Playout Delay of TV Signals: Measurement System Design, Validation and Results

Wouter J. Kooij
TNO
Delft, The Netherlands
wouter.kooij@gmail.com

Hans M. Stokking
TNO
Delft, The Netherlands
hans.stokking@tno.nl

Ray van Brandenburg
TNO
Delft, The Netherlands
ray.vanbrandenburg@tno.nl

Pieter-Tjerk de Boer
University of Twente
Enschede,
The Netherlands
p.t.deboer@utwente.nl

ABSTRACT
Due to new interactive TV services, synchronizing the playout of content on different TVs is becoming important. To synchronize, knowledge of delay differences is needed. In this study, a measurement system is developed to gain insight into the magnitude of delay differences of different TV setups in an automated fashion. This paper shows the measurement system, which is validated for precision and accuracy. Preliminary measurements results show that regular TV broadcasts differ up to 6 seconds in playout moment and that web based TV broadcasts can introduce more than a minute delay. Furthermore, we measured a broadcasting before encoding and modulation, which resulted in a time about 4 second before the fastest receiver. On a side note, while developing the measurement system we found out that GPS timing on consumer Android devices was inaccurate, with fluctuations of up to 1 second.

Author Keywords
TV; broadcast; media synchronization; delay; Android

ACM Classification Keywords
H.5.1 [**Multimedia Information Systems**]: Video, Broadcasting

BACKGROUND
The following scenario might sound familiar: imagine you are watching an exciting soccer match, such as a world cup game. The team you are supporting is in ball possession and is setting up an attack when you suddenly hear loud cheering noises coming from your neighbors. A little later you see where this cheering came from: a goal was scored. This is an example often given to illustrate a playout difference. A playout difference is the difference in delay between the displaying of a certain piece of content on different TV systems, possibly using different techniques or obtaining content from different content providers. These playout differences have been shown to be noticeable or annoying, even for differences as small as just 1 second [1].

TVX '14, June 25 – 27, 2014, Newcastle Upon Tyne, United Kingdom.

http://dx.doi.org/10.1145/2602299.2602310

A playout difference is not just annoying when watching football, there are other applications where playout differences cause problems. Examples of this are applications that involve real time content or user interaction through television. Sources such as [2] and [3] report or expect an increase in the use of these kind of advanced television services.

Knowledge of playout differences is essential for the design of such services. An example where this is essential is a second screen application for live TV quizzes that require live viewer interaction. Knowing the playout delays make it possible for service developers to account for this in their design. It will also be interesting to know if there is delay dispersion for geographically different locations or different moments in time.

Thinking one step ahead, knowing or being able to quickly measure these delays might even be a step in the direction of a possible synchronization of TV broadcast chains. In this paper, we report on a system we have developed that can measure playout differences in an automated fashion using smartphones as measurement devices, and using a technique called audio fingerprinting. We have validated the system's accuracy, and report on (preliminary) measurement results.

RELATED WORK
For media synchronization, a good overview is given in [4], explaining and discussing intra-media synchronization, inter-media synchronization (i.e. lip-sync) and inter-destination media synchronization. Our paper focusses on playout differences, which are mostly relevant for both inter-media synchronization and inter-destination media synchronization (IDMS), topics discussed extensively in [5]. More recently, various synchronization use cases are presented in [6], showing synchronization requirements ranging from very high (<10ms) to low (< 2 sec) for various IDMS use cases, while [7] recommends synchronization of approximately 1 second for a seamless shared experience. The results presented in our paper will show that current playout differences can be much larger than this, and thus synchronization solutions will be required for many interactive TV services.

The accuracy of our measurement system depends largely on the accuracy of the clock synchronization. [8] contains a number of recommendations on how to achieve such accuracy, among which the recommendation to use a single time server and the recommendation to first achieve a good so-called difference clock, i.e. a good internal clock.

Furthermore, an overview of applications that fit into the social TV paradigm, including examples where synchronization is needed, can be found in [9]. More examples of such applications involving real-time interaction through a TV can be found in [10] and [11].

As far as the authors know, no large scale measurement of TV playout delays in an automated way has been performed before. Similar studies were performed that measured TV playout differences on small scales or in a non-automated fashion, such as [1] and [12]. These studies have shown that playout differences of up to a few seconds are possible between different TV broadcasts within the same region. At that time, only regular broadcasts were studied, i.e. web players and tablet-based television broadcasts were not included. There are also other systems for managing delay, such as presented in [13]. But this work is not applicable here, as it requires access to both the ingest and playout point

TV BROADCASTING DELAY

Different TV broadcasting techniques have different distribution channels and techniques. This imposes a difference in delays between the broadcasting moment and the delivery of content via different broadcasts. It can often be seen in practice that analog television is faster than digital television. This is because digital television has more delay introducing steps such as encoding or transcoding.

