

PERFORMANCE MEASUREMENTS AND EVALUATION OF VIDEO STREAMING IN HSDPA NETWORKS WITH 16QAM MODULATION

Haakon Riiser¹, Pål Halvorsen^{2,3}, Carsten Griwodz^{2,3}, Bjørn Hestnes⁴

¹Netview Technology AS, Norway

²Simula Research Laboratory, Norway

³Department of Informatics, University of Oslo, Norway

⁴Telenor Research and Innovation, Norway

ABSTRACT

The High-Speed Downlink Packet Access (HSDPA) communication protocol improves downlink performance on mobile networks and is currently being deployed in networks around the world. In Norway, two operators started offering this service in 2007. This article discusses performance measurements done in Telenor's HSDPA network in Oslo in the context of video streaming. In particular, we present the results of streaming experiments measuring characteristics such as packet loss, latency, jitter and bit-errors. Based on this data, we evaluate the HSDPA network's suitability for real-time video streaming, and discuss what can be done to improve the quality of video streaming under the observed network characteristics.

1. INTRODUCTION

Video streaming to mobile terminals is an industry that is growing quickly and thus receives great attention from the telecommunications industry. A large challenge in this respect is that a lot of bandwidth is required, and for wireless networks, this poses some problems because of two inherent problems compared to wired networks: (1) much lower signal-to-noise ratio, and (2) potentially more people that share the same bandwidth. The first point means that one needs to make a compromise between reliability, robustness and bandwidth, and the second point means that the available bandwidth per user (in a shared frequency spectrum) will be much less than with wired networks since it is not possible to simply draw additional cables to add more bandwidth.

A promising technology in this scenario is the High-Speed Downlink Packet Access (HSDPA) which is a communications protocol that improves downlink performance on mobile networks based on Universal Mobile Telecommunications System (UMTS). HSDPA is rapidly being deployed in Wideband Code Division Multiple Access (W-CDMA) networks around the world. In Norway, two operators started offering this service in 2007.

This article presents results from streaming experiments and discusses the performance and characteristics of an HSDPA network in Oslo (Norway) in the context of video streaming. We have collaborated with and used the network of Telenor which is a large telecom operator. This network, using components manufactured by Ericsson, allowed for a raw bandwidth of up to 3.6 Mbps on the air interface using 16-ary quadrature amplitude modulation (16QAM).

In this respect, Ericsson published in 2006 some bandwidth measurements from a commercial HSDPA network in Asia [1], where they measured bandwidths remarkably close (within 10 %) to the theoretical limits, and also median transfer rates that even in poor

signal conditions exceeded 50 % of the theoretical limit. In our work, we look not only at bandwidth, but also address latency and loss characteristics of such a network, and discuss the findings in the context of a streaming service, both with respect to the number of users and in a stationary and a mobile scenario. Most tests were done in an urban environment (downtown Oslo), except for the mobility tests that were performed in the sparsely populated northern parts of Oslo. The network also contained R99 (UMTS) traffic, i.e., voice and data.

Our results indicate that the achieved per-user bandwidth and packet latencies can support on-demand or live video streaming in Telenor's 200 kbps target rate. However, such a system also needs mechanisms to deal with varying resource availability due to unfair bandwidth allocation between different streaming users, sawtoothed loss patterns on congestion and vast fluctuations in bandwidth during handover in rural locations where base stations are more sparse.

2. EXPERIMENTS

Our experiments were performed by streaming data from a server to both stationary and mobile receivers in the HSDPA network using UDP. The server was running on a computer with a 100 Mbps connection, only four hops from the Norwegian Internet exchange, and in total, about 15 hops to the laptops in the HSDPA network (we tested many different locations and thus different HSDPA cells). We used sequence numbers and time stamps to measure bandwidth and latency, both with a single receiver exhausting the bandwidth all by itself, and then with four simultaneous receivers to see the network behavior in a multi-user setting. Moreover, we used the Network Time Protocol to synchronize clocks and we found that our clocks would never drift apart by more than 15 μ s over our longest tests, which is insignificant when compared to latencies in the order of 100 ms. Additionally, we quantified the loss at the application level by disabling checksums (though bit-errors were extremely rare with less than 0.003 % corrupted packets). We used several bitrates, but within a session the bitrate was constant.

3. RESULTS

We first present results showing the perceived *bandwidth* and *packet loss*. The first tests were performed at several locations in downtown Oslo, using stationary users. This location was chosen because the signal quality here was very good, i.e., virtually no packets were dropped due to noise.

