

Adaptive Multimedia Cloud Computing Center Applied on H.264/SVC Streaming

Wei-Ting Cho¹ and Chin-Feng Lai²(✉)

¹ National Cheng Kung University, Tainan 720, Taiwan

² National Chung Cheng University, Chiayi 621, Taiwan
cinfon@ieee.org

Abstract. In recent years, the multimedia streaming technology becomes increasingly mature, and as network bandwidth and computing power of personal hand-held devices are developed; therefore, it leads the requirements for multimedia quality increased. In order to provide the quality multimedia content to numerous users, how to divide the loading of content servers to improve the streaming quality is an important challenge. As the concept of multimedia cloud network is formed, how to allocate multimedia streaming service to the nodes in the media cloud network is discussed in this paper. This study designs H.264/SVC streaming service at the media cloud computing center, with the video most suitable for client-side quality provided based on H.264/SVC features (temporal scalability, spatial scalability, and quality scalability) and network bandwidth. In addition, the loading balance and communication mechanisms between nodes are discussed, where the best node is selected by the evaluating node, client-side bandwidth and computing power, for determining the appropriate video streaming path that is used to provide quality multimedia streaming service. According to the experimental results, the bandwidth prediction error rate for general multimedia network streaming service can be maintained at about 6%, while the utilization rate of various nodes of the multimedia cloud computing center is maintained in a balanced state during the period of executing multi-streaming service.

Keywords: Adaptive multimedia streaming · H.264/SVC · Cloud computing center

1 Introduction

As the internet and hand-held devices have rapidly developed, the smart phone with its multimedia related services become more and more popular. In order to provide users with the best video quality, highly compressed video formats are presented in succession. With fixed network streaming resources, almost all current streaming mechanisms use the unicast mode to transfer video of specific quality, as required by each user. However, the working efficiency of video servers is very likely to decrease as the number of users rapidly increases, thus, the needs of all users cannot be met [1-3].

In general cases, in order to meet user requirements, the video server side stores massive video data, including many kinds of video, and the same video data can be subdivided according to quality, thus, the data volume is very large and consumes a lot of storage resources. In addition, the server must act as streaming server to transfer videos to appropriate users, so the computing resource is more critical [4]. As network bandwidth rapidly develops and the computing power of the PC increases, all computers are able to be a computing node in a network, resulting in the formation of a cloud computing architecture, where available nodes in the network are managed for data storage and video streaming, thus, reducing the working load of specific servers [5–7].

This study discusses a new coding technique H.264/SVC, which uses Inter Layer Prediction so the file size after coding is greatly reduced, and the play quality of video is determined by packet analysis/filtering. This study uses the cloud computing mechanism and manages the resource of computing nodes in a media cloud for specific nodes to execute storage and streaming of video data, and dynamically evaluate the network bandwidth conditions and computing power of the nodes in the media cloud, and uses an Index Node to select storage and streaming nodes, thus, providing users with the best viewing quality and using the resources in the media cloud most efficiently. The contributions of this study are, as follows:

1. Construction of an H.264/SVC analyzer and decoder.
2. Construct management and communication mechanisms for nodes in a multimedia cloud.
3. User side bandwidth prediction model.
4. Computing resource balance in a media cloud and a dynamic node selection mechanism.

2 Related Work

This section briefly provides an outline of multimedia cloud computing and relevant studies, which can help readers outside the specialty of the article to understand the paper structure.

2.1 Multimedia Cloud Computing

For the multimedia services of cloud computing, the cloud center must provide an efficient, flexible, and scalable data processing mode, in order to provide a solution for user requirements of high quality and diversified multimedia [8,9]. On the other hand, with the popularization of smart phones and wireless networks, users are no longer limited to home network services, and they can obtain multimedia information and easily enjoy ubiquitous network services by using mobile devices. However, the present mobile streaming service has a bottleneck bandwidth and different requirements for device performance, cloud computing provides multimedia content suitable for a terminal unit environment, which

further considers the overall network environment and dynamically adjusts the transmission frequency of both sides and multimedia transcoding, thus, avoiding waste of overall network bandwidth and terminal equipment power [10, 11]. The theses [12, 13] designed a due management architecture control module for a Multimedia Content Cloud. Therefore, many studies have proposed a series of opinions on Cloud management design.

The ideal cloud computing multimedia encoding and decoding can double the efficiency of multimedia services, and there are usually three major services in a Cloud.

