This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2010 proceedings.

Quality Improvement of Mobile Video Using Geo-intelligent Rate Adaptation

Jun Yao, Salil S. Kanhere and Mahbub Hassan
School of Computer Science and Engineering
University of New South Wales
Sydney, Australia 2052
Email: {jyao, salilk, mahbub}@cse.unsw.edu.au

***Abstract*—Adaptive video is a popular technique to continuously deliver a video stream to a user in the best quality possible when the underlying network bandwidth cannot be guaranteed. As such, quality of adaptive video depends critically on the agility of the rate adaptation algorithms in tracking the varying bandwidth. In this paper, we investigate the performance of a popular rate adaptation algorithm, namely, TCP-friendly rate control (TFRC), in vehicular environments. Our results show that TFRC cannot cope well with the pattern of bandwidth changes faced by a user travelling in a fast moving vehicle, resulting in poor viewing experience. Motivated by the observation that bandwidth changes in vehicular environment is significantly influenced by the rapid change of user's geographic location, we propose *Geo-TFRC*, which empowers TFRC with a novel street-level bandwidth map that holds summary of past bandwidth observations for each segment of the street. We conduct simulation experiments which are driven by the real High-Speed Downlink Packet Access (HSDPA) bandwidth traces collected from a vehicle traveling along a route in Sydney. Our results reveal that Geo-TFRC can track the bandwidth changes much more effectively, which in turn improves the quality of the mobile video. We find our proactive approach can significantly reduce the time that a user suffers from pixelated viewing experience by up to *five* folds as compared to TFRC.**

I. Introduction

Streaming a video over the Internet where the available bandwidth cannot be guaranteed is a challenging problem. Adaptive video is a promising technology that addresses this problem by dynamically and transparently changing the video bit-rate and quality to adjust to the changes in the available bandwidth. The quality of adaptive video depends largely on its ability to track the available bandwidth, so the most appropriate bit-rate can be selected all the time. Several algorithms have been proposed in recent years for adaptive video streaming [1]–[5], with TCP Friendly Rate Control (TFRC) [6] being a popular choice. These algorithms are reactive in nature, as they monitor certain end-to-end network parameters (e.g., delay, jitter, packet loss) and adapt the bit-rate based on the changes in the observed values. While TFRC has been shown to perform well for traditional desktop applications [7], its performance in the mobile context, especially in the fast moving vehicular environment, is largely unknown.

In this paper, we study the performance of TFRC when a user is travelling in a fast moving vehicle and the available bandwidth is mostly dictated by the wireless link between the user and a nearby base station. In such context, the available bandwidth can vary significantly from location to location, even at the granularity of a few hundred meters on the same street [8]. The location influence on the available wireless bandwidth can be attributed to the fact that the wireless base stations dynamically select their channel coding (or modulation) and data rate to accommodate the varying radio propagation characteristics, e.g., the distance between the user and the base station, multi-path fading, and co-channel interference [9]. Since a vehicular user changes its location rapidly, the available bandwidth can vary significantly. How to adapt effectively to the bandwidth variation patterns of such vehicular environments, so as to achieve a better video experience, is the core issue addressed in this paper.

We first conduct simulations of MPEG-4 Variable Bit Rate (VBR) video streaming using the bandwidth traces of a High-Speed Downlink Packet Access (HSDPA) network on a 7Km route in Sydney, Australia, which uses TFRC for adaptation purposes. Our simulations show that TFRC cannot cope well with this bandwidth pattern resulting in poor video viewing experience during the trip. Motivated by the observation that bandwidth changes in vehicular environment is significantly influenced by the rapid change of geographic location of the user, we propose Geo-TFRC which empowers TFRC with novel street-level *bandwidth-road maps* that holds summary of past bandwidth observations for each segment of the street. We use the bandwidth traces from repeated trips to construct the bandwidth-road maps for the 7Km route and simulate the same video stream with Geo-TFRC, which consults the bandwidth map as the vehicle moves along the route. We demonstrate that the video quality is enhanced significantly as a result of access to the bandwidth-road maps. More specifically, we find our proactive approach can effectively reduce the time that a user suffers from pixelated viewing experience by up to *five* folds as compared to TFRC.

The rest of the paper is organized as follows. We review the background of adaptive video streaming, TFRC, and bandwidth-road maps in Section II. In Section III, we present the design of Geo-TFRC. Trace-driven simulation setup and results are presented in Section IV. Finally, Section V concludes this paper.

II. Background

In this section, we first provide a brief overview of the current state-of-the-art in adaptive streaming. Following this we present a overview of TFRC and highlight its drawbacks

978-1-4244-6398-5/10/$26.00 ©2010 IEEE

in an environment where the bandwidth fluctuates significantly. Lastly, we present a brief review of bandwidth-road maps.