Furthermore, the transmission technique used for broadcasting influences the amount of delay introduced in a TV broadcast. For example, television from satellite (DVB-S) imposes a delay introduction of at least some hundreds of milliseconds of delay, just from the delay introduced by the signal propagation to and from the satellite at the speed of light. Other broadcasting techniques such as cable television (DVB-C) or terrestrial broadcasts (DVB-T) have a much smaller lower boundary of the minimum transmission time in terms of what is physically possible. And, depending on the format of the original signal, HD and SD signals will differ as one has to be transcoded to the other.

Apart from this, TV broadcasting with Internet Protocol TeleVision (IPTV) has other Quality of Service (QoS) or Quality of Experience (QoE) related techniques that influence the delay of the broadcast. Examples of these are error correction or retransmission schemes used.

To get an idea where these delays in a television network arise, Figure 1 presents a global overview of delay introducing factors in a general TV broadcasting chain. This figure includes the chain structure of both the scenario depicting live broadcasts and the scenario where pre-recorded content is broadcasted. The aforementioned elements encoding and transmission are theoretically the largest causes of delay. In practice there might be additional or different provider-specific steps involved. The actual amount of delay introduced in a TV broadcast can depend on many other factors such as the usage of specific hard- and software and the associated configuration that comes with it.

This paper focusses on the delay introduced by the distributor, the transport and the user side, i.e. the boxed region in Figure 1.

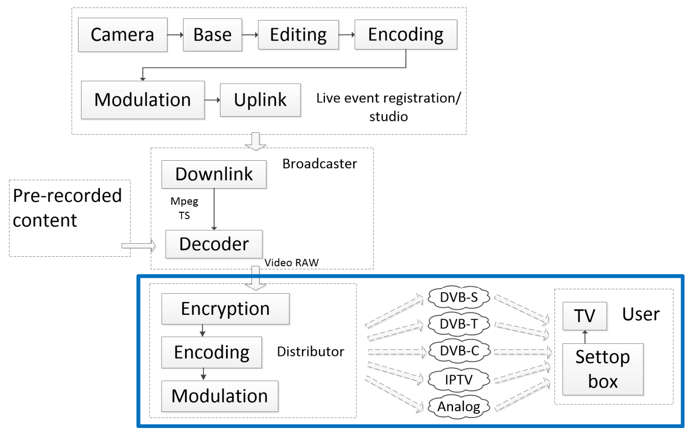

Figure 1: Chain structure of a TV content delivery chain

MEASUREMENT SYSTEM

To measure the playout delay of a TV broadcast, we use a technique called audio fingerprinting. Audio fingerprinting techniques make use of so-called fingerprints that digitally summarize the content of audio. These fingerprints are then matched against a previously calculated (or in case of live TV fingerprinting a real-time calculated) set of reference fingerprints in a fingerprint database. The general principle is more or less the same as for example the popular app Shazam, of which the workings are globally described in [14]. Our system makes use of the freely available live TV fingerprinting platform "Entourage" provided by the company Gracenote [15]. This TV fingerprinting platform provides real-time recording and fingerprinting of live TV channels done by Gracenote. This acts as the reference that we use for comparing a locally (i.e. a normal TV) recorded fingerprint (and the corresponding starting moment in time) with. The platform of choice is Android, which allows for performing easy measurements, not only by ourselves but also by people that are not familiar with our measurement system.

More specifically, an Android device records audio from a TV through its internal microphone, calculates an audio fingerprint and compares this with the mentioned reference. The playout difference is then calculated as the difference in time between the recording of both audio fingerprints.

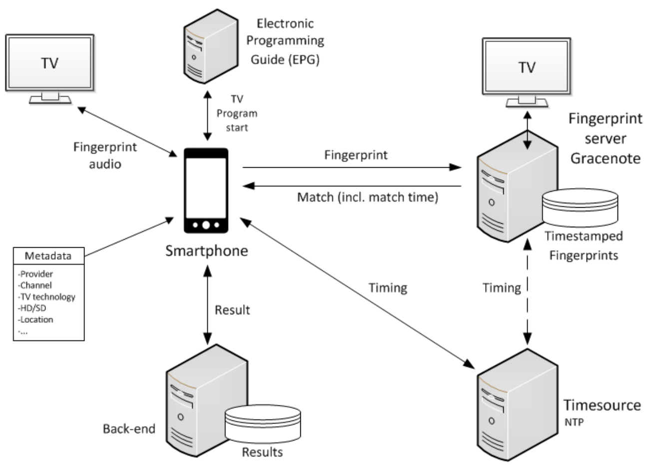

Figure 2: Architecture of the measurement system

Furthermore, our measurement application allows for input of metadata such as the broadcaster name, channel name, TV technology, TV quality (HD/SD) and location. A measurement of the playout difference combined with this manually entered metadata is submitted to a back-end database server on which the measurements are stored.

To accurately compare the moment the fingerprint is recorded on the smartphone with the moment the fingerprint is recorded on the reference, both sources need to have an accurate notion of time. To accomplish this, the smartphone obtains a timestamp by querying an NTP server.

