Performance Study of a Video Application over Multi Hop Wireless Networks with Statistic-Based Routing

(Work in Progress)

Alexander Klein[1] and Jirka Klaue[2]

[1] University of Wuerzburg, Institute of Computer Science, Germany
lastname@informatik.uni-wuerzburg.de
http://www3.informatik.uni-wuerzburg.de
[2] EADS Innovation Works, Munich, Germany

Abstract. In recent years, technologies like Ultra Wideband (UWB) were developed that can be used to boost the data rate of wireless nodes to a new level. The higher data rate makes wireless networks capable of transporting multimedia content for real time applications like In Flight Entertainment (IFE), surveillance applications, or structural health monitoring. In this paper we measure and simulate the performance of a video application for IFE over a high data rate multi hop wireless network using a Directed Diffusion based routing protocol. Therefore, we take a look at the perceived video quality instead of focusing solely on the packet delivery ratio. Furthermore, we discuss the parameters of the Statistic-Based Routing (SBR) protocol which are relevant for this particular application and focus on their impact on the video quality during topology changes.

Keywords: routing, wireless, multi-hop, video, performance.

1 Introduction

Frequent topology changes represent a serious problem for IFE since most of the traffic has to be transmitted in real time which limits the possibility of retransmitting lost packets. Depending on the used wireless technology, carry-on luggage or clothes that are placed on the back rest might lead to topology changes. For that reason, the routing protocol has to frequently send out messages to detect topology changes and reroute traffic.

Thus, we have to deliberate over the question whether wireless solutions can be competitive to wired solutions for our particular application of IFE. First of all, the video quality that is recognized by a customer has to be approximately on the same level to allow a comparison of wired and wireless solutions which mainly depends on the routing protocol if we assume that the used physical layer is capable to achieve a high point-to-point data rate. Other characteristics that have to be taken into account are the costs for deployment and general maintenance work. The installation costs of wired solutions in already assembled planes

L. Fratta et al. (Eds.): NETWORKING 2009, LNCS 5550, pp. 955–966, 2009.

are more expensive than the installation costs of wireless solutions. Flexibility is a very important issue since the airline companies are interested in adapting the configuration of a plane, e.g. by extending the business or the economy class.

Our work is organized as follows. In Section 2 we take a look at wireless network simulation tools and the way they simplify the signal propagation to allow large-scale simulations. The problem of topology change detection is discussed in Section 3. We describe the simulation of the routing protocol which is used to evaluate the performance of the video application in Section 4 and its implementation in Section 5 respectively. The framework that is used to estimate the performance of the received video is introduced in Section 6. The results are presented and analyzed in Section 7. Finally, we summarize the results and introduce our future fields of research.

2 Related Work

As a consequence of rapid improvements in technology and miniturization, Wireless Multimedia Sensor Networks (WMSN)become more and more interesting for a large number of new applications [1]. However, sensor nodes with UWB or other high data rate wireless interfaces are hardly available at the moment. Thus, simulation is the most common approach to estimate the performance of WMSNs. Many simulation tools like ns-2 [2] or OPNET Modeler [3] come with simplified propagation models, e.g. free space, which neglect most of the characteristics that have great impact on the communication in a multi-hop wireless network. Often, these models are even further simplified to allow the simulation of large-scale networks within a justifiable amount of time. Kotz et al. [4] summarized the typical assumptions that are made by simulations like circular transmission area, equal transmission range, and symmetric links. Another work that is published by the same research group [5] deals with the problem of simulation validation by using measurements from an environment typical to the one of interest. Their results show that it is important to use measurements in order to find out which propagation model meets the desired requirements.

Therefore, we decided not to focus solely on the standard performance metrics like packet delivery ratio, end-to-end delay and jitter since they do not necessarily represent the perceived link quality by the user. If we take a closer look e.g. at video encoding and decoding, we recognize that the video quality over a lossy link strongly depends on the way packets are lost. It is important to know whether consecutive packets are lost as a consequence of topology changes, single packets which might be caused by interference, or a low signal to noise ratio. For that reason, we focus on the Mean Opinion Score (MOS) of a video transmission to evaluate the performance of our network.

