Dynamical Aspects of Spatial Audio in Multi-participant Teleconferences

David Roegiers

Promotoren: prof. dr. ir. Dick Botteldooren, prof. Alexander Raake

Masterproef ingediend tot het behalen van de academische graad van
Master in de ingenieurswetenschappen: elektrotechniek

Vakgroep Informatietechnologie
Voorzitter: prof. dr. ir. Daniël De Zutter
Faculteit Ingenieurswetenschappen en Architectuur
Academiejaar 2013-2014

UNIVERSITEIT
GENT

Extended Abstract

Pro Forma – Ghent University

In this thesis the dynamical aspects of spatial audio in multi-participant teleconferencing are researched. We propose several algorithms that determine the sequential rendering of each conferee's 3-dimensional acoustical configuration, after a participant joins or exits the session. Thereafter, conversation test scenarios are devised, simulated and recorded. These artificial conferences are then used to submit some of the rendering techniques to subjective tests, in order gain insight on the Quality-of Experience (QoE) by assessing the user impression.

In the present-day global telecommunications ecology there is an ever-increasing demand for (long-distance) teleconferencing. However, if feasible, people will generally prefer to meet in the same room, instead of a virtual environment. This is as such, because humans do not only communicate via diotic speech signals. Numerous other effects come into play during conversations, such as facial expression, body language, spatial acoustics et cetera. By developing technologies that allow these extra effects to function, the conferences will appear to heave a higher degree of realism and make room for communication proficiency. Not considering visual aspects, there is one real-life auditory facet, that is not reproduced in traditional mono-VoIP or telephone conferences, namely the presence of a 3D-audio environment. When working with dislocated sound sources, the human binaural hearing helps us to better analyse the environment. This relates to the process by which the brain can filter out a specific direction from a mix of acoustic events, cf. the cocktail party effect [3].

Baldis et al. [1] have shown us that implementing spatial voice streams in a high quality and noise-free desktop conference system improves the listener's enhanced memory, focal assurance and perceived comprehension. A clear preference for spatial audio was proclaimed by all test subjects, meaning that besides functional benefits, it provides a better Quality of Experience (QoE). The use of 3D conferee allocation also showed a reduction in attention requirements for speaker identification, which is most useful when handling unknown or unfamiliar voices. Kilgore et al. [2] designed The Vocal Village, a communications tool that allows for real-time spatial teleconferencing over the Internet. Their experimental research shows that this within-the-head localization provides performance benefits in user perception compared to traditional monaural audio conferencing methods. Important to note is that in contrast to earlier work, this experiment was conducted using recording, spatial rendering, and transport methods suitable for the Internet in real time and with readily available peripheral equipment. Quite some research has been performed the past decade concerning the different techniques and methods used to apply spatial audio. Begault et al. [4] determined the relative contribution of head-tracking, room reverberation and individualized-HRTF's to the reduction in localization errors of speech stimuli within an auditory display. In [5] and [6] Hyder et al. looked into different variables, such as the virtual room size,

geometries, HRTF's and listener and speaker coordinates.In addition, Raake et al. presented in [7] and [8] novel test methodologies to assess the conversational speech quality for three-person audio conferences. Scenarios were developed to create structured content for listening and conversational tests. Next to characterizing the conferencing scenarios the users' preferences and abilities to discriminate the quality of different audio conferencing settings (such as bandwidth) in a conversational context was studied.

So it can be concluded that extensive research has been performed concerning the implementation of spatial audio in teleconferences. These were mostly about validation of improvement and static configuration/arrangement options. We did not find rublished work about temporary and transient effects. Contemplating about the progress over time, the dynamical aspects of spatial audio in teleconferences will be the central issue of this thesis. More exactly, we will take on the question of what to do, when conference-specific events occur such as arrivals or departures of conferees. If a (distributed) system, running for example a 4-party session, provides four audio streams, that all contain a different acoustic rendering, it should adapt each of those environments when a participant joins or leaves the session. "Which alterations should be done to the spatial arrangements to provide the best user experience? What framework would suit the most to regulate the changes that in- or exclude participants throughout the session?" As will be seen further there are many different ways to do this.

