Adaptive Video Transmission Control System Based on Reinforcement Learning Approach Over Heterogeneous Networks

Bo Cheng, Jialin Yang, Shangguang Wang, and Junliang Chen

***Abstract*—Video may pass through various types of heterogeneous networks during the process of transmission, which has adverse impacts on the real-time video quality. Traditional methods focus on how to compress videos based on the video flow without considering the real-time network information. This paper presents an adaptive method that combines video encoding and the video transmission control system over heterogeneous networks. This method includes the following steps: first, to collect and standardize the real-time information describing the network and the video, then to assess the video quality and calculate the video coding rate based on the standardized information, and then to process the encoded compression of the video according to the calculated coding rate and transfer the compressed video. The experiments show that there is a significant improvement for the quality of real-time videos transmission without changing the existing network, particularly the core equipment. Our solution is easy to deploy and implement quickly and may help to extensively ensure video quality for normal users.**

***Notice to Practitioners*—The main objective of this work is to provide an adaptive video transmission control system and methodology to improve the real-time video quality, which takes the real-time network information into the video transmission control over heterogeneous networks. Our solution is an application-layer protocol and includes three phases: 1) to collect the network and video flow status simultaneously; 2) to adjust the parameters for video quality dynamically that come from the network and video environment feedback; and 3) to optimize the video coding rate that is in accordance with the current environment conditions. Our solution is easy to deploy and implement quickly, may help to extensively ensure video quality for normal users.**

***Index Terms*—Adaptive, heterogeneous networks, neural networks, reinforcement learning, video transmission control.**

I. Introduction

THE transmission of real-time videos still faces huge challenges through the current Internet. The traditional Internet offers best-effort communication services in which the network transfers all of the messages with its best effort, with no guarantees of the Quality of Service (QoS) [1]–[4]. To ensure the transfer of real-time video data, researchers have conducted many studies. The Internet Engineering Task Force (IETF) has proposed several QoS technical solutions, including integrated service, differentiated services, multi-protocol label switching, and traffic engineering. However, as the main QoS issue is always the problem of end-to-end transmission, which involves the entire network, changes to one or a few links will not solve the problem. Therefore, researchers have started to consider adding processes, such as retransmission at the application level, to increase the QoS, yet there have been no good results to date. Currently, the QoS problem during the video transmission process remains unsolved. Test data that have been recorded over a longer period of time may experience several heterogeneous networks that have different physical characteristics, calculation methods, and transmission methods from each other and, therefore, have adverse impacts on the QoS. From the perspective of the distribution range, the video communication network can be divided into the Local Area Network (LAN), Wireless LAN (WLAN), intercollegiate network, and the Internet. From the perspective of video terminals, we chose parameters representing the characteristics of the network: network time delay, jitter, and packet loss. These parameters are not only representative of the external characteristics of the entire network, but they are also easy to obtain without regard for the actual configuration or topology of the network. With two days of parameter testing, we collected almost all data for both the free and congested network scenarios. Fig. 1(a)–(d) shows the delay time distributions for the four types of networks.

We can see that the time delays of the LAN are approximately 1–2 ms. Most of the time delays of the WLAN are under ten milliseconds, and the majority is approximately 2–3 ms. The variation is much greater than that of the LAN. The time delays of the China Education and Research Network (CERNET) between the Beijing University of Posts & Telecommunications and Tsinghua University are approximately 10 ms and have relatively large variations. The time delays of the Internet (Beijing University of Posts and Telecommunications and Stanford University) are far greater than those of the previous three, concentrated between 180 to 300 ms. Not only are the absolute values large, but the variation range is also large, which may be a result of the international gateway and the long routing path. Table I lists the statistics of the parameters for each network in accordance with Fig. 1(a)–(d).

Manuscript received August 03, 2014; revised November 24, 2014; accepted December 16, 2014. This paper was recommended for publication by Associate Editor W. Tan and Editor H. Ding upon evaluation of the reviewers' comments. This work was supported in part by the National Natural Science Foundation of China under Grant 61132001, the Program for New Century Excellent Talents in University under Contract NCET-11-0592, and the Project of New Generation Broad band Wireless Network under Grant 2014ZX03006003.

The authors are with the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications; Beijing 100876, China(e-mail: chengbo@bupt.edu.cn; yangjialin@bupt.edu.cn; sgwang@bupt.edu.cn; chjl@bupt.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASE.2014.2387212

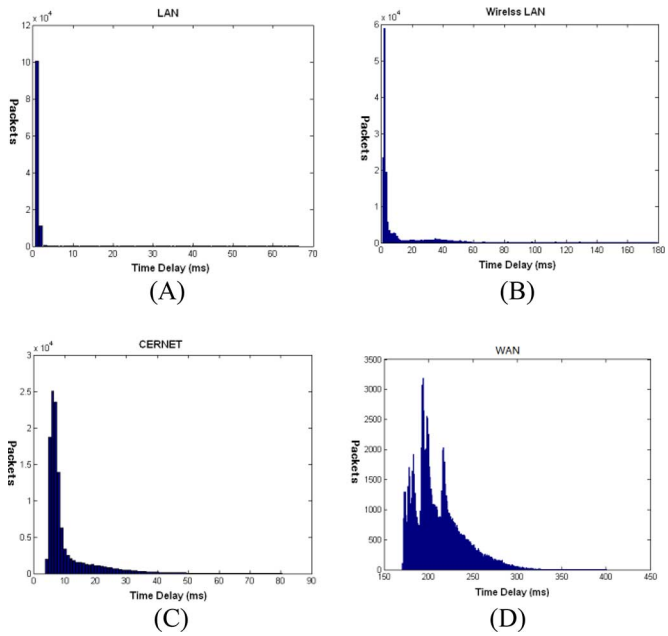

Fig. 1. Delay time distributions for different network types.

TABLE I
CHARACTERISTICS OF EACH NETWORK TYPE

Network Type	Average Time Delay (ms)	Average Jitter (ms)	Packet Loss
LAN	1.35	2.76	0.05%
WLAN	14.59	20.78	0.12%
CERNET	7.17	8.20	0.31%
INTERNET	211.09	28.41	0.48%

Currently, researchers are studying ways to compress the video better, hoping to find a new video coding method that is network friendly. The majority of studies [5]–[10] focus mainly on video compression based on the characteristics of the video, without considering the real-time network status information as time delay, jitter, and packet loss [12].

II. STATE OF THE ART AND OUR CONTRIBUTION

The related work to this paper can be generally classified into three categories: 1) video coding; 2) assessment of the video quality; and 3) Reinforcement Learning approach.