To make sure the time obtained with NTP is accurate and consistent for individual measurements, we use a single stratum 1 NTP server managed by ourselves. To reduce the influence of congestion spikes, the smartphone performs multiple timestamp requests to this server and only uses the response with the lowest round-trip-time (RTT). This is done according to the recommendations from [8]. If the delay of the RTT is lower, the "quality" of the timestamp is better. The server side fingerprinting mechanism that assigns fingerprints a timestamp is assumed to be sufficiently accurate. This is validated as well, as can be seen from the accuracy measurements later in this paper.

An architectural overview of the system can be found in Figure 2. The smartphone application acts as the center of communication between the mentioned components. The smartphone application records sound from a TV, queries the NTP server for a timestamp and starts creating a fingerprint right at this moment. Next, the created fingerprint is submitted to the reference server of Gracenote which compares the fingerprint with its database of live, continuously created fingerprints of its own TV source. If a match is found, the time offset (relative to the start of matching TV program) is returned to the smartphone.

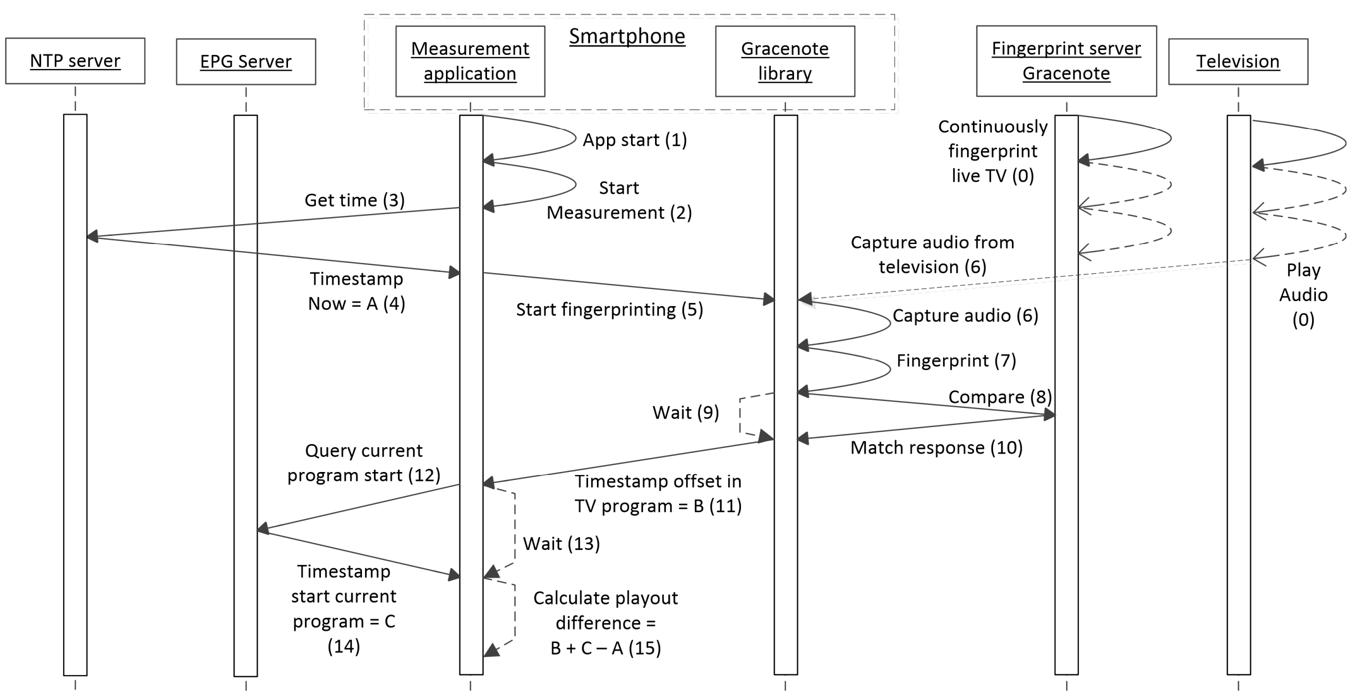

Figure 3: Measuring playout difference

Next, the smartphone queries the EPG server to determine the absolute moment in time the fingerprint was recorded on the reference. The moment of local fingerprint recording and the moment of reference fingerprint recording are compared against each other to obtain the playout difference. Finally, combined with manually entered metadata, the playout difference measurement is submitted to a back-end database, storing the results. For a better understanding and to get an action sequence overview, shows a sequence diagram of this process.

MEASUREMENT SYSTEM PERFORMANCE

Factors in performance

This section shows the test results of performance (for precision and accuracy) of the measurement system as a whole and for some of the individual parts of the measurement system. Factors influencing the performance of the measurement system include the internal clock accuracy of Android, the server-side fingerprint offset matching timestamp accuracy, EPG program start timestamp accuracy, the reference time source (NTP server) accuracy and measurement outliers. Of these factors, the Gracenote servers provide fingerprint offset matching and EPG program start times. The fingerprint matching seems to give us some outliers, see the section on outliers. The EPG program start times seem quite stable, i.e. we did not notice any large jumps in playout difference when measuring delays during different programs.

