

Upstream Content Scheduling in Wi-Fi DenseNets during Large-scale Events

Glenn Daneels*, Jeroen Famaey*, Steven Bohez†, Pieter Simoens†, Steven Latré*

*Department of Mathematics and Computer Science, University of Antwerp - iMinds, Belgium

†Department of Information Technology, Ghent University - iMinds, Belgium

Email: glenn.daneels@uantwerpen.be

Abstract—The smartphone revolution and widespread availability of wireless LAN and mobile Internet technologies has changed the way people interact with the world. These technologies can be exploited by event organisers to boost audience involvement and immersion, for example, by integrating user-generated content into the event experience. In this paper, we developed a large-scale event participation platform for the wireless transmission of user-generated videos to be used during the event. Such events often bring together thousands of users on a small geographical area and providing wireless connectivity in such dense environments is highly challenging. We analysed the efficiency of several upload scheduling strategies in Wi-Fi DenseNets based on extensive experiments performed in a shielded lab environment. We showed that intelligent scheduling improved throughput over 20% compared to uncoordinated uploading in a dense network, with more expected gains when the density would further increase. Moreover, we also calculated the theoretical scalability of the platform. Based on our results, we confirm the importance of content scheduling to efficiently utilise WLAN technologies in highly dense environments.

Index Terms—Content scheduling, large-scale user participation, video upstreaming, Wi-Fi DenseNets.

I. INTRODUCTION

The smartphone revolution and widespread availability of wireless LAN and mobile Internet technologies has changed the way people interact with the world. These technologies can be exploited by event organisers to boost audience involvement and immersion, for example, by integrating user-generated content into the event experience. In this paper we focus on a large-scale event participation platform for the wireless transmission of user-generated videos, which was developed in cooperation with an event organizer in Belgium. The goal of this platform is to increase user experience by allowing crowds attending music concerts to capture videos on their mobile devices and upload them to a central server through Wi-Fi. This user-generated content is displayed live at the event location to involve the crowd in the concert experience. However, not just any content suffices, as the supervising directors prefer video of good quality. The main challenge of this prototype is to support a scale of more than 5000 smartphone users at a high density, while maintaining satisfactory network performance.

When deploying a bandwidth intensive platform in such highly-dense environments, Quality-of-Service (QoS) policies have to be maintained and the non-trivial challenges of Wi-Fi DenseNets [1] are to be kept in mind. An important

challenge is the low per-client throughput due to the high number of devices that are located in a small area and thereby increasing the data collision probability. The densification of the wireless environment also implies high-interference levels leading to more packet errors that further impact the per-client throughput. The network back-end of the system should be able to cope with these DenseNet characteristics so every user can reliably stream its videos.

As we want to simultaneously ensure each client is able to upload a minimum amount of video, take into account directives from directors as well as cope with the characteristics of a dense wireless network, care needs to be taken when and how content is uploaded. If the assumption is made that as many videos as possible need to be collected (i.e., directives are only for content type, not content count), it makes sense to schedule uploads in such a way as to maximize throughput. However, as a prerequisite to create a scheduling algorithm, the impact of different uploading strategies on the network needs to be measured. In this paper we try to answer the question: “Which user generated content scheduling technique is best to maximize performance in a Wi-Fi DenseNet?”.

The contributions of this paper are twofold. First, the architecture of the developed large-scale upstream intensive user participation platform is presented. The individual functionalities of the different components are explained together with the component communication. Second, in order to find a scheduling strategy that can support the broadband network traffic we have performed experiments with Commercial Of-The-Shelf (COTS) smartphones in a shielded lab environment. These experiments aim to evaluate the scalability of the system and to examine the maximal throughput as a basis to dimension the system correctly for deployments with thousands of users. Specifically, the experiments compare the cases of strictly sequential upload scheduling with parallel scheduling, where a varying number of clients upload simultaneously. Based on the results obtained during these small-scale experiments, we discuss the amount of recorded video that the system is able to upload in deployments up to 5000 users.

The remainder of this paper is organized as follows: in Section II the works related to this paper are summarized. In Section III, the large-scale user participation architecture is explained and illustrated. Section IV first briefly discusses both the experiment environment and hardware, then it lists

and discusses the results of the different experiments and we calculate the theoretical scalability of the platform. Finally, Section V presents the conclusion of this work.