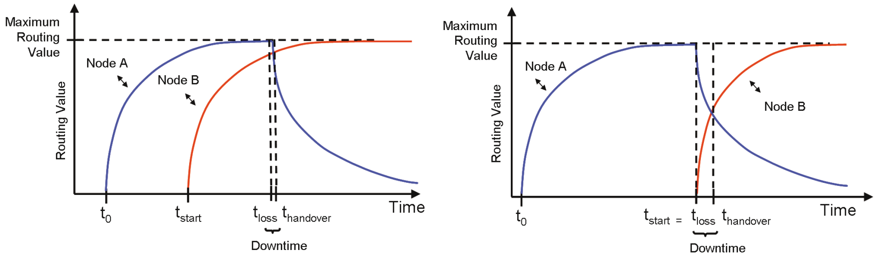
Kuladinithi et al. [6] implemented the AODV [7] protocol using the programming language Java to allow the experimental evaluation of different AODV implementations on various systems. Inspired by the their implementation we decided to implement the SBR protocol [8] to evaluate its performance in a testbed and compare the measurements with the simulation results gathered from the OPNET Modeler simulation.

3 Toplogy Change Detection

Many popular routing protocols like AODV [7] and OLSR [9] use time outs to detect link breaks. In the following we refer the time interval as downtime that is required by the routing protocol to detect a topology change and to find a new valid route. The problem is to find an optimized expiry interval in order to minimize the downtime on the one hand, and the probability that a link is untruly assumed to be broken due to short temporary interference on the other hand. For that reason, we have chosen the SBR protocol since it is able to detect link breaks within a short amount of time.

SBR is based on the concept of Directed Diffusion which is a data-centric and application aware paradigm which was introduced by C. Intanagonwiwat et. al. [10]. Each node in the network broadcasts (hello) messages. These messages are similar to route requests in AODV and are forwarded by intermediate nodes. The neighbor through which a new hello message was received is rated by using a cumulative function. Thus, a higher routing value corresponds to a higher link quality as a consequence of the cumulative function. Figure 1 shows the downtime of the SBR protocol in the average case and in the worst case depending on the development of the corresponding routing entry values.The graphs represent the routing entry values of node A and B towards a destination node X from the perspective of another node C. At time t_0 node C receives the first routing message via node A and at time t_{start} from node B respectively. The link between node A and node C breaks at t_{loss}. The alternative route via node B becomes the route with the highest routing entry at $t_{handover}$.

Node C forwards its data traffic for a destination node X either to node A or node B. Node C always selects the neighbor with the highest routing entry towards the destination as next hop. Therefore, the downtime represents the time until the routing value of the alternative route becomes higher than the previously best route. A topology change is detected quickly if the routing values are on an equal level from the time when a link break occurs. Thus, the worst case is represented by a scenario in which the routing entry of a single neighbor is very high whereas the routing values of all other neighbors are very low.



(a) Average case

(b) Worst case

Fig. 1. Topology change detection

4 Simulation

We use a framework for WSNs which we have developed with the OPNET Modeler 14.0 [3] software to simulate all layers of the communication stack. For this particular simulation, we decided to use the IEEE 802.11 mac and physical layer that is provided by OPNET to allow the comparison of the simulated performance of the video application with the performance of a testbed using IEEE 802.11g interfaces. Therefore, we put our network and application layer on top of OPNETs mac and physical layer to simulate the communication stack. Trace files were recorded from video applications which are used for traffic generation within the simulation in order to generate realistic traffic patterns.

