Using Conversation Metrics for the Automated Orchestration of Video Conference View Modes: Two Comparative Studies

Matthias Traub

Using Conversation Metrics for the Automated Orchestration of Video Conference View Modes: Two Comparative Studies

Master's Thesis

at

Graz University of Technology

submitted by

Matthias Traub

Institute for Information Systems and Computer Media (IICM),
Graz University of Technology
A-8010 Graz, Austria

14th December 2016

© Copyright 2016 by Matthias Traub

Advisor:	Ao.Univ.-Prof. Dr. Keith Andrews
Co-Advisor:	DI (FH) Rene Kaiser

Anwendung von Konversationsmetriken zur automatisierten Orchestrierung von Video Konferenz Anzeigemodi: Zwei Vergleichsstudien

Diplomarbeit

an der

Technischen Universität Graz

vorgelegt von

Matthias Traub

Institut für Informationssysteme und Computer Medien (IICM),
Technische Universität Graz
A-8010 Graz

14. Dezember 2016

© Copyright 2016, Matthias Traub

Diese Arbeit ist in englischer Sprache verfasst.

Begutachter: Ao.Univ.-Prof. Dr. Keith Andrews
Mitbetreuender: Di (FH) Rene Kaiser

Abstract

In the last decade, video conferencing systems have become an essential part of modern communication. Initially predominantly a business application due to its high acquisition cost, video conferencing has made its way from the boardroom to the personal sector and even to hand-helds and mobiles. In addition to the basic combination of audio and video streams, there are many extra capabilities like on-screen drawing, file sharing, and facial recognition. Video conferencing enables real-time, synchronous communication independent of the participants' location. Although technology has improved, video conferencing systems are still not considered to be as good as face-to-face meetings and therefore constitute a separate communication situation. One of the major problems of video conferences is that each participant has a different perception of the conversational situation and communication.

The goal of this thesis is the evaluation of automated orchestration in the Vconect video conferencing system through two comparative studies. In the first study, two different view modes (tiled and full screen) were compared with regard to their impact on the communication and system quality. The study was designed as a repeated measures study with one independent measure being the view mode. A previous study showed that certain view modes are more suitable for particular scenarios. The goal of this study was to see whether this hypothesis holds true in a slow turn-taking scenario. The study was performed with 16 participants split into 4 groups of 4. The study showed no statistically significant preference for a particular view mode, but did reveal a tendency in preference towards tiled view mode, and also revealed other problems with the system.

The second comparative study investigated the impact of voice activity detection sensitivity (start delay). Three different degrees of sensitivity were compared within full screen view mode. The thresholds for the three start delays were chosen at 300, 600, and 900 ms according to insights from previous evaluations and simulations. The study was designed as a repeated measures study with one independent measure start delay. The study was performed with 40 participants divided into 10 groups of 4. The analysis of the subjective measures showed that the shortest start delay of 300 ms (highest sensitivity) was rated statistically significantly worse than longer start delays (lower sensitivity) in three aspects. However, overall preference showed only a tendency towards the two longer start delays (lower degrees of sensitivity).

Kurzfassung

In der heutigen Zeit bieten Videokonferenzsysteme eine günstige und wertvolle Alternative zu physischen Treffen. Diese Systeme reduzieren die meist hohen Zeit- und Reisekosten und bieten Unternehmen, Schülern und Studenten, sowie auch Privatpersonen die Möglichkeit eines synchronen Informationsaustausches. Videokonferenzen sind jedoch nicht zu vergleichen mit Face-to-Face Meetings und stellen somit eine eigenständige Kommunikationssituation dar. Hauptproblem hierbei ist die unterschiedliche Wahrnehmung der Gesprächssituation eines jeden Teilnehmers.

Ziel dieser Arbeit ist es, zwei ausschlaggebende Faktoren in dem Videokonferenzsystem Vconect in zwei Vergleichsstudien zu evaluieren. Die erste Vergleichsstudie behandelt die Hypothese, dass eine unterschiedliche Anzahl und Anordnung von Ansichten die Gesprächsqualität positiv oder negativ beeinflusst. Hierbei wurden zwei Darstellungsvarianten verglichen. In der ersten Darstellungsvariante (Fullscreen) ist immer nur eine Person sichtbar und diese wird automatisch vom System gewechselt. Die zweite Darstellungsvariante (Tiled) zeigt immer alle Gesprächsteilnehmer in einem Raster zur gleichen Zeit. In dieser Studie wurden 16 Teilnehmer in 4 Gruppen geteilt und testeten anhand eines Messwiederholungsdesigns abwechselnd beide Darstellungsvarianten. Die Auswertung dieser Vergleichsstudie ergab keinen signifikanten Unterschied zwischen den beiden Darstellungsvarianten, jedoch konnte ein Trend zur zweiten Darstellungsvariante (Tiled) festgestellt werden.

Die zweite Vergleichsstudie behandelt die Hypothese, dass eine unterschiedliche Sensitivität in der Spracherkennung einen Einfluss auf die Gesprächsqualität hat. Hierbei wurden drei Schwellwerte ermittelt die eine geringe, mittlere und hohe Sensitivitätsausprägung abbilden (Start Delay von 300, 600, und 900 ms). Die Darstellungsart wurde so gewählt, das die Teilnehmer den Unterschied im Verhalten des Systems bestmöglich feststellen konnten. In dieser Studie wurden 40 Teilnehmer in 10 Gruppen zu je 4 Personen geteilt und testeten ebenfalls anhand eines Messwiederholungsdesigns alle drei Ausprägungen. Die Auswertung der subjektiven Befragung lässt rückschließen, dass die Teilnehmer eine geringere Sensitivität (höheres Start Delay) in der Spracherkennung statistisch signifikant bevorzugen, da eine hohe Sensitivität (niedriges Start Delay) eine gewisse Unstetigkeit in die Unterhaltung einbringt.

Statutory Declaration

I declare that I have authored this thesis independently, that I have not used other than the declared sources / resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

Place Date Signature

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Ort Datum Unterschrift

Contents

List of Figures

List of Tables

List of Listings

Acknowledgements

I am indebted to my former colleagues at JOANNEUM RESEARCH, in particular Rene Kaiser who supported me throughout the process of working in the VConect Project and also during the writing of this thesis. I also thank my team members at the Know-Center in particular Elisabeth Lex for being very patient and supportive.

I especially wish to thank my advisor, Keith Andrews.

Last but not least, without the support and understanding of my loving partner Julia, this thesis would not have been possible.

Matthias Traub
Graz, Austria, 14th December 2016

Chapter 1

Introduction

This thesis describes the theoretical background of conversation metrics and the usage of video conference view modes for automated orchestration. Two comparative studies were conducted as part of the EU-funded project Vconect: Smart Video Communications [Vconect, 2016]. Vconect was a project within the Seventh Framework Programme for Research and Technological Development (FP7).

The thesis is divided into two parts. The first part describes the basic theory around video conferencing systems and statistical methods for user studies. The second part describes the two comparative studies and their results.

Chapter 2 of the thesis provides an introduction into video conferencing systems. It outlines the main functions and possibilities such systems can bring, but also discusses open questions and problems of such systems. Chapter 3 introduces terminology later used in the comparative studies. It also describes influential factors within the system and how these affect conversations. Chapter 4 gives an overview of statistical analysis with respect to analysing the results of user studies. The chapter only gives an introduction into the field of statistical analysis and illustrates the major methods available. The application of the statistical analysis methods in R is described in Chapter 5. R is a programming language and environment for statistical computing and graphics.

Finally, Chapters 6 and 7 describe the two comparative studies CS1 and CS2. They elaborate on the motivation and goals of each study, the experimental setup, the results, and a discussion. The first comparative study (CS1) compares the applicability of two view modes (tiled and full screen) within a slow turn- taking conversational situation with regard to their impact on communication and system quality. A previous study suggests a preference for full screen view mode in a slow turn-taking situation. The study was performed with 16 participants split into 4 groups of 4 using a repeated measures design. The second comparative study (CS2) investigated the impact of the start delay parameter of orchestration, which is mainly responsible for the sensitivity of the cutting behaviour. Three different degrees of sensitivity (start delay) were compared within full screen view mode. The thresholds for these three degrees of sensitivity were chosen according to insights from previous evaluations and simulations. The study was performed with 40 participants divided into 10 groups of 4 using a repeated measures design.

Chapter 2

Video Conferencing Systems

This chapter introduces the domain of video conferencing systems. A short history of video conferencing systems is followed by a discussion of open questions and problems such systems face. Some common concepts of video conferencing systems are explained and the Vconect project is introduced.

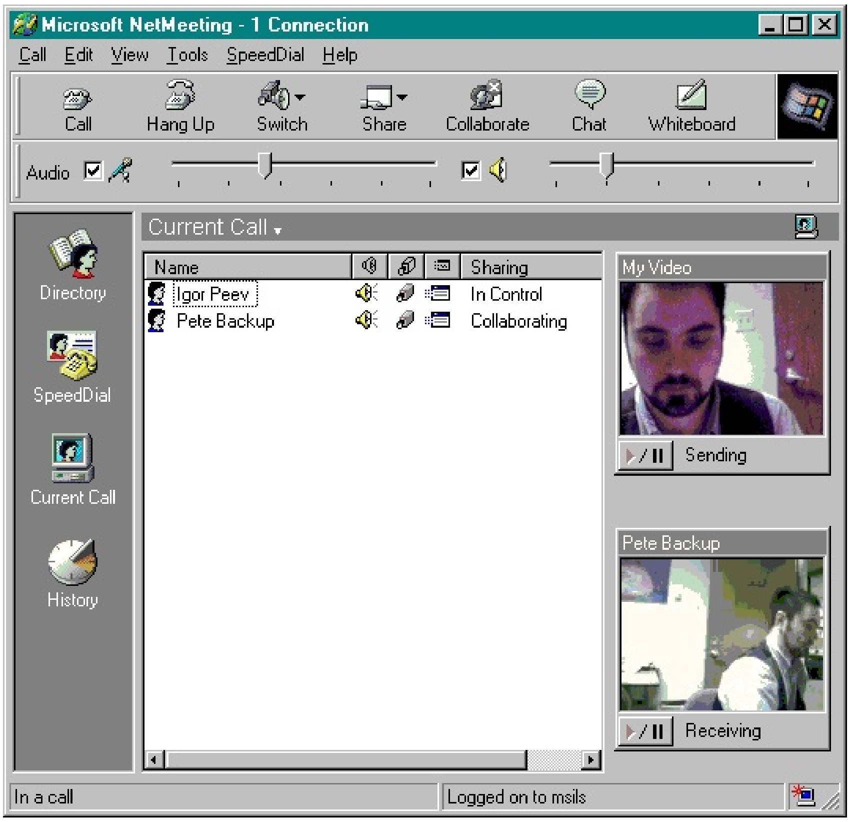
2.1 History of Video Conferencing

Video conferencing is different to video phone calls since it is designed to be used in conferences connecting several locations rather than simply two individuals. Its first commercial use was in the thirties in Germany and later, during the early seventies, in the US developed by AT&T. Other known video conference systems date back to the early eighties. Back then, the costs for such systems were enormous, ranging from US$80,000 to US $250,000. In the beginning of the nineties, technical advances in the internet protocol (IP) and more efficient video compression techniques enabled video conferencing for personal computers. In 1991, IBM together with PicTel developed the first video conferencing system for personal computers that was relatively inexpensive [FU, 1999]. From that point on, many different companies introduced video conferencing tools including Apple, Novell, and INRIA. Microsoft developed a system in 1996 called NetMeeting [Microsoft, 1998] which was updated twice in the late nineties. Figure 2.1 shows NetMeeting version 2.11 from 1998.

Specifications and standards for video encoding created by the International Telecommunication Union (ITU) enabled video conferencing to be taken seriously. The ITU established the H.263 standard [ITU, 2016a], which reduced bandwidth transmission for low bit-rate communications, as well as the H.323 standard [ITU, 2016b] for packet-based multimedia applications. At the same time, MPEG-4 [ISO, 2016] was developed by the Moving Picture Experts Group as an ISO standard for multimedia content. The combination of all these standards further advanced the concept of interoperability with respect to video content and its transmission.

In 2001, the first hand held video conferencing device was released by Samsung. Back then, video conferencing already had significant traction in business, education (e.g. online teaching and classrooms), and media (e.g. television news reporting). It was also used in the field of medicine to support remote surgery: in the first remote operation, a surgeon in the US controlled a robot in France to perform a gall bladder removal.

By 2003, the general public was able to buy affordable webcams and the personal computer had become a household commodity. Furthermore, technological advances reduced the cost of high speed internet and made it widely available. This was also the time when Skype, shown in Figure 2.2, was introduced. For a long time, Skype was unstable and low quality, but it opened the market for video conferencing systems for the public.

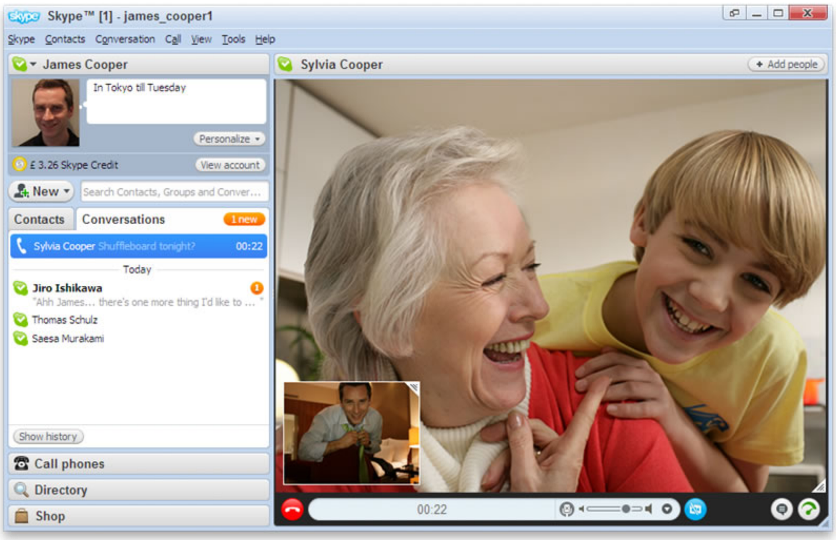


Name	Sharing
Igor Peev	In Control
Pete Backup	Collaborating

In a call

Logged on to msils

Figure 2.1: The current call window of Microsoft NetMeeting version 2.11 from 1998 [Image extracted from Microsoft [1998] and used under the terms of Austrian Copyright Law [BKA, 2015, § 42f].]



In Tokyo till Tuesday

£ 3.26 Skype Credit

Sylvia Cooper Shuffleboard tonight? 00:22

Jiro Ishikawa

"Ahh James… there's one more thing I'd like to … "

Thomas Schulz

Saesa Murakami

00:22

Figure 2.2: An early version of Skype from 2003. [Image extracted from VS, 2015 and used under the terms of Austrian Copyright Law [BKA, 2015, § 42f].]

2.2 Benefits and Drawbacks of Video Conferencing

Video conferencing systems are a backbone of modern workplace communication. Their greatest advantage is in saving time and travel costs. Video conferencing tools enable cost savings for companies and organisations due to their low acquisition and maintenance costs compared to travel costs for face-to-face meetings. Video conferencing systems are not only beneficial for corporations but also for learning environments. For example, enabling guest lecturers to speak from a remote location to a whole class. Students from one class can start a discussion with students in a different location, and students can attend class remotely or have interactive study trips to otherwise unreachable remote locations.

However, there are not only benefits to video conferencing tools, there are also some drawbacks and technical issues. Flaws in technology such as distortions and delays are sometimes perceived as flaws in the person communicating [Chen, 2003]. Since first impressions count, this can be a barrier for parties to start using online communication through a video conferencing tool. The technology has made great improvements in recent decades, but nevertheless, video distortion and network delays are ongoing weaknesses and need to be taken into consideration.

Other research has focused on the analysis and importance of audio over video in communications, again underlining the issue of audio delays. In particular, when a person tries to make a good impression (e.g. for an job interview), audio delays can cause significant detriment to the conversation. In 1988, the economist Carmen Egido discovered that delay in video conferencing reduces the other person's perceived intelligence [Egido, 1988]. Later, Kitawaki and T. Kurita [1991] concluded that audio delays may cause a speaker to be perceived as slow. Another study by Tang and Isaacs [1993] showed the influence of audio delays with respect to the quality of collaboration. Participants adapted to higher audio delay by making fewer interruptions, which also lead to fewer speaker changes and less interaction, reducing the overall quality of collaboration.

Audio delays also affect the turn-talking interactions within a discussion. Ruhleder and Jordan [2001] found that unintentional pauses may lead to negative perceptions like incompetence, negativity, or being socially awkward. In the case of a team working at a distance, it may have less impact, but makes the creation of trust with and between new conference participants harder.

Not only audio problems affect the perception of video conferencing communication. Reeves and Nass [1996] reported that video delays causing lips to be out of sync with speech made participants appear less trustworthy and less believable.

Certain reactions of people are not transmitted well through a video conferencing system, as reported by Blokland and Anderson [1998]. For example, a person leading the conversation in a real face-to-face discussion can see who is following the conversation and who is not. In a video conferencing system, on the other hand, not every participant can be displayed at the same time. A participant who is mindfully listening might not be displayed. The speaker relies on the system to put their message across. Some people rely more on audio information than on the video. This effect has to do with the topic of the conversation. If non-verbal (mimic) or deictic (pointing gestures, referring to objects) aspects play an important role, the video information takes precedence. However, with other topics, participants rely mostly on audio information.

Eye contact is hard to maintain in a video conference, and is often caused by displaced webcam positions. The influence of webcam position was investigated by Huang et al. [2002]. They found that webcam angle has an impact on perceived power and influence since certain angles can make a participant look taller or smaller.

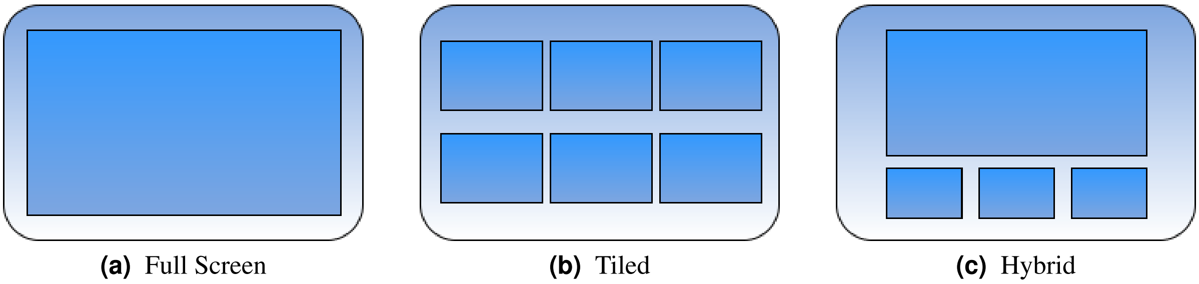

(a) Full Screen (b) Tiled (c) Hybrid

Figure 2.3: Three view modes widely used in videoconferencing systems. [Diagram drawn by the author.]

2.3 View Modes

The layout configuration of live video windows on the screen is referred to as a *view mode* or *layout*. A view mode is a pre-determined composition of 1 to n regions laid out on the screen. The regions of the view mode define the size and the spatial arrangement of the video windows. Figure 2.3 illustrates three widely used view modes: *full screen*, *tiled*, and *hybrid* (or *hangout*).

2.3.1 Full Screen

Full screen view mode, shown in Figure 2.3(a), displays only one person in the centre of the screen. The orchestration engine is fully responsible for selecting the current user to be displayed. It is also possible that the orchestration engine selects different persons to show to certain other persons, for example to display active participants to inactive participants, and vice versa.

2.3.2 Tiled

In tiled view mode, shown in Figure 2.3(b), participants see each other and themselves in a mosaic of tiles, typically arranged in two or more rows of equally sized tiles. This view mode is thought to afford good group awareness, including being able to identify where any vocal back channels originate. However, tiled view mode might be less effective compared to the full screen view mode at providing feedback through facial expressions. The video composition is static and no orchestration happens as long as there is enough space to show each participant.

2.3.3 Hybrid

Hybrid view mode,shown in Figure 2.3(c) is similar to the view mode used in Google Hangouts [Google Hangout, 2016]. It shows the current speaker in a large window in the middle of the screen and depending on the screen size, several smaller participants underneath. This allows participants to follow the current speaker, but also to keep track of some of the remaining participants. Compared to full screen view mode, switching between users becomes more forgiving. The presumption is that users do not mind if the user in the main view is not the current speaker, because they can also see some of the other participants.

2.4 Self View and Self Hearing

Self view indicates whether users can see themselves on the screen. Sometimes, it is also called a mirror or mirroring function. With self view, users have a better idea how others users might see them. It can also be useful to check whether the camera is really working or everything is on screen. Wegge [2006] studied the effects of self view with a group of 60 students. Some students became anxious when they could see themselves in the video conference. Sometimes, this anxiety was subsequently apparent to other participants in the video conference.

Self hearing refers to the possibility of a speaker hearing how their own voice is received by other participants. However, hearing one's own speech (almost always with a short transmission delay) can be very distracting, since it is perceived as another person talking.

2.5 Telepresence

Telepresence describes the user's feeling of being in the remote location. In terms of video conferencing, telepresence can be divided into individual and group telepresence. Individual telepresence is related to a single person and how this person is perceived through the video conference. For example, if their facial expressions are visible to others and themselves. On the other hand, group telepresence is related to the complete group and how the group is perceived through the video conference. For example, awareness of the other participants and who is part of the conversation.

2.6 Vconect: Smart Video Communication

The Vconect project promoted the adoption of high quality enriched video to foster mass communication within communities. The Vconect video conferencing system was developed to make intelligent decisions for mediation of communication by making innovations in four key areas [Vconect, 2016]:

- Capture: Multiple cameras and microphones transmit signals from each participant giving the ability to switch camera views and flexibility about which elements of the audio channels are used to capture (e.g. global microphone or individual).

- Composition: Different view modes can suit different participants depending on the context and the communication situation. In some situations, it my be best to see just one person talking and in other situations it may be better to see all participants.

- Transmission: A service-aware network is used within the video conferencing system to transmit all signals. Such a network can automatically decide where to place certain network components, which can reduce costs and improve the quality of experience.

- Orchestration: Orchestration decides where and how content from all the available video sources should be displayed, depending on the conversation context and the chosen view mode.

Two use cases were used to gather practical experience with Vconect:

- Performance: Using the video conferencing system to connect two sets of actors in order to deliver a scripted performance. This use case was created and developed in cooperation with the Miracle Theatre Company [Miracle Theatre Group, 2016].

- Socialisation: Together with Telecom Portugal and SAPO, the video communication system was integrated into a social network for schools [SAPO, 2016]. The system is shown in Figure 2.4.

In addition to the two use cases, several experiments were run to evaluate Vconect.

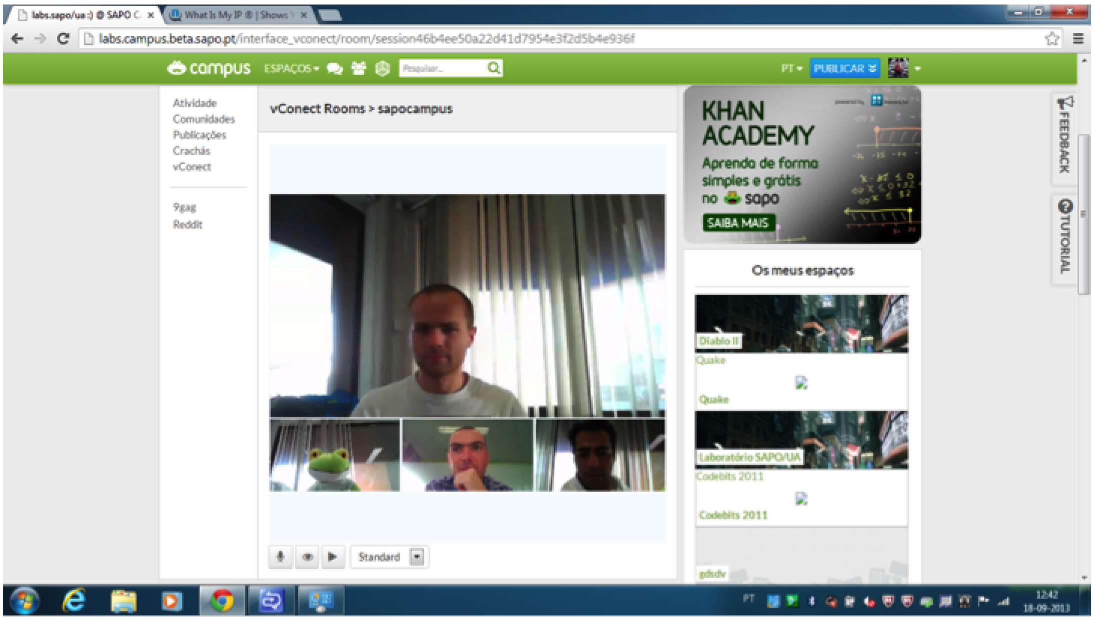


Figure 2.4: Integration of the Vconect video conferencing system into the social network of SAPO. [Image extracted from Vconect [2016] and used under the terms of Austrian Copyright Law [BKA, 2015, § 42f].]

2.7 Orchestration

This section focuses on orchestration as implemented within the Vconect project, since there is very little published work related to orchestration. Orchestration is the process which decides where and how content from all the available video sources should be displayed, depending on the conversation context and the chosen view mode. This process can be compared to compiling a live TV show, such as a discussion or debate. However, for video conferencing, it has to address the needs of the communication and conversation rather than narrative needs. The process itself can be seen as a reasoning process about audio- visual cue streams for every participating location in the communication. Orchestration operates in real time, building upon the audio processing infrastructure, and is responsible for audio-visual composition and selecting the camera to be displayed. [Weiss, Kaiser, and Falelakis, 2014]

Orchestration is typically implemented as a three step process [Kaiser et al., 2012]: The first step is *cue extraction*. Low-level cues are extracted from audio-visual streams by a number of analysis modules. The extraction is performed in real-time by aggregating events within a small sliding window of a few milliseconds. The output of this first processing step is a voice activity stream, containing all start and stop events from each source.

The second step is *fusion and interpretation* of the previously collected low-level cues. The cues from all available locations are aggregated in the *semantic lifter* module, where they are transformed into high-level cues on the basis of certain thresholds and rules. One of the major high-level cues is the detection of significant voice activity events. This process will be explained further in Chapter 3. The significant voice activity stream serves as input for various other high-level semantic events, such as *turn-shifts* and *crosstalks*. The semantic lifter module models current communication at a semantic level and also calculates conversation metrics such as turn-shifts per active participant based on a sliding temporal window. The conversation metrics are then further used to detect monologues or a heated discussion.

The last step of the process is *decision making*. The results of the semantic lifter and other modules are aggregated in the *director* module. Based on high-level cues and rules, the director module controls camera shot selection and visual layout, for each individual instance separately. The best visual layout is selected in combination with the corresponding video streams to provide the best support for each user

```
turnShift(P) ∧ isInOtherLocation(L,P) → cut2CUFront(P)
(timeSinceLast(cut2Wide) > threshold) ∧ activePerson(P) → cut2Wide(P)
patternOfShortTurnTaking(P1,P2) ∧ isInOtherLocation(L,P1) → cut2Wide(P1)
relevancyMarker(P) ∧ isInOtherLocation(P) → cut2CUFront(P)
```

Listing 2.1: Four example rules for defining the cutting decisions of the orchestration.

by respecting their given constraints.

The latter two processing steps are part of the *orchestration engine* which is the central, server-side software component. In Vconect [Ursu, Falelakis, et al., 2015], its logic is implemented by declaration using forward-chaining rules. Interpretation is done by the *JBoss Drools2* [Drools, 2016] reasoning engine. For example, Listing 2.1 shows four rules which define cutting decisions used in the Vconect system.

The main task of orchestration is to make decisions in order to mediate the conversation as it progresses. It therefore must consider two interrelated principles [Ursu, Falelakis, et al., 2015; Ursu, Groen, et al., 2013]:

- Effective message communication: In order to transfer a message effectively, it is necessary to ensure communication continuity, the transmission of social cues, and conversation markers must be visible for every participant.
- Creation of an engaging experience: The creation of an engaging and immersive experience can be compared to film and television production, where it defines the best way to represent the message.

Orchestration is based on conversation metrics derived from the conversation itself and the way it is held. The conversation metrics described in Chapter 3 can be used to model concepts such as conversation rhythm, conversation speed, communication topology, and social interaction cues like backchanneling.

The general architecture of an orchestrated video communication system is shown in Figure 2.5. The top level depicts the audio and video input and output devices from each location. These devices are connected via a configurable communication infrastructure comprising three major components: the capture and encoding component, the network processing and transmission component, and the composition and rendering component. Each of those components is controlled by the orchestrator. For the Vconect project, the communication infrastructure was implemented to provide low- delay, high-definition audio and video and to ensure continuity of video mixing [Jansen et al., 2011]. The main input to the orchestrator is provided by the feature extractor component, which extracts conversation metrics like voice activity and turn shifts. The command dispatcher is used as an output component for the orchestrator to direct commands to each participant.

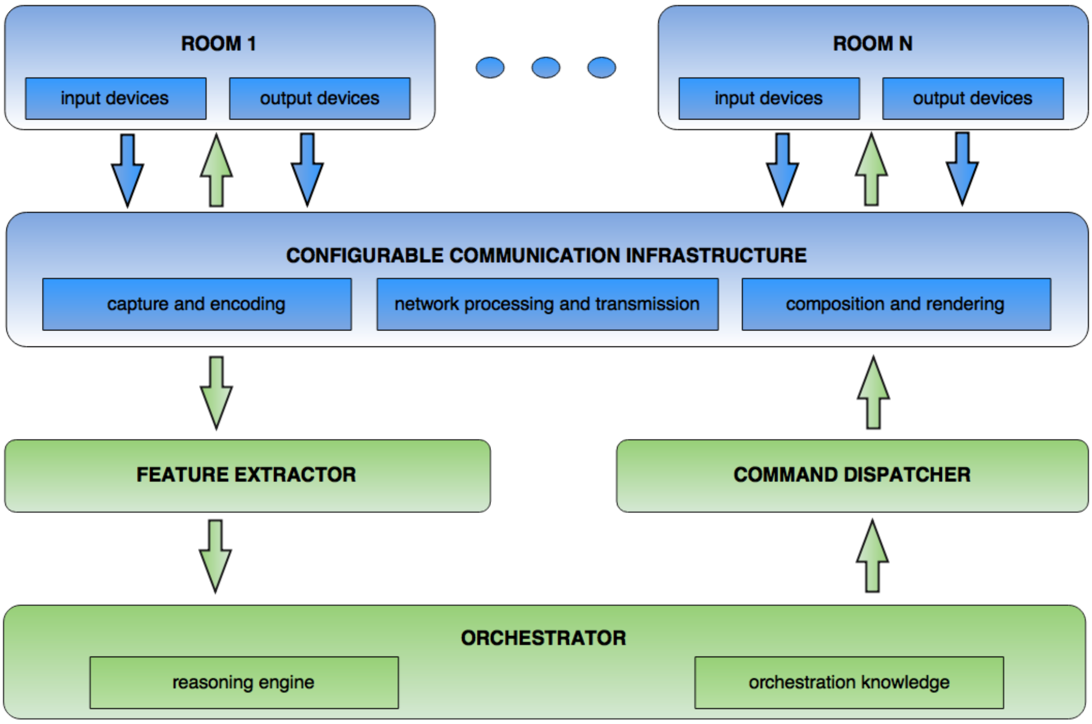


Figure 2.5: The architecture of an orchestrated video communication system. [Image redrawn from Ursu, Groen, et al., 2013]

Chapter 3

Conversation Analysis

This chapter introduces important terms and definitions which are used for the experiments described in Chapter 6 and Chapter 7.

3.1 Basic Terminology

This section introduces various terms regarding basic conversational events, building on definitions found in the scientific literature including Sellen [1992], Dabbs and Ruback [1987], Jaffe and Feldstein [1970], Brady [1968], Issing and Farber [2012], Wang et al. [2010], Hammer et al. [2004] , Reichl [2007], Berndtsson et al. [2012], Hammer et al. [2005], Ruhleder and Jordan [1999] , Gravano and Hirschberg [2011], Shriberg et al. [2001], Weiss, Kaiser, Falelakis, and Ursu [2014]. Where similar but inconsistent definitions exist, they have been adapted to make them useful for the purposes of this thesis.

3.1.1 Voice Activity (VA)

A *voice activity* (VA) or *talk spurt* begins with the first unilateral sound of a speaker. In the Vconect video conferencing system, Voice activity is detected using the “G.720.1 : Generic sound activity detector” [ITU, 2010]. Any detected low-level voice activity cues are transmitted via Apache ActiveMQ [Apache, 2015] to subsequent components. The cues are binary coded and indicate when a voice activity starts or ends. Low-level audio activity is very sensitive to background noise or irregular audio activity such as scratching or tapping the microphone, breathing directly onto the microphone, or high-pitched audio signals from the background. For a person speaking in a continuous pattern, the audio signal wave looks something like Figure 3.1. Words are separated by short drops in audio activity, due to breathing and breaks in speech. As a consequence, the voice activity cue will also contain some drops in activity level.

3.1.2 Mutual Silence

Mutual silence occurs when nobody is speaking and no voice activity is recorded. It is used to detect the end of a turn and is an indication of the current state of the conversation.

