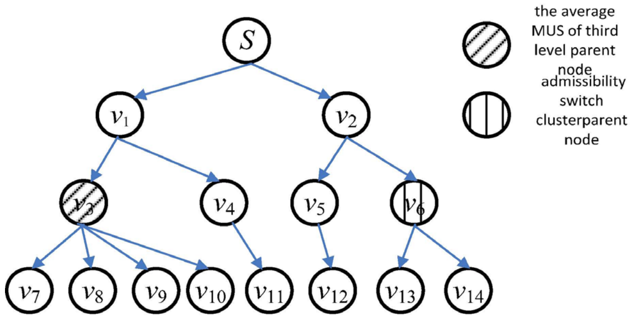

Figure 2. Schematic diagram of multicast tree before ALBS-SCS adjustment mechanism operation

In order to achieve better load balancing adjustment mechanism [14], ALBS-SCS adjustment mechanism divide the parent node of the cluster whose average MUS belongs to third level into two states: idle state and switching state. Idle state represents that the parent node of the cluster has no any operations currently, when the parent node of this cluster turn its state into switching state after receive a handover request which send by its admissibility switch cluster. When the switching operation is completed, this parent node will turn into idle state again. As shown in Figure 3.

Similar to the parent of the cluster whose average MUS belongs to the third level, ALBS-SCS divides the admissibility switch cluster into three states: idle state, pre-switching state and switching state. Idle state represents the parent node of the cluster has no any operation; you can receive a handover request which sent by the parent of the cluster whose average MUS is at the third level. When the admissibility switch cluster receives the handover request and determine the switching operation can take place, the state will be converted into pre-switching state [13]. If the admissibility switch cluster receives the confirmation feedback handover message which is send by the parent of the cluster whose average MUS is at third level, the pre-switching state is converted into switching states. Finally, when the switching operation is completed, the admissibility switch cluster of state converted into an idle state. As shown in Figure 4.

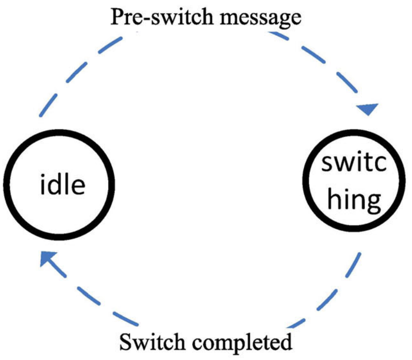

Figure 3. State diagram of the parent node of the cluster who's average MUS belongs to third level

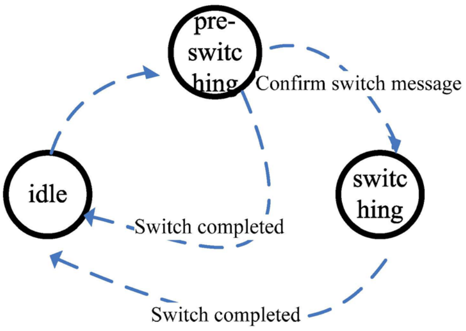

Figure 4. State diagram of admissibility switch cluster

Based on the number of the of cluster whose average MUS belongs to third level and the number of admissibility switch cluster defied in definition 5, we classify the ALBS-SCS core regulatory mechanisms into five scenarios as follows [16].

Scene 1. The number of admissibility switch cluster is 0.

When the number of admissibility switch cluster is 0, regardless of the number of cluster whose average MUS is at third level, the results are the same. As shown in Figure 5, the average MUS of the cluster led by v_3 is at the third level and the average MUS of the cluster led by v_2 is the same as the cluster whose average MUS is at third level. When v_3 and v_2 sends a handover request to the surrounding cluster, v_3 and v_2 will not receive any pre-switching command that the surrounding cluster sent, so their own state cannot be converted into a state of switching state and the corresponding switch will not operate. Over a period of cycle v_3 and v_2 transform into idle state, and when the next cycle comes, v_3 and v_2 detect whether there is the conditions of admissibility switch cluster or not. When the number of admissibility switch cluster is 0, no matter how many clusters whose average MUS is at third level, switching operation is permitted. ALBS-SCS adjustment mechanisms are the same, so the number of admissibility switch cluster is zero, ALBS-SCS adjustment mechanism is no longer based on the number of clusters whose average MUS is at third level.