The graph in figure 1(a) show a representative sample from one of

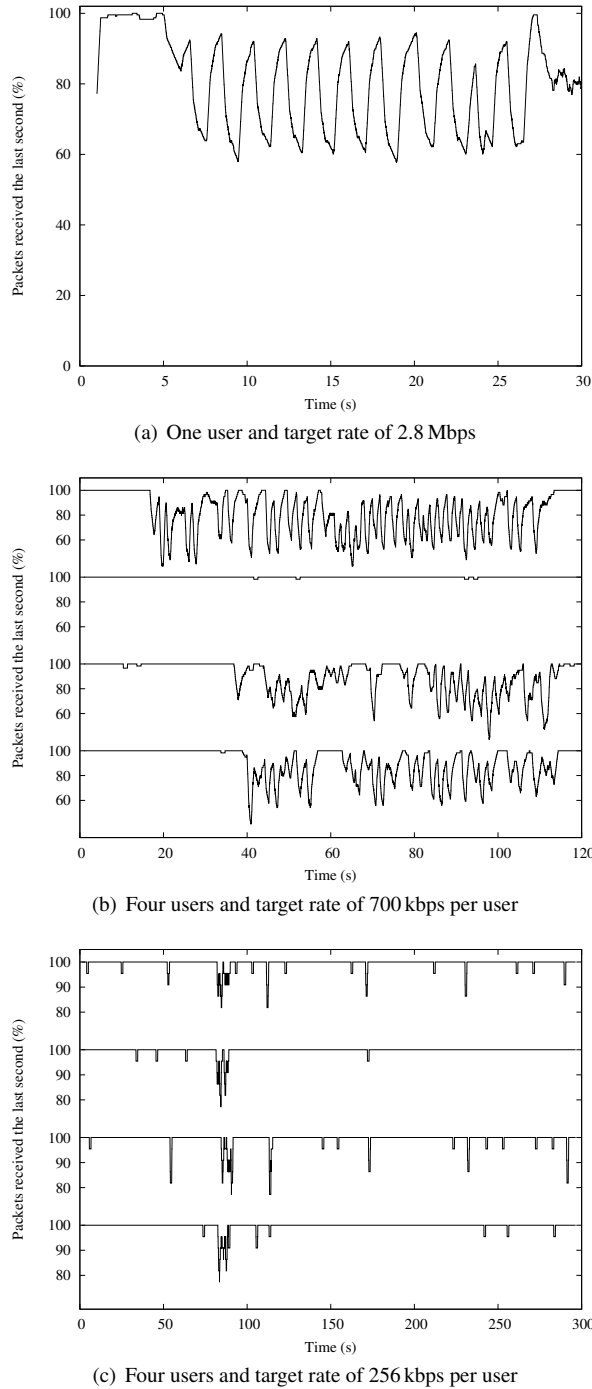


Fig. 1. Packet drop patterns with stationary receivers

the HSDPA cells where a single receiver tried to stream at 2.8 Mbps, a rate known in advance from preliminary tests to be above the effective downlink capacity of the HSDPA cell. Figure 1(a) plots the percentage of packets that were lost in a one-second window. Thus, a perfect signal would produce a horizontal line at 100%. Dips in the curve represent loss of data. As can be seen in figure 1(a), we observed a saw-toothed drop pattern. The effective average downlink

bandwidth was 2.25 Mbps.

This is well below Ericsson's results [1] in similar networks, but this may be because some of the bandwidth here was reserved for R99 traffic. In the same scenario, but with four receivers (i.e., four different machines placed next to each other with their own HSDPA cards), each user tried to receive their own unicast stream at 700 kbps, which add up to 2.8 Mbps (i.e., the same as in the single-receiver test).

In figure 1(b), the same test was performed with four receivers. The aggregated effective average downlink bandwidth of the four receivers was 2.53 Mbps, over 12 % higher than what was achieved with one receiver in figure 1(a). This shows that cross-traffic interference between receivers is not a problem, and thus that the total bandwidth in the HSDPA cell can be divided linearly among a set of users. As an example, an effective cell bandwidth of 2.5 Mbps should mean that the cell can support ten users streaming at 250 kbps.

Also, note that packet dropping happens very differently for different receivers, even though they are located at the same place, at the same time, streaming at the same bitrate. The uppermost receiver in figure 1(b) starts losing data just before 20 seconds have passed, the lowermost two start at approximately 40 seconds, while receiver number two from above never sees packet dropping due to congestion in the wireless hop.