3.1.3 Turn

A *turn* indicates the activity of one participant of the conversation. A turn consists of a sequence of voice activity and pauses by one and the same speaker. The turn begins after a certain amount of continuous voice activity of the speaker. A turn ends when a turn shift occurs or the speaker stops talking. If the current speaker is interrupted briefly by another participant, the turn does not end, but the current speaker

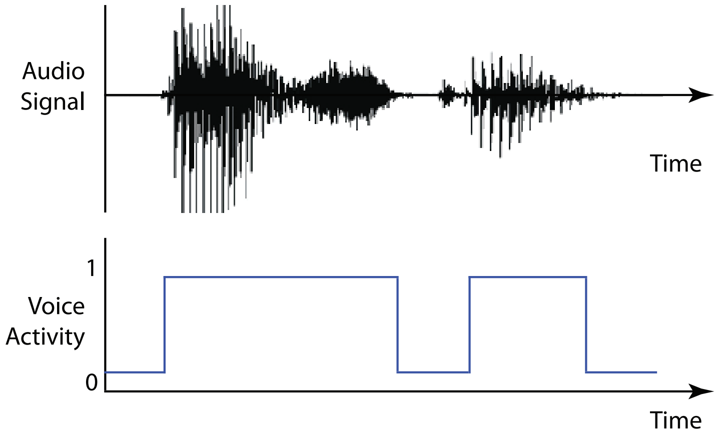


Figure 3.1: Voice Activity Detection (VAD): typical speech is displayed in the audio signal wave at the top. The equivalent voice activity signal is displayed at the bottom. [Image created by the author]

shares the floor with the interrupting person since both were talking. Turns can also contain periods of mutual silence at the end of a speaker's utterance, when no other participant takes the floor.

Sellen [1992] defines a turn, as beginning only when the participant begins to speak to the exclusion of everyone else, and when the participant is not interrupted by anyone else for at least 1.5 seconds. Without this threshold, even the briefest unilateral sound could be misconceived as a turn. 1.5 seconds is also the estimated mean duration of a phonemic clause, which is a basic unit in the encoding and decoding of speech [Jaffe and Feldstein, 1970].

3.1.4 Floor

A speaker gains the *floor* when either nobody was talking before (mutual silence), or a turn shift occurs and the floor switches to the current speaker. Conversation is an exchange of turns, and having a turn means having the right to hold the floor. Interrupting is not considered a violation as long as it does not steal the floor.

3.1.5 Turn Shift

The transfer of a turn from one person to another is called a *turn shift*. A turn shift occurs when one speaker passes the floor to another speaker. This event is time-independent, meaning there is no minimal or maximal time threshold limiting this event especially when it is due to an interruption. It is detected as soon as a different person gains the floor. A turn shift is important for the detection of the current speaker in a conversation. It can also be an indicator of the liveliness of a conversation. For example, a large number of turn shifts within a short time period may imply that at least two people are having an active discussion. In contrast, a small number of turn shifts over a long time period may imply that the conversation is more like a formal discussion, where each participant states an opinion and then passes the floor on to the next participant.

3.1.6 Crosstalk (or Double Talk)

Crosstalk situations are where one person talks while another person holds the turn. These situations can emerge out of interrupts or simultaneous starts. They situation is not restricted to only two speakers. It can happen that multiple people speak at the same time for example due to backchanneling.

3.1.7 Pause

A *pause* is a small period of mutual silence. Pauses occupy the time between sequential voice activities of a single speaker and together they comprise a turn.

3.1.8 Significant Voice Activity (SVA)

A *significant voice activity* (SVA) is extracted from the voice activity stream. It is determined by two thresholds: the start delay and stop delay. The *start delay* is defined by a certain length of time (ms) of continuous voice activity. The *stop delay*, defines the waiting time from the start of the voice activity until the voice activity is considered to be significant. In this way, small breaks within speaking or drops of voice activity are filtered out.

Significant voice activities are used to control orchestration. When a person achieves the state of SVA, the system identifies this person as an active speaker. This information is then passed on to the director engine, which is responsible for cutting (switching the displayed person) between participants.

3.1.9 Interruptions

An *interruption* occurs when a participant starts speaking while another speaker holds the floor. This can happen either on purpose or due to delay effects. Hammer et al. [2005] introduced two types of interrupts based on the difference in time. An, *active interruption* is an intended interruption or an overlap performed by one of the speakers while still listening to the other speaker. A *passive interruption* is an unintended interruption by another speaker. Both cases are illustrated in Figure 3.2. It is necessary to distinguish two different states: the state of speaker A and the state of speaker B on an absolute time scale. In Figure 3.2, these states are illustrated for a simple example voice activity stream for each of the two speakers, A and B. Between these voice activity streams, the delayed transmission channel shows the time shift between the two speakers.

In this example, speaker B receives speaker A's delayed utterance and responds after a certain amount of think time. B's response, as perceived by speaker A, is delayed by one round-trip time. After some time, A starts to talk, again assuming that B is not responding to A's first talk spurt. In effect, the delayed utterance of B interrupts speaker A without intention. On speaker B's side, B is first interrupted by A before then interrupting A on purpose. It is interesting to observe the different state sequences perceived by speaker A and speaker B, in comparison to the timestamps registered by the absolute clock in between.

The total amount of mutual silence generally increases where there are delayed responses. Furthermore, Hammer et al. [2005] concluded that higher delay times cause an increase in passive interruptions. Higher transmission delay shuffles the structure of the conversation itself and may cause irritation among the participants. It also massively disrupts the conversation, especially in highly interactive situations, and participants have to adapt their conversation behaviour in order to overcome the effect of delay on the conversation.

3.1.10 Simultaneous Start

A *simultaneous start* (or *group turn*) begins when the current speaker stops talking and gives up the floor and two or more participants start speaking together. A simultaneous start can be seen either as a single

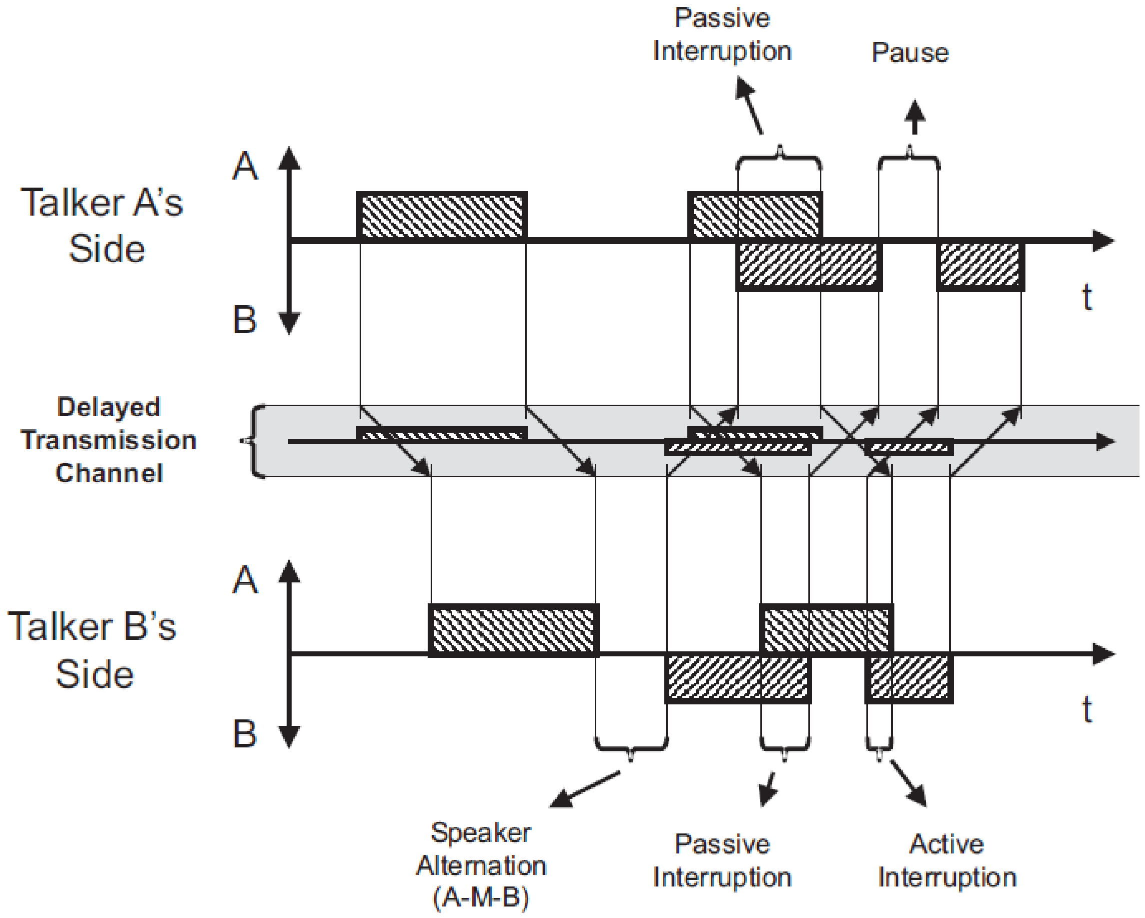


Figure 3.2: Impact of transmission delay and the two types of interruption: active and passive. The voice activity stream of speaker A is shown at the top and of speaker B at the bottom and M indicating mutual silence. The time shift in voice activity of the two speakers is shown in the delayed transmission channel in between. [Image taken from Hammer et al. [2005] under the terms of Austrian Copyright Law [BKA, 2015, § 42f].]

state indicating that it happened or as a separate event which ends when there is only one speaker left. This was described by Dabbs and Ruback [1987]. They also proposed that a group turn should also cover instances where individual speakers are effectively drowned out by the group.

3.1.11 Backchannels

A *backchannel* is short feedback given by the listener indicating that they are accepting, disagreeing, or basically just listening (paying attention) to the speaker. Examples of such backchannels are “okay”, “aha”, “mmm”, and “yes”. They are often used at the end of the speaker’s sentence or statement, but also within. Transmission delay of a backchannel can reduce communicative impact and possibly disrupt the speaker due to its late arrival [O’Conaill et al., 1993].

3.2 Conversation Metrics

Conversation metrics represent the characteristics and attributes of a conversation. They provide information about the conversation’s progression through the session and also about the current state. With the help of these metrics, it is possible for example to determine how active a conversation is and who is actively participating. The metrics are based upon the previously described terms and are calculated from the output of low-level events. Hence, conversation metrics are at a higher level of abstraction. Their calculation can be conducted for the whole conversation or just for a certain time period. The aim of the metrics is to achieve a computational interpretation of the current communication situation, providing an understanding of what is going on in the conversation and therefore of what is going on in front of the cameras. In other words, they model and represent the social aspects of a conversation, and therefore the interaction between the participants.

In practice, conversation metrics are primarily based on voice activity events and are calculated continuously within a given sliding window. For some metrics, it makes no sense to compute them for a whole session, because they do not allow any useful conclusions. The main problem with conversation metrics is that their accuracy depends on voice activity event detection. Errors in voice activity detection falsify the outcome of the metrics, which can lead to misinterpretation of the conversation status and therefore poor orchestration decisions. Interlacing and amplification of this error can occur if the calculated metric (crosstalk ratio) is based on high-level events like significant voice activity.

For the following equations, let $P = \{p_k\}$ be the set of participants, W be the time window, and $\Delta_t = 1s$ the time discretisation. The time interval $[0, W]$ is a set of $N_t := \frac{W}{\Delta_t}$ discrete time intervals , where $t_i = \Delta_t(i + 1/2)$.

For one participant p_k let $A_k(t_i)$ be the indicator if the participant is speaking at the time t_i:

$$A_k(t_i) = \begin{cases} 1 & p_k \text{ is talking} \\ 0 & p_k \text{ is silent} \end{cases} \tag{3.1}$$

Also for each participant, let $C_k(t_i)$ be the indicator if the participant is crosstalking at the time t_i:

$$C_k(t_i) = \begin{cases} 1 & \sum_k A_k(t_i) > 1 \\ 0 & \text{otherwise} \end{cases} \tag{3.2}$$

Further, continuously divide the interval $[0, W]$ into $N_{TS} \in \mathbb{N}^+$ turn shift intervals $[t^{start}_{TS\ n}, t^{end}_{TS\ n}]$. Note that the symbol $n \in [1, N_{TS}]$ denotes the turn shift index. The starting time of the first and the end time of the last turn shift interval are given by the start and end time of the moving window $t^{start}_{TS\ 1} = 0$ and $t^{end}_{TS\ N_{TS}+1} = W$, respectively. Furthermore, since the interval is continuously split into turn-shift subintervals, $t^{end}_{TS\ n} = t^{start}_{TS\ n+1}$ holds. The end of the n^{th} turn shift interval $t^{end}_{TS\ n}$ is given by

$$t^{end}_{TS\ n} : A_k(t^{end}_{TS\ n}) = 1 \ \wedge \ A_{j \neq k}(t^{start}_{TS\ n} < t < t^{end}_{TS\ n}) = 1 \text{ for } t^{end}_{TS\ n} \in [t^{start}_{TS\ n}, W] \tag{3.3}$$

Note that in the last turn shift interval, no actual turn shift occurs. Consequently, the total number of turn shifts is given by the number of turn shift intervals minus one ($N_{TS} - 1$).

3.2.1 Number of Turn Shifts

The *number of turn shifts*, ts, is the accumulated number of turn shift events of all participants during the time window:

$$ts = \sum_n \sum_k t_{TS\ n,k} \tag{3.4}$$

3.2.2 Number of Crosstalks

The *number of crosstalks*, ct, is the accumulated number of crosstalk events of all participants during the time window:

$$ct = \sum_i \sum_k C_k(t_i) \tag{3.5}$$

3.2.3 Number of Turn Shifts per Participant

The *number of turn shifts per participant*, ts_k is the accumulated number of turn shifts of all participants in the session:

$$ts_k = \sum_n t_{TS\ n,k} \tag{3.6}$$

3.2.4 Number of Active Participants

The *number of active participants*, ap, is the number of participants who were active during the time window:

$$ap = \sum_k \begin{cases} 1 & \sum_i A_k(t_i) \geq 1 \text{ participant was active at least once} \\ 0 & \sum_i A_k(t_i) = 0 \text{ participant was never active} \end{cases} \tag{3.7}$$

3.2.5 Turn Shifts Ratio

The *turn shifts ratio*, tsr is the accumulated number of turn shifts of all participants divided by the number of participants that have been active during the time window:

$$tsr = \frac{ts}{ap} \tag{3.8}$$

3.2.6 Active Participation Ratio

The *active participation ratio*, a, is the proportion of time that involves at least one active participant during the time window:

$$a = \frac{\Delta_t}{W} \sum_i \begin{cases} 1 & \sum_k A_k(t_i) \geq 1 \text{ at least 1 is talking} \\ 0 & \sum_k A_k(t_i) = 0 \text{ nobody is talking} \end{cases} \tag{3.9}$$

3.2.7 Active Participation Ratio per Participant

The *active participation ratio per participant*, a_k, is the active participation time (turn time) divided by the length of the sliding time window. It is calculated for each participant individually:

$$a_k = \frac{\Delta_t}{W} \sum_i A_k(t_i) \tag{3.10}$$

3.2.8 Silence Ratio

The *silence ratio*, s, is defined as the proportion of time that involves no active participant i.e., essentially the inverse of the active participation ratio:

$$s = 1 - a \tag{3.11}$$

3.2.9 Crosstalk Ratio

The *crosstalk ratio*, c, is defined as the proportion of time that involves at least one participant crosstalking:

$$c = \frac{\Delta_t}{W} \sum_i \begin{cases} 1 & \sum_k C_k(t_i) \geq 2 \text{ at least 1 is crosstalking} \\ 0 & \sum_k C_k(t_i) < 2 \text{ nobody is crosstalking} \end{cases} \tag{3.12}$$

3.2.10 Crosstalk Ratio per Participant

The *crosstalk ratio per participant*, c_k is defined as the proportion of time that one participant is crosstalking within the window:

$$c_k = \frac{\Delta_t}{W} \sum_i C_k(t_i) \tag{3.13}$$

3.2.11 Heated Discussion

A *heated discussion* occurs when the conversational temperature is high, which is indicated by a high number of turn shifts within a short time period, suggesting that the participants might be excited. The heated discussion metric evaluates to true if the number of turn shifts per active participant divided by the number of active participants is above a certain threshold. It reverts to the state of not heated when it falls below a second threshold.

In current implementation of Vconect, the threshold for beginning a heated discussion is 8.0, and for dismissing a heated discussion 6.5.

3.2.12 Monologue

A *monologue* is when one person holds the turn for more than a certain proportion of the time in the sliding time window. The monologue state reverts back to the normal state when this value falls below a second threshold.

In the current implementation of Vconect, the threshold for a creating a monologue is set to 60%, and for dismissing a monologue to 70%. These values were determined empirically through technical trials within the project group.

3.3 Influential Factors in Orchestration

The orchestration engine's main input are the voice activity events created by the voice activity detection module. These voice activity events are transformed into significant voice activity events. Two major factors influence this transformation: the start delay regulating after what period of continuous activity it will be significant, and the stop delay regulating the expiration time of a significant voice activity. In addition, the cut freeze time regulates the minimum amount of delay between director cuttings.

3.3.1 Start Delay

As defined in Section 3.1.8, *start delay* is the time (in ms) from which a voice activity is considered to be a significant voice activity. It has great influence both in terms of distinguishing a real speaker from arbitrary background noise and in terms of perceiving and keeping track of a conversation. The principle is illustrated in Figure 3.3. In the topmost plot the voice activity of a user is displayed. If the length of active voice activity exceeds the start delay point, the voice activity is considered to be significant. Significant voice activity is displayed in the middle plot. The reaction of the director is displayed in the bottom plot, tracking the behaviour of the significant voice activity.

In the case of a shorter start delay, the orchestration engine would consider short voice activities to be significant voice activities. This would lead to fast reactions by the director if a participant starts talking or makes a noise. Such behaviour could be perceived as very dynamic and active if the conversation is heated. However, it could also increase the false positive rate when detecting speaking persons and the perceived behaviour of the orchestration would be more or less random, as illustrated in Figure 3.4.

A longer start delay would mean that the orchestration engine waits longer before it considers a voice activity to be significant. Longer start delays lead to higher precision, since short voice activities produced by background noise would not be considered significant. Communication back channelling, depending on the length, would not be considered as significant audio activity either, which reduces the recall of the orchestration accuracy. A short "yes" or "no" to a question would not be recognised by the director and the system would not cut to the back channelling person. On the other hand, cutting would be delayed and the beginning of a spoken word or sentence might be missed by the longer start delay. In addition, voice activities which are only slightly longer than the start delay would cause the director to cut, but the selected person would have already stopped talking. The effects of a longer start delay are shown in Figure 3.5. In the upper plot, the voice activity is jagged and is not considered to be significant. After a period of time, the voice activity lasts long enough to reach significance, as shown in the lower plot.

3.3.2 Stop Delay

The *stop delay* is defined by the waiting time after a person stops talking before a significant voice activity is closed. When a person stops talking only for a few milliseconds, a stop delay of only a few milliseconds would lead to the sudden ending of significant voice activity and loss of the floor. The

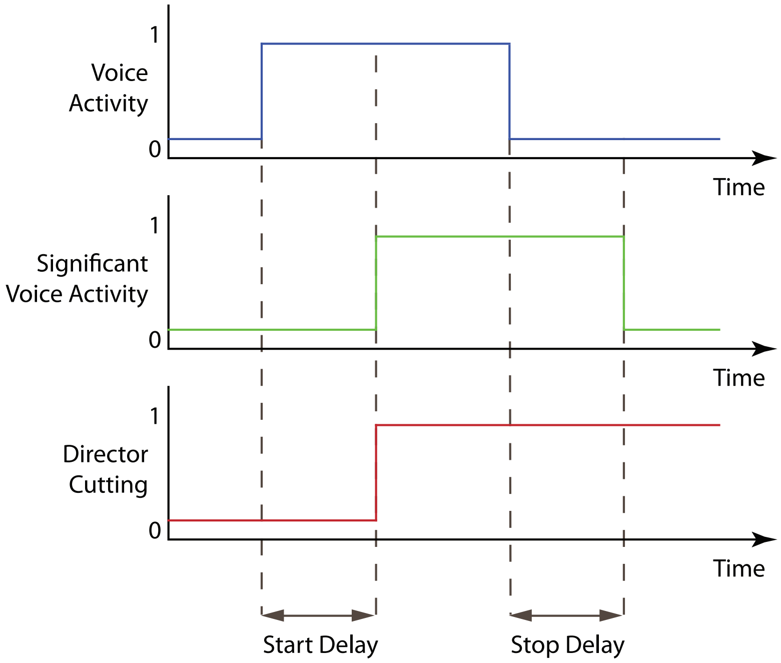


Figure 3.3:

Start delay and stop delay determine when a voice activity is considered significant. The voice activity of one participant is displayed in the topmost plot. If the voice activity exceeds the start delay threshold, the voice activity becomes significant, as shown in the middle plot. This significant voice activity leads to a director cutting action (e.g. switching to the active speaker).

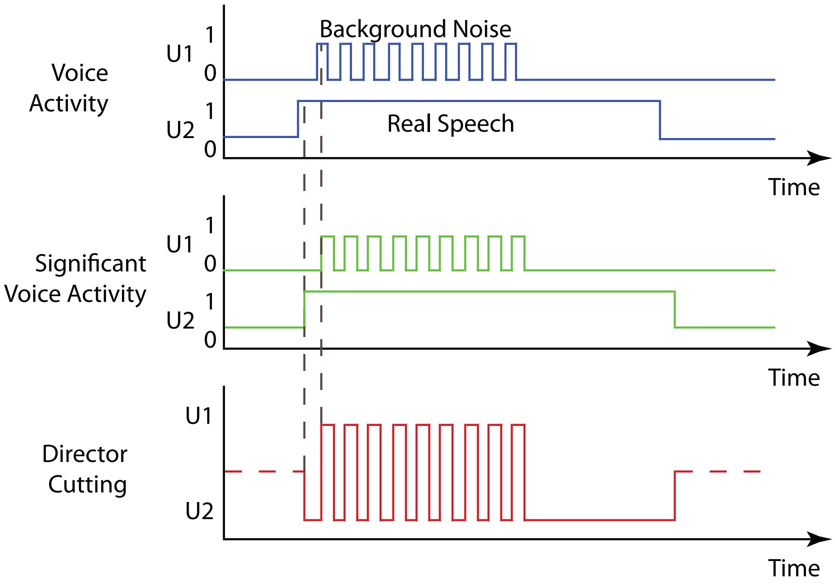


Figure 3.4: The effects of too short a start delay on detecting significant voice activity. This example shows two users U1 and U2. In the topmost plot, the voice activity events reaching the orchestration engine are illustrated. U1 has many on and off events due to background noise, whereas U2 has continuous speech with stable voice activity. With too short a start delay, the on and off events of U1 would be considered to be significant voice activity, as shown in the middle plot. This leads to many wrong director cutting actions, as shown in the bottom plot.

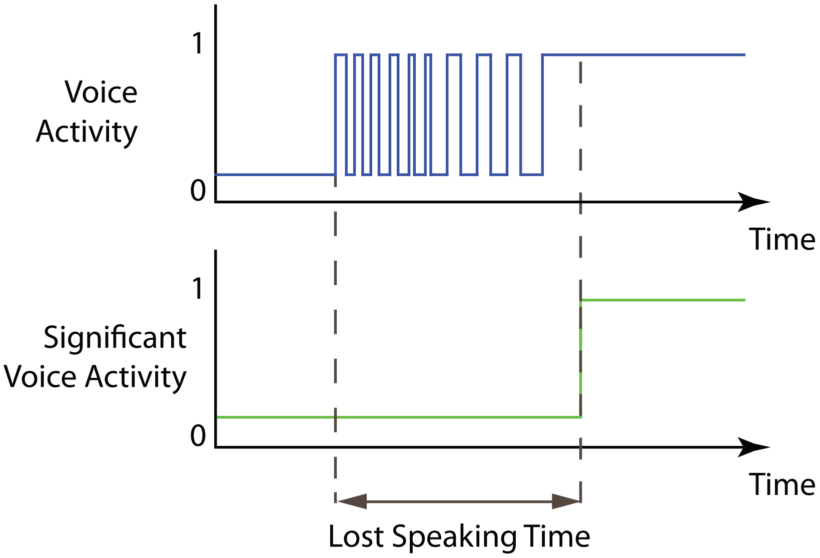


Figure 3.5: The effects of too long a start delay on detecting significant voice activity. The upper plot shows the voice activity events of a single user. In the beginning, the voice activity stream is very fragmented. If the start delay is too long, the orchestration engine would not recognise these events as significant. This leads to a delay in cutting time and an active speaker not being on camera.

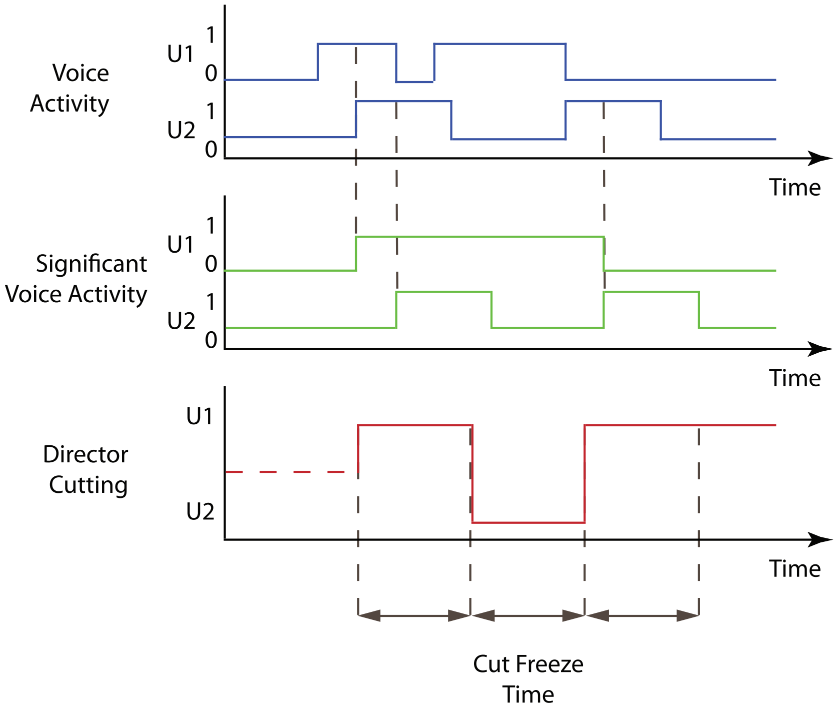


Figure 3.6: The effects of too long a stop delay in combination with cut freeze time based on two users U1 and U2. The bottom plot shows the director's cutting actions regulated by the cut freeze time. The cuts are mainly driven by too long a stop delay. In plot at the bottom where the cut freeze time has a negative influence on the directors cutting actions, which results in cuts to the end of a speaking time of the users.

director would react and switch to the next active speaking person. However, human speech has short pauses between words and sentences and those pauses should not lead to instant turn shifts or director cuts. The stop delay keeps the turn alive for a certain amount of time unless there is another voice activity. The stop delay is reset and starts again with the next drop of voice activity. The stop delay is illustrated in Figure 3.3. After the voice activity level drops back to 0, the orchestration engine waits for a certain amount of time and then drops the significance of the voice activity. The director stays with the current active user until another significant voice activity is determined to have started.

In the case of too short a stop delay, the orchestration would cut off significant voice activities rather quickly. A significant voice activity (SVA) could be terminated simply by a short break of speech. These short breaks could be caused by sensitivity of the system or minor breaks in speaking. After every drop of an SVA, the start delay blocks the participant from gaining the floor again quickly.

Too long a stop delay would mean that a participant would keep the SVA active and the orchestration would possibly consider this participant to be an active speaker for too long. This could lead to situations where the director switches to perceived active participants after they have already stopped talking. The focus should remain on this participant, until at least the cut freeze time has passed. Too long a stop delay would also alter other higher level metrics like the conversation temperature, since participants have longer voice activity significance. The system could falsely switch into heated discussion mode more often. The effects of too long a stop delay are shown in Figure 3.6

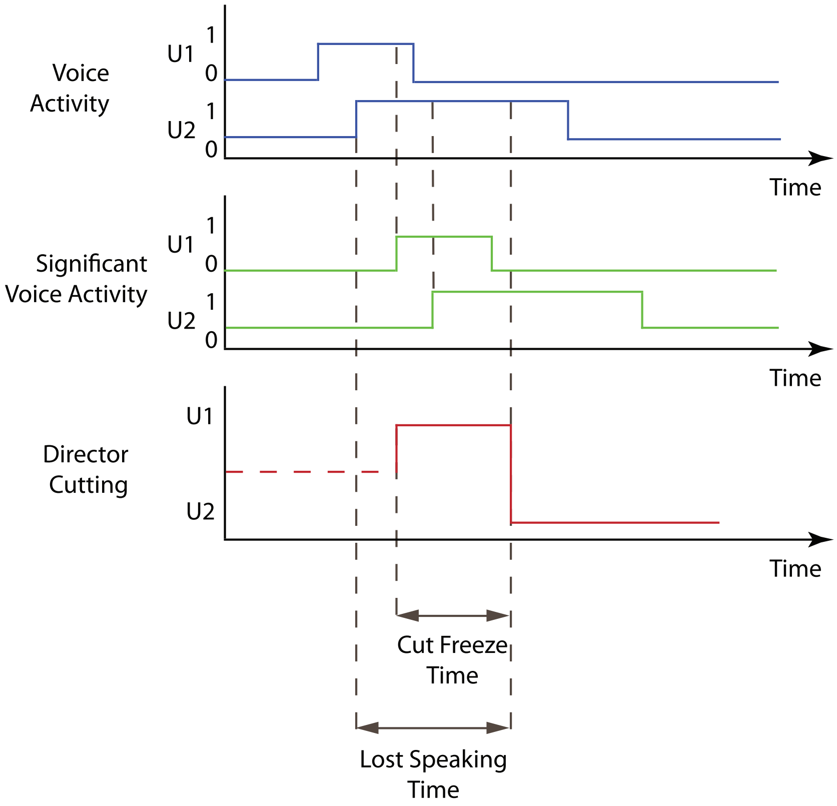


Figure 3.7: The negative effects of too long a cut freeze time on orchestration decision making. Here, the cut freeze time is twice the length of the start delay. The bottom plot shows director cutting actions. The first cut is made to U1 shortly before U1 stops talking. The long cut freeze time prevents the director from cutting to U2 and keeps the view on a non-speaking person.

3.3.3 Cut Freeze Time (CFT)

The parameter *cut freeze time* blocks the director from cutting too quickly after the previous cut. It was introduced to reduce the frequency of cutting in a heated discussion or if the sensitivity of the system was too high. In Figure 3.6, the negative effect of too long a cut freeze time is shown. In the example, the cut freeze time is twice the length of the start delay. In the top plot (voice activity), the second speaker interrupts the first speaker. This is reflected in the significant voice activity plot in the middle. In this case, the first break of speaker U1 does not result in a loss of significance, whereas the interruption of speaker U2 results in a significant voice activity. This reaction would cause the director to switch to speaker U2, but the cut freeze time of the first cut delays the director from doing so. As a consequence, the director cuts shortly before speaker U2 stops talking. In this case, speaker U2 is displayed on camera without having any voice activity. After the cut freeze time has once again elapsed, the director cuts back to speaker U1. In this case, speaker U1 has already stopped talking, but retained significance due to the stop delay and the start delay prevented speaker U2 from regaining the floor before the director cuts back to speaker U1. The same effect can be seen in Figure 3.7.

Chapter 4

Statistical Analysis

The chapter introduces the field of statistical analysis for comparative studies and illustrates the major methods available. The definitions and equations are are based on different sources, including from Moosbrugger [2002], Griffiths [2009], Eder [2007], Bortz and Döring [2006], Montgomery [2009], Sison and Glaz [1995], and Pinheiro and Bates [2000].

4.1 Statistical Analysis of Formal Experiments

The results of a formal experiment are analysed with statistical methods. Depending on the experimental design, different methods may come into operation. Table 4.1 gives an overview of different analysis methods and their goals. The left side of the table covers parametric methods which presuppose an estimation of population parameters such as the mean. The right side covers non-parametric methods, which are distribution-free and rely only on ordering (ranking) of the observations. The distribution in the measured data determines the choice between these two groups of methods. If the data appears to follow a normal distribution, a parametric method can be used. Otherwise, if the distribution is unknown or not clearly distinguishable, non-parametric methods are used. For example, Kruskal Wallis [StatSoft, 2016c] is the non-parametric alternative corresponding to ANOVA (Analysis of Variance) [StatSoft, 2016a] and is used where the data does not follow a normal distribution. Similarly, the Wilcoxon signed ranks test [StatSoft, 2016f] is the non-parametric alternative to the one-sample T-Test [StatSoft, 2016e]. Not only the distribution plays a role in deciding which method to use. The type of data is also important. Nominal or ordinal data demands the use of non-parametric methods. For interval and ratio data, non-parametric methods have to be used if the population cannot be assumed to be normally distributed. It is necessary to check whether the data follows a normal distribution before the main statistical comparison of the data starts. The outcome of this parametric inspection is essential to determine which further analysis method should be used.

4.2 The Null Hypothesis and Type I and II Errors

Type I and II errors describe the erroneous detection of statistical significance. A *null hypothesis* is a statement that the analysed factor has no influence or makes no difference. An example of such a null hypothesis would be "Cutting speed has no influence on conversation behaviour".

A *Type I error* is the incorrect denial of a true null hypothesis. It would be a false positive detection with respect to a non-null hypothesis. This type of error leads to conclusions that an alleged relationship exists when there is none.

A *Type II error* is the failure to reject a null hypothesis. It would be a true negative detection with respect to a non-null hypothesis. This type of error leads to the conclusion that an alleged relationship

Parametric Methods	Goal	Non-Parametric Methods	Goal
Two Sample T-Test	To see if two samples have identical population means.	**Wilcoxon Rank-Sum Test**	To see if two samples have identical population medians.
One Sample T-Test	To test a hypothesis about the mean of the population a sample was taken from.	**Wilcoxon Signed Ranks Test**	To test a hypothesis about the median of the population a sample was taken from.
Pearson's Chi-Squared Test for Goodness of Fit	To see if a sample fits a theoretical distribution, such as a normal curve.	**Kolmogorov Smirnov Test**	To see if a sample could have come from a certain distribution.
ANOVA	To see if two or more sample means are significantly different.	**Kruskal Wallis Test**	To test if two ore more sample medians are significantly different.

Table 4.1: Parametric statistical methods are used if the data are sufficiently normally distributed. Otherwise, non-parametric methods must be used.

does not exist when in fact it does.