II. RELATED WORK

Work on better support for Wi-Fi DenseNets is still at an early stage: it was only in May 2013 that the IEEE802.11 High Efficiency WLAN Study Group (HEW SG) concerned with densely deployed Wi-Fi networks, was established. One year later, a Task Group was assigned to start the standardization of the work in the high-throughput IEEE 802.11ax-2019 amendment that will replace both IEEE 802.11n-2009 and IEEE 802.11ac-2013 amendments [2]. The goal is to increase user experience by improving average per-client throughput and area throughput in dense in- and outdoor Wi-Fi networks in interference-prone environments. In order to adjust to the dense environments multiple techniques at the PHY and MAC layer are considered. For example, dynamical channel bonding [3] and Orthogonal Frequency-Division Multiple Access (OFDMA) [4] are applied to alleviate the spectrum usage. OBSS interference handling techniques such as Clear Channel Assessment (CCA) and Transmit Power Control (TPC) should allow for better control of the interference levels in the network environment [1]. In combination with other state-of-the-art PHY and MAC layer enhancements, these techniques should help to satisfy the user experience requirements in dense Wi-Fi networks. Although these advanced techniques are the way forward to properly operate highly dense Wi-Fi networks, it will take time for them to trickle through into COTS products. As such, our paper instead focuses on the use of COTS IEEE802.11n and IEEE802.11ac hardware in combination with intelligent content scheduling to achieve satisfactory performance in dense deployments.

Scheduling data transmissions in (wireless) networks has been a popular research subject the past decade. A well-known approach in fixed networks is to use advance bandwidth scheduling [5], [6], where a certain amount of bandwidth is reserved during a certain time frame for each requested data transmission. For wireless networks, however, much research focuses on packet-level scheduling, whereas the system described in this paper schedules on a content- or file-level. Borst [7] describes a set of flow-level channel-aware scheduling algorithms for dynamic wireless environments that aim to maximize user-level throughput. A two-layer scheduler is proposed by Genguly et al. [8] that first provides guarantees on content download times and uses opportunistic scheduling to maximize the throughput of the entire system. Our system poses different challenges however: instead of content being downloaded to the mobile device, it is uploaded from it instead. Moreover, the specific content being transmitted is not determined by the mobile users but by the supervising directors. Finally, these algorithms are built for cellular networks and do not account for the issues and opportunities specific to Wi-Fi DenseNets.

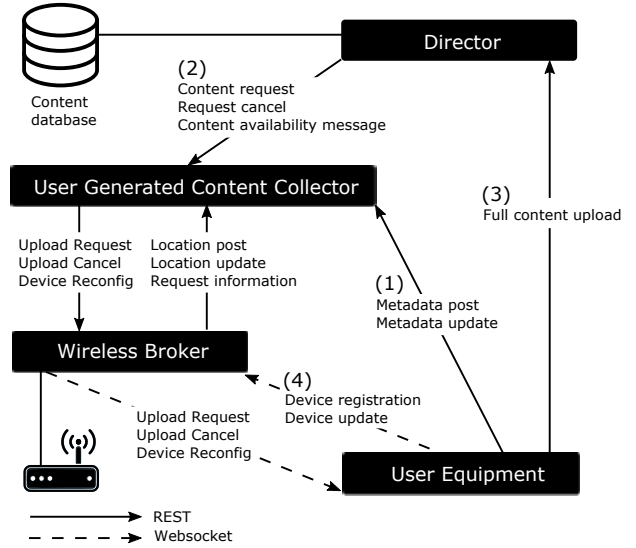


Fig. 1: Architecture of the large-scale user participation platform consisting of the UE, WB, UGCC and DR. Communication is handled either using RESTful web services or WebSockets for the bilateral communication between the UE and WB.

III. SYSTEM ARCHITECTURE

The system architecture of the large-scale user participation platform consists of four components: (i) the User Equipment (UE) being the end-user devices, (ii) the Wireless Broker (WB) that is directly connected to the Wi-Fi hardware, (iii) the User-Generated Content Collector (UGCC) which is responsible for the upstream scheduling and (iv) the Director (DR) that determines what sort of content should be uploaded. The system and communication scheme is illustrated in Figure 1.

A. User Equipment

The UE component of the platform architecture represents all end-user devices that generate content (e.g., smartphones, tablets or wearables). The end-user records their own videos by using the Android application which they are encouraged to install before the concert. This application also allows them to get scheduled to upload their recorded video. The UE wirelessly communicates with the other components of the platform through a set of Access Points (APs) deployed at the concert location. The recorded content is first analysed by the application itself and its metadata (e.g., resolution, brightness, encoding) is sent to the UGCC (1). If the recorded content is requested by DR (2), the application uploads the user-generated content to the system back-end (3).

