AN ADAPTIVE MECHANISM OF TFRC FOR STREAMING MEDIA SERVICE'S TRANSMISSION IN WIRELESS NETWORK

DOI 10.15589/SMI20170110

Zhang Xiao-wei
Master Degree Candidate
Research direction: transmission of wireless streaming media
conan1988@163.com

College of Mechanical and Engineering, Yancheng Vocational Institute of Industry Technology, YanCheng 224005, China

Abstract. In the wireless network environment with a high error rate, packet loss caused by wireless error will be misunderstood as congestion packet loss by the classic mechanism of TFRC, leading to lower throughput. To solve the transmission control problem of the wireless network for real-time streaming media services, the paper proposes a mechanism called Adaptive-TFRC. It is an improved mechanism of TFRC, which can reflect the true state of the network by using loss differentiation parameter of the receiving end, and then the value of parameter would be sent back to the sending end. We have improved the throughput model formula of the classic mechanism of TFRC in this paper. Finally, it can realize the dynamic adaptation of transmission rate. The simulation result shows that Adaptive-TFRC can greatly improve the throughput, reduce the delay jitter of real-time traffic, and it has a TCP friendly characteristic very well, thus guaranteeing service quality of the real-time streaming media in the wireless network.

Keywords: mechanism of TFRC; one-way delay; Adaptive-TFRC; wireless network; streaming media.

References

[1] S. Al-Majeed and M. Fleury, "Multi-connection TFRC Video Streaming in a Concatenated Network: Latency and Video Quality,"Recent Trends in Wireless and Mobile Networks, pp. 34–45, 2010.

[2] Yao X.W., Wang W.L., Yang S.H., et al. PABM-EDCF: parameter adaptive bi-directional mapping mechanism for video transmission over WSNs. Multimedia Tools and Applications, 2011, doi: 10.1007/s11042-011-0934-7.

[3] Tobe Y., Tamura Y., Molano A., et al. Achieving Moderate Fairness for UDP Flows by Path Status Classification. In Proc. 25th Annual IEEE Conf. on Local Computer Networks (LCN 2000), Tampa, FL, 2000:252

[4] S. Floyd, M. Handley, J. Padhye, and J. Widmer, "Equation-based congestion control for unicast applications," ACM SIGCOMM Computer Communication Review, vol.30, no.4, pp.43–56, 2000.

[5] S. Floyd, M. Handley, J. Padhye, and J. Widmer, "RFC 5348: TCP Friendly Rate Control (TFRC): Protocol specification," RFC 5348, 2008.

[6] Zhu Xiao-liang, Du Xu, Yang Zong-kai, et al. Rate control mechanism for wireless sensor network realtime media transmission [J]. Journal of Chinese Computer Systems, 2007, 28(2):199–204.

[7] Jiang Ming, Wu Chun-ming, Zhang Min, et al. Research of the algorithm to improve fairness and smoothness of TFRC protocol [J]. Journal of Chinese Computer Systems, 2009, 37(8):1723–1727.

[8] Wang Dong, Chen Ming, Zhang Da-fang. A multiplexing-based TCP-friendly congestion control mechanism for multimedia Streaming applications[J].Acta Electroneca sinica, 2006, 34(3):567–572.

[9] Cen S., Cosman P. C., Voelker G. M. End-to-end differentiation of congestion and wireless losses[C]. USA: IEEE/ACM Trans on Networks, 2003.

[10] Chen, M. and A. Zakhor, Multiple TFRC Connections Based Rate Control for Wireless Networks. Multimedia, IEEE Transactions on, 2006, 8(5):1045–1062.

[11] Shen, H., L. Cai, and X. Shen, Performance analysis of TFRC over wireless link with truncated link-level ARQ. Wireless Communications, IEEE Transactions on, 2006, 5(6):1479 –1487.

[12] Zhou, B., et al. An Enhancement of TFRC over Wireless Networks.Multimedia, IEEE Transactions on, 2007, 8(5):1045–1062.