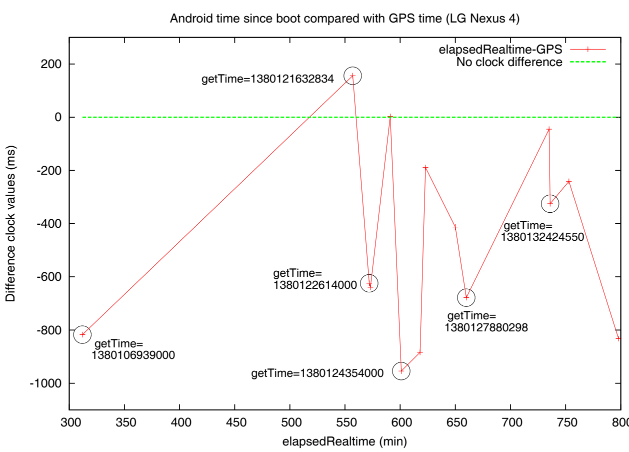

Figure 4: Inconsistency of GPS timing on an LG Nexus 4

No time related accuracy measurements are possible without an accurate time source. Using the timestamp obtained from the GPS receiver on Android turned out to be inaccurate, or at least it was on the specific test device (an LG Nexus 4 smartphone). This was tested by comparing the internal clock with the GPS time as obtained through the Android API. The time obtained this way on the specific test device through the Android API can be wrong by up to a whole second. More details on this test can be found in [16]. The accuracy of GPS timestamp information seems to be device-, firmware- and hardware-specific. Because of this, direct GPS timestamp information on Android is not an option for obtaining consistent and precise timing information.

In Figure 4, the inaccuracy and inconsistency of the GPS timing on the test device is shown. The GPS timing clearly deviates from the line that represents a situation where the GPS timing would be in sync with the internal clock of Android (elapsedRealtime). The latter is shown to be monotonic with a small linear drift in the next section. Even when this drift is compensated for, we must conclude that the GPS timing information on Android is not accurate enough for our purposes.

Since GPS timing on Android turns out to be too inaccurate, we chose to use NTP (Network Time Protocol). We set up our own Stratum-1 NTP server using a GPS receiver as its reference. For more detail on this, see [16].

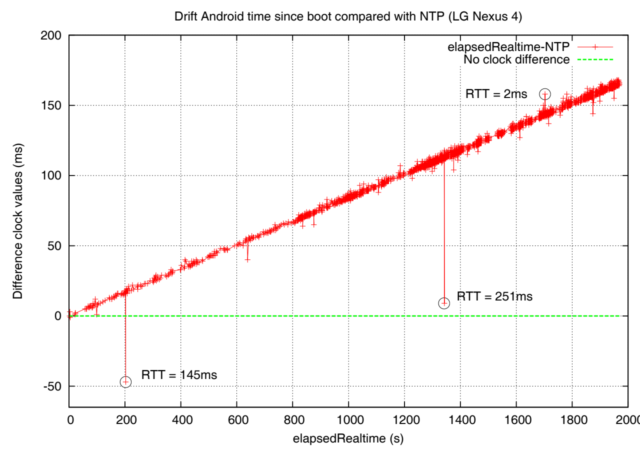

Figure 5: Accuracy internal clock Android (LG Nexus 4)

Internal clock Android

The Android internal clock (relative to the start of the device, in the API called "elapsedRealtime") is used for calculating several time deltas and for keeping track which operation was started when. More specifically, it is used to correlate the NTP time with the moment of a fingerprint creation, in order to obtain the exact timestamp of the fingerprint. This is especially important, since the measurement system performs multiple NTP requests and only uses the one with the lowest round-trip-time. For the app to be accurate the internal clock must be sufficiently accurate.

In order to verify the accuracy of the internal clock, we created a wireless network with a NTP server, a router and an Android smartphone. No other devices were connected to this network. This allows the smartphone to communicate with the NTP server directly without interference of other network traffic. This way, the smartphone can obtain timestamps from the NTP server with maximum accuracy. To test the accuracy of the internal clock of Android on the test device, a series of timestamps requests to the NTP server were performed over a period of about 30 minutes. The timestamps obtained this way were then compared with the time of the internal clock by calculating the time difference elapsed since a fixed starting point for both timestamps. The differences between both values were then compared with each other and indicate the accuracy of the internal clock of Android. The results of this test are displayed in Figure 5. The test device used here, and for the remainder of this paper is an LG Nexus 4, using a native version of Android 4.2.

These test results show that the internal clock in Android is monotonic with a small linear drift. The slope of the line that runs through the middle of all points represents the amount of linear drift of the clock. This line has a slope of 84.23µs/s. For the purpose of the measurement system, this is accurate enough. We require the clock to remain accurate during a single measurement. As a measurement last about a couple of minutes, the inaccuracy remains in the order of magnitude of 10s of milliseconds. As the expected measurement results are in the order of seconds of difference, this is acceptable for the internal clock.

