Multicast middleware for performance and topology analysis of multimedia grids

Jerry Z. Xie[1,2], *Shaoli Chen*[2], *Zongkai Yang*[1]

[1]*National Engineering Research Centre for E-Learning, Central China Normal University, No. 152 Luoyu Road, Wuhan 430079, People's Republic of China*
[2]*ECCOM Network System Co., Ltd., 8F South Tower, No. 9 Lujiazui Software Park, No. 20, Lane 91, E'shan Road, Shanghai 200127, People's Republic of China*
E-mail: jerryxie@eccom.com.cn

Published in *The Journal of Engineering*; Received on 9th March 2017; Accepted on 29th March 2017

Abstract: Since multicast reduces bandwidth consumption in multimedia grid computing, the middleware for monitoring the performance and topology of multicast communications is important to the design and management of multimedia grid applications. However, the current middleware technologies for multicast performance monitoring are still far from attaining the level of maturity and there lacks consistent approaches to obtain the evaluation data for multicast. In this study, to serve a clear guide for the design and implementation of the multicast middleware, two algorithms are developed for organising all constituents in multicast communications and analysing the multicast performance in two topologies – 'multicast distribution tree' and 'clusters distribution', and a definitive set of corresponding metrics that are comprehensive yet viable for evaluating multicast communications are also presented. Instead of using the inference data from unicast measurements, in the proposed middleware, the measuring data of multicast traffic are obtained directly from multicast protocols in real time. Moreover, this study makes a middleware implementation which is integrated into a real access grid multicast communication infrastructure. The results of the implementation demonstrate the substantial improvements in the accuracy and real time in evaluating the performance and topology of multicast network.

1 Introduction

This paper is a research result from the National Engineering Research Centre for E-Learning (which was founded in 2004, and has devoted to the research of multimedia network education for over 10 years). During the research process, it is found that although the performance of multimedia grids is crucial for the user experience of multimedia grid applications, there still lacks satisfactory tools to realise the collection and analysis of the multimedia grid performance data. For this reason, as shown in Fig. 1, it is necessary to add a middle layer between the 'lower-layer multimedia grid cloud infrastructure' and the 'upper-layer multimedia grid applications'. Such layer should have two main functions: (i) Performance monitoring, which is able to collect and evaluate the reliable performance data of multimedia grids in real time; (ii) Topology optimisation, which is in charge of analysing and improving the current topology of multimedia grids.

Multimedia grid applications require the transfer of high-bandwidth data streams within the grid and to a large number of users in a distributed environment. Since the traditional Internet protocols could not keep up the pace with the increasing demand of bandwidth consumption, this has motivated computer scientists to design more efficient protocols for managing and using the network bandwidth, and the multicast is one flexible and inexpensive solution. As a virtual network, multicast backbone (MBone) [1] has been in existence since 1992. MBone is being used to develop protocols and applications for collaborative work, which provide one-to-many and many-to-many services for communications simultaneously over heterogeneous networks. Today, multicast becomes a part of the standard TCP/IP protocol suite [2]. Therefore, the motivation of this paper is to develop a middleware which has the capability to monitor and analyse the performance and topology of multimedia grids in multicast environment.

For the collection and analysis of multimedia grid performance/topology data in the multicast environment, one of the earliest multicast tools is mrinfo [3]. Mrinfo gathers information of multicast routers and determines which neighbouring routers are equivalent to a router. If a multicast router is queried by mrinfo, the version number of the router, the list of neighbouring routers, and the pertinent information (such as the metrics, thresholds and flags) about each interface will be displayed, therefore it can be obtained the current status of multicast routers and tunnels fed by that router.

After the mrinfo, many studies focused on the design of multicast middleware. Wang *et al*. [4] investigated the scaling law for multicast traffic with hierarchical cooperation, and proposed a new class of scheduling policies for multicast traffic. Xing and Qu [5] investigated the optimisation of the network coding-based multicast routing problem with respect to two widely considered objectives: the cost and the delay. Xu and Qu [6] investigated the first hybrid scatter search and path relinking metaheuristic for the delay-constrained least-cost multicast routing problem. Patel *et al*. [7] presented a swarming agent based intelligent algorithm using a hybrid ant colony optimisation/particle swarm optimisation technique to optimise the multicast tree. Ikeda *et al.* [8] presented a new middleware for supporting development and performance evaluation of application layer multicast protocols on real environments. Xu *et al.* [9] presented a new hybrid evolutionary algorithm to solve multi-objective multicast routing problems in telecommunication networks. Jiang *et al.* [10] provided an end-to-end middleware for smart TV live multicast systems, which could recover lost packets completely with few computational resources and very little impact on users' real-time experience. Yu *et al.* [11] proposed an enhanced round-robin multicast scheduling algorithm with a function of searching deeper into queues to reduce the head-of-line blocking problem and thereby the multicast latency. Ghasvari *et al.* [12] considered the problem of finding a minimum cost multicast sub-graph based on network coding, where delay values associated with each link, limited buffer-size of the intermediate nodes and link capacity variations over time were taken into account.

However, the current middleware technologies for multicast performance/topology data measurement are still far from attaining the level of maturity, such technologies have several deficiencies in common:

(i) Inaccuracy: The techs cannot obtain the measuring data from a multicast session directly, they are based largely on the measuring methods for the end-to-end delay or loss rate of unicast

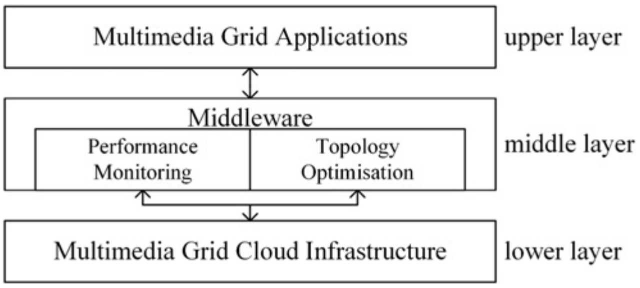

Fig. 1 *Three-layers structure of multimedia grids service*

communications. Multicast and unicast have different communication principle [13], thus, by using the techs, the accuracy of measuring results cannot be guaranteed. However, the accuracy of multicast performance data as well as the insight of the multicast topology are important to the better design of multimedia grid applications.