B. Wireless Broker

The WB component communicates with the different APs to which the UE is connected. Therefore, the WB is responsible for end-user device registration at the platform (4) and has several monitoring tasks. Specifically, it monitors the wireless spectrum and makes sure that the medium does not become

congested due to video uploads, in face of dynamic conditions caused by user mobility and unexpected interference. Moreover, the WB determines the AP and channel each user is connected to and passes this information to the UGCC for use in the scheduling decisions. The WB also regulates the QoS policy which favours video uploads and platform management communication. Users are granted access to the Internet, although this is categorized as best-effort traffic.

C. User-Generated Content Collector

Based on information provided by the WB and requests made by the DR, the UGCC schedules content uploads in order to meet DR requests as soon as possible. When new content is available on the UE, it is advertised to the UGCC while providing any available metadata. This content is compared to the request(s) made by the DR in order to assign it a score. This score is based on the Top- k matching algorithm [9] that matches events with subscriptions. Content are modelled as the events, which are described by a set of attributes and/or attribute ranges that correspond to the available metadata. The requests on the other hand are modelled as the subscriptions, which are described by a set of constraints on the possible attributes. Each constraint has an associated weight which is set by the DR, expressing its importance. The attribute and constraint sets do not have to be complete i.e., contain values for all possible metadata. The score is then calculated as the sum of the weights of the attributes of the event matching the constraints of the subscription. In the case of attribute ranges, prorationing is used. Based on this score, the content is added to a priority queue, waiting to be uploaded. The WB meanwhile notifies the UGCC of any handovers occurring in the wireless network as well as device-specific network conditions, allowing the scheduler to effectively optimize throughput for each individual AP. Requests to upload specific content are first sent to the WB to allow it to optimally manage wireless resources. The scheduler supports two types of scheduling:

- 1) *Sequential scheduling*: multiple clients upload their videos one-by-one to the DR. This technique however works on a per-channel basis meaning that in the case of multiple channels (on different APs or on one AP in case of simultaneous dual band APs allowing traffic both on the 2.4 and 5 GHz frequency band) there will be parallel streams (i.e., one stream per channel) of clients uploading their content sequentially to the DR.
- 2) *Parallel scheduling*: multiple clients upload their content simultaneously to the DR. The number of videos that are uploaded in parallel can be limited to a certain threshold.

In order to find the scheduling strategy that can best support the broadband requirements of the platform, we compare and evaluate the different scheduling strategies in Section IV.

D. Director

The final component in the system is the Director (DR), which determines what types of content should be uploaded by sending general requests to the UGCC. These requests contain directives regarding for example, video quality, user location

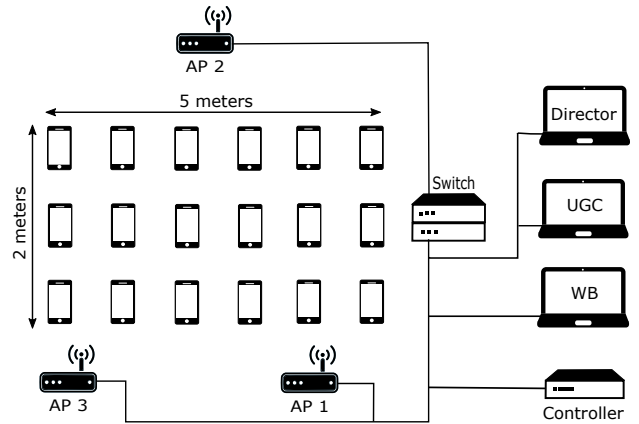


Fig. 2: The shielded lab environment set-up consisting out of a maximum of 18 smartphones, three Meru APs, one Meru controller and the platform components.

and time of recording. Content is uploaded directly to the DR, minimizing the overhead. Finally, the DR also provides an interface that allows for an easy integration of the incoming videos into a professional video-processing workflow which is managed by the (human) show director. This human director has the last call on which videos are selected for display at the event.

E. Communication

Nearly all inter-component communication is implemented using Representational State Transfer (REST) web services, allowing components to communicate using the standardized HTTP protocol. Only the bilateral communication between the WB and UE is implemented using WebSockets, as to avoid having to set up a HTTP server on mobile hosts.

IV. SHIELDED LAB EVALUATION RESULTS

The broadband requirements of the developed platform and Wi-Fi DenseNets in general are challenging. In order to support these requirements, we try to answer the question which user-generated content scheduling strategy has the optimal performance in such an environment, by evaluating and comparing different scheduling strategies. Moreover, we also discuss the impact of multiple APs, the throughput performance of IEEE802.11n and IEEE802.11ac and the performance of the different types of devices. First, the shielded environment set-up in which the experiments are executed, are presented. Afterwards, the results are shown and discussed and we calculate the theoretical scalability of the platform when deploying it at events up to 5000 users.