Outliers

While using the app for measuring, correct measurements were usually obtained, but there were also moments where the measurement system would provide a value that was clearly not correctly representing the playout difference at that moment. These incorrect measurements, which are called outliers from here on, occurred on more or less random moments. These outliers are usually a fixed value roughly 3 seconds off from the real value (as determined by the accuracy test later in this paper). What might be going on here is that a sub-fingerprint next to the matching fingerprint is invalidly marked as match (a false positive), although this is speculation on our part. The mechanism used for dealing with these outliers operates by performing multiple measurements and only keeping the median value of the measured values. For this purpose, we have performed 8 measurements each time, and kept only the 5th lowest, as we saw more outliers on the low side. This is a very severe outlier removal mechanism, but ensures proper measurement results in practice. If the distribution of outliers would follow a normal distribution, the chance of an outlier would be as good as zero. This is not the case however, and there is still a slight chance that the system might provide an occasional outlier, despite the outlier removal mechanism.

Precision

Next, the measurement system was tested for overall precision, i.e. closeness of agreement to which subsequent measurement show the same value under the same circumstances. The test setup is shown in Figure 6. This test uses two smartphones with the app installed to calculate the playout difference at roughly the same moment. The overall precision error is then calculated as the difference in playout differences calculated by both smartphones.

The results of the precision test can be found in Figure 7. This obviously contains outliers. We watched the TV show during the measurements, and no actual jumps in the playout were seen. So, we applied our previously described outlier removal, with the results displayed in Figure 8. After the outlier removal, the precision error is found to be at most 200ms in this test. This is adequate for our measurement application, as we are expecting to measure playout difference in the order of magnitude of seconds.

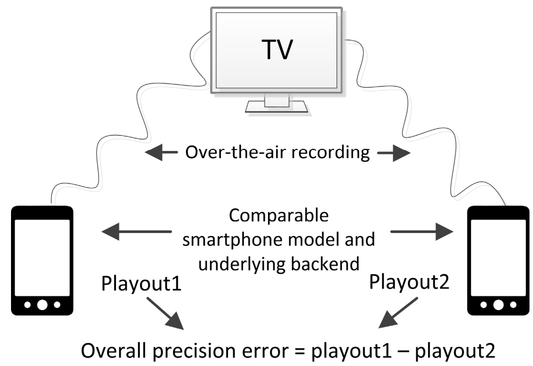

Figure 6: Precision measurement setup

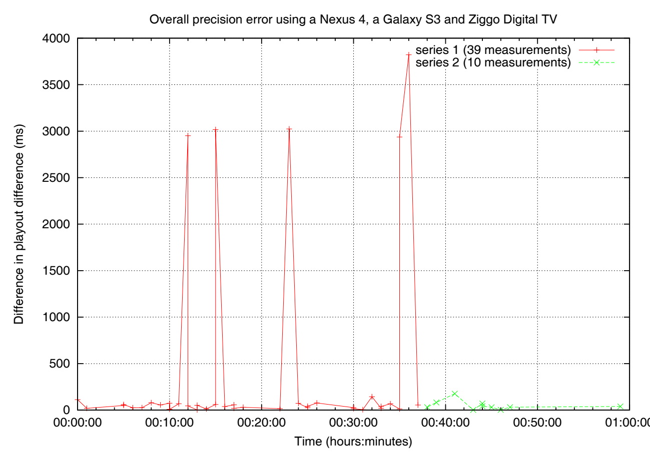

Figure 7: Overall precision error of the system.

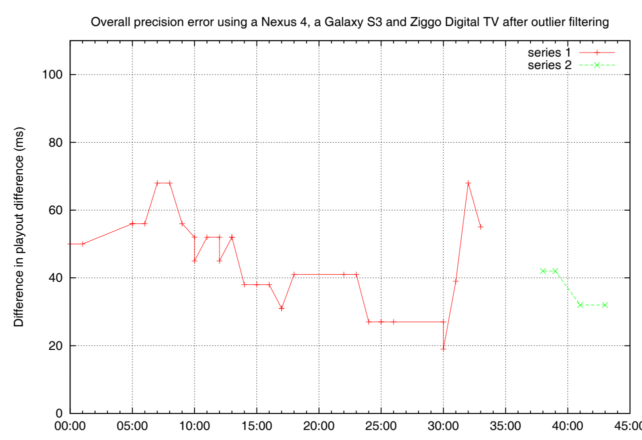

Figure 8: Precision error, after outlier removal

Accuracy

Figure 9 shows the test setup used to test the system for overall accuracy. This setup is a bit more complicated than the precision test setup. It includes two TVs, two smartphones and an audio interface. The goal of this test is to measure the degree to which measured playout differences correspond with the actual playout differences. Without access to the actual TV reference, this test is not straightforward. The difference in playout times of the actual audio (as locally recorded from two TVs) is compared with the difference in playout differences obtained by two similar smartphones using the application. This way, the calculated accuracy error constitutes again the error introduced by both phones.