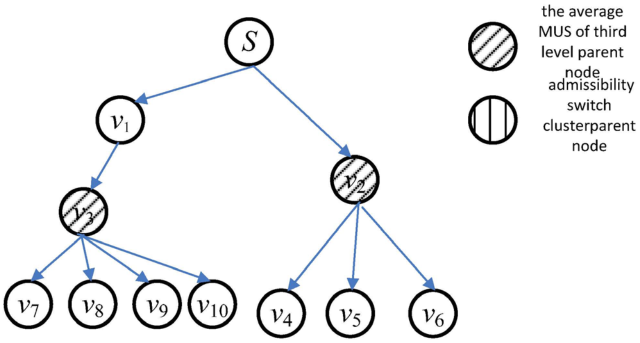

Figure 5. Diagram of multicast tree in scene 1

Scene 2. The number of clusters whose average MUS is at third level cluster is 1, and the number of admissibility switch cluster is also 1.

As shown in Figure 2, in the definition 5 we assume that the average MUS of the cluster which is led by v_3 is at the third level, and there is only one admissibility switch cluster which is led by v_6. Therefore, when v_3 sends a handover request message to v_6 which is in the idle state, v_6 will send pre-handover message to v_3 and meanwhile it will convert into the pre-state switching state (prevent conflict cause by received handover switching request message by other nodes). After v_3 receives the pre-shift feedback message, it will send a handover confirmation message to v_6, and its own state is converted into the switched state. Then according to the analysis in definition 5, ALBS-SCS will choose v_7 as the switch node. After v_6 receives a confirmation switching message send by v_3, its own state is converted into the switched state, at the same time v_6 process the switching operation. After the first step to switch operation, v_3 and v_6 sends their average MUS to RP, finally the average MUS of cluster led by v_3 and v_6 are updated and saved by the RP. After completion of the switching, the minimum average MUS of the cluster led by v_3 is 60 and the minimum average MUS of the cluster led by v_6 is 70. v_3 and v_6 are at least at the second level, both meet the equilibrium condition. Handover complete schematic is shown in Figure 6. After v_3 switches its child node v_7, it sends minimum MUS of the cluster to the RP and RP node discover that the average MUS of the cluster led by v_3 is still at the third level. ALBS-SCS will choose v_8 continue the switching operation. Until the average MUS of the cluster led by v_3 is at least at the second level, ALBS-SCS will stop adjustment.

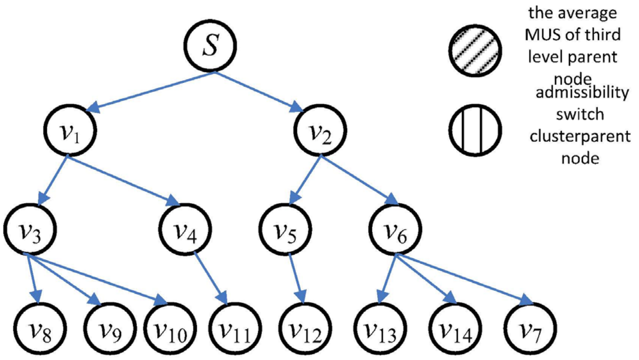

Figure 6. Diagram of multicast tree in scene 2

Scene 3. The number of cluster whose average MUS belongs to third level is 1, and the number of admissibility switch cluster is more than one.

In Figure 2, if the evaluation of MUS of the cluster led by v_{12} is 90, when v_3 sends a handover request. The cluster led by v_5 and v_6 judges whether their own minimum MUS are still more than 60 after receive switching or not. So the cluster led by v_5 and v_6 are the admissibility switch cluster of the cluster led by v_3. As is shown in Figure 7, when v_5 and v_6 receive a handover request that v_3 sends, v_5 and v_6 will send pre-handover feedback message to v_3 just as Scene 2. v_3 based on the timestamps received from v_5 and v_6 and selects an earlier one (assuming that v_5 qualifies). Then v_5 will switch with v_3 as the same in scene 2. However, the state of v_3 has been converted into switching state from idle state at the time when v_6 sends pre-handover commands to v_3, so it will not send a confirmation switching message to v_6. After a period of time v_6 hasn't receive the feedback confirmation switching message from v_3, the switching operation won't continue. The state of v_6 has been converted into idle state from pre-switching state again.

