This regular paper was presented as part of the main technical program at IFIP WMNC'2013

Dynamic Parallel TCP For Scalable Video Streaming Over MIMO Wireless Networks

Avinash Kumar Chaurasia*†, Aditya K. Jagannatham†
Email: avinash.aviank2003@gmail.com, adityaj@iitk.ac.in
*Student author
†Indian Institute of Technology, Kanpur, India

***Abstract*—Video streaming over wireless (networks) has gained popularity in recent times with the development of 3G and 4G mobile networks. The demand and market for video based services is expected to grow substantially in the future. When UDP is used for video streaming, the inherent unreliability of UDP frequently results in corrupted video packets and jitter during video playback. Such an aberration is especially pronounced in wireless networks due to the fading and multipath interference impediments in wireless channels. TCP based video transmission provides reliability at the cost of severely increased delay arising from frequent retransmissions in case of packet error over wireless links and constraints from session and application layers for in-sequence delivery. These delays are substantially worsened in low SNR (signal-to-noise power ratio) wireless environments. Multiple TCP connections, which efficiently utilize the bandwidth of the wireless channel, have been demonstrated to yield significantly improved quality of video streaming over wireless channels while enhancing the reliability of video delivery compared to UDP. In this paper we propose a new scalable hierarchical wavelet decomposition based layered video coding over parallel TCP scheme for reliable video transmission over MIMO wireless channels while reducing the delay and jitter of video streaming.**

I. Introduction

Cellular networks, starting from their nascent beginning limited to a few thousand users, have spawned across the globe to provide wireless connectivity to billions of users. The evolved packet core (EPC) in 4G networks such as LTE enables a vast array of broadband services for mobile users, thus establishing a flat IP architecture based packet routing as the overbearing mode for data delivery in such networks.

However, compared to conventional wired networks, communication over wireless networks is significantly more challenging due to the erratic received signal quality over the interference limited radio channel. Several revolutionary PHY layer wireless technologies have been developed lately to combat the ill effects of fading in radio channels and enable high data rate transmission over wireless channels. Multiple-Input Multiple-Output (MIMO) [1], Orthogonal Frequency Division Multiplexing (OFDM), Orthogonal Space-Time Block Codes (OSTBC) have radically enhanced the data rates possible over modern wireless networks. In addition, *Diversity* combining [2] techniques have enabled reliable signal transmission over the erratic fading wireless channel.

Being modular in nature by design, TCP and higher layers in the TCP/IP stack are transparent to this radically transformed wireless PHY/ MAC layers. However, the drastically changed scenario in wireless networks introduces several inefficiencies in the TCP and higher layer mechanisms. For instance, as elucidated in [3], packet error and not packet drop is the dominant cause for discarding packets at the LLC (Logical Link Control) over a radio channel. This severely increases the delay arising due to AIMD (Additive Increase/Multiplicative Decrease) based congestion avoidance in TCP. This problem is further worsened in real time delay constrained content such as video transmission in wireless networks. Hence, implementing QoS guarantees such as jitter and latency in streaming video applications and integrating such a paradigm in the existing OSI protocol layers over modern 4G networks is a key challenge in modern packet cores.

Towards this end, H.264 based Scalable Video Coding (SVC) has been gaining ever increasing appeal for video transmission over wireless channels. The hierarchical layering based structure of the digitally encoded video streaming renders it ideal for video transmission over the fading wireless channel since its rate can be dynamically adapted to the time-varying radio environment. Parallel TCP has been demonstrated to be well suited for reliable video transmission over packet switched wireless systems. The performance of parallel TCP for wireless video transmission can be significantly enhanced through Scalable Video Coding and modern PHY layer techniques such as MIMO. Hence, in this paper we present a framework for multiple TCP based scalable layered video transmission for video jitter and latency reduction in wireless video streaming. Further, we consider a broadband MIMO wireless system for high data rate video transmission to significantly enhance resilience of the PHY layer. Below, we present a brief overview of the related research in TCP and video QoS over wireless packet networks.

Paper outline: The remainder of the paper is organized as follows. In section II we give a brief background on parallel TCP and video transmission over wireless. We present our technique in section III. Section IV shows the comparison between different schemes. We describe the experiments performed over ns2 in section V and we conclude in section VI.