Adaptive Video Streaming. The current popular approach for serving video content over the Internet is static. The bit-rate and quality of the streaming is manually selected by the user, and fixed during the whole playback. In this approach, the stream will experience substantial packet loss or rebuffering, if the available bandwidth along the end-to-end path reduces significantly. This leads to jerky video playback and ultimately impacts the user viewing experience.

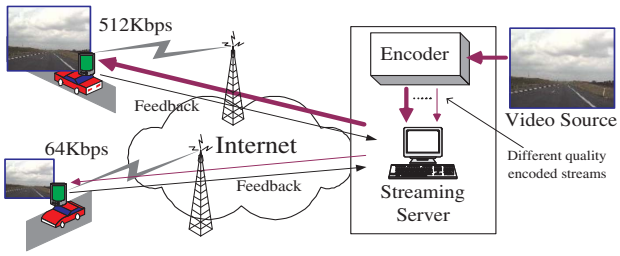


Fig. 1. Adaptive video streaming

Adaptive streaming solves this issue by dynamically modifying the bandwidth of the video stream. Fig. 1 illustrates a typical adaptive streaming setup. The streaming server pre-encodes the source video file (or live event) at multiple resolutions and bit-rates. During playback, the streaming server can transparently change the bit-rate of the video to seamlessly adjust to the changes in the available bandwidth, by switching the stream between those pre-encoded videos with different qualities. For example, the server can switch to a lower rate stream when it detects the available bandwidth reduces, thus resulting in graceful degradation of viewing quality.

An effective rate adaptation algorithm, which can agilely track network bandwidth conditions, is critical for the success of adaptive streaming. Several such algorithms have been proposed for streaming applications [1]–[6], with TCP Friendly Rate Control (TFRC) [6] being a popular choice. These algorithms are reactive in nature, in that, they monitor certain end-to-end network parameters (i.e. delay and packet loss) and adapt the transmission rate (i.e. select the appropriate video stream) depending on the changes in the observed values.

TCP Friendly Rate Control (TFRC). TCP Friendly Rate Control (TFRC) [6] is a popular candidate for congestion control of various streaming media applications [7], [10]. TFRC maintains similar average sending rate as TCP running under comparable network conditions, while providing a relatively smooth sending rate, which helps packets to meet the real-time constraints required by streaming media. TFRC inherits the TCP slow-start mechanism. Once the receiver reports a packet loss event, TFRC sender enters the congestion avoidance state. In this state, the TFRC sender controls the sending rate based on the following simplified TCP throughput model,

$$TFRC_{rate} = \frac{s}{R\sqrt{\frac{2p}{3}} + t_{RTO}\sqrt{\frac{27p}{8}}p(1+32p^2)}. \quad (1)$$

In the above, p denotes the loss event rate, which is received as feedback from the receiver, t_{RTO} refers to the TCP retransmission timeout and s is the packet size. When used in conjunction with an adaptive streaming application, the streaming bit-rate is directly controlled by the sending rate $TFRC_{rate}$.

