ABSTRACT
The core problem for any adaptive video streaming solution, particularly over wireless networks, is the detection (or even prediction) of congestion. IEEE 802.11 is especially vulnerable to fast movement and change of antenna orientation. When used in UAV networks (Unmanned Aerial Vehicles), the network throughput can vary widely and is almost impossible to predict. This paper evaluates an approach originally developed by Kofler for home networks, in a single-hop UAV wireless network setting: the delay between the sending of an IEEE 802.11 packet and the receipt of its corresponding acknowledgment is used as an early indicator of the link quality and as a trigger to adapt (reduce or increase) the video stream's bitrate. Our real-world flight-tests indicate, that this avoids congestion and can frequently avoid the complete loss of video pictures which happens without adaptation.

Categories and Subject Descriptors
C.2.1 [Computer-Communication Networks]: Network Architecture and Design—Wireless communication

Keywords
Video Streaming, Adaptive Streaming, UAVs, UAV Communication

1. INTRODUCTION
In most application scenarios, UAVs will require a permanent link to some base or control station. Especially in civilian and scientific systems, this link will often use the IEEE 802.11 wireless LAN technology. 802.11’s greatest advantages are that it is cheap, readily available, well understood by developers, and widely supported by software and operating systems. However, 802.11 was never intended to be used for fast moving and direction changing devices. The intended use cases were static devices like laptops. The high mobility of UAVs leads to ever-changing antenna orientations and network topologies. The induced problems can be so severe that some authors [2, 14] suggest that a new wireless LAN standard might be needed. Asadpour et al. [2] write that the “special ecosystem [of UAV networking] undoubtedly requires a rethinking of wireless communications and calls for novel networking approaches.”

Streaming a video over such an unreliable channel causes additional problems:
• Video streaming requires a lot of bandwidth,
• reacts strongly to link degradation, and
• a single missing packet can cause the loss of several seconds of video data.

Yet the quality of video streams can be adapted to fit (almost) any available bandwidth. If a link degradation can be detected (or even predicted) in time, it is possible to reduce the stream’s bitrate and avoid congestion and a complete loss of the video picture.

The goal of our research was to address these problems and prevent the video signal’s complete loss by using a method described by Kofler [8, 9] to indicate near-term congestion and to adapt the UAV’s video stream bitrate to the current network conditions.

2. RELATED WORK
2.1 Wireless Networks
In a survey paper, Bekmezci et al. [4] identify communication as the biggest challenge for multi-UAV systems and coin the term “Flying Ad-Hoc Networks (FANETs).” This term is inspired by previous work on “Mobile Ad-Hoc Networks” (MANETs) and “Vehicle Ad-Hoc Networks” (VANETs) and can be considered a special form thereof. MANETs generally consider people moving in an environment using mobile devices and VANETs generally consider cars moving on a road. Bekmezci et al. differentiate FANETs from MANETs and VANETs for several reasons:
• UAVs move much faster and have more degrees of freedom than people or cars.
• Because of UAVs’ higher mobility the network topology will change much more frequently.
• FANETs deliver control commands as well as sensor data.
The distance in MANETs is usually much greater than in VANETs. Thus communication hardware needs to be different.

Andre et al. [1] focus on the design choices of aerial communication networks and cover all common wireless communication technologies, namely IEEE 802.15.4 (XBee Pro), IEEE 802.11 (WiFi), 3G/LTE (mobile phone network), and infrared. For video streaming from UAVs, Andre et al. recommend IEEE 802.11 technology. They find that WLAN can deliver the required data rates even over longer distances and that this distance can easily be extended by using multi-hop networks. They conclude that “(...) the measured throughput confirms the practicality of WLAN. Nevertheless, achievable throughput heavily depends on antenna placement and the mobility of the MAV [micro aerial vehicle], and technological improvements are required to guarantee necessary throughput." [1]

2.2 Video Streaming

Despite the prominence of the use case, there is scant literature that focuses on UAV networks (or MANETs) and video transmission. Most papers seem to focus either on the task of recording a video (quality, coding, object tracking, etc.) or on the UAV network and its properties. The authors are not aware of any paper describing a system that combines networking and adaptive video streaming or other video transmission techniques. For MANETs, however, Lindeberg et al.’s [10] survey paper is a good resource to get an overview of the “challenges and techniques” for video streaming in MANETs. They stress the fact that meeting bandwidth, delay and jitter constraints in wireless networks is far more challenging than in wired networks. They state that “(...) a significant amount of overall packet drops in wireless networks are perceived as being random, rather than caused by congestion (...)”.