Problem statement. With the development of wireless network technology and the maturity of multi-media technology, amounts of stream media such as IP telephone, VOD, video conference, distance education, etc. have sprung up in the wireless network. The stream media pose a large part of whole internet data, and easily cause overloading. So, the transmission of stream media technology challenges internet a lot, which becomes one of research problems [1, 2].

The stream media business requires the QoS of transmission data. Meanwhile, it also maintains the friendliness to TCP data stream. At present, the port-to-port control protocol includes TCP and UDP. TCP's internet jam mechanism and re-transfer mechanism result in huge delay not fit for multi-media stream transmission. When network gets into jam, the backoff mechanism of TCP will cause multi-media rate rebound. Therefore, the majority of transmission system take UDP as its protocol. Compared to TCP, UDP is much more suitable for steaming media as a kind of unreadable wireless protocol. However, UDP supplies limited service which does not control data stream and jam mechanism. It cannot adjust the rate from the sending port according to the network's state. It could bring serious problems about network resource allocation if it directly takes UDP as transmission protocol [3, 4]. For example, a lot of UDP stream and TCP stream share network brand bond together, and UDP would obtain the majority of resources unfairly, which would result in intensifying jams or even causing breakdown. Above all, traditional TCP and UDP are both not entirely for in transmission of stream media.

At present, TFRC is massively adopted by streaming media (TFRC: TCP-friendly rate control protocol) [5]. TFRC has been applied to wired networks earlier, but it did not have a good performance in wireless networks. Because packet loss in the wireless network is caused by network link errors and congestion, the TFRC receiver unreasonably adjusts the speed of the data stream, only to cause packet loss for congestion.

THE ARTICLE AIM is to focus on TFRC in wireless networks and propose an adaptive TFRC, which is more suitable for streaming media transmission because of its dynamic rate adaptability.

Recent research and publication analysis. TFRC is a port-port one way TCP-friendly protocol which bases on formula-controlled sending rate. IEIF formally promulgated the TFRC protocol RFC file in 2003. In recent years, it has drawn extensive attention and in-depth study of the scholars both here and aboard.

In the paper of Zhu Xiaoliang and et al. [6] IEEE802.11 is used to build WSN. The transport layer length adjustment equation satisfying the retransmission delay is derived, and the rate control mechanism (WSNRC) for the WSN real-time media transmission is established by combining TFRC. The mechanism could decrease packet-loss rate, average transmission delay and counterbalance between nodes which could not improve the throughput capacity in the whole internet. Jiangming and et al. [7] introduced weight coefficients in different power levels of the packet loss rate in the TFRC rate calculation formula. It reduced the transmission rate when the network congestion was intensified, and increased the transmission rate when the network congestion was lower, thus reducing the influence of network congestion on the TFRC stream transmission rate. Wang Dongand others [8] improved TFRC and proposed a TCP-friendly rate control algorithm based on multiplexing — MTFRC. The algorithm maintains the TFRC smoothness and at the same time increases the threshold limit of multimedia stream characteristics and multiplex link to environmental considerations, also the whole network throughput capacity has not been improved.

Song Cenet al. [9] proposed an end-to-end loss differentiation algorithm by ZigZag, Biaz, Spike — three kinds of algorithms for packet loss differentiation; ZBS, Spike, Biaz, ZigZag performance in different network environments are different. According to the network environment, the ZBS algorithm switches between these three; however, the misclassification rates of ZBS is high, and it can not take full use of the wireless bandwidth. Chen et al. [10] proposed the MulTFRC model, which increases the number of TFRC flows to increase the utilization rate of the wireless channel. Shen et al. [11] made up a wireless channel attenuation of the discrete-time Markov model which was based on the TFRC performance under different condition.