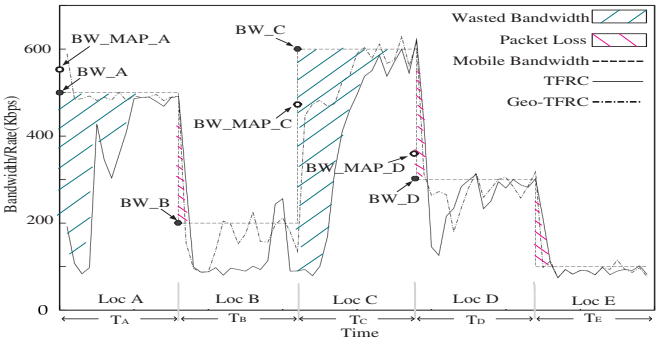


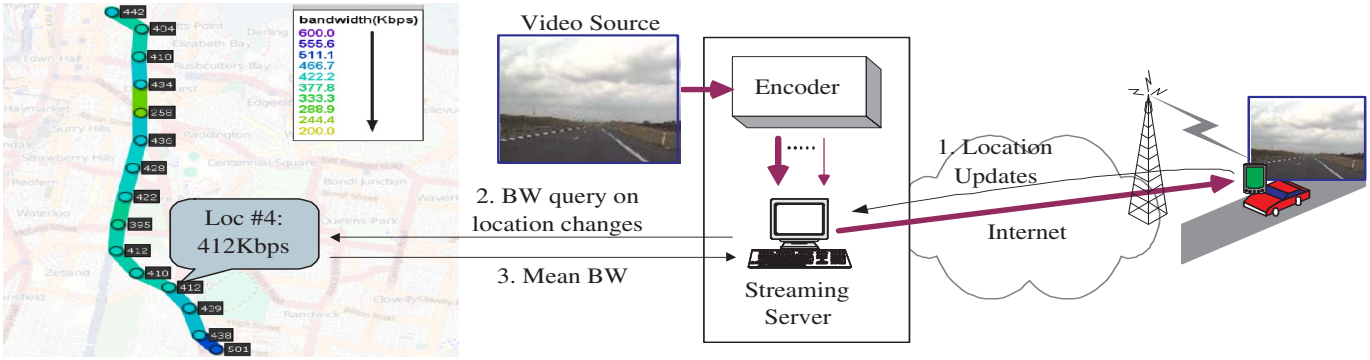
Fig. 2. Illustrative example explaining TFRC rate adaptation

However, TFRC is known to be slow to adapt to frequent changes in bandwidth. We demonstrate this in the context of a vehicular communication scenario, where an on-board TFRC receiver is connected with a TFRC sender in the Internet. Fig. 2 shows the TFRC rate and the WWAN available bandwidth as a function of locations. Observe that bandwidth (in the dashed line) fluctuates significantly, as the mobile client rapidly changes its location [8]. The TFRC rate (in the solid line) is generated from the simulation experiments (details in Section IV). In particular, we focus on the following instances:

- **Slow start**. Since TFRC adopts TCP's slow-start algorithm, it slowly converges to the actual bandwidth during the start of the trip. This results in evident underutilization of the WWAN link.
- **Increase in bandwidth**. Consider the transition from location B to C. TFRC again requires significant time to increase its rate to the bandwidth at the new location due to the large difference between BW_B and BW_C. As a result, the link undertilization is significant.
- **Reduction in bandwidth**. Consider the transition from C to D. The TFRC rate remains greater than BW_D for a noticeable duration. This leads to the WWAN link congestion and subsequently causes packet loss.

When used in conjunction with an adaptive streaming application in a fast-moving scenario, the slow convergence exhibited by TFRC is bound to lead to sub-optimal streaming quality. The periods of time when the TFRC rate is lower than the actual bandwidth represent missed opportunities when the streaming server could have possibly transmitted a higher-quality stream to the user. Further, the time duration in which the TFRC rate is greater than the actual link capacity can lead to lost frames, thus affecting the user viewing experience. The simulations in Section IV confirm this. Our primary objective in this paper is to address the aforementioned problems. We seek to use historical location-specific knowledge of the WWAN link bandwidth so that TFRC can quickly converge to the actual bandwidth when location transitions occur.

Bandwidth-Road Maps. Recent studies [8], [11]–[13] have demonstrated that there is a significant correlation between wireless link characteristics (both WiFi and 3G/HSDPA) such as bandwidth and location. In our earlier work [14], we proposed the principle of *geo-intelligence*, which seeks to



(a) Bandwidth-road map (b) Bandwidth-road map assisted adaptive streaming

Fig. 3. Geo-intelligent adaptive streaming

exploit this strong correlation. The main idea behind geo-intelligence is to collect historical data of observed networking conditions with the aim of constructing *bandwidth-road maps*, which illustrate the expected downlink bandwidth along each segment of the road. Such *a priori* knowledge of bandwidth in the upcoming locations of a moving user can help to achieve a faster adaptation to the bandwidth change, thus resulting in an improved user experience. In this paper, we demonstrate the use of such maps to improve the Quality of Service (QoS) of adaptive video streaming to vehicular users.

The historical data for creating the bandwidth-road maps can be readily collected by wardriving, i.e., running repeated measurements on user devices as they drive around. Passive or active measurement tools can be readily adopted for measuring bandwidth samples. The geo-tagged samples are stored and analyzed to generate the desired profiles. Users can create personalized bandwidth-road maps, which are relevant to their individual mobility profile. The maps could also be shared with other users via websites as is popular with WiFi wardriving communities. These maps may also be of interest to network providers who may offer incentives (e.g., free credits) to users for collecting and uploading their data.

In our earlier work [14], we have conducted an extensive wardriving campaign and created bandwidth-road maps for two distinct routes in the Sydney metropolitan area. We used off-the-shelf hardware and a simple packet-pair measurement tool and collected downlink bandwidth samples for three WWAN providers by making 75 repeated trip along chosen routes. Fig. 3(a) plots the bandwidth-road map for a HSDPA network over the 7Km route from the University of New South Wales to Sydney CBD. The location granularity is 500m. The average bandwidth is computed using exponential weighted moving average (EWMA). We use this map and the corresponding bandwidth traces for the simulation in Section IV.