In addition, we implemented a filter process which allows us to dynamically modify the signal propagation to simulate unpredictable signal loss which leads to major topology changes in the network. Furthermore, the process is used to limit the signal propagation such that only a certain topology can be used which corresponds exactly to the topology in the testbed. The statistics of the data traffic are collected by a centralized node to allow a more flexible evaluation and visualization. Thus, we extended the data source and sink OPNET models to meet our requirements of traffic generation and evaluation.

5 Implementation

The programming language Java was used to implement the routing protocol to allow its usage on different Operating Systems (OS). Most common OSs, e.g. Linux and Windows, come with tools that allow the modification of their routing table without much effort. Therefore, the Java routing application has to detect which OS is used in order to know which commands are used by the OS. This enables us to manipulate routes in the table without the need of notifying other applications. The implementation consists of three major packages. The first one is represented by the network package which is used to receive and transmit data packets via the IEEE 802.11g interface. We use the Jpcap 0.7 [11] library which is based on WinPcap to grab packets from the interface. The second package covers configuration, routing table, and time management functions, e.g. timer and statistic tasks, which are then used by the routing protocol. The behavior of the routing protocol and the used messages build the third package. Incoming packets are detected and evaluated by a receiver task which sends a callback to the routing task to further evaluate the packet. The routing task then decides what actions have to be performed according to the content of the packet, e.g. modification of the routing table, changing of routing entries, forwarding or dropping of the packet. Additionally, periodic tasks like hello message transmission timer or routing entry decrease timer send callbacks to the routing application.

We added a filter class to the network package which covers the same functionality as the filter process within the OPNET simulation. Thus, we can restrict the topology of the testbed according to the topology used in the simulation. Furthermore, time triggered topology changes can be used to study the behavior of the protocol to deal with link breaks depending on its configuration.

6 Video Quality Evaluation

The standard method to assess the performance of video transmission systems is to calculate the Peak Signal to Noise Ratio (PSNR) between the source and the received (possibly distorted) video sequence.It is a differential metric which is calculated image-wise and very similar to the well-known SNR but correlating better with the human quality perception [12]. The PSNR calculation yields a quality indicator for each image of the video sequence in relation to the original image. Thus, this metric is only meaningful if the quality of the original image sequence is high in terms of human perception which is not necessarily the case. For instance, if the video sequence is passed through a state-of-the-art video encoder to reduce the bit-rate the compressed video will be already distorted since modern video-codecs – like MPEG-4 or H.264 – are usually lossy. Loss of packets will lead to decoding errors at the decoder/player while delay can cause buffer under-runs. Both will ultimately cause the loss of images at the player. Since modern video-codecs make extensive use of the temporal redundancy (encoding only the differences) in most videos, the loss of single images also leads to the distortion of all following images that are differentially encoded based on the lost image. Lost frames usually will cause the video player to ”freeze” – to show the last successfully received and decoded image. It is important for an image-by-image metric to reproduce this behavior in case of transmission losses or delay in order to avoid alignment issues between the source and the received video. For a better illustration of the meaning of quality measures for non-experts the ITU-R developed a quality indication scale which is tied to the quality impression of human observers [14]. This scale is shown in Table 1.

ITU-R recommendation BT.500 [14] further describes a methodology to gain these quality indicators by subjective assessment series (by a group of humans). Such a scale is often called Mean Opinion Score and used in several quality assessment systems. In [13] there is a mapping of PSNR values to MOS values which can be used to roughly estimate the human quality perception for videos with relatively low motion (like for instance videos from surveillance cameras). This mapping from PSNR to MOS is shown in Table 2 and used in this paper.