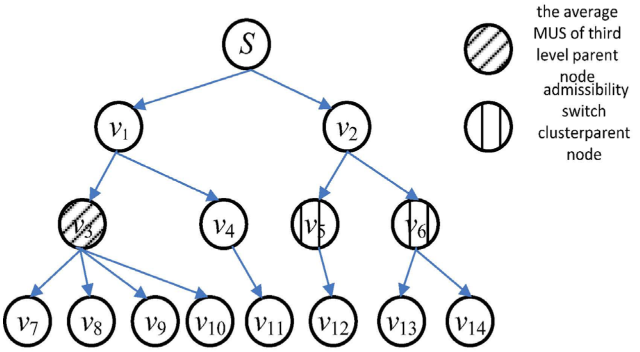

Figure 7. Diagram of Multicast Tree in Scene 3

Scene 4. The number of clusters whose average MUS belongs to the third level is more than one, and the number of admissibility switch cluster is more than one.

In Figure 8, the average MUS of cluster which is led by v_3 and v_4 are at the third level, the cluster led by v_5 and v_6 are the admissibility switch clusters of the cluster led by v_3 and v_4. When v_3 and v_4 sends a request switching message to v_5 and v_6, v_5 and v_6 will select the node with a smaller timestamp. Assuming that v_3 first sends a handover request message to v_5 and v_6, the state of v_5 and v_6 has been converted into a pre-switching state from idle state after received the handover request message, and then send the pre-handover information back to v_3. v_3 will choose the node with a smaller timestamp based on the feedback timestamp that v_5 and v_6 has send. Assuming that v_3 first receives the message from v_5, then v_3 converts its own state into switching state from idle state, and will continue switching operation with v_5 as the same in Scene 2. When v_6 cannot receive a confirm message it will convert its state into idle state as the same in the Scene 3. Similarly, when v_4 sends a handover request to v_5 and v_6, it will find that v_5 is in pre-shift state and v_6 is in idle state. The node v_4 will switch with v_6 as the same in Scene 2.

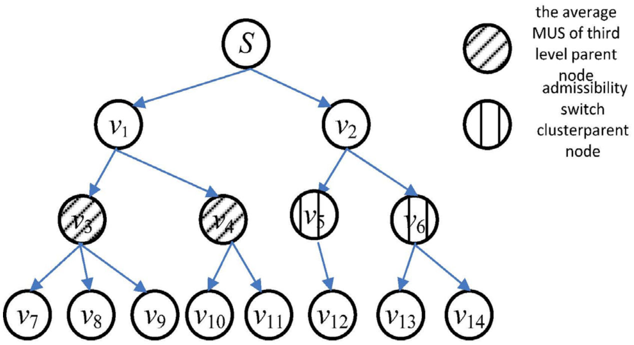

Figure 8. Diagram of multicast tree in scene 4

Scene 5. The number of the cluster whose average MUS belongs to the third level is more than one, and the number of admissibility switch cluster is 1.

As shown in Figure 9, the average MUS of cluster led by v_3 and v_4 are at the third level, the cluster led by v_5 is the only admissibility switch cluster of the cluster led by v_3 and v_4. So v_5 use the timestamp as a parameter index to select v_3 and v_4 switching operation, which is on the contrary of Scene 3.