We start off this thesis by devising a theoretical framework, in which conceptual and mathematical reasoning is used in combination with simulations to develop a selected subset of algorithms, which are to be tested. This process split the system up in two parts. Firstly, a *Sequencing Algorithm* is proposed, that determines the relative order of the participants in the acoustic image. This sequence is parsed to a *Rendering Algorithm* that will determine the exact location of the conferees and their variation over time. The Sequencing Algorithm will be investigated in the following two perspectives. The global considerations look at a *Virtual Meeting Table* that passes the same relative ordering to all instances of the *Rendering Algorithm*. This ensures a consistent formation to all members of the conference. In the individual considerations we optimize this sequence for each participant separately. We end this theoretical analysis with 15 possible combinations/algorithms.

Due to the need for material to exert these algorithms on, we continue with *Conversational Test Scenarios*. This relates to the process of creating structured conversations, that differ in content. We extend previous work, that only looked into static conditions, in order to use this framework for conferences that build up incrementally. An architecture is proposed to systemize the whole process of creating scripts that describe and determine a scenario. These are used to stimulate the voice actors that will simulate the conversation. This is consequently recorded and used later in subjective tests.

Listening-only experiments were performed to gain insights on the relationship between the algorithms and their influence on the QoE. Four algorithms were selected for 20 test subjects using 4 recorded conversations. Questionnaires were presented with the purpose of extracting the user's perception of the QoE through scaled answers. The results of this

experiment did not report statistical significance for user preferences concerning the dynamical algorithm (repeated-measures ANOVA). Our conclusion is two-fold : Either the experiment was not devised well-enough to capture the differences in the user perception for the algorithmic products, or the differences are practically immeasurable and/or vary for each subject, resulting in group preferences. In addition, we notice that investigating the dynamical aspects of spatial audio in teleconferencing is pioneering work and that users might be indifferent which algorithm is used, in contrast with the impression of the static conditions. As a practical take-away, we would suggest developers to select the easiest-to-implement of the tested products, as at this point – where consumers have no experience with spatial audio – it will not weigh on the QoE of the application.

In summary, the following three contributions were made. A conceptual and theoretical framework was developed, that allowed us to propose an optimal and efficient subset of algorithms that determine the transitional dynamics. An architecture was proposed that systemizes the creation of structured content, called conversational test scenarios, that should bring out the effects of the algorithm to a maximal degree. Finally, subjective experiments were performed by which we gained insights on the relationship between the transitional rendering techniques and the QoE.

Acknowledgments

September 28^{th}, 2012
Berlin, Germany

This work has been the icing on the cake of my five years of university studies at *Ghent University*. Obtaining the bachelor and master degree in electrical engineering was definitely not a walk in the park. This thesis symbolizes that statement for me. I learned that decent research requires intensive work and, above all, responsibility. Now, I feel honoured to have gathered an analytical skill-set, that is typically associated with engineers. Additionally, this period was in parallel with my Erasmus exchange in Berlin. As I found myself in a beautiful environment of inspiring people, I felt most happy about the work/pleasure-balance I underwent.

Firstly, I would like to thank my parents, who always supported and encouraged me in my studies, extracurricular activities and time abroad. The independence and trust I received from them, was the perfect incentive to successfully complete my academic course.

I would equally like to express my gratitude to professor Alexander Raake for the warm welcome to his department as a non-native student and the most generous recommendation letter for further studies. When it comes to the research, I want to thank Ing. Janto Skowronek to the fullest. As my active supervisor, I must say I could not expect a better helping hand than from him. Emphasizing 'better' in quality, rather than quantity. He gave me the necessary space for letting my own ideas flourish, while I could always count on his constructive and motivating feedback. Meeting with him felt like discussing with a co-worker, instead of having a down-stream flow, all the while effectively sharing his experience. Whenever time constraints or inexperience were too much of a hazard for me, he was there to help me out, even missing out on quality time with his family.

I was particularly content to work in the TEL-building, which had such a nice atmosphere : an international environment, friendly colleagues, a magnificent view over the city and good-tasting free coffee.