A MOS value is assigned to each image according to Table 2 which is based on the PSNR values that are calculated for every single image of a received video sequence. These values are averaged over all images of a sequence to produce a single quality indicator for a video transmission as proposed by the methodology

Table 1. ITU-R quality and impairment

Scale	Quality	Impairment
5	Excellent	Imperceptible
4	Good	Perceptible
3	Fair	Slightly annoying
2	Poor	Annoying
1	Bad	Very annoying

Table 2. PSNR[dB] to MOS conversion

PSNR	MOS
> 37	5 (Excellent)
31 - 37	4 (Good)
25 - 31	3 (Fair)
20 - 25	2 (Poor)
< 20	1 (Bad)

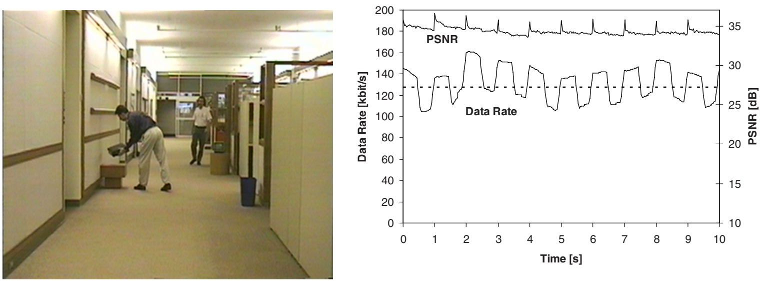
described in [14]. However, averaging can be problematic for long videos since short temporal distortions will not influence the average significantly. To avoid this effect another depiction of the MOS values of the single images of the video sequence is given in this paper in addition to the average values. The percentage of images with a certain MOS is displayed and compared to the original video.

In this paper we evaluate the performance of a wireless video transmission system by means of measurements and simulations taking into account the general principles of video quality evaluation. Therefore, we use a video quality evaluation tool-set – called EvalVid – which provides the necessary tools [15].

7 Results

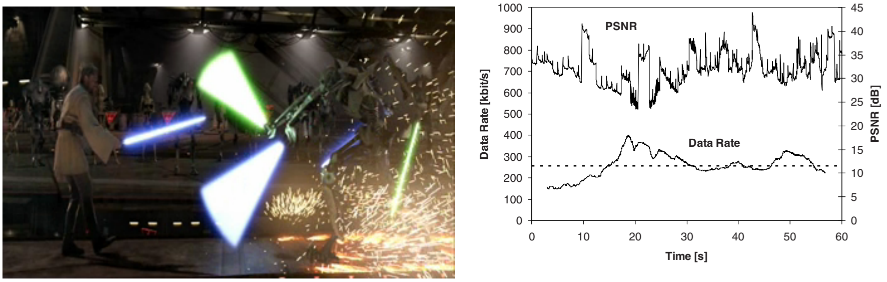
We selected one of the standard video sequences which is used by a variety of video encoding and transmission studies by, e. g., the Video Quality Experts Group [16]. This video sequence is called ”Hall Monitor” and consists of 300 frames in CIF resolution (352x288 pixel) with 30 Hz frame rate. It is a relatively low-motion sequence so that the PSNR to MOS mapping shown in Table 2 can be applied. Due to the fact that it is only 10s long, we concatenated the sequence six times. Since the video is recorded with static camera and there is little motion in the scene, the influence of this concatenation on the video encoder performance is low – even at the junctions. The resulting one minute long video was then encoded with the state-of-the-art H.264 video encoder x264 [17] with an average target bit-rate of 128 kbit/s. A key-frame was encoded every second in order to have a good balance between coding efficiency and error recovery capabilities. To give a better impression of the video sequence used, Figure 2 displays a sample image together with the bit-rate profile and the PSNR between the encoded and the original video.

A second video sequence with more motion was selected in order to stress the performance evaluation methodology with content appropriate for IFE. Figure 3 shows a sample image from the one minute long scene from the Movie “Star Wars III”. The resolution is 360x216 pixels and the frame rate is 25 Hz. With current video encoding technology it is not possible to achieve an acceptable