- Multimedia file information analysis function.
- Multimedia Codec decoding service.
- Terminal connection network quality state analysis function.

There is usually data dependence problems when decoding frames in a multimedia format; taking the decoding processes of H.264/SVC as an example, the frames can be divided into I frame, P frame, and B frame. The P frame is decoded by referring to the picture data of I-Frame of GOP (Group Of Pictures), and B frame refers to picture data other than I frame and P frame. The scalability of H.264/SVC can be divided into temporal scalability, spatial scalability, and quality scalability. Therefore, a standard H.264/SVC decoding process can be approximately divided into entropy decoding (ED), inverse quantization and inverse transform (IQ/IT), intra or inter prediction (PPC), and deblocking (DF). Therefore, many studies configure a module for each processor according to the concept of dispersal to a cloud center for computing powerparallel processing according to the decoding process [14, 15], thus, as the processors share the processing load, processing is accelerated. The advantage of this mechanism is to eliminate data dependence simply and efficiently; however, its defect is that the distribution of modules is limited to the quantity of assigned tasks in cloud computing.

3 Proposed Center Architecture

This chapter introduces the proposed H.264/SVC based multimedia cloud service architecture model, which aims to provide users with real-time and high quality videos. Based on the layered coding architecture of SVC and the analytical separation of bit stream, the nodes in a media cloud are divided into Index Node, Content Node, and Streaming Node, in order to solve the excessive loading on traditional multimedia servers. In order to handle the requirements of numerous users, the traditional multimedia servers must have on-line management, data storage, and media streaming, with simultaneous functions, and if the server-side cannot dynamically assign users to nodes with better bandwidth, the overall streaming service quality will decline. Therefore, when bandwidth loading for streaming video quality is dynamically distributed to various nodes in the media cloud in order to balance the loading on the nodes, the overall streaming efficiency can be greatly increased.

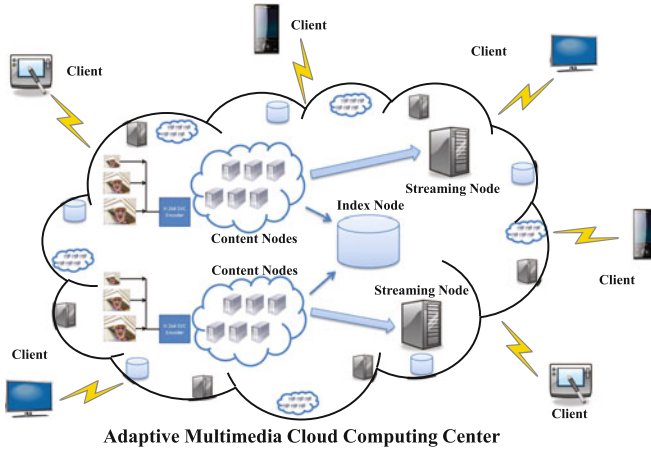


Fig. 1. The architecture of adaptive multimedia cloud computing center

3.1 Center Components

The center structure proposed in this study is shown in Fig. 1; there is one index node and multiple content nodes and streaming nodes in the media cloud, and their functions are described below. Index node, it records the dynamics of all nodes in the media cloud, including the registration and departure of nodes, records the type and IP information of nodes, and actively sends inquiry or control commands to the content and streaming nodes; Content node stores specific media segments. When a new video is added in the media cloud, it is separated into multiple segments, and dispersed for storage in multiple content nodes in the media cloud, thus, avoiding all the streaming nodes accessing the same video content node, which causes bandwidth bottlenecks; Streaming node is the node that streams video to users according to the layering characteristic of H.264/SVC; the streaming node captures specific video segments from the content node according to the user request to the index node for video, and then refers to the bandwidth of the streaming node and the lowest bit rate for playing the video segments in order to determine the level of video streaming to the user.

3.2 Center Operation Flow

This chapter introduces the practical process of center operation, and further introduces the command communications among the different nodes and the data transfer direction. As mentioned in the previous chapter, the index node records the operating state and information of all the nodes in the media cloud. Therefore, all command transfers must be actively sent and executed by the index node. Figure 2 shows the center operation flow.

First, when the client-side wants to view a video, it sends a video request to the index node, the corresponding instruction is step 1, Request Content.