In this setup, the actual difference in audio between both TVs is measured by connecting a PC using the audio interface that has a direct connection to the 3,5mm headphone connections of the TVs. It records both audio signals as two mono signals combined into one stereo signal. This audio was then aligned manually by looking at a graph of the audio signals. In this way, we manually determined the playout difference between both TVs.

The test results for the accuracy test, after outlier removal as previously discussed, are presented in Figure 10. In this graph, one line represents the actual difference in audio (as is measured directly from the TV audio, Δ audio in Figure 9) and the other line represents the measurement difference between the two simultaneous smartphone measurements (Δ measurement in Figure 9). The actual difference (Δ audio) is measured for every tenth smartphone measurement, to deal with possible audio hick ups from the TV (caused by frame skipping) and to reduce the chance of a human reading error.

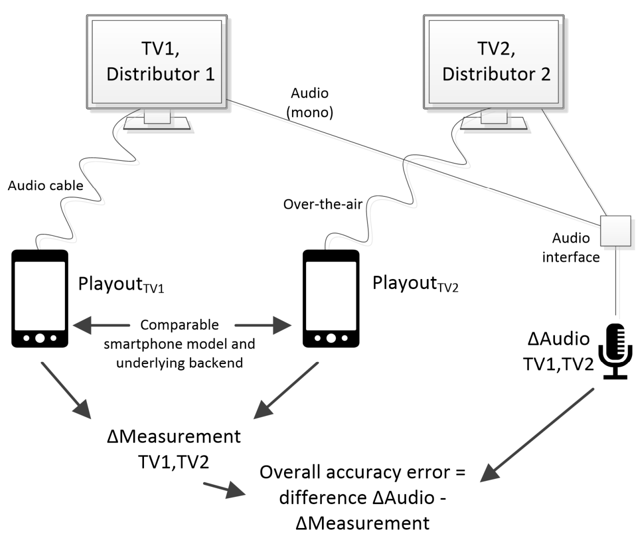

Figure 9: Accuracy measurement setup

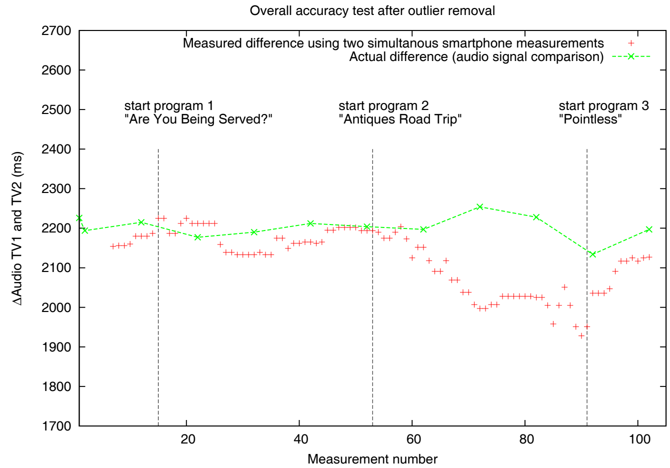

Figure 10: Accuracy error after outlier removal (using the median of a sliding window of 7 measurements)

The degree to which both values agree indicates the accuracy of the measurement system. Note that the distance between the smartphone and the TV is neglected as the goal is not to provide frame-accurate measurements. As can be seen from this figure, the maximum combined overall inaccuracy error for both phones is 250ms. Of course, on average the accuracy is much better compared to the maximum inaccuracy.

The program names indicating the start of different TV programs are also included in the picture. Recall that the system queries an EPG server to determine the start time of a TV program. Since a fingerprint matching results only delivers the offset in time of a match in a current TV program, using the EPG data allows for having an absolute time of the occurrence of the measured piece of audio. This in turn allows it to be compared with the absolute moment in time as measured using the NTP server representing the moment the piece of audio was fingerprinted locally.

Between around measurement number 60 and 90, the accuracy becomes less than in earlier measurements. This seems to be right in between the start of program 2 and program 3. A reason for this might be that the server side of the fingerprinting is experiencing clock drift at this moment. After the start of program 3, the measurements become more accurate again. Again, the accuracy seems adequate for the measurements we want to perform.

RESULTS

The application has been used to perform playout difference measurements mainly in the Netherlands and also in the UK. The results of these measurements, independent of geographical location or time can be found in Figure 11. Remember that the reference is the live fingerprinting TV server from Gracenote, which is the reference to which all measurements were compared with. Therefore the playout difference of the reference with itself is logically zero. This figure displays the average of the measured combination of broadcaster, subscription (technology) and quality (HD or SD) for both the channels BBC One and BBC Two combined. Experience shows that the playout delay for these channels is the same.