III. Geo-intelligent Adaptive Streaming

In this section we present a detailed overview of the proposed geo-intelligent enhancement of TFRC, which we refer to as Geo-TFRC. As highlighted in Section II, plain vanilla TFRC is slow in reacting to the frequent bandwidth changes that are encountered when moving at vehicular speeds. We employ a proactive approach, which uses past information of the network characteristics in the form of bandwidth-road maps to achieve faster adaptation to the bandwidth variations.

As shown in Fig. 3(b), our idea requires minimal changes to the video streaming infrastructure. We assume that the streaming server has access to bandwidth-road maps. We assume that the streaming client running on the user's mobile device is aware of its current location (using GPS or WiFi localization). The client updates the server when it changes its location (depending on the location granularity). The server looks up the bandwidth-road map and determines the historical average bandwidth at that location. The Geo-TFRC sender at the server is forced to change its sending rate to this average. This is achieved by freezing the sender for a short period and disabling the normal operation of TFRC. Once this rate change is effected, normal TFRC operation resumes. In other words, our idea is to bootstrap the TFRC sending rate to the observed mean bandwidth at each location.

We use the example in Fig. 2 to explain this further. Consider the transition from location B to C. Assume that the bandwidth-road map indicates that the average bandwidth at location C is BW_MAP_C. The Geo-TFRC sender is now forced to change its sending rate (in the dot-dashed line) to BW_MAP_C. It is evident that Geo-TFRC can rapidly converge to the actual bandwidth as compared to plain vanilla TFRC. Consequently, the underutilization of the WWAN link reduces significantly. This implies that the streaming server can start playing the appropriate higher quality stream much earlier than using TFRC. One can observe a similar improvement when the bandwidth decreases. For example, with the transition from location C to D, the Geo-TFRC sender is enforced to transmit at BW_MAP_D. The time taken by Geo-TFRC to converge to the link bandwidth is shorter as compared to TFRC, thus reducing the associated packet loss and improving the QoS. Also, note that the Geo-TFRC sender bypasses the TFRC's slow-start phase (in location A) and is enforced to send at BW_MAP_A during the initialization.

The intuition behind our idea is that the mean bandwidth based on past observations at a given location is likely to be closer to the current bandwidth at that location, than the bandwidth encountered at the previous location. For example, as observed in Fig. 2, the difference between BW_MAP_C and BW_C is much smaller than the difference between BW_B and BW_C. As a consequence, Geo-TFRC is able to achieve quicker convergence. We show our intuition holds using empirical bandwidth traces. First we formally define two metrics that measure the bandwidth differences.

Let $x_{i,j}$ be the bandwidth encountered at the j^{th} location

during the i^{th} repeated trip along a certain route. Let $\bar{x}_j$ denote the mean of the past bandwidth samples measured at location j (i.e. the value stored in the bandwidth-road maps). The Root Mean Squared Error (RMSE), E_loc_j between the current bandwidth and the past mean at location j computed over N samples is defined as,

$$E\_loc_j = \sqrt{\frac{1}{2(N-1)} \times \sum_{i=2}^{i=N} (x_{i,j} - \bar{x}_j)^2}. \quad (2)$$

The RMSE, E_adj_j between the current bandwidth at a location, $x_{i,j}$ and the current bandwidth at the previous location, $x_{i,j-1}$ (known as Allan Deviation) is defined as,

$$E\_adj_j = \sqrt{\frac{1}{2N} \times \sum_{i=1}^{i=N} (x_{i,j} - x_{i,j-1})^2}. \quad (3)$$

We use the empirical bandwidth samples from our wardriving-exercise described in Section II and compute E_loc_j and E_adj_j at all locations. Table I compares the averages. Observe that $\overline{E\_loc}$ is 26% lower than $\overline{E\_adj}$. This confirms our intuition that the observed mean bandwidth at a location is closer to the actual bandwidth as compared to the bandwidth at the previous location.