And last but not least, thank you, Debora, for putting up with me in the stressful times.

David Roegiers

Summary

In this thesis the dynamical aspects of spatial audio in multi-participant teleconferencing are researched. We propose several algorithms that determine the sequential rendering of each conferee's 3-dimensional acoustical configuration, after a participant joins or exits the session. Thereafter, conversation test scenarios are devised, simulated and recorded. These artificial conferences are then used to submit some of the rendering techniques to subjective tests, in order gain insight on the Quality-of Experience (QoE) by assessing the user impression.

This work is situated in the area of spatial audio and multi-participant teleconferences. It is widely known that the use of 3D or spatial audio offers substantial benefits to the user experience in desktop conferences. Amongst others, this has been shown by Baldis and Kilgore in [1] and [2]. In today's telecommunications world we witness a continuously expanding network traffic capacity, offering us increasing Quality of Service (QoS) in on-line multimedia. This brings the implementation of spatial audio in VoIP, and similar applications, to the foreground. Extended research has been performed concerning various 3D audio rendering techniques and quality assessment methods. However, up to now, we didn't find any published work on the dynamical aspect of spatial audio in teleconferences.

In the first chapter a theoretical framework is developed, containing a multitude of algorithms that will determine these transitional change-overs. Realistic reasoning, analysis and simulation are the tools applied to design that structure, by two different perspectives : individual optimization and group considerations.

Chapter 3 briefly goes over the implementation options. Some code is delivered for recommendation purposes, but we are far from delivering a working full-end system. In Chapter 4 we construct the content that will be used for the experiments later on. An architecture is developed, that can be used for the creation of conversational test scenarios. These contain a set of rules and instructions, that serve for the simulation of realistic conferences. Quite qualitative recordings are collected, that are used in Chapter 5. Listening experiments are conducted to test 2 different dynamical aspects, where four rendering techniques are rendered on the material. 20 subjects gave feedback via rated questions on the user experience that those algorithms bring about. Through statistical analysis we realized that none of them provided a significantly better QoE, as the variances of the ratings were too overlapping. However, as will be explained in Chapter 6, this is not per se a non-successful realization.

Contents

Chapter 1

Introduction

1.1 Introduction

In the present-day global telecommunications ecology there is an ever-increasing demand for (long-distance) teleconferencing. However, if feasible, people will generally prefer to meet in the same room, instead of a virtual environment. This is as such, because humans do not only communicate via diotic speech signals. Numerous other effects come into play during conversations, such as facial expression, body language, spatial acoustics et cetera. By developing technologies that allow these extra effects to function, the conferences will appear to heave a higher degree of realism and make room for communication proficiency. Not considering visual aspects, there is one real-life auditory facet, that is not reproduced in traditional mono-VoIP or telephone conferences, namely the presence of a 3D-audio environment. When working with dislocated sound sources, the human binaural hearing helps us to better analyse the environment. This relates to the process by which the brain can filter out a specific direction from a mix of acoustic events, cf. the cocktail party effect [3].

Baldis et al. [1] have shown us that implementing spatial voice streams in a high quality and noise-free desktop conference system improves the listener's enhanced memory, focal assurance and perceived comprehension. A clear preference for spatial audio was proclaimed by all test subjects, meaning that besides functional benefits, it provides a better QoE. The use of 3D conferee allocation also showed a reduction in attention requirements for speaker identification, which is most useful when handling unknown or unfamiliar voices. Kilgore et al. [2] designed The Vocal Village, a communications tool that allows for real-time spatial teleconferencing over the Internet. The Vocal Village system uses binaural audio signals to present the voices of individual conference participants from different apparent positions in space by adding location cues to audio streams. Their experimental research shows that this within-the-head localization provides performance benefits in user perception compared to traditional monaural audio conferencing methods. Important to note is that in contrast to earlier work, this experiment was conducted using recording, spatial rendering, and transport

methods suitable for the Internet in real time and with readily available peripheral equipment.