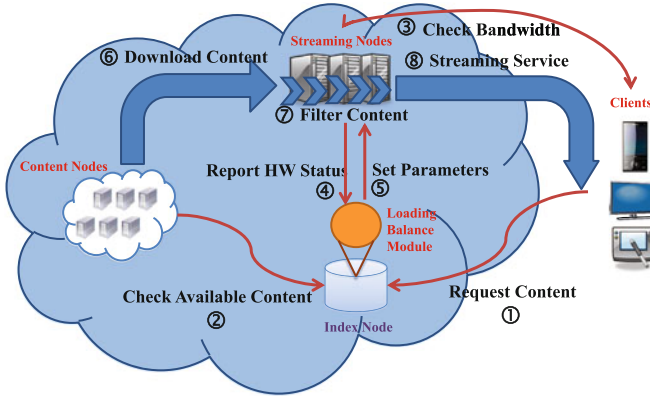


Fig. 2. The center operation flow

When the index node receives the request from the client-side, a series of Loading Balance mechanisms begin, and the index node sends information inquiring about the availability of video content to all content nodes in the media cloud, as step 2, Check Available Content. As mentioned in the previous chapter, a complete video is separated into multiple segments before it enters the media cloud, for storage in various content nodes of the media cloud. This step checks which nodes contains the video segments to be played, and then determines which content nodes have the requested video segments.

Afterwards, the index node sends a request inquiring about bandwidth to the streaming node, as step 3, Check Bandwidth, the index node sends inquiries, one by one, to all the streaming nodes in the media cloud regarding bandwidth for the client-side, and then reports to the index node. Then the index node sends request inquiries about bandwidth to the screened content nodes. Finally, the values are fed back to the index node. Step 4 inquires about the hardware resource information of each streaming node in order to determine the work loading of each streaming node, and rearranges the evaluation of the equilibrium stock. After the aforesaid four steps, the results are handed to the loading balance model to evaluate the overall network and select nodes. Practical content capture, analysis, separation, and streaming are conducted after the Loading Balance Mechanism. In step 5, the index node commands the selected streaming node to set the streaming quality level, shown as Set Parameters in the figure. The index node transfers the selected content node as a parameter to the streaming node, which downloads a content based on the information, shown as step 6, Download Content, to the streaming node, and then analyzes and separates the content according to the given parameters in the Set Parameters command. Due to the layering characteristic of H.264/SVC coding, the packets can be separated according to the information stored in the NAL Header. Finally, the streaming path is fed back to the index node, which feeds the streaming address back to the client-side.

3.3 Loading Balance Model

The loading balance model which aims to render the work load of all the nodes in the media cloud consistent is discussed in this section, while providing the client-side with better streaming quality. The loading balance model contains 1: a prediction model for the bandwidth of streaming node for the client-side; and 2: the selection and balancing mechanism of the streaming and content nodes.

Prediction Model for Client-Side. The authenticity of bandwidth has a significant influence on bandwidth prediction, as it must depend on substantive historical data, and the self-designed mathematical model is used to predict future bandwidth. Therefore, the filing and authenticity of historical data significantly influence the accuracy of the overall bandwidth prediction model. In order to implement the client-side bandwidth prediction model of a streaming node, this study analyzes and predicts the bandwidth results measured at various endpoints. First, the video segment length is set as n seconds, and separated into m segments, during the play of No. 0 segment, the quality level of No. 1 video segment must be determined; therefore, this study sets every k seconds as a data group, which are analyzed and polynomial regression is used to calculate the regression polynomial curve in order to predict the data distribution pattern of data points. The polynomial regression is defined as Eq. 1, where ε is an error constant.

$$y = a_0 + a_1x + a_2x^2 + \cdots + a_mx^m + \varepsilon \quad (1)$$

According to the polynomial regression curve deduced from Eq. 1, extend the data into multidimensional data distribution, where the polynomial curve can be expressed, and the data distribution pattern is identical to Eq. 1. Express the distribution of polynomial curves of multidimensional data in a matrix form, as Eq. 2. Finally, the simplified results can be expressed in vector form, as Eq. 3.