The domestic and foreign scholars studying the TFRC mechanism eventually can achieve inter protocol fairness and make the TFRC business flow rate maintain a high smoothness, but for packet loss in wireless networks it still could not accurately distinguish whether the cause is network congestion or link error. An improved adaptive TFRC mechanism (Adaptive-TFRC) is proposed in this paper, which can be used to effectively distinguish between packet loss parameters and accurately reflect the state of the network (i.e., congestion or error) according to the real-time dynamic adaptation of transmission rate, which is the ultimate guarantee of the data transmission quality for streaming media services in wireless networks.

Basic material.

TFRC analysis

1. TFRC control algorithm working principle

TFRC is a TCP-friendly Congestion Control Protocol which is mainly used for streaming media transmission mentioned by S. Floyd et al. In order to make the TFRC stream and TCP steam enjoy bandwidth fairness in the same network environment and achieve a TCP-friendly stream and little fluctuation of the TFRC transmission rate, the throughput of TFRC is calculated with the formula of the TCP Reno throughput model [12]. The throughput calculation of TFRC adopts the TCP Reno throughput model, as shown in formula (1):

$$X = \frac{S}{R\sqrt{\frac{2bp}{3}} + t_{RTO}\cdot\left(3\sqrt{\frac{3bp}{8}}\right)p(1+32p^2)} . \qquad (1)$$

In formula (1), X is the sending rate at the sending end, S is the size of the packet, R is the round-trip delay RTT, and t_{RTO} is the retransmission timeout time, usually $t_{RTO} = 4RTT$. The parameters R and b both play a decisive role in this formula. The value of b is 1 in the absence of the TCP validation mechanism; p is packet loss rate, which is the ratio of the number of loss events to the number of the packets transferred.

The linchpin of the TFRC control algorithm is how to obtain round-trip delay and packet loss event rate p. It is calculated at the control end, being the reciprocal of the average packet loss intervals in a given period of time. The formula for p is as follows:

$$p = \sum_{i=0}^{n} \omega_i \Big/ \sum_{i=0}^{n} C_i \omega_i . \qquad (2)$$

In formula (2), C_i is the nearest packet loss interval for Number I. The difference between the first packet loss sequence numbers of two consecutive packet loss events is defined as the packet loss interval, which refers to the number of packets between two packet dropouts. The average loss interval is obtained by weighted average of the nearest n loss event intervals, and the size of n determines the response speed of the TFRC when the congestion level changes. In order to compete with TCP fairly, TFRC is recommended to have $n = 8$ as the perfect balance. ω_i is the corresponding weight, and the values of $\omega_0 \sim \omega_7$ are 1, 1, 1, 1, 0.8, 0.6, 0.4, 0.2. Formula (2) shows that the calculation of the loss event rate is directly affected in the calculation of the loss event interval. The relationship with the size of the second packet loss event is determined by calculating the second packet loss number of the second packet loss event at the packet loss event interval. There is no direct serial number grouping.

2. Disadvantages of TFRC application in wireless networks

Traditional TFRC takes packet loss as a congestion signal and performs well in a stable network environment so it is widely accepted. However, due to the unstable network environment, TFRC throughput calculation formula can only be based on packet loss and round-trip delay to update the throughput estimates, so TFRC can not adapt to changes in the network. In the wired network, packet loss events are mainly caused by network congestion, so TFRC can be used effectively. The sketch map of packet loss intervals is shown in Figu. 1,*a*. When TFRC is directly applied to the wireless link in the network, the wireless receiver will take the packet loss as a congestion loss due to a wireless error, and then affect the loss event judgment, thus blindly reducing the rate, as shown in Fig. 1,*b*. In the diagram, C represents a congestion packet loss, while W represents a wireless packet loss. Due to latter, the packet loss interval is reduced from 6 (10–4) to 5 (9–4). As shown in Fig. 1–2, this will lead to a sudden decrease in the sending rate, decline the network throughput, and can not guarantee the smoothness of real-time data transmission. At the same time, when the calculated transmission rate is lower than the minimum, the packet will be sent at the lowest rate — resulting in invalid transmission, wasting network bandwidth, and seriously affecting the quality of streaming media playback. Therefore, improvement of the existing transmission, effective division of wireless and congestion error packet loss, and improvement of the performance and quality of the streaming media transmission in a wireless network all become urgent in the field of congestion control.