In the scenario above, the primary cause of packet loss has been congestion on the air interface. Another cause of packet loss is external noise. We analyzed this at a location with fair, but not perfect, signal quality, and reduced the streaming rate to 256 kbps, which is well below the capacity of the network. This ensured that packet loss was due to noise. Four simultaneous receivers in the same location were used to see if different receivers get significantly different loss patterns. As can be seen in figure 1(c), minor packet loss happens at random places (i.e., not affecting each receiver at the same time), while longer bursts of lost packets tend to affect more receivers at the same time. This is expected, since these most likely result from more significant amounts of external noise.

For testing bandwidth and packet loss during handover between base stations (often denoted "Node Bs" in UMTS terminology), we found a road in a sparsely populated area in the northern parts of Oslo, and drove back and forth between two points while trying to maintain a constant speed of 50 km/h and at the same time streaming at various bitrates. Figures 2(a) and 2(b) were both generated while driving between two points A and B at 2 Mbps and 256 kbps respectively. Points A and B were chosen because they both represented locations that gave good transfer rates in stationary tests. The top-most graphs in the figures show the results for driving from A to B, and the graphs at the bottom show the results while driving back, B to A. The distance between the two points was 3.7 km.

Looking at figure 2(a), one can see that the handover between base stations happens abruptly. In the 2 Mbps tests, the transfer rate starts to drop after 60 seconds, reaches almost zero after about 150 seconds, and remains low after that until it suddenly and immediately bounces back up to 100%. This behavior is seen both while driving from A to B and B to A, the only difference being the time at which the signals rise sharply from 0 to 100%.

The test was then repeated at 256 kbps, with results presented in figure 2(b). Reducing the streaming rate gave a much more stable transmission, but there are still bursts of lost packets. In the top-most graph in figure 2(b) (A to B), the two bursts last approximately 15 seconds (disregarding the small peaks inside that represent short "holes" in the burst loss). In the B to A direction, we still see two bursts, but this time much shorter, each being approximately 2 seconds long.

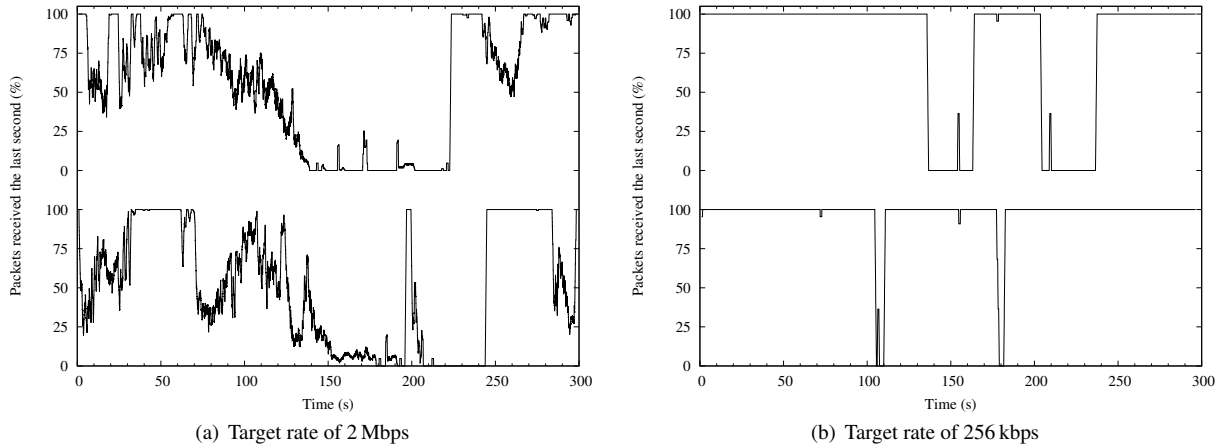


Fig. 2. Packet drop patterns with a mobile receiver driving at 50 km/h

Throughout our tests, latency was consistently measured between 70 and 120 ms. This is more than good enough for non-interactive video streaming applications, so it will not be discussed further. Jitter, however, is more interesting, because it causes problems for video transfer when large variations in latency can fool the receiver of a video into believing that a packet was lost, when it was actually just delayed.

As seen in figure 1(a), the packet drop pattern is saw-toothed. This is because of active queue management in the base station, and it implies that there is a buffer in the base station that allows for fluctuations in bandwidth where peaks exceed the maximum bandwidth of the wireless network for short amounts of time. When the buffer overflows, huge numbers of packets are dropped. If the available bandwidth is briefly exceeded (not long enough for the buffer to overflow), there's still a negative side-effect: jitter.