(a) Sample image (b) Video profile & quality

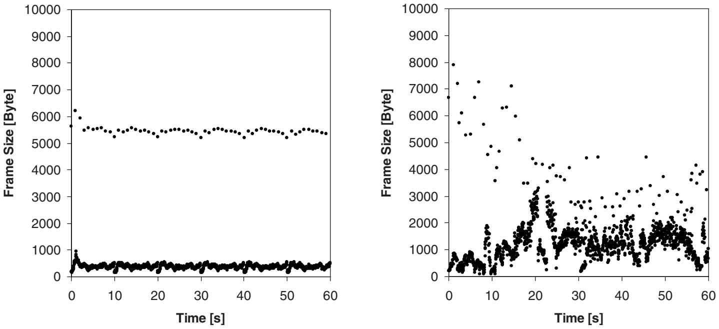
Fig. 2. Profile of “hall” video clip, (a) sample image (b) data rate & PSNR profile



(a) Sample image

(b) Video profile & quality

Fig. 3. Profile of “sw3” video clip, (a) sample image (b) data rate & PSNR profile



(a) hall clip

(b) sw3 clip

Fig. 4. Frame sizes of (a) “hall” video clip and (b) “”sw3“” video clip

PSNR with an average target bit-rate of 128 kbps. Consequently, the video clip was encoded with a target bit-rate of 256 kbps. The different content of the selected clips is also reflected in the variations of the size of the encoded frames as shown in Figure 4. While the only variations in the hall clip are basically the different sizes of the I and P frames, the frame size fluctuations in the sw3 clip are much higher.

In order to calibrate the simulation with the measurements we performed a set of test runs in a specific scenario. Five wireless nodes are initially connected to each other and transmit the encoded hall video. sequence using RTP (Real-time Transport Protocol) [19] from Node 1 to Node 5. The nodes in the simulation and the testbed are configured such that they are forced to build a string topology. Thus, they are only able to receive messages from their direct neighbors. Node 5 represents an exception since it temporarily connects to the other nodes as shown in Figure 5. We have chosen this extraordinary example due to the fact that it is the worst case scenario for the routing protocol. A description of the connectivity during the simulation and the measurement is given in Figure 5. Note, the u_1, u_2, and u_3 are random variables which are selected at the

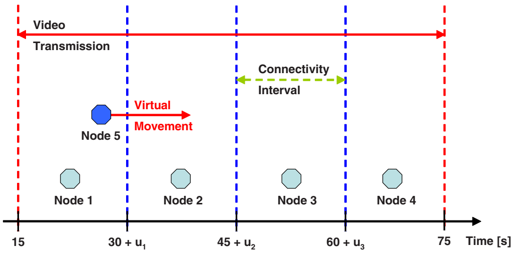
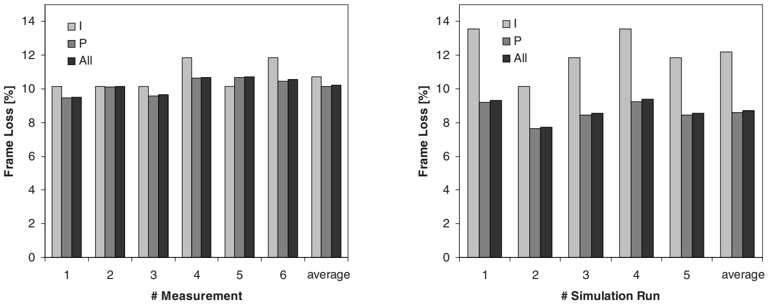


Fig. 5. Illustration of the connectivity during video transmission

beginning of the simulation according to a uniform distribution between -1s and 1s. The variables are required to shift the disconnection times in order to avoid the alignment with I frames.

In the 15+u seconds interval, the direct connections between Node 5 and Node 2, 3 and 4 respectively were detached which caused the system to find a new route. This represents a relatively harsh scenario since abrupt disconnections represent the worst case for real-time applications. The most relevant parameters represent the hello message interval and the routing decrease interval which are set to 1s. Due to the fact that the other parameters have no significant impact in our scenario we skip their description since a detailed description of all parameters is given in [8]. Furthermore, the implementation and the simulation will be made available to download from [21].