Improved TFRC algorithm

1. The basic notions of Adaptive-TFRC

Different network conditions can be distinguished by the delay and delay jitter, and many TCP-friendly congestion control algorithms in a wireless network determine the network status, which is the use of a single delay signal as the indication of congestion, and the values of the network status differentiation. When there is congestion, the packets in the injected network exceed the capacity of the network, resulting in the retention of the buffer area. Eventually, due to buffer overflow, the congestion will become much more serious. The longer the queuing time is in the router, the longer it will take to reach the receiver and the larger the one-way delay ROTT value will be. However, when the wireless link error is lost, the value will be lower, so the change of the value can reflect the congestion change of the network. When the Rott value is relatively small, if the packet loss is detected due to the wireless error, the sending end does not need to slow down, otherwise a packet loss caused by congestion is likely to occur. At this point, the sending end needs to reduce the sending rate.

ROTT eliminates jitter during link layer error control retransmission, thus affecting the accuracy of identifying the cause of packet loss. In this paper, we propose an adaptive TFRC eight data of adjacent packet difference calculation. Experiments show that the eight data packets using the adjacent variance calculations can better reflect the current network conditions. The *ROTTMEAN* is calculated according to the difference between the judgment conditions and the corresponding calculation. But the error control of the wireless link error response will be out of order packets; for that, adaptive-TFRC will not be used to calculate the average *ROTTMEAN* value and the long-term average *ROTTMEAN* sample, so as to effectively improve the accuracy of the algorithm.

In the Adaptive-TFRC algorithm, the first step it to calculate the data packet one-way delay $ROTT_i$. In formu-

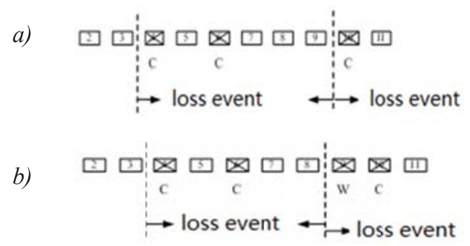

Fig. 1. Diagram of packet loss interval:
a — without wireless packet dropout; *b* — with wireless packet dropout

la (3), S_i is the timestamp generated for sending i packets, R_i is the timestamp that is recorded when the *i*-th packet arrives at the receiver. The $ROTT_i$ value is obtained by sending timestamp packet i arrival time at the receiving end minus the packet i at the transmitting end.

$$ROTT_i = R_i - S_i. \quad (3)$$

Then, the weighted average *ROTTMEAN* of the long-term sampling is obtained at the receiving end, and the transmission rate fluctuation due to the large change of the value can be avoided. Equations (4) and (5) are adopted for calculation:

$$ROTTMEAN_n = \mu * ROTTMEAN_{n-1} + (1-\mu) ROTT_n \ . \quad (4)$$

$$\mu = \begin{cases} 1 & (|ROTT_n - ROTTMEAN_n| > \sqrt{\eta S_{n-1}^{\ 2}}) \\ 0.95 & (|ROTT_n - ROTTMEAN_n| \le \sqrt{\eta S_{n-1}^{\ 2}}) \end{cases}. \quad (5)$$

In equations (4) and (5), ROTT is the measured one-way delay value, and S^2_{n-1} is the sample variance obtained by weighting the ROTT of the last eight packets. The value of μ as shown in formula (5), the value of the coefficient of determination is extremely important, η because of a small packet loss is divided into increased possibility of reordering packet loss, η selectively lead to packet loss is divided into congestion or link error caused by packet loss probability increases. The experiments prove that the η value should be 1.55~1.75, this paper selects η =1.65.