$$\begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} + \begin{bmatrix} 1 & \cdots & x_1^m \\ \vdots & \ddots & \vdots \\ 1 & \cdots & x_n^m \end{bmatrix} \begin{bmatrix} a_0 \\ \vdots \\ a_m \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{bmatrix} \quad (2)$$

$$\mathbf{a} = (X^T X)^{-1} X^T \mathbf{y} \quad (3)$$

The polynomial regression curve, as deduced from the previous data group, can be used to predict the data of the next data point in order to determine the value of A_n ; use the polynomial curves obtained from $A_{n-1} \sim A_{n-5}$ in the X-coordinate of A_n to obtain a corresponding Y value, which is the predicted bandwidth, and the mathematical expression. However, as A_n is a value calculated by the mathematical model, and not a measured value, a second revision is required, as shown in Eq. 4. We can average the A_n est obtained from polynomial regression and the A_n rel of a real bandwidth measuring device in order

to obtain the new value, A_{n_rev} . Finally, A_{n_rev} is substituted in another regression prediction curve to continuously revise the regression prediction curve for accurate prediction.

$$\begin{aligned} A_{n_rev} &= \alpha A_{n_est} + (1 - \alpha) A_{n_rel} \\ &= \alpha f(A_{n-1}, \dots, A_{n-5}) + (1 - \alpha) A_{n_rel} \end{aligned} \quad (4)$$

Selection and Balancing Mechanism of Streaming Nodes. The fundamental purpose of the loading balance model is to render the Work Loading of all the nodes in the media cloud consistent, while providing the client-side with better viewing quality. Therefore, the index node considers the present state of the streaming node, including the bandwidth of the streaming node of the client-side, the computing power of the streaming node and the bandwidth of the content node for the streaming node, with the operation architecture as shown in Fig. 3.

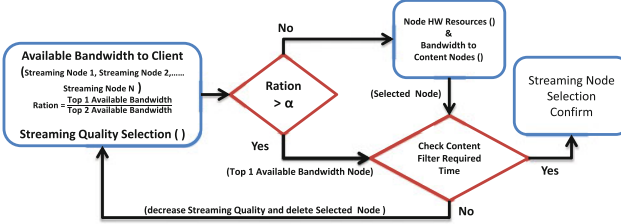


Fig. 3. Selection model of loading balance

The bandwidth of the streaming node for the client-side is the first position considered, as the fundamental purpose of the overall streaming center design is to provide users with better viewing quality, and the balance of the nodes is the second consideration. First, the index node selects the best three nodes for the client-side bandwidth from the media cloud, in descending order, and defined as Max_{BW1} and Max_{BW2} , and $Ration$ defined as Eq. 5. If $Ration$ is greater than bandwidth gap (defined α), meaning Max_{BW1} is far larger than Max_{BW2} , and the function of Check Computing Time is directly conducted in order to ensure the node fulfills the work within the determined time.

$$Ration = Max_{BW1} / Max_{BW2} \quad (5)$$

When the bandwidth ratio of more than two nodes in the media cloud of the client-side is less than gap, the second round of judgement must be conducted according to the work load of the streaming node. However, as the streaming nodes have different hardware specifications and computing power, using only the CPU utilization rate to determine the work load of the nodes will result in errors to some extent, thus, the hardware specifications of nodes must be

calculated to judge the actual work load of the nodes. Another factor influencing the computing time is the bandwidth of the content node for the streaming node. This study uses the same streaming node selection mechanism as the loading balance model, where two better content nodes are selected for the streaming node bandwidth, and calculated at *Ration*. If *Ration* is greater than gap, the total computing time is validated. If the total computing time is longer than the predetermined time, which does not meet center requirements, this node shall be abandoned. If the *Ratio*₁₋₂ is less than gap, meaning the two content nodes have similar bandwidths, the center selects a content node at random, and validates the total computing time, and finally, the selection of the content node is confirmed.

When the streaming node and content node are selected, the data transmission path is determined, the index node transmits the result to the streaming node, and the streaming node captures the video from a specific content node, according to the path given by the index node, and analyzes and separates the video according to the user's bandwidth. Finally the video is streamed to the client-side, and the operation of the loading balance model is complete.

4 Test Results and Analysis

This study validated the bandwidth prediction model of a streaming node for the client-side, and the selection and balancing mechanism of the streaming node, as proposed in Chap. IV, the bandwidth prediction model calculates the error rate defined as Eq. 6.

$$Err = |B_{est} - B_{rel}| / B_{rel} \quad (6)$$

In the case of varying bandwidths, and determines the precision of the bandwidth prediction model according to the Average Delay Time of the video on the client-side. The selection and balancing mechanism of the streaming node evaluates overall efficiency according to the time belt average service rate of each node, and analyzes the usability of the balancing mechanism by calculating the utilization rate and standby time of each node.