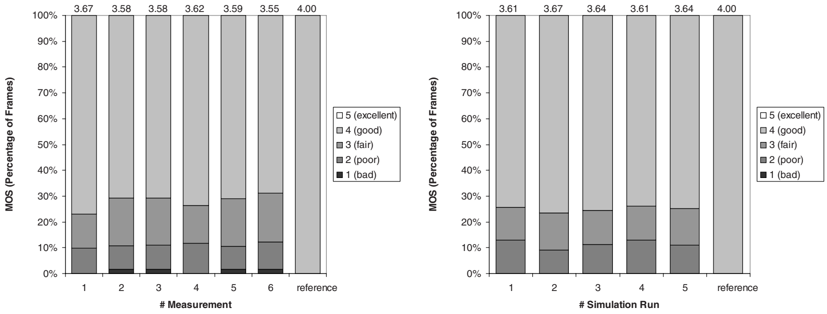
Using EvalVid, a trace of the video file was generated, containing the size and type of each video packet transmitted over RTP. Additionally, an IP-level packet trace was created using Wireshark [20] at the transmitting and receiving node. These traces were used by EvalVid to calculate packet and frame loss figures as well as reconstructing the received (possibly distorted) video files. The received videos were then decoded using FFmpeg [18] to be able to calculate the PSNR and MOS figures for the video quality evaluation. Figure 6 compares the frame loss of the measurements and the simulations while Figure 7 shows the corresponding MOS values for the received video. The overall frame loss is slightly higher for the measurements which is caused by single packet losses due to interferences, multi-path propagation, and moving obstacles. Moreover, the percentage of key I frames lost in the simulation was slightly higher which was quite suprising. A closer look at the trace files revealed that the starting times of the disconnections were varying more during the measurements due to the human reaction time. Against, in the simulations the disconnection interval was quite stable and accidentally always during an I frame transmission. This effect is avoided in the following parameter study by equally distributing the disconnection intervals in a certain range. The bars in Figure 7 show the percentage of frames with a certain MOS in comparison to the reference videos (rightmost) MOS distribution. The reference video reflects the coding loss and consists of 100% frames with a MOS of 4 (good). In contrast to the raw frame loss these results include the quality degradation caused by frames that could not be correctly decoded due to losses of previous frames. Though the key frame loss in



(a) Measurement

(b) Simulation

Fig. 6. Hall - Comparison of frame loss, (a) measurements (b) simulations



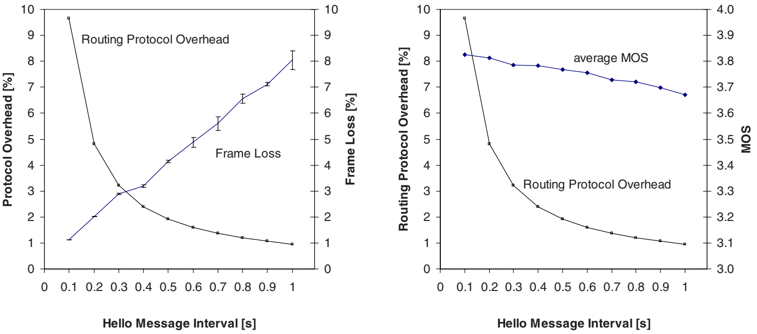
(a) Measurement

(b) Simulation

Fig. 7. Hall - Comparison of MOS, (a) measurements (b) simulations

the simulations was higher the quality of the video was worse in the measured scenario. This is caused by the rare random single packet losses during the measurements which influence all following P frames after the packet loss until the next I frame. The impact that single packet losses have on the MOS and PSNR depends on the used encoder, its configuration e.g. the I frame rate, and the type of video which is encoded e.g. action sequence or landscape stills.