One thing that can be seen from these measurements is that analog delivery is faster than other delivery mechanisms s from the same broadcaster. The reason for this is probably in the encoding part, which makes digital TV slower than analog TV.

Furthermore, in general HD broadcasts are slower than their SD equivalents. This also seems logical, because HD broadcasts introduce more encoding delay (due to multi-pass encoding for example) and are therefore more likely to introduce delay in the broadcast.

Looking at the absolute values, it can be seen that playout differences of up to almost 5 seconds are possible in TV broadcasts in the Netherlands. International playout differences are larger when measurements from the UK are also included. These measurements are the fastest, and compared with the slowest measurement in the Netherlands (excluding internet streaming TV), playout differences can become almost 6 seconds.

Also, with the help of the BBC, a measurement was performed at the broadcasting chain prior to coding and multiplexing. This is indicated as "internal" in Figure 11. Comparing the playout difference between this internal measurement and the fastest measured average playout difference, we see that there is a difference of around 4 seconds. This value consists of the delay caused by encoding, modulation and also distribution to the fastest receiver (which is terrestrial and SD quality). Since it is not known which part of this value is broadcasting delay, we can only conclude that in this case the encoding and modulation at the BBC takes at most around 4 seconds.

Furthermore, the measurements clearly show that the playout delay in the United Kingdom is lower than in the Netherlands. This is not surprising as it is broadcasted from the UK itself, although analog delivery can be faster than the slower broadcasts in the UK (such as HD or Satellite).

Something else that can be seen from these measurements is that internet streams (KPN ITV Online, BBC iPlayer) are much slower than normal television. These streams can be around 20 seconds or even more than a minute slower than a regular TV broadcast. This is not very surprising and is something that could be expected since content for internet streams is normally prepared separately and is distributed separately as well. The underlying techniques and systems that deliver these streams can vary greatly, both in terms of architecture as well as in terms of delay. E.g. in case MPEG-DASH is used for delivering the internet streams, the content is segmented and encoded in several versions, and usually delivered using Content Delivery Networks.

A geographical delay dispersion analysis at a national level, where exactly the same broadcaster and setup combination (subscription type/quality) are directly compared with each other was not performed due to a lack of data. We did notice that there is definitely some playout differences between geographically distributed measurements with the same setup. But without sufficient geographically distributed measurements of the same broadcaster and setup combination, no clear conclusion can be drawn. Apart from this, the fact that different TVs or settop boxes will cause some variance in the measured delays will only make it harder to draw a conclusion on this.

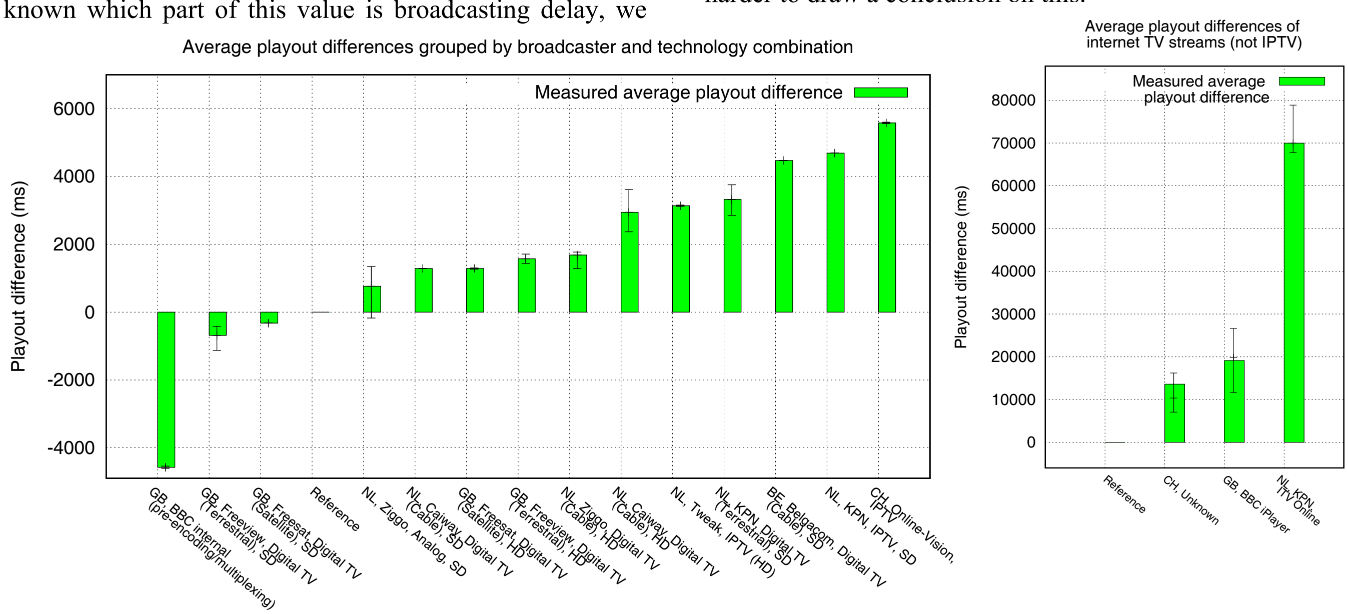

Figure 11: Playout differences by provider setup for regular TV (a, left) and for internet streams (b, right) The error bars indicate maximum and minimum measurements. The measurements that seem very accurate only contain one or two measurement results. The total number of measurements is 182.