Formula (6) shows a packet loss discrimination parameter τ, which is defined as the ratio of the current ROTT value to the weighted average of the previous long-term sample.

$$\tau = ROTT_n / ROTTMEAN_{n-1}. \quad (6)$$

where $ROTTMEAN_{n-1}$ is the weighted average of the previous long-term sample, $ROTT_n$ is the current one-way delay value. When $\tau > 1$, the network is considered to be in a congestion state, and the corresponding packet loss would be considered as congestion loss, and the sender would need to reduce the transmission rate. When $\tau \le 1$, the network is in non-congestion state, and the packet loss is considered a corruption; this should be in accordance with a certain proportion of the current expansion factor loss interval in order to increase the transmission rate of the transmitter and ultimately improve the network throughput.

Considering the instantaneous jitter of the network, we propose formula (7) to correct formula (6). The main idea is to smooth the noise in the network measurement using a weighted moving smooth filter, that is:

$$\omega_n = \gamma \omega_{n-1} + (1-\gamma)\tau_n, \quad (7)$$

where γ is 0.125 when the network packet loss is detected, at the time of $\omega_n > 1$, the corresponding packet loss is congestion packet loss, and the sender should reduce the transmission rate; at the time of $\omega_n \le 1$, it shows that the corresponding packet loss is a wireless error packet loss, and the current packet loss interval should be expanded according to a certain scaling factor to increase the sending rate of the current sender.

According to the formula of TCP Veno, the adjustment factor of C_i standard for packet loss interval should be 3, so the set adjustment factor segmentation function should take 3 as the segmentation point. At the time of $\omega_n > 1$, the network is in a congestion state, and the adjustment factor, the value of the piecewise function, should be between [0, 1], thus reducing the packet loss interval and ultimately reducing the transmission rate. At the time of $\omega_n \le 1$, the network is in a non-congestion state, the value of the adjustment factor segmentation function should be between [1, 3], thus increasing the packet loss interval and ultimately increasing the transmission rate.

The adjustment factor piecewise function $C(t)$ is defined in formula (8) as

$$C(t) = \begin{cases} e^{-(\omega_n - 1)^2/(2\sigma^2)}, & \omega_n > 1 \\ -2\omega_n + 3, & \omega_n \le 1 \end{cases}. \quad (8)$$

where σ is the minimum; when $\omega_n > 1$, $C(t) \in [0, 1]$, one should reduce the current packet loss interval; when $\omega_n \le 1$, $C(t) \in [1, 3]$, one should increase the current packet loss interval.

At this point, the formula (2) for calculating the rate p of packet loss can be updated as:

$$p' = \sum_{i=0}^{n} \omega_i \Big/ \sum_{i=0}^{n} C(t) C_i \omega_i. \quad (9)$$

2. Formula for the improved TFRC throughput model in the wireless network environment

The TFRC mechanism adopts the formula for the TCP Reno throughput model to adjust the transmission rate, thereby reflecting its friendliness to the TCP. However, if wireless channels are suddenly affected by noises and interferences, wireless links will experience a high bit error rate (BER). Packet loss caused by bit errors or congestion cannot be distinguished using the TFRC mechanism. When pack loss occurs on the network, the transmission end will unintentionally reduce the transmission rate too much, causing a significant decrease in the throughput, especially when the network is severely congested. To address this issue, let us revise the original TFRC throughput model formula by dividing the latter part of formula (1) by m. If the pack loss rate p' is low, the latter part of formula (1) can be neglected, increasing the transmission rate at the transmitting end. If the pack loss rate p' is high, a constant m is added to the latter part to ensure TCP-friendliness if the network is seriously congested. The new formula below is the one which is more suitable for wireless transmissions.

$$X' = \frac{S}{R\sqrt{\frac{2bp'}{3}} + \frac{1}{m} t_{RTO} \cdot \left(3\sqrt{\frac{2bp'}{8}}\right) p'(1+32p'^2)}. \quad (10)$$

In formula (10), X' indicates the transmission rate at the transmitting end, S indicates the size of data packets, R indicates the RTT, and t_{RTO} indicates the retransmission timeout (RTO). Generally, $t_{RTO} = 4RTT$, b is 1, p' is the packet loss event ratio when the wireless environment changes, and m is a constant. Lab tests show that, when m is 1.5, TFRC can totally play its TCP-friendly role.