Quite some research has been performed the past decade concerning the different techniques and methods used to apply spatial audio. Begault et al. [4] determined the relative contribution of head-tracking, room reverberation and individualized-HRTF's to the reduction in localization errors of speech stimuli within an auditory display. In [5] and [6] Hyder et al. looked into different variables, such as the virtual room size, geometries, HRTF's and listener and speaker coordinates.

In addition, Raake et al. presented in [7] and [8] novel test methodologies to assess the conversational speech quality for three-person audio conferences. Scenarios were developed to create structured content for listening and conversational tests. Next to characterizing the conferencing scenarios the users' preferences and abilities to discriminate the quality of different audio conferencing settings (such as bandwidth) in a conversational context was studied.

So it can be concluded that extensive research has been performed concerning the implementation of spatial audio in teleconferences. These were mostly about validation of improvement and static configuration/arrangement options. We did not find reported work about temporary and transient effects. Contemplating about the progress over time, the dynamical aspects of spatial audio in teleconferences will be the central issue of this thesis. More exactly, we will take on the question of what to do, when conference-specific events occur such as arrivals or departures of conferees. If a (distributed) system, running for example a 4-party session, provides four audio streams, that all contain a different acoustic rendering, it should adapt each of those environments when a participant joins or leaves the session. *"Which alterations should be done to the spatial arrangements to provide the best user experience? "What framework would suit the most to regulate the changes that in- or exclude participants throughout the session?"* As will be seen further there are many different ways to do this.

Before breaking off, we mention that Appendix A contains a short glossary containing known as well as made-up terminology. We advice the reader to in case of doubt have a look, in order to eliminate any unambiguity.

Chapter 2

Analysis & Development

2.1 Introduction

In this chapter we develop and propose several solutions to allocate the participants of a conference at different spatial audio cues throughout the listener's acoustic environment. Besides the absolute positions, we delve into their mutual relations and their variation over time, when conferees join or leave the conversation. In Section 2.2 we take the decision to split the system in two components, that are each described in the two consequtive sections. The first one (Section 2.3) is fine-tuned via theoretical analysis (Section 2.3.1) and simulation (Section 2.3.2). In addition, some extra variants are proposed intuitively in Section 2.3.3. The second component (Section 2.4) is mainly based on literature (Section 2.4.1) and common sense (Section 2.4.3). We summarize the whole at the end (Section 2.5), to clarify how the two components communicate with each other.

2.2 General Design

The idea of implementing spatial audio into teleconferences comes from the general tendency of making virtual communication sessions ressemblent to real life interactions. For this reason we connect the inner workings of the system with those of an actual round meeting table. Different aspects of such a realistic situation can be overtaken and implemented in our rendering processes for the worse or the better. As we proceed to develop and present several products throughout this chapter, we shall mostly relate to this analogy, hence referred to as the *virtual meeting table*.

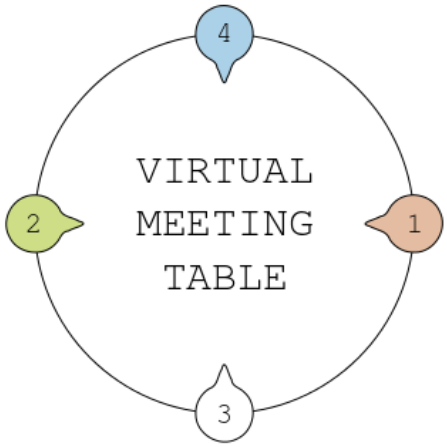


Figure 2.1: Schematic example of conference around meeting table

As a twofold part of the system we define the *sequencing algorithm*, that determines how the conferees are relatively ordered. In other words, this algorithm specifies at which position around the *virtual meeting table* a newly arriving conference participant is 'seated'. Although this scheme does not fix the absolute positions in the acoustically rendered images of the conferees, it does represent a significant property. The output of this algorithm will allow us to, in Figure 2.1 for example, state that Mr. Green (2) will sequentially hear in his spatial environment : Mr. Blue (4), Mr. Orange (1) and Mr. White (3). The sequence also changes over time, as participants join or leave the conference. Let us say a fifth person arrives in the prior example and he or she is given a chair. The relative position he chooses to take at the table, will have an effect on the others. This effect will also depend on how the previous conferees arrived. It is that aspect, the *sequencing algorithm* tries to isolate.