(ii) Inconsistency: There is no single model that provides a definitive set of metrics which can describe the multicast communications in a clear and consistent manner.

(iii) Non-real time: The performance data collected are usually later than the fact of nature, therefore it is difficult to response the performance requirements immediately.

To make up for the deficiencies above, in this paper, two algorithms are outlined for developing the multicast middleware which allows obtaining logical topologies of multicast communications dynamically and collecting performance data directly from multicast protocols in real time, therefore, the immediate response to the multicast events can be acted to improve the performance of multimedia grid applications. Meanwhile, a set of metrics that are comprehensive yet viable for implementing this middleware are also presented. The metrics are able to measure both the multicast physical limitations and the quality of connectivity, which are critical to the manageability of multimedia grid applications. Moreover, an implementation of the multicast middleware (with a corresponding user interface) which demonstrates the viability and quality of the algorithms and metrics with real world data is presented, and the implementation is integrated into the access grid infrastructure.

The rest of this paper is organised as follows. Section 2 explains the detailed theoretical methods for the development of multicast middleware. In Section 3, an implementation of the proposed method is realised. Section 4 carries out some discussions of the whole method. Finally, the conclusion is presented in Section 5.

2 Methodology

This section first develops algorithms for analysing network performance in two topologies dynamically and interactively – ‘multicast distribution tree’ and ‘clusters distribution’. The algorithms can contribute to construct a multicast middleware which is able to organise all existing multicast users or communities as well as collect and respond accurate multicast performance data in real time. After that, to support this middleware, a set of metrics which are suitable for multicast performance measurement are also defined.

2.1 Multicast logical topologies analysis methods

2.1.1 Multicast distribution tree analysis method: Generally, in a multicast network, a multicast distribution tree is constructed to control the paths which multicast traffic takes through the network in order to deliver traffic to all receivers [14]. A suitable distribution tree reduces the number of copies of packets transmitted in the network and improves the efficiency of bandwidth usage, especially for multimedia grid applications involving large volumes of data. Therefore, analysis of a multicast distribution tree can provide users and service providers with basic information about routing, network traffic, and bottlenecks [15].

Multicast-capable routers create distribution trees to control the path that multicast traffic takes through. One of the useful distribution tree topologies for multicast performance analysis is the source-rooted tree. This topology can find the shortest path from a source to each receiver. Based on the source-rooted tree [16], the minimum-cost spanning tree for a selected source node and a selected multicast group can be computed before starting or during a multicast service. Therefore, in this paper, the multicast distribution tree is constructed by a two-step procedure as follows:

Step 1: A connected graph is constructed as a rich mesh. Then, to improve the quality of the mesh, two metrics thresholds (which will be introduced in Section 2.2) – the spatial metric and the connective metric, are used.

Step 2: Based on the qualified mesh, a minimum-cost spanning tree can be constructed (using the following pseudo code) at the selected source root, using well-known algorithms such as Prim’s algorithm (see, Fig. 2).

For example, as shown in Fig. 3, a group of seven users in an access grid are selected, then the multicast beacon on the fly is

```
Graph G; Heap S;
ComputeMesh(G,S);
S.sort();
while (!S.empty()) {
  u = S.deletemin();
  T.join(G.left(),G.right());
  for (e in G) {
    v = G.mate(e);
    if (S.member(v) and G.w(e) < S.key(v)) {
      S.changekey(v,G.w(e));
      cheap[v] = e;
  }
    elif (!S.member(v) and T.first(v) = Null) {
      S.insert(v,G.w(e));
      cheap[v] = e;
    }
   }
  }
```

Fig. 2 *Prim’s algorithm*

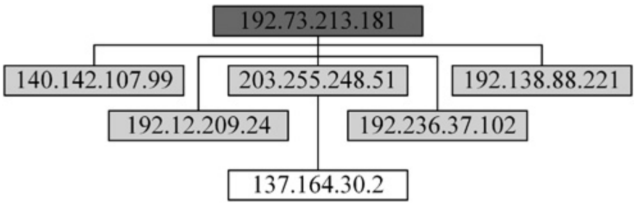

Fig. 3 *Minimum-cost spanning tree for selected multicast group and selected source root (IP address: 192.73.213.181)*

collected and a minimum-cost spanning tree is computed for this small-multicast session. The dark-grey node is the root node which sends data to the other members (receivers) of this group; five users (light-grey nodes) have a direct link to the root; the remaining node (at the lowest layer) has the shortest path from the root through one node (203.355.248.51) in the middle layer. Before this multicast session is started, the middleware can send strong positive feedback to the source root to confirm this small group have a good connectivity and indicate the distribution of the multicast data streams, also, these data streams can be used as an application-level routing map.

2.1.2 Cluster distribution analysis method: In the multicast control process, a general question is how to select the appropriate algorithm to organise multicast users into a meaningful structure. To answer this question, three characteristics of multicast communities are distinguished:

(i) Hierarchical: From source to final receivers, data streams are always transferred through several routers and network layers. An optimal multicast tree usually has multiple levels of hierarchy.
(ii) Local: Because of geographical factors, users in the same local area seem to have very good connectivity and share some common characteristics.
(iii) Dynamic: Each distributed user participates in or leaves a multicast session freely.