Simulation results and analysis. Using the NS2 network simulation result and single-bottleneck network topology, a simulation test is performed to analyze TCP-friendliness and overall network performance of TFRC and Adaptive-TFRC in the wireless network.

In the simulation test, the well-known dumbbell network topology is used. The packet loss rate is 1%. As shown in Fig. 2, R0 and R1 are the nodes at the transmitting end, R2 and R3 are the routing nodes, and R4 and R5 are the nodes at the receiving end. The TFRC protocol agent and Adaptive-TFRC protocol agent are respectively bounded to R0 and R4. The TCP background stream is bounded between R1 and R5. The TCP stream, TFRC stream, and Adaptive-TFRC stream form a competitive relationship. The link between R2 and R3 is a bottleneck link. The bandwidth of all other links is greater than that of the bottleneck link, but the delay is smaller than the latter one. DropTail management applies to the nodes. The bandwidth and delay of the bottleneck link are 1.6 Mbit/s and 10 ms. The bandwidth and delay of other links are 2 Mbit/s and 5 ms. The size of each data flow group is 1000 bytes. The ACK size is 40 bytes. The simulation duration is 100 s. This simulation test mainly analyzes the Adaptive-TFRC performance based on the average throughput and justice in different pack loss rates.

1 Analysis of throughput performance

The TFRC protocol agent and Adaptive-TFRC protocol agent are respectively bounded to R0 and R4. The TCP background stream is bounded between R1 and R5. The TCP stream is transmitting at second 0. The TFRC stream and Adaptive-TFRC stream are transmitting at second 5. The two network configurations are completely the same, and the simulation duration is set to 100 s. Fig. 3 and Fig. 4 show the network throughput.

As shown in Fig. 3, when the TCP stream and TFRC stream both exist, the average throughput of the TCP stream and the TFRC stream is 934 kbit/s and 615 kbit/s, respectively. Fig. 4 shows that when the Adaptive-TFRC protocol agent is bounded between R0 and R4, the average throughput of the TCP stream and Adaptive-TFRC stream is 900 kbit/s and 690 kbit/s, respectively. By comparing Fig. 3 and Fig. 4, we find that the average throughput of the Adaptive-TFRC stream improves by 75 kbit/s in contrast with the TFRC stream, but the average throughput of the TCP flow reduces by 34 kbit/s. Therefore, with the Adaptive-TFRC protocol introduced, the overall throughput increases. Compared with the

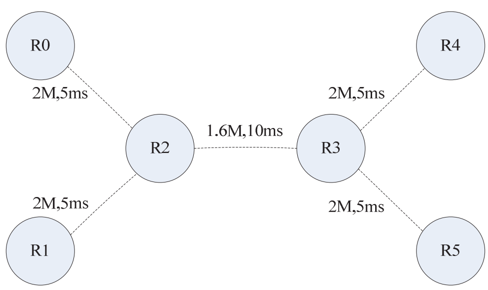

Fig. 2. Network topology

TFRC protocol, the Adaptive-TFRC protocol provides better friendliness to the TCP protocol.

Based on the same network configurations, the TFRC protocol agent and Adaptive-TFRC protocol agent are respectively between R0 and R4 to perform several simulation tests. Table 1 lists the comparison results for network throughput.

Table 1 shows that the throughput ratio between the TCP stream and the TFRC stream is 0.658. When the Adaptive-TFRC algorithm is used, the throughput ratio is 0.767. This reflects that Adaptive-TFRC maintains

Table1. The comparison results for network throughput

Algorithm	Average throughput of TFRC (kb/s)	Average throughput of TCP (kb/s)	Total throughput of network (kb/s)
TFRC	615.1	934.4	1549.6
Adaptive-TFRC	690.2	900.6	1590.8