A. Experimental Set-up

In the shielded lab environment, the platform was tested in a highly controlled setting which suffers minimal wireless interference. This allowed us to determine the best performing scheduling technique in a Wi-Fi DenseNet. The shielded lab

is located at Zwijnaarde, Belgium and is part of the w-iLab.t testbed accommodated by iMinds¹.

The smartphones used during these experiments are listed in Table I. During most experiments, IEEE802.11n (in the 2.4 GHz band) was used except for the experiment described in Section IV-B3 comparing IEEE802.11n and the more recent IEEE802.11ac. Only a channel width of 20 MHz was considered. The 3 used APs were Meru AP832e APs, controlled by a MC1550 Meru WLAN controller. The APs support a 3x3:3 MIMO configuration. A general overview of the set-up is shown in Figure 2. The devices were placed on the ground in an area of 5 by 2 meters, at an approximate equal distance from each other. The three APs surrounded the mobile devices and were set up at a height of one meter.

The idea of the developed platform is to let all people share short moments of their experience at the concert, therefore only videos up to ten seconds are allowed. The videos in these experiments all have that maximal length. The average video size is different for each type of smartphone and is shown in Table I together with its characteristics. In all experiments each client only uploads one video to the platform back-end. The results are averaged over a total of ten iterations and error bars show the standard deviation of those averages. We evaluate the performance using 3 metrics:

- *Per-client throughput*: the network layer throughput of one client. Calculated by dividing the number of network layer transmitted bytes by the upload time.
- *Average throughput*: the average of all measured aggregated network layer throughputs (of all clients) throughout the experiment.
- *Total upload time*: the time difference between the first client starting its upload and the last client finishing its upload.

B. Results & Discussion

In this section, we compare sequential and parallel scheduling, the scalability of the system and the impact of multiple APs. Moreover, we elaborate on the throughput performance of IEEE802.11n and IEEE802.11ac and discuss the performance of the different types of devices.

1) *Scalability and scheduling analysis*: In order to compare parallel and sequential scheduling and its effect on scalability, the throughput and the total upload time is compared. Figure 3 shows the per-client throughput for parallel scheduling as well as total throughput for parallel and sequential scheduling for an increasing number of clients, using only AP 1. As expected, the per-client throughput for parallel scheduling is inversely proportional to the number of clients. Moreover, the average throughput also dropped from 47.7 Mbps down to 39.5 Mbps when increasing the number of clients from 2 to 18, meaning that scheduling 2 clients in parallel had a 20.8% improvement on 18 clients uploading their videos uncoordinated to 1 AP. This is explained by the increasing amount of collisions caused by a high network density. For sequential scheduling on the

TABLE I: The different types of smartphones used during the shielded environment experiments, listed together with their specifications.

Type	Specifications
LG Nexus 5	802.11 a/b/g/n/ac (dual band, 1x1:1 configuration), Qualcomm MSM8974 Snapdragon 800 chipset, 2.26 GHz quad-core Krait 400 (ARMv7 Processor rev0 v7l), 2 GB RAM, Android 5.0.1 Video: 18.4 MB, 1080p, H.246, AAC
Sony XPERIA E3	802.11 b/g/n (1x1:1 configuration), Qualcomm MSM8926-2 Snapdragon 400 chipset, 1.2 GHz quad-core CPU (ARMv7 Processor rev3 v7l), 1 GB RAM, Android 4.4.4 Video: 22.8 MB, 1080p, H.246, AAC
Motorola Moto G (2nd Gen., LTE)	802.11 b/g/n (1x1:1 configuration), Qualcomm MSM8226 Snapdragon 400 chipset, 1.2 GHz quad-core CPU (ARMv7 Processor rev3 v7l), 1 GB RAM, Android 5.0.2 Video: 10.8 MB, 720p, H.246, AAC
Motorola Moto E (2nd Gen., LTE)	802.11 b/g/n (1x1:1 configuration), Qualcomm MSM8916 Snapdragon 410 chipset, 1.2 GHz quad-core CPU (ARMv7 Processor rev0 v7l), 1 GB RAM, Android 5.0.2 Video: 11.2 MB, 720p, H.246, AAC

other hand, the achieved throughput showed no significant deterioration for an increasing number of clients. Sequential scheduling will assure that each upload is scheduled to transmit its video without any contention for the wireless medium and thus it will not suffer any data collisions.

When comparing the two scheduling approaches, one can observe that for 2 clients, parallel scheduling achieved a 4% gain in throughput over sequential scheduling, whereas for 6 clients a 5% loss was observed. For more clients, the loss in throughput increased even further. Similar remarks can be made when looking at Figure 4, which shows the total time required to upload all offered content by all clients for both scheduling approaches. For 2 clients parallel scheduling was 7% faster than sequential, for 6 clients the difference was negligible, and for all 18 clients parallel uploads were 20% slower than sequential ones.