Although technologies like Scalable Video Coding (SVC) [13] or lightweight encoders [5] will become interesting in the future, there are currently no common off-the-shelf solutions that make them readily available (especially for real-time applications). So in real-world settings, one is reduced to tweaking the H.264 encoder’s parameters.

2.3 Congestion Detection

Kofler [8, 9] combines the idea of packet pairing and passive sensing from the field of bandwidth estimation (e.g., [6, 12]) with the fact that video streams (using H.264 and RTP) will have many packets of equal size. He presents a proxy server that can adapt SVC video streams according to current network conditions. However, “instead of reacting on packet loss, [the] approach uses an increase in queueing delay at the router to detect phases of throughput degradation." [9] Kofler measures the queueing delay indirectly. He noticed that the Linux kernel’s wireless monitoring interface will only show frames after they were acknowledged by the recipient. So the worse a link is, the longer the time between sending of the frame and the receipt of the corresponding acknowledgment will be (because, for example, a frame has to be retransmitted). It could be shown that this queueing delay corresponds with the link quality and the packet loss. Increasing delay usually precedes actual packet loss by almost two seconds. This approach has three advantages:

1. The information is available locally – no extra information (like sync or probe messages) needs to be transmitted.
2. Being a local approach, it does not need any support from clients. Any client that supports RTP and H.264/SVC can be used.
3. An increase in queueing delays precedes packet loss, thus allowing fast reaction to upcoming link degradation.

The adaptation logic then works as follows: If the delay exceeds a certain threshold, the proxy will reduce the video’s quality. If, however, the proxy measures a delay which is lower than the threshold, the proxy will evaluate past delays and if it finds them low as well, it will try to increase the stream’s quality. Previous packets serve as “packet pairs” and their delay is used to estimate available link capacity. This works because video stream packets are usually the same size and sent back-to-back.

3. HYPOTHESES

Kofler [8, 9] could link the delay between the sending of an IEEE 802.11 packet and the receipt of its acknowledgment to the link quality and adapt a video stream accordingly. This paper tries to apply this idea to UAV video streaming and to evaluate three hypotheses:

1. The delay is related to video quality.
2. The delay is a good indicator of impending problems with the video stream.
3. The delay can be the input for an adaptation proxy that improves the overall video quality.

Kofler could show that this delay is a good indicator for impending link degradation in a home-use setup. This setup consisted of a common-off-the-shelf router and a laptop. Because the scenario was home use, the movement evaluated was only at walking speed (a person walking through his/her flat). For this slow movement, the approach worked remarkably well. However, as touched upon in the introduction, UAV networks change much faster and more violently. This paper provides the first evaluation of an adaptive video streaming approach in such an environment.

4. SYSTEM DESIGN

4.1 Overview

The experiments used a Logitech C920 HD webcam [11] mounted on an Ascending Technologies (AscTec) Firefly [3] UAV (shown in Figure 1). It carried a “Mastermind board” that was outfitted with a 1.86GHz processor and 4GB of RAM. The wireless card was a common off-the-shelf Atheros AR93xx chipset with three external antennas and IEEE 802.11 a/b/g/n support. The system used AscTec’s Ubuntu 12.04 (LTS) image with Linux kernel 3.5.

Figure 2 shows an overview of the system architecture. The camera delivers an H.264 video stream and is controlled by gstreamer [7], an open-source multimedia framework. gstreamer can request (almost) any bit rate from the camera and passes the H.264 encoded video stream on to the proxy as a normal RTP stream. gstreamer (or more exactly, the shell script controlling it) then offers the proxy...
an interface to request a rate change. In one thread, the proxy forwards the RTP stream unaltered to the base station (that is running the video streaming client software), but keeps track of when each RTP packet is sent. At the same time, another thread listens on the wireless monitoring interface of the Linux kernel for the MAC layer ACK of the sent frame. Depending on the delay between the sending of an RTP packet and its acknowledgment, the proxy will adapt the stream.