II. Overview of Parallel TCP and Video over Wireless

TCP based congestion control is especially suited for standard coaxial cable or fibre based networks, which have higher reliability compared to fading wireless links multiplexed over

978-1-4673-5616-9/13/$31.00

the shared frequency spectrum as explained in the previous section. As a result, packet error rate over a wireless link is significantly higher due to the burst error characteristic of the fading wireless channel. Several techniques such as I-TCP(indirect TCP) [4], Link level retransmissions [5], Snoop protocol [6] have been proposed in literature to improve the performance of TCP over wireless networks. However, these schemes distort conventional TCP semantics. Other schemes such as Adaptive Frame length Control [7], Parallel TCP for packet transmission over wireless networks [8] are much more suited for practical implementation since they enhance the performance of TCP over wireless networks without violating the TCP semantics. We describe some of these schemes in detail below.

A. Adaptive Frame length Control [7]

This scheme is implemented at the data link layer in the network protocol stack. It is based on the following motivation. The probability of bit-error for transmission across wireless channels is significantly high. In such scenarios, high packet corruption rate leads to retransmission of entire packets, leading to wastage of bandwidth. Further, it is important to note that the probability of packet error increases with the packet length. As the packet length increases, the probability of retransmission increases, with loss of efficiency. Hence, packet length adaptation in sync with the signal strength of the fading wireless channel is key to efficient bandwidth utilization in wireless scenarios.

B. Parallel TCP for video transmission over wireless network [8]

Streaming videos over UDP suffers from frame drops as it does not consider retransmission on packet loss, thereby affecting the reliability. Several works have focused on employing TCP for video streaming. However, employing a single TCP connection may lead to intolerable jitter in case of congestion, especially in bandwidth constrained and error prone wireless networks. Several research works have tried to address the problem of reducing the jitter in video streaming applications. It has been shown that parallel TCP is ideally suited in such an endeavor. Below we describe the specific mechanism and properties of parallel TCP.

1) Parallel TCP: It has been demonstrated in works such as [9], [10] that parallel TCP substantially improves the rate of transmission and efficiency of bandwidth utilization over wireless links. This arises due to the following reason. Conventional transmission over a single TCP connection requires some amount of time to reach the optimum congestion window size. In fact, the larger the bandwidth delay factor of the link, the higher is the time required to reach the optimum congestion window size. Thus, the bandwidth is typically not utilized with maximum efficiency. On the other hand communication links are prone to random losses of transmitted packets. Hence, instead of data transmission over a single TCP connection, data fragmentation and transmission over multiple TCP connections leads to an effective reduction in the probability of random packet loss by a factor of N as described in [9]. Further, parallel TCP connections also help in reducing the bandwidth delay product by a factor of N, where N is the number of parallel streams. For N TCP connections, the initial window size starts with N instead of 1 window as in the case of a single TCP connection (shown in Fig.1). Moreover, the time required to reach the optimum congestion window size is significantly reduced. Shivkumar et. al. in [10] have demonstrated that higher throughput can be achieved with multiple TCP connections by optimizing the number of TCP connections according to the nature of the fading wireless environment.

In a wired medium, once the single TCP reaches its optimum congestion window size, a decrease in window size occurs only in the case of congestion. However, in a wireless medium, packet corruption rates often are extremely high owing to the poor signal quality over faded wireless channels. Hence, it can be observed that in the case of parallel TCP connections, if one of the constituent TCP connections enters the congestion control phase, the bandwidth released by this connection is utilized by the other TCP constituents. This leads to efficient utilization of the wireless bandwidth, thus motivating the use of parallel TCP over wireless links.

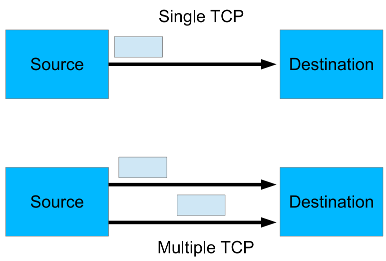

Fig. 1. Data transfer over parallel TCP

2) Performance of video transmission over parallel TCP: In parallel TCP, increasing the number of TCP connections up to an optimal number increases the video frame rate. Further increase in the number of TCP connections leads to congestion in the network and results in performance degradation as explained in [8]. Parallel TCP is well suited for high bandwidth links compared to low bandwidth ones [8]. Conventional single TCP connections result in acceptable performance over low bandwidth links. Another parameter that varies significantly with the number of TCP connections is the frame rate distribution. With a large number of TCP connections, the frame rate has a high rate of variation with time, whereas for a lower number of TCP connections, this distribution remains constant over time.