Clustering algorithms are used to categorise data into clusters such that objects are grouped in the same cluster when they are similar according to specific metrics [17]. They are helpful in developing a tool for logical topology analysis of multimedia grids. In general, there are two main types of clustering algorithms: hierarchical and non-hierarchical [18]. Hierarchical algorithms join objects into successively larger clusters by using some measure of similarity or distance. An example is the hierarchical tree. Non-hierarchical algorithms partition objects into a given number of clusters as distinct as possible. An example is the *k*-means clustering [19]. In this paper, these two types are combined into a hierarchical *k*-means clustering algorithm (HKC, as the following pseudo code illustrates). In this algorithm, a hierarchical tree is formed by performing *k*-means clustering at each hierarchical level and choosing a representative member to act as a server for each of the k clusters found. Each cluster may also be partitioned by *k*-means to form a new layer of sub-clusters, whose servers become children of the server in the top-level cluster just partitioned. The no server children cannot be partitioned and become leaf nodes in the tree. The process is repeated until some criterion is reached. Servers in each level receive multicast traffic from their parent and pass them on to their children.

```
def hkc(nodes)
clusterSize = n;
seeds = initializeSeeds(nodes,n);
clusters = k_means(nodes, seeds, n);
for cluster in clusters
if length(cluster) < n then
hkc(cluster)
else
output(cluster)
output(clusters)
```

Since the hierarchical clustering algorithm tends to produce well-spaced clusters and can reflect the dataset under consideration, this method can be used to select the initial seeds of *k*-means. At each iteration, two nearest, or least-dissimilar, clusters are merged into one until k clusters are found. The average-linkage are used to compute the dissimilarity $D(I, J)$ between cluster I and J as follows:

$$D(I, J) = \frac{\sum_{i\in I, j\in T} d(i, j)}{|I| \times |J|} \quad (1)$$

where $d(x, y)$ is the dissimilarity between node x and y, $|X|$ is the cardinality of cluster X. At each iteration of *k*-means, the new set of cluster centroids are computed, and nodes are reassigned to these centroids. In this process, the following algorithm is used for cluster S:

$$\text{diss}(i, S) = \frac{\sum_{j\in S} d(i, j)}{|S|} \quad (2)$$

As an example, all the active users in a sample access grid are selected for a multicast group. The maximum number of nodes in each cluster is set at six. The cluster topology is presented in Fig. 4. The centre circle is a first-level cluster, and the clusters around it are second-level clusters. This figure gives a brief idea of where the network servers should be put in each cluster. Specifically, to avoid or reduce the extra delay and bandwidth cost, each server could be embedded into each level according to the hierarchical cluster distribution.

2.2 Multicast metrics

Various multimedia grid applications have different criteria for network transmission. Therefore, making a one-size-fits-all metric for multicast is impossible in reality. An alternative and flexible solution is to leave the selection of metrics to the users while providing users a set of comprehensive metric components. In this paper, the following two aspects are taken into consideration when designing the metrics.

(i) Multicast router: Most multicast routers are of the drop-tail type. This type of router uses a first-in first-out buffer to store arriving packets and to drop the arriving packets if the buffer is already full. Because such router has the nice properties of Kleinrock's classical model [20], some measurement parameters that work for unicast are still useful for multicast traffic. Thus, the comprehensive multicast metrics could

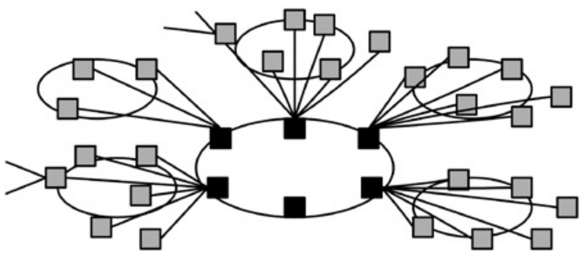

Fig. 4 *Hierarchical cluster distribution for a sample access grid*

include several parameters that are also used to measure unicast traffic.
(ii) Multicast congestion control: Main objective of the multicast congestion control algorithm is to balance the traffic to increase the network throughput [21]. Considerable research efforts have focused on the design of multicast congestion control, which provide high performance and scalability [22]. The multicast metrics should be able to evaluate the fairness for different multicast congestion control protocols and mechanisms.

For illustration, in this paper, a network is modelled as a weighted digraph $G = (V, E)$, where V denotes the set of receiver nodes and E denotes the set of arcs. The arc between each pair of nodes includes the set of network communication links and routing devices, such as routers, switches, and reflectors. The metrics designed in this paper are divided into two categories: local and group. They are defined as follows.

2.2.1 Local metrics

Definition 1: A local multicast metric function $f_M: E \to R^+$ assigns a nonnegative number to the link of two nodes in the multicast group. A local metric comprises two basic components: a spatial metric f_S and a connective metric f_C

$$f_M = a \times f_S + (1 - a) \times f_C, \quad 0 \le a \le 1 \tag{3}$$

Definition 2: A spatial metric indicates the physical distance between two nodes, it is defined as

$$f_S(v_1, v_2) = \text{RTT}, \quad v_1, v_2 \in V, \; 0 < \text{RTT} \le \text{RTT threshold} \tag{4}$$

where RTT is the round-trip time (i.e. the actual hops and delay cost). It reveals the physical distance and geographical relationship between two nodes. In unicast networks, RTT is the most important measurement factor [23]. In multicast networks, the higher bandwidth cannot reduce the delay time between a pair of nodes; however, it always attracts more streams than lower bandwidth, therefore, in a multicast topology, RTT is still the principal performance factor.