A. Video Coding

H.264/AVC, which is defined by International Organization for Standardization (ISO) and International Telecommunications Union (ITU-T), has relatively high coding efficiency and error resilience. Moreover, the code stream structure has relatively high adaptability and error recovery ability and saves approximately 50% of the coding rate of H.263 with the same picture quality. As with other standards, the H.264/AVC standard only prescribes the coding stream that can be accepted by decoders and does not specify how the coding machine should realize it. Another research area of high interest is the problem of controlling the coding rate, that is, how to reach the best video coding quality under the limits of the output coding rates. Wang *et al.*[12], Chen *et al.*[13], and Jimenez-Moreno *et al.*[14] have discussed coding rate controlling methods based on the rate-distortion optimization (RDO). Harsini and Zorsi [15] studied an adaptive coding rate control algorithm based on frame complexity. The application of H.264/AVC to different networks has drawn wide attention. Kim and Hong [16] investigated the configuration problem of applying H.264/AVC to WLAN real-time video, discussed different video coding and network characteristics under different scenarios, and proposed and tested principles for choosing the coding machine and the network parameter configurations. Kambhatla *et al.*[17] discussed the switching of video resources with different coding based on H.264/AVC. Hsiao *et al.*[18] studied the coding problem of H.264/AVC under different network bandwidths.

B. Assessment of the Video Quality

1) Peak Signal-to-Noise Ratio (PSNR): We normally use the PSNR or the mean square error (MSE) to measure the RDO during the coding and decoding processes of the video, that is,

$$\mathrm{PSNR} = 0 \log\left(\frac{255^2}{\mathrm{MSE}}\right)$$

$$\mathrm{MSE} = \frac{1}{N^2 \sum_{i=1}^{N} (x_i - \widehat{x}_i)^2}$$

among which x_i and $\widehat{x}_i$ are the pixels of the original and rebuilt pictures, respectively; N^2 is the overall number of pixels in an $\mathrm{N} \times \mathrm{N}$ picture. The algorithm is very attractive because it is easily calculated and improved mathematically, and it has a specific physical definition. However, the algorithm has the disadvantages of overlooking the sensory habits of the human visual system when looking at pictures. The assessment result may not be consistent with people's sensory habits for pictures.

2) Structural Similarity Index Measurement (SSIM): The SSIM is a method of assessing video quality based on structural distortion and was first proposed by Wang *et al.*[19]. Unlike PSNR, SSIM is developed based on the human visual system and draws on the structural information of a visual scenario to calculate the information change of the structural information before and after coding, so that it can assess the ratio distortion that people will perceive. The SSIM provides an objective evaluation method that is very close to that of human perceptual image distortion and is more accurate than the PSNR. However, the calculation is much more complicated than that of PSNR. The SSIM with corrections for the network situation will be used as the assessment standard for real-time video QoS in the present paper.

3) NTIA General Model: The National Telecommunications and Information Administration (NTIA) General Model stood out in the test of video assessment tools organized by the Video Quality Experts Group (VQEG). The model generated results consistent with those of objective assessments and was accepted by ANSI as the standard in 2003. NTIA has been working to find parameters that are not related to techniques and can be used to depict the image-quality perception behavior. The parameters could then be combined with linear regression models to obtain results that are close to people's objective assessment. When realized, the NTIA General Model [19]adopts a reduced-reference technology that uses the low-pass characteristic components extracted from the original video flow and the reconstructed video

flow. Although the method has very good video assessment results, the method is too complicated to be calculated and is not applicable to online real-time usage compared to the previous two methods. In consideration of the complexity of the method, the present study only used the method as an offline video quality assessment tool to evaluate the video quality received after network transmission.

C. Reinforcement Learning (RL) Approach

The RL approach to problem solving may be described as an agent that can sense its surroundings, learn by continuous trial-and-error and reach the highest skill level. Kaelbling *et al.*[20] introduced some basic problems in the RL approach area and summarized some classic application scenarios. In classic RL models, the agent is always undergoing dynamic change and is able to sense the environmental state. A corresponding action that could change the environmental state is specified. The agent chooses an action according to the real-time situation. The results are returned to the agent as a reward. The agent is expected to choose the action that will increase the long-term benefits that could be realized by the systematic trial and will receive corresponding reward feedback. The systematic learning process can be realized by various types of algorithms. Rummery *et al.*[21] also discussed the application of reinforcement learning to the Robert control area in detail, as well as the characteristics, advantages and shortages, in his doctoral dissertation. Reinforcement learning is similar to Dynamic Programming to some degree, and it can be used to solve optimization problems. One classic application of the RL approach is to learn the optimal control strategy in a real-time control system. Robert *et al.*[22] was the first to apply reinforcement learning to the video transmission area, solving the coding rate control problems of the WLAN transmission process for video and images in the medical field. Pradhan and Subudhi [23] proposed a real-time adaptive control for a flexible manipulator using reinforcement learning approach. Mastronarde and van der Schaar [24] proposed a fast reinforcement learning algorithm for energy-efficient wireless communication network. Yang and Jagannanthan [25] designed a reinforcement learning controller for affine nonlinear discrete-time systems.

In summary, current video transmission mechanism and video coding process research are relatively independent and fail to combine. If we are able to combine the research of the two fields, it may be possible to use the network information during video transmission to guide the coding process of the video so that the coding will have the characteristic of network adaptability, which may lead to good real-time video performance. Based on this problem, the present study proposed a network adaptive real-time video transmission system. The system mainly uses feedback information from the application level during video transmission and changes the parameters dynamically during the coding process, with an automatic control system based on Reinforcement Learning approach. The proposed adaptive video transmission control solution is an application-layer protocol and includes three phases: 1) to collect the network and video flow information simultaneously; 2) to adjust the key parameters for intelligent control based on the video quality information dynamically that come from network and video environment feedback; and 3) to optimize the video coding rate that is in accordance with the current environment conditions. Specifically, the contributions of this paper can be summarized in the following.

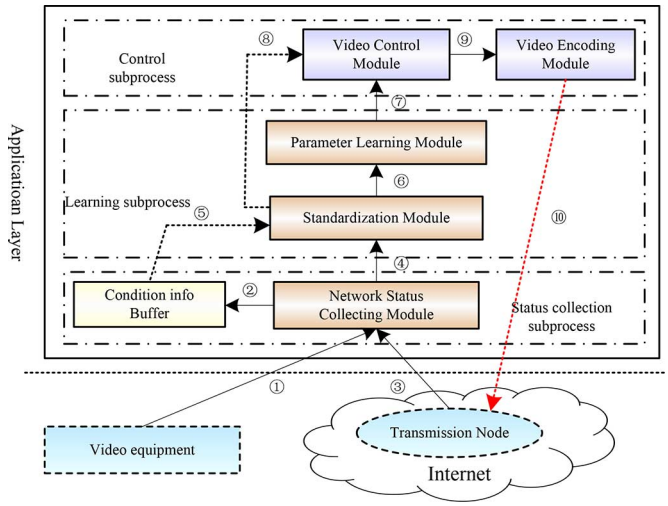

Fig. 2. Proposed system framework.