3) DTCP: DTCP (dynamic TCP) [8] is employed to adapt the number of parallel TCP connections in multiple TCP scenarios based on an additive increase multiplicative decrease (AIMD) paradigm. It initializes the process with a single TCP connection and progressively increases the number of TCP connections by one until congestion is not detected or a predefined maximum TCP connection limit is reached. On detection of network congestion, it reduces the number of TCP connections by half and restarts the process of increasing the number of TCP connections by one.

III. Parallel TCP for Wireless Scalable Video Transmission

As described above, wireless links are highly prone to packet error due to the high RF noise floor in wireless devices and the severe multipath interference in the radio channel. Further, as seen in the description of Parallel TCP in section II-B1, use of multiple parallel TCP connections provides enhanced throughput over wireless links. Further, since we are employing parallel TCP in conjunction with multi-antenna MIMO wireless systems, each separate antenna can employ an independent TCP connection. However, the number of TCP connections need not be restricted by the number of antennas. The limited power at the transmitter has to be distributed between the multiple N_t transmit antennas. Further, one needs to restrict the inter frame arrival delay so as to reduce the jitter of video playback at the receiver. The proposed scheme for parallel TCP based scalable layered video transmission consists of the following sequence of encoding operations.

- Hierarchical Wavelet Layering of Video Frames.
- Motion Estimation and Prediction.
- DCT and Quantization
- Variable Length Coding (VLC)

As shown in Fig.2, application of the 2-D Haar wavelet transformation [11] on the target video frame leads to a hierarchical base and enhancement layer decomposition of the frame. Also, the base layer contains energy compacted information of higher significance compared to the enhancement layer, which contains high frequency component data of lower significance. Subsequent to the decomposition, the individual layers are compressed temporally through *motion estimation* followed by differential temporal encoding, *DCT* [12], *Quantization* [13] and *VLC* stages respectively. We employ the standard MPEG-2 encoding [14] quantization table during the quantization phase. On completion of all the above phases, the encoded video base layer comprises of the most significant video information and is transmitted over the TCP connection of lowest index at a higher SNR and the enhancement layer over the second TCP connection at lower SNR. At the receiver, on successful reception of the video frame packets, the video decoder repeats the above stages in reverse order to rebuild the original video frames using the received video layers. The proposed application employs a sequential number for each TCP connection so that the receiver is aware of all the packets that have been lost due to corruption. Towards this end the application layer sets a field for the start of each frame. At the receiver, if any enhancement layer frame overshoots its expected arrival time, while the base layer packets have been successfully received, the receiver replies with a `KILL` signal on the second parallel TCP component by setting the first reserved bit in the TCP header as 1. After receiving the `KILL` signal, the sender drops all the packets related to the enhancement layer of the current frame so that it can commence transmission of the enhancement layer packets of the subsequent frame. This procedure can be seen to significantly reduce delay and jitter of video streaming at the receiver by adaptively scheduling a combination of base only and base-enhancement layers as per the packet drop rate and wireless link quality. Implementation of the `KILL` signal requires modification of the sender and receiver side TCP while maintaining the semantics of TCP and those of the lower layers. On transmission of the `KILL` signal, the receiver notifies the application layer regarding the non-receipt of enhancement layer packets. These are then substituted by the NULL field during playback. Since the enhancement layer components consist information of lesser significance, it results in lower degradation of video quality while tremendously reducing the delay and enhancing the reliability of video transmission in wireless fading scenarios.

IV. Comparison of Proposed and Existing Schemes

In this section we describe existing schemes such as DCCP and Dynamic Adaptive Streaming for real time video streaming applications in packet networks.

A. Dynamic Adaptive Streaming

Dynamic Adaptive Streaming is an application layer technology with TCP as the underlying transport layer mechanism. This scheme is designed to stream video by dynamically switching the streams according to the bandwidth available to the user [15]. This technology has several drawbacks as described below. Dynamic adaptive streaming requires multiple bit-rate video sequences already encoded and stored at the server side and hence can not be used in live broadcast video streaming scenarios. Further, the storage of video sequences at several rates tremendously increases the storage space required at the streaming server. Since it is designed to wait for the next keyframe, if the rate of the fading wireless channel is lowered momentarily, the playback at the client is halted, leading to jitter in the received video. Moreover, it switches between different bit-rate video sequences based on bandwidth, whereas the key criterion in wireless channel is packet corruption resulting from burst errors and not network congestion. Hence, the efficiency of bandwidth utilization will degrade significantly leading to lower quality video at the user end.