Definition 3: A connective metric indicates the quality of connectivity between a pair of nodes, it is defined as

$$\begin{aligned} &f_C(v_1, v_2) = f(\text{loss, jitter, order, dup}),\\ &v_1, v_2 \in V, \; 0 < f(\text{loss, jitter, order, dup}) \le f(\text{threshold}) \end{aligned} \tag{5}$$

where $f_C(v_1, v_2)$ includes one or more connective parameters: loss rate, delay variance (jitter), out of order and duplicate rate.

While the spatial metric indicates the latency performance between two nodes, the connective metric quantifies the quality of service (QoS) of the link. These two metrics have different functionalities for different multimedia grid applications. Moreover, each connective metric may include one or more terms for various precision levels. For example, for a video conference, the weight of the spatial metric could be higher than that of the connective metric. Since QoS is not the main issue here, the connective metric may have only one term, such as loss rate. On the other hand, when requiring data from a distributed database, QoS issues become the most important. In that case, more consideration would be given to the connective terms, while the spatial metric could be removed if the network bandwidth satisfies some criterion. Thus, above metrics have a generality that makes them flexible for various multimedia grid applications.

2.2.2 Group metrics

Definition 4: The group metric of node x in a multicast group X comprises two components: latency and fairness. The latency metric computes the mean and variance of the local metric of each group member, while the fairness metric computes the fairness of them.

Latency metric:

$$\text{Mean}(f_S(x)) = \frac{\sum_{v \ne x, v \in X} f_S(x, v)}{n - 1} \tag{6}$$

$$\text{var}(f_S(x)) = \frac{\sum_{v \ne x, v \in X} \left(f_S(x, v) - \text{Mean}(f_S(x))\right)^2}{n - 1} \tag{7}$$

$$\text{Mean}(f_C(x)) = \frac{\sum_{v \ne x, v \in X} f_C(x, v)}{n - 1} \tag{8}$$

$$\text{var}(f_C(x)) = \frac{\sum_{v \ne x, v \in X} \left(f_C(x, v) - \text{Mean}(f_C(x))\right)^2}{n - 1} \tag{9}$$

Fairness metric:

$$F_S(x) = \frac{\left[\sum_{v \ne x, v \in X} f_S(x, v)\right]^2}{(n - 1) \times \sum_{v \ne x, v \in X} f_S^2(x, v)} \tag{10}$$

$$F_C(x) = \frac{\left[\sum_{v \ne x, v \in X} f_C(x, v)\right]^2}{(n - 1) \times \sum_{v \ne x, v \in X} f_C^2(x, v)} \tag{11}$$

where n is the number of nodes in the multicast group. If all nodes have the same local metric, the fairness is 1, which means the topology configuration is completely fair. As the disparity increases, the fairness decreases, and some nodes may have fairness near 0. For instance, for interactive traffic, such as real-time audio and video applications, the fairness metric is generally the key performance indicator [24]. In this case, the fairness should be based on equality.

3 Implementation

In this section, the proposed theoretical methods for multicast middleware will be implemented in an access grid to verify and demonstrate their viability and quality.

3.1 Scenarios

Fig. 5 illustrates the location and role of the multicast middleware for performance/topology monitoring on the sample access grid.

An access grid client can issue an inquiry about assessing multicast performance and the access to the access grids facilities is accomplished by using access grid RAT grid client tool [25]. Once the access grid virtual venue receives the inquiry, it will store the handle and related client information, and send a request to the beacon server for multicast beacon data. After the multicast beacon data is collected, the beacon server sends back the current statistics for live nodes to the venue. Then, the venue performs some initial data processing and calls the multicast middleware with the beacon data and the user's specific criteria. The middleware sends back the results of performance and topology analysis to the venue. Finally, the venue stores the results and returns the performance statistics and logical topology analysis to the client who issued the inquiry.

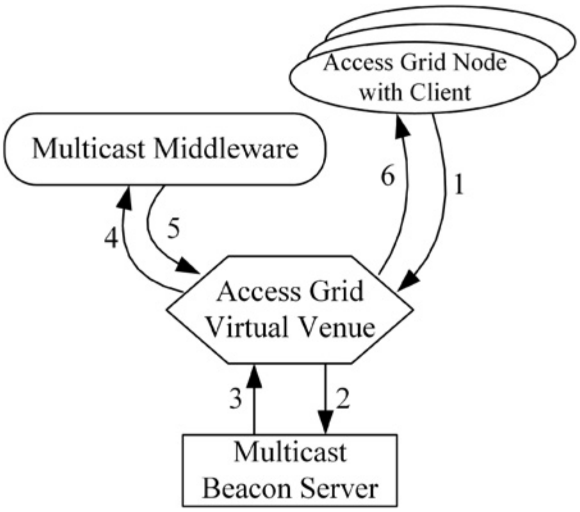

Fig. 5 *Middleware in access grid multicast communications*

3.2 Graphic user interface (GUI)

As shown in Fig. 6, a GUI is developed for the multicast middleware. It presents the results of performance and topology analysis for the access grid.

In Fig. 6, the 'Venue Client' is used to connect and participate in an access grid virtual venue. It displays the contents of the virtual venue, the connections to other venues, and an interface to configure the user's node arrangement. The 'Multicast Monitor Service' is in the 'Venue Client' interface (under the 'Services' directory). By right clicking the mouse, one can activate this service manually (right-hand side of Fig. 6). After triggering the multicast performance analysis tool, the 'Venue Client' opens a window (the left middle window in Fig. 6) for the user's input. Then, a user can select an IP address from his group or anyone from another multicast session. Finally, the results will be presented in the bottom left of Fig. 6.