4.3 P-Value and Significance level

The p-value is the estimated probability of rejecting the null hypothesis when that hypothesis is true. The null hypothesis is usually a hypothesis of "no effect or difference". If the p-value is less than the chosen significance level, the null hypothesis is rejected and there was in fact a difference. The term significance level, α, is used to refer to a chosen probability. The fidelity of the experiment is defined by the significance level and is conventionally set at:

- $\alpha = 5\% \rightarrow p < 0.05$ is common in human-computer interaction.
- $\alpha = 1\% \rightarrow p < 0.01$
- $\alpha = 0.1\% \rightarrow p < 0.001$ is common in medical experiments.

If the null hypothesis of "no effect" is rejected at $\alpha = 5\%$, then the result is considered "statistically significant at p < 0.05".

4.4 Familywise Error Rate (FWER)

The Familywise Error Rate (FWER) is the probability of making one or more false positive detections (type I errors) among individual hypotheses when conducting a multiple hypothesis test. FWER methods (like the Bonferroni correction method) provide more control over false detection and try to reduce the probability of even one false detection.

Consider m null hypotheses, labelled as $H_1, H_2, \ldots, H_m$. By performing a statistical analysis, each hypothesis is assigned as significant or non-significant. The combined results over H_i are displayed in the Table 4.2, whereby:

	Null Hypothesis is True	Alternative Hypothesis is True	Total
Significant	FP	TP	R
Non - Significant	TN	FN	m - R
Total	m_0	m - m_0	m

Table 4.2: Combined result of H_i and its related variables.

- m_0 is the number of true null hypotheses, an unknown parameter.
- fp is the number of false positives (type I error).
- tp is the number of true positives.
- fn is the number of false negatives (type II error).
- tn is the number of true negatives.
- r is the number of rejected null hypothesis.
- r is an observable random variable, while fp, tp, fn, tn are unobservable random variables.

The FWER,fw is the probability of making even one type I error in the compound family of hypotheses:

$$fw = Pr(fp \geq 1) \quad or \quad fw = 1 - Pr(fp = 0) \tag{4.1}$$

Therefore, if the $fw \leq \alpha$, the probability of making even one type I error is controlled at level α itself. Possible controlling procedures include Bonferroni method covered in Section 4.5 and Holm-Bonferroni in Section /refsec:holm

4.5 Bonferroni Correction Method

The Bonferroni Correction Method [StatSoft, 2016b] is used to counteract the problem of multiplicity which increases Familywise Error Rate. It is considered the simplest and most conservative method to address this issue. The correction method adjusts the alpha value of the main hypothesis. Assuming, the experiment is testing n independent hypotheses on a set of data, one way of maintaining the familywise error rate (type I error) is to test each hypothesis separately. When doing so, every statistical significance level is reduced to 1/n times what it would be for the overall hypothesis.

For example, considering a combined hypothesis with $m \geq 2$ paired comparisons where each individual hypothesis is tested with the significance value α'. This would result in the following inequation with α being the overall significance value:

$$\alpha' \leq \alpha \leq m \cdot \alpha' \tag{4.2}$$

The combined significance value is capped. Hence the individual significance value is set to

$$\alpha' = \frac{\alpha}{m} \quad , \quad \sum_{i=1}^{m} \frac{\alpha}{m} = \alpha \tag{4.3}$$

and the combined significance value can not exceed α. For example, for multiple comparisons between 3 conditions at $\alpha = 5\%$:

- Condition 1 against Condition 2
- Condition 1 against Condition 3
- Condition 2 against Condition 3

each pair must be analysed with an α of $5/3 = 1.67$ to prevent type I errors.

4.6 Holm-Bonferroni Method

The Holm-Bonferroni Method [ME, 2016] is a variant of the Bonferroni Correction Method. It is a stepwise algorithm which is uniformly more powerful than the simple Bonferroni Correction Method. The Holm-Bonferroni Method is split into 6 steps:

1. Set global α-Niveaus α_g.
2. Conduct all single tests and calculate the p-values.
3. Sort the p-values from small to large.
4. Calculate the local α-Niveau as a ratio between the global and the number of tests.
   $i = 1, \dots, k \quad \alpha_1 = \frac{\alpha_g}{k}, \quad \alpha_2 = \frac{\alpha_g}{k-1}, \quad \dots \quad \alpha_i = \frac{\alpha_g}{k-i+1}$
5. Compare the calculated p-values with the sorted local α-Niveau starting with α_1. Repeat this step until a p-value is larger then the corresponding α_i
6. All H_0 with a lower p-value than the local α_i-Niveau can be rejected. Every H_0 with a higher p-value than the local α_i- Niveau can not be rejected.

4.7 Shapiro-Wilk Test of Normality

The Shapiro-Wilk Test is a test of normality. It was published in 1965 by Samuel Sanford Shapiro and Martin Wilk [StatSoft, 2016d]. It uses the null hypothesis principle to check whether the test sample x came from a normally distributed population. Hence, if the p-value is less than the chosen alpha level, the null hypothesis is rejected, and there is strong evidence that the analysed data are not normally distributed. The specification of the test is as follows:

$$W = \frac{(\sum_{i=1}^{n} a_i x_{(i)})^2}{\sum_{i=1}^{n} (x_i - \bar{x})^2} \tag{4.4}$$

where:

- $x_{(i)}$ is the i^{th} order statistic (i.e. the i^{th}-smallest number in the sample).
- $\bar{x} = \frac{(x_1 + \cdots + x_n)}{n}$ is the sample mean.
- the constants a_i are given by $(a_1, \dots, a_n) = \frac{m^T V^{-1}}{(m^T V^{-1} V^{-1} m)^{1/2}}$ where $m = (m_1, \dots, m_m)^T$ and $m_1, \dots, m_m$ are the expected values of the order statistics of independent and identically distributed random variables sampled from the standard normal distribution, and V is the covariance matrix of those order statistics.

The null hypothesis may be rejected if W is below a predetermined threshold. For experimental datasets, if the Shapiro-wilk test is significant, the data is not sufficiently normally distributed, and non-parametric methods must be used for further analysis.

4.8 Sison-Glaz Multinomial Proportions

The method proposed by Sison and Glaz [1995] uses a vector of observations with the number of samples falling in a class of a multinomial distribution. The multinomial distributions are then used to build

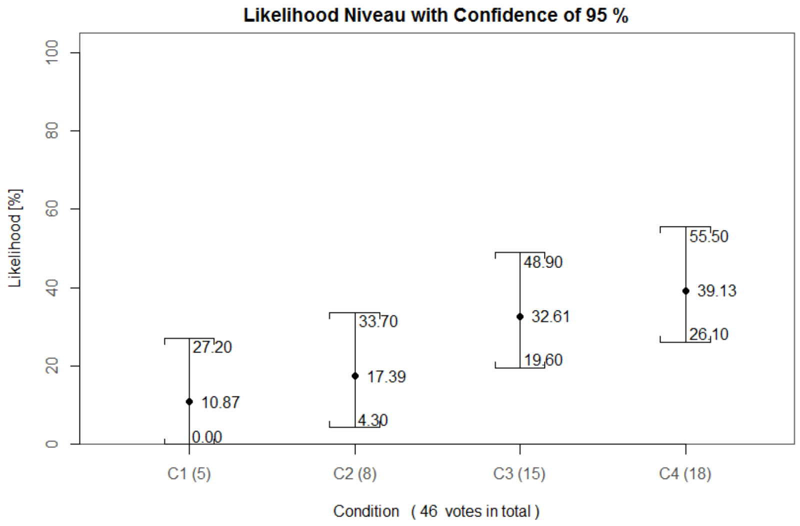


Figure 4.1: Confidence intervals calculated with the Sison-Glaz method. Every interval overlaps with every other interval, hence there are no statistically significant differences.

simultaneous confidence intervals for the probabilities. If two confidence intervals do not overlap, the difference between those distributions is statistically significant. Figure 4.1 shows a plot for four intervals. Every interval overlaps with every other interval, hence there are no statistically significant differences. In Figure 4.2, for the voting distribution $[5, 8, 15, 27]$, the interval for C4 overlaps with the interval C3, hence the difference in votes between C4 and C3 is not statistically significant. However, C4 does not overlap with either C1 or C2, hence the number of votes for C4 is statistically significantly higher than for C1 and C2. All of the other intervals overlap, so there are no further statistically significant differences.

4.9 Pearson's Chi Square Test

The Pearson's or one-way chi square test [Plackett, 1983] describes the discrepancy between a data set and its assumed distribution. The test statistic is a χ^2 distribution when the null hypothesis is true. The expression of a chi square test compares each preference to each associated expected frequency:

$$\chi^2 = \sum \left(\frac{(f_o - f_e)^2}{f_e} \right) \tag{4.5}$$

where f_o is the sample frequency and f_e the expected frequency. The degree of freedom is calculated as $df = k - 1$, where k is the number of categories.

For example, to test the hypothesis that a random sample of 40 people have to choose between 2 options, the observed number of choices for A and B would be compared to the theoretical distribution of 20 for A and 20 for B. If there were 16 votes for A in the sample and 24 votes for B, then

$$\chi^2 = \frac{(16 - 20)^2}{20} + \frac{(24 - 20)^2}{20} = 1.6 \tag{4.6}$$

The null hypothesis is true if A and B are chosen equally. For this example, the degree of freedom $df = 2 - 1 = 1$ and with a p value of 0.005 the χ^2 must be bigger than 3.84 to reject the null hypothesis. In this example, $\chi^2 = 1.6 < 3.84$ therefore the sample distribution is not significantly different from the theoretical. Thus, the apparent preference for B is not statistically significant.

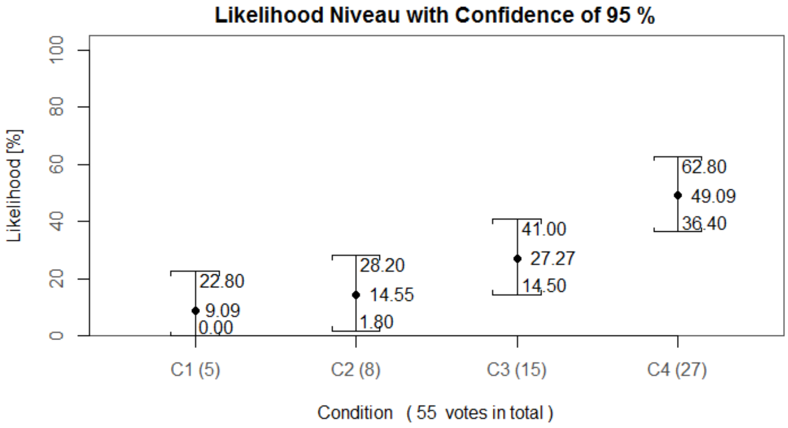


Figure 4.2: Confidence intervals calculated with the Sison-Glaz method. C4 has no overlap with C1 and C2, hence there is a statistically significant difference between these votes. C4 overlaps with C3, meaning this difference in votes is not statistically significant.

4.10 Repeated Measures T-Test

A t-test, also called Student's t-test [StatSoft, 2016e], is a statistical hypothesis test based on Student's t-distribution. It is used to check if two datasets are significantly different from each other, where both datasets follow a normal distribution. The t-test looks at the differences in mean values of the two datasets. The null hypothesis is defined as:

$$H_0 : \mu_x - \mu_y = \omega_0 \tag{4.7}$$

where μ_x and μ_y are the two means and ω_0 is the difference.
The test statistic is calculated with the following equation:

$$T = \sqrt{\frac{nm}{n+m}} * \frac{\overline{X} - \overline{Y} - \omega_0}{S} \tag{4.8}$$

where m is the size of sample set X, n is the size of sample set Y, and the weighted variance S is calculated from the data set variances S_x^2 and S_y^2 by:

$$S^2 = \frac{(n-1)S_x^2 + (m+1)S_y^2}{n+m-2} \tag{4.9}$$

The degree of freedom is defined as $df = m + n - 2$.

4.11 Wilcoxon Rank-Sum Test

The Wilcoxon rank-sum test [StatSoft, 2016f] analyses whether two datasets are identical (i.e. come from the same population) without the prerequisite that the datasets following a normal distribution. It is also known as the Mann-Whitney-Wilcoxon test and is a nonparametric test of the null hypothesis. The Wilcoxon rank-sum test is often confused with the Wilcoxon signed-rank test, which also involves summation of ranks but uses dependent data pairs. The Wilcoxon rank-sum test is a combination of two test statistics, the Wilcoxon rank-sum:

$$W_{m,n} = U + \frac{m(m+1)}{2} \tag{4.10}$$

where m is the size of sample set X, n is the size of sample set Y, and the Mann-Whitney-U statistic:

$$U = \sum_{i=1}^{m} \sum_{j=1}^{n} S(X_i, Y_i) \tag{4.11}$$

where $S(X, Y) = 1$ if Y < X (Y and X being the samples), and 0 otherwise.

The null hypothesis is defined as:

$$H_0 : \mu_x = \mu_y vs.H_1 : \mu_x \neq \mu_y \tag{4.12}$$

where μ_x and μ_y are the two means.

4.12 ANOVA

ANOVA (Analysis of Variance) [StatSoft, 2016a] is an important technique for analysing the effects of categorical factors on two or more means. The method is called analysis of variance, because the inferences about means are based on an analysis of variance. It is used to test general difference among means, so in paractice further post-hoc tests are usually needed. The basic one-way ANOVA is used if there is only one single categorical factor (e.g. start delay value) dividing the participants into several groups. The analysis itself can be based on several different approaches, but the most commonly used is a linear model. The model is linear in terms of its parameters, but can be non-linear in factors. Two prerequisites for ANOVA are normal distribution of the data and independence between factors.

The basic linear model or linear regression model is defined in Equation 4.13, where β is the regression coefficient, ϕ the non-linear function, and X_i the independent variable:

$$Y_i = \beta_0 + \beta_1\phi_1(X_{i1}) \tag{4.13}$$

A statistically significant result of an ANOVA test is usually accompanied by one or more follow-up tests. For example, the pairwise t-test can be used to compare every group mean with every other group mean in order to control type I errors.

Other popular varieties of ANOVA include MANOVA and Factorial ANOVA [StatSoft, 2016a]. Multivariate analysis of variance or multiple analysis of variance (MANOVA) is used if there are multiple dependent variables. If the dependent variables are correlated, MANOVA is most effective, but if these variables are overly correlated, it can be assumed that they project the same process (behaviour) and a simple ANOVA would be more suitable. Factorial ANOVA is used to study interaction effects in a factorial experiment, where there are two or more factors and all possible combinations of these are taken into account.

4.13 Mixed Model

The mixed model, or linear mixed-effects model, is an extension of the linear regression model. It is used for data which can be divided into groups and it is particularly useful for experiments where there are repeated measurements on the same factor, but also for grouped data such as from longitudinal, multilevel, and block designs. One benefit of a mixed model is its capability to compensate missing values, where it is favoured over repeated measures ANOVA. The relationship between independent variables and response variables is described within the mixed model according to one or more classification factors and contains both fixed effects and random effects. The fixed effects represent the linear regression part, whereas the random effects are linked to randomly drawn individual experimental units from a population. [Pinheiro and Bates, 2000]

4.14 Friedman Test

The Friedman Test [Galili, 2010] is a non-parametric alternative to one-way ANOVA with repeated measures, developed by the U.S. economist Milton Friedman. It is used to test for differences between groups when the dependent variable being measured is ordinal. It can also be used for continuous data that has violated the assumptions necessary to run the one -way ANOVA with repeated measures, such as a marked deviation from normality. The Friedman test is used for one-way repeated measures analysis of variance by ranks. In its use, it is similar to the Kruskal-Wallis one-way analysis of variance by ranks.

The main differences between the Friedman and the Wilcoxon test is the number of conditions to be assessed. The Wilcoxon test is limited to two conditions, whereas the Friedman test allows two or more. This restriction of the Wilcoxon test is why the Friedman test was used in the second experiment (CS2), where participants were evaluated over three different conditions (see Section 7.9.2).

If the result of the Friedman test is significant (i.e. there is a significant difference between the conditions where the groups were tested), it is necessary to run a post-hoc analysis to determine where the specific differences lie. This can be accomplished using the Wilcoxon Signed-Rank Test pairwise on the conditions.

Analysing multiple comparisons can lead to statistical interference. Increasing the number of hypotheses in a test also increases the likelihood of detecting a rare event, and thereby, the chance of rejecting the null hypotheses when it is true (type I error).

Chapter 5

Statistical Analysis With R

R is a powerful programming language and environment for statistical computing and graphics. It is a free, open-source project, which grew out of the S language and environment developed at Bell Laboratories (formerly AT&T, now Lucent Technologies) by John Chambers and colleagues [R, 2016].

The following information is taken from an introduction to r 1 Clark [2014] and 2 Clark [2012]. R is a true object-oriented programming language, much like C++ or Python. It is quite similar to other programming packages such as MatLab (not freeware), but more user-friendly than programming languages such as Fortran. Objects are manipulated by functions, creating new objects, which may then have more functions applied to them. Objects can be just about anything: a single value, a variable, a dataset, lists comprising several types of objects, etc. The object's class (e.g. numeric, factor, data frame, matrix, etc.) determines how a generic function (like summary or plot) will treat the object.

5.1 Using R

An installation of R and accompanying packages provides a fully functioning statistical environment in which one may conduct any number of typical and advanced analyses may be conducted. R is usually installed in combination with RStudio [RStudio, 2016] or another GUI. R can be downloaded from the main project site [R, 2015] or one of its many mirrors. CRAN (Comprehensive R Archive Network) is a network of servers which store up-to-date mirrors of the R project.

In this chapter, an example data file with three sets of ratings, called PS1, PS2, and PS3 will be used. PS1 can be read as "parameter set 1", and so forth. The CSV file in this case uses German syntax with ";" as the delimiter.

5.1.1 Session and Working Directory

After R is started, a console awaits input, as shown in Listing 5.1. At the command prompt, ">", it is possible to enter numbers and perform calculations. The working directory is the folder in which commands will be executed. When instructed to open a certain file, R will look for it in the working directory. When instructed to save a data file or figure, R will save it in the working directory. All text after a hash character "#" on the same line is considered to be a comment.

5.1.2 External Data

R offers many options for loading external data, including Excel, CSV, and SPSS files. The example CSV file shown in Listing 5.2 is read and printed in Listing 5.3. Each different format has dedicated read

```
R version 3.1.0 (2014-04-10) -- "Spring Dance"
Copyright (C) 2014 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.
>
> 1 + 2
[1] 3
>
>getwd() #Get or Set Working Directory
[1] "C:/Users/"
```

Listing 5.1: An example session in the R console, showing a simple calculation and a command with a comment.

functions. Most of them can be found through “?read”. If a certain format is not supported, support may be available in additional packages or libraries.

5.1.3 Scalars, Vectors, and Matrices

To assign values to variables, the assignment operator “=” is used. Just typing the variable by itself at the prompt will print out its value. There is also another form of assignment operator “ <– ”. Like in many other programs, R organises numbers into scalars (a single number, 0–dimensional), vectors (a row of numbers, also called 1-dimensional arrays) and matrices (like a 2-dimensional table). These data types and their assignment are shown in Listing 5.4.

To define a vector, the function `c()` (short for concatenate) is used. Matrices are simply 2-dimensional vectors. To define a matrix, the function `matrix()` is used. The argument data specifies which numbers should be in the matrix. Either funcncol is used to specify the number of columns, or funcnrow to specify the number of rows. It is very important to check how the rows and columns are defined since the data vector is decomposed differently. Individual elements of a matrix can be addressed as [row,column]. To select a whole row, the column designator is left empty (and vice versa).

5.1.4 Packages, Libraries, and Help

Sometimes, it is necessary to have additional functionality beyond that offered by the core R library. The `install.packages` function is used to download and install R packages from CRAN repositories or from local files. Sometimes, it is necessary to separately download an external package, since CRAN does not allow undocumented packages. To obtain a list of all installed packages, go to the packages window in RStudio or type `library()` in the console window.

```
User No.;PS 1;PS 2;PS 3
1;1;2;3
2;4;6;5
3;4;3;2
4;6;5;4
5;4;5;3
6;2;6;5
7;2;4;5
8;3;4;4
9;4;6;5
10;4;4;4
```

Listing 5.2: An example CSV file showing a header line with 4 columns and the first 10 lines of data.

```
> read.csv2("ratings.csv", header = TRUE, sep = ";")
  User.No. PS.1 PS.2 PS.3
1        1    1    2    3
2        2    4    6    5
3        3    4    3    2
4        4    6    5    4
...
```

Listing 5.3: Reading external data from a CSV file into R using a German delimiter “;” .

```
> x = 1 # scalar
> x
[1] 1
>
> c(1,2,3) # vector
[1] 1 2 3
>
> mat = matrix( data=c(1,2,3,4,5,6), ncol=3 ) # matrix
> mat
     [,1] [,2] [,3]
[1,]    1    3    5
[2,]    2    4    6
> mat = matrix( data=c(1,2,3,4,5,6), nrow=3 )
> mat
     [,1] [,2]
[1,]    1    4
[2,]    2    5
[3,]    3    6
> mat[,2]
[1] 4 5 6
```

Listing 5.4: Data types in R include scalars, vectors, and matrices.

R provides extensive documentation. For example, entering `?c` or `help(c)`E at the prompt gives documentation about the function `c`.

5.2 Data Preparation

Experimental data such as from questionnaires is often gathered in a spreadsheet file. In preparation for statistical analysis, the data for each individual question should be stored in a separate spreadsheet file to prevent confusion. It is often easier to prepare spreadsheet files than to manipulate data in R to the required format. Most spreadsheet software can export simple tables of data as CSV (comma-separated value) files, which R can then read. An example of such exported data is shown in Listing 5.2.

The process of reading a CSV file is shown in Listing 5.5. Before analysis begins, R's working directory should be changed to the location where the CSV file is located using the function `setwd()`. To read data from a CSV file, the function `read.csv2` is used. Individual parameter sets data is then separated by selecting single columns of the data. To check if the dataset is complete, it is good practice to calculate the mean and standard deviation and compare it with the same calculations inside the spreadsheet.

5.2.1 Normality Check

To check the normality of a dataset, i.e. to see whether is it (sufficiently) close to a normal distribution, a parametric analysis method such as the Shapiro-Wilk test can be used. The built-in function of R is called `shapiro.test(x)` and expects a vector as input. In the example shown in Listing 5.6, the input *parameterset*1 is already in the form of a vector. If the resulting p-value is above 0.05, the dataset can be considered to be normally distributed (parametric).

In the case of a repeated measures design with more than 2 conditions, the following applies. If the data is parametric (i.e. all the datasets are sufficiently normally distributed), it can be analysed using

```
# Set Working Directory
getwd()
setwd("working-directory-path")

# Read in Data
mydata = read.csv2("fb-q1-data.csv", header = TRUE, sep = ";") # ; for German CSV

# Read in Parameter Sets
ps1 = mydata\$PS.1
ps2 = mydata\$PS.2
ps3 = mydata\$PS.3

# Calculate Mean and Standard Deviation of each Parameter Set
ps1_mean = mean(ps1)
ps1_std  = sd(ps1)

ps2_mean = mean(ps2)
ps2_std  = sd(ps2)

ps3_mean = mean(ps3)
ps3_std  = sd(ps3)
```

Listing 5.5: Reading in questionnaire data from a CSV file.

```
# Check each dataset for normality
ps_normcheck = shapiro.test(parameterset1)
if (ps_normcheck[2] > 0.05) {
    ps_norm = 1   # normal distribution
  } else {
    ps_norm = 0   # no normal distribution
  }

# Other normality checks
# ad.test(ps)
# pearson.test(ps)
```

Listing 5.6: Checking a dataset for normality in R using the Shapiro-Wilk test.

```
if (ps1_norm && ps2_norm && ps3_norm)
  {
    # Parametric case --> one-way repeated measures ANOVA
  }
else
  {
    # Non-Parametric case --> Friedman's test
  }
```

Listing 5.7: In the case of a repeated measures study with more than two conditions, if all datasets are (sufficiently) normally distributed, then a parametric analysis method may be used.

```
# Structure data for ANOVA input
data <- data.frame(
  Ratings = c(cond1,cond2,cond3),
  Groups = factor(c(rep(c("Cond1"),20), rep(c("Cond2"),20)), rep(c("Cond3"),20
))),
  Users = factor(rep(1:20,3)) )

# Parametric case --> one-way repeated measures ANOVA
# use anova(object) to test the omnibus hypothesis
# Is there a significant difference amongst the condition means?
aov.out = aov(Ratings ~ Groups + Error(Users/Groups), data=data)
summary(aov.out)
```

Listing 5.8: The R function `aov` performs ANOVA and expects a data frame with a specific structure as input.

ANOVA. If it is non-parametric, Friedman's Test must be used. This distinction is done in R via an `if` clause, where the results of corresponding Shapiro-Wilk tests are used, as shown in Listing 5.7.

5.3 Statistical Analysis

Many of the methods for statistical analysis which were described in Chapter 4 are included in the base package of R. The following sections show how the different methods are installed if necessary, and are then used and interpreted.

5.3.1 One-Way Repeated Measures ANOVA

R supports multiple ways to perform ANOVA [Quick, 2016; King, 2014]. For example, the simple `aov` function which takes the values and groups vectors as an input. This can also be extended with the error between the users and groups. The command is shown in Listing 5.8 and the output of the analysis is shown in the Listing 5.9.

```
Error: Users
          Df Sum Sq Mean Sq F value Pr(>F)
Residuals 39  155.9   3.998

Error: Users:Groups
          Df Sum Sq Mean Sq F value  Pr(>F)
Groups     2  19.12   9.558   6.806 0.00189 **
Residuals 78 109.55   1.404
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Listing 5.9: The output from the aov function in Listing 5.8 shows the residuals and the p- value indicating whether any statistically significant difference is present between any of the groups.

```
modAB <- lme( Ratings ~ Groups ,random = ~1 | Groups / Users)
summary(modAB)
```

Listing 5.10: Linear Mixed Effects Model using the same input as Listing 5.9.

5.3.2 Mixed Model

The commonly used mixed model approach is implemented in an external R library called nlme. Its usage is very similar to the ANOVA approach with the `aov` function [Pinheiro and Bates, 2000]. The `lme` function expects the ratings and group vectors as input with the groups to user error in addition. The execution of the `lme` function is shown in Listing 5.10.

5.3.3 Friedman Test

The Friedman test is executed via the `friedman.test` function. It uses either a numeric vector of data values, or a data matrix as an input. In the first case, the groups and blocks must be declared as additional parameters. The vector form previously defined in the ANOVA case can be used directly for the Friedman analysis [Galili, 2010].

The output of the Friedman test is the chi-squared value, the degree of freedom, and the corresponding p-value. In Listing 5.11 the p-value is above 0.05, thereby failing to reject the null hypothesis and concluding that there is not enough evidence in the data to suggest that the medians of the parameter set ratings are significantly different.

5.3.4 Friedman Post Hoc Analysis

If the Friedman test reveals a statistically significant difference between the factors in the dataset, it is necessary to complete a post hoc analysis to determine which factors are significantly different. For this, it is necessary to put the data into another structure, in which the values of each set are combined into a

```
# Non Parametric case --> Friedman's test
> psBind <- cbind(p1, p2, p3)
> friedman.test(psBind)

  Friedman rank sum test

  data:  psBind
  Friedman chi-squared = 4.2807, df = 2, p-value = 0.1176
}
```

Listing 5.11: Combining all parameter vectors to create the data matrix `psBind` as input for the `friedman.test` function. The output of the Friedman test shows a p value of 0.1176 which indicates no significance at p < 0.05.

single vector. This vector needs to be connected to the Group and User as well. For that reason the `data.frame()` function is used. It creates a matrix with named columns. The function `rep(value,quantity)` is used to duplicate a certain value or vector.

After creating the correct data format, the function `friedman.test.with.post.hoc` is called. Listing 5.12 shows the creation of the data frame and the execution of the post-hoc analysis [Galili, 2010]. The values of Ratings are compared against the Groups for each Users in the dataframe `friedman.data`. An example result of the post-hoc analysis is shown in Listing 5.13.

5.4 Overall Preference

Two methods are available to determine whether differences in preference votes are statistically significant. The classic Pearson's Chi-Squared test [Plackett, 1983] can only determine whether a significant difference exists somewhere in the distinction of votes. The more powerful Sison-Glaz method Sison and Glaz [1995] uses multinomial proportions to determine whether confidence intervals overlap and thus where exactly any significant differences exist.

5.4.1 Sison-Glaz

The Sison-Glaz method is used to calculate simultaneous confidence intervals for multinomial proportions. It uses a function `multinomialCI(dist, acc)` of the MultinomialCI package [MultinomialCI, 2015], where `dist` is a vector of positive integers representing distribution and `acc` is the significance level for the confidence intervals.

Listing 5.14 shows the usage of the function and interpretation of the output. The input for this example was a distribution vector of $[5, 8, 15, 27]$ representing votes for four conditions, C1 to C4. The output shows the calculated confidence intervals. The analysis proceeds by pairwise comparison of the confidence intervals to see whether or not they overlap. The pairings C1–C4 and C2–C4 do *not* overlap, indicating that these differences in vote counts are statistically significant. All other pairings do overlap, indicating the differences are not statistically significant.

```
# restructure dataset
  friedman.data <- data.frame(
      Ratings = c(ps1,ps2,ps3),
      Groups = factor(c(rep(c("PS1"),40), rep(c("PS2"),40), rep(c("PS3"),40))),
      Users = factor(rep(1:40,3))
    )
> friedman.data
    Rateings  Groups  Users
1   5.000     PS1     1
2   5.000     PS1     2
3   4.000     PS1     3
4   4.000     PS1     4
....
41  4.000     PS2     1
42  2.000     PS2     2
43  6.000     PS2     3
44  1.000     PS2     4
....

# start post hoc
  friedman.result = friedman.test.with.post.hoc(Ratings ~ Groups | Users ,
      friedman.data)
  print(friedman.result)
```

Listing 5.12: Restructuring data and then running the Friedman test with post-hoc analysis.

```
Friedman.Test
  Asymptotic General Symmetry Test

data:  Ratings by Groups (PS1, PS2, PS3) stratified by Users
maxT = 1.3887, p-value = 0.3468
alternative hypothesis: two.sided

PostHoc.Test
PS1 - PS3 0.8886148
PS1 - PS2 0.3467845
PS2 - PS3 0.6239320
```

Listing 5.13: An example result of the Friedman post hoc analysis showing the p values of the pairwise comparison showing no statistically significant difference.

```
# Load Required library
if(!require(MultinomialCI))
{
  install.packages("MultinomialCI")
  library(MultinomialCI)
}

# Create Distribution
distribution = c(5,8,15,27)
numberOfDistribution = length(distribution)

# Calculate likelihood niveau of distribution with given accuarcy
likelihoodNiveau = multinomialCI(distribution, 0.05)
likelihoodNiveau = round(likelihoodNiveau, 3)

# Print Result
print(paste("Likelihood Niveau with confidence level of 95% for given distribution:
    "))
print(distribution)
for( i in 1:numberOfDistribution)
{
  print(paste("Distribution ", i, ": with distribution value: ", distribution[i] ,"
      [",  likelihoodNiveau[i,1], ",",likelihoodNiveau[i,2], "]"));
}

    > "Likelihood Niveau with confidence level of 95% for given distribution: "
    > "5 8 15 27"
    > "Distribution  1 : with distribution value:  5  [ 0 , 0.228 ]"
    > "Distribution  2 : with distribution value:  8  [ 0.018 , 0.282 ]"
    > "Distribution  3 : with distribution value:  15  [ 0.145 , 0.41 ]"
    > "Distribution  4 : with distribution value:  27  [ 0.364 , 0.628 ]"
```

Listing 5.14: Using the Sison-Glaz method by running the `MultinomialCI` function to calculate simultaneous confidence intervals. The output is structured to a readable format and printed.

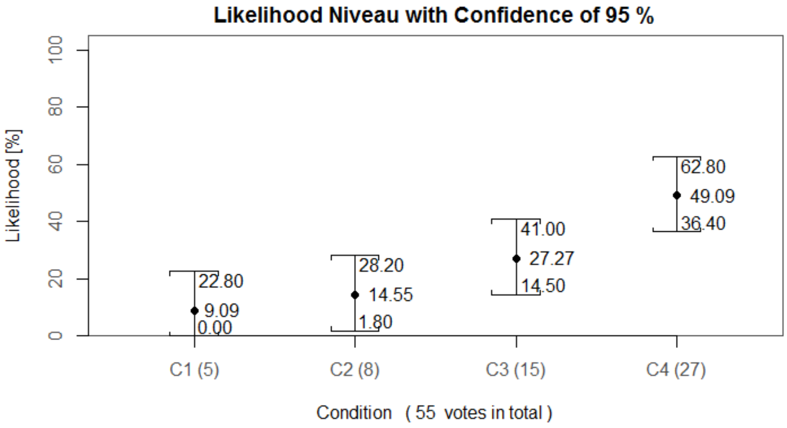


Figure 5.1: Confidence intervals plotted with the `plotMulitnomnalCI` function. The interval for C4 does not overlap with the intervals for either C1 or C2. All other parings do overlap. Hence, only the differences in votes between C4 and C1 and between C4 and C2 are statistically significant.

5.4.2 Plotting Confidence Intervals

The output of the Sison-Glaz method is a set of confidence intervals, represented by a lower boundary and an upper boundary. When comparing multiple distributions, it is often hard to see whether or not intervals overlap. Therefore, the function `plotMultinomialCI` was created. It takes the likelihood niveaus, the distribution, the overlap vector and the accuracy value as input, and plots the intervals graphically. Listings 5.15 and 5.16 show the implementation of the function in detail.

The first part of the function reorders the input data based on the likelihood niveaus. It extracts the values of each distribution and scales the values for better distinction. It then defines the boundaries, axes, labels, and title for the plot. The second part of the function draws the intervals within a for loop. It creates labels for the lower and upper boundary as well as the likelihood niveau. The result of this plotting function is shown in Figure 5.1 using the same distribution of votes for four conditions C1 to C4 as in Listing 5.14, namely $[5, 8, 15, 27]$ and a confidence level of 95%. It is much easier to spot whether intervals overlap when they are plotted graphically.