TABLE I
MEAN RMS ERROR COMPARISON

	$\overline{E\_loc}$	$\overline{E\_adj}$
RMS Error (Kbps)	107.69	145.42

We use a state machine (Fig. 4) to explain our proposed algorithm in detail. The normal operation of TFRC is represented by the *Plain Vanilla TFRC* state. While in this state, the sending rate is determined as per (1). Geo-TFRC introduces two additional states at the sender side: *Location Changed* and *Restoring*. The receiver remains unchanged. When the Geo-TFRC receiver moves to a new location the sender transitions to the Location Changed state. The sending rate is forced to BW_MAP, the mean bandwidth at the new location as reported by the bandwidth-road maps. The Geo-TFRC sender remains frozen in this state for a sufficient period to ensure that the rate change takes effect and that the receiver provides feedback based on the new rate. From our experiments, we find that a time period of 2RTT is sufficient. Following this, the sender enters the Restoring state. In this state, the sending rate is still maintained at BW_MAP. However, unlike in the Location Changed state, the Geo-TFRC sender resumes to estimate the $TFRC\_rate$ according to (1). Note that, even though the average bandwidth as reported in the bandwidth-road maps is close to the current link bandwidth, it is never going to be exactly equal. Hence, it is important that the sender begins to monitor the feedback from the receiver to gauge if the current sending rate BW_MAP is either causing excessive congestion (i.e. if $TFRC\_rate$ decreases) or underutilization (i.e. if $TFRC\_rate \geq BW\_MAP$) of the link. In either situation, the sender transitions to normal TFRC operation.

IV. TRACE-DRIVEN SIMULATION

In this section, we present results from trace-driven simulations. We first present details of the simulation setup. Next, we compare the performance of Geo-TFRC and plain vanilla TFRC using empirically collected HSDPA bandwidth traces.

A. Simulation Setup

We have used the *NS-2* simulator for our experiments. Our simulation setup is illustrated in Fig. 3(b). We assume that an adaptive streaming server is connected via a high-speed wired link to the Internet. The streaming client is connected to the Internet via a HSDPA link and is housed in a vehicle that travels along the 7Km route discussed in Section II. We split our 75-trip HSDPA bandwidth traces into two groups. The first group of 45 trips are exclusively used as historical samples to create the bandwidth-road maps. To simulate vehicular mobility, we simply vary the bandwidth of the HSDPA link by playing back the empirical bandwidth traces for a particular trip from the second group of 30 trips. The one way delay over the HSDPA link is set to 65ms, the average one-way link delay observed in our measurements. Since the wired links in the Internet have sufficiently high bandwidth and small delays as compared to the last-hop HSDPA link, we use a 100Mbps wired link with a small propagation delay of 10ms to abstract the wired Internet. We assume that the streaming server always has knowledge of the vehicle location.

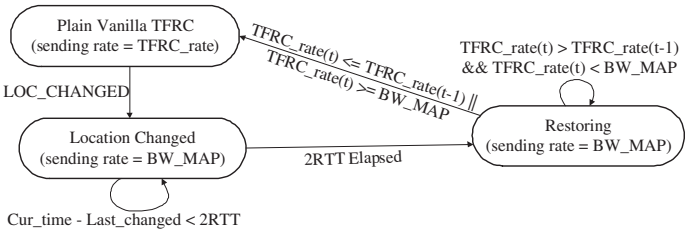


Fig. 4. State machine of Geo-TFRC

For implementing adaptive streaming we use the *Evlavid-RA* framework [1]. In this framework, a video is pre-encoded at 31 different rates. Each of the encoded video streams is then partitioned into a sequence of small "chunks", e.g., Group of Pictures (GoP). Therefore, the streaming server can dynamically vary the streaming quality and bit-rate, by switching between different quality GoPs from the corresponding encoded streams. We use a medium motion media sequence ("Foreman" QCIF sequence) encoded at 30 $frame/second$ using the popular MPEG-4 codec.

We compare the performance of Geo-TFRC with that of plain vanilla TFRC. To implement the former, we have modified the stock TFRC implementation in NS-2 according to Fig. 4. As a benchmark, we also include a hypothetical rate adaptation algorithm, *optimal*, which has perfect knowledge of the HSDPA bandwidth at each location. Thus this scheme can perfectly adjust the streaming rate to match the actual bandwidth and thus achieves the best possible performance.

Recall that, the HSDPA bandwidth changes with location. The quality of adaptive video depends critically on the agility of the rate adaptation algorithms in tracking the varying bandwidth. We quantify this using *convergence time*. Assume that the client enters a location l at time T_l and remains in l till time T_{l+1}. Let $r(t)$ denote the sending rate at time t and let the HSDPA bandwidth at l be BW_l. The difference between