Here, the IP address is used as the source-root. The results indicate the local time (at the first line) and the averages and variances for both delay and connective metrics (lines 2–5). The smaller these statistics, the better the quality of communication. The number of nodes in the multicast group is also displayed (line 6). The last two lines show the fairness indices for this multicast group. As shown in Fig. 6, the fairness matric of latency is 0.695, that means almost 70% of the receivers in this group have a fair delay cost – they have good connectivity. Each access grid user can refresh the results for a specific IP address or change to another IP address every one minute.

3.3 Sensitivity of metrics

A set of experiments are conducted to measure the performance metrics for the whole multicast group as well as one specific access grid node. All experiments were executed in the same time period – they have the same starting time and the same finishing time, about 33.3 h. Data points are obtained once a minute and totally 2000 points for each experiment.

Fig. 7 plots the mean delay measurements for the whole group as well as one specific access grid node. All the data points in this figure seem to split into two major curves (clusters). The upper curve comprises data points of the measurements for the whole group while the lower curve is for an access grid node with a specific IP address. The lines of average value for each cluster are also drawn separately. It can be seen that each curve varies considerably along the straight line. The result shows that this selected access grid node has better delay performance than the average of the whole group according to the RTT metric.

Note that both curves are not smooth or continuous as expected. Especially, the lower curve reveals this feature explicitly. Some data points increase or decrease dramatically from their adjacent areas. For example, at the time point 1370, the mean delay of the whole group is about 180 ms while the mean delay of the selected node is about 170 ms. Both points increase significantly from their adjacent areas. In the lower cluster, in the time range from 1140 to 1410, all the data points have big values above the average. This range corresponds to the interval from 8:30 a.m. to 1 p.m. and it implies that the heavy traffic load of daily work might cause this interesting pattern. Moreover, in Fig. 7, it can also be observed that several value gaps emerge in each curve. For example, in the lower curve, at the time point 410, there is a 15 ms value gap; at the time point 1140, the RTT value increases about 7~8 milliseconds suddenly. Besides reasons discussed above, the modification and reconfiguration of the underlying distribution topology of multicast network might cause these gaps. For instance, when members join or leave a group, especially when they play important roles in the delivery tree such as gateways or bridge nodes, the distribution topology could create or shutdown an entire branch.

Traditional metrics only have the mean delay metric (i.e. spatial metric) for multicast online monitoring. The connective quality is

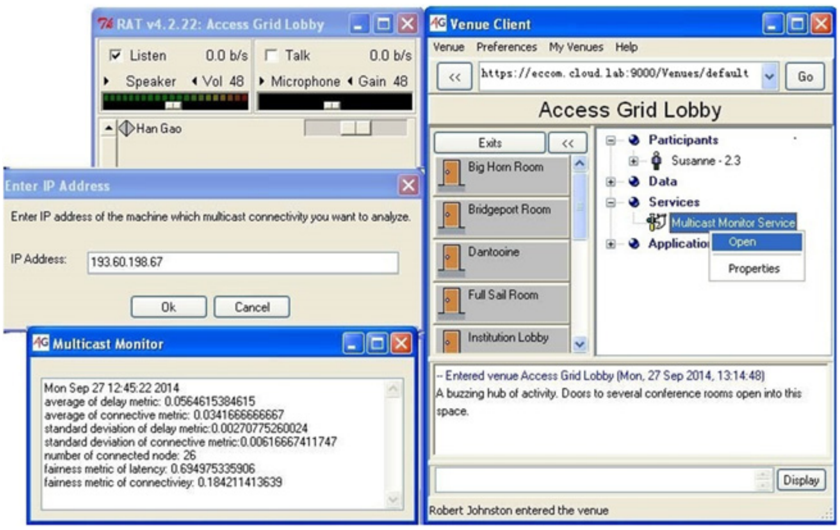

Fig. 6 *Graphic user interface of the multicast middleware*

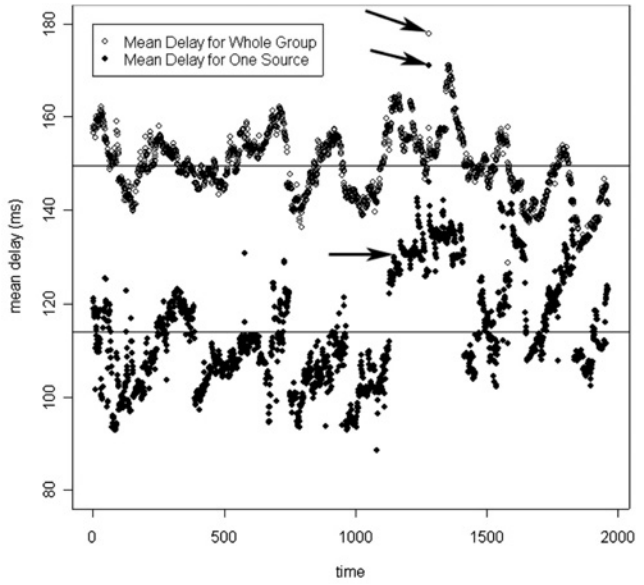

Fig. 7 *Mean delay measurements for the whole group and a specific access grid node*

another critical requirement for multicast networks. For connective metrics, Fig. 8*a* plots the loss rate measurements for both the whole group and one specific access grid node, and Fig. 8*b* plots the delay jitter measurements of the whole group. The connective metrics have different functionalities from spatial metrics. For example, during the time interval around 1500, Fig. 7 shows the mean delay of whole group is below the average, but Fig. 8*a* shows the loss rate values increase a lot. This result reveals that although the RTT is better at time point 1500, the connective quality for some multimedia grid applications of this multicast network is not good. Also in the time interval around 1000, Fig. 7 shows the delay performance is better than the average value. However, in Fig. 8*b*, the delay jitter metric has the highest value around time point 1000, which indicates that the multicast network topology is not stable enough for the quality-guarantee applications, although the RTT metric is lower. Therefore, in the last two examples, it might mislead users to an inaccurate multicast performance analysis if they only have the values of spatial metrics. The measurements of connective metrics provide users complementary measurements to spatial metrics. Then, the combination of connective and spatial metrics enables users to obtain comprehensive performance monitorings and analyses of multicast networks.