```
# Read Data
rdt = data.frame(likelihoodNiveau = likelihoodNiveau,
                  distribution = distribution,
                  overlap = overlap,
                  labelx = labelx)
# Reorder Output Values
rdt = rdt[with(rdt,order(distribution)),]

# Read Parameter from Dataframe
likelihoodNiveau[,1] = rdt$likelihoodNiveau.1
likelihoodNiveau[,2] = rdt$likelihoodNiveau.2
distribution = rdt$distribution
overlap = rdt$overlap
labelx = rdt$labelx

# Defining distribution steps for scaling
cond = c(1:length(distribution))

# Defining values of likelihood niveaus
lh = distribution/sum(distribution)*100
lhmin = likelihoodNiveau[,1]*100
lhmax = likelihoodNiveau[,2]*100

#Start creating Plot
plot.new()

# Defining Margin of Plot c(bottom, left, top, right)
par(mar=c(5,5, 2, 2))
plot.window(xlim = c(1, (numel*2 +1)), ylim = c(0, 105), yaxs = "i")
box(col="grey30")

# create axis values
axis(1, at = cond*2, labels = labelx, col.axis = "grey30")
axis(2, col.axis = "grey30")

# create title and axis descriptions
title(main=paste("Likelihood Niveau with Confidence of",accuracy,"%"),
      xlab=paste("Condition (",sum(distribution)," votes in total)"),
      ylab="Likelihood [%]")
```

Listing 5.15: Function `plotMultinomialCI` Part 1. Firstly, the input data is reordered. Next, the values of each distribution are extracted. Then, boundaries, axes, labels and a title for the plot are defined.

```
# create likelihood boxes for each distribution
for( i in 1:length(distribution))
{
  # define linetype and colour
  linetype = "solid"
  colour = "black"

  # draw points for distribution
  points(cond[i]*os,lh[i], pch=19, col=colour)

  # define y range
  miny = lhmin[i]
  maxy = lhmax[i]
  x = i*os # calculate x position

  # draw vertical line
  segments(x,       miny, x,       maxy,      col=colour, lty=linetype)

  # draw box edges bottom
  segments(x-0.02, miny, x+0.02, miny,      col=colour, lty=linetype)
  segments(x-0.02, miny, x-0.02, miny+1.5, col=colour)
  segments(x+0.02, miny, x+0.02, miny+1.5, col=colour)

  # draw box edges top
  segments(x-0.02, maxy, x+0.02, maxy,      col=colour, lty=linetype)
  segments(x-0.02, maxy, x-0.02, maxy-1.5, col=colour)
  segments(x+0.02, maxy, x+0.02, maxy-1.5, col=colour)

  # draw edge values
  if(round(miny,2) != round(lh[i],2))
  {
    text(cond[i]*os - 0.014, lh[i], sprintf("%.2f",lh[i]), pos = 2, cex = 0.8)
    text(x - 0.014, miny + 3, sprintf("%.2f",miny), pos = 2, cex = 0.8)
  }
  # add Value text to the points
  else
  {
    text(cond[i]*os - 0.014, lh[i] + 3, sprintf("%.2f",lh[i]), pos = 2, cex = 0.8)
  }

  # draw center value
  if(round(maxy,2) != round(lh[i],2)) {
    text(x - 0.014, maxy - 3, sprintf("%.2f",maxy), pos = 2, cex = 0.8)
  }

} # end for
```

Listing 5.16: Function `plotMultinomialCI` Part 2. Each interval is plotted as a box. The values of each upper and lower boundary as well as the centre value are added to the plot.

```
# Defining distribution
dist = c(1,5,8,9)

# Calculate Chi squared test
chisq.data = chisq.test(dist)

# Defining accuracy level
pval = 0.05

# Print chi squared result
print(chisq.data)
> Chi-squared test for given probabilities
>
> data:  dist
> X-squared = 6.7391, df = 3, p-value = 0.08069

if(chisq.data\$p.value < pval)
{ print("there is an overall statistically significant effect")
}else
{ print("there is *no* overall statistically significant effect")
}
> "there is *no* overall statistically significant effect"
```

Listing 5.17: Pearson's Chi-Squared Test of the overall preferences. Firstly, the distribution vector is defined, in this case containing four conditions with 1, 5, 8, and 9 votes respectively. Then the test is run and its results were printed. Finally, a message is printed stating whether or not any statistically significant difference is present.

5.4.3 Pearson's Chi-Squared Test

The overall preference result can also be analysed with Pearson's Chi-Squared Test using the function `chisq.test`. The distribution is provided as an input vector and additionally a probabilities vector can be added for comparison. By default, the probabilities are considered to be evenly distributed. Listing 5.17, illustrates the use of the `chisq.test` function.

In the case of a usability experiment, test users are often asked to state a preference. The critical value of the chi-square distribution is defined by the level of significance and the degree of freedom. In Listing 5.17, the level of significance is 0.05 and the degree of freedom is 3 (k-1). After calculating the chi-squared test, if the p-value is below 0.05, a statistically significant difference is present somewhere in the preference vote distribution. In this example the p value is 0.08069, hence there is no statistically significant difference.

Chapter 6

Comparative Study 1 (CS1) - Tiled versus Full Screen View Mode

This first study will be referred to as Comparative Study 1 (CS1). Two different view modes, tiled and full screen, were compared within a slow turn-taking scenario. The study used a repeated measures design with one independent measure, the view mode and had 16 participants. A previous, unpublished study showed that certain view modes are more suitable for particular scenarios. The goal of this study was to see whether this hypothesis holds true in a slow turn-taking scenario.

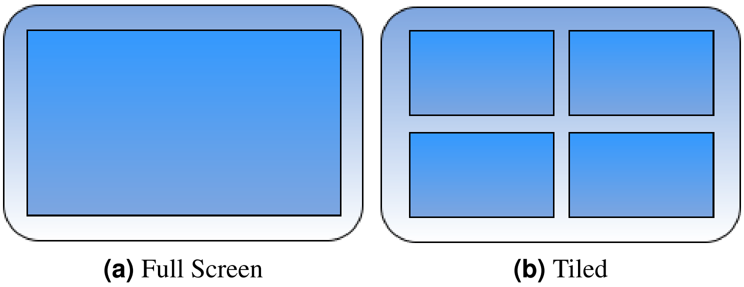
6.1 Motivation and Focus

Group video conferencing tools should accommodate multiple view windows of different sizes, each individually arrangeable, in order to make the conversation as efficient and enjoyable as possible. A view mode (also known as a layout) is a pre-determined composition of 1 to n regions laid out on the screen. The regions of the view mode define the size and the spatial arrangement of the video streams. Figure 6.1 shows the two view modes compared in CS1 full screen and tiled. The view modes are explained in detail in Section 2.3.

Each view mode was thought to be more or less suitable depending on the scenario. In some cases, it may be best to see only one participant in full screen. However, in other situations, it might be better to have an overview of all the (other) participants at once. As conversation roles change, it is likely that video streams may be switched between windows. Indeed, the layout itself may have to change from time to time, as the communication context changes. Orchestration is the process that decides where and how content from different sources should displayed, depending on the conversation context.

The main purpose of the experiment is the comparison of tiled and full screen view modes, in the context of a slow turn-talking scenario. The scenarios used were designed to induce slow turn-taking.

It was postulated is that in a slow turn-taking situation, where people communicate more formally and do not interrupt others very much, full screen view mode is more suitable than the tiled view mode. A previous unpublished experiment compared communication experiences in the three main view modes: full screen, hybrid , and tiled. Based on cluster analysis, the initial conclusion suggested that tiled view mode is more suitable for fast turn-taking conversations, since it provides a more general view and thus better group cohesion. Tiled view mode also provides better conveyance of the individuals presence, since the own video stream is displayed as well. It is also more suitable since it gives a better group telepresence, since multiple participants are visible at once. Full screen view mode was perceived to be better in slow turn-taking conversations, in which one person holds a monologue and the audience closely listens. Listeners have a better view of the speaker's facial expressions. Hybrid view mode is a compromise between the two previous cases, supporting both slow turn-taking conversations as well



(a) Full Screen

(b) Tiled

Figure 6.1: The two view modes compared in the study.

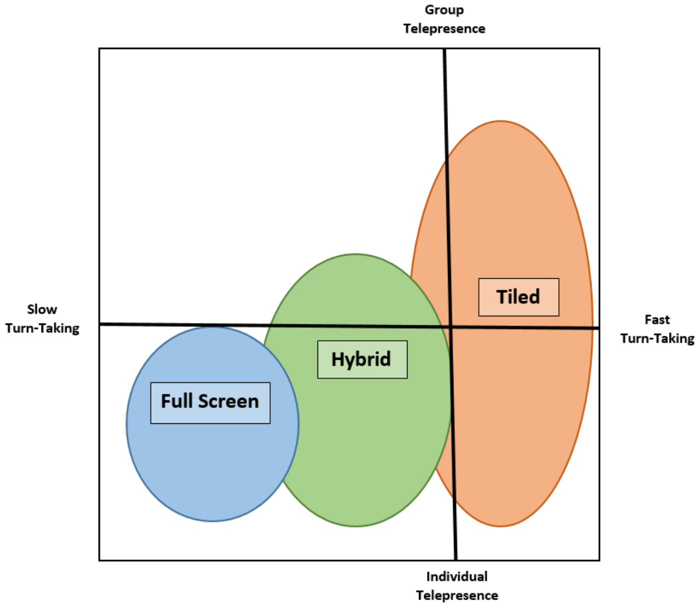


Figure 6.2: The design space of view modes according to a previous, unpublished study. The results suggested that tiled view mode is more suitable in a fast turn-taking situation. Full screen view mode is more suitable for slow turn-taking situations, where one person is speaking for a longer period of time. [Image redrawn from unpublished work of Erik Geelhoed under the terms of Austrian Copyright Law [BKA, 2015, § 42f].]

as fast turn-taking conversations. These conclusions were represented in the design space shown in Figure 6.2.

The focus of the study was chosen to be on a slow turn-taking scenario, because experiments with fast turn-taking have already been conducted elsewhere in the Vconect project [Geelhoed et al., 2014]. The tasks were designed such that crosstalking should be minimal during the experiment, except for back channelling (“mhm”, “ok”, “ya”, “aha”, etc.) to the current speaker, which is necessary to keep in touch with the listeners as described in Section 3.1.11. Inducing slow turn-taking behaviour is dependent on the user’s experience with video conferencing tools, as well as social manners and conduct. The participants were mostly university students and were unfamiliar with formal conversations, so they were given further instruction. It was pointed out that they should not interrupt an active speaker in the middle of their sentences. They were still encouraged to contribute to the discussion as much as they could and to try to focus on the interaction.

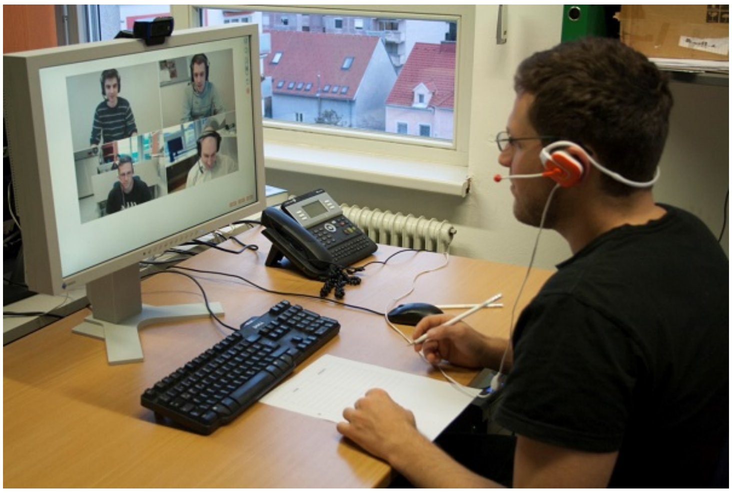


Figure 6.3: The tiled view mode used in CS1.

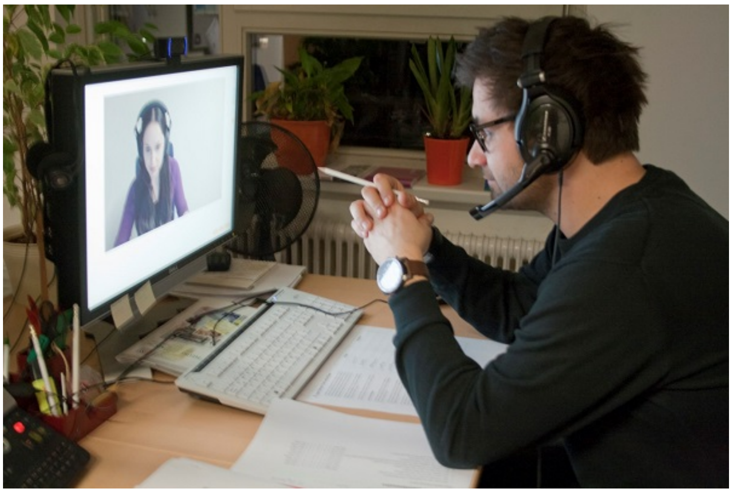


Figure 6.4: The full screen view mode used in CS1.

6.2 View Modes

In CS1, the view modes tiled and full screen were compared in slow turn-taking scenarios in a counterbalanced 2 × 2 repeated measures design. In the first condition, tiled view mode participants could see each other and themselves in a mosaic of tiles, arranged in two rows of two equally sized tiles as shown in Figure 6.3. This view mode is thought to afford good group awareness, including being able to identify where any vocal back channels originated. However, tiled view mode might be less effective at providing feedback through facial expressions. The video composition is static and no orchestration happens, i.e. no matter who is speaking, the screen always shows the same layout.

In the second condition, full screen view mode, participants could only see the active speaker as a full screen image, as shown in Figure 6.4. Orchestration (switching between active speakers) based on significant voice activity was activated, as described in Section 3.1.8. This condition is thought to maximise telepresence and enable better recognition of facial expressions of the current speaker, but group awareness might suffer.

The main problem when testing view modes is the personal preference of users themselves. One user may prefer to see all other participants in a conversation, whereas another may prefer to see only one or even no other members, which is like being in a telephone conference call. Seeing other participants in a video conference call gives the speaker the opportunity to see the reaction of their counterparts, like in a group face-to-face discussion.

6.3 Task Descriptions

In each of the two conditions, participants were asked to perform a short video conference task of about ten minutes in groups of four. After the sessions, they were asked to fill out a short questionnaire, which is described in Section 6.8. Two equivalent tasks were prepared, and divided into two parts. The first part was intended to create an open atmosphere where every participant can bring in their ideas. This helps to establish a conversation basis on which it is possible to evaluate the system behaviour. In the second part, participants were asked to rank and order the previously defined points. However, to encourage slow turn-taking, participants were asked not to interrupt another speaker and to keep the discussion formal. It is necessary to have two equivalent tasks for each part of the study to create comparable results. The two equivalent tasks were:

- **Task 1: An Ideal Holiday.** In the first part of the task, participants were given five minutes to generate a list of at least seven items describing what constitutes an ideal holiday. For the second part, participants were given another five minutes in which to prioritise the seven items in a collaborative effort. Number one in the list should be the most important and number seven the least important. Of course, the four participants might not immediately agree on what is most and least important, but they were asked to arrive at an acceptable compromise, through a formal discussion.
- **Taks2: An Ideal Home.** Analogous to Task 1, but items relating to an ideal home.

It was difficult to come up with two tasks which might lead to recurring behaviour within each group and still constitute slow turn-taking behaviour. Since every group is different, basic topics were chosen where everybody has an opinion and can talk about without any preparation. It was also necessary to keep the testing time short, so that participants can still distinguish between the two view modes, after both tasks are completed.

The groups had to self-control their behaviour. The facilitator only interfered if there were technical problems or when the designated time period had expired. A moderation of the conversations would have changed the whole situation and lost the feeling of a private social conversation. It would probably also lead to an increase in attention towards the moderator, detracting attention from the other participants.

6.4 Participants

A total of 16 volunteers, 5 females and 11 males with a median age of 25 years took part in the experiment. Figure 6.5 shows the age distribution of the participants. Twelve participants were students, the other four were employees of various companies. Two thirds of the participants have a higher degree at Bachelor or Master's level. Participants have an average of 13.25 years experience with computers, meaning nearly all participants used a computer for half of their lives or more. On average, participants use a computer around 29 hours per week. In terms of operating system, twelve participants mainly use Microsoft Windows, followed by three Mac user and one Linux user.

Participants use a social network more often than a video conferencing tool. On a scale from 0 to 6 where 6 is most often, average social network usage lies at 5 with median at 5.5. In contrast, video conferencing usage is on average 2.3 with median 1.5. This distribution can be seen in Figure 6.6. The most common application for video conferencing or chatting was Skype [VS, 2015], which was used by

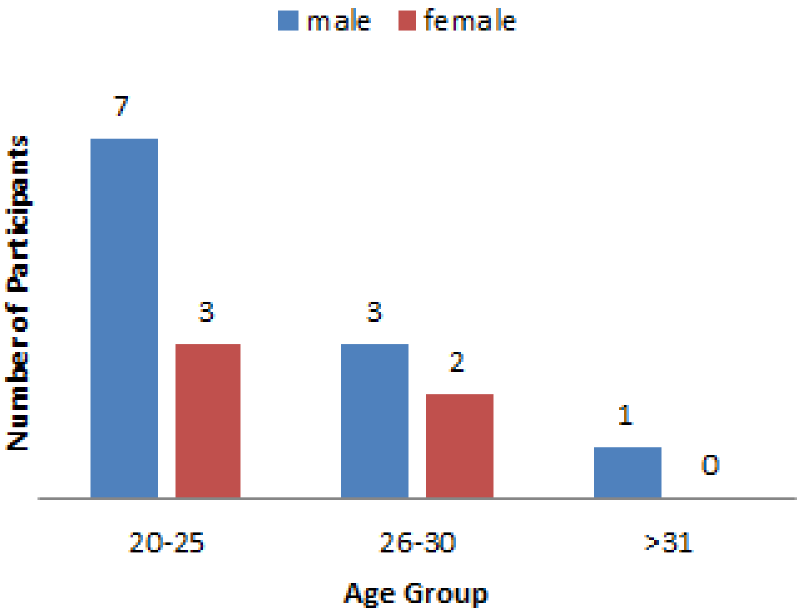


Figure 6.5: Participants by age group and gender.

13 participants. Video conferencing tools are rarely used compared to social networks, but are still widely known. Out of 16 participants, only one person was not familiar with any specific video conferencing tool. Every participant was familiar with the concept of video conferencing.

The participants within each group nearly always knew one another. There was only one participant who did not know any of the other three group members. This provided a good basis for communication and enabled participants to talk freely without inhibitions towards other participants in the session. For full screen view mode, it is crucial that every participant contributes to the conversation, so that every stream can be incorporated into the decision making and cutting behaviour.

All participants signed a consent form that photos and videos of the experiment can be used for research purposes. The consent form can be found in Section A.2.

6.5 Experimental Setup

The experiment was set up in the office building of JOANNEUM RESEARCH in Graz. Four separate rooms were connected via a webcam and microphone using the Vconect video-conferencing plugin (VClient) for the Chrome web browser, installed on a PC running Windows 7 in each of the four rooms. All four rooms were connected by gigabit LAN, and a connection to the Internet was available. Each user had a headset and a HD webcam. An illustration of the setup is shown in Figure 6.7.

All sessions were recorded on each computer via Camtasia Recorder 8 which was preinstalled and set up to use the XViD Codec at 1080p and 25 frames per second. Audio was recorded separately, because the VConect Plugin blocks all other audio channels and it was not possible to record it on the same PC. Therefore, an additional invisible client named ACE Client recorded the combined audio from all the rooms. This output was then edited and cut with Audacity due to different time offsets within each audio file. Figure 6.8 shows the recording setup of all the participants in one session and the additional audio client.

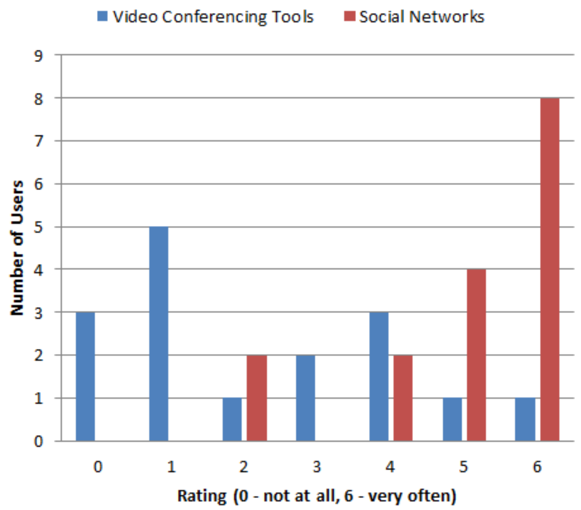


Figure 6.6: Usage of video conferencing tools and social networks among the 16 test participants.

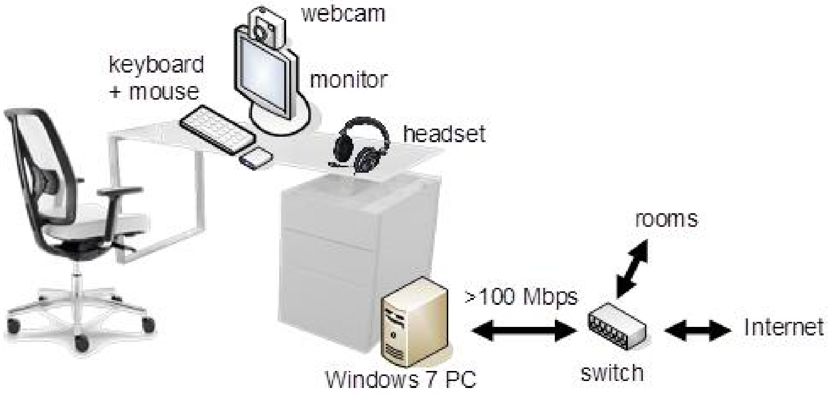


Figure 6.7: In each of the four rooms, the participant had an experimental setup like this one.

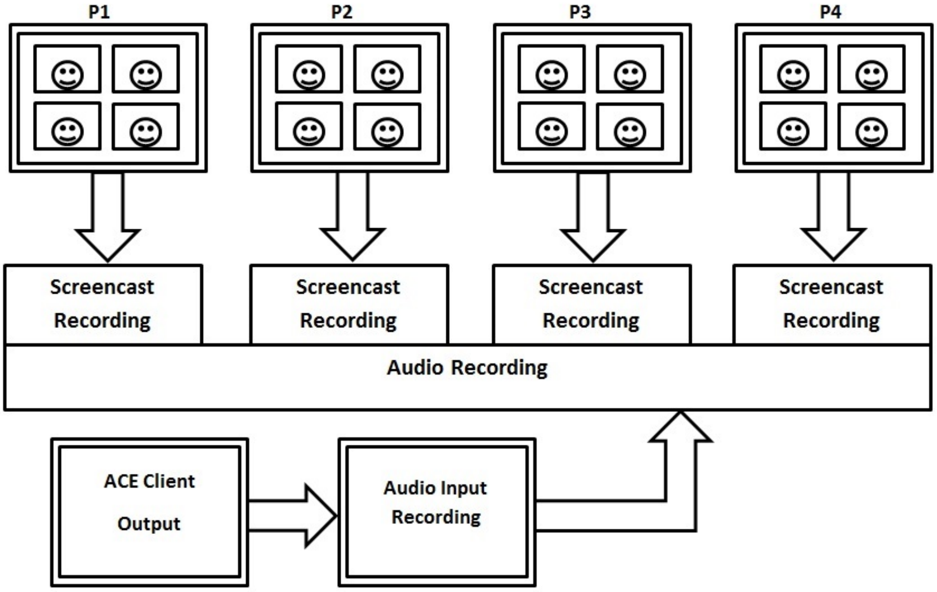


Figure 6.8: The recording setup for experiment CS1. The display of each participant (P1-P4) was screen-captured and the combined audio was recorded through an additional invisible client (ACE Client).

6.6 Schedule

An internal trial run was held on 14 Jan 2014 with JOANNEUM RESEARCH staff members. The main experiment took place on 17 Jan 2014 with the first group starting at 16:00 and the other three groups at hourly intervals thereafter. The procedure for each session was as follows:

- Introduction and background questionnaire (5–10 mins).
- First task, create list items (10 mins).
- Feedback questionnaire (5 mins).
- Second task, prioritise list items (10 mins).
- Feedback questionnaire (5 mins).
- Group discussion (5–10 mins).

Each session was designed to take 40–50 minutes in total and leave around 10 minutes preparation time before the next group arrived.

6.7 Experimental Design

At first, participants received a short introduction into the experiment and were asked to fill out a background questionnaire about their personal connection to the other participants and their knowledge of video conferencing tools. Two of the four groups started in full screen view mode, the other two in tiled view mode. After each videoconferencing task, the participants were asked to rate the view mode they had just used. The counterbalancing of view mode for each session is shown in Table 6.1. After the session, each group was debriefed, and the debriefing session was recorded with a video camera.

	Task 1	**Task 2**
G1	Tiled	Full screen
G2	Full screen	Tiled
G3	Tiled	Full screen
G4	Full screen	Tiled

Table 6.1: Experimental design of CS1. The presentation order of the two view modes was counterbalanced to prevent bias.

Number	**Question**
Q1	How easy was it to keep track of the discussion?
Q2	How well did you feel you came across to the group?
Q3	How well could you see who was talking?
Q4	How active were the other people?
Q5	How close did you feel to the other people?
Q6	How well did you see the facial expressions of other people?
Q7	How often did it happen that you and someone else started talking at the same time?
Q8	How often were there awkward silences?
Q9	How lively were the discussions?
Q10	How easy was it to contribute to the discussion?

Table 6.2: The ten questions used for ratings in CS1.

6.8 Feedback Questionnaire

Immediately after using each view mode, participants were asked to rate the view mode according to ten criteria on a seven point scale, using a paper questionnaire. The ten criteria (questions) are listed in Table 6.2. The original questionnaire can be found in Appendix A.4. The questions were presented to the participants in the form of semantic differentials along a scale [3 2 1 0 1 2 3], which were then converted internally to points from 0 (worst) to 6 (best) for analysis.

6.9 Subjective Measures

Three kinds of subjective measures were collected during the study.

6.9.1 Group Debriefings

The group debriefings indicated that a majority of users would like to have a combination of both view modes. A summary of statements can be found in Table 6.3. A few users discovered the hybrid view mode accidentally while restarting or switching session rooms. One participant particularly mentioned that he would have preferred this solution over both others.

6.9.2 Ratings for Each Condition on Ten Criteria

After each task, participants filled out a short feedback questionnaire with ratings about the view mode they had just used. The results are shown in Table 6.4. For both conditions, the values of Q7 and Q 8 are rather negative compared to the other questions. These two aspects are somewhat related, since it often

Session	Feedback
1	The full screen was sometimes really lagging and also reacted on some small side noises although nothing was said. Therefore you stop watching the video and just talk. It is also not good if you cannot see yourself, because you never know what the others might see. It is good with four people because you can see yourself and the others at the same time. Also it gives the possibility to vote via signs or such. Should also be possible with more participants. Additionally it may be good to mark the speakers with a red border. Also, maybe a selection could be made which persons should be displayed bigger in a larger group. More interaction would be nice. Full screen loses some of the functionality of the video conference because you just can see one person and the others are kind of left behind.
2	Full screen seemed a bit random. Just flipping between people and no focus on the active speaker. At some point I gave up figuring out which rhythm it was and so I just focused on my list. Tiled: I was more confidant to speak when I could see all of the other 3 and see if they were about to speak. I could also see how the others react to what I was saying or if they were reacting at all. I just felt like more of a group discussion. More alive! I was able to see myself full screen: I could see the other persons better. I had no problem when I could not see myself. The Hybrid Mode that I just saw in the beginning would also be nice I think. A middle solution would be most adaptable. Seeing everybody in a discussion and also laying the focus on the current speaker.
3	Tiled: You had the feeling that everyone is part of the discussion rather than just one person. More like a real life discussion. It was easier to follow. In a formal business discussion it may be better to see just the head of the discussion and who is talking but with a conference with friends it is more natural to see all of the other participants. In full screen you knew who was talking because you knew the voices of the others but often the video was showing someone else. It changed slightly too fast but it was still easy to follow. It had a little bit of a delay in switching to the speaker. The design of the whole thing should be developed (make it prettier) A writing option would also be nice. Maybe it would be nice to choose who is supposed to be big. A disadvantage would be that if the speaker switches a lot you also have to switch the view.
4	Full screen: You are more attentive. You feel closer to the other persons and you could understand them better. The task was sometimes distracting from the video conference itself. Also you are more distracted by the smaller windows because you always try to see how the others react. I was really amazed about the "orchestration" that it could really detect who was talking. The facial expression were very easy to see and it felt more like a real conversation. Tiled: It shows more of the group and you can see who was talking. The full screen had a fast switching behaviour and it was often delayed.

Table 6.3: Combined views of the users gathered during the group debriefing sessions.

Question	Tiled		Full Screen		ANOVA	Friedman
	Avg	Std Dev	Avg	Std Dev	p	p
Q1 - How easy was it to keep track of the discussion?	5.19	0.83	4.81	1.17	0.211	0.256
Q2 - How well did you feel you came across to the group?	5.00	1.21	4.44	1.63	0.132	0.317
Q3 - How well could you see who was talking?	4.00	1.83	3.50	1.55	0.333	0.527
Q4 - How active were the other people?	4.44	1.15	4.88	1.02	0.186	0.317
Q5 - How close did you feel to the other people?	4.31	1.58	3.88	1.59	0.410	0.563
Q6 - How well did you see the facial expressions of other people?	3.50	2.07	4.25	1.44	0.266	0.248
Q7 - How often did it happen that you and someone else started talking at the same time?	2.63	1.59	2.13	1.41	0.116	0.157
Q8 - How often were there awkward silences?	2.00	1.55	2.13	1.41	0.750	0.738
Q9 - How lively were the discussions?	3.88	1.75	4.44	1.26	0.069	0.157
Q10 - How easy was it to contribute to the discussion?	4.50	1.55	4.63	1.26	0.669	0.738

Table 6.4: Average ratings and standard deviation for the two view modes. Ratings range from 0 (worst) to 6 (best) for each criteria. None of the p values are < 0.05, indicating no statistically significant differences between the two view modes for any of the ratings.

happens that when two participants start talking simultaneously, both stop talking and nobody talks for a while. In both conditions, participants felt that it was easy to keep track of the conversation and that they came across to the group. Overall, the results of the feedback were generally.

There were no statistically significant differences between the two view modes for all ten ratings at a confidence level of 95%.

Listing 6.1 describes the R scripts used for analysis of feedback question ratings. Firstly, the CSV data is read in and stored as a dataframe. Having the data in the correct format the conditions (in this example `mydata\$Full` and `mydata\$Tiled`) can be tested regarding their homogeneity of variances. This is determines if a parametric or a non-parametric test should be used.

In the case of homogeneity of variances, ANOVA and Linear Mixed Models methods can be used as parametric tests. For the ANOVA the `aov()` function is used which is described in detail in Section 5.3.1 and for the Linear Mixed Models method the `lme()` function is used which is described in detail in Section 5.3.2. The both functions are followed by a pairwise t-test with adjusted variances using the bonferroni-holm method described in Section 4.6.

In the case of inhomogeneity of variances, the Friedman test is used as an non-parametric alternative. Here, the function `friedman.test()` is used which is described in detail in Section 5.3.3. After conducting the Friedman test a post hoc analysis needs to be carried out to find out the pairwise differences. This post hoc analysis is described in detail in Section 5.3.4.

```r
# Load necessary libraries
library(car)
library(nlme)

# Read in Data
mydata = read.csv2("fb-q1-data.csv", header = TRUE, sep = ";")
data <- data.frame(
  Ratings = c(mydata$Full, mydata$Tiled),
  Groups = factor(c(rep(c("Full"),size), rep(c("Tiled"),size))),
  Users = factor(rep(1:length(cond1),2)))
attach(data)

# Test homogeneity of variances
y = c(mydata$Full,mydata$Tiled)
conditions = factor(rep(1:2, c(size,size)))
fligner.test(y~conditions)

# If Parametric:
# ANOVA
aov.out = aov(Ratings ~ Groups + Error(Users/Groups), data=data)
summary(aov.out)

# Linear mix model
modAB = lme( Ratings ~ Groups ,random = ~1 | Groups / Users)
summary(modAB)

# Pairwise t-tests with adjusted p-values
pairwise.t.test(Ratings, Groups, p.adjust.method="holm", paired=T)

# If Non-Parametric:
# Friedmans test
friedman.test(cbind(mydata$Full, mydata$Tiled))

# Run post hoc analysis
source("func__post-hoc-friedmans-test.R")
friedman.result = friedman.test.with.post.hoc(Ratings ~ Groups | Users , data)
```

Listing 6.1: R code used to analyse ratings. Firstly, read data from CSV and creating dataframe. Secondly, perform analysis of variances to decide whether parametric or non-parametric tests should be use. Further, ANOVA and Linear Mixed Model are calculated as parametric tests as well as Friedman and corresponding post hoc methods as non-parametric test.

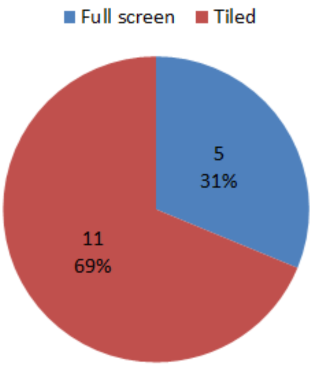


Figure 6.9: Overall preference for view mode. 11 out of 16 participants preferred the tiled view mode.