1) Evaluate the video quality with the network condition parameters as time delay, jitter, and packet loss rate and use the feedback information from RTCP to assess the network condition, which means no extra probe packet and a lesser burden to the network.
2) Adopt the Actor-Critic Model of the RL-based approach to adjust the key parameters that come from network and video environment feedback dynamically, derive the solution for video optimal transmission rate assignment, and optimize the real-time video quality over heterogeneous networks.
3) Deploy and test real experiments over different heterogeneous networks involving real-time H.264 video streaming and Session Initiation Protocol (SIP) communicator. Experimental results show that the system does not perform very well in the heterogeneous networks at first. The Video Quality Metrics (VQM) values are very high, and the video qualities are relatively poor. However, as the learning process progresses, the significant improving process of the video quality, which means that the system exhibits strong network adaptability.

The remainder of this paper is structured as follows. In Section III, proposed adaptive video transmission control system in detail. Experiments and performance evaluation is provided in Section IV and conclusion remarks are given in Section V.

III. System Architecture

The RL-based network adaptive real-time video communication system is composed of three processes that can be listed successively as the collection process for the network and video conditions, the intelligent controlled learning process, and the video coding adjustment process. The details of the proposed system framework are shown in Fig. 2.

As shown in Fig. 2, the system works on the application level and adjusts the parameters to accommodate dynamic changes in the network and video environment. The following sections introduce each function module in detail:

1) Network Status Collecting Module: This module is used to collect the condition status information of the network and video flows, which can be realized by monitoring the Real-time Transport Control Protocol (RTCP) flow (as shown in ③in Fig. 2), and the video condition information can be acquired in the process of video coding (shown in ①in Fig. 2). However, there is the problem of information inconsistency in the steps between the two conditions. The network condition information that is received from RTCP feedback is characterized by low frequency, while the video coding is high frequency. We adopted the following strategy to realize the condition-collection module. We divided the video frames into groups, each of which contained one key frame and all of the frames between two key frames, while waiting for the RTCP feedback. The complexity of the frames of the group is defined as the average complexity of all frames in the group, and the SSIM value is defined as the average SSIM values of all frames in the group. All complexity and SSIM information for each group is stored in the condition information buffer for updating, as shown in ②of Fig. 2. When the RTCP feedback reaches the system, it calculates the video reward value according to the network condition information from the feedback and the SSIM stored. The network condition information collected (as shown in ④of Fig. 2) and the video condition information stored in the condition information buffer (as shown in ⑤of Fig. 2) is sent to the standardization module for further processing.

2) Standardization Module: This module is used to standardize the condition values, and the following method to standardize the data: each real value entered is divided into N outputs valued between 0–1. N is identified based on the value range and distribution of the input. In consideration of the condition variables of the three networks, including the delay, jitter, and packet loss, and the video condition variable, the complexity of the frame, we performed the standardization as follows: we do not standardize the packet loss because its values are between 0 and 1, and the standardized values of the delay and the jitter are in accordance with the actual values of the four classic network characteristics, and the standardization of the complexity of the frame reviews the distribution of the real complexity data. We obtain 17 standardized values ranging between 0 and 1 after standardization. These values are used as the inputs of the parameter learning module and the video control module (as shown in ⑥and ⑧of Fig. 2).

3) Parameter Learning Module: This module is responsible for dynamic adjustments of the key parameters for intelligent control based on the video quality information that come from network and video environment feedback, it is the core module that gives the system the capability of online learning and environment adaptability. The module takes the outputs of the standardization module, which include the standardized network information such as the delay, jitter and packet loss, as well as the video information and the complexity of the frames, as the environment condition inputs (as shown in ⑥and ⑧of Fig. 2) and takes the SSIM values that have been offset by the network conditions as the reward of the current environment to adjust the corresponding parameters. The main frame adopts the Actor-Critic Model of RL-based approach. In the model, the Actor is responsible for the action under the current conditions, while the Critic learns to estimate the possible rewards under the current conditions. In the learning process, the Critic accepts the environment reward feedback, adjusts the forecasting function of the environment rewards with the Q-learning update rules, and sends the reward forecasting bias back to the Actor module as outside feedback. The Actor updates the chosen action strategy according to the reward feedback provided by the Critic. If the reward feedback from the Critic is positive, which means that the previous choice of the video-coding rate led to better video quality, the Actor adjusts the internal parameters so that the video-coding rate has a relatively high possibility to be chosen. If the reward feedback is negative, which means the previously chosen video-coding rate led to worse video quality, the Actor adjusts the internal parameters so that the previous coding rate has a relatively low possibility of being chosen. The parameters updated by the parameter learning module are used by the video control module for the selection process of the video coding rate (as shown in ⑦of Fig. 2).

4) Video Control Module: This module is responsible for the optimal video coding rate that is in accordance with the current environment conditions, which takes the standardized condition data that come from the standardization module and the internal parameters updated by the parameter learning module (as shown in ⑦and ⑧of Fig. 2) and chooses an appropriate video coding rate for the current environment following the strategy specified by internal parameters according to the current situation. The video control module must consider exploring the problem when choosing the video-coding rate, that is, it must choose whether to try a new video coding rate or to pick one from previous video coding rates and consider what strategy to take when choosing a new rate, and which uses the random sampling strategy, in which the possibility of choosing an existing rate is directly proportional to its corresponding reward. There are specific probabilities of choosing new video coding rates, and the probability of choosing a new rate is directly proportional to the known rewards of the rates near it. The input of a random selection is the condition values of the current conditions, and the internal parameters used are updated and adjusted by the parameter learning module. The output of the video control module is the video coding rate that should be used under the current conditions in the coding process of the video coding module (as shown in ⑨of Fig. 2).

Video Coding Module: This module is responsible for the coding process of the original video images. The module accepts the video coding rate that comes from the video control module as the input (as shown in ⑨of Fig. 2) and uses the value as the targeted coding rate of the video-coding module. The video flow after the coding process is transferred to the network in the form of Real-time Transport Protocol (RTP) flow (as shown in ⑩ of Fig. 2).