The fairness metrics reveal the resource allocation of the whole group. Fig. 9 plots the fairness of RTT measurements for a selected access grid node. The curve fluctuates a lot along the mean value line. The highest value of fairness is about 85% while the lowest is about 35%, which means, in the best case, 85% of network resources are distributed fairly, and in the worst case, 65% of resources need to be reallocated. Since the fairness is based on equality, it provides a reference for the current resource allocation of multicast distribution topology. If possible, multicast application users or providers can reconfigure the underlying network systems and distribution topologies according to the fairness indices.

3.4 Comparison of topologies

Several multicast topology configurations are possible for multicast services [26]. In the developed multicast middleware, there is a set of performance metrics for all these topologies, thus, the service providers can choose the best topology according to the features of their multicast services.

To illustrate, two different topologies for access grid nodes are evaluated. The resource nodes of these two topologies are the

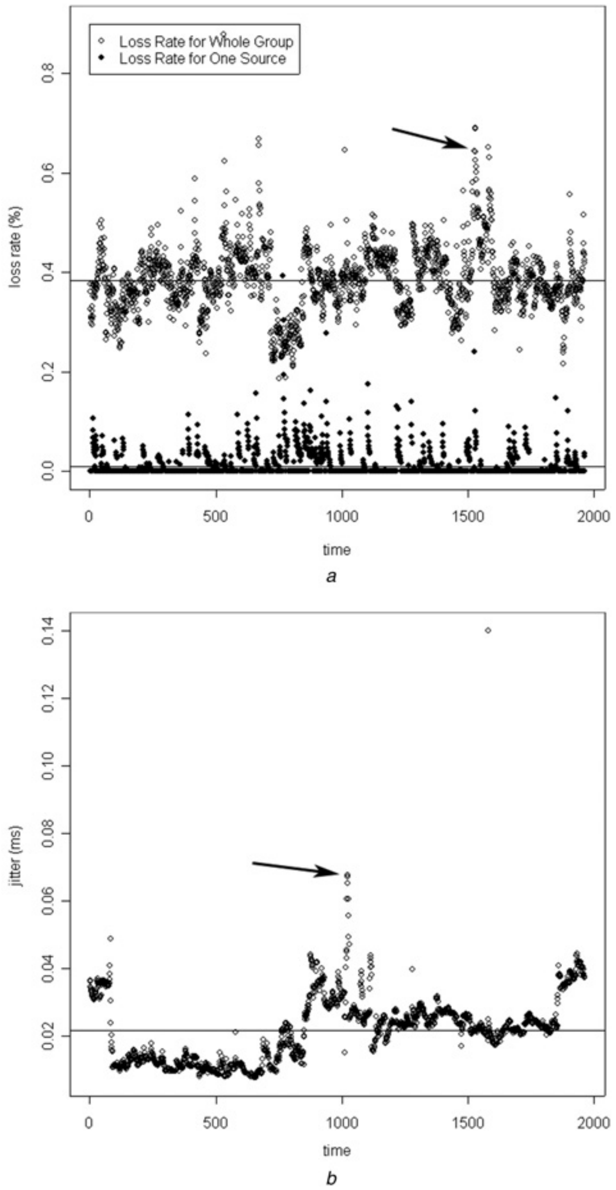

Fig. 8 *Measurements of connective metrics for the sample access grid multicast communications*
a Loss rate measurements for the whole group and a specific access grid node
b Delay jitter measurements for the whole group

same. Table 1 presents the latency statistics (mean and variance) and fairness results of them. It can be observed that values of the latency metrics in topology A ($\mu_S = 0.18$, $\sigma_S^2 = 0.02$) are bigger than in topology B ($\mu_S = 0.13$, $\sigma_S^2 = 0.01$) while the spatial fairness metric of topology A, $F_S = 0.65$, is better than that of B, $F_S = 0.59$.

Such table could be used as a reference when a user or service provider wants to select the best topology for a specific multicast application. If the user is more concerned with the QoS, topology A could give better results of fairness metrics which indicates this topology has more reliable implementation for the multicast service. On the other hand, if the user cares more about performance, the smaller group delay metrics would make topology B preferable.

Besides, if the overlay of the multicast network cannot be reconfigured easily, the middleware can provide general information about which kind of multicast services or applications can be implemented in the current situation.