B. Dynamic Congestion Control Protocol(DCCP)

DCCP [16] is designed for applications such as streaming which can trade off between the reliability and delay of the streamed video. It employs two types of congestion control namely: TCP-Friendly rate control (TFRC) [17] and TCP-like congestion control [18]. These schemes, which were primarily designed for wired networks, inherently have drawbacks similar to the conventional schemes and perform poorly in wireless conditions with high levels of noise and interference.

V. Experiments and Results

We implemented the proposed scheme for parallel TCP based scalable video transmission on ns-2. We tested and

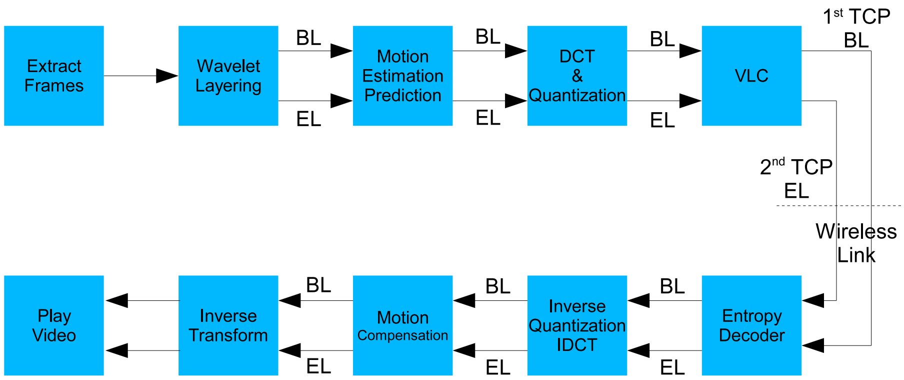

Fig. 2. Scheme for video transmission over 2 × 2 MIMO channel with 2 TCP connections. BL = base layer and EL = enhancement layer.

compared our algorithm for the three standard video sequences *Akiyo*, *Foreman* and *Coastguard*. We considered the Quarter Common Intermediate Format (QCIF) i.e. 176×144 resolution versions for each sequence. The following parameters are employed as the basis of the performance comparison. These are standard comparison metrics for video streaming employed in works such as [8].

1) Total Delay Time (D_{all}): Sum of the delay time of all the delayed frames.
2) Maximum Delay Time (D_{max}): The maximum delay time of a frame during playback.
3) Average Delay Time for All Frames (D_{avg}): Total Delay Time divided by the Total Number of Frames Played.
4) Standard Deviation of Delay Time for All Frames (σ_{all}): Standard deviation of frame delay which includes frames received without any delay.
5) Average Delay for Delayed Frames (D_{davg}): This is similar to the average delay time for all frames except that it excludes the frames received without delay.
6) Standard Deviation of Delay Time for Delayed Frames ($\sigma_{delayed}$): This is similar to the standard deviation of delay time for all frames but only accounts for the delayed frames.
7) Number of Frames Played on Time (N_{ontime}): Frames that are played without delay.
8) Number of Frames Delayed ($N_{delayed}$): Frames that are played but delayed.
9) Total Number of Frames Played (N_{all}): Sum of Number of Frames Played on Time and Number of Frames Delayed.
10) Average Frame Rate (F): Mean frame rate.
11) Standard Deviation of Frame Rate (Φ): Standard deviation value of frame rate.

A. Comparison of Scalable and Non-Scalable Video Transmission for Parallel TCP

In this section we compare the video coding schemes based on VLC with wavelet layering and power adaptation with the conventional video coding employing VLC without wavelet layering. The frame rates of the video sequences considered

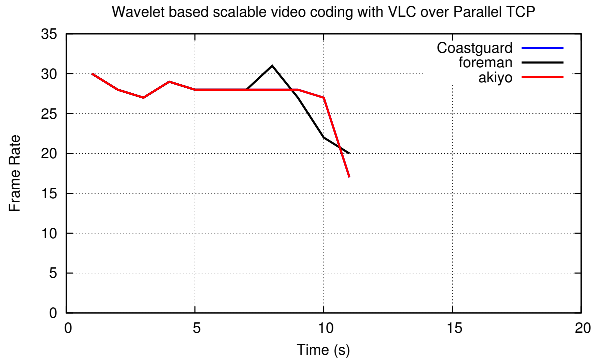

Fig. 3. Frame rate variation for scalable coded video streaming over parallel TCP connections with VLC.