The complementing part, the *rendering algorithm*, computes for each participant the parameters, indicating the talkers' exact positions in his or her acoustical environment, based on the sequence provided by the *sequencing algorithm*. It will attend on building the geometrical configuration in the spatial environment of each conferee over time and assigning the remaining members of the conference to well-defined positions.

Having initiated this framework, we will now explore different options and methodologies, and select a set of end products for further handling.

2.3 Sequencing Algorithm

One property of the *virtual meeting table* is that two neighbouring participants will hear each other from opposing directions. This feature is not noticeable from the perspective

of one conferee, but might when considering both of them. Utilizing this trait in spatial audio teleconferences could have it's benefits, especially if head-tracking and/or graphical visualization is implemented. To extrapolate this effect for multiple users, is equivalent to delivering the same output of the *sequencing algorithm* to all participants. In what follows that output shall be referred to as the global ordering/sequence. This signifies, provided that all heads are directed to the center of the table in the case of head-tracking, that if one hears the other from the left, the other will hear the former on the right at the same angular displacement (see Figure 2.2). In other words, the goal is to offer a common table experience to the entire group of conferees. As we will see further on, given a certain time-varying pattern, some individuals experience better or worse dynamics than others with this condition. However, now we think about the big picture. As a global consideration, the advantage should be evaluated over all participants. Although in this work we only look into static stereo audio reproduction, we strongly believe head-tracking and/or visual representation could reinforce the *virtual meeting table* asset.

It is important to mention however, that [2] proclaimed that user perceptions were found to be greatest when subjects were given control over the conferees' locations. Implementing this would alter the principle of the *virtual meeting table*. Adding such a overwrite mechanism, does not jeopardize the automatic procedure, although the impression of a common table experience can be lost.

Firstly, we work out some theoretical analysis to translate our issue in mathematical terms. Afterwards, *Matlab* simulations are carried out in order to acquire more insight about the performance of each variant of the *sequential algorithm*. For this, the total amount of participants is chosen to be 6. No relevant changes in the inner workings and performance of the algorithm are expected above this number, but we do believe it is ample enough to include the events and effects we wish to investigate. Finally, a small subset of end products is chosen, based on our estimations of user preferences.

2.3.1 Theoretical Analysis

The general procedure is rather simple : when a new partaker arrives, a decision must be taken concerning his seating position around the *virtual meeting table* relative to the other - already present - participators. When it comes to the ordering, no intrinsic difference in pattern can be noticed up to the third person for all possible positioning rules. This can be seen in the user cases depicted in Figure 2.3. Although no actual communication occurs yet, we start, for the sake of completeness, with a single user. His or her position is taken as a zero degree reference. Afterwards all participants are equally and symmetrically distributed over the anti-clockwise 360 degree scale. [1] The second participant is subsequently placed at 180

[1] Please note that this angular enumeration is exclusively inserted for overview and clarity, as we only extract the ordering at this stage, and has nothing to do with the final acoustical positions.

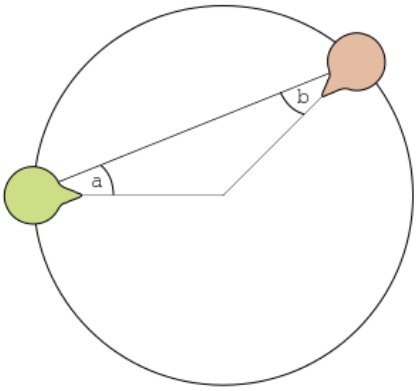


Figure 2.2: Hearing angles of a circular table conversation. Angles a and b are equal.

degrees. For the third arrival we have, in accordance with the current framework, two choices, shown below in Figure 2.3. Nonetheless, the two residual set-ups are essentially equivalent, because all participants have the same two neighbours. Choosing one of the two distributions should not influence the inter-participant positioning relations in a definite way, provided that human-inherent, psycho-acoustical left-right distinctions are not accounted for (a brief word is dedicated to this feature at the end of Section 2.3.3).