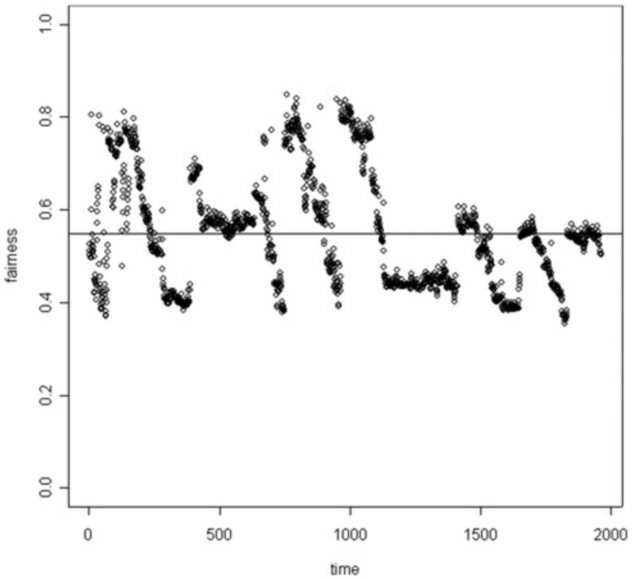

Fig. 9 *Fairness measurements for a selected access grid node*

4 Discussion

The real-time analysis for multicast performance and topology are critical for users and multicast server providers, however, since multicast communications has the following characteristics:

- Communications between clients and servers are handled in real time protocol via user datagram protocol.
- The measurements of communications have the distributed nature.
- Clients can start/stop the multicast server at any time.
- Multicast has a specialised communication protocol.

The different behaviour between multicast and unicast will lead to errors in the conventional performance statistics methods [27]. Those methods can minimise the errors at some level but cannot avoid them completely. Therefore, an ideal middleware for multicast diagnosis should comprise a set of measurement hosts which send small probe packets to a particular multicast session and receive packets from the session in order to determine the session transfer performance. These packets should use multicast protocol as the underlying backbone for generating statistics.

Compared with other studies which measure the multicast performance with the unicast measurement methods, the algorithms and metrics proposed in this paper allow the development of such a multicast middleware that one can obtain performance measurements data directly from the multicast communication protocol and thus avoid complicated data inference methods (which transform unicast data to multicast data). The benefits of this middleware are highly accurate real-time performance data collection and real-time topology analysis. With the more reliable data on both performance and topology, the accurate analysis of 'whether the current network condition meets the requirements of a multicast application' and 'which topology is the best choice for a certain multicast server' can be realised, then, immediate actions can be taken in response to performance events. Since these benefits can improve the manageability of large scale multimedia grids, the middleware provides a baseline for the development of new high-performance multimedia grid applications.

The metrics designed for multicast performance evaluation provide a precise and comprehensive guide for the middleware implementation. The features of these metrics are summarised as follows:

- Scalability: Both local and group metrics are applicable to any number of multicast nodes. Thus, the number of users does not affect the values of metrics.
- Continuity: The connective metric can be a continuous function, that is, it can be changed slightly for each node. The metric is very sensitive to different multicast topologies.
- Independence: The fairness metric does not vary if the unit of a measurement is changed. For example, the fairness of latency and jitter metrics remain the same whether using second or million second as the unit.
- Boundedness: The fairness metric is bounded between 0 and 1. It can also be expressed as a percentage, where 0 per cent means totally unfair and 100% means totally fair.
- Function independence: The fairness metric is independent of a defined metric function. This feature enables users or providers to select their own preferred metric function.

In addition, it needs to be emphasised that such data analysed by the proposed middleware are becoming increasingly valuable as new technology develops. For instance, for the convergence among grid services, web services and network services, the logical topology provided by the middleware can integrate these new services into the existing multimedia grids. The further investigation is also directed to the integration with cloud computing where the nodes are on virtual machines [28].

5 Conclusion

In this paper, a set of algorithms and metrics are proposed for the design and implementation of a multicast middleware which can accomplish the performance and topology evaluations in multimedia grids environment. The algorithms and metrics combine both physical and connective characteristics of multicast networks and provide a comprehensive evaluation means without much technological sophistication, meanwhile, the metrics can describe the multicast communications in a clear and consistent manner.

Because the quality of network performance data relies on the manner that the data are obtained by, in this paper, instead of inferring from unicast measuring data, the measurements of multicast are collected directly from the multicast session. Besides, along with client access and diagnostic tools, the deployment of the proposed middleware is able to integrate with the multicast protocol.

Table 1 Metric comparison of two different multicast topologies A and B

Topology	Active nodes	Latency metric				Fairness metric	
		Spatial		Connective			
		Mean, s	Variance, s^2	Mean, s	Variance, s^2	Spatial	Connective
A	8	0.18	0.02	0.056	0.0018	0.65	0.72
B	35	0.13	0.01	0.016	0.0008	0.59	0.50

An implementation of the multicast middleware is realised in this paper to demonstrate the viability of these proposed algorithms and metrics. As a result of the implementation, substantial improvements are made in the capability to analyse multicast network performance in real time and to re-route multicast traffic dynamically as the performance needs rise. As a particular benefit, the implementation can provide the base line data for further improving the existing applications on multimedia grids.

6 References

[1] Macedonia M.R., Brutzman D.P.: 'MBone provides audio and video across the Internet', *Computer*, 1994, **27**, (4), pp. 30–36, doi: 10.1109/2.274996

[2] Fall K.R., Stevens W.R.: 'TCP/IP illustrated, volume 1: the protocols' (Addison-Wesley, 2nd edn. 2011)

[3] https://technet.microsoft.com/en-us/library/cc957933.aspx, accessed September 2016

[4] Wang X., Fu L., Hu C.: 'Multicast performance pith hierarchical cooperation', *IEEE/ACM Trans. Netw.*, 2012, **20**, (3), pp. 917–930, doi: 10.1109/TNET.2011.2170584

[5] Xing H., Qu R.: 'A nondominated sorting genetic algorithm for bi-objective network coding based multicast routing problems', *Inf. Sci.*, 2013, **233**, pp. 36–53, doi: 10.1016/j.ins.2013.01.014

[6] Xu Y., Qu R.: 'A hybrid scatter search metaheuristic for delay-constrained multicast routing problems', *Appl. Intell.*, 2012, **36**, (1), pp. 229–241, doi: 10.1007/s10489-010-0256-x