Considering the differences between the measurements and the simulation, the loss and MOS statistics are similar enough such that we can focus on the simulation in order to evaluate the performance of the testbed. In the following we want to demonstrate how to use the simulation for performance evaluation and parameter optimization of the routing protocol to achieve an acceptable video quality even in the case of abrupt disconnections. Thus, we varied the hello message interval of SBR from 1.0s down to 0.1s in steps of 0.1s and transmitted the video 100 times for each setting. The scenario simulated was again the multi-hop setup with three abrupt disconnections in a 15s interval. The exact

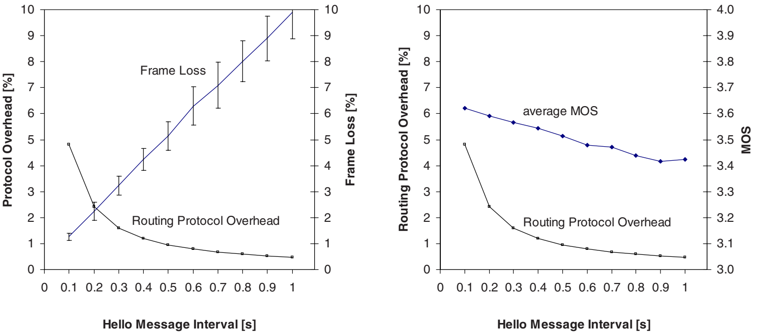


(a) Frame loss (b) MOS

Fig. 8. Hall - Average frame loss (a) and average MOS (b) against the routing overhead

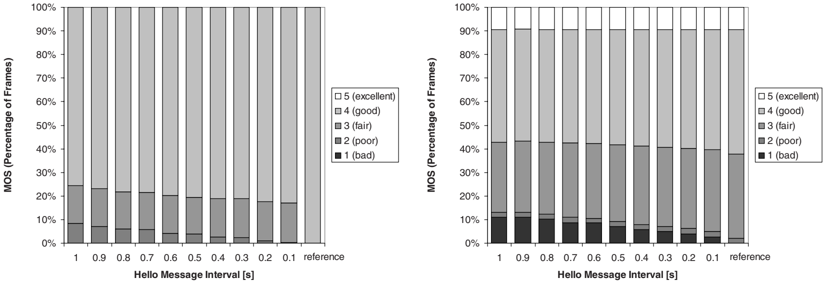
disconnection times were equally distributed in a window of ±1s to avoid the exact alignment with a key frame.

Figure 8 shows the resulting average frame loss as well as the average MOS against the overhead of the routing protocol in percent of the video traffic for the hall clip. Though the frame loss varies between 1% and 8% the average MOS only varies between about 3.7 and 3.8. The reason for this is that each lost frame can influences the following frames up to the next key frame. Figure 10 shows the percentage of frames with a certain MOS. Due to the fact that the expressiveness of the average MOS is limited in case of longer videos. Figure 9 shows the frame loss and MOS statistics for the high-motion scene sw3. Although the frame loss rate is not higher than in the low-motion hall clip, the average MOS is suffering more from the losses. This results from the higher differences between adjacent frames which lead to a higher sensitivity to lost frames. Another factor is the appearance of frames with a very low MOS (1-2). In fact, the disturbances of the



(a) Frame loss (b) MOS

Fig. 9. SW3 - Average frame loss (a) and average MOS (b) against the routing overhead



(a) hall

(b) sw3

Fig. 10. Percentage of frames with certain MOS values depending on the hello message interval in comparison to the reference video quality

video quality are short in both investigated cases. Figure 10 shows the number of frames with a certain MOS in comparison to the undistorted reference. In contrast to the average MOS curves in Figure 8(b) and Figure 9(b), it is shown here that the quality impact on the sw3 clip is much smaller than on the hall clip, which results from the faster recovering in case of losses due to the higher number of intra-coded parts.

The overhead of the routing protocol rises exponentially with the downsizing of the hello message interval. It is acceptable up to around 2-3% of the application traffic, since this is in the range of the protocol overhead of RTP (1.7% in this scenario). The relative routing overhead is lower for the sw3 scenario since the bit-rate is higher than in the hall scenario. The MOS distribution bars in Figure 10 show that the difference in quality between the message interval of 0.3s and 0.4s is not noticeable by human observer. Considering the smaller overhead a hello message interval of 0.4s of the SBR protocol would be optimal in this scenario regarding the perceived video quality.