```
# Define vote counts
distribution = c(5,11)
label = c("Full Screen (5)", "Tiled (11)")

# Calculate statistical preferences
pvalue = 0.05 # 95\% confidence
likelihoodNiveau = multinomialCI(distribution, pvalue)
likelihoodNiveau = round(likelihoodNiveau, 3)
print(likelihoodNiveau)
>        [,1]  [,2]
> [1,] 0.125 0.535
> [2,] 0.500 0.910

# Plot statistical preferences
source("func__plotMultinomialCI.R")
plotMultinomialCI(likelihoodNiveau, dist, , 95, label)
```

Listing 6.2: R code used to analyse overall preferences with a confidence of 95%.

6.9.3 Overall Preference

Participants were also asked which of the two view modes they preferred overall. Out of 16 participants 11 (69%) voted for tiled and 5 (31%) for full screen, as shown in Figure 6.9. However, this difference is not statistically significant, as is explained below. Every participant was asked to elaborate on their decision. The results are shown in Table 6.5. No participant was indecisive about the view mode, but one participant declined to elaborate on his decision.

Listing 6.2 describes the R scripts used for analysis of overall preferences. Firstly, the distribution vector as well as the label of the overall preference votes is created. The p value for the corresponding confidence is set and the multinomial confidence intervals, using the Sison-Glaz method (described in Section 5.4.1), are calculated and printed. Finally, the confidence intervals are ploted using the `plotMultinomialCI` described in Section 5.4.2.

In Figure 6.10, the likelihood niveau of each view mode is displayed. The two ranges overlap and thus the difference in preference is not statistically significant. However, the intersection is relatively

Session	Participant	Preferred View Mode	Comment
1	1	Tiled	You can keep an eye on everybody, no irritating switches and pixelization. Easy voting by hand signals, immediate overview on everybody and also on reactions of my own. At the end not even look at the display necessary.
1	2	Tiled	Switch-over caused a massive retardation and because of the permanent to and fro switching, you stop watching the video stream. If you can keep an eye on all four persons, you have a better overview concerning the group and you also can watch the reactions of the people, who are not speaking in the moment. This helps to make communication split fair and open.
1	3	Tiled	I preferred the first mode (tiled), because I had an immediate overview on all participants. Furthermore, the second mode was jerking.
1	4	Tiled	Nice to see all persons all the time. Switching on full screen took too long time and then you could not always watch the right person.
2	5	Full	No comment.
2	6	Tiled	It is easier when you can see all participants and select your contact person or speaker (the automatic change of the pictures is irritating).
2	7	Full	Because so you can concentrate on the essentials.
2	8	Tiled	You could see if somebody wanted to say something and the interruption did not worry. I could watch all non-verbal reactions on what I was saying.
3	9	Tiled	Better clarity in tiled mode, because the to and fro of the speaker became disturbing.
3	10	Tiled	It is much easier to get the feeling, that everybody is chating at the same time.
3	11	Tiled	It is better to see all people, involved in the discussion. When you can see them all you get the feeling of a real discussion with real people.
3	12	Tiled	You get a better feeling for the people, when you can see them. It is also easier to start talking and to follow the discussion.
4	13	Full	Better understanding of what is said. You are more attentive when you do not have all four screens.
4	14	Tiled	The feeling was like speaking in the group. It was easy to identify the speaker. It was more fun to speak.
4	15	Full	You feel closer to the interlocutor and you are more attentive. You cannot see who is speaking in this moment, on mode tiled viewmode.
4	16	Full	You were able to follow the discussion in a better way because you could see the speaking person. Facial expression could be better identified and you felt closer to the others.

Table 6.5: Overall preference expressed by each of the test users.

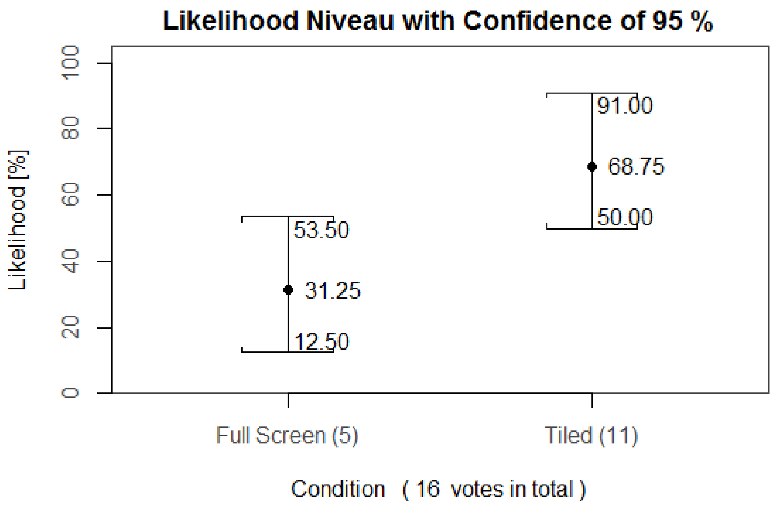


Figure 6.10: Overall preference analysis with confidence of 95% (p <0.05) . The overlap is small but present. Hence, the difference is not statistically significant at a confidence level of 95%.

small. If the confidence level were reduced to 90%, the difference is be statistically significant, as can be seen in Figure 6.11. The two distributions no longer overlap and therefore the difference is statistically significant at a confidence level of 90%.

6.10 Objective Measures

The logging functionality of the tested software version did not work correctly and created overlapping log files. Unfortunately, i was not possible to distinguish when one session finished and the next started. Therefore, there are no objective measures for this study.

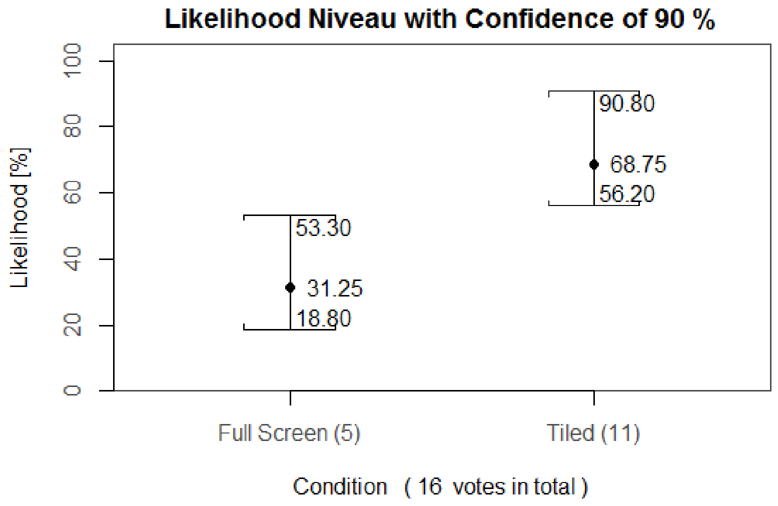


Figure 6.11: Overall preference analysis with confidence of 90% (p <0.1). The gap between the intervals is small but present. Hence, the difference is statistically significant at a confidence level of 90%.

Chapter 7

Comparative Study 2 (CS2) - Start Delay in Full Screen View Mode

In this comparative study, three different start delay settings (300 ms, 600 ms and 900 ms) were compared within full screen view mode. The study used a repeated measures design with one independent variable (start delay) at three levels. These three settings were chosen due to insights from previous evaluations and simulations. The study was held with 40 participants. Each participant rated each of three conditions on 10 different criteria.

7.1 Motivation and Focus

Start delay is one of the main influential factor in orchestration, as described in Section 3.3.1. Start delay is responsible for the designation of voice activity events as being significant voice activity (SVA) events. These designations form the basis for orchestration decisions and cutting behaviour. Keeping the start delay too short would result in too many turn shifts and transitions between displayed speakers. Such fast transitions are hard to follow and could distract from the conversation itself. On the other hand, if the start delay is too long, orchestration decisions would be noticeably delayed, which can be perceived as insensitive. Furthermore, if voice activity is very choppy and the start delay is too short, it could prevent the active speaking person from being recognised by the orchestration.

Thus, it is necessary to determine the most suitable parameter for the start delay, neither too short nor too long, such that orchestration behaviour can be close to optimal.

7.2 Determining Suitable Start Delay Settings

In order to determine a suitable range of values for the start delay parameter, the log files of recorded sessions from previous unpublished studies, conducted within the Vconect project, were analysed.

7.2.1 Log File Analysis of Previous Experiments

The log file of a test session can be replayed by the orchestration system, as if it were a live session. Depending on the parameter configurations, the orchestration behaves differently. For this pre-analysis, a logfile was taken from a view mode experiment with 6 participants at British Telecom from 2013. The start and stop delays were defined to cover a wide range of possibilities. Table 7.1 shows the ten different configurations and the corresponding analysis results. The main focus of the analysis was the detection of significant voice activity events, which influence the director's cutting behaviour. The relevant part of the log files containing conversations comprised about 38 minutes with 6901 voice activity events.

Parameter Set	Delay (ms)		SVA	Event Duration (ms)						Combined Duration	
	Start	Stop	Events (#)	min	Q1	median	avg	Q3	max	(ms)	(min)
PS1	400	400	2595	402	716	1227	1640	2066	15928	4256080	70 .93
PS2	800	800	1332	803	1297	2067	2961	3723	26315	3944665	65.74
PS3	1200	1200	781	1202	1750	3061	4454	5429	30935	3478315	57.94
PS4	400	800	2039	802	1259	1995	2823	3461	26694	5756458	95.94
PS5	800	1200	1168	1202	1803	2892	4323	5290	31335	5048883	84.15
PS6	800	400	1565	402	773	1248	1724	2143	20958	2698716	44 .98
PS7	1200	800	867	802	1232	2020	3019	3852	25906	2617624	43.63
PS8	2000	1000	322	1005	1665	3056	4170	5379	24386	1342611	22.38
PS9	1500	800	585	804	1383	2390	3306	4231	22815	1934184	32.24
PS10	1200	400	866	802	1236	2021	3018	3857	25913	2613743	43.56

Table 7.1: Start delay and stop delay analysis of a previous, unpublished view mode experiment within the Vconect project.

The output of the orchestration is the internal timestamp for the start and end of each SVA event. Further analysis is based on these timestamp values. The duration of a SVA event is calculated by subtracting the beginning from the end, giving a duration in milliseconds. The logic of the orchestrator is based on a continuous stream of events. The output SVA is shifted due to the start delay, so the beginning of an SVA event is not the true beginning of the person speaking. Similarly, the stop delay extends every SVA event by its own value. For example, a voice activity event starts at 300 ms and ends at 900 ms. With a parameter set of 400 ms start delay and 800 ms stop delay, the SVA would begin at 700 ms and end at 1700 ms, and its duration would be 1000 ms rather than the 600 ms of the real voice activity.

A brief examination of Table 7.1 shows that the number and duration of SVA events depends on the configured start delay and stop delay. For example, consider parameter sets PS1 and PS6, where only the start delay is different. The stop delay is 400 ms in both cases. Increasing the start delay from 400 to 800 ms, decreases the number of SVA events from 2595 to 1565. It also shows that there are many short voice activities which are not significant with a longer start delay. The system can no longer distinguish between a short drop of voice activity or background noise. By cutting more than one third of SVA events, the average SVA duration rises from 1640 ms to 1724 ms, which indicates that the majority of SVA events are longer than 800 ms. The total duration of SVA events drops from 70.93 minutes down to 44.98 minutes. The length of the total duration of SVA is longer than the recording length, because it sums up the event lengths of all 6 users together (the maximum would be 6 times the log file length if all participants spoke the entire time).

Another example is to compare parameter sets PS1 and PS4, where the start delay is fixed to 400 ms and the stop delay varies from 400 ms to 800 ms. A longer stop delay typically reduces the number of SVA events, because it may merge consecutive voice activity into a single SVA event. In this particular comparison, the number of SVA events does indeed decrease by 559 from 2595 to 2039 events. The total duration of SVA increases from 70.93 minutes to 95.94 minutes and the average duration per SVA event increases from 1640 ms to 2823 ms.

Considering Table 7.1 alone, it was not possible to find a suitable set of start delay parameters for experiment CS2, since the differences are too big. Therefore, an R script was implemented which simulates the orchestration behaviour based on the input log file.

7.2.2 Exploration of Test Parameters with R Script

The R script simulates orchestration behaviour by using voice activity to detect SVA events based on certain start and stop delay parameters. Furthermore, it makes it possible to analyse the influence of SVA detection on other conversation metrics like crosstalk and simultaneous start. Since the real orchestration rule engine is based on real-time event streaming, its output can differ slightly from the script. However, the influence of start and stop delay is largely unaffected by this mismatch.

With the script, it is possible to see the influence in a continuous manner, rather than the discreet example sets of Table 7.1. The result of the script is shown in the Figure 7.1. Each line represents the behaviour of the orchestration depending on a fixed stop delay and a varying start delay. Both factors were calculated from 10 ms to 1000 ms in 10 ms steps. For better display, the step size of the stop delay is plotted at 50 ms intervals. Start delay values above 1000ms do not have much influence on the number of significant voice activities, since audio recordings are fragmentary even if a person is talking constantly, as was described in Section 3.3. For this reason, all values higher than 1000 ms were eliminated as possible test parameter values. For stop delay values above around 900 ms, the curves become very similar.

The relationship between start delay and stop delay can be explored in Figure 7.1. For both low start and low stop delay, the number of significant voice activities is close to the number of voice activities. Slight changes, not bigger than 50 ms, do not influence the system at first. By increasing the start and stop delay to 100 ms, the number of SVA events drops around 30% from nearly 7000 down to 5000. Increasing the parameters to 200 ms each, the number of SVA events decreases down to under 3500. This is only half of the number of input voice activity events. The reduction seems rigorous, but the higher stop delay merges multiple voice activity events into single ones and filters out very small voice activities which are more likely to be background noise. These 200 ms can be seen as lower bound for the start delay parameter. If the start delay is less than 200 ms, the number of significant voice activities would be too high for the director to follow, and would result in too many cuts.

For the CS2 study, start delay values of 300 ms, 600 ms and 900 ms were chosen. 300 ms and 900 ms are at the lower and higher ends of the scale, but are still reasonable settings. 600 ms is in the middle and a good trade-off between the two endpoints. For all three start delay settings, the stop delay for CS2 was set to 600 ms. Varying the stop delay as well would lead to inseparable effects.

7.3 Task Descriptions

The tasks for CS2 build upon the tasks that were used for CS1 described in Chapter 6. For CS2, however, three tasks were needed, one for each experimental condition. In addition, the tasks were shortened to reduce the length of each session. Since the number of task per group increased from two to three, short and concise tasks were needed, which allowed participants to start a discussion right away without any long preparation time.

In each session, four participants completed three group discussion tasks using the videoconferencing system. They had around 7 minutes to discuss each topic. The three discussion topics were:

- An Ideal Home.
- An Ideal Vacation.
- An Ideal Job.

Each session used the same sequencing of topics. There was no permutation or counterbalancing of topics. The choice of topic can potentially influence speaking behaviour, because people might talk more or less about a certain topic. Fully counterbalancing the presentation order of both topics and start delay parameters would have resulted in 27 sessions and 108 users instead of the projected 9 sessions and 36 users.

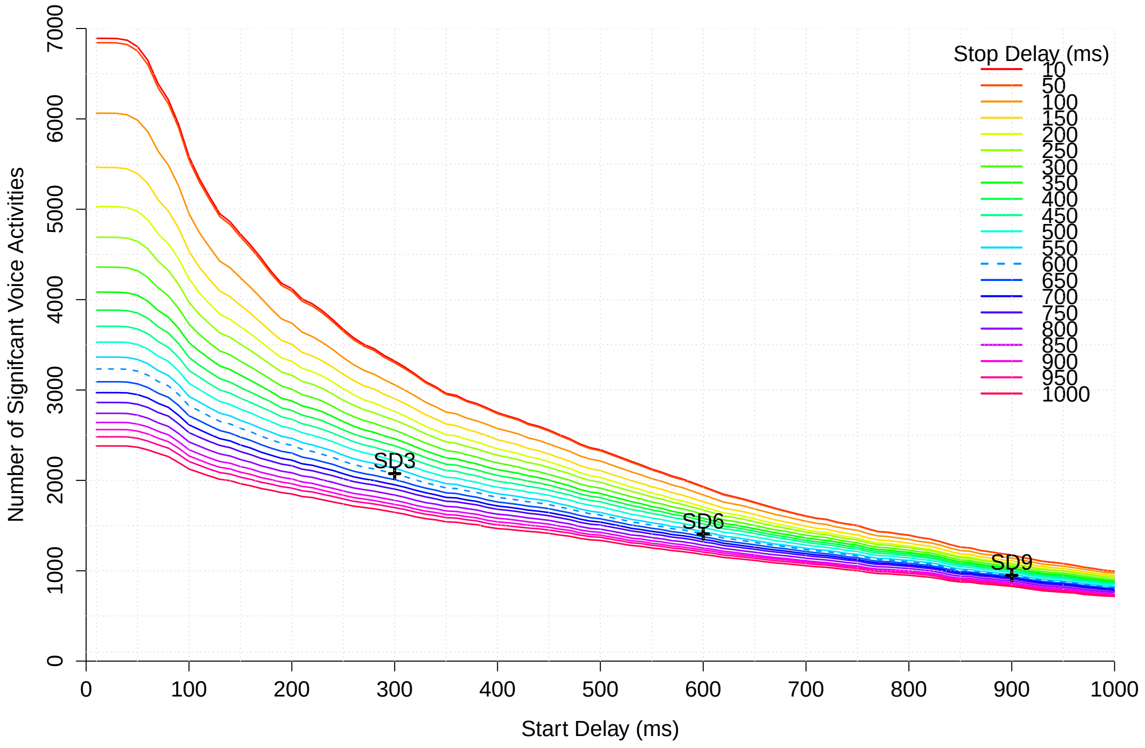


Figure 7.1: The influence of start delay and stop delay on the number of significant voice activity events.

7.4 Experimental Design

The study used a repeated measures design with one independent variable (start delay) at three levels. The start delay parameter was permuted based on a latin square in order to counterbalance the presentation order. The tenth session reused the first ordering. The three test conditions were denoted as SD3, SD6, and SD9 for start delays of 300, 600, and 900 ms respectively. The presentation order used is shown in Table 7.2.

7.5 Participants

In total, 40 persons took part in this comparative study, 28 male and 12 female. As Figure 7.2 shows, 28 participants were older than 40 years of age, with a median of 36.5 years.

Compared to the first comparative study CS1, the median age of the participants is about 10 years higher. CS1 was conducted mainly with university students, whereas in CS2 the majority of people were recruited from JOANNEUM RESEARCH. Participants were predominantly (37) users of Microsoft Windows, with two Unix users and one Mac user, with an average of 22.8 years of computer experience. Education level was not part of the questionnaire, but due to the affiliation with a research organisation, most participants had a higher education level with at least a bachelor's degree.

Participants used video conferencing tools about the same as social networks. On a scale of 0 (not at all) to 6 (very often), average usage of video conferencing tools was rated at 2.4 and average usage of social network at 2.5, neither of which is particularly high. The distribution is shown in Figure 7.3. In CS1, the usage of social networks was significantly higher than video conferencing tools.

Set	Condition 1	Condition 2	Condition 3
S1	SD3	SD6	SD9
S2	SD9	SD3	SD6
S3	SD6	SD9	SD3
S4	SD3	SD6	SD9
S5	SD9	SD3	SD6
S6	SD6	SD9	SD3
S7	SD3	SD6	SD9
S8	SD9	SD3	SD6
S9	SD6	SD9	SD3
S10	SD3	SD6	SD9

Table 7.2: Permutation of start delays for CS2.

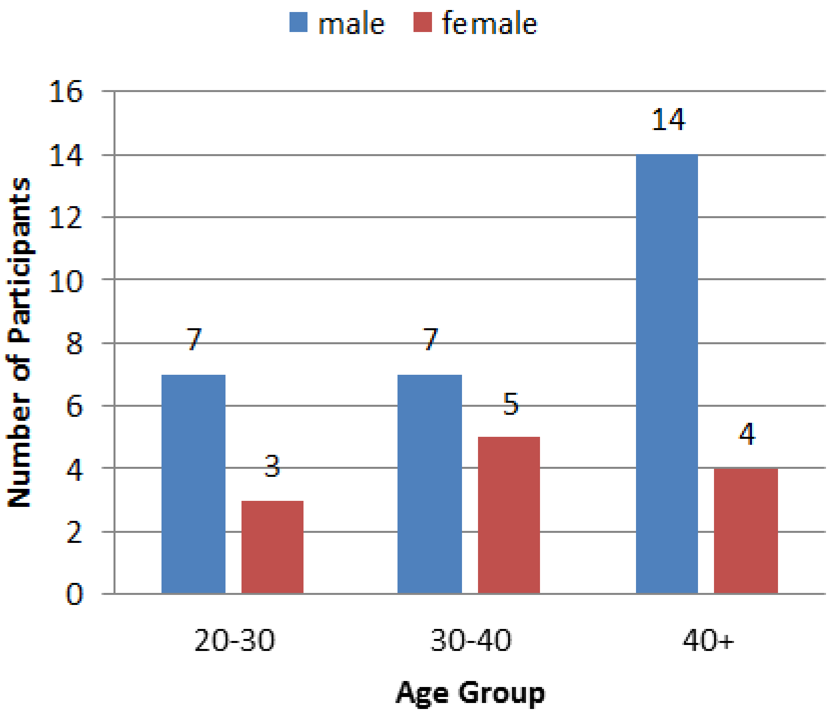


Figure 7.2: Participants by age group and gender.

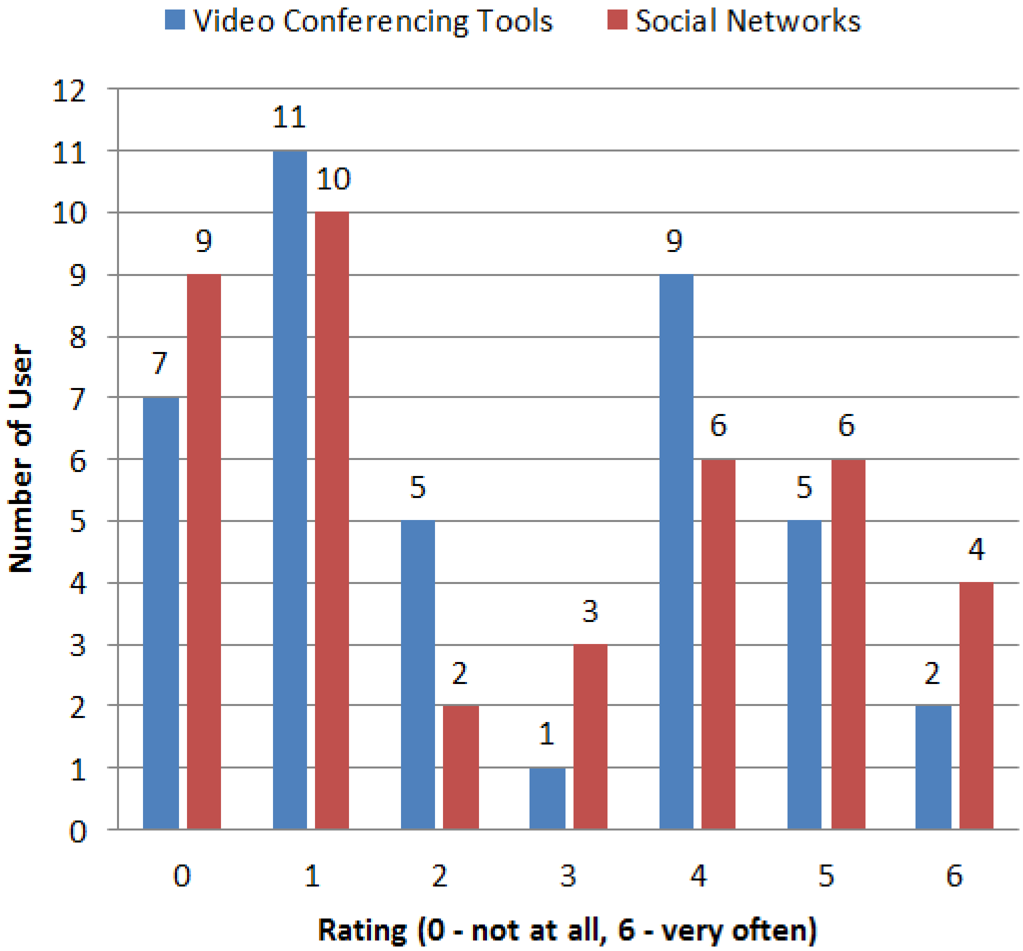


Figure 7.3: Usage of video conferencing tools and social networks among the 40 test participants.

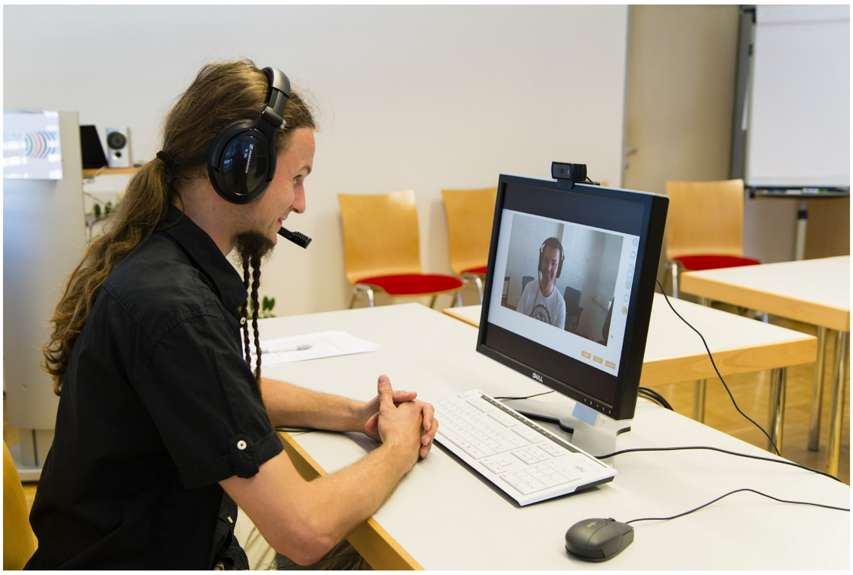


Figure 7.4: Experimental Setup of CS2 showing a participant interacting with the system.

In nearly every session, the group participants were work colleagues and already knew one another. There was only one group where nobody knew any of the other participants. The close relationship of the participants was helpful to establish a baseline of free communication, where no speaker was reluctant to participate in the discussion. 17 participants had previously participated in other kinds of usability tests not related to CS2.

7.6 Experimental Setup

The experimental setup is analogous to that of CS1 described in Section 6.5. In addition, one participant in each session was recorded with a HD camera from over the shoulder the capture interaction with the system. In CS1, it was not possible to capture each screen individually and merge the video with the separate audio recording. In CS2, audio and video recording was changed to focus on one participant's view. The audio and video experienced by one participant was recorded over the shoulder of the participant, was recorded making it possible to record and replay a complete session from the perspective of this one user. This is shown in Figure 7.4

7.7 Schedule

Each session with one group of four users completing three videoconferencing tasks followed the same schedule to manage the procedure of introduction, background questionnaire, tasks with recording and feedback questionnaires, and group debriefings. A session was designed to last less than one hour. The whole study was spread over 4 different days with 2 to 4 sessions per day. It was sometimes necessary to restart the session, because orchestration was not working correctly and one participant was either frozen or not displayed to the others. In those cases, the video conference was closed and restarted and the log files were stitched together.

Number	Question
Q1	How well could you see who was talking?
Q2	How often did it happen that 2 persons started talking at the same time?
Q3	How lively were the discussions?
Q4	How easy was it to contribute to the discussion?
Q5	How often did you see a non-speaking person?
Q6	How often did the system show a person that you did not want to see at that moment?
Q7	How often did the system show a speaking person too late?
Q8	How often did the system inappropriately switch to a person who had not started speaking?

Table 7.3: The eight questions used for ratings in CS2.

7.8 Feedback Questionnaire

Immediately after each of the three tasks, participants were asked to rate to preceding discussion according to eight criteria on a seven point scale, on a paper feedback questionnaire. Questions 1 to 4 were for comparison with CS1, questions 5 to 8 were orchestration-specific. The questions are listed in Table 7.3

The original version of the questionnaire can be found in Appendix B.4. The questions were presented to the participants in the form of semantic differentials usning a scale of [3 2 1 0 1 2 3], which were then converted internally to points from 0 (worst) to 6 (best) for analysis.

Determining questions relevant to orchestration was quite difficult. Question Q5 targets the accuracy of detection of significant voice activity events. If a non-speaking persons is displayed too often, it indicates a problem with the detection. This factor is influenced by many other factors as described in Section 3.3. Q6 is related to Q5, because a non-speaking person is often a person who should not be shown at a particular moment.

Q7 attempts to evaluate the orchestration delay time. The start delay theoretically delays the response time of the orchestration engine. If the start delay is longer, the time to detect a significant voice activity event is longer, and therefore the corresponding cut is delayed. This effect can be disturbing, especially when speaking times are relatively short compared to the start delay parameter. The effect of the delay can also be abolished through the transfer delay (end to end transport of audio and video signals) of the system. This is possible since the transfer delay is sometimes longer than the time to make a orchestration decision, which reduced the effect of the long start delay. In some cases, it can happen that orchestration switches to another participant before the person even starts to speak due to the delay. Since all information is gathered at the orchestration server, the system can make a decision while video and audio data is being transported to the recipients. To keep the audio and video synchronised, the system uses a certain amount of buffer time in which the command for a cut can be received before the event that caused it is displayed.

Q8 can be seen as a validation of Q5. It tries to assess the sensitivity of significant voice activity detection. A higher score with this question means that the system might be too sensitive, because the background noise or other variant factors are not being filtered correctly.

7.9 Subjective Measures

7.9.1 Group Debriefings

Table 7.4 summarises the group debriefings. Remarks originally in German have been translated into English for this report. Furthermore, the comments of each participant are denoted by A,B,C, and D, since the original mapping to the participant number was not recorded.

Session	Statements
1	**A:** Switching is no problem, when you can see yourself and the others immediately. The further into the test and the faster the switching was, the more disturbing it became. Requirement: One should see everything, or, and this is the minimal requirement, perhaps a small window that indicates, whether your own picture is sent or not. One wants to know this. I need a feedback, otherwise some sort of monitoring scenario develops. The different impressions of the participants are due to the fact that everybody had another view. The fact that you could not see yourself was very irritating, and significantly influenced the view to the whole conversation. **B:** All should see the same, this was a multiple response. And you can watch the other speaking, but not yourself, when speaking. In addition, one does not know, what the others see at that moment, that is very irritating. You often could see D, although D was not speaking. **C:** No matter what session, it is not usable in this way. 60% of the time, you see participants who do not say anything. I want to see a view of everyone. I feel uncomfortable to watch someone at random. When somebody starts to talk, I want to see him at once, but it takes 5 to 6 sentences, until you see a picture of the speaking person. You should see him after the first word. Is it intended, that one cannot see oneself? **D:** It is strange that you only can see the speaker, but not yourself. Wants to see himself too.
2	**A:** Concerning the switching behavior, session 1 and 2 are the best. Session 3 often shows the wrong person. Also wants to see himself. **B:** Session 2 is the best, although often non-speaking persons were shown. **C:** Session 1 is very satisfactory, because you see the picture of the speaker. But it is irritating, that you cannot see yourself speaking. And one does not know if the others can hear or see you. **D:** Session 1 is the best, session 3 the worst. Wants to see everybody, especially the speaker. Wants a reporting system for questions and interruptions.
3	**A:** Unmotivated switching in session 1 and 2. Session 3 was best. **B:** No comment. **C:** Not knowing if you are seen or not is very irritating. **D:** Prefers session 3, because previously non-speaking persons were often shown.
4	**A:** Session 3 was the worst, because most often non-speaking persons were shown. The system is very sensitive. **B:** Session 3 is the worst one, because too often there is a switch to non-speaking persons. Wants to see all participants. **C:** The third session was the worst because of the frequent switching to non-speaking persons. **D:** Session 1 was best, because non-speaking persons were shown least.
5	**A:** Session 1 is rated the best. The whole system is quite sensitive. **B:** Prefers to see all persons, possibility in a bar at the bottom. **C:** Likes the switching behavior of session 1. Session 2 and 3 significantly worse. No criticism of slow switching thought. **D:** Praises balance of configuration in session 1, noticeable change in 2 and 3.
6	**A:** Really wants to know, when he can be seen and wants to see himself when he is speaking. He thinks it is terrible, that he cannot listen to himself and heavily criticises the headset so as to be able to see the reactions and facial expressions of the other participants, he wants them all together in the picture. **B:** Does not feel irritated, when he sees no picture of the current speaker, but would like to see all persons. **C:** Praise for session 1, switching behavior of session 2 is described as too hectic. In session 3 the system seems unresponsive. Would like to see the others at least on a small picture. **D:** Feels session 1 is the best and would like to see himself speaking. Also wants to know, when he can be seen by the others.

7	**A:** Praises session 1, because he easily could keep an eye on the others. Likes switching behavior. **B:** Session 1 was rated worst. Session 2 left the best impression. **C:** Session 3 was best. Switching was not disturbing. **D:** Clear criticism of session 1. Almost always the same person in picture. Prefers session 3
8	**A:** In session 3, the switching caused a hectic feeling. Session 2 was pleasant effect. **B:** Switching of session 1 is unresponsive, praise for session 2. Switching of session 3 appears poor and at random. **C:** Agrees with A. **D:** Criticises too slow switch-over to the current speaker.
9	**A:** Definitely wants to know, when he can be seen by others. Feeling, that the most prominent person hardly was switched through. Sessions 1 and 3 seemed more natural, criticism of incomprehensible cuts in session 2. **B:** Wants everybody in the picture, because he does not want to miss nonverbal reactions(facial expressions) of the other participants. Too long dwelling on the same person is perceived more irritating than too fast switching, referring to session 2. **C:** Likes the good and smooth switching of session 3, calls it the best session. **D:** Does not prefer any session. Only noticed the delay in switching to the speaker.
10	**A:** Criticises closed headset, wants to be able to listen to himself. **B:** It does not matter, when you cannot listen to yourself. It also does not matter, when you cannot see the others. Prefers session 1, criticises that in session 2, too often, the wrong person was shown. **C:** Wants to have the speaker on the picture in time, not delayed, but also switched off immediately. Likes session 2 best, switching in session 1 is too hectic, and in session 3 is felt tedious. **D:** Inert behavior is more disturbing than fast or too fast switching. Wants to see all other persons on a bar. Session 1 turned out well, session 2 and 3 were tedious due to camera staying on the same person.

Table 7.4: The views of the users gathered during the group debriefing sessions.

7.9.2 Ratings for Each Condition on Eight Criteria

After each task, participants filled out a short questionnaire with eight ratings about the video conference they had just participated in. A summary of the ratings given for each of the three conditions SD3, SD6, and SD9 is shown in Table 7.5. An R script similar to the one described in Section 6.9.2 was used to analyse the differences between the ratings for the three conditions (start delays) for statistical significance. The analysis revealed statistically significant differences for questions Q5, Q6, and Q8. None of the other questions exhibited statistically significant differences between the ratings for the three conditions.

7.9.2.1 Q5 - How often did you see a non-speaking person?

For Q5, users rated SD3 statistically significantly worse than both SD6 and SD9. Hence, users significantly preferred a higher start delay. There was no statistically significant difference between SD6 and SD9. Figure 7.5 shows a boxplot of the ratings for each condition. This figure already suggests a preference for the higher start delays. The larger standard deviation of the two higher start delays is interesting, suggesting that differences between the conditions are not as apparent for every participant.

Listing 7.1 shows the result of the ANOVA analysis. The p-value of 0.000149 is well below the significance boundary of 0.05, indicating the existence of statistically significant differences somewhere within the three test conditions. Pairwise comparison with paired t-tests showed statistically signifi-

			SD3		SD6		SD9	
No.	Short Form	Question	Avg	SD	Avg	SD	Avg	SD
Q1	See who was talking	How well could you see who was talking?	3.95	1.38	4.40	1.24	4.35	1.14
Q2	Simultaneous start	How often did it happen that 2 persons started talking at the same time?	2.56	1.57	2.53	1.34	2.40	1.37
Q3	How lively	How lively were the discussions?	4.10	1.35	4.15	1.44	3.93	1.42
Q4	How easy to contribute	How easy was it to contribute to the discussion?	4.67	1.34	4.78	1.23	4.90	0.98
Q5	None-speaking person	How often did you see a non-speaking person?	4.33	1.06	3.25	1.46	3.33	1.61
Q6	Person not wanted at that moment	How often did the system show a person that you wouldn't want to see in that moment?	3.56	1.31	2.45	1.63	2.88	1.51
Q7	Delay before switching	How often did the system show a speaking person too late?	3.23	1.58	2.73	1.62	2.98	1.58
Q8	Inappropriate switching	How often did the system inappropriately switch to a person that didn't start speaking?	3.60	1.32	2.65	1.61	2.93	1.58

Table 7.5: Summary of user ratings for each of the three conditions (short-delays) on eight different criteria.

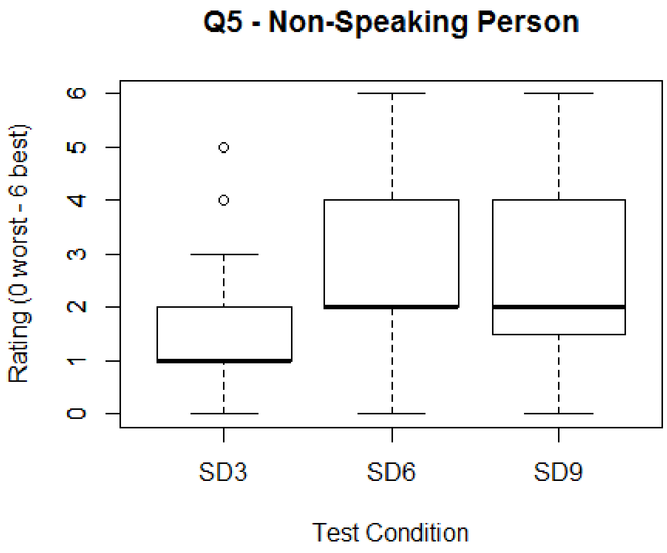


Figure 7.5: Boxplot of ratings for Q5 for each test condition (start delay).