According to the classic statement of the RL-based approach, the present problem can be re-described in the form of reinforcement learning: the present control system can obtain a specific video quality on the customer's side by adjusting the

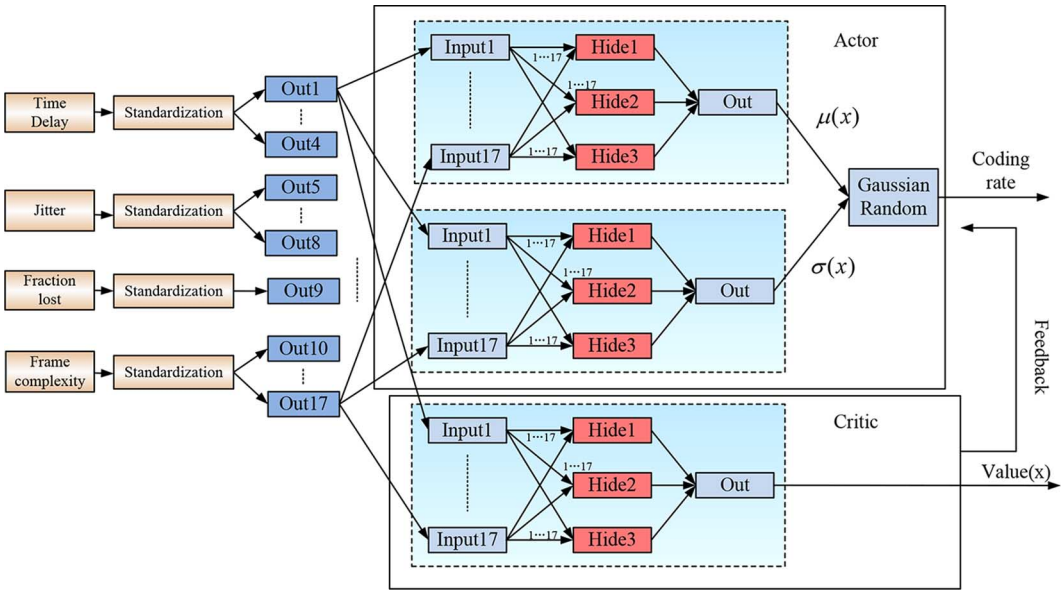

Fig. 3. Adaptive video transmission control system.

video coding process under dynamically changing network and video conditions. We can see that the system acts as an agent in RL-based approach. The real-time conditions of the network and the video constitute the environment conditions (Status), and the adjustment process of the video coding represents the possible actions of the system (Action) and the video quality on the customer's side is the reward in the system (Reward). The target of the present system is to learn the optimal video adjusting strategy by continuous trials, so that the estimation value of the video quality from the customer's side can be maximized. The core realization frame of the adaptive video transmission control system is shown in Fig. 3.

The controlling core accepts the three networks' information, including the delay, jitter, and packet loss, as well as the video information and frame complexity, as the conditional inputs for the current environment, and it then exports the video coding rate that should be set according to the input conditions. The main frame uses the Actor-Critic model (AHC) in the RL-based approach. The Actor is responsible for choosing the next action under the current conditions, and the Critic learns to forecast the possible rewards under the current conditions. The Value (x) in Fig. 3 is the forecasting function of the Critic regarding the environment reward. In the learning process, the Critic accepts the environment reward feedback, updates the Value (x) and sends the Actor module the reward prediction bias in the format of outside feedback to guide the Actor to adjust the selection strategy. In the present study, Value (x) learns though updating the rules according to Q-learning, the Actor adopts the Gaussian function to realize, and the Actor and Critic use a Back-Propagation (BP) neural network to generalize the data. Additionally, because the value ranges of each condition data differ from each other, we need to standardize the data before using them.

A. Collection of Network Video Characteristic Parameters

The network probe method can send a probe packet to the network periodically and then assess the current network conditions according to the time delay and loss experienced by the probe packet. One substantial problem with this method is the selection of the probe interval. The video flow is transferred through the RTP, whose flow can be seen as the data channel. In correspondence with each RTP is the control channel, that is, the RTCP, which is used to control the transmission of the RTP and the feedback statistics of the RTP flow, including the delay and jitter of the RTP packets. The statistics are sent to the video sender in the format of the Sender Report (SR) or Receiver Report (RR). The RTCP itself also helps to realize a time interval adjustment that is network friendly. Therefore, the present study uses the feedback information from RTCP to assess the network condition, which means no extra probe packet and a lesser burden to the network. In particular, we introduce some concepts: fraction lost rate, which is the fraction lost rate of the RTP data packet from sending SR or RR to the time of the feedback is the ratio of the actual packet loss to the expected packet number, and the value in the SR or RR is the result of the actual number times 256. Inter-arrival jitter, which is the statistical errors of the RTP packet arrives in the time interval, and use the same time unit as the timestamp, which is an unsigned integer. Last SR timestamp (LSR) means the time at which the other side receives the last SR, and if the other side never received the SR, the LSR is 0. Delay since the Last SR (DLSR) means the delay between the time of sending the feedback and the LSR. The packet loss rate and the inter-arrival jitter can be used directly, while the time delay must be calculated by the A_LSR_DLSR equation with LSR and DLSR. Here, A is the system time when the feedback is received. In this way, we can find the corresponding variables that can describe the real-time network condition. The process can be realized to monitor the SR or RR arriving event.

The QoS for video transmission represents the guarantee of successfully delivering packets, without delaying or dropping packets, which can be described by parameters such as the time delay, inter-arrival jitter and the packet loss rate. The real-time

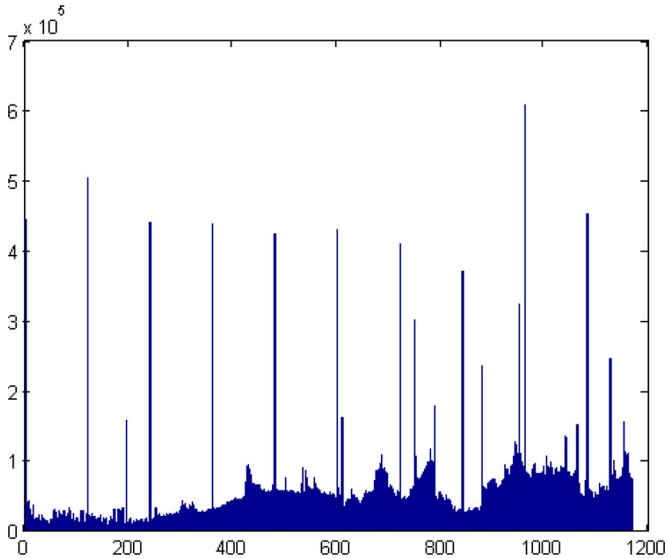

Fig. 4. Value range of the frame complexity.

video flow is transferred through RTP, and there is a control-channel RTCP flow that is used to control the transmission of the RTP and RTP flow feedback statistics, such as the delay and jitter of the RTP packet in accordance with the RTP data channel. The present study uses the feedback information in the RTCP as the assessment of the network condition, which requires no extra probe packet and causes a lesser burden to the network. The real-time condition of the video can be described by the frame complexity that depicts the relative complexity of each frame, for example, the changes compared to the last frame and the displacement shifts. The frames with a relatively high complexity can require relatively many bits, while the low-complexity ones will require less bits. Integrating the real-time requirement and the assessment reliability, we propose the assessment

$$ssim * e^{-\varepsilon_d \mathrm{delay}} * e^{-\varepsilon_j \mathrm{jitter}} * e^{-\varepsilon_l \mathrm{lost}}$$
$$\varepsilon_d = 1/100, \quad \varepsilon_j = 1/20, \quad \varepsilon_l = 256 \tag{1}$$

for the video quality as follows. In (1), the $ssim$ is the video quality value calculated with the assessment method for the structural distortion videos, and the time delay, jitter, and packet loss are in accordance with the three network condition parameters. The ε_d, ε_j, and ε_l reflect the relative influences of those three parameters on the quality of the video. The upper equation is based on the hypothesis that the video quality will decrease as the three parameters increase and that the influences of the delay, jitter and packet loss rate increase ordinarily.