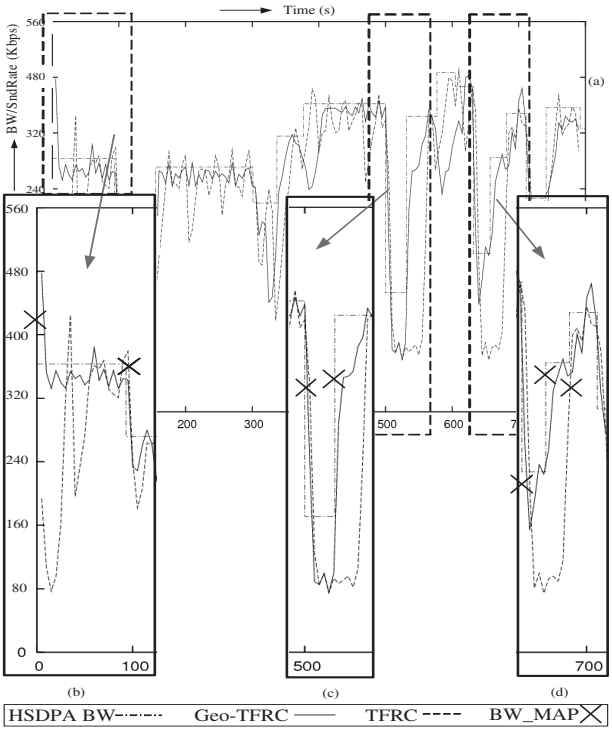


Fig. 5. Sending rate and HSDPA bandwidth dynamics for trip #19

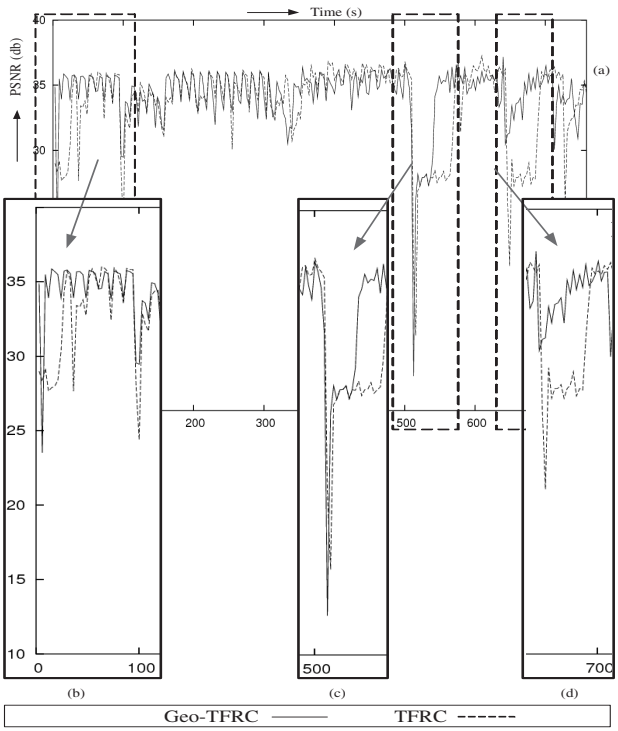


Fig. 6. PSNR dynamics for trip #19

the two is $D(r(t), BW_l) = |r(t) - BW_l|$. In location l, we define $r(t)$ converges to BW_l at time T, if and only if, for a small value ϵ, $D(r(t), BW_l) < \epsilon,\ \forall t \in [T, T_{l+1}]$. Finally, the convergence time of r(t) is $T - T_l$. We choose $\epsilon = 30Kbps$.

For evaluating the video quality, we use the popular objective video quality metric, Peak Signal-to-Noise Ratio (PSNR). PSNR is calculated (in dB) for each frame, by a pixel-by-pixel comparison of the decoded frame to the corresponding source frame. It is well-established that if the PSNR is above 31 dB, then the frame quality is good. Humans can perceive a drop in the streaming quality, when the $PSNR$ is less than or equal the 31 dB threshold consistently for 1 second or longer [15]. We refer to such an event as a *streaming disruption*. To quantify the disruptions experience over a given trip, we define the *disruption time* as the cumulative sum of the durations in which a viewer suffers from streaming disruption.

B. Results

Fig. 5 plots the HSDPA bandwidth and the sending rate of TFRC and Geo-TFRC as a function of time for one of the trips. The corresponding PSNR values are shown in Fig. 6. Note that the optimal scheme is excluded since it perfectly tracks the HSDPA bandwidth. We are particularly interested in studying how schemes behave during location transitions. Hence, we focus on 3 such instances in Fig. 5(b)-(d) and Fig. 6(b)-(d).

Fig. 5(b) focuses on the start of the trip. Since TFRC employs TCP's slow start algorithm, it requires considerable time to ramp up its sending rate. Further, observe that the exponential increase during slow start leads to packet loss, following which the sender enters congestion avoidance. As a result, TFRC only converges at around 70s. On the contrary, Geo-TFRC bypasses slow-start completely. The sending rate is bootstrapped at the observed mean bandwidth (from the bandwidth-road maps), thus leading to rapid 10s convergence.