are 25 Fps. For a fair comparison, each scheme employs 2 parallel TCP connections for video transmission. The other ns-2 simulation parameters are as follows. The number of nodes for this wireless transmission scenario is equal to 2. The fading wireless channel is considered to be Rayleigh in nature. The physical layer is simulated as a 2×2 MIMO wireless channel. To reflect the higher significance of the base layer compared to the enhancement layers, the SNR of the 1^{st} TCP connection is set at 25dB while that of the 2^{nd} TCP is at 10dB for both the schemes. The maximum packet size is set to 1000 bytes while the standard AODV protocol is employed for packet routing at the network layer. The bandwidth considered is 1.5 Mbps. Table I shows the performance comparison of the scalable and non-scalable schemes for the *Akiyo* video. It can be seen from the observed metrics therein that the average frame rate of the video improves from 1.17 to 29 when we used our scheme. Table II and Table III show similar results for the *Foreman* and *Coastguard* videos respectively. From Fig.3 and Fig.4 one can observe that the frame rate variation of the proposed scalable layered video transmission scheme is significantly lower compared to the frame rate variation of the non-scalable scheme. The reason for poor performance of the existing non-scalable video transmission scheme can be stated as follows. When a transmitted packet is corrupted, TCP (with

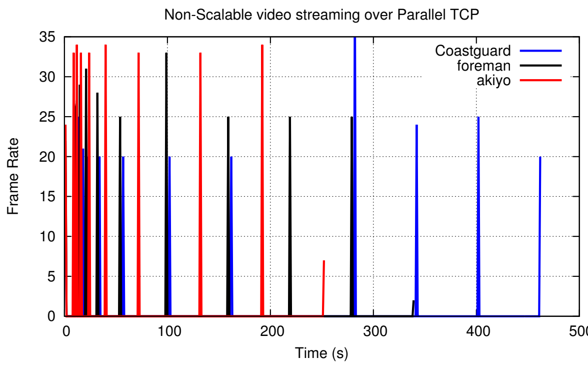

Fig. 4. Frame rate variation for non-scalable video streaming over parallel TCP connections with VLC.

low SNR at both antennas) will drop the packet. Subsequently, TCP starts queuing the packets until the lost packet is delivered successfully. Following this, all the queued packets are sent to the application layer which is indicated by the sudden spikes in Fig.4. However, in the proposed scalable video transmission scheme, since the base layer is transmitted at high SNR, most of the packets for this layer are received error free. As the enhancement layer is sent at low SNR, the unsuccessfully received packets of this layer can be dropped and the video decoder can employ zero padding for error concealment as described earlier with almost no perceptible loss of visual quality. The reception of the base layer frames with a high degree of reliability is the central reason for improvement in frame rate and low frame rate variation. From Fig.3 and Fig.4 one can also observe that total time taken by the proposed scheme is much lower compared to the conventional non-scalable scheme. For instance, from Fig.3 the total time is 11 seconds compared to 255 seconds in Fig.4 for the same number of frames of the *Akiyo* video sequence. The instances when the base layer packets are corrupted is reflected by the jitter in the Fig.3.

From these figures and tables it can also be observed that the proposed scheme significantly reduces the inter frame delay compared to VLC without scalable video coding in multiple TCP connection scenarios over MIMO channels. Also the proposed scheme yields the desired average frame rates for video transmission and successfully plays all the video frames in time as shown by second last row of tables Table I, Table II and Table III, thus minimizing jitter.

B. Video Streaming with Multiple TCP vs Single TCP

In this section we compare the performance of the proposed scalable video coding scheme over parallel TCP with that of non-scalable video coding over single TCP. The frame rate for the considered video sequences is 25 Fps. The parallel TCP simulation setup is similar to the one described earlier. The frame rate variation for the single TCP scenario is shown in Fig.5 for the video sequences *Akiyo*, *Foreman* and *Coastguard*.