```
Error: Users
          Df Sum Sq Mean Sq F value Pr(>F)
Residuals 39  111.5   2.858

Error: Users:Groups
          Df Sum Sq Mean Sq F value   Pr(>F)
Groups     2  29.28   14.64    9.89 0.000149 ***
Residuals 78 115.46    1.48
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

Listing 7.1: Analysis of variance (ANOVA) for Q5 showing a statistically significant difference between the distributions.

```
Pairwise comparisons using paired t tests

data:  Ratings and Groups

       SD3     SD6
SD6 0.00065 -
SD9 0.00097 0.79387

P value adjustment method: holm
```

Listing 7.2: Pairwise comparison with paired t-tests for Q5 showing statistically significant differences in the pairs SD3–SD6 and SD3–SD9.

cant differences between conditions SD3 and SD6 and between SD3 and SD9, as shown in Listing 7.2. However, the result of the Fligner-Killeen test in Listing 7.3 shows that the variances are not distributed homogeneously, and thus the results of the Friedman test are considered more reliable.

Figure 7.6 shows the results of the Friedman post hoc analysis. Again, the pairs SD3–SD6 and SD3–SD9 are statistically significant at $p < 0.05$. Q5 is the only question where Friedman post hoc analysis showed a statistically significant difference between more than one pair of conditions. In terms of seeing a non-speaking person (Q5), SD3 was rated statistically significantly worse than SD6 ($p = 0.00031$) and SD9 ($p = 0.00149$), apparently showing non-speaking persons more often.

7.9.2.2 Q6 - How often did the system show a person that you wouldn’t want to see in that moment?

Q6 also showed statistically significant differences for SD6. Figure 7.7 shows a boxplot of ratings for each condition, where there is some tendency towards SD6. The average of the SD6 is twice that of SD3 and also higher than SD9. However, the standard deviation is also quite high.

Listing 7.5 shows the result of the ANOVA analysis. The p-value is 0.000578, indicating the exis-

```
   Fligner-Killeen test of homogeneity of variances

data:   y by conditions
Fligner-Killeen:med chi-squared = 4.0781, df = 2, p-value = 0.1302
```

Listing 7.3: Fligner-Killeen test of Q5 showing that the variances of the data are not homogenous.

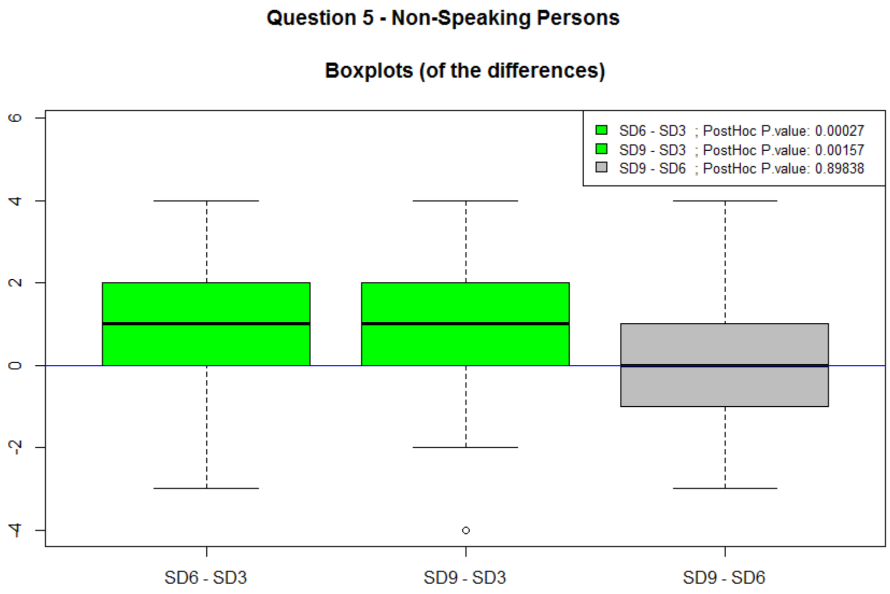


Figure 7.6: Friedman post hoc analysis for Q5 – None-Speaking Persons. The pairs SD3–SD6 and SD3–SD9 exhibit statistical significance at $p < 0.05$.

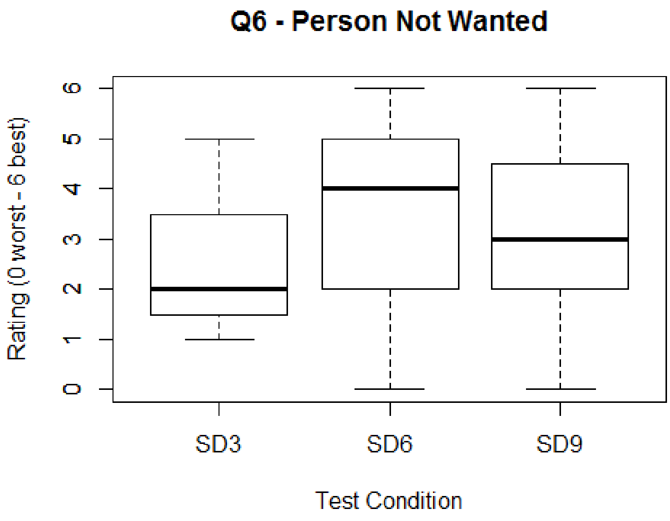


Figure 7.7: Boxplot of ratings for Q6 for each test condition (start delay).

```
  Fligner-Killeen test of homogeneity of variances

data:  y by conditions
Fligner-Killeen:med chi-squared = 3.8915, df = 2, p-value = 0.1429
```

Listing 7.4: Fligner-Killeen test of Q6 showing that the variances of the data are not homogenous.

```
Error: Users
          Df Sum Sq Mean Sq F value Pr(>F)
Residuals 39  137.8   3.534

Error: Users:Groups
          Df Sum Sq Mean Sq F value   Pr(>F)
Groups     2  25.29  12.645   8.216 0.000578 ***
Residuals 78 120.05   1.539
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Listing 7.5: Analysis of variance (ANOVA) for Q6 showing a statistically significant difference between the distributions.

tence of statistically significant differences somewhere within the three test conditions. Pairwise comparison with paired t-tests showed statistically significant differences between conditions SD3 and SD6 and between SD3 and SD9, as shown in Listing 7.6. However, the result of the Fligner-Killeen test in Listing 7.4 shows that the variances are not distributed homogeneously, and the results of the Friedman test are considered more reliable.

The results of the Friedman post hoc analysis are shown in Figure 7.8. The pair SD3–SD6 shows a statistically significant difference at $p > 0.05$. However, in the Friedman post-hoc analysis, the difference between SD3 and SD9 is no longer statistically significant.

In terms of seeing an unwanted person (Q6), SD3 was rated statistically significantly worse than both SD6 and SD9. For SD6 and SD9, the orchestration apparently performed better and showed the “right” person more often. There is no statistically significant difference in the ratings between SD6 and SD9.

7.9.2.3 Q8 - How often did the system inappropriately switch to a person that didn’t start speaking?

For Q8, users rated SD3 statistically significantly worse than both SD6 and SD9. Figure 7.9 shows a boxplot of ratings for each condition. The average rating for SD3 is lower, but the standard deviations for SD6 and SD9 are quite broad.

Listing 7.8 shows the result of the ANOVA analysis. The p- value is 0.00189 indicating the existence of statistically significant differences somewhere within the three test conditions. Pairwise comparison

```
Pairwise comparisons using paired t tests

data:  Ratings and Groups

       SD3     SD6
SD6 0.00071 -
SD9 0.03678 0.13315

P value adjustment method: holm
```

Listing 7.6: Pairwise comparison with paired t tests for Q6 showing a statistically significant difference in the pairs SD3 – SD6 and SD3 – SD9.

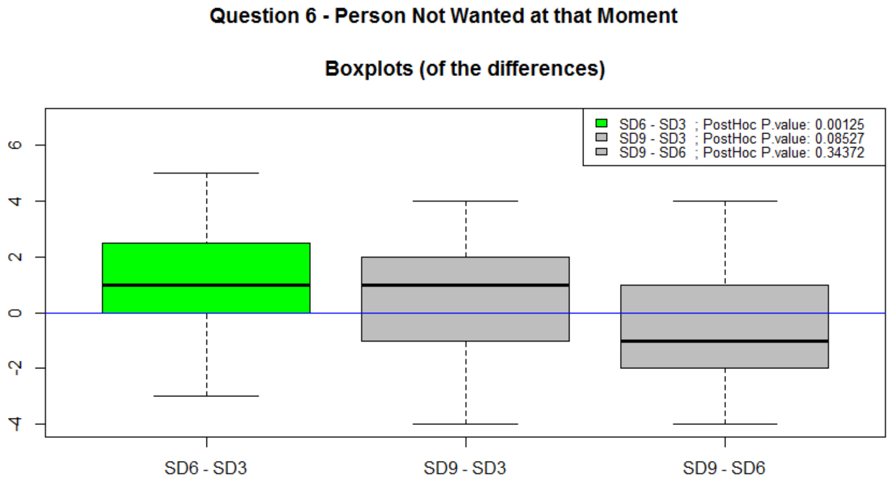


Figure 7.8: Friedman post-hoc analysis for Q6 – Person Not Wanted at that Moment. The pair SD3-SD6 is statistically significant at p<0.05.

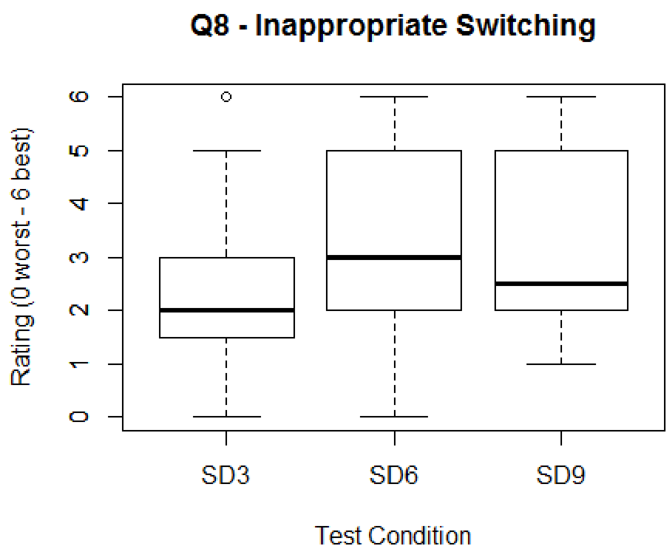


Figure 7.9: Boxplot of ratings for Q8 for each test condition (start delay).

```
  Fligner-Killeen test of homogeneity of variances

data:  y by conditions
Fligner-Killeen:med chi-squared = 4.1168, df = 2, p-value = 0.1277
```

Listing 7.7: Fligner-Killeen test of Q8 showing that the variances of the data are not homogenous.

with paired t-tests showed statistically significant differences between SD3 and SD6 (p = 0.0051) and between SD3 and SD9 (p = 0.0326), as shown in Listing 7.9. The difference between SD3 and SD6 is more significant than that between SD3 and SD9, which is close to the confidence level of 0.05. However, the result of the Fligner-Killeen test in Listing 7.7 shows that the variances are not distributed homogeneously, so the results of the Friedman test are considered more reliable.

The results of the Friedman post hoc analysis are shown In Figure 7.10. The pair SD3–SD6 shows a statistically significant difference at $p < 0.05$. In the Friedman post hoc analyses, the differences between SD3 and SD9 is no longer statistically significant, although the p-value is close at p=0.056. In terms of inappropriate switching (Q8), SD3 was rated statistically significantly worse than both SD6 and SD9. The shorter start delay of 300ms apparently leads to more inappropriate switching.

```
Error: Users
          Df Sum Sq Mean Sq F value Pr(>F)
Residuals 39  155.9   3.998

Error: Users:Groups
          Df Sum Sq Mean Sq F value  Pr(>F)
Groups     2  19.12   9.558   6.806 0.00189 **
Residuals 78 109.55   1.404
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Listing 7.8: Analysis of variance (ANOVA) for Q8 showing a statistically significant difference between the distributions.

```
Pairwise comparisons using paired t-tests

data:  Ratings and Groups

       SD3    SD6
SD6 0.0051 -
SD9 0.0326 0.2643

P value adjustment method: holm
```

Listing 7.9: Pairwise comparison with paired t-tests for Q 8 showing statistically significant differences in the pairs SD3–SD6 and SD3–SD9.

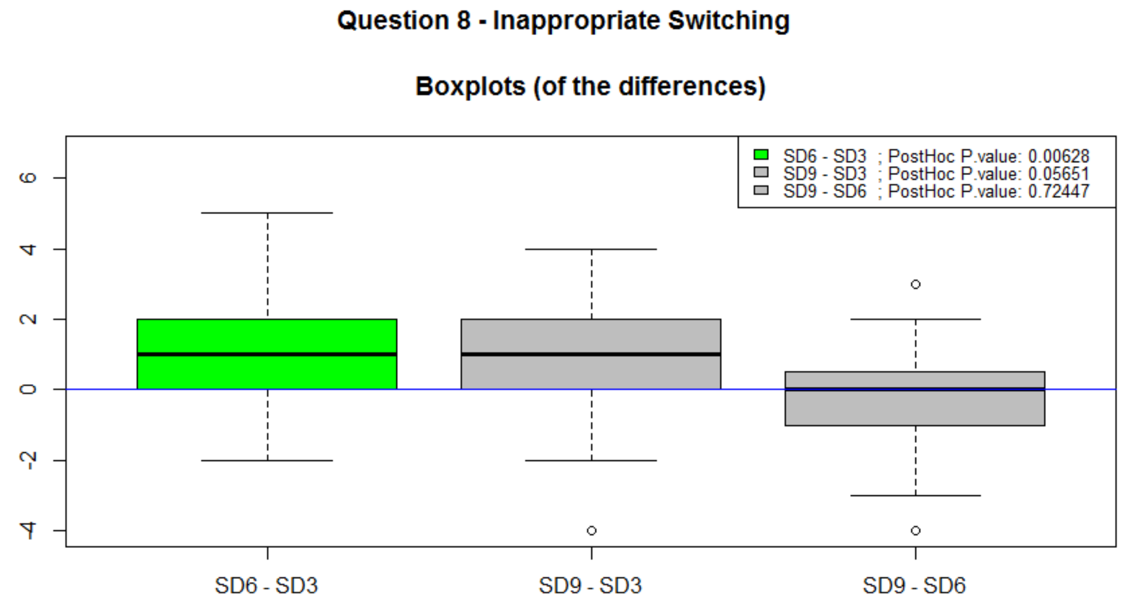


Figure 7.10: Friedman post hoc analysis for Q8 – Inappropriate Switching. The pair SD3–SD6 is statistically significant at $p < 0.05$.

7.9.3 Overall Preference

Users were asked to state an overall preference for one of the three test conditions. The results are summarised in Table 7.6. Every participant was also asked to explain their decision. Due to the permutation of conditions for each session, the participants choice is relative to the counterbalancing order, so the real condition is written in brackets. Remarks originally in German have been translated into English for this report.

Session	Participant	Preferred Condition	Comment
1	1	SD9	No session is usable in principle, because it seems, that 50% of the time always another person is seen.
1	2	SD6	My feeling is, that in the second session (SD6) I mostly see the person I expected to see, with the shortest delay.
1	3	SD3	Although wrong persons have been shown (non-speaking), intuitivly the best context speaker/speech has been achieved.
1	4	-	I did not recognise a big difference between the sessions, maybe the last (SD9) showed non-talking persons more than 2 or 1.
2	5	SD6	When I am speaking myself, I would like to have feedback, whether the others have me “switched on”.
2	6	SD6	No comment.
2	7	SD6	Could better see the person who was speaking
2	8	SD9	The second part of the session 2 (SD9) was good. Third session (SD3) was too fuzzy. First session (SD6) was also OK, but there were some delays.
3	9	SD6	It felt the most natural one of cutting, but maybe the topic was the reason that mostly only one (or none) person was speaking.
3	10	SD6	1st = Third session (SD6): One participant was displayed often although he did not speak. This was at least the case in this session 3 (SD9). The second Session (SD3) was worst because we couldn’t hear one user.
3	11	SD9	Good conversation and system working at its best.
3	12	SD6	Switching was that good in the third session, that nothing negative attracted my attention.
4	13	SD6	Made the most stable impression.
4	14	SD3	No comment.
4	15	-	No comment.
4	16	SD9	Most fluent performance of the software, only few wrong indications.
5	17	SD3	Best balance between sensitivity and duration of insertion.
5	18	SD6	Seemed the session with the least time of non-speaking persons on video.
5	19	SD3	This condition seems the most natural and most lively one.
5	20	SD6	Because the first session, the discussion was more lively, it is hard to compare all sessions.
6	21	SD6	Best switching behaviour.

Session	Participant	Preferred Condition	Comment
6	22	SD6	More precise switching, almost always the speaking person is seen.
6	23	SD3	Faster switching between talkers, fewer unwanted talkers.
6	24	SD6	The system did not switch to and fro in too hectic a way. Often the speaking person could be seen and in time too.
7	25	SD3	The first time the speaking person could be seen most commonly.
7	26	-	No comment.
7	27	SD6	I had the feeling that the switches were more accurate in session 2 (SD6). Too many switches in session 1 (SD3). Session 3 also performed very well despite the technical problems.
7	28	SD9	No comment.
8	29	SD3	The system worked best in session 3 and I could follow the discussion best. But one participant was often shown in all three discussions, even though he did not speak anything.
8	30	SD9	First condition (SD6) had delayed cutting between participants, second condition (SD9) was somewhat balanced, third condition (SD3) cut very badly and often.
8	31	SD9	No frequent switching between persons if no one was talking.
8	32	SD9	Probably the best match between speaking persons & shown person (however, it is difficult to tell / compare).
9	33	SD6	The system seemed to switch between the participants in the best manner (most appropriate & quickly).
9	34	SD6	It was the condition where you could follow the conversation in the most natural way compared too the others.
9	35	SD9	Second session (SD3) showed persons too long / didn't switch to speakers, not too much difference between 1st (SD9) und 3rd (SD6) session.
9	36	SD3	Seemed most balanced, however, this may be biased by the fact that this was the most calm discussion.
10	37	SD9	Concerning the appearance of the speaking persons, the 1st session felt more effective. The 2nd session was the worst in my opinion (wrong person, too late switching).
10	38	SD9	The first session (SD9) had the best quality in terms of showing the person who was actually speaking.
10	39	SD6	First session switching was too hectic. The third session too often showed non-speaking persons.
10	40	SD6	Second session seemed most comfortable (SD6). 1 and 2 were nearly equal (SD9& SD6), 3 (SD3) was a bit too often showing the "wrong" person.

Table 7.6: Overall preference expressed by each of the test users.

Of the 40 participants, 37 expressed a preference for one of the three test conditions: 8 voted for SD3 (start delay 300 ms), 18 for SD6 (600 ms), 11 for SD9 (900 ms) and 3 expressed no preference. The distribution is displayed in the Figure 7.11. Nearly half of the participants who expressed a preference

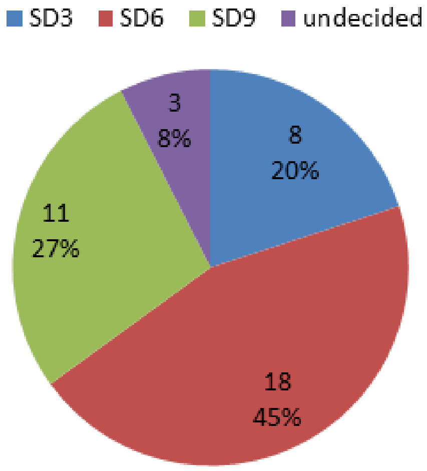


Figure 7.11: The distribution of votes for overall preference for one of the three test conditions (start delays). Three test users expressed no preference.

voted for SD6, suggesting that a start delay setting of 600 ms might be most suitable. A statistical analysis was performed to see if this preference is statistically significant.

An R script similar to the one described in Section 6.9.3 was used to analyse the overall preference for one of the three conditions (start delays) for statistical significance.

The analysis was performed with the simultaneous confidence intervals procedure of Sison-Glaz, as described in Section 4.8. The three no preference votes were discarded and the analysis was performed on the remaining 37 expressed preferences, as shown in Figure 7.12. More votes were cast for the higher start delays SD6 and SD9 than for SD3. However, the analysis showed that the differences in expressed overall preference are not statistically significant.

However, a tendency can be detected by combining the votes for the two longer start delays SD6 and SD9 into one category SD6+9 with 29 votes. The new distribution is shown in Figure 7.13. The likelihood niveau of the combined distribution SD6+SD9 is statistically significant compared to SD3, as shown in Figure 7.14 This extended analysis allows us to make the proposition that, overall participants prefer one of the two higher start delays SD6 and SD9 to the smaller start delay SD3.

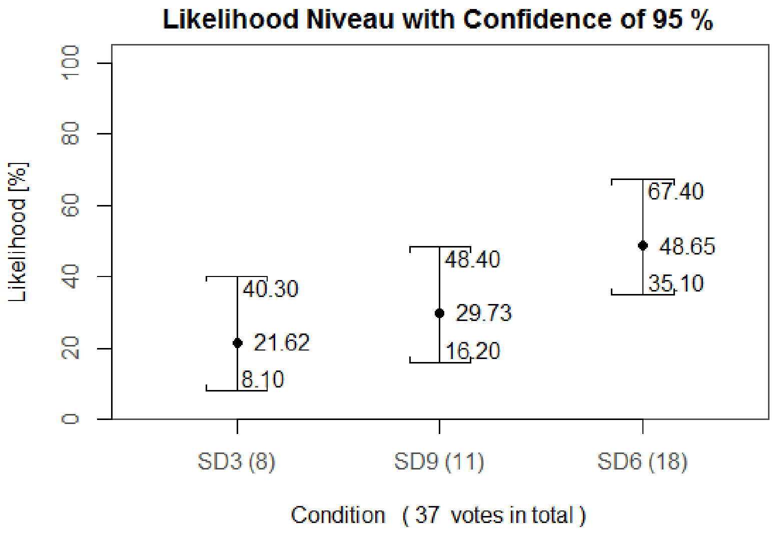


Figure 7.12: Overall preference analysis with confidence of 95% ($p < 0.05$) for overall preference for one of the three conditions. Each interval overlaps each of the others, so there are no statistically significant differences.

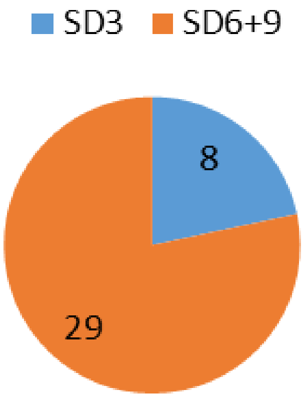


Figure 7.13: Combining votes for the two longer start delays SD6 and SD9 into one category SD6+9 with 29 votes.

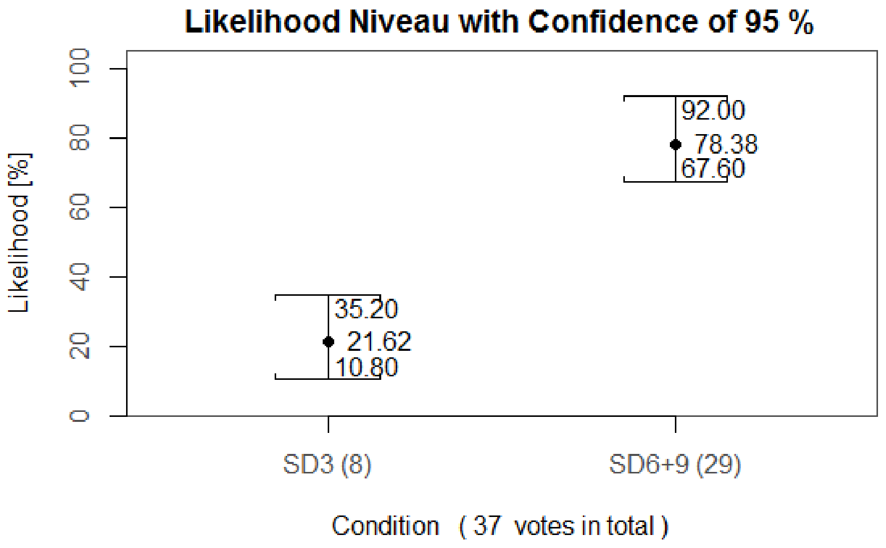


Figure 7.14: The number of votes for the two longer start delays SD6 and SD9 combined (SD6+SD9) is statistically significantly higher than for SD3.

Session No.	Condition	Problem – Solution
7	C	Multiple Value 0.0 for User S5C deleting: Row 1771 with 1402404838230,00 S5C 0.0 S5C_143.224.72.107_EXnNc33
7	C	Multiple Value 100.0 for User S4R2 deleting: Row 1767 with 1402404837333 S4R2 100.0 S4R2_143.224.72.177_nVE0jog
8	C	Combining with Session 8C1 combining at row 1170 deleting: Row: 1173 1402409351718, S5C, 0.0, S5C_143.224.72.107_zZu8V6J Row: 1169 1402409317749, S4R1, 100.0, S4R1_143.224.70.136_jPa8ePT Row: 1171 1402409351267, S4R1, 0.0, S4R1_143.224.70.136_qhjB7zn Row: 1156 1402409309110, S4R3, 100.0, S4R3_143.224.72.159_EWQ6ZG4 Row: 1178 1402409356175, S4R3, 0.0, S4R3_143.224.72.159_IyQru6l Row: 1213 1402409369113, S4R2, 0.0, S4R2_143.224.72.177_tqgdLHm
8	B	Combining with Session 8B1 combining at row 794 deleting: Row: 795 1402408544411, S5C, 0.0, S5C_143.224.72.107_XotMYE2 Row: 796 1402408548076, S4R3, 0.0, S4R3_143.224.72.159_kTkwuuU Row: 757 1402408497338, S4R1, 100.0, S4R1_143.224.70.136_k5YCSFb Row: 800 1402408550878, S4R1, 0.0, S4R1_143.224.70.136_iUTcmDW Row: 781 1402408503492, S4R2, 100.0, S4R2_143.224.72.177_5ww5Kbp Row: 855 1402408568770, S4R2, 0.0, S4R2_143.224.72.177_ek7Cl7w
7	C	Combining with Session 7C1 combining at row 2102 deleting: Row: 2101 1402404927211, S4R2, 100.0, S4R2_143.224.72.177_nVE0jog Row: 2103 1402404959342, S4R2, 0.0, S4R2_143.224.72.177_CN52fzA Row: 2109 1402404962515, S5C, 0.0, S5C_143.224.72.107_b9DWdhK Row: 2155 1402404978561, S4R1, 0.0, S4R1_143.224.70.136_Cyp35Tm Row: 2070 1402404912379, S4R1, 100.0, S4R1_143.224.70.136_U13e4lk Row: 2164 1402404982915, S4R3, 0.0, S4R3_143.224.72.159_i7IGfKW
4	A1	Changed to Session 4A

Table 7.7: Every alteration which was made to the original log files.

7.10 Objective Measures

The objective measures are based on log file analysis done by R scripts. Each session of CS2 was logged in the background, so the output of the orchestration engine can be recreated. Furthermore, the previously introduced conversation metrics can be calculated for the whole session. Hence, individual sessions can be compared on an objective level by comparing conversation metrics.

7.10.1 Log File Cleaning

A separate log file was created for each session and condition. Conditions which were interrupted due to technical problems were restarted and the log files were stitched together. The log files then were manually cleaned up. For example, the first event of a user cannot be a stop event, and a start event cannot immediately follow another start event. Such discrepancies had to be removed by hand to enable the script to work properly. Table 7.7 documents every alteration to the original log files. Each user in a session was assigned on an id based on the floor and room number. For example, S4R3 indicates room 3 on the 4th floor.

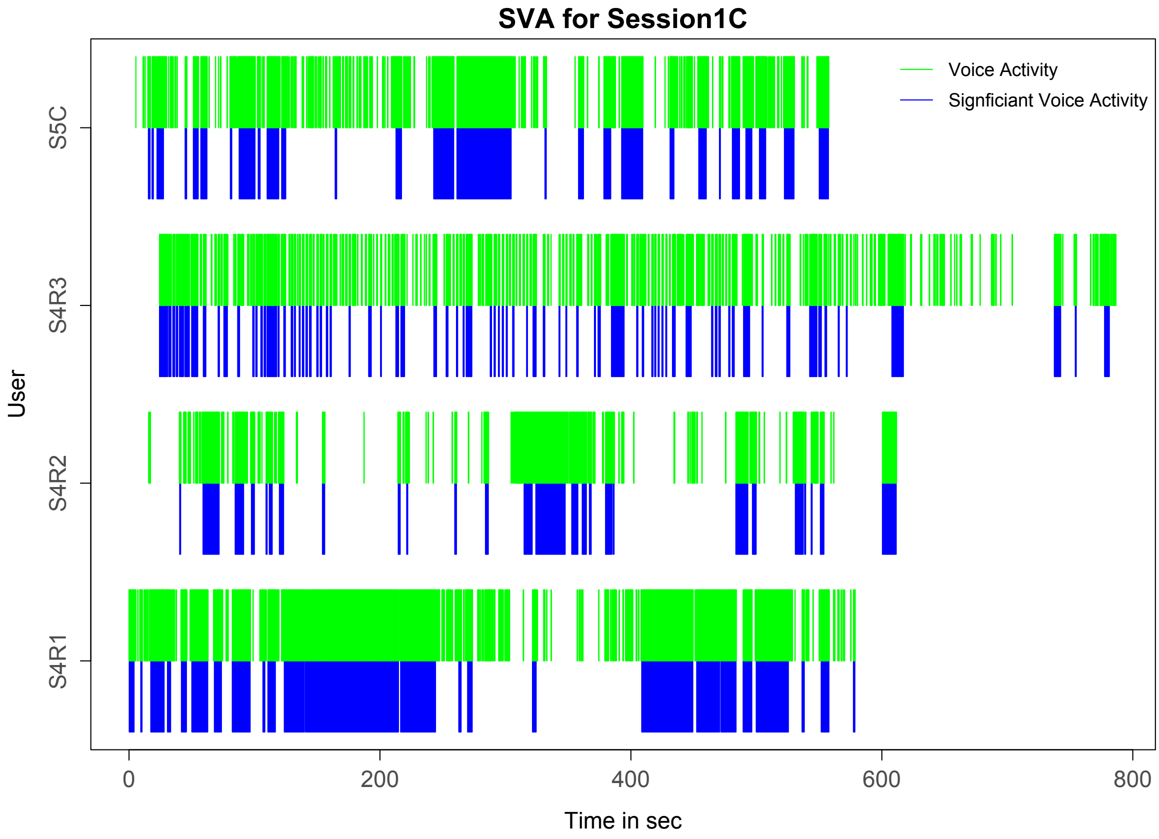


Figure 7.15: Significant voice activity events in blue compared to voice activity events in green for each of the four users in Session 1C with SD9. Unfortunately, the time spans do not match up.