B. Standardization of the Network Characteristic Parameters

The network characteristic parameters have high variance, so that the standardized values distribute inside a consistent range and provide with consistent standardized network condition descriptions. Time delay has very large distribution range, from one millisecond to a few hundred milliseconds. The value of jitter is relatively small, and the theoretically value ranges from 0 to 1, while it typically as a value lower than 0.01. Also, the value range of the frame complexity is shown in Fig. 4.

TABLE II
PARAMETER CONFIGURATION OF THE STANDARDIZATION

Condition Parameter	Number	B_n(0...N-1)
Time Delay	4	0.7, 3.6, 7.3,100
Jitter	4	1.38, 4.6, 13.9, 20
Frame Complexity	8	15000, 25000, 35000, 45000, 55000, 70000, 90000, 120000

Fig. 4 shows the distribution of the complexity of the video. Most of the frame complexity is under 10^5, and a minority frame could reach above $4 * 10^5$. Hence, using the video frame complexity values with large range differences as the input is not appropriate. It is necessary to standardize the data to a consistent range. We use the following method to perform the standardization. Each real number x will be divided into N outputs ranging between 0 to 1, and the nth output value will be as shown in

$$i_n = \frac{1}{1 + e^{w_n(b_n - x)}}$$
$$w_n = \frac{4N}{r} \tag{2}$$

among which $N$and $\mathrm{b_n}$ are defined by the distribution characteristics of the input data, and r is the possible value range of the input. For the four above-mentioned condition variables, the value range of the packet loss is between 0 and 1 and does not require standardization. The other three condition variables have the range information shown in Table II when they have been standardized with (2).

The values in the table were chosen in consideration of the analysis results for the different time delay distribution and the frame complicities of different networks. The $\mathrm{b_n}$ values of the time delay and the jitter are in accordance with the four different networks. We obtain 16 output data points after the standardization, with the packet loss rate contributing to a total of 17. The 17 data points are used to describe the environment condition of the system and as the input of the neural network used in the combination process of the Actor and the Critic.

C. Feedback Updating Mechanism

The Actor always chooses a coding rate at time $\mathrm{t}-1$and then obtains the reward feedback information that is the adjusted video quality information. Then, the Critic updates its reward predicting function using the Q-learning function. The bias of the Q-learning, that is, the predicting function at time $\mathrm{t}-1$ can be described by

$$\delta = \mathrm{r_{t-1}} + \gamma \mathrm{Value}_t - \mathrm{Value}_{t-1} \tag{3}$$

among which $\mathrm{r_{t-1}}$ is the received reward at time $\mathrm{t}-1$, and γ is a learning parameter.

The core idea of (3) is to replace the actual reward that will be obtained by the prediction using the predicting function and add the rewards received from time $\mathrm{t}-1$ to time t to receive the reward that should be obtained at time $\mathrm{t}-1$. Some errors are introduced because the prediction function cannot precisely reflect the actual rewards that the environment will return for the theoretical prediction. Somehow, the update mechanism has proven to be effective. As the learning process continues, the Value (x) function will approach the actual reward function. As

Fig. 3 mentioned, the $\mu(x)$ and $\sigma(x)$ are needed to estimate the average and the standard deviation, and the Gaussian distribution random number generator, which generates outputs in Gaussian distribution with $\mu(x)$ as its average and $\sigma(x)$ as its standard deviation, can be used. The output is the video coding rate under the current situation. Both $\mu(x)$ and $\sigma(x)$ are the current condition functions, and the learning process of the Actor is the process by which it updates $\mu(x)$ and $\sigma(x)$ according to the feedback that came from the Critic. Because of the Actor output is selected from the Gaussian random number generator, and the updated rules for $\mu(x)$ and $\sigma(x)$ are different from the Value (x) in the Critic. We used the updated rule based on the log of the Gaussian distribution. The detailed rule is shown in

$$\begin{aligned}\Delta\mu &= (a_{t-1} - \mu_{t-1})\delta \\ \Delta\sigma &= \left[(a_{t-1} - \mu_{t-1})^2 - {\sigma_{t-1}}^2\right]\delta\end{aligned} \tag{4}$$

where $\mathrm{a_{t-1}}$ is the output action value at time $\mathrm{t-1}$, σ_{t-1} is the predicted average and standard deviation of the output at time $\mathrm{t-1}$, and δ equals the feedback that the Critic provides to the Actor. With the updated rules, the Actor tends to increase the possibility that actions with positive feedbacks are chosen and to decrease the possibilities of actions with negative feedbacks. As the learning process continues, $\mu(x)$ will approach the optimal action value, and $\sigma(x)$ will decrease gradually, narrowing the range of selectable actions. The Value (x) in the Critic and the $\mu(x)$ and $\sigma(x)$ in the Actor are all functions that pertain to the current condition parameters. We use the BP neural network to combine these parameters, which involves parts of the updating rules that will be introduced in detail later.

D. Realization of the Generalization

A common problem in the RL-based approach is the generalization problem, that is, how to handle the scenario when the agent faces a new condition. For the agent to be able to address conditions it has never encountered before, the agent must have the ability to generalize. Function fitting is a popular method of implementation, and the application of the neural network is relatively extensive. We uses the three neural networks to fit the Value (x) in the Critic and $\mu(x)$ and $\sigma(x)$ in the Actor. The inputs for those three neural networks, which are the standardized condition values of the environment, are the same. There is a hidden layer in each neural network, which consists of three neural units, and there is an output neural unit in the output layer. The threshold of each neural unit in the neural network is the differentiable sigmoid function shown as follows:

$$\frac{1}{1+e^{-x}}, \qquad x = \sum_i w_i x_i \tag{5}$$

where $\mathrm{w_i}$ and $\mathrm{x_i}$ are the weight and input value of the $\mathrm{i_{th}}$ input of the neural unit, respectively. The practice of using three neural networks to fit the three functions individually ensures that the three functions will not affect each other when updating values, which is beneficial for the fitting of the results. The initial weights of the three networks are random numbers in the range $[-0.1, 0.1]$. To accelerate the learning efficiency and to decrease the error updating practice of the weight, we used the updating method with eligibility trace, which means that we only updated the weights that actually worked in the calculation. The method is a relatively widely used and effective method in the reinforcement learning area, and it could be used with TD (λ). The updating rules are as follows for each weight w in the network:

$$\begin{aligned}w_t &= w_{t-1} + \Delta w \\ \Delta w &= \alpha e_{t-1}\Delta o \\ e_t &= \lambda\gamma e_{t-1} + \frac{\partial o}{\partial w}\end{aligned} \tag{6}$$

where Δo is the error between the output of the neural network and the actual value, $(\partial o)/(\partial w)$ is the partial derivative of the output to the weight, α is the learning efficiency, e_t is the eligibility trace value of w at time t, and λ is the value of TD(λ), which reflects the reward distribution strategy.