(a) Geo-TFRC (35 dB) (b) TFRC (28 dB)

Fig. 7. Picture quality comparison at 20s in trip #19

The improved convergence achieved by Geo-TFRC is reflected in the PSNR results. Fig. 6(b) shows that the PSNR of Geo-TFRC quickly converges at 35 dB and remains steady around that value, which indicates the good streaming quality. On the contrary, TFRC significantly underutilizes the bandwidth and thus misses out on opportunities to transmit a better quality stream. As a result, it can only achieve a PSNR of 28 dB until 50s. Fig. 7 offers a snapshot of the viewer experience for both schemes by displaying the decoded frame played at 20s. Note that the picture quality with TFRC is pixelated, whereas the frame delivered by Geo-TFRC is crystal clear.

Fig. 5(c) captures bandwidth fluctuations due to two consecutive changes in location, at 500s and 530s. When the bandwidth drops from 440Kbps to 160Kbps at 500s, both TFRC and Geo-TFRC suffer from poor streaming quality (shown in Fig. 6(c)). This is because the link bandwidth is so low that good quality streaming is not possible. At 530s, the mobile bandwidth recovers back to around 410Kbps. Fig. 5(c) shows that TFRC is conservative in increasing its rate, as it is affected by the sudden increase in packet loss and delay during the previous location transition. However, Geo-TFRC does not suffer from this, since it treats each location independently and uses the past mean bandwidth as an initial estimate for the sending rate. One can observe that the convergence time for both schemes are very similar in this case. However, Geo-

TFRC can use the bandwidth more effectively. This is evident in Fig. 6(c), where Geo-TFRC quickly restores the PSNR to 35 dB at 540s, nearly 20s faster than TFRC.

Fig. 5(d) shows that the mobile bandwidth drops sharply from around 470Kbps to 240Kbps at 630s. Again, Geo-TFRC is quicker to react to this sharp decrease as compared to TFRC. Fig. 6(d) illustrates that as Geo-TFRC reduces the sending rate, it is still able to maintain reasonable streaming quality. On the contrary, excessive packet loss causes TFRC to cut the rate more aggressively, thus reducing the streaming quality. When the bandwidth recovers at 660s, Geo-TFRC converges twice as fast as TFRC. Further, observe that by the time TFRC has successfully converged (680s), the client has already moved to the next location.

TABLE II
COMPARISON OF AVERAGE RESULTS

	TFRC	Geo-TFRC	Optimal
Convergence Time (s)	26.13	18.50	0
Bandwidth Utilization	76.35%	87.99%	93.10%
PSNR (dB)	31.03	34.02	34.28
PSNR_variance (dB^2)	4.47	2.44	1.27
Packet_loss	2.81%	1.94%	0.12%

Fig. 5 and 6 are representative of the performance observed in the rest of the 29 trips. Table II compares the mean convergence time, bandwidth utilization, PSNR, and packet loss results for all location transitions in the simulations. Excluding convergence time, the performance of Geo-TFRC is on-par with that of the Optimal scheme. In comparison with TFRC, Geo-TFRC reduces the mean convergence time by 29%. Further, Geo-TFRC achieves 15% and 10% improvement over TFRC in bandwidth utilization and PSNR, respectively. Also, the average PSNR achieved with TFRC is marginally over 31 dB, which is the threshold for good streaming quality. This coupled with the larger variance in PSNR seems to suggest that with TFRC, users may often encounter a poor viewing experience. Lastly, Geo-TFRC results in 33% less packet loss than TFRC. This implies that using past history from the bandwidth-road maps is particularly useful in achieving convergence when there is a significant drop in bandwidth.

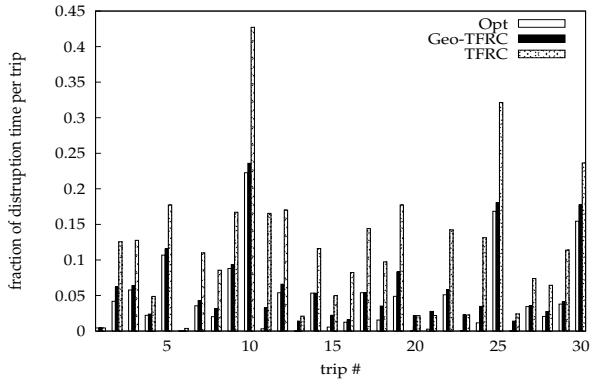


Fig. 8. Disruption time comparison

In Fig. 8, we plot the disruption time as a percentage of the entire trip duration for all 3 schemes. Note that, even the optimal scheme can suffer from disruptions, since occasionally the HSDPA bandwidth can be too low to sustain good quality streaming. Clearly, Geo-TFRC suffers less streaming disruption, as compared to TFRC. In some trips (e.g., #11 and #16), Geo-TFRC achieves over *five-fold* reduction in the disruption time. Geo-TFRC outperforms TFRC in 27 out of total 30 trips (90%). In the 3 remaining trips (#20, #21 and #23), the disruption time is similar with both. On average, with TFRC disruptions account for 11.6% of the trip time. Geo-TFRC reduces this figure to only 5.6%, leading to a 51% reduction.