To put it in mathematical terms : extracting the order of the *virtual meeting table* is equivalent to taking a subset of all permutations of the set $\{1, 2, 3, ...N\}$ with N being the number of participants. These are from here onwards referred to as orderings or sequences. The elements represent the conferees in their chronological order of arrival. So, the higher the number, the later the corresponding participant joined the conference. These should not be considered as participant identifications, but more as *order tags*. This is in the interest of further proceedings, where the exiting of a conferee is accompanied by a reassignment of these numbers. After removing the departed element or order tag from a sequence, the remainders are reallocated keeping their arrival order in mind and filling the gap. If nr. 5 leaves a conference, structured by the ordering { 1 4 2 3 7 5 6 }, then 7 becomes 6 and 6 becomes 5, resulting in { 1 4 2 3 6 5 }. It is important to realize, that if a certain sequence, that is built up by a fixed process, discards a participant, the new reassigned ordering could very well not be the same as the ordering before the last arrival.

In the particular case of $N = 3$, we find $N! = 6$ permutations :

$$\{1, 2, 3\}\ \{1, 3, 2\}\ \{2, 1, 3\}\ \{2, 3, 1\}\ \{3, 1, 2\}\ \{3, 2, 1\} \tag{2.1}$$

We notice that for example $\{1, 3, 2\}$ and $\{3, 2, 1\}$ describe the same setup, but designate

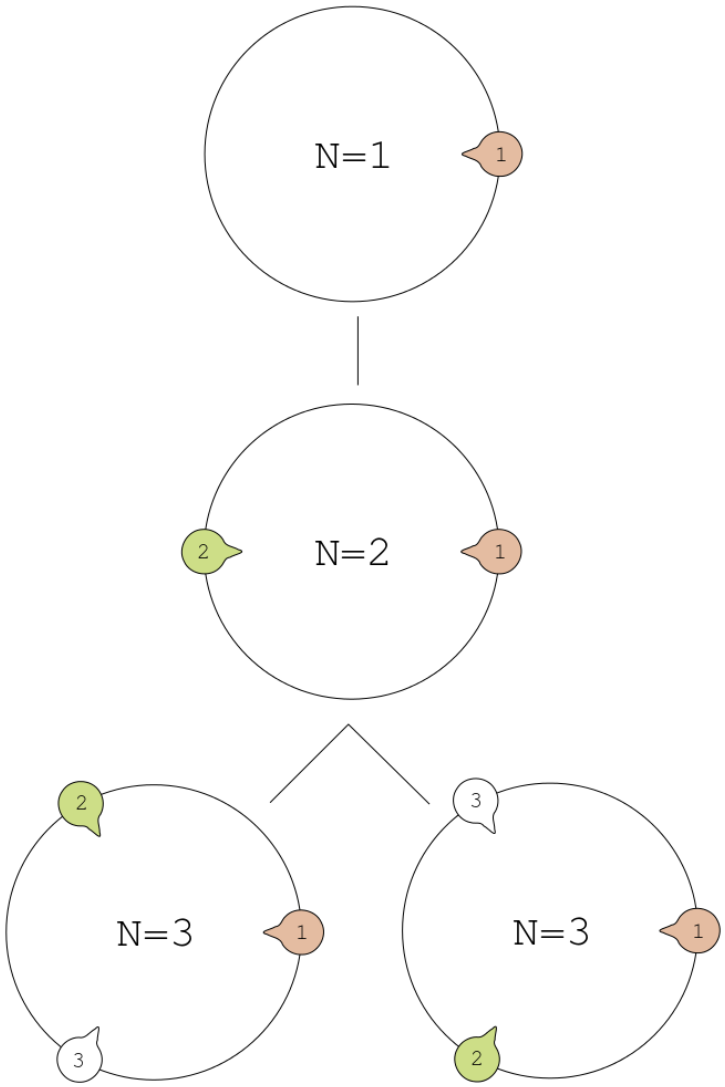


Figure 2.3: Graphical user case of the *virtual meeting table* set-up up to three participants

another participant as reference. I.e. the ordering has just been extracted from another position around the *virtual meeting table*. These ranked sets are cyclic permutations. So from each group of cyclic permutations only one set is to be taken, the rest can be omitted without losing optionality.