From these results, it seems that sequential scheduling is the best performing technique when considering more clients. Parallel scheduling however has shown a slight advantage when uploading only two videos in parallel. It could be that this advantage is even larger for up to 5 clients, but we do not have any results to confirm this. Letting many clients upload their video uncoordinated regardless of the current network density was observed to end in throughput loss because of the increasing number of collisions. This loss will certainly increase when considering events with a minimum of several thousands of clients. These results make clear that in future work scaling to thousands of clients should consider intelligent content scheduling and particularly they show an optimal throughput when scheduling only two uploads in parallel per AP and channel.

2) *Multiple APs analysis*: The throughput and total upload time are also compared when using 3 APs to using only

¹iMinds technical testing: iLab.t, <http://ilabt.iminds.be>

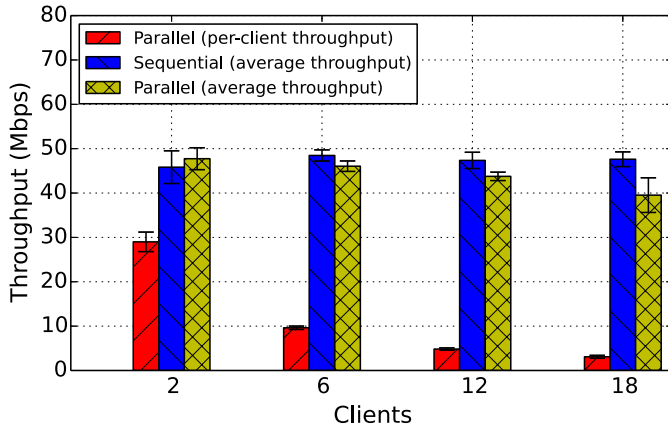


Fig. 3: Throughput comparison when applying parallel and sequential scheduling.

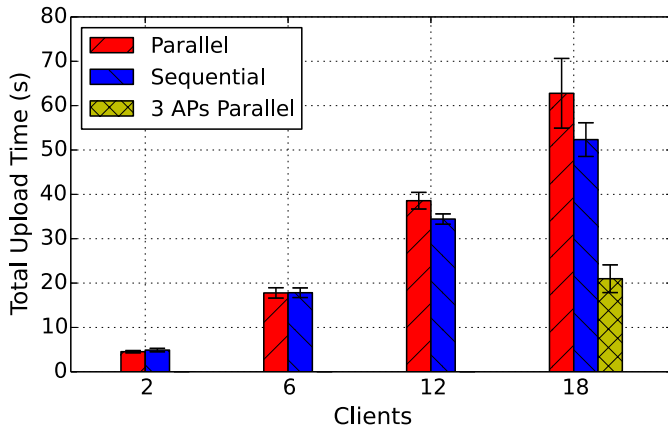


Fig. 4: Total upload time comparison for varying numbers of clients in the parallel and sequential scheduling experiments.

AP 1, as shown in Figure 2. The 18 mobile devices were uniformly distributed among the 3 APs, connecting 6 devices to a single AP. Only parallel scheduling was considered in this experiment as it is actually an extension of sequential scheduling over more APs, allowing more than one client upload its video per channel.

The results are shown in Table II. As expected, the results improved the average throughput of uploading 18 videos to a single AP by a factor of three. Similar results are shown in Figure 4 where the total upload time equalled 21 seconds when distributed over three APs, while using only a single AP resulted in a total upload time of 62.8 seconds. The results are similar to uploading 6 videos in parallel to a single AP (finished after 17.7 seconds) which seems intuitive as now we also uploaded only 6 videos to each AP. The minor difference in performance can be explained by the fact that the back-end now had to support 18 video uploads instead of only 6. This shows the need to further optimize the DR.

3) *IEEE802.11 protocol analysis*: To compare the performance of IEEE802.11n in the 2.4 GHz band and

TABLE II: The different throughput values and standard deviations (SDs) for parallel scheduling when using 1 and 3 APs.

Set-up	Per-client throughput	Average throughput
2 clients, 1 AP	29 Mbps (SD 2.2)	47.7 Mbps (SD 2.5)
6 clients, 1 AP	9.6 Mbps (SD 0.4)	46.1 Mbps (SD 1.2)
12 clients, 1 AP	4.8 Mbps (SD 0.3)	43.7 Mbps (SD 0.9)
18 clients, 1 AP	3.1 Mbps (SD 0.3)	39.5 Mbps (SD 3.9)
18 clients, 3 APs	10.5 Mbps (SD 1.3)	116.9 Mbps (SD 13.3)

IEEE802.11ac in the 5 GHz band, only the Nexus 5 smartphones that support both protocols were used. For both protocols, a channel width of 20 MHz was considered. The number of clients was varied from three to nine smartphones and client uploads were scheduled in parallel. The network was set up using only AP 1, shown in Figure 2.