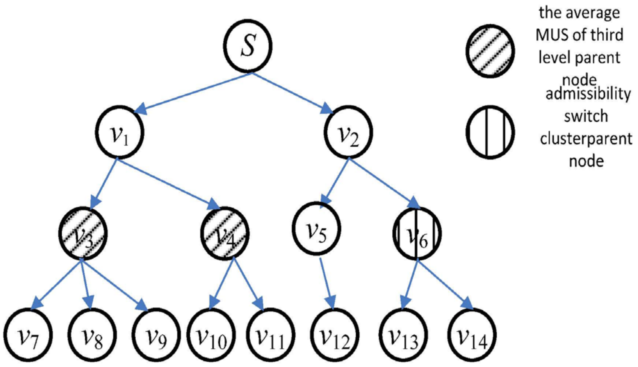

Figure 9. Diagram of Multicast Tree in Scene 5

4 Performance Simulations and Analysis

4.1 Experimental Environment Configuration

The experiment compares the performances of ALBS-SCS that put forward by this paper with NICE. In this paper, hardware environment of the simulation is as follows. Dual-core processor is Inter (R) Pentium (R), the memory is 3.4 G, and the operating system is 32-bit Window7. This paper uses the network simulation environment of OMNeT4.1++ [20], and the overlay network simulation framework is INET-OverSim-20101019 [21]. Bottom network model is Simple network model and the bottom of the generator node is oversim.common.NoChurn. The underlying network coverage is OverSim.Overlay.AlbsModules. The simulation schematic ALBS-SCS is shown in Figure 10.

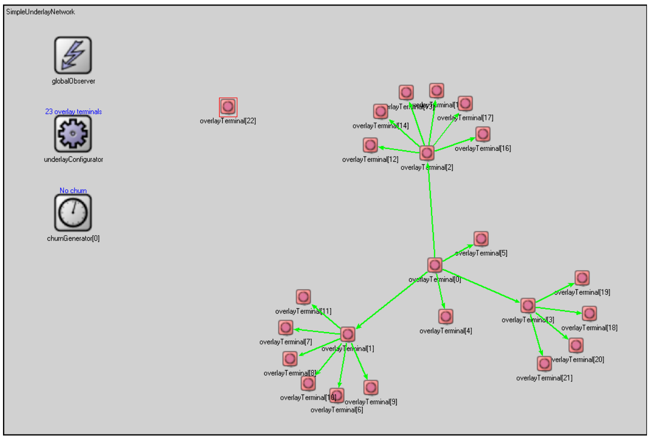

Figure 10. Simulation Schematic of ALBS-SCS

4.2 Performance Metrics [22]

The outdegree of the node [15]. The outdegree of a node is the number of its children nodes that the node needs to carry, it also reflects the distribution of the nodes in the whole system.

Average transmission delay. ATD (Average Transmission Delay) is the ratio of sum (TD_i) and N, in which sum(TD_i) is the total transmission time the request packet traveled between the sending node and the receiving node. To be specific, the request packet is sent by the sending node in the multicast system to the server and received by the receiving node after the transmission of each father node until it reaches the receiving node. For another, N is the number of the nodes. The calculation of ATD is described in Formula 9.

$$ATD = \frac{\sum_{i=0}^{N-1} TD_i}{N} \tag{9}$$

Average link pressure. In the multicast tree system, ALP, average link pressure is defined as the average time that a packet needs to be copied from the sending server node to the destination node in one transmission process.

Average multicast user satisfaction. AMUS reflects the overall performance of the whole system, including the average bandwidth, channel capacity and QoS. It is calculated as Formula 10.

$$AMUS = \frac{\sum_{i=0}^{N-1} MUS_i}{N} \quad \textbf{(10)}$$

4.3 The Analysis of Experimental Results

The statistics of the simulate results use the number of the nodes when it reaches 16, 32, 64, 128, 256 as examples to analysis the performance of ATD,ALP and AMUS.

As shown in Figure 11, when the scale of the node is 16, the AMUS of NICE and ALBS-SCS are approximately reach 65. When the node scale is increased, the AMUS of NICE Protocol is decreased significantly. However, the AMUS of ALBS-SCS has remained at about 60. The result shows that NICE is not good at adjusting the system according to the quality of the multicast system when it has large-scale nodes. As a result, hot spots are arisen, while the AMUS and the whole system are reduced. On the other hand, ALBS-SCS considered the performance of the multicast network, adjusted the nodes with poor performance, therefore hot spots are avoided. As a result, the AMUS is fixed at 60 and the performance of the system is kept stable.