The jitter for the optimal case of streaming in a constant bitrate is shown in figure 3 where the statistical distribution of jitter corresponds to the topmost graph in figure 1(c). As we can see from figure 3, the variations in latency for constant bitrate streaming (that does not exceed the available bandwidth), have a Gaussian distribution with a standard deviation (σ) of approximately 7 ms. The average latency was approximately 100 ms. This shows that there is very little jitter in normal conditions, and that the latency is more than good enough for non-interactive applications.

Measurements at other bitrates were also performed, and showed that σ does not depend on the streaming rate, unless the base station detects congestion. In the latter case, the buffer in the base station is repeatedly filled and flushed, causing great fluctuations in latency.

To illustrate when jitter *does* become a problem, we show the jitter distribution when the buffer overflows. We present the jitter distribution for the topmost graph from figure 1(b) in figure 4. It clearly shows that the distribution is no longer Gaussian, and that the variations in latency are *huge* compared to figure 3, spanning almost 7.5 seconds. It is also worth mentioning that the average latency in this case was over 6 seconds, compared to 0.1 seconds in figure 3). Even for a non-interactive application like one-way video streaming, this is unacceptable, as it forces the buffer size (and thus the startup delay) much too high to simulate the experience of television (channel switching should not take more than one second, preferably less [7]).

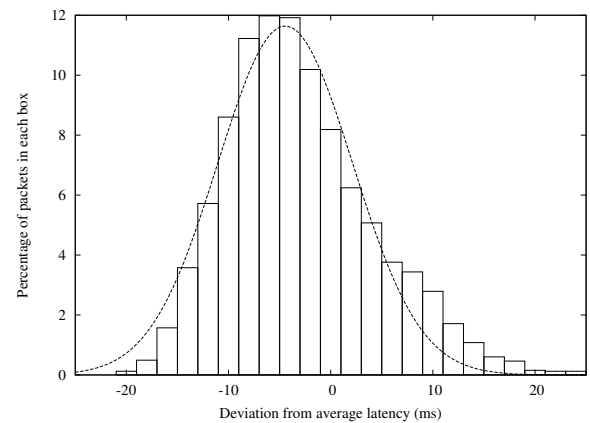


Fig. 3. Statistical distribution of jitter at a constant bitrate of 256 kbps

4. EVALUATION OF HSDPA FOR VIDEO STREAMING

The first potential streaming problem we presented is shown in figure 1(b) where four adjacently placed simultaneous receivers of the same unicast stream experience very different drop patterns when the cell's bandwidth is exhausted. This may be caused by the scheduling algorithm used in the base station. If the base station uses streaming-aware scheduling algorithms (described in [2] and [3]) that try to maintain the bandwidth of streaming users, the consequence may be that mere coincidence decides which streams are able to sustain their required streaming rate and which streams are victims to packet dropping.

We have shown that total downlink bandwidth per base station is around 2.5 Mbps in Oslo. Given the distance between base stations and the population density of central Oslo, we calculated that the network can stream at 200 kbps (a targeted streaming rate by Telenor) to almost 3 % of the population at any given time. This is more than enough for current usage of mobile TV and video. The proposed upgrade path of UMTS systems also promises improved spectral efficiency and broader frequency bands, which may allow

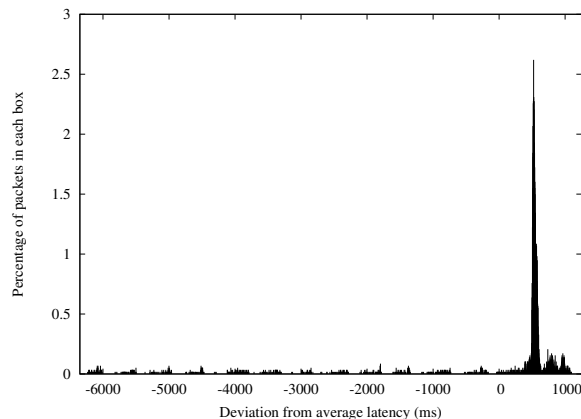


Fig. 4. Statistical distribution of jitter at 700 kbps with buffer overflows in the base station

the bandwidth to scale to match the growing video streaming market in the future.