The decision when to adapt is directly influenced by the three observations mentioned above. The algorithm tries to avoid delays higher than 1.0 seconds by reducing the video stream’s bit rate as soon as the delay crosses the threshold of 0.3 seconds. As it usually takes some time for the delay to react to the lower bitrate, i.e., to decrease, the algorithm will wait for a period of 25 consecutive packets (“cooldown period”) before reducing the quality further. Without this cooldown period, the proxy would reduce the quality down to the lowest level most of the time.

When there is an extended period (250 packets) of packets with a low delay (<0.1 sec), the proxy will increase the bit rate by one level. Please note that all these values were gathered empirically. After the initial experiments, these values were determined to give the best results. However, tuning these values was not a priority – so there are probably combinations that would provide better results.

5. RESULTS
The final data set consists of 12 evaluation flights (9 with adaptation, 3 without adaptation) taken on two consecutive days (May 5–6, 2014). The flights were conducted in an area close to the University and the UAV was controlled by one of the authors. During the flights, the UAV was flown towards a small wood, where three white hay bales were “hidden.” The goal of the flights was to find the “hidden” hay bales. Around 50 flights were conducted, however, only 12 used the same software version and settings and are thus comparable. Each test flight produced two artifacts: (1) a log file that contains information about the packets and (2)
the recording of the video stream as it was received at the base station. The log file was stored on the UAV and contains the following variables:\(^2\): distance in meters between the UAV and its take-off position; UNIX timestamp in seconds; time delay between the sending of a packet and the receipt of the ACK in seconds; RTP sequence number. Flights that performed adaptation logged an integer representation of the quality level that was being used and the number of packets with a “low” delay (less than 0.1 seconds).

In contrast to the log files, the videos are impossible to analyze objectively as there is no “perfect” video, i.e., a reference, to compare them to. There are no clear-cut indicators when a video signal is “good” and at what point it becomes “bad.” In the course of this work, “good” means that significant features on the ground can still be made out (even though the resolution is low) and “bad” means that artifacts blur the picture beyond recognition. Figure 5, for example, shows a low resolution image of the hay bales. Although this image is of significantly lower resolution than the image in Figure 4, the hay bales can still be made out and the image is thus considered “good.” Figure 6 on the other hand is blurred beyond recognition and is thus considered “bad.”

Figure 4: High quality frame showing the hay bales

Figure 5: Low quality frame showing the hay bales

5.1 Characteristics of the Delay

Figure 7 shows how quickly the delay can rise from low levels to high peaks. Without a mechanism to counter the effect, the delay will rise until the UAV stops moving and/or the antenna orientation improves.

When active bit rate adaptation takes place, it can counter the rise of the delay and keep the periods of high delay shorter. Figure 8 shows how the quality level is reduced as the delay crosses the threshold of 0.3 seconds. It does not fall immediately, because the cooldown period keeps the bit rate too high. As soon as quality level 3 is reached, the delay immediately falls.

Figure 6: “Bad” frame, with hay bales lost due to artifacts

Figure 7: Development of delay without adaptation

Figure 8: Development of delay with adaptation

The following figures were created by aggregating the data from all 12 flights. The first three bars (in blue) in each figure are the aggregate value of all flights (both with and without adaptation). The next nine bars (in red) show only the flights with adaptation and the final three bars (in yellow) show only the flights without adaptation.

5.2 Delay

Figures 9 and 10 show the maximum and mean delay, respectively. It is interesting to see that the flights with adaptation had a significantly lower mean delay than the flights without adaptation. While the data set is small, these results confirm Koller’s findings and suggest that the delay is indeed a very good indicator of the link quality.

\(^2\)The raw data files can be found on http://kacianka.at/ma/raw_flight_data.zip.
5.3 Number of Packets to Low Delay
Figures 11 and 12 show the number of consecutive packets that have a delay greater than 0.3s. 0.3s is the threshold used to reduce the bit rate of the video. The fewer packets have a high delay, the better the video quality is. The flights with adaptation have far shorter periods of “bad connection” than the flights without adaptation. As longer periods of lower delays mean a better video quality, these results suggest that applying Kofler’s congestion detection method and adaptive video streaming to UAV networks will indeed lead to a better video picture quality.