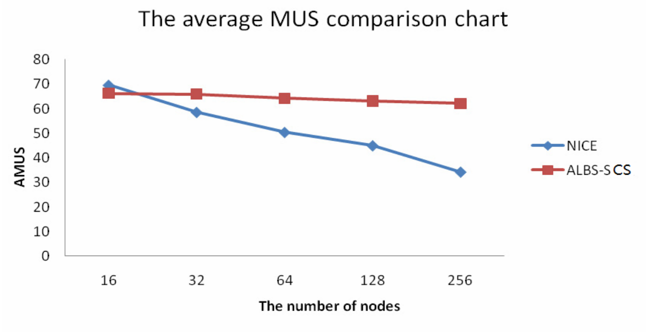

Figure 11. Node's average MUS comparison

The percentage of nodes' outdegree is showed in Figure 12. When the number of nodes reaches 256, there are almost 80% nodes whose outdegree is 0 in both NICE and ALBS-SCS (these nodes are the leaf nodes). In NICE, about 6% nodes' outdegree is 4, about 5% nodes' is 5 or 6, and about 9% nodes' is 7 or 8. But in ALBS-SCS, according to the average MUS of current network fed back by nodes, it has made some adjustment, so none of the nodes' outdegree is very small or large. It has balanced the whole system's performance.

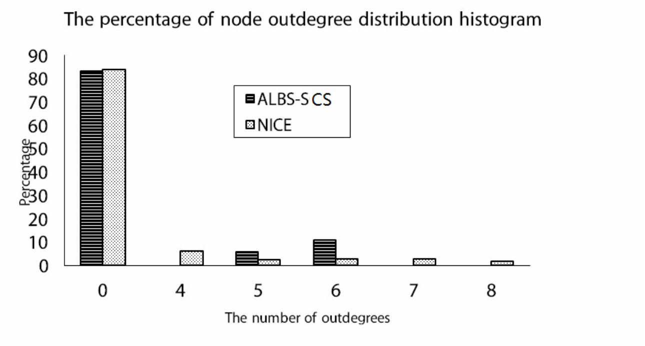

Figure 12. Histogram of percentage of node outdegree distribution

The average transmission delay is shown in the Figure 13. When there are only 16 nodes in the system, the ATD of NICE and ALBS-SCS are also at a low degree (about 15ms), but as the number of node increasing, the increasing speed of the ATD of NICE is faster than ALBS-SCS's obviously. When there are 256 nodes in the systems, the ATD of NICE is about 140ms, and the ALB-SCS's is 100ms. Because as the number of node increasing, the depth of the tree of the nodes is also increasing obviously and the ATD is decided by serious of the transmission by parent node, so that the ATD is increasing rapidly. The ALBS-SCS is adjusted by the average MUS, so that the node can choose the better parent node to join in, and the system can be improved.

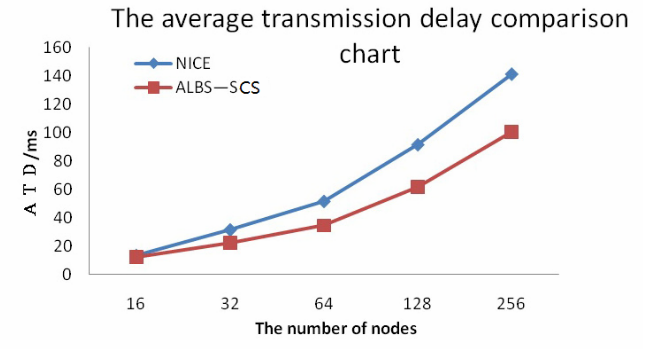

Figure 13. Average transmission delay comparison

As node's average link stress is shown in Figure 14, when the node scale is 16 and 32, the link pressure of NICE and ALBS-SCS are both about 1.5. Because when the network size is small, the NICE model is easy to cause the local optimal solution. But with the increase of nodes scale, the increase in average link stress of NICE is apparently greater than ALBS-SCS. It can be seen that ALBS-SCS is better at controlling the link pressure compared than NICE. Because ALBS-SCS system adjusts the balance of the nodes outdegree according to the average MUS, it will effectively decrease the height of the tree, thereby reduces the number of data packets forwarded and the average link pressure.