[7] Patel M.K., Kabat M.R., Tripathy C.R.: 'A hybrid ACO/PSO based algorithm for QoS multicast routing problem', *Ain Shams Eng. J.*, 2014, **5**, (1), pp. 113–120, doi: 10.1016/j.asej.2013.07.005

[8] Ikeda K., Baduge T.M., Umedu T., *et al.*: 'ALMware: a middleware for application layer multicast protocols', *Comput. Commun.*, 2011, **34**, (14), pp. 1673–1684, doi: 10.1016/j.comcom.2011.03.004

[9] Xu Y., Qu R., Li R.: 'A simulated annealing based genetic local search algorithm for multi-objective multicast routing problems', *Ann. Oper. Res.*, 2013, **206**, (1), pp. 527–555, doi: 10.1007/s10479-013-1322-7

[10] Jiang X., Ye D., Chen Y.: 'OPRSFEC: A middleware of packet loss recovery in live multicast smart TV systems'. 2013 IEEE Conf. on Computer Communications Workshops, Turin, Italy, April 2013, pp. 3–4, doi: 10.1109/INFCOMW.2013.6970710

[11] Yu H., Ruepp S., Berger M.S.: 'Enhanced first-in-first-out-based round-robin multicast scheduling algorithm for input-queued switches', *IET Commun.*, 2011, **5**, (8), pp. 1163–1171, doi: 10.1049/iet-com.2010.0378

[12] Ghasvari H., Raayatpanah M.A., Khalaj B.H., *et al.*: 'Optimal sub-graph selection over coded networks with delay and limited-size buffering the authors consider', *IET Commun.*, 2011, **5**, (11), pp. 1497–1505, doi: 10.1049/iet-com.2010.0612

[13] Rajashekara H.G., Arul A.: 'Performance study connecting unicast and multicast setting in wireless networks', *Int. Res. J. Comput. Sci.*, 2015, **2**, (1), pp. 1–8

[14] http://www.cisco.com/en/US/tech/tk828/tech_brief09186a00800a4415.html#wp17758, accessed September 2016

[15] Ratnasamy S., Mccanne S.: 'Inference of multicast routing trees and bottleneck bandwidths using end-to-end measurements'. Proc. Eighteenth Annual Joint Conf. of the IEEE Computer and Communications Societies, New York, USA, March 1999, pp. 353–360, doi: 10.1109/INFCOM.1999.749302

[16] Luo J., Liu X., Ye D.: 'Research on multicast routing protocols for mobile ad-hoc networks', *Comput. Netw.*, 2008, **52**, (5), pp. 988–997, doi: 10.1016/j.comnet.2007.11.016

[17] Fahad A., Alshatri N., Tari Z., *et al.*: 'A survey of clustering algorithms for big data: taxonomy and empirical analysis', *IEEE Trans. Emerg. Topics Comput.*, 2014, **2**, (3), pp. 267–279, doi: 10.1109/TETC.2014.2330519

[18] Miyamoto S.: 'An overview of hierarchical and non-hierarchical algorithms of clustering for semi-supervised classification'. Proc. 9th Int. Conf. on Modeling Decisions for Artificial Intelligence, Girona, Catalonia, Spain, November 2012, pp. 1–10, doi: 10.1007/978-3-642-34620-0_1

[19] Celebi M.E., Kingravi H.A., Vela P.A.: 'A comparative study of efficient initialization methods for the k-means clustering algorithm', *Expert Syst. Appl.*, 2013, **40**, (1), pp. 200–210, doi: 10.1016/j.eswa.2012.07.021

[20] Mohammed G.A.: 'Improving fairness in packetized computer data networks', *J. King Saud Univ. – Comput. Inf. Sci.*, 1997, **9**, pp. 95–123, doi: 10.1016/S1319-1578(97)80006-7

[21] Maheshwari G., Gour M., Chourasia U.K.: 'A survey on congestion control in MANET', *Int. J. Comp. Sci. Inf. Technol.*, 2014, **5**, (2), pp. 998–1001

[22] Chen L., Ho T., Chiang M., *et al.*: 'Congestion control for multicast flows with network coding', *IEEE Trans. Inf. Theory*, 2012, **58**, (9), pp. 5908–5921, doi: 10.1109/TIT.2012.2204170

[23] Wong J., Sauve J., Field J.: 'A study of fairness in packet-switching networks', *IEEE Trans. Commun.*, 1982, **30**, (2), pp. 346–353, doi: 10.1109/TCOM.1982.1095465

[24] Seada K., Helmy A.: 'Fairness evaluation experiments for multicast congestion control protocols'. GLOBECOM '02 – IEEE Global Telecommunications Conf., Taipei, Taiwan, November 2002, pp. 2614–2618, doi: 10.1109/GLOCOM.2002.1189103

[25] http://www.accessgrid.org/project/Sumover, accessed September 2016

[26] Jiang D., Xu Z., Li W., *et al.*: 'Topology control-based collaborative multicast routing algorithm with minimum energy consumption', *Int. J. Commun. Syst.*, 2017, **31**, (1), doi: 10.1002/dac.2905

[27] Phonphoem A., Li-On S.: 'Performance Analysis and Comparison Between Multicast and Unicast over Infrastructure Wireless LAN'. Proc. Second Asian Int. Engineering Conf. on Technologies for Advanced Heterogeneous Networks II, Pathumthani, Thailand, November 2006, pp. 75–89, doi: 10.1007/11930181_6

[28] Peng I.H., Lee Y.C., Ho Y.H.: 'Study of multicast video streaming in cloud computing environment'. 2015 21st Asia-Pacific Conf. on Communications, October 2015, pp. 416–421, doi: 10.1109/APCC.2015.7412548