The results are shown in Figure 5. As expected, IEEE802.11ac in the 5 GHz band performed better than IEEE802.11n in the 2.4 GHz band. The per-client IEEE802.11ac throughput was on average 21.9% higher than the IEEE802.11n per-client throughput. The average throughput was 18% higher with a maximum of 69.5 Mbps for IEEE802.11ac (six clients) and 57.8 Mbps for IEEE802.11n (three clients). As the theoretical data rate of IEEE802.11n and IEEE802.11ac with one spatial stream and 20 MHz bandwidth was 64 Mbps and 78 Mbps respectively, the throughput results and the IEEE802.11ac improvement over IEEE802.11n seems acceptable.

When comparing the IEEE802.11n results with six simultaneous uploads from this experiment to the results in Figure 3, it is clear that the per-client and average throughput of the experiment discussed in this section was higher, increasing 24% and 18.4%, respectively. As can be derived from the technical specifications in Table I, the Nexus devices have the best technical specifications of all used smartphones. In the previous experiment different types of devices uploaded videos which decreases the average throughput.

4) *Client type analysis*: To observe the differences between the clients, the results of the experiment in Figure 3 are revised. The sequential scheduling results were used because in that case there is only 1 client uploading data to the platform and this allows us to observe the individual performance of a particular client type. The results of the four different experiments were averaged over the (combined) 40 iterations and shown in Figure 6.

The four types of smartphones and their technical specifications are listed in Table I. While three out of four clients only support IEEE802.11b/g/n, the LG Nexus 5 smartphone also supports IEEE802.11ac. With a more advanced chipset, a better CPU and more internal RAM memory, the Nexus 5 device has the best technical specifications. The differences among the other three smartphones are minor.

The results are shown in Figure 6. As expected, the Nexus 5 client clearly performed better than the other device types.

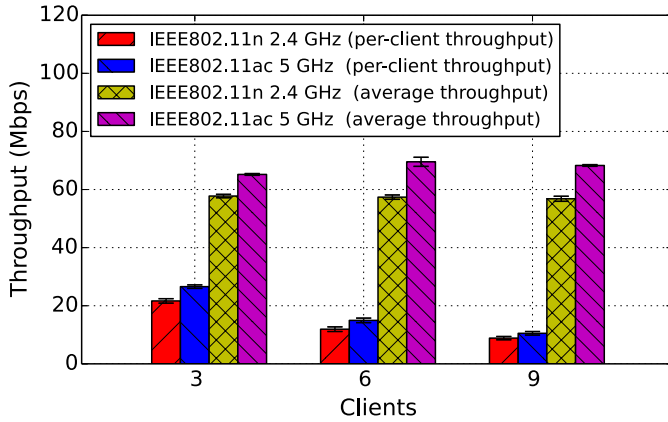


Fig. 5: Performance comparison between the 2.4 GHz (IEEE802.11n) and 5 GHz (IEEE802.11ac) band while applying parallel scheduling.

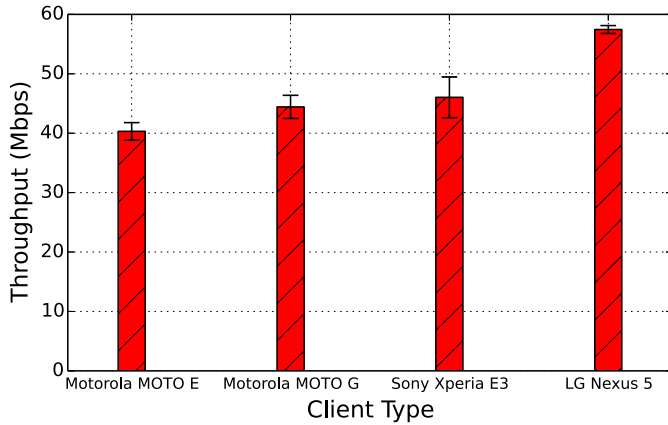


Fig. 6: Throughput comparison of each smartphone type during sequential scheduling experiments.

Supporting a throughput of 57.8 Mbps, it performed 23.5% better than the Sony XPERIA E3 which had a throughput of 46.8 Mbps. The Nexus device had a 34% throughput improvement on the Moto G and 40% on the Moto E, respectively uploading at 43 Mbps and 41.3 Mbps. The better performance of the Sony XPERIA E3 device compared to the performance of the Motorola devices is probably due to the more recent chipset compared to that of the Moto G and while the Moto E chipset was released in the same year, it lacks additional CPU memory caches which could influence performance.