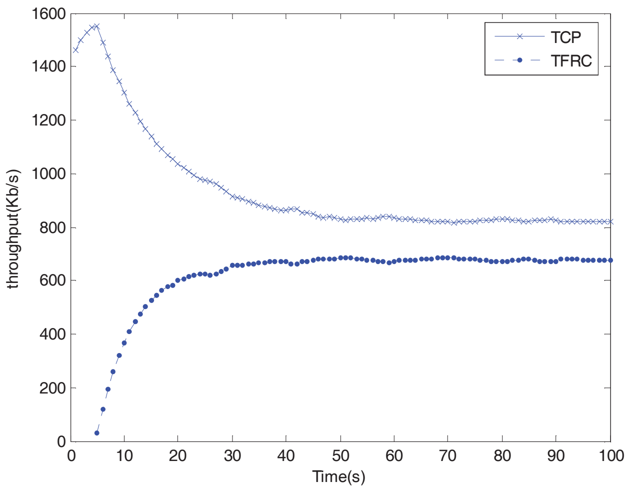

Fig. 3. Network throughput comparison between TCP and TFRC

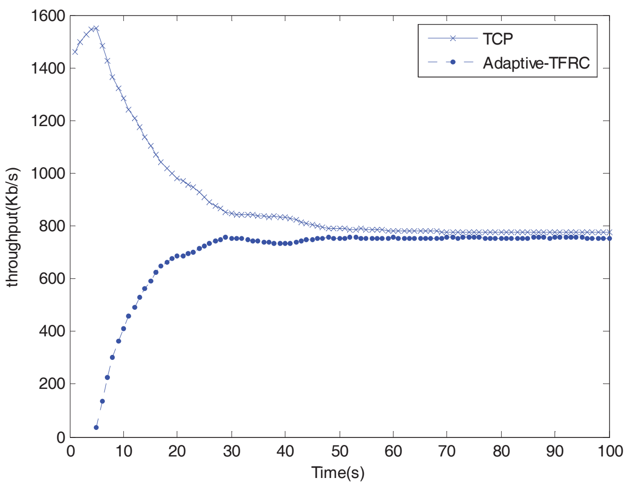

Fig. 4. Network throughput comparison between TCP and Adaptive-TFRC

better friendliness to the TCP. The network throughput increases by 41.3 kbit/s compared to the result when the TFRC algorithm is used. This is because the improved TFRC protocol can accurately identify pack loss and adjust the transmission rate in a timely manner to improve the overall network throughput.

2. Analysis of delay jitter changes

Fig. 5–6 show the delay jitter changes of the Adaptive-TFRC stream and TFRC stream based on the same network configuration and environment. The average delay jitter of the TRFC stream and Adaptive-TFRC stream is 0.006 s and 0.003 s, respectively, which indicates that the Adaptive-TFRC algorithm provides better real-time performance, transmission stability, and more effective pack loss control, as well as is more suitable for wireless streaming media transmission.

3. Analysis of wireless bandwidth utilization

In simulation, the error rate of wireless link is 1%~5%. The relationship between BER and bandwidth utilization of wireless link is calculated by AWK script, as shown in Fig. 6.

As you can see from Fig. 7, when the Adaptive-TFRC algorithm is applied, the bottleneck bandwidth utilization of the network has been maintained at more than 85%. However, when using the TFRC algorithm, the network bottleneck bandwidth utilization rate decreases with the increase of error rate. When the bit error rate is from 1% to 2%, the network is in a congestion state, and the bottleneck bandwidth utilization can be maintained at a higher level of more than 90%. When the bit error rate reaches 3%, the end-to-end packet loss rate increases linearly with the bit error rate of the wireless link. Due to the lack of effective mechanism of network loss differentiation, we can not make accurate judgment on the network state, resulting in a sharp decline in network bottleneck bandwidth utilization. When the bit error rate rises to 5%, the TFRC algorithm of network bottleneck bandwidth utilization is maintained at 86%; TFRC algorithm network bandwidth bottlenecks in the application and utilization rate dropped to 53%. The above data show that, compared with the TFRC algorithm, the network bottleneck bandwidth utilization rate is much higher after using Adaptive-TFRC. This is because Adaptive-TFRC is based on packet loss discrimination mechanism, which can effectively distinguish the link error, packet loss and network congestion loss, and has a dynamic self-adaptive transmission rate regulation mechanism to improve network utility.