7.10.2 Significant Voice Activity

Unfortunately, the log files were again found to be corrupted like in study CS1. Technicians tried to alter the logging process, but nevertheless the time series of each participant and the sessions were not synchronised as shown in Figure 7.15. It was not possible to cut out certain parts of the log file for comparison, since the difference in the speaking behaviour of each participant is only represented in the complete comparison.

Chapter 8

Concluding Remarks

The goal of this thesis was to evaluate automated orchestration of video conferencing within the video conferencing system of Vconect. Although technology has improved, video conferencing systems are still not considered to be as good as face-to-face meetings and therefore constitute a separate communication situation. One of the major problems of video conferencing is that each participant has a different perception of the conversational situation and communication. Orchestration is responsible for selecting the best view mode for each particular conversational situation and selecting which participant(s) should be actively shown for every participant individually.

The evaluation comprised two comparative studies. The first comparative study (CS1) compared the applicability of two view modes (tiled and full screen) within a slow turn-taking conversational situation with regard to their impact on communication and system quality. The second comparative study (CS2) investigated the impact of the start delay parameter on orchestration, which is mainly responsible for the sensitivity of the cutting behaviour. Three different degrees of sensitivity (start delays of 300, 600, and 900 ms) were compared within full screen view mode. The thresholds were chosen according to insights from previous evaluations and simulations.

The thesis first described the history of video conferencing systems and the drawbacks and benefits of such systems. It introduced the method of orchestration and how it is used within the Vconect project. It introduced three different, widely supported view modes and covered aspects of self view and self hearing. Basic terminology, the theoretical background of conversation metrics, and their usage for automated orchestration of video conference view modes were all described, as well as the influential factors in an orchestration system such as the one used in Vconect. The main statistical methods used to analyse the results of comparative studies and their implementation in R were described.

CS1 was carried out to determine if there is evidence for the suggestion from a previous study of a preference for full screen view mode in a slow turn-taking situation. The study was performed with 16 participants split into 4 groups of 4 using a repeated measures design. The participants' feedback suggested the reverse. A majority of participants preferred tiled view mode to see other (non-speaking) participants during such a conversation. This overall preference was not statistically significant at a confidence level of 95%, but was at 90%, indicating at least a tendency towards tiled view mode.

Individual feedback showed a need for a hybrid solution. Among other things, the system's cutting behaviour needs to be more adaptive, since users are not aware of the orchestration behaviour. The outcome of CS1 may have been distorted by poor voice activity detection and untested threshold estimations, which was the reason for CS2. Enhancing and tuning orchestration behaviour could decrease the need for more group telepresence. Including non-speaking participants into orchestration engine behaviour could increase the capability for non-verbal backchanneling and increase the perceived coherence of a conversation. Automatic view mode switching when detecting a monologue would give the speaker a better sense of the group's reaction.

CS2 was set up to discover a suitable candidate for the start delay threshold for full screen view mode. The study was performed with 40 participants divided into 10 groups of 4 using a repeated measures design. An analysis of participant feedback led to the conclusion that the optimal start delay threshold lies somewhere between 600 and 900 ms. The assumption that too long a start delay would reduce the quality of orchestration behaviour was not supported by the results of CS2. The results indicated that both the 600 ms and the 900 ms settings for start delay are statistically significantly preferred to 300 ms in the ratings Q5 (non-speaking persons), Q6 (persons not wanted at that time), and Q8 (inappropriate switching). Non-speaking persons were displayed less frequently, and the orchestration showed the “correct" person more often than with the smaller threshold. Likewise, inappropriate cutting was reduced with a higher start delay. Retrospectively, these observations also showed that the fixing the start delay in CS1 to 300 ms was not the best choice for comparing the two view modes.

Unfortunately, CS2 revealed no clear statistically significant overall preference, although a tendency towards one of the two higher start delay settings was detected. A possible solution for better suppression of false voice activity detection could be a factorisation of the threshold value. For example, counting a voice activity as significant if, during the period of the start delay, voice activity was present for a certain percentage of the time. This would reduce the negative effects of a long start delay and short drops of voice activity would be balanced out of the detection. The conversation logs gathered during the experiment were not useful for this study, but can be used for further simulations to help specify adaptive behaviour regarding the start delay.

Appendix A

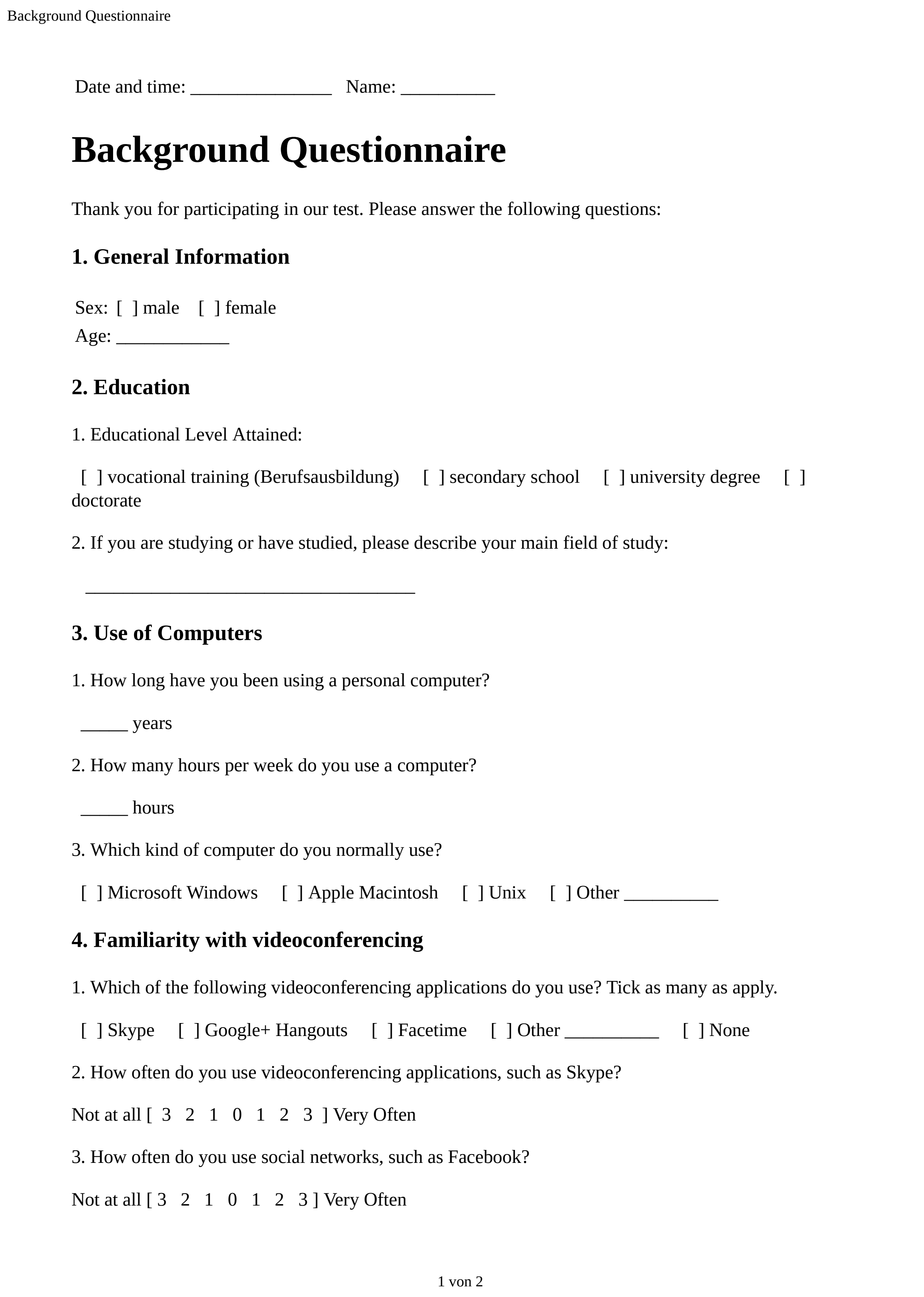
Materials for CS1

A.1 Background Questionnaire

A.2 Consent Form

A.3 Overall Feedback

A.4 Session Feedback



Background Questionnaire

Date and time: _______________ Name: __________

Background Questionnaire

Thank you for participating in our test. Please answer the following questions:

1. General Information

Sex: [ ] male [ ] female
Age: ____________

2. Education

1. Educational Level Attained:

[ ] vocational training (Berufsausbildung) [ ] secondary school [ ] university degree [ ] doctorate

2. If you are studying or have studied, please describe your main field of study:

____

3. Use of Computers

1. How long have you been using a personal computer?

_____ years

2. How many hours per week do you use a computer?

_____ hours

3. Which kind of computer do you normally use?

[ ] Microsoft Windows [ ] Apple Macintosh [ ] Unix [ ] Other __________

4. Familiarity with videoconferencing

1. Which of the following videoconferencing applications do you use? Tick as many as apply.

[ ] Skype [ ] Google+ Hangouts [ ] Facetime [ ] Other __________ [ ] None

2. How often do you use videoconferencing applications, such as Skype?

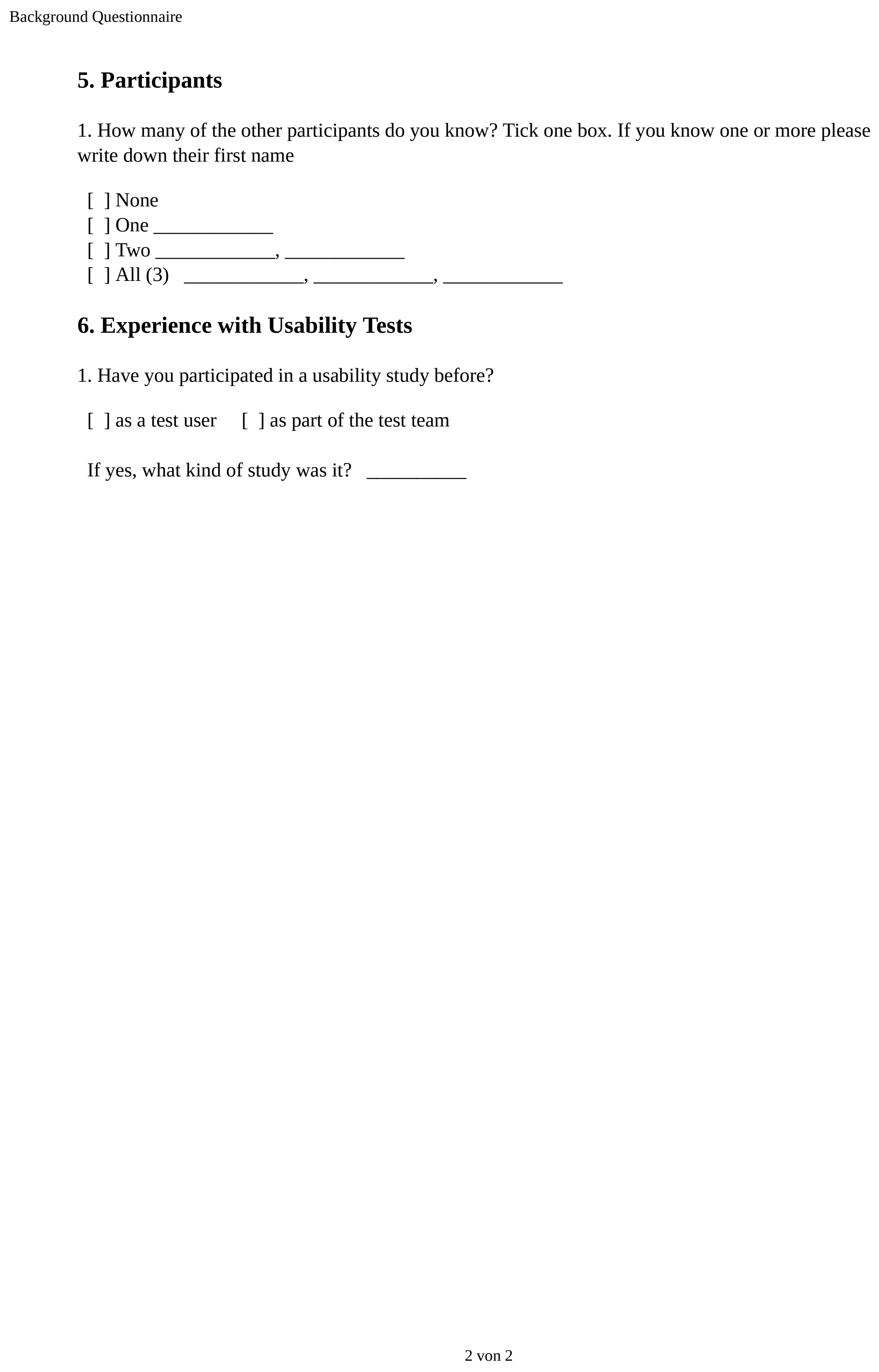
Not at all [3 2 1 0 1 2 3] Very Often

3. How often do you use social networks, such as Facebook?

Not at all [3 2 1 0 1 2 3] Very Often

1 von 2

Figure A.1: CS1 – Background Questionnaire 1



Background Questionnaire

5. Participants

1. How many of the other participants do you know? Tick one box. If you know one or more please write down their first name

[ ] None
[ ] One ____________
[ ] Two ____________, ____________
[ ] All (3) ____________, ____________, ____________

6. Experience with Usability Tests

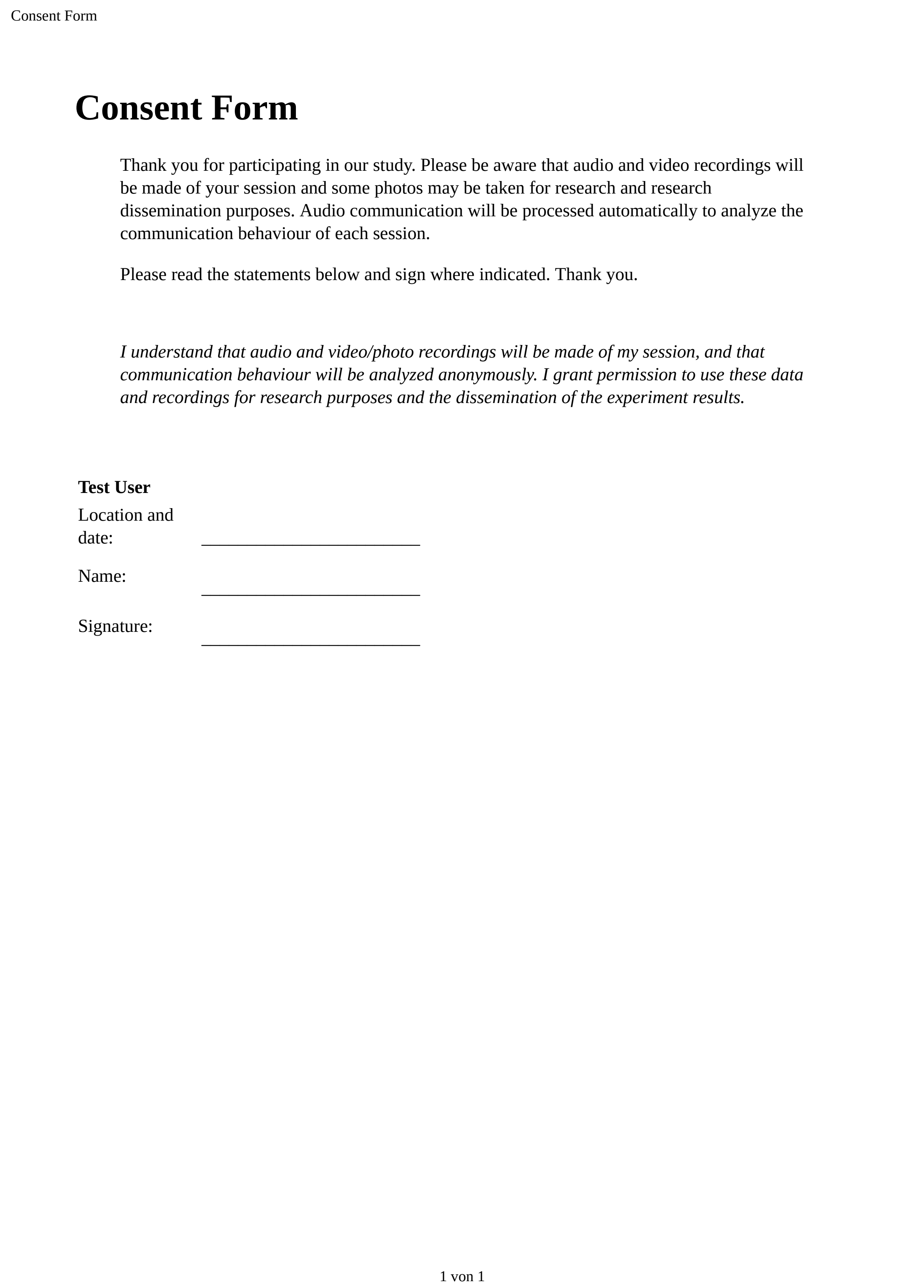
1. Have you participated in a usability study before?

[ ] as a test user [ ] as part of the test team

If yes, what kind of study was it? __________

2 von 2

Figure A.2: CS1 – Background Questionnaire 2



Consent Form

Consent Form

Thank you for participating in our study. Please be aware that audio and video recordings will be made of your session and some photos may be taken for research and research dissemination purposes. Audio communication will be processed automatically to analyze the communication behaviour of each session.

Please read the statements below and sign where indicated. Thank you.

I understand that audio and video/photo recordings will be made of my session, and that communication behaviour will be analyzed anonymously. I grant permission to use these data and recordings for research purposes and the dissemination of the experiment results.

Test User

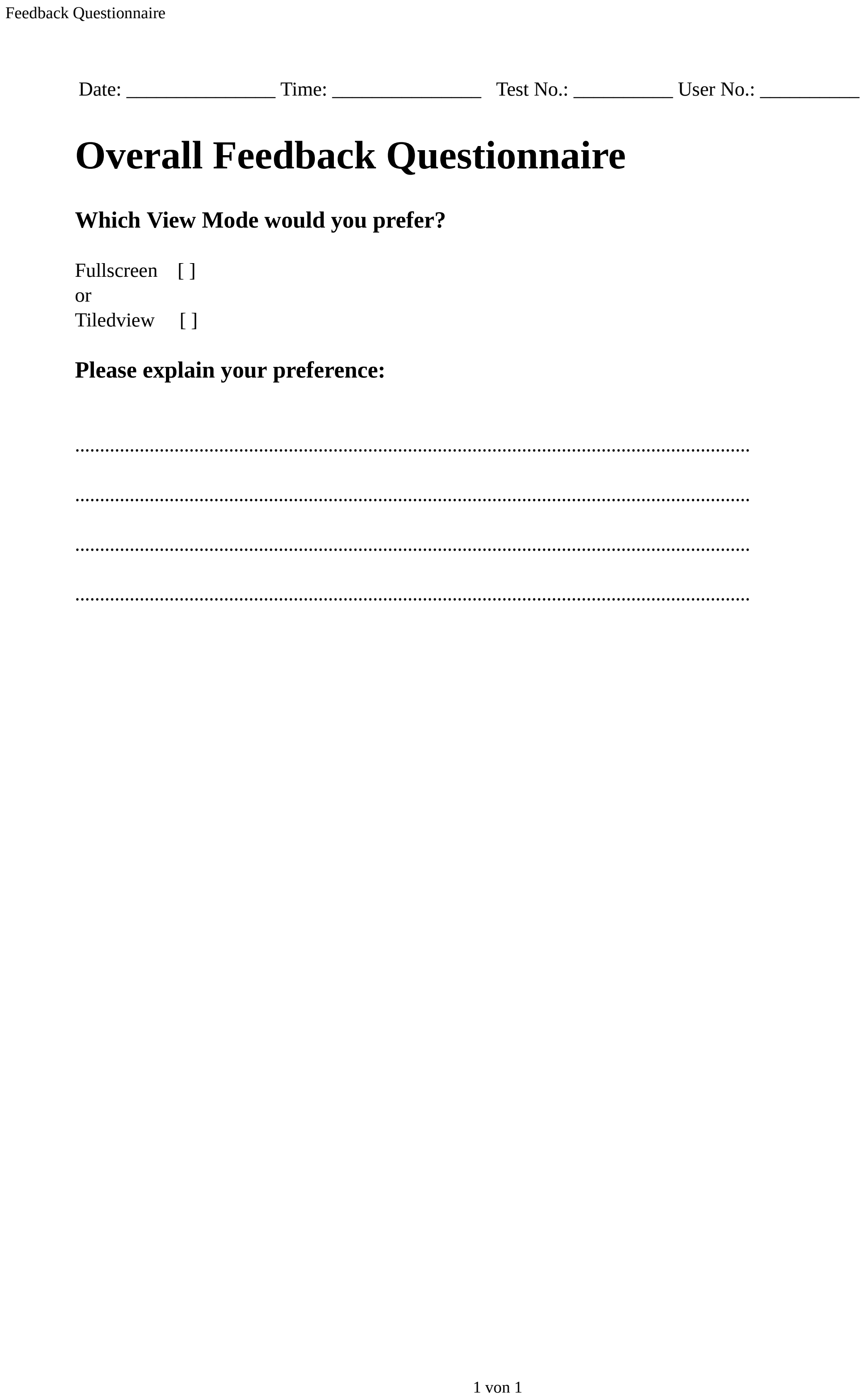
Location and date: ________________________

Name: ________________________

Signature: ________________________

1 von 1

Figure A.3: CS1 – Consent Form



Feedback Questionnaire

Date: _______________ Time: _______________ Test No.: __________ User No.: __________

Overall Feedback Questionnaire

Which View Mode would you prefer?

Fullscreen [ ]
or
Tiledview [ ]

Please explain your preference:

....

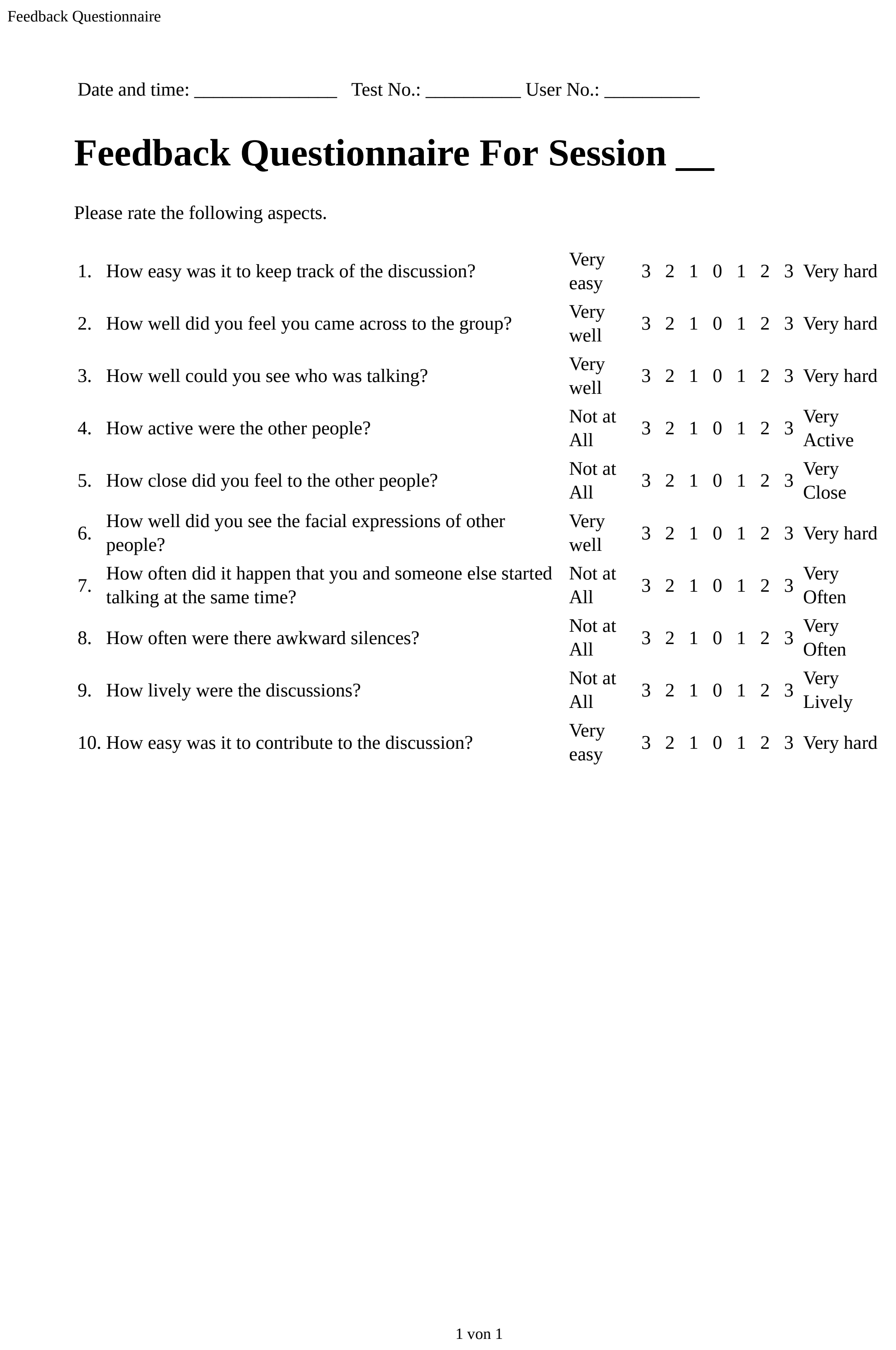
....

....

....

1 von 1

Figure A.4: CS1 – Overall Feedback



Feedback Questionnaire

Date and time: _______________ Test No.: __________ User No.: __________

Feedback Questionnaire For Session __

Please rate the following aspects.

1.	How easy was it to keep track of the discussion?	Very easy	3	2	1	0	1	2	3	Very hard
2.	How well did you feel you came across to the group?	Very well	3	2	1	0	1	2	3	Very hard
3.	How well could you see who was talking?	Very well	3	2	1	0	1	2	3	Very hard
4.	How active were the other people?	Not at All	3	2	1	0	1	2	3	Very Active
5.	How close did you feel to the other people?	Not at All	3	2	1	0	1	2	3	Very Close
6.	How well did you see the facial expressions of other people?	Very well	3	2	1	0	1	2	3	Very hard
7.	How often did it happen that you and someone else started talking at the same time?	Not at All	3	2	1	0	1	2	3	Very Often
8.	How often were there awkward silences?	Not at All	3	2	1	0	1	2	3	Very Often
9.	How lively were the discussions?	Not at All	3	2	1	0	1	2	3	Very Lively
10.	How easy was it to contribute to the discussion?	Very easy	3	2	1	0	1	2	3	Very hard

1 von 1

Figure A.5: CS1 – Session Feedback

Appendix B

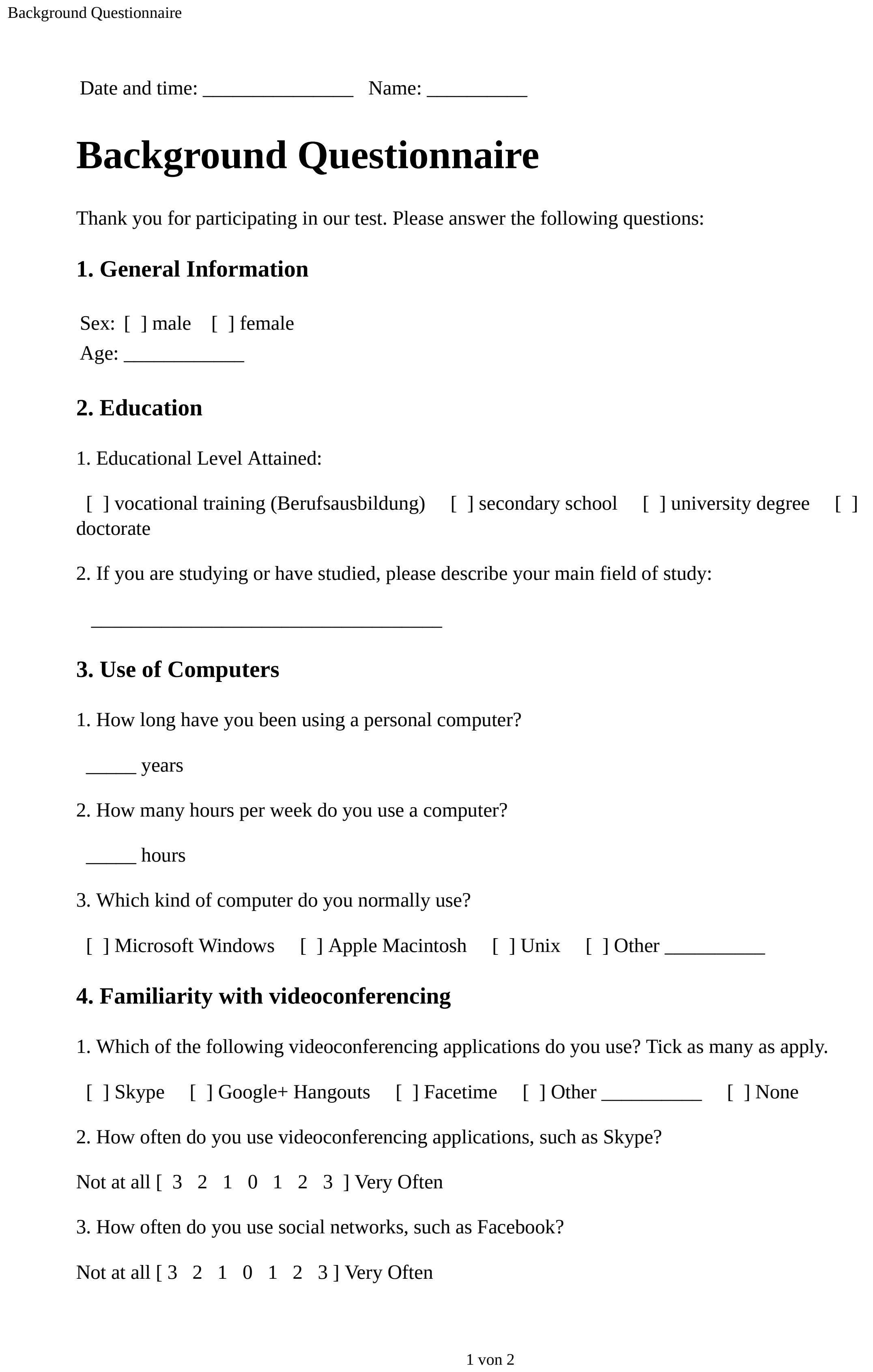
Materials for CS2

B.1 Background Questionnaire

B.2 Consent Form

B.3 Overall Feedback

B.4 Session Feedback



Background Questionnaire

Date and time: _______________ Name: __________

Background Questionnaire

Thank you for participating in our test. Please answer the following questions:

1. General Information

Sex: [ ] male [ ] female
Age: ____________

2. Education

1. Educational Level Attained:

[ ] vocational training (Berufsausbildung) [ ] secondary school [ ] university degree [ ] doctorate

2. If you are studying or have studied, please describe your main field of study:

____

3. Use of Computers

1. How long have you been using a personal computer?

_____ years

2. How many hours per week do you use a computer?

_____ hours

3. Which kind of computer do you normally use?

[ ] Microsoft Windows [ ] Apple Macintosh [ ] Unix [ ] Other __________

4. Familiarity with videoconferencing

1. Which of the following videoconferencing applications do you use? Tick as many as apply.

[ ] Skype [ ] Google+ Hangouts [ ] Facetime [ ] Other __________ [ ] None

2. How often do you use videoconferencing applications, such as Skype?

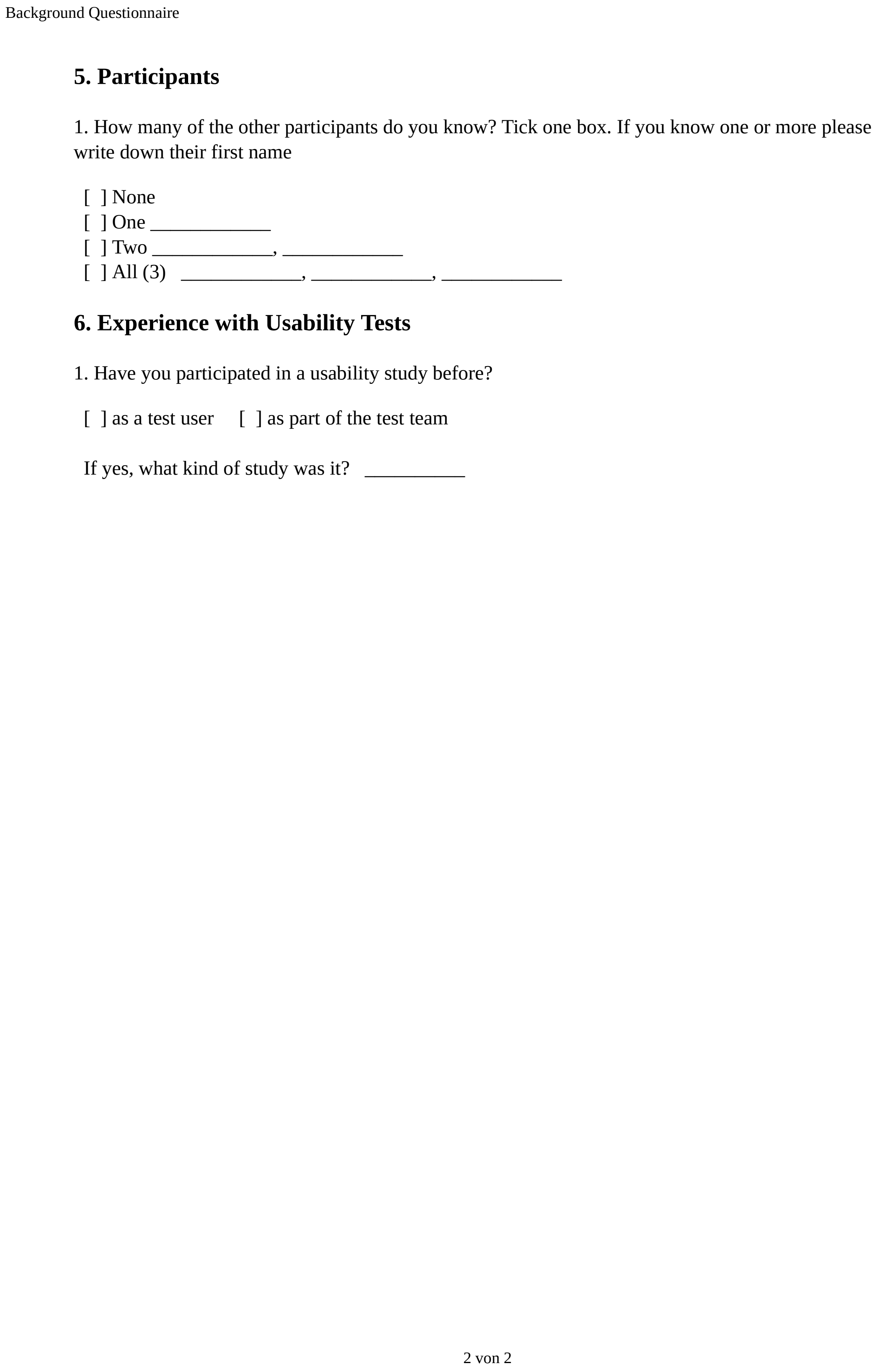
Not at all [3 2 1 0 1 2 3] Very Often

3. How often do you use social networks, such as Facebook?

Not at all [3 2 1 0 1 2 3] Very Often

1 von 2

Figure B.1: CS2 – Background Questionnaire 1



Background Questionnaire

5. Participants

1. How many of the other participants do you know? Tick one box. If you know one or more please write down their first name