C. Discussion

The experiments in Section IV-B1 show clearly that letting clients upload their videos simultaneously and uncoordinated decreases performance considerably and intelligent content scheduling is of great importance. Therefore the results helped us finding a possible content scheduling technique to scale up to high-density events with thousands of users. The most optimal strategy was to simultaneously upload anywhere between

2 and 5 videos in parallel per AP and channel, but we do not have the results to determine the exact optimum. To show how the developed platform and the optimal scheduling technique would scale for massive events, we will calculate the number of seconds of video a client can upload when varying the number of clients up to 5000 during an interval of 15 minutes.

The total amount of video seconds that can be uploaded on a particular band is calculated in following way:

$$V_{band} = I \cdot \frac{T_{band} \cdot C_{band}}{S_{video} \cdot 8} \quad (1)$$

where I is the length of the interval in which we want to calculate the total number of seconds of video that can be uploaded, S_{video} the average size (in bytes) of one second of video content, T_{band} the throughput for the particular frequency band and C_{band} the number of available channels in the frequency band. So, V_{band} represents the amount of seconds of video that can be uploaded in a particular frequency band during an interval I . For calculating the number of seconds of content one client can upload, we have:

$$L_{client} = \frac{V_{2.4GHz} + V_{5GHz}}{N} \quad (2)$$

where N is the number of users attending the event, which we assume are uniformly distributed among the available channels. Thus, L_{client} is the total number of seconds of video a client can upload during an interval of I seconds while attending an event with N users.

As in most COTS hardware IEEE802.11n and IEEE802.11ac is supported (or will be in the future), we assume the use of both IEEE802.11n in 2.4 GHz frequency band and IEEE802.11ac in the 5 GHz band and a total of 11 available channels with 3 non-overlapping channels in 2.4 GHz and 8 in 5 GHz, resulting in $C_{2.4GHz} = 3$ and $C_{5GHz} = 8$. We do not consider any background traffic as the QoS policy of our system can assign background traffic to a limited bandwidth pipe resulting in a limited and predictable throughput loss. The average throughput $T_{2.4GHz}$ when uploading two videos in parallel in the 2.4 GHz frequency band is 47.7 Mbps. From the comparison between IEEE802.11n (2.4 GHz band) and IEEE802.11ac (5 GHz band) in Section IV-B3 we know that the IEEE802.11ac on average has a 21.9% improvement on IEEE802.11n, resulting in a per-client throughput T_{5GHz} of 58.1 Mbps. For S_{video} , we consider 1.1 MB (720p), 2 MB (1080p) and 1.7 MB (both 720p and 1080p), based on the video sizes used in our experiments. The interval length I is 900 seconds and the number of clients N varies from 0 to 5000 clients.

Figure 7 shows the results. At events up to 1000 clients the number of seconds of uploaded video per client decreases rapidly. From 1000 clients up to 5000 the diminution is more steady. When there are 1000 clients in the concert hall, each user can upload 62.1 seconds of video (which is 6.9% of the 15 minutes interval) when everyone uploads in 720p, 34.1 seconds (3.8%) for 1080p videos and 40.2 seconds (4.5%) when there are both 720p and 1080p videos being uploaded. When there are 5000 clients, a user can upload 12.4 seconds

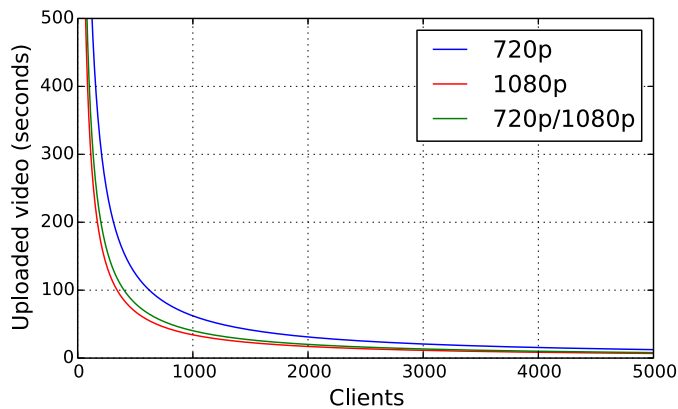


Fig. 7: The number of seconds of recorded video a user could upload when scaling up to events of 5000 clients.

of content (1.4%) for 720p videos, 6.8 seconds (0.8%) for 1080p and 8 seconds (0.9%) for both 720p and 1080p. As the platform aims at videos of a maximal length of 10 seconds, this means that a user can upload approximately only one video every 15 minutes when there are 5000 people attending the event. If higher performance is needed, several techniques to deal with the challenging characteristics of Wi-Fi Densenets can be applied. For example, (i) more efficient scheduling, (ii) improved PHY and MAC protocols that can better cope with the high density, (iii) channel reuse if the venue allows it and (iv) directed antennae. These techniques should allow users to stream more videos to the back-end of the system.