V. CONCLUSION

In this paper, we have addressed the adaptive video streaming problem in mobile environment by considering the strong correlation between the location and WWAN bandwidth. We incorporate the previously proposed bandwidth-road maps with the rate adaptation algorithms, and propose Geo-TFRC. By using the prior bandwidth knowledge in making proactive rate adaptation decision during location transition, Geo-TFRC is able to track the varying WWAN bandwidth faster and smoother. We have implemented and evaluated Geo-TFRC on MPEG-4 VBR video streaming. We showed that Geo-TFRC can effectively reduce the rate adaptation convergence time by 29%. Further, we found our approach can significantly reduce the time that a user suffers from pixelated video quality by up to *five* folds as compared to TFRC in a given trip.

REFERENCES

[1] A. Lie and J. Klaue, "Evalvid-RA: Trace Driven Simulation of Rate Adaptive MPEG-4 VBR Video," in *ACM/Springer Multimedia Systems Journal*, 2007.

[2] R. Rejaie, M. Handley, and D. Estrin, "RAP: An End-to-end Rate-based Congestion Control Mechanism for Realtime Streams in the Internet," in *Proc. of IEEE Infocom99*, Mar. 1999.

[3] D. Sisalem and A. Wolisz, "LDA+ TCP-Friendly Adaptation: A Measurement and Comparison Study," in *Proc. of NOSSDAV00*, 2000.

[4] S. Wenger, U. Chandra, M. Westerlund, and B. Burman, "Codec Control Messages in the RTP Audio-Visual Profile with Feedback (AVPF)," *RFC 5104*, 2008.

[5] V. Singh, J. Ott, and I. Curcio, "Rate Adaptation for Conversational 3G Video," in *Proc. of IEEE INFOCOM 2009 Workshop on Mobile Video Delivery*, Rio de Janeiro, Brazil, Apr. 2009.

[6] S. Floyd, M. Handley, J. Padhye, and J. Widmer, "TCP Friendly Rate Control (TFRC): Protocol Specification," *RFC5348*, Sep. 2008.

[7] L. Xu and J. Helzer, "Media Streaming via TFRC: An Analytical Study of the Impact of TFRC on User-Perceived Media Quality," in *Proc. of IEEE INFOCOM*, 2006.

[8] J. Yao, S. S. Kanhere, and M. Hassan, "An Empirical Study of Bandwidth Predictability in Mobile Computing," in *Proc. of ACM WinTech 2008*, San Francisco, CA, USA, Sep. 2008.

[9] J. Derksen, R. Jansen, M. Maijala, and E. Westerberg, "HSDPA Performance and Evolution," *Ericsson Review*, no. 03, pp. 117–120, 2006.

[10] B. Zhou, C. P. Fu, and V. O. K. Li, "TFRC Veno: An Enhancement of TCP Friendly Rate Control over Wired/Wireless Networks," in *Proc. of IEEE ICNP 2007*, vol. 0, Los Alamitos, CA, USA, 2007.

[11] A. J. Nicholson and B. D. Noble, "BreadCrumbs: Forecasting Mobile Connectivity," in *Proc. of 14th ACM MobiCom*, San Francisco, CA, USA, Sep. 2008.

[12] A. Balasubramanian, R. Mahajan, A. Venkataramani, B. N. Levine, and J. Zahorjan, "Interactive Wifi Connectivity for Moving Vehicles," *SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 4, pp. 427–438, 2008.

[13] J. Pang, B. Greenstein, M. Kaminsky, D. Mccoy, and S. Seshan, "Wifi-Reports: Improving Wireless Network Selection with Collaboration," in *Proc. of ACM MobiSys 2009*, Krakow, Poland, Jun 2009.

[14] J. Yao, S. S. Kanhere, and M. Hassan, "Geo-intelligent Traffic Scheduling For Multi-homed On-Board Networks," in *Proc. of ACM MobiArch09*, Krakow, Polland, Jun. 2009.

[15] Z. Orlov, "Network-driven Adaptive Video Streaming in Wireless Environments," in *Proc. of IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC08*, Sep. 2008.