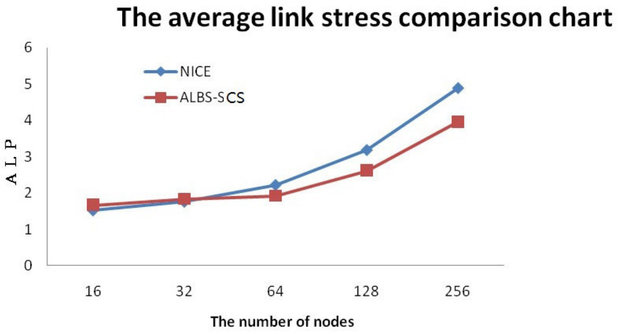

Figure 14. Average link stress comparison

5 Conclusion and Future Work

First of all, this paper analyzes the current situation of ALM, due to diversity in hardware devices and software applications, as well as communication environment and other factors, resulting in an imbalance of the overall system utilization. In view of this characteristic, we weighted the service of quality, average bandwidth and channel capacity, this paper puts forward the concept of system average MUS. Secondly, adjusted the system end nodes according to the average system MUS, we proposed ALBS-SCS algorithm. Finally, through the simulation software OMNET4.1++ we experiment on ALBS-SCS. Comparison with the classical protocol NICE, we verified that ALBS-SCS can reduce the average transmission delay and average link stress better than NICE, and increasing system Average MUS, improving overall performance and user experience.

Of course, ALBS-SCS mechanism has shortcomings, mainly because the value of the average MUS is mainly composed of service quality, the average bandwidth and channel capacity indicators. In the future research if we add other performance indicators, the value of average MUS maybe reflect a more comprehensive and closer to the actual situation. Meanwhile node can accept switch cluster mainly according to the value of average MUS in the regulatory mechanisms. If we add different node communications business in the future research, the effect of the adjustment mechanism will be better. Meanwhile, the next steps we will study how to combine nodes' other performance indicators and nodes' relative leaving probability, and how to recovery new leaders after the old leaders leave from the multicast tree.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 61370108, 61672257, 61272112, 61170017) and State Scholarship Fund of China (No. 201706775029).

References

[1] J. F. Kurose, K. W. Ross, *Computer Networking: A Top-Down Approach Featuring the Internet*, Pearson/Addison Wesley, 2005.

[2] Y. Lin, S. R. Cheng, Q. Li, Peer Network, *ZTE Technology*, Vol. 12, No. 1, pp. 57-60, February, 2006.

[3] P. Suthar, V. Swami, R. Tamboli, C. Kshirsagar, S. Hirve, Load Balancing In Structured Peer To Peer Network Using DSLS and ASH Algorithm, *International Journal of Advanced Research in Computer Science and Software Engineering*, Vol. 3, No.4, pp. 284-287, April, 2013.

[4] R. K. Singh, N. S. Chaudhari, K. Saxena, Load Balancing in IP/MPLS Networks: A Survey, *Communications & Network*, Vol. 4, No.2, pp. 151-156, May, 2012.

[5] N. Xiong, X. Jia, L. T. Yang, A. V. Vasilakos, Y. Li, Y. Pan, A Distributed Efficient Flow Control Scheme for Multirate Multicast Networks, *IEEE Transactions on Parallel & Distributed Systems*, Vol. 21, No. 9, pp. 1254-1266, September, 2010.

[6] M. Mitzenmacher, The Power of Two Choices in Randomized Load Balancing, *IEEE Transactions on Parallel & Distributed Systems*, Vol. 12, No. 10, pp. 1094-1104, October, 2001.

[7] B. Godfrey, K. Lakshminarayanan, S. Surana, R. Karp, I. Stoica, Load Balancing in Dynamic Structured P2P Systems, *Joint Conference of the IEEE Computer and Communications Societies*, Hong Kong, China, 2004, pp. 2253-2262.

[8] D. R. Karger, M. Ruhl, Simple Efficient Load-Balancing Algorithms for Peer-to-Peer Systems, *Peer-to-Peer Systems III*, La Jolla, CA, 2004, pp. 131-140.

[9] P. Ganesan, M. Bawa, H. Garcia-Molina, Online Balancing of Range-Partitioned Data with Applications to Peer-to-Peer Systems, *Proceedings of the 30th International Conference on Very Large Data Bases - Volume 30*, Toronto, Canada, 2004, pp. 444-455.

[10] Q. H. Vu, B. C. Ooi, M. Rinard, Histogram-based Global Load Balancing in Structured Peer-to-Peer Systems, *IEEE Transactions on Knowledge & Data Engineering*, Vol. 21, No.4, pp. 595-608, April, 2009.

[11] J. Cui, K. Gao, Y. J. Ye, J. Fan, L. Wu, Y. Yang, Multicast Tree Construction: A Nodes' Relative Leaving Probability Algorithm based on NICE in ALM, *International Journal of Autonomous & Adaptive Communications Systems*, Vol. 9, No. 2, pp. 4-19, March, 2016.