6. DISCUSSION
In this paper three hypotheses were presented:

1. The delay is related to video quality.
2. The delay is a good indicator of impending problems with the video stream.
3. The delay can be the input for an adaptation proxy that improves the overall video quality.

Kofler [8] could confirm these hypotheses in the context of (indoor) wireless home networks. The work at hand is based on these previous results and evaluated them in a single-hop UAV network. Kofler’s idea was implemented with different tools, but the basic results can be confirmed. This paper found a clear correlation between high delays (higher than 1 second) and bad video quality (up to the complete loss of the video stream). All experiments in the lab as well as around 50 real-world flight tests exhibited this behavior. A connection breakdown without a period of high delays was never observed.

However, sometimes artifacts that do not seem to be related to any measured factors still arise. They might be in the camera’s H.264 encoder or in the decoder. So the delay will not account for all possible problems in a video stream. The results strictly assume that a good connection is characterized by a low delay (less than 0.1 seconds) and that a bad connection is indicated by an increase of the delay.

The results of the real-world flight tests indicate that the described solution works indeed better than just using a static video bit rate. The authors’ (biased) observation is that flying with video adaptation leads to shorter “break downs” of the video pictures and that one can still get a good overview of the area of flight when streaming in low quality. This is backed up by the data displayed in Figure 12 that shows that flights with adaptation have far fewer delayed packets than flights without adaptation.

In summary, this paper cautiously confirms all three hypotheses in the context of UAV networks. However, the number of test flights was low and thus the results should be considered preliminary. They are encouraging and beckon further research.

7. LIMITATIONS OF THE EXPERIMENTAL DESIGN
The experiments’ greatest weakness is the low number of flights. Due to new laws and problems with Austria’s flight control agency Austro Control, the authors were only able to conduct twelve comparable flights of which three were done without video adaptation. “Comparable,” however, does not mean that all influencing factors were actively controlled. On the contrary, the experimental design did not control the UAV’s flight path and the antenna orientation. As laid out in the introduction, Wi-Fi connectivity is highly impacted by antenna orientation, movement speed, and height. All these factors were not logged and not fixed. It is possible that the flights without adaptation were (unconsciously) conducted with a worse UAV positioning. As with the ori-
Another significant weakness is that the Wi-Fi equipment was used with default settings. It is likely that the settings can be tweaked to improve performance and reliability of the connection. Fixing the channel’s bit rate or disabling block acknowledgments could change the video stream’s behavior. Another weak point is that the described solution requires acknowledgment packets, which are not sent for multicast transmissions.

8. REPRODUCIBILITY

These shortcomings in the experimental design violate a cornerstone of the scientific method: reproducibility. While the configuration and setup of the UAVs is easy to reproduce, the evaluation of Kofler’s congestion detection algorithm relies on preliminary “ad-hoc” tests. The aforementioned problems with Austro Control prevented the authors from designing the flight tests more carefully and forced them to rely on data gathered during “proof-of-concept” test flights.

Future work should therefore be mindful of these influencing factors:

• The position of the base station and the UAV take-off spot should be kept stable.
• The UAV’s flight path should not be manually controlled. Flying to a predefined set of GPS coordinates should be preferred.
• The UAV’s altitude and cruise speed should be predefined.
• The UAV’s yaw (and thus the antenna’s orientation) should be predefined.
• The interferences of the fuselage and the mainboard on the antennas should be reduced by placing them on a base that extends below the UAV.
• All wireless traffic could be recorded by a third listening station that is positioned along the flight path. Later analysis of this data might yield useful information.

However, as touched upon by [1, 2, 14] reproducibility for UAV networks and the design of testbeds are still open and difficult problems.

9. FUTURE WORK

The results of these experiments suggest that the delay is a good indicator for the link quality. It can be used to adapt the quality of the video stream to fit the available bandwidth. Further research should first establish whether the effect indeed exists and how (statistically) significant the effect really is. A follow-up study could repeat the experiment described in this paper, but take great care to conduct the experiments in a reproducible way.

10. ACKNOWLEDGMENTS

The work was supported by the ERDF, KWF, and BABEG under grant KWF–20214/24272/36084 (SINUS). It has been performed in the research cluster Lakeside Labs.

11. REFERENCES