From the results for single TCP based video streaming given

TABLE I
PERFORMANCE COMPARISON FOR STREAMING SEQUENCE *Akiyo*

	VLC with Scalable Coding over Parallel TCP	VLC with Non-Scalable Coding over Parallel TCP	VLC coded streaming over Single TCP
D_{all}	0	16407.48	800.99
D_{max}	0	239.41	7.03
D_{avg}	0	55.06	2.69
σ_{all}	0	65.4	2.9
D_{davg}	0	59.88	2.74
$\sigma_{delayed}$	0	66.03	2.91
N_{ontime}	298	24	5
$N_{delayed}$	0	274	293
N_{all}	298	298	298
F	29	1.17	20.58
Φ	0.5	5.89	14.52

TABLE II
PERFORMANCE COMPARISON FOR STREAMING SEQUENCE *Foreman*

	VLC with Scalable Coding over Parallel TCP	VLC with Non-Scalable Coding over Parallel TCP	VLC coded streaming over Single TCP
D_{all}	0	22420.07	1254.94
D_{max}	0	326.80	9.11
D_{avg}	0	75.24	4.21
σ_{all}	0	88.28	3.53
D_{davg}	0	81.23	4.28
$\sigma_{delayed}$	0	89.00	3.52
N_{ontime}	298	22	5
$N_{delayed}$	0	276	293
N_{all}	298	298	298
F	27	0.88	17.99
Φ	1.09	4.67	12.53

TABLE III
PERFORMANCE COMPARISON FOR STREAMING SEQUENCE *Coastguard*

	VLC with Scalable Coding over Parallel TCP	VLC with Non-Scalable Coding over Parallel TCP	VLC coded streaming over Single TCP
D_{all}	0	45437.66	1909.76
D_{max}	0	449.98	12.03
D_{avg}	0	152.48	6.41
σ_{all}	0	156.55	4.53
D_{davg}	0	162.69	6.45
$\sigma_{delayed}$	0	156.51	4.52
N_{ontime}	298	19	2
$N_{delayed}$	0	279	296
N_{all}	298	298	298
F	27	0.65	15.30
Φ	0.5	3.83	9.1

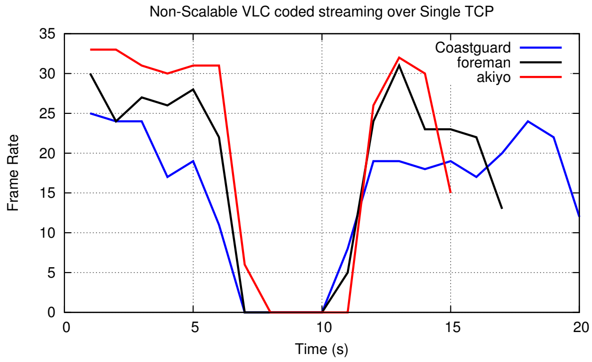

Fig. 5. Frame rate variation of video transmission over Single TCP connection

in Tables I, II, III, it can be seen that the total delay and number of delayed frames is much higher compared to the proposed scheme. This can be observed by comparing the values in second column of Table I with that of the fourth column of same Table for the *Akiyo* video sequence and so on. From these results it can be seen that the proposed scheme has a significantly superior performance compared to single TCP based video transmission. Further, the reason for the poor performance of non-scalable VLC coded video over multiple TCP in terms of total delay compared to single TCP based video streaming is the low SNR on the second TCP connection. This results in a high packet drop rate over the second TCP connection leading to queuing of undelivered packets until the arrival of the lost packets. On the other hand, since the receiver does not have the complete sequenced packets, it can not play until the second TCP successfully receives the corrupted packet. Also, an artifact that can be observed from the tables is that the number of frames played on time with the proposed scheme is better than with a single TCP connection. This arises due to the fact that initially when there are no packet corruptions in the parallel TCP connection, there is continuous playback and hence, zero jitter at the receiver. On occurrence of the first packet drop, the performance of the parallel TCP scheme begins to degrade leading eventually to poor performance.

VI. Conclusion and Future Work

In this work it has been shown that scalable layered coding through hierarchical wavelet layering with power adaptive transmission over MIMO wireless links has superior performance in terms of frame delay, average frame rate and standard deviation of delay compared to the existing scheme for video streaming over single and multiple TCP connections. The proposed scheme can achieve the required frame rate in MIMO wireless channels and strongly reduces jitter. The performance of the proposed algorithm has been comprehensively evaluated employing a simulated MIMO wireless packet network. This work can be extended to employ variable data size packets for transmission and fixed size packets with CRC for variable size intra blocks within the packet so that only the corrupted sub-blocks can be transmitted, improving the efficiency of bandwidth usage. Combining such a transmission scheme with the parallel TCP based scalable video coding scheme described in this work can result in a further enhancement of the quality of video streaming.