In the neural network of the Value (x), Δo is the δ in the upper description. In the neural networks of the $\mu(x)$ and the $\sigma(x)$, Δo is the $\Delta\mu$ and the $\Delta\sigma$ in the upper description, respectively. At each time point of the update in the learning process, the network weights are updated by (6). The updating method is a popular one based on the decline of the gradient, with good theoretical support. The calculations converge to the solution with the smallest standard deviation error.

E. Adjustment of the Video Coding

The network status condition is received in the format of RTCP feedback, which has relatively low frequency while at the same time the video coding has relatively high frequency. Thus, synchronizing the two is a problem. After conducting in-depth research, we adopted the following strategy to realize the condition collecting module. We divided the video frames into groups, each of which contains two continuous key frames and the frames between, while waiting for the RTCP feedback. The complexity of the frames of the group is defined as the average complexity of all of the frames in the group, and the SSIM value is defined as the average SSIM values of all of the frames in the group. We used the frame complexity of the previous group to calculate the video coding rate for a new group when there is one, and we borrowed the interface provided by ffmpeg to make the rate effective. All of the complexity and SSIM information for each group is stored in the condition information ROM for updates. When the RTCP feedback is received, the system calculates the video reward value according to the network condition information from the feedback and the SSIM stored with (1). For each group of videos collected during the waiting time, we used the information stored and the network condition information to update those three neural networks in the adaptive control system. Then, we adjusted the current network conditions of the three neural networks, and calculated the new video coding rate and made it effective.

IV. Experiment and Discussion

To study the effect of each element in the video transmission process on the video transmission quality, we need a flexible and reusable experiment environment. The experiment environment in the present study, as well as the whole process and tools used in each stage, will be described in brief. As shown in Fig. 5, the

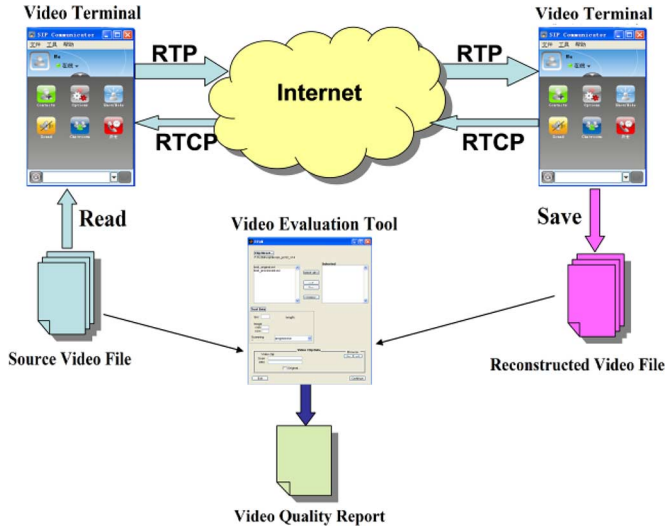

Fig. 5. Experiment environment and the process.

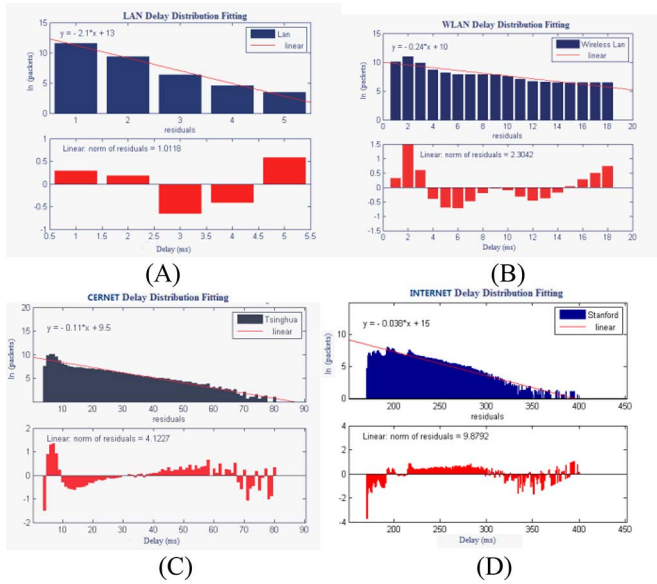

Fig. 6. Fitting condition for the time delay distributions of different networks.

entire experiment system is mainly composed of video terminals, network and video evaluation tools. The entire experiment process is as follows: the sender-side video terminal reads the video documents selected for test and sends the coded video flow to the other-side video terminal through the RTP protocol. The receiver side video terminal decodes and re-constructs the video image after receiving the video flow that RTP loads, and the receiver side video terminal then saves the reconstructed video images into the documents. The video assessment tool is used to assess the video quality received, taking the original video and the reconstructed video as the inputs. However, the video flow may pass various types of networks in the process of the RTP transmission, which causes problems such as uncertain time delay and packet loss. The receiver-end video terminal may not be able to obtain a complete well-aligned video flow, and the reconstruction of the video flow is greatly affected.

Because H.264/AVC is a coding and recoding pattern that is suitable for network video transmission, it was adopted in the present study for video coding. The video terminal adopted the SIP communicator, which supports H.264. We chose the Batch Video Quality Metric (BVQM), whose assessment is close to people's objective assessment as a video quality assessment tool. To describe the time delay distribution precisely, we need to process the data in Fig. 1(a)–(d). A classic method is to perform function fitting. With further investigation, we found that each curve in the Fig. 1(a)–(d) looked like a negative exponential curve; only the length of the tail and the decline speed of the ordinate differed. To test the idea and to make the process easy, we chose to use the log value of the ordinate (discard the 0 before log-transforming the value) and then fit the values with a line. If we obtain a well-fitted line, the upper idea is proved, and precise descriptions of the time delay distributions of the four networks will be acquired, as shown in Fig. 6(a)–(d).

Fig. 6(a)–(d) shows the fitting conditions of the four data sets. The upper part of each picture shows the original time delay and the fitting line, while the lower part shows the errors in the fitting process. Except for the beginning parts of the last three pictures, the errors are relatively small, and all four time delay distributions are well fitted by the lines. Because the ordinate is the logarithm value, we can draw the following conclusions: all the time delay distributions of the four networks are in accordance with the negative exponential distribution, and the distribution parameter λ is the slope of the line in the corresponding figures. According to the characteristics of the negative exponential distribution, the average time delay is $1/\lambda$, and the average time jitter equals $1/\lambda$ as well.