8 Conclusions and Future Work

We studied the performance of the network to deal with frequent topology changes depending on the duration of the hello message interval of the SBR protocol. The parameter hello message interval was adjusted in order to achieve an acceptable video quality which was measured by comparing the MOS value of the source and the received video.

The implemented simulation and performance evaluation framework will be used to optimize intra-aircraft applications ranging from surveillance to entertainment. Since the criticality of the applications is different, it is essential that the evaluation methodology reflects the specific requirements in terms of perceived quality. We are confident that the performance evaluation framework introduced in this study is useful for other researchers who want to assess the performance of a wide range of wireless video transmission systems.

References

1. Akyildiz, I.F., Melodia, T., Chowdhury, K.R.: Wireless Multimedia Sensor Networks: A Survey. IEEE Wireless Communications Magazine 14(6), 32–39 (2007)
2. Network Simulator, ns-2, http://www.isi.edu/nsnam/ns/
3. OPNET Modeler, University Program, http://www.opnet.com/services/university/
4. Kotz, D., et al.: Experimental Evaluation of Wireless Simulation Assumptions. In: Proceedings of the 7th ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems, pp. 78–82. ACM Press, New York (2004)
5. Liu, D., et al.: Simulation Validation Using Direct Execution of Wireless Ad-Hoc Routing Protocols. In: Proceedings of the 18th Workshop on Parallel and Distributed Simulation PADS, pp. 7–16. ACM Press, New York (2004)
6. Kuladinithi, K., et al.: Experimental Evaluation of AODV Implementations. In: CEWIT 2004, USA (2004)
7. Perkins, C.E., Belding-Royer, E.M., Das, S.R.: Ad Hoc On-Demand Distance Vector (AODV) Routing RFC 3561, IETF MANET Working Group (August. 2003)
8. Klein, A., Tran-Gia, P.: A Statistic-Based Approach towards Routing in Mesh Networks. In: IEEE MASS, Pisa, Italy, October, pp. 1–6 (2007)
9. Clausen, T., Jaqcquet, P.: Optimized Link State Routing (OLSR) RFC 3626, IETF Networking Group (October 2003)
10. Intanagonwiwat, C., Govindan, R., Estrin, D.: Directed Diffusion: a Scalable and Robust Communication Paradigm for Sensor Networks. In: Proceeding of ACM MobiCom 2000, Boston, MA, pp. 56–67 (2000)
11. Jpcap - Java library for capturing and sending network packets, JPCAP, http://netresearch.ics.uci.edu/kfujii/jpcap/doc/
12. Riley, M., et al.: Digital Video Communications. Artech House (1997)
13. Ohm, J.-R.: Multimedia Communication Technology. Springer, Heidelberg (2004)
14. ITU-R, Recommendation BT.500-10, Methodology for the Subjective Assessment of the Quality of Television Pictures (2000)
15. Klaue, J., et al.: EvalVid - A Framework for Video Transmission and Quality Evaluation. In: Proceedings of the 13th International Conference on Modelling Techniques and Tools for Computer Performance Evaluation, pp. 255–272 (2003)
16. Video Quality Experts Group, http://www.vqeg.org/
17. Merritt, L., et al.: x264 - a free h264/avc encoder, http://www.videolan.org/developers/x264.html
18. FFmpeg - Multimedia Framework, http://ffmpeg.mplayerhq.hu/
19. Schulzrinne, H., et al.: RTP: A Transport Protocol for Real-Time Applications, RFC 3550
20. Wireshark - Network Protocol Analyzer, http://www.wireshark.org/
21. Overview of Currently Available Routing Protocols (under construction), http://www.routingprotokolle.de/