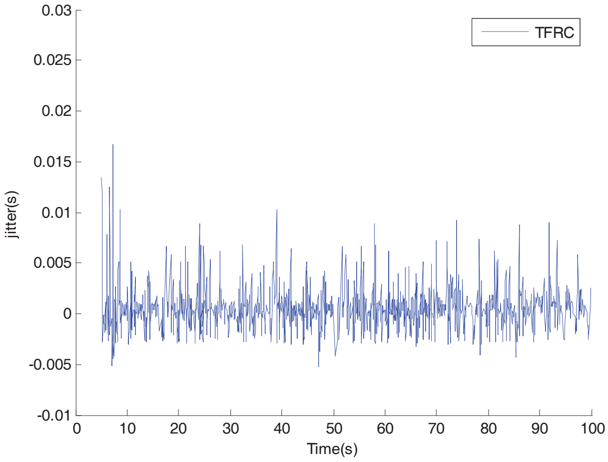

Fig. 5. Delay variation of TFRC

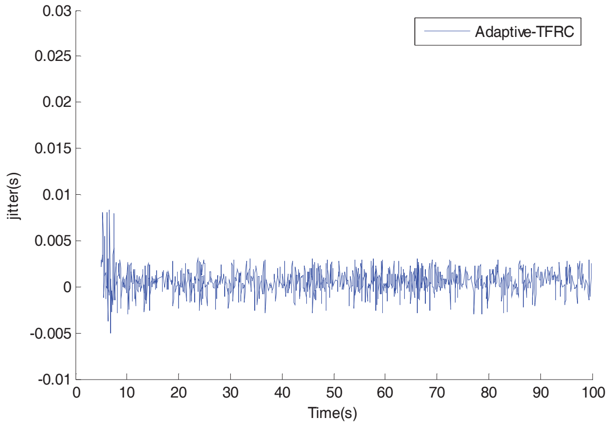

Fig. 6. Delay variation of Adaptive-TFRC

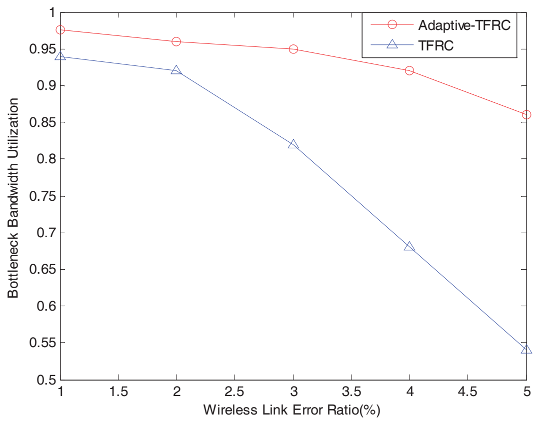

Fig. 7. Relationship between wireless link error ratio and its bandwidth utilization

CONCLUSION. This paper presents an Adaptive-TFRC mechanism for wireless networks. When it is applied to streaming media transmission, the mechanism uses the packet loss to distinguish parameters from the receiver and reflect the state of the network, and then form a feedback to the sending end. Meanwhile, the throughput model of the classical TFRC mechanism is improved so that the transmission rate can be adjusted dynamically. The experimental results show that the adaptive-TFRC mechanism improves the overall network throughput performance and maintains the efficient use of link TCP business-friendly at the same time. The network bottleneck bandwidth utilization rate has remained at a high level, and the delay jitter of real-time traffic is reduced so that it is proved to possess better real-time transmission and stability. Thus, the QoS of real-time streaming media with multi protocol coexisting in wireless networks can be guaranteed.

Статью рекомендует в печать
д-р техн. наук, проф. *Г. В. Павлов*