TABLE III
DESCRIPTION OF THE NETWORK CHARACTERISTICS

Network Type	Time Delay Distribution	Average Time Delay (ms)	Average Jitter (ms)	Packet Loss Rete
LAN	$2.1e^{-2.1t}$	0.5	0.5	0.05%
WLAN	$0.24e^{-0.24t}$	4.17	4.17	0.12%
CERNET	$0.11e^{-0.11(t-4)}$	13.09	9.09	0.31%
INTERNET	$0.038e^{-0.038(t-170}$	196.32	26.32	0.48%

Table III summarizes the upper inference and describes the characteristics of the four networks precisely. The column of the packet loss rate is inconsistent with Table I, and the columns of the average time delay and the time jitter are calculated from the time delay distribution of the second column. Comparing Tables I and III, we can see that the values and the statistical data of the inter-college network and the Internet are very close, while the Ethernet and wireless LAN differ from each other significantly. This result is related to the practice of giving up the long tail of the statistic, which leads to a relatively conservative data fitting outcome, that is, the values of the time delay and the time jitter are narrowed at some level, but there is still inconsistency with the statistics.

According to the upper analysis, we can see that Table III provides a good summary of the characteristics of the four classic networks. Therefore, we can use the data in Table III to determine the parameters of the networks when testing the performance of the adaptive system in each network.

The initial parameters used to fit the RL-based approach are usually random values that require continuous adjustments through a trial-and-error process. The performance of the

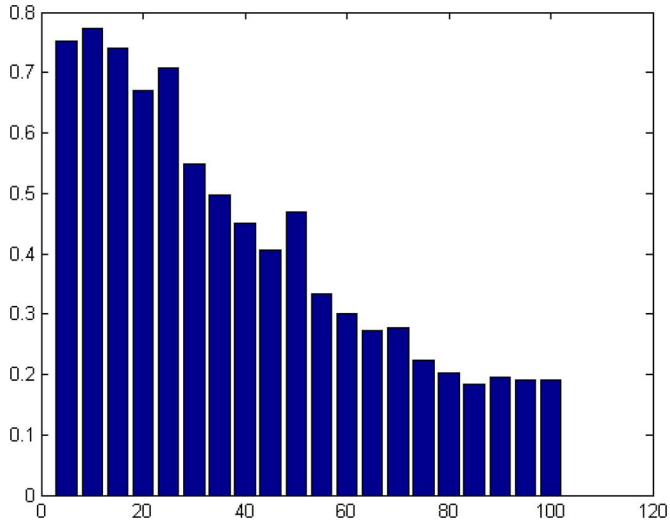

Fig. 7. VQM values during the learning process under ideal conditions.

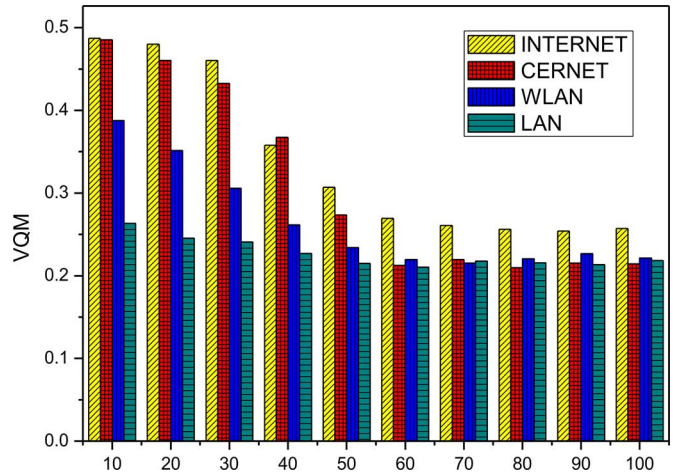

Fig. 8. VQM values during the learning process in different networks.

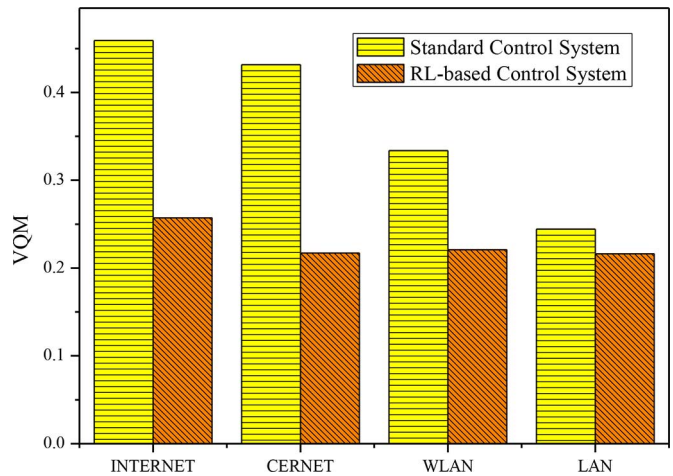

Fig. 9. VQM comparisons in different networks.

whole system is usually very poor in the beginning, and it sometimes cannot be compared with the original system. After a period of learning, the performance improves greatly, which is an important characteristic of machine learning algorithms. Therefore, we allowed the system to run the learning process for a long time before we tested the control performance of the intelligent control system. The entire learning process occurs in four steps: transfer the test video 100 times in each of the four networks, during which process the adaptive control system gains adaptability to special network environments. Ultimately, test the control capability that the adaptive control system learns from the four networks with the assessment of the video quality that the receiver side SIP communicator obtains. To study the learning process of the adaptive control system, we recorded the quality of the video received after every five learning cycles. Fig. 7 shows the quality condition of the video received in the learning process under ideal conditions.

We can see that the VQM value oscillates as the learning process goes on, but overall, the trend shows a decline, which indicates that the video quality is improving. After 80 cycles of learning, the VQM values remain at a steady value (approximately 0.2), which means that the intelligent control system has converged and that the first phase of learning is over.

Fig. 8 shows the video quality during the learning process of the intelligent control system in four different classic networks after the first phase of learning.

The RL-based adaptive video transmission control system does not perform very well in the four networks at first. The VQM values are very high, and the video qualities are relatively poor. However, as the learning process progresses, we can see the significant improving process of the video quality, which means that the system exhibits strong network adaptability. Additionally, we can see that the system converges to an optimal VQM value at the fastest speed in the LAN and WLAN follows, and the converging speed is slowest for the Internet. The result may be related to the similarity degree of each network with the ideal environment. The comparison of the video qualities of the RL-based adaptive video transmission after learning and of the standard H.264 rate control system is shown in Fig. 9.

We can see from Fig. 9 that the VQM for RL-based adaptive video transmission control system is better than the standard H.264 rate control system over the four different networks. The RL-based adaptive video transmission control system improves the video quality, especially in the Internet and CERNET Network.

V. Conclusion and Future Work

This paper proposes and realizes a network adaptive video transmission adaptive control system based on RL-based approach, and obtaining real-time network information through RTCP feedback, the H.264 video coding figuration and controlling, a comparison of the proposed RL-based adaptive controller and the "standard" H.264 rate control algorithm is also provided, showing the better capability of the proposed scheme to dynamically satisfy the network condition. Simulation results show that the proposed algorithm keeps both the video quality and the frame rate above the minimum quality of experience, also fulfilling the delay requirements. In the future work, we will consider eliminating some unimportant video frames to indirectly decrease the coding-rate export.