[12] J. Cui, N. N. Xiong, L. Wu, K. Jia, A Novel Source Path Topology Map Construction Method in ALM. *Journal of Internet Technology*, Vol. 16, No. 6, pp. 1015-1022, November, 2015.

[13] J. Cui, W. Zhang, F. Huang, L. Wu, Hierarchical Adaptive Recovery Algorithm in Mobile ALM, In S. Zhang, K. Xu, M. Xu, J. Wu, C. Wu, Y. Zhong (Eds.), *Frontiers in Internet Technologies*, Springer-Verlag Berlin Heidelberg, 2015, pp. 95-105.

[14] J. Cui, N. Xiong, J. H. Park, K. Jia, L. Wu, A Novel and Efficient Source-path Discovery and Maintenance Method for

Application Layer Multicast, *Computers & Electrical Engineering*, Vol. 39, No. 1, pp. 67-75, January, 2013.

[15] N. Xiong, A. V. Vasilakos, L. T. Yang, L. Song, Y. Pan, R. Kannan, Y. Li, Comparative Analysis of Quality of Service and Memory Usage for Adaptive Failure Detectors in Healthcare Systems, *IEEE Journal on Selected Areas in Communications*, Vol. 27, No.4, pp. 495-509, May, 2009.

[16] M. Hara, T. Yoshihiro, Adaptive Load Balancing Based on IP Fast Reroute to Avoid Congestion Hot-Spots, *IEEE International Conference on Communications*, Kyoto, Japan, 2011, pp. 1-5.

[17] W. Quan, Y. Liu, W. Jiang, J. Guan, Intelligent Popularity-aware Content Caching and Retrieving in Highway Vehicular Networks, *EURASIP Journal on Wireless Communications and Networking*, Vol. 2016, No. 1, pp. 200, August, 2016.

[18] M. Antic, A. Smiljanic, Routing with Load Balancing: Increasing the Guaranteed Node Traffics, *IEEE Communications Letters*, Vol. 13, No. 6, pp. 450-452, June, 2009.

[19] H. Zhang, P. Dong, W. Quan, B. Hu, Promoting Efficient Communications for High-speed Railway Using Smart Collaborative Networking, *IEEE Wireless Communications*, Vol. 22, No. 6, pp. 92-97, December, 2015.

[20] A. Varga, Using the OMNeT++ Discrete Event Simulation System in Education, *IEEE Transactions on Education*, Vol. 42, No. 4, pp. 11, November, 1999.

[21] I. Baumgart B. Heep, S. Krause, OverSim: A Flexible Overlay Network Simulation Framework, *Proceedings of 10th IEEE Global Internet Symposium (GI 2007) in Conjunction with IEEE INFOCOM 2007*, Anchorage, AK, 2007, pp. 79-84.

[22] J. Cui, N. Xiong, L. Wu, K. Jia, K. Gao, Design and Analysis of the Gateway-Level Topology Map in Topology-Aware ALM Systems, *Proceedings of the 2013 IEEE International Conference on Systems, Man, and Cybernetics*, Manchester, UK, 2013, pp. 4409-4414.

Biographies

Jianqun Cui received her Ph.D. degree in Computer Science at Wuhan University. Now she is a professor at Central China Normal University. Her current research interests include computer network, high-performance computing and application layer multicast.

Liang Ma received his M.S. degree in Computer Science from the Central China Normal University in the year of 2015. Now, he is working at the NO.1 Middle School affiliated to Central China Normal University. At present, his work is about the construction of information technology in middle school.

Kuan Gao received his M.S. degree in Computer Science from the Central China Normal University in the year of 2014. His current research interests include computer network, high-performance computing and application layer multicast.

Libing Wu received Ph.D. degree in Computer Science from Wuhan University. Now he is a professor at Wuhan University. His current research interests include computer network, Grid computing and Cloud computing.

Yi Yang received her Ph.D. degree in Computer Science and Engineering from the Pennsylvania State University in the year of 2010. Now, she is a tenure-track assistant professor at Fontbonne University in St Louis, MO. She is also the director of BS in cyber security at Fontbonne University. Her research interests are computer/network security and network management.