V. CONCLUSION

In this paper, the architecture of a large-scale upstream-intensive user participation platform is presented. Results of small-scale shielded lab experiments of the developed prototype are shown and the effect of content scheduling on scalability is discussed. In general, the results show the importance of content scheduling in Wi-Fi DenseNets as intelligent scheduling improved throughput over 20% compared to uncoordinated uploading in a dense network. While sequential scheduling has a benefit over parallel scheduling because of the increasing number of collisions when letting many clients upload uncoordinated, our findings show that the most optimal strategy is to upload anywhere between 2 and 5 videos simultaneously per AP and channel, but currently we do not have the results to determine the exact optimum. Based on those findings, we calculated the theoretical scalability of the platform and in its current form 5000 users would be capable of each uploading one 10 second video per 15 minutes. This performance could be increased by applying more efficient scheduling or new techniques such as improved PHY and MAC protocols to deal with the high density characteristics. Therefore, we showed the importance of intelligent scheduling to efficiently manage the wireless spectrum in highly dense Wi-Fi networks.

For future work, the throughput of the platform will be

tested on a larger scale. Both sequential scheduling and parallel scheduling with only two clients currently performed best and should be evaluated more thoroughly. Concerning the architecture itself, we consider more detailed monitoring and throughput estimation in the WB component in order to allow the UGCC to take more informed and accurate scheduling decisions.

ACKNOWLEDGMENT

The research presented in this article relates to SELVIE (<http://www.iminds.be/en/projects/2014/11/28/selvie>), a MiX-ICON project facilitated by iMinds Media and funded by IWT. Steven Bohez is funded by Ph.D. grant of the Agency for Innovation by Science and Technology in Flanders (IWT). The authors would also like to thank Lander Van Herzeele, Niels Van Kets, Glenn De Haes, Philippe Soutaer, Matthias De Geyter and Sam Decrock for their valuable help in setting up and running the experiments.

REFERENCES

- [1] K. Shin, I. Park, J. Hong, D. Har, and D.-H. Cho, "Per-node throughput enhancement in Wi-Fi densenets," *Communications Magazine, IEEE*, vol. 53, no. 1, pp. 118–125, 2015.
- [2] D.-J. Deng, K.-C. Chen, and R.-S. Cheng, "IEEE 802.11 ax: Next generation wireless local area networks," in *Heterogeneous Networking for Quality, Reliability, Security and Robustness (QShine), 2014 10th International Conference on*, pp. 77–82, IEEE, 2014.
- [3] M. Park, "IEEE 802.11 ac: Dynamic bandwidth channel access," in *Communications (ICC), 2011 IEEE International Conference on*, pp. 1–5, IEEE, 2011.
- [4] S. Valentin, T. Freitag, and H. Karl, "Integrating multiuser dynamic OFDMA into IEEE 802.11 WLANs-LLC/MAC extensions and system performance," in *Communications, 2008. ICC'08. IEEE International Conference on*, pp. 3328–3334, IEEE, 2008.
- [5] N. Rao, Q. Wu, S. Ding, S. Carter, W. Wing, A. Banerjee, D. Ghosal, and B. Mukherjee, "Control plane for advance bandwidth scheduling in ultra high-speed networks," in *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, pp. 1–5, April 2006.
- [6] Y. Lin and Q. Wu, "Complexity analysis and algorithm design for advance bandwidth scheduling in dedicated networks," *Networking, IEEE/ACM Transactions on*, vol. 21, pp. 14–27, Feb 2013.
- [7] S. Borst, "User-level performance of channel-aware scheduling algorithms in wireless data networks," *IEEE/ACM Trans. Netw.*, vol. 13, pp. 636–647, June 2005.
- [8] S. Ganguly, M. Chatterjee, and R. Izmailov, "Non-real-time content scheduling algorithms for wireless data networks," *Computers, IEEE Transactions on*, vol. 55, pp. 893–905, July 2006.
- [9] W. Culhane, K. R. Jayaram, and P. Eugster, "Fast, expressive top-k matching," in *Proceedings of the 15th International Middleware Conference, Middleware '14*, (New York, NY, USA), pp. 73–84, ACM, 2014.