REFERENCES

[1] H. E. Egilmez, S. Civanlar, and A. M. Tekalp, "An optimization framework for QoS-enabled adaptive video streaming over OpenFlow networks," *IEEE Trans. Multimedia*, vol. 15, no. 3, pp. 710–715, Aug. 2013.

[2] W. Ji, Z. Li, and Y. Q. Chen, "Joint source-channel coding and optimization for layered video broadcasting to heterogeneous devices," *IEEE Trans. Multimedia*, pp. 443–455, Jun. 2012.

[3] M. Bystrom and J. W. Modestino, "Combined source-channel coding schemes for video transmission over an additive white Gaussian noise channel," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 6, pp. 880–890, Jun. 2000.

[4] E. Maani and A. K. Katsaggelos, "Unequal error protection for robust streaming of scalable video over packet lossy networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 3, pp. 407–416, Mar. 2010.

[5] S. Ahmad, R. Hamzaoui, and M. Al-Akaidi, "Adaptive unicast video streaming with rateless codes and feedback," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 2, pp. 275–285, Feb. 2010.

[6] E. Dedu, W. Ramadan, and J. Bourgeois, "A taxonomy of the parameters used by decision methods for adaptive video transmission," *Multimedia Tools and Applications*, Nov. 2013, DOI 10.1007/s11042-013-176-6.

[7] C. Gong and X. Wang, "Adaptive transmission for delay-constrained wireless video," *IEEE Trans. Wireless Commun.*, vol. 13, no. 1, pp. 49–61, Jan. 2014.

[8] Y. G. Lee and B. C. Song, "An intra-frame rate control algorithm for ultralow delay H.264/advanced video coding (AVC)," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 5, pp. 747–752, May 2009.

[9] H.-P. Shiang and M. van der Schaar, "A quality-centric TCP-friendly congestion control for multimedia transmission," *IEEE Trans. Multimedia*, vol. 14, no. 2, pp. 896–909, Aug. 2012.

[10] S. Soltani, K. Misra, and H. Radha, "Delay constraint error control protocol for real-time video communication," *IEEE Trans. Multimedia*, vol. 11, no. 4, pp. 742–751, Jun. 2009.

[11] C. Greco, M. Cagnazzo, and B. Pesquet-Popescu, "Low-latency video streaming with congestion control in mobile ad-hoc networks," *IEEE Trans. Multimedia*, vol. 14, no. 4, pp. 1337–1350, Aug. 2012.

[12] R. J. Wang, J. T. Fang, Y. T. Jiang, and P. C. Chang, "Quantization-distortion models for interlayer predictions in H.264/SVC spatial scalability," *IEEE Trans. Broadcasting*, vol. 60, no. 2, pp. 413–419, Jun. 2014.

[13] J. Y. Chen, C. W. Chiu, G. L. Li, and M. J. Chen, "Burst-aware dynamic rate control for H.264/AVC video streaming," *IEEE Trans. Broadcasting*, vol. 57, no. 1, pp. 89–93, Mar. 2011.

[14] A. Jimenez-Moreno, E. Martinez-Enriquez, and F. Diaz-de-Maria, "Mode decision-based algorithm for complexity control in H.264/AVC," *IEEE Trans. Multimedia*, vol. 15, no. 5, pp. 1094–1109, Aug. 2013.

[15] J. S. Harsini and M. Zorzi, "Transmission strategy design in cognitive radio systems with primary ARQ control and QoS provisioning," *IEEE Trans. Commun.*, vol. 62, no. 6, pp. 1790–1802, Jun. 2014.

[16] M.-J. Kim, K.-H. Kim, and M.-C. Hong, "Adaptive rate control in frame-layer for real-time H.264/AVC," in *Proc. ICACT*, 2008, pp. 1875–1880.

[17] K. K. R. Kambhatla, S. Kumar, S. Paluri, and P. C. Cosman, "Wireless H.264 video quality enhancement through optimal prioritized packet fragmentation," *IEEE Trans. Multimedia*, vol. 14, no. 5, pp. 1480–1495, Oct. 2012.

[18] Y. M. Hsiao, C. H. Chen, J. F. Lee, and Y. S. Chu, "Designing and implementing a scalable video-streaming system using an adaptive control scheme," *IEEE Trans. Consumer Electron.*, vol. 58, no. 4, pp. 1314–1322, Nov. 2012.

[19] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[20] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: A survey," *J. Artif. Intelligence Res.*, pp. 237–285, May 1996.

[21] G. A. Rummery, "Problem solving with reinforcement learning," Ph.D. dissertation, Eng. Dept., Cambridge University, Cambridge, U.K., 1995.

[22] S. H. Robert, N. Y. Philip, and G. Maria, "Martini medical QoS provision based on reinforcement learning in ultrasound streaming over 3.5G wireless systems," *IEEE J. Sel. Areas Commun.*, vol. 27, no. 4, pp. 566–574, May 2009.

[23] S. K. Pradhan and B. Subudhi, "Real-time adaptive control of a flexible manipulator using reinforcement learning," *IEEE Trans. Autom. Sci. Eng.*, vol. 9, no. 2, pp. 237–249, Apr. 2012.

[24] N. Mastronarde and M. van der Schaar, "Fast reinforcement learning for energy-efficient wireless communication," *IEEE Trans. Signal Process.*, vol. 59, no. 12, pp. 6262–6266, Dec. 2011.

[25] Q. M. Yang and S. Jagannathan, "Reinforcement learning controller design for affine nonlinear discrete-time systems using online approximators," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 2, pp. 377–390, Apr. 2012.

Bo Cheng received the Ph.D. degree in computer science from the University of Electronics Science and Technology of China, Chengdu, China, in 2006.

His research interests include multimedia communications, and services computing. Currently, he is an Associate Professor with the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, China.

Jialin Yang is currently working toward the M.S. degree in computer science and technology from Beijing University of Posts and Telecommunications, Beijing, China.

His research interests include multimedia communications software, value added service provision.

Shangguang Wang received the Ph.D. degree in computer science from Beijing University of Posts and Telecommunications, Beijing, China, in 2011.

He is an Associate Professor with the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, China. His research interests include service computing and cloud services.

Junliang Chen received the B.S. degree from Shanghai Jiaotong University, Shanghai, China, in 1955, and the Ph.D. degree in from Moscow Institute of Radio Engineering, Moscow, Russia, in 1961, both in electrical engineering.

He is a Professor with Beijing University of Posts and Telecommunications, Beijing, China. His research interests are in the area of service creation technology.

Prof. Chen is a member of the Chinese Academy of Science and the Chinese Academy of Engineering.