Our measurements (a small subset of which is presented in figure 1(c)) show that background noise in central Oslo's urban environment is modest, indicating that application layer FEC may prove to be a useful tool in assuring a stable delivery of video. Indeed, internal user studies done by Netview Technology AS has showed significant improvements in perceived video quality when 20 % of the bitrate was traded for redundancy, enough for eliminating most sporadic occurrences of packet loss.

The mobility tests that analyzed handover (figures 2(a) and 2(b)) show that the network is still not capable of providing a stable connection while in motion. Note that the problems observed in handover tests do not represent mobile measurements in downtown Oslo. Due to very good coverage, the signal in central Oslo was much more stable, and abrupt handovers between base stations were not observed. Still, the measurements shown here illustrate the types of connectivity problems one may encounter while moving through areas with reduced coverage, and a video streaming solution needs to be aware of such phenomena in order to minimize their impact. It is clear that quickly adapting the bitrate will do much to prevent packet loss. A scheme based on scalable video coding (such as H.264 SVC [5]) in combination with an intelligent bit-allocation strategy that provides increasing amounts of redundancy on the lower quality layers could allow for a more graceful degradation in quality when the signal characteristics change. Such a strategy is discussed in [6], using Reed-Solomon codes for FEC. We believe, however, that the FEC should be based on Raptor codes [4] for their flexibility and reduced computational complexity. Furthermore, if the round trip time between sender and receiver is long, additional improvements may be achieved by using a proxy/filter node in or near the access network so that the system can respond more quickly to changes in bandwidth and signal quality. Such a node would use receiver feedback to reduce the SVC stream appropriately and generate new FEC codes according to a bit-allocation strategy that does its best to maintain video playback without interruptions while at the same time maximizing quality at any given time.

Our results on latency and jitter show that one-way streaming can be supported easily, as long as the streaming rate is fairly constant. This is not a very strict requirement, however, since problems will only occur when bitrate peaks from several independent streams oc-

cur at the same time (unless, of course, the bitrate of a single stream is so unstable that it alone can overflow the buffers). Because of this, a video-on-demand system would be fairly tolerant for some degree of bitrate variations.

5. SUMMARY

We have together with Telenor evaluated their HSDPA network in the context of video streaming. We found characteristics like unfair bandwidth allocation between different streaming users, saw-toothed loss patterns on congestion, and vast fluctuations in bandwidth during handover in rural locations. However, the achieved per-user bandwidth and packet latencies indicate that on-demand video streaming at the target rate of about 200 kbps is viable in densely populated areas with good coverage. Still, such a system needs mechanisms to deal with varying resource availability, most prominently shown by figures 2(a) and 2(b). We are therefore doing further research on technologies such as scalable video coding (H.264/SVC) and unequal error protection (based on Digital Fountain's Raptor codes).

6. REFERENCES

- [1] J. Derksen, R. Jansen, M. Maijala, E. Westerberg, *HSDPA performance and evolution*, Ericsson Review No. 3, 2006, pp. 117–120.
- [2] M. Lundevall, B. Olin, J. Olsson, N. Wiberg, S. Wänstedt, J. Eriksson, F. Eng, *Streaming Applications over HSDPA in Mixed Service Scenarios*, Vehicular Technology Conference 2004, IEEE 60th, pp. 841–845 Vol. 2.
- [3] T. E. Kolding, *Link and system performance aspects of proportional fair scheduling in WCDMA/HSDPA*, Vehicular Technology Conference 2003, IEEE 58th, pp. 1717–1722 Vol. 3.
- [4] *Universal Mobile Telecommunications System (UMTS); Multimedia Broadcast/Multicast Service (MBMS); Protocols and codecs*, 3GPP TS 26.346.
- [5] H. Schwarz, D. Marpe, T. Wiegand, *Overview of the Scalable Video Coding Extension of the H.264/AVC Standard*, IEEE Transactions on Circuits and Systems for Video Technology, Vol. 17, No. 9, September 2007.
- [6] Q. Zhang, W. Zhu, Y. Zhang, *Channel-adaptive resource allocation for scalable video transmission over 3G wireless network*, IEEE Transactions on Circuits and Systems for Video Technology, pp. 1049–1063, Vol. 14, No. 8.
- [7] J. De Wachter, D. Tytgat, V. Verstraete, M. Strobbe, F. Van Quickenborne, P. De Neve, F. De Turck, P. Demeester, *Optimizing Channel Switching for Digital Television Through Low Quality Streams*, Proceedings (on CD-ROM) of the 6th IASTED International Conference Communication, Systems and Networks, Palma de Mallorca, Spain, 29–31 August 2007, pp. 55–62.