References

[1] G. J. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas," *Bell Labs Technical Journal*, vol. 1, no. 2, pp. 41–59, 1996. [Online]. Available: http://dx.doi.org/10.1002/bltj.2015

[2] D. Tse and P. Viswanath, *Fundamentals of wireless communication*. New York, NY, USA: Cambridge University Press, 2005.

[3] M. Chen and A. Zakhor, "Multiple TFRC connections based rate control for wireless networks," *Multimedia, IEEE Transactions on*, vol. 8, no. 5, pp. 1045 –1062, oct. 2006.

[4] A. Bakre and B. Badrinath, "I-TCP: indirect TCP for mobile hosts," in *Distributed Computing Systems, 1995., Proceedings of the 15th International Conference on*, 1995, pp. 136 –143.

[5] S. Paul, E. Ayanoglu, T. La Porta, K.-W. Chen, K. Sabnani, and R. Gitlin, "An asymmetric protocol for digital cellular communications," in *INFOCOM '95. Fourteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Bringing Information to People. Proceedings. IEEE*, apr 1995, pp. 1053 –1062 vol.3.

[6] H. Balakrishnan, S. Seshan, E. Amir, and R. H. Katz, "Improving TCP/IP performance over wireless networks," in *Proceedings of the 1st Annual International Conference on Mobile computing and networking*, ser. MobiCom '95. New York, NY, USA: ACM, 1995, pp. 2–11. [Online]. Available: http://doi.acm.org/10.1145/215530.215544

[7] P. Lettieri and M. Srivastava, "Adaptive frame length control for improving wireless link throughput, range, and energy efficiency," in *INFOCOM '98. Seventeenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 2, 1998, pp. 564 –571 vol.2.

[8] H.-H. Wan and Y.-K. Kwok, "High data rate video transmission using parallel TCP connections: approaches and performance evaluation," *J. Supercomput.*, vol. 35, pp. 119–139, February 2006. [Online]. Available: http://portal.acm.org/citation.cfm?id=1145928.1145929

[9] J. Lee, D. Gunter, B. Tierney, B. Allcock, J. Bester, J. Bresnahan, and S. Tuecke, "Applied techniques for high bandwidth data transfers across wide area networks," in *In Proceedings of International Conference on Computing in High Energy and Nuclear Physics*, 2001.

[10] H. Sivakumar, S. Bailey, and R. Grossman, "Psockets: The case for application-level network striping for data intensive applications using high speed wide area networks," in *Supercomputing, ACM/IEEE 2000 Conference*, nov. 2000, p. 38.

[11] S. Theodoridis and K. Koutroumbas, *Pattern Recognition, Third Edition*. Orlando, FL, USA: Academic Press, Inc., 2006.

[12] N. Ahmed, T. Natarajan, and K. Rao, "Discrete cosine transform," *Computers, IEEE Transactions on*, vol. C-23, no. 1, pp. 90 –93, jan. 1974.

[13] R. M. Gray and D. L. Neuhoff, "Quantization," *IEEE Trans. Inform. Theory*, vol. 44, no. 6, pp. 2325–29, 1998.

[14] J. Lee, "Rate-distortion optimization of parametrized quantization matrix for MPEG-2 encoding," in *Image Processing, 1998. ICIP 98. Proceedings. 1998 International Conference on*, vol. 2, oct 1998, pp. 383 –386 vol.2.

[15] D. Hassoun, "Dynamic streaming in flash media server 3.5 part 1: Overview of the new capabilities." [Online]. Available: http://www.adobe.com/devnet/flashmediaserver/articles/dynstream_advanced_pt1.html

[16] E. Kohler, M. Handley, and S. Floyd, "RFC 4340: Datagram congestion control protocol (DCCP)," 2006. [Online]. Available: http://tools.ietf.org/html/rfc4340

[17] E. Kohler, J. Padhye, and S. Floyd, "RFC 4342: Profile for datagram congestion control protocol (DCCP) congestion control id 3: TCP-Friendly Rate Control (TFRC)," 2006. [Online]. Available: http://tools.ietf.org/html/rfc4342

[18] E. Kohler and S. Floyd, "RFC 4341: Profile for datagram congestion control protocol (DCCP) congestion control id 2: TCP-like congestion control," 2006. [Online]. Available: http://tools.ietf.org/html/rfc4341