[ ] None
[ ] One ____________
[ ] Two ____________, ____________
[ ] All (3) ____________, ____________, ____________

6. Experience with Usability Tests

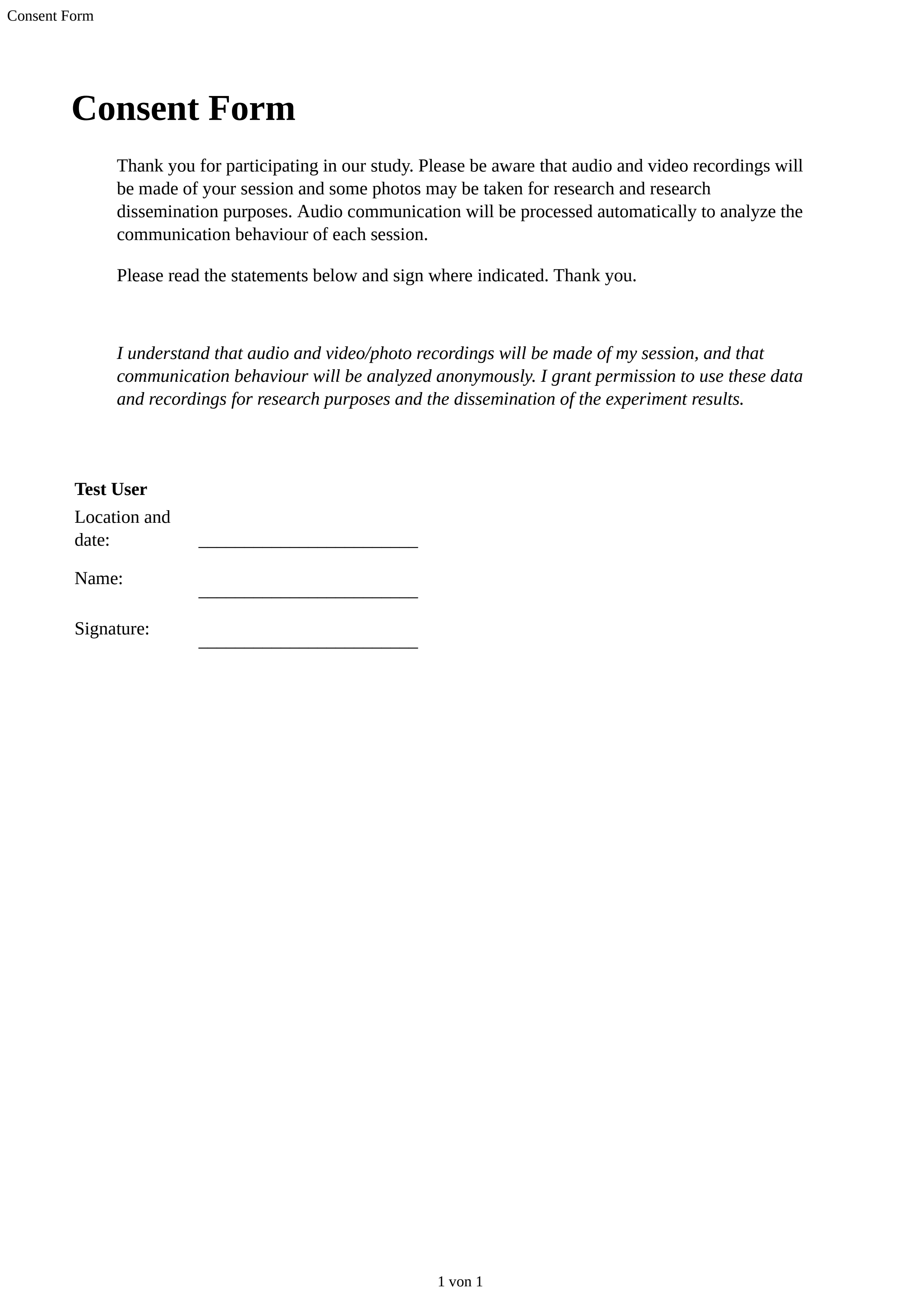
1. Have you participated in a usability study before?

[ ] as a test user [ ] as part of the test team

If yes, what kind of study was it? __________

2 von 2

Figure B.2: CS2 – Background Questionnaire 2



Consent Form

Consent Form

Thank you for participating in our study. Please be aware that audio and video recordings will be made of your session and some photos may be taken for research and research dissemination purposes. Audio communication will be processed automatically to analyze the communication behaviour of each session.

Please read the statements below and sign where indicated. Thank you.

I understand that audio and video/photo recordings will be made of my session, and that communication behaviour will be analyzed anonymously. I grant permission to use these data and recordings for research purposes and the dissemination of the experiment results.

Test User

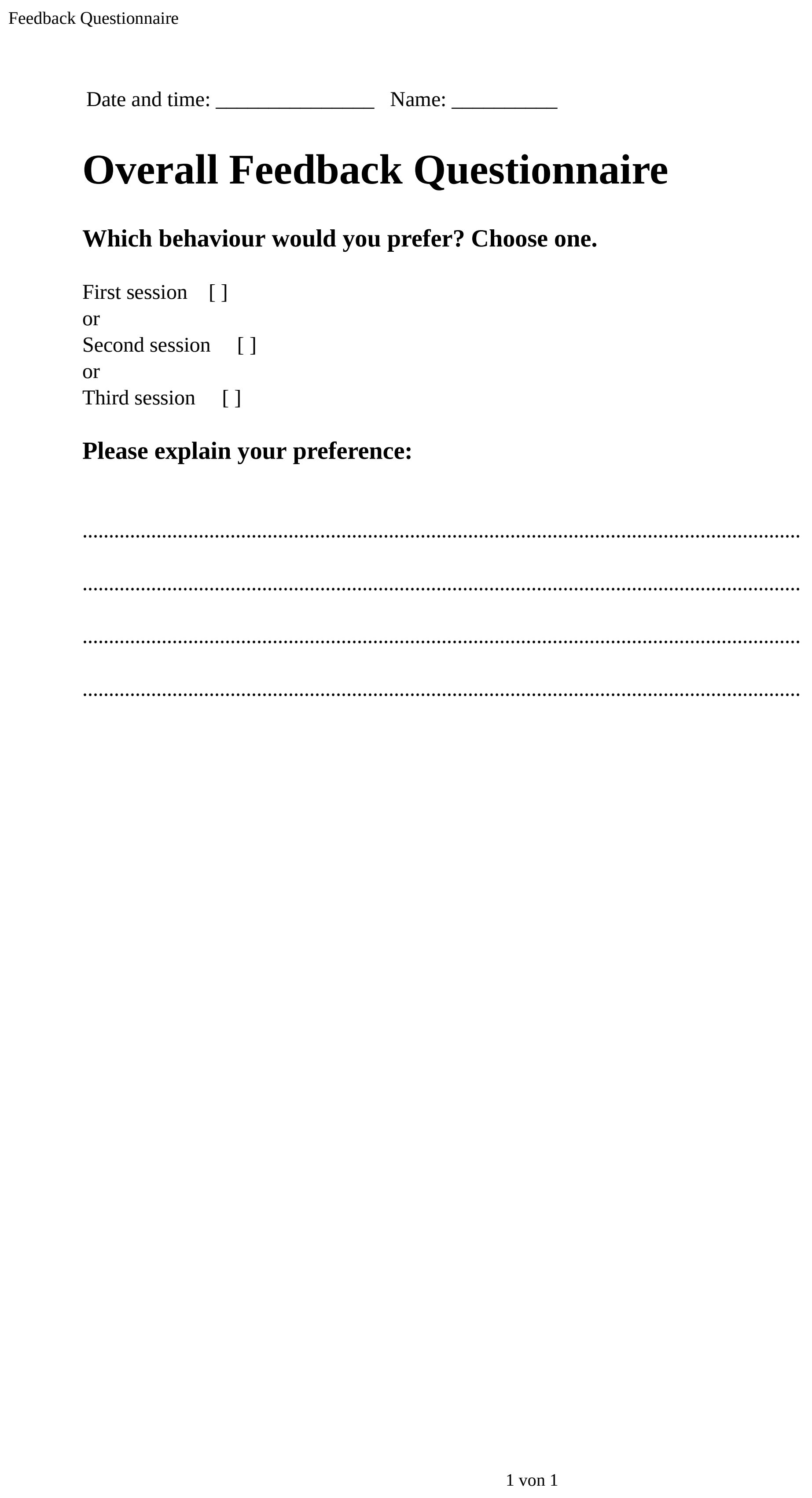
Location and date: ________________________

Name: ________________________

Signature: ________________________

1 von 1

Figure B.3: CS2 – Consent Form



Feedback Questionnaire

Date and time: _______________ Name: __________

Overall Feedback Questionnaire

Which behaviour would you prefer? Choose one.

First session [ ]
or
Second session [ ]
or
Third session [ ]

Please explain your preference:

....

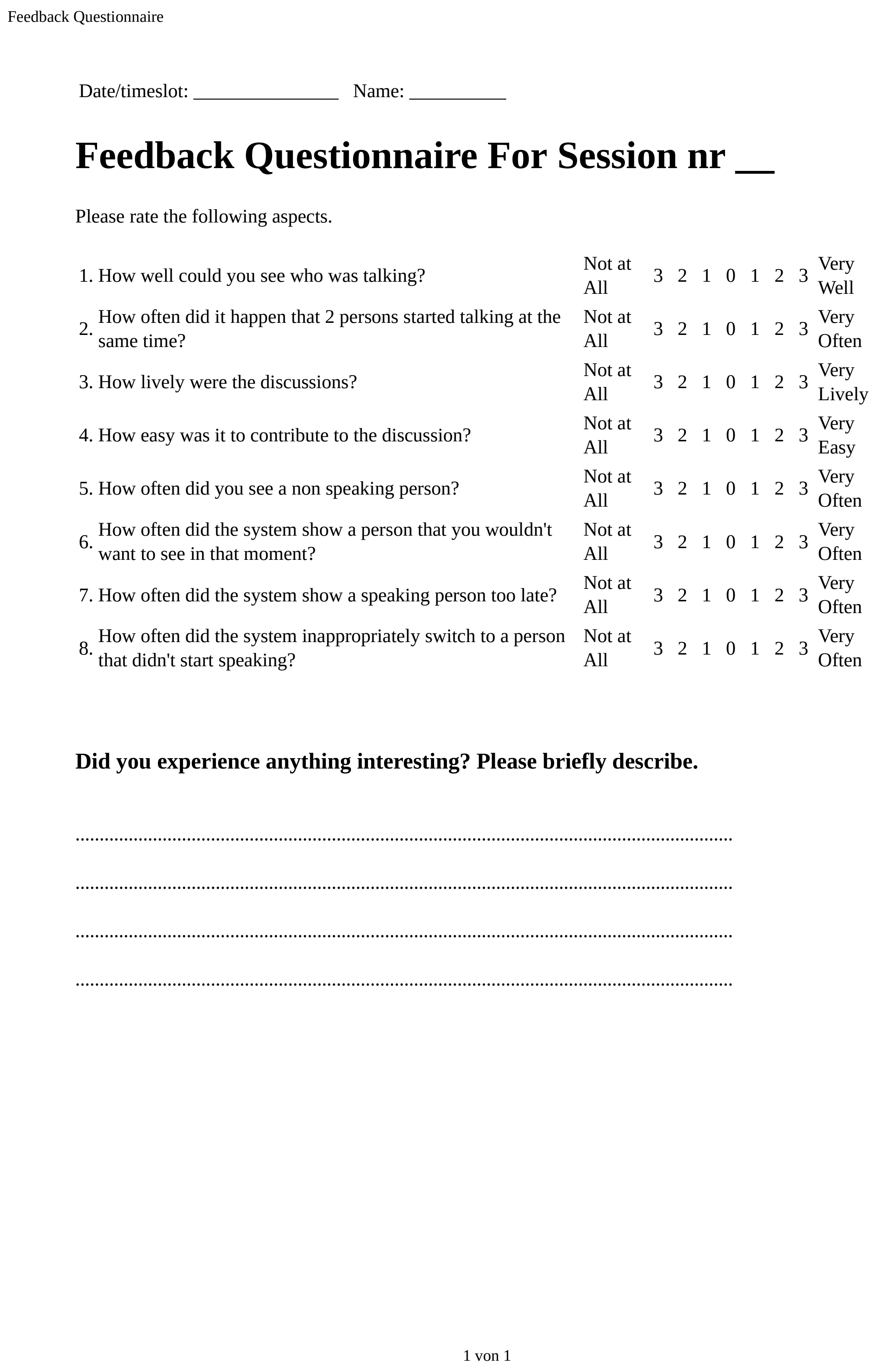
....

....

....

1 von 1

Figure B.4: CS2 – Overall Feedback



Feedback Questionnaire

Date/timeslot: _______________ Name: __________

Feedback Questionnaire For Session nr __

Please rate the following aspects.

1.	How well could you see who was talking?	Not at All	3	2	1	0	1	2	3	Very Well
2.	How often did it happen that 2 persons started talking at the same time?	Not at All	3	2	1	0	1	2	3	Very Often
3.	How lively were the discussions?	Not at All	3	2	1	0	1	2	3	Very Lively
4.	How easy was it to contribute to the discussion?	Not at All	3	2	1	0	1	2	3	Very Easy
5.	How often did you see a non speaking person?	Not at All	3	2	1	0	1	2	3	Very Often
6.	How often did the system show a person that you wouldn't want to see in that moment?	Not at All	3	2	1	0	1	2	3	Very Often
7.	How often did the system show a speaking person too late?	Not at All	3	2	1	0	1	2	3	Very Often
8.	How often did the system inappropriately switch to a person that didn't start speaking?	Not at All	3	2	1	0	1	2	3	Very Often

Did you experience anything interesting? Please briefly describe.

....

....

....

....

1 von 1

Figure B.5: CS2 – Session Feedback

Bibliography

Apache [2015]. *Apache ActiveMQ*. The Apache Software Foundation. 14 Aug 2015. `http://activemq.apache.org/` (cited on page 11).

Berndtsson, Gunilla, Mats Folkesson, and Valentin Kulyk [2012]. "Subjective Quality Assessment of Video Conferences and Telemeetings". In: *19th International Packet Video Workshop (PV 2012)*. IEEE, May 2012, pages 25–30. ISBN 1467303011. doi:10.1109/PV.2012.6229740 (cited on page 11).

BKA [2015]. *Urheberrechtsgesetz*. Bundeskanzleramt Rechtsinformationssystem des Bundes (RIS). 13 Aug 2015. `http://ris.bka.gv.at/GeltendeFassung.wxe?Abfrage=Bundesnormen&Gesetzesnummer=10001848` (cited on pages 4, 8, 14, 46).

Blokland, Art and Anne H. Anderson [1998]. "Effect of Low Frame-rate Video on Intelligibility of Speech". *Speech Communication - Special Issue on Auditory-Visual Speech Processing* 26.1-2 [Oct 1998], pages 97–103. ISSN 0167-6393. doi:10.1016/S0167-6393(98)00053-3 (cited on page 5).

Bortz, Jürgen and Nicola Döring [2006]. *Forschungsmethoden und Evaluation: für Human- und Sozialwissenschaftler*. Springer, 2006. ISBN 3540333053 (cited on page 23).

Brady, Paul T. [1968]. "A Statistical Analysis of On-Off Patterns in 16 Conversations". *Bell System Technical Journal* 47.1 [Jan 1968], pages 73–91. ISSN 0005-8580. doi:10.1002/j.1538-7305.1968.tb00031.x (cited on page 11).

Chen, Milton [2003]. "Conveying Conversational Cues through Video". PhD Thesis. Stanford University, Jun 2003. `https://graphics.stanford.edu/~miltchen/thesis.pdf` (cited on page 5).

Clark, Michael [2012]. *Introduction to R: A Second Course*. Center for Social Research at the University of Notre Dame. Nov 2012. `http://nd.edu/~mclark19/learn/Introduction_to_R_II.pdf` (cited on page 31).

Clark, Michael [2014]. *Introduction to R: A First Course in R*. Center for Social Research at the University of Notre Dame. Apr 2014. `http://nd.edu/~mclark19/learn/Introduction_to_R.pdf` (cited on page 31).

Dabbs Jr, James M. and R. Barry Ruback [1987]. "Dimensions of Group Process: Amount and Structure of Vocal Interaction". *Advances in Experimental Social Psychology* 20 [1987]. Edited by Leonard Berkowitz, pages 123–169. ISSN 0065-2601. doi:10.1016/S0065-2601(08)60413-X (cited on pages 11, 15).

Drools [2016]. *JBoss Drools Business Rules Management System (BRMS)*. 11 Dec 2016. `https://drools.org/` (cited on page 9).

Eder, Anselm [2007]. *Statistik für Sozialwissenschaftler*. Facultas, 2007. ISBN 3708900960 (cited on page 23).

Egido, Carmen [1988]. "Video Conferencing as a Technology to Support Group Work: A Review of Its Failures". In: *Proc. Conference on Computer-supported Cooperative Work (CSCW '88)*. CSCW

’88. ACM, 1988, pages 13–24. ISBN 0897912829. doi:10.1145/62266.62268. http://projects.ischool.washington.edu/mcdonald/courses/info447_au02/wk4/Egido.Fai%20lure.CSCW88.pdf (cited on page 5).

FU [1999]. *PictureTel Corp. History*. Funding Universe, citing the International Directory of Company Histories, Vol. 27, 1999. 1999. http://fundinguniverse.com/company-histories/picturetel-corp-history/ (cited on page 3).

Galili, Tal [2010]. *Post Hoc Analysis for Friedman’s Test*. Feb 2010. http://r-statistics.com/2010/02/post-hoc-analysis-for-friedmans-test-r-code/ (cited on pages 30, 37–38).

Geelhoed, Erik, Marian F. Ursu, Peter Stollenmayer, Doug Williams, Pedro Torres, Pablo Cesar, and Niko Farber [2014]. “Smart Video Communication for Social Groups - The Vconect Project”. In: *Integrating Social Media with Video Communication*. Volume 2. 2. IEEE Computer Society Special Technical Community on Social Networking E-Letter, 2014. http://stcsn.ieee.net/e-letter/stcsn-e-letter-vol-2-no-2/smart-video-communication-for-social-groups---the-vconect-project (cited on page 46).

Google Hangout [2016]. *Google Hangout Website*. 11 Dec 2016. https://hangouts.google.com/ (cited on page 6).

Gravano, Agustín and Julia Hirschberg [2011]. “Turn-Taking Cues in Task-Oriented Dialogue”. *Computer Speech Language* 25.3 [Jul 2011], pages 601–634. doi:10.1016/j.csl.2010.10.003. http://www.researchgate.net/publication/220629598_Turn-taking_cues_in_task-oriented_dialogue (cited on page 11).

Griffiths, Dawn [2009]. *Statistik von Kopf bis Fuß*. O’Reilly, 2009. ISBN 3897218917 (cited on page 23).

Hammer, Florian, Peter Reichl, and Alexander Raake [2004]. “Elements of Interactivity in Telephone Conversations”. In: *Proc. 8th International Conference on Spoken Language Processing (ICSLP 2004)*. 2004, pages 1741–1744. http://florianhammer.com/publications/Hammer2004_icslp.pdf (cited on page 11).

Hammer, Florian, Peter Reichl, and Alexander Raake [2005]. “The Well-Tempered Conversation: Interactivity, Delay and Perceptual VoIP Quality”. In: *IEEE International Conference on Communications (ICC 2005)*. Volume 1. May 2005, pages 244–249. doi:10.1109/ICC.2005.1494355. http://florianhammer.com/publications/Hammer2005_icc.pdf (cited on pages 11, 13–14).

Huang, Wei, Judith S. Olson, and Gary M. Olson [2002]. “Camera Angle Affects Dominance in Video-mediated Communication”. In: *Extended Abstracts on Human Factors in Computing Systems (CHI ’02)*. ACM, 2002, pages 716–717. ISBN 1581134541. doi:10.1145/506443.506562. http://doi.acm.org/10.1145/506443.506562 (cited on page 5).

ISO [2016]. *MPEG-4*. ISO Standard. 21 Mar 2016. http://www.iso.org/iso/iso_technical_committee?commid=45316 (cited on page 3).

Issing, Jochen and Nikolaus Farber [2012]. “Conversational Quality as a Function of Delay and Interactivity”. In: *Proc. 20th International Conference on Software, Telecommunications and Computer Networks (SoftCOM 2012)*. (Split, Croatia). 11 Sep 2012, pages 1–5. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6347574 (cited on page 11).

ITU [2010]. *G.720.1: Generic Sound Activity Detector*. International Telecommunication Union (ITU). 13 Jan 2010. http://itu.int/rec/T-REC-G.720.1 (cited on page 11).

ITU [2016a]. *H.263: Video Coding for Low Bit Rate Communication*. International Telecommunication Union (ITU). 21 Mar 2016. https://www.itu.int/rec/T-REC-H.263/en (cited on page 3).

ITU [2016b]. *H.323: Packet-Based Multimedia Communications Systems*. International Telecommunication Union (ITU). 21 Mar 2016. http://www.itu.int/rec/T-REC-H.323/en (cited on page 3).

Jaffe, Joseph and Stanley Feldstein [1970]. *Rhythms of Dialogue*. Personality and Psychopathology. Academic Press, Jul 1970. ISBN 0123798507 (cited on pages 11–12).

Jansen, Jack, Pablo César, Dick C. A. Bulterman, Tim Stevens, Ian Kegel, and J. Issing [2011]. "Enabling Composition-Based Video-Conferencing for the Home." *IEEE Transactions on Multimedia* 13.5 [2011], pages 869–881. doi:10.1109/TMM.2011.2159369. http://homepages.cwi.nl/~garcia/material/ieeetmm2011.pdf (cited on page 9).

Kaiser, Rene, Wolfgang Weiss, Manolis Falelakis, Spiros Michalakopoulos, and Marian F. Ursu [2012]. "A Rule-Based Virtual Director Enhancing Group Communication". In: *Proc. International Conference on Multimedia and Expo Workshops (ICMEW '12)*. IEEE, 2012, pages 187–192. ISBN 076954729X. doi:10.1109/ICMEW.2012.39 (cited on page 8).

King, William B. [2014]. *ANOVA with Repeated Measures Factors*. Coastal Carolina University. Oct 2014. http://ww2.coastal.edu/kingw/statistics/R-tutorials/repeated.html (cited on page 36).

Kitawaki, N. and K. Itoh T. Kurita [1991]. "Effects of Delay on Speech Quality". *NTT Review* 3.5 [Sep 1991], pages 88–94. doi:10.1207/s15327051hci2103_1 (cited on page 5).

ME [2016]. *Bonferroni-Korrektur nach Holm*. Methoden der Entwicklungspsychologie. 12 Dec 2016. http://methoden-psychologie.de/alphafehlerkumulierung2.html (cited on page 26).

Microsoft [1998]. *NetMeeting*. 1998. https://technet.microsoft.com/en-us/library/dd361923.aspx (cited on pages 3–4).

Miracle Theatre Group [2016]. *Cornwall's Miracle Theatre Group*. 11 Dec 2016. http://www.miracletheatre.co.uk/ (cited on page 7).

Montgomery, Douglas C. [2009]. *Design and Analysis of Experiments*. 7th Edition. Wiley, 20 Jan 2009. 680 pages. ISBN 0470398825 (cited on page 23).

Moosbrugger, Helfried [2002]. *Lineare Modelle: Regressions- und Varianzanalysen*. Hans Huber, 2002. ISBN 3456839014 (cited on page 23).

MultinomialCI [2015]. *MultinomialCI: Simultaneous confidence intervals for multinomial proportions according to the method by Sison and Glaz*. The R Foundation. 12 Dec 2015. https://cran.r-project.org/web/packages/MultinomialCI/index.html (cited on page 38).

O'Conaill, Brid, Steve Whittaker, and Sylvia Wilbur [1993]. "Conversations over Video Conferences: An Evaluation of the Spoken Aspects of Video-Mediated Communication". *Human Computer Interaction* [Dec 1993], pages 389–428. ISSN 0737-0024. doi:10.1207/s15327051hci0804_4. http://www.interruptions.net/literature/O_Conaill-HCI93-L.pdf (cited on page 15).

Pinheiro, J. and D. Bates [2000]. *Mixed-Effects Models in S and S-PLUS*. Statistics and Computing. Springer, 2000. ISBN 0387989579. http://verde.esalq.usp.br/~jorge/cursos/modelos_longitudinais/Mixed%20Effects%20Models%20in%20S%20and%20S-Plus.pdf (cited on pages 23, 29, 37).

Plackett, R. L. [1983]. "Karl Pearson and the Chi-Squared Test". *International Statistical Review* 51.1 [1983], pages 59–72. http://floppybunny.org/robin/web/virtualclassroom/stats/basics/articles/chi_square/chi_square_review_plackett_1983.pdf (cited on pages 27, 38).

Quick, John M. [2016]. *R Tutorial Series: One-Way ANOVA with Pairwise Comparisons*. 7 Dec 2016. `http://rtutorialseries.blogspot.com/2011/01/r-tutorial-series-one-way-anova-with.html` (cited on page 36).

R [2015]. *The Comprehensive R Archive Network*. The R Foundation. 28 Sep 2015. `https://cran.r-project.org/` (cited on page 31).

R [2016]. *The R Project for Statistical Computing*. The R Foundation. 26 Jul 2016. `https://r-project.org/` (cited on page 31).

Reeves, Byron and Clifford Nass [1996]. *The Media Equation: How People Treat Computers, Television, and New Media Like Real People and Places*. Cambridge University Press, 1996. ISBN 157586052X (cited on page 5).

Reichl, Peter [2007]. "From "Quality-of-Service" and "Quality-of-Design" to "Quality-of-Experience": A Holistic View on Future Interactive Telecommunication Services". In: *Proc. 15th International Conference on Software, Telecommunications and Computer Networks (SoftCOM 2007)*. Sep 2007, pages 1–6. doi:10.1109/SOFTCOM.2007.4446062 (cited on page 11).

RStudio [2016]. *R Studio*. 12 Dec 2016. `http://rstudio.com/` (cited on page 31).

Ruhleder, Karen and Brigitte Jordan [1999]. "Meaning-Making Across Remote Sites: How Delays in Transmission Affect Interaction". In: *Proc. 6th European Conference on Computer Supported Cooperative Work (ECSCW 1999)*. 1999, pages 411–429. ISBN 9401144419. doi:10.1007/978-94-011-4441-4. `http://ecscw.org/1999/22.pdf` (cited on page 11).

Ruhleder, Karen and Brigitte Jordan [2001]. "Co-Constructing Non-Mutual Realities: Delay-Generated Trouble in Distributed Interaction". *Computer Supported Collaborative Work* 10.1 [Jan 2001], pages 113–138. ISSN 0925-9724. doi:10.1023/A:1011243905593. `http://dx.doi.org/10.1023/A:1011243905593` (cited on page 5).

SAPO [2016]. *Portugese Telecom / SAPO Online Social Network*. 11 Dec 2016. `http://www.sapo.pt/` (cited on page 7).

Sellen, Abigail J. [1992]. "Speech Patterns in Video-Mediated Conversations". In: *Proc. of the SIGCHI Conference on Human Factors in Computing Systems (CHI 1992)*. (Monterey). May 1992, pages 49–59. ISBN 0897915135. doi:10.1145/142750.142756 (cited on pages 11–12).

Shriberg, Elizabeth, Andreas Stolcke, and Don Baron [2001]. "Observations on Overlap: Findings and Implications for Automatic Processing of Multi-Party Conversation". In: *Proc. European Conference on Speech Communication and Technology (EUROSPEECH 2001)*. 2001, pages 1359–1362. doi:10.1.1.29.3036. `http://www1.icsi.berkeley.edu/ftp/pub/speech/papers/euro2001-overlap.pdf` (cited on page 11).

Sison, Cristina P. and Joseph Glaz [1995]. "Simultaneous Confidence Intervals and Sample Size Determination for Multinomial Proportions". *Journal of the American Statistical Association* 90.429 [Mar 1995], pages 366–369. doi:10.1080/01621459.1995.10476521. `http://tx.liberal.ntu.edu.tw/~purplewoo/literature/!Methodology/!Distribution_SampleSize/SimultConfidIntervJASA.pdf` (cited on pages 23, 26, 38).

StatSoft [2016a]. *ANOVA*. 12 Dec 2016. `http://statsoft.de/glossary/G/GeneralANOVAMANOVA.htm` (cited on pages 23, 29).

StatSoft [2016b]. *Bonferroni Adjustment Method*. 12 Dec 2016. `https://statsoft.de/glossary/B/BonferroniAdjustment.htm` (cited on page 25).

StatSoft [2016c]. *Kruskal-Wallis Test*. 12 Dec 2016. `http://statsoft.de/glossary/K/KruskalWallisTest.htm` (cited on page 23).

StatSoft [2016d]. *Shapiro-Wilk Test of Normality*. 12 Dec 2016. https://statsoft.de/glossary/S/ShapiroWilksWtest.htm (cited on page 26).

StatSoft [2016e]. *T-Test für Unabhängige Stichproben*. 12 Dec 2016. http://statsoft.de/glossary/T/ttestForIndependentandDependentSamples.htm (cited on pages 23, 28).

StatSoft [2016f]. *Wilcoxon-Test*. 12 Dec 2016. http://statsoft.de/glossary/W/WilcoxonTest.htm (cited on pages 23, 28).

Tang, John C. and Ellen Isaacs [1993]. “Why Do Users Like Video? Studies of Multimedia-Supported Collaboration”. *Computer Supported Cooperative Work (CSCW)* 1.3 [1993], pages 163–196. ISSN 1573-7551. doi:10.1007/BF00752437. 10.1007/BF00752437 (cited on page 5).

Ursu, Marian F., Manolis Falelakis, Martin Groen, Rene Kaiser, and Michael Frantzis [2015]. “Experimental Enquiry into Automatically Orchestrated Live Video Communication in Social Settings”. In: *Proc. International Conference on Interactive Experiences for TV and Online Video (TVX ’15)*. ACM, 2015, pages 63–72. ISBN 1450335268. doi:10.1145/2745197.2745211. http://doi.acm.org/10.1145/2745197.2745211 (cited on page 9).

Ursu, Marian F., Martin Groen, Manolis Falelakis, Michael Frantzis, Vilmos Zsombori, and Rene Kaiser [2013]. “Orchestration: TV-Like Mixing Grammars Applied to Video-Communication for Social Groups”. In: *Proc. 21st International Conference on Multimedia (MM ’13)*. ACM, 2013, pages 333–342. ISBN 1450324045. doi:10.1145/2502081.2502118. http://doi.acm.org/10.1145/2502081.2502118 (cited on pages 9–10).

Vconect [2016]. *Vconect: Smart Video Communications*. 11 Dec 2016. http://www.vconect-project.eu (cited on pages 1, 7–8).

VS [2015]. *History of Video Conferencing*. Videoconferencing Solutions. 16 Oct 2015. http://www.videoconferencing-solutions.net/history-of-video-conferencing/ (cited on pages 4, 48).

Wang, Jian, Fuzheng Yang, Zhiqing Xie, and Shuai Wan [2010]. “Evaluation on Perceptual Audiovisual Delay using Average Talkspurts and Delay”. In: *Proc. 3rd International Congress on Image and Signal Processing (CISP 2010)*. Volume 1. Oct 2010, pages 125–128. doi:10.1109/CISP.2010.5646049 (cited on page 11).

Wegge, Jürgen [2006]. “Communication via Videoconference: Emotional and Cognitive Consequences of Affective Personality Dispositions, Seeing One’s Own Picture and Disturbing Events”. *Human Computer Interaction* 21.3 [Sep 2006], pages 273–318. ISSN 0737-0024. doi:10.1207/s15327051hci2103_1 (cited on page 6).

Weiss, Wolfgang, Rene Kaiser, and Manolis Falelakis [2014]. “Vconect - Orchestration for Group Videoconferencing”. In: *2nd International Workshop on Interactive Content Consumption at (TVX ’14)*. (Newcastle, UK). 25 Jun 2014. http://www.joanneum.at/de/digital/publikationen/detail/publicationlibrary/6885.html (cited on page 8).

Weiss, Wolfgang, Rene Kaiser, Manolis Falelakis, and Marian F. Ursu [2014]. “Models for Decision Making in Video Mediated Communication”. In: *Proc. Workshop on Understanding and Modeling Multiparty, Multimodal Interactions (UMMMI ’14)*. (Istanbul, Turkey). Workshop at ACM International Conference on Multimodal Interaction (ICMI 2014). ACM, 16 Nov 2014, pages 45–50. doi:10.1145/2666242.2666250. http://joanneum.at/uploads/tx_publicationlibrary/WEW-UMMMI.pdf (cited on page 11).