CONCLUSION

A measurement system that is able to automatically measure playout differences of TV content compared to a reference is presented. The system is able to measure the TV playout accurately, although in a rare occasion the system may mistake an outlier as a valid measurement. The system is not so accurate that it allows for frame accurate synchronization, but it is adequate to give a good overview of the order of magnitude of playout differences of TV signals. This is something that has not been done before on a large scale, as far as the knowledge of the authors goes.

The measurement results show that analog delivery is delivered faster than non-analog delivery in general. Also, in general, HD broadcasts seem to be slower than SD broadcasts. Furthermore, internet streams of TV can be more than 1 minute slower than a “regular” TV broadcasting technique. Also, we measured a broadcasting before encoding and modulation, which resulted in a time about 4 second before the fastest receiver.

FUTURE WORK

To gain a better insight in the geographical distribution of specific TV setups (broadcaster and technology), more geographically distributed measurements are needed. Furthermore, as a means of verification, cross-checking the system with latency measurements on an SFN network is something that will be useful.

ACKNOWLEDGEMENTS

The work by TNO is partially funded by the European Commission’s 7th Framework Programme under grant agreement no. 318343 (STEER project).

REFERENCES

1. Rufael Mekuria, Pablo Cesar, and Dick Bulterman. 2012. Digital TV: the effect of delay when watching football. In Proceedings of the 10th European conference on Interactive tv and video (EuroITV '12). ACM, New York, NY, USA, 71-74.
2. Evelien D'heer, Cédric Courtois, and Steve Paulussen. 2012. Everyday life in (front of) the screen: the consumption of multiple screen technologies in the living room context. In Proceedings of the 10th European conference on Interactive tv and video (EuroITV '12). ACM, New York, NY, USA, 195-198.
3. Lochrie, M., & Coulton, P. 2012. Tweeting with the telly on!. In Consumer Communications and Networking Conference (CCNC), 2012 IEEE (pp. 729-731). IEEE.
4. Ud Din, Shahab, and Dick Bulterman. "Synchronization Techniques in Distributed Multimedia Presentation." MMEDIA 2012, The Fourth International Conferences on Advances in Multimedia. 2012.
5. Fernando Boronat, Jaime Lloret, Miguel García, Multimedia group and inter-stream synchronization techniques: A comparative study, Information Systems, Volume 34, Issue 1, March 2009, Pages 108-131, ISSN 0306-4379, Oxford, UK.
6. Montagud, M., Boronat, F., Stokking, H., & van Brandenburg, R. (2012). Inter-destination multimedia synchronization: schemes, use cases and standardization. Multimedia Systems, 18(6), 459-482.
7. Geerts, D., Vaishnavi, I., Mekuria, R., Van Deventer, O., & Cesar, P. (2011, May). Are we in sync?: synchronization requirements for watching online video together. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (pp. 311-314). ACM.
8. Julien Ridoux and Darryl Veitch. 2010. Principles of Robust Timing over the Internet. Queue 8, 4, Pages 30 (April 2010), 14 pages.
9. Pablo Cesar and Konstantinos Chorianopoulos. 2009. The Evolution of TV Systems, Content, and Users Toward Interactivity. Found. Trends Hum.-Comput. Interact. 2, 4 (April 2009), 373-95.
10. Pedro Almeida, Jorge Ferraz, Ana Pinho, and Diogo Costa. 2012. Engaging viewers through social TV games. In Proceedings of the 10th European conference on Interactive tv and video (EuroITV '12). ACM, New York, NY, USA, 175-184.
11. Goranka Zoric, Louise Barkhuus, Arvid Engström, and Elin Önnevall. 2013. Panoramic video: design challenges and implications for content interaction. In Proceedings of the 11th European conference on Interactive TV and video (EuroITV '13). ACM, New York, NY, USA, 153-162.
12. van Deventer, M.O., Stokking, H.M., Niamut, O.A., Walraven, F.A., Klos, V.B., Advanced interactive television services require content synchronization. In Proceedings of the 15th International Conference on Systems, Signals and Image Processing, pp.109-112, 2008.
13. Jansen, J., & Bulterman, D. C. (2013, February). User-centric video delay measurements. In Proceeding of the 23rd ACM Workshop on Network and Operating Systems Support for Digital Audio and Video (pp. 37-42). ACM.
14. Wang, A.L. An industrial-strength audio search algorithm, 2003.
15. Gracenote. https://www.gracenote.com/
16. Kooij, W. Playout delay of TV broadcasting, University of Twente, the Netherlands, Master Thesis, 2014.