4.1 Error Rate and Video Average Delay Time of Bandwidth Prediction Model

As the bandwidth prediction model uses polynomial regression to obtain the bandwidth prediction curve, the denser the samples, the closer the prediction curve to the actual value. There is usually a rapid buffer in the transmission of massive files in a network, thus, the bandwidth transmission becomes stable after a period of time; therefore, the bandwidth measuring device requires time in order that the measured data can approximate to the bandwidth values of the network. Therefore, the sampling time and bandwidth measurement time must be balanced.

Table 1. Error rate of bandwidth estimation

Interval (s)	Limited band (Kbits)	Measured band (Kbits)	Error rate (%)
2	400	688	72
2	800	1032	29
4	400	412	3
4	800	826	3.25
8	400	405	1.25
8	800	815	1.875
10	400	405	1.25
10	800	810	1.25
20	400	402	0.5
20	800	807	0.875

Therefore, this paper analyzed the accuracy rate corresponding to the bandwidth measurement time. The analytic procedure is described below. Two bandwidth limits were set at 400 Kbits and 800 Kbits, with measurement intervals of 2, 4, 8, 10, and 20 s, the bandwidth was measured by the bandwidth measuring device under two speed limits, respectively, and their error rates were calculated. The test results are shown in Table 1.

According to the results, the error rates of the values measured by the bandwidth measuring device, at measurement intervals of two seconds, are as high as 72% and 29%, respectively, and the error rate is lower than 5% when the measurement interval is larger than four seconds, thus, the bandwidth measurement interval is selected as four seconds. Meanwhile, the bandwidth prediction model proposed is used, with four seconds defined as the measurement interval, and using 20-second video segments as the examples, thus, five data can be measured, and a data group can be formed. The polynomial curve is predicted using the data of the data group, and the bandwidth for playing the next video segment can be predicted.

The delay time of each segment is calculated by Eq. 7. The client-side has a Timer, which begins counting when No. 0 segment begins to play; T_{ER} represents the real time when the video segment ends, the T_{ES} represents the scheduled time that the video segment ends, T_{SR} represents the real time when the video segment starts, and the T_{SS} represents the scheduled time when the video segment starts, not counting the delay time of other videos, thus, the delay of a single video can be independently analyzed.

$$T_{delay} = (T_{ER} - T_{ES}) - (T_{SR} - T_{SS}) \quad (7)$$

4.2 Analysis Results of Bandwidth Prediction Model

This chapter records the predicted values of an actual bandwidth prediction model measured by a bandwidth measuring device according to the real center operation results with varying bandwidths. In this test, the conditions when the bandwidth increases or decreases suddenly are tested, the initial bandwidth is

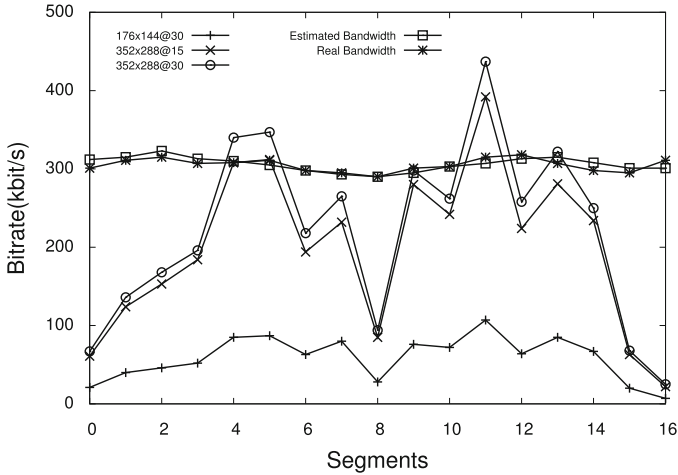


Fig. 4. Streaming bitrate of different bandwidths

limited to 300 Kbits, and the bandwidth is changed to 250 Kbits when playing the seventh video segment. The schematic diagram is shown in Fig. 4. The bandwidth prediction value, actual bandwidth value, error rate, and single video delay time, are recorded. The purple curve represents the values measured by the bandwidth measuring tool, and the cyan represents the bandwidth values predicted by the bandwidth prediction model. The data results are further discussed below.

In this experiment, No. 0 segment to No. 6 segment are limited to 300 Kbits, and this speed is adjusted to 250 Kbits from the seventh segment, as shown in Fig. 4. The error rate between the bandwidth prediction value and actual value of the seventh segment is 20 %, meaning the index node selected the wrong video level with a delay time of 2.13 s; however, the bandwidth prediction mechanism immediately corrected it. The error rates of the eighth and ninth video segments decreased to 5.8 % and 5.9 %, respectively, and the video delay was greatly reduced.

4.3 Analysis Results of Selection and Balancing Mechanism of Streaming Nodes

This chapter analyzes the node selection and node load under the streaming condition of an actual multi-node. In order to simplify the testing environment, the media cloud architecture is simplified, and includes an index node, a content node, and two streaming nodes. The actual node selections and load conditions are analyzed for two streaming nodes and two client-sides, with different bandwidths. This study analyzed the node load with two client-sides. In terms of hardware, this study used a CPU with improved computing power for Node Four, and used the worse CPU for Node Five. In terms of bandwidth, Case 1: the bandwidth of Node 4 is limited to 600 Kbits, and the bandwidth of Node 5

Table 2. Analysis streaming section for balancing mechanism Case 1

Seg	N4 300 Kbits & N5 600 Kbits					
	CPU_{N4}	CPU_{N5}	Sec_{C1}	Lev_{C1}	Sec_{C2}	Lev_{C2}
0	7.22	10.38	4	3	5	3
1	52.16	7.31	4	3	4	3
2	68.32	6.56	4	3	4	3
3	72.18	10.32	4	3	4	3
4	65.66	5.56	4	3	5	2
5	32.77	67.32	4	3	5	2
6	38.14	71.93	4	3	4	3
7	74.13	22.56	4	3	5	3
8	44.38	68.13	4	3	5	3
9	56.59	63.32	4	3	5	3
10	56.39	78.12	4	3	5	3
11	66.53	85.12	4	3	5	2
12	43.88	71.55	4	3	4	3
13	51.38	66.57	4	3	5	2
14	39.18	78.15	4	3	5	3
15	62.23	69.88	4	3	4	3
16	67.85	21.72	4	3	4	3
Avg	50.27	47.03				

is limited to 300 Kbits. Case 2: the bandwidth of Node 4 is limited to 300 Kbits, and the bandwidth of Node 5 is limited to 600 Kbits. The video play is tested, and the test results are shown in Tables 2 and 3.

According to Table 2, when two client-sides compete for bandwidth, the available bandwidth of each client-side obviously decreases; therefore, when two streaming nodes have similar bandwidths, the center adjusts the nodes according to the center load, and transfers the bandwidth demand centered in Node Four to Node Five. Finally, the measured center average load is 50.27% and 47.03%, respectively, which was obviously increased as compared with the single client-side.

With the test results in Table 3, when bandwidth limitations are interchanged, it is obvious that the nodes are allocated to Node Four and Node Five, because although the center judges that Node Five has better bandwidth, once the node is occupied, the center load of Node Five obviously increases, thus, the center assigns the video request of the second client-side to Node Four, as it has better computing power. The test results show that the average loads of Node Four and Node Five are 45.81% and 63.64%, respectively.

Based on the above test, with the operation of the balancing mechanism, if a node with strong computing power is provided with better bandwidth, the center will allocate most requests for videos to the node with the stronger capability. On the contrary, if a node with weak computing power is provided with better bandwidth, the center will allocate requests for videos to the nodes in the media cloud.

Table 3. Analysis streaming section for balancing mechanism Case 2

Seg	N4 600Kbits & N5 300Kbits					
	CPU_{N4}	CPU_{N5}	Sec_{C1}	Lev_{C1}	Sec_{C2}	Lev_{C2}
0	2.88	7.52	5	3	4	3
1	58.34	75.41	5	3	4	3
2	61.13	69.85	5	3	4	3
3	63.99	78.15	5	3	4	3
4	63.18	76.88	5	3	4	2
5	54.18	67.32	5	3	5	2
6	52.96	72.79	5	3	4	3
7	49.86	66.13	5	3	4	3
8	51.33	71.16	5	3	4	3
9	38.41	69.32	5	3	4	2
10	44.62	62.58	5	3	4	3
11	65.39	78.55	5	3	4	1
12	66.51	79.23	5	3	5	3
13	15.32	89.16	5	3	4	2
14	48.93	68.15	5	3	4	3
15	49.52	69.47	5	3	4	3
16	38.12	61.23	5	3	4	3
Avg	48.51	63.64				

5 Conclusion

This study designs a video quality scalable streaming mechanism for the media cloud computing center, and constructs communication and management mechanisms for the media cloud, including the construction of a bandwidth measurement center and load monitoring mechanisms for nodes of specific purposes, and designs a balancing mechanism for node resources in the media cloud. During test and analysis, we discussed the error rate and client-side delay time of a bandwidth prediction model with varying bandwidths, and analyzed the actual operational effects of the multinode balancing mechanism.

Acknowledgment. The authors would like to thank the National Science Council of the Republic of China, Taiwan for supporting this research under Contract NSC 101-2628-E-194-003-MY3, 102-2219-E-194-002 and 101-2221-E-197-008-MY3.

References

1. Zhou, L., Wang, X., Tu, W., Mutean, G., Geller, B.: Distributed scheduling scheme for video streaming over multi-channel multi-radio multi-hop wireless networks. *IEEE J. Sel. Areas Commun.* **28**(3), 409–419 (2010)
2. Han, J., Hu, M., Sun, H.: Search engine prototype system based on cloud computing. In: *Proceedings of the First International Conference on Cloud Computing*, Beijing, China, pp. 332–337, 1–4 December 2009

3. Ghandeharizadeh, S., Krishnamachari, B.: C2P2: a peer-to-peer network for on-demand automobile information services. In: Proceedings of 15th International Workshop on Database and Expert Systems Applications, Zaragoza, Spain, pp. 538–542, August 30–September 3 2004
4. Wu, S.Y., He, C.E.: QoS-aware dynamic adaptation for cooperative media streaming in mobile environments. *IEEE Trans. on Parallel Distrib. Syst.* **22**(3), 439–450 (2011)
5. Huang, D., Zhang, X., Kang, M., Luo, J.: MobiCloud: building secure cloud framework for mobile computing and communication. In: Proceedings of the Fifth IEEE International Symposium on Service Oriented System Engineering, Nanjing, China, pp. 27–34, 4–5 June 2010
6. Ferretti, S., Ghini, V., Panzieri, F., Turrini, E.: Seamless support of multimedia distributed applications through a cloud. In: Proceedings of the 3rd IEEE International Conference on Cloud Computing, Miami, FL, USA, pp. 548–549, 5–10 July 2010
7. Diaz-Sanchez, D., Almenarez, F., Marin, A., Proserpio, D., Cabarcos, P.A.: Media cloud: an open cloud computing middleware for content management. *IEEE Trans. Consum. Electron.* **57**(2), 970–978 (2011)
8. Lai, C.F., Vasilakos, A.V.: Mobile multimedia services over cloud computing. *IEEE COMSOC MMTCC E-Letter* **5**(6), 39–42 (2010)
9. Hu, G.Q., Tay, W.P., Wen, Y.G.: Cloud robotics: architecture, challenges and applications. *IEEE Netw.* **26**(3), 21–28 (2012)
10. Shao, L.L., Zhu, Z., Wang, Q., Sabhikhi, R.K.: Wireless network cloud: architecture and system requirements. *IBM J. Res. Dev.* **54**(1), 1–12 (2010)
11. Lai, C.F., Chen, M.: Playback-rate based streaming services for maximum network capacity in IP multimedia subsystem. *IEEE Syst. J.* **5**(4), 555–563 (2010)
12. Ali, T., Nauman, M., Hadi, F.-E., bin Muhaya, F.: On usage control of multimedia content in and through cloud computing paradigm. In: Proceedings of 5th International Conference on Future Information Technology, Busan, Korea, pp. 1–5, 21–23 May 2010
13. Chang, J.H., Lai, C.F., Huang, Y.M., Chao, H.C.: 3PRS: a personalized and popular programs recommendation system of digital TV for P2P social network. *Multimedia Tools Appl.* **47**(1), 31–48 (2010)
14. Huang, Y., Peng, J.L., Kuo, C.C.J., Gopi, M.: A generic scheme for progressive point cloud coding. *IEEE Trans. Visual Comput. Graph.* **14**(2), 440–453 (2008)
15. Chang, S.Y., Lai, C.F., Huang, Y.M.: Dynamic adjustable multimedia streaming service architecture over cloud computing. *Comput. Commun.* **35**(15), 1798–1808 (2012)