To continue the three element case, we identify the two cyclic permutation groups :

$$\{1,2,3\}\ \{2,3,1\}\ \{3,1,2\} \quad - \quad \{1,3,2\}\ \{2,1,3\}\ \{3,2,1\}$$

One permutation of each group is kept, the rest is removed :

$$\{1,2,3\} \quad - \quad \{1,3,2\}$$

We end up with two sequences. These can be interpreted as the two $N = 3$ cases below in Figure 2.3. When one of the two is selected and inverted, we find it to be a cyclic permutation of the other. In reality there is no sensible difference in an inversion of an ordering. It is exactly the same as enumerating the conferees of the virtual table in a clockwise fashion, instead of anti-clockwise. It is just a matter of perception, selecting one of the two will not influence the dynamic behaviour of the acoustic rendering. So the *sequencing algorithm* can allocate the conferees arbitrarily up to the third arrival, without intrinsically influencing the perceived effect.

When the fourth participant arrives, the cyclic and non-inverted sequences are distinguishable for some different configurations. Four objects have $4! = 24$ different permutations, that consist of 6 cyclic groups of 4 permutations each. We can omit half of those, in analogy with previous paragraph, as they are the inverse of the residual subset. This leaves us with three variants. Performing the same analysis for the fifth and sixth participant (see Table 2.1), we end up with 60 alternatives.

These findings are confirmed by the following uncomplicated thinking method. There are 3 distinct positions where the fourth arrivee can take place at the *virtual meeting table*, 4 for the fifth and 5 for the sixth. This gives rise to 3x4x5 = 60 possible solutions, which corresponds to Table 2.1.

Participant	Permutations	Cyclic Permutation Groups	Inversed Pairs	Residu
1	1	1	0	1
2	2	1	1	1
3	6	2	1	1
4	24	6	3	3
5	120	24	12	12
6	720	120	60	60

Table 2.1: Permutational Features of Sets up to 6 Elements

In conclusion, this theoretical analysis permitted us to reduce the number of possible global orderings for a particular number of participants, by eliminating redundant sequences.

2.3.2 Global Considerations

Of all possible ways to allocate users around the virtual meeting table, some may show better characteristics than others. Here we simulate and extract measurements to form an idea about the global set-up methods. To simulate user cases using several versions of the *sequencing algorithm*, a rough prototype of the *balanced rendering algorithm* (Section 2.4.3) is created. In Section 2.4.1 it will be decided to place the talkers in the frontal azimuth half plane of the listener. For each listener the other conferees are distributed linearly over a half circle (180 degrees) in the frontal plane with exclusion of the borders. A graphical example can be found in Figure 2.4. The performance coefficients are measured in angular shifts, as we want to learn about and minimize the total movement of talkers. In order to end up with symmetrically balanced acoustical environments, all present talkers are distributed linearly over each listener's individual view after every step/arrival. One might remark that shifts around 60 degrees are less noticeable then around 0 degrees, being that the human audible localization resolution is not constant. We reject this claim by stating that we will try to mimic this property, when implementing the actual *rendering algorithm* (Section 2.4.2). So the current linear scale is merely a bijection with the human perception range and therefore valid.

In the simulations we will only look into the behaviour of the arrival setup of a conference, i.e. only up until the sixth participant has joined. This is done accordingly because the exiting scenario could take place in any order and will thus on average not differ for several global sequencing variants. With the condition of arranging a direct setup with no intermediate exits for six conferees, the time-dependent build up is entirely specified by the final global ordering. Obtaining the output of the *sequencing algorithm* at stages where $N < 6$, conforms to omitting the still absent order tags from the final global ordering. So { 1 4 3 6 2 5 } would become { 1 4 3 2 } after the arrival of the fourth participant. For what follows in this Section the six-dimensional sequences will represent the entire set-up output of *sequencing algorithm*.

We define the following variables, of which the first one can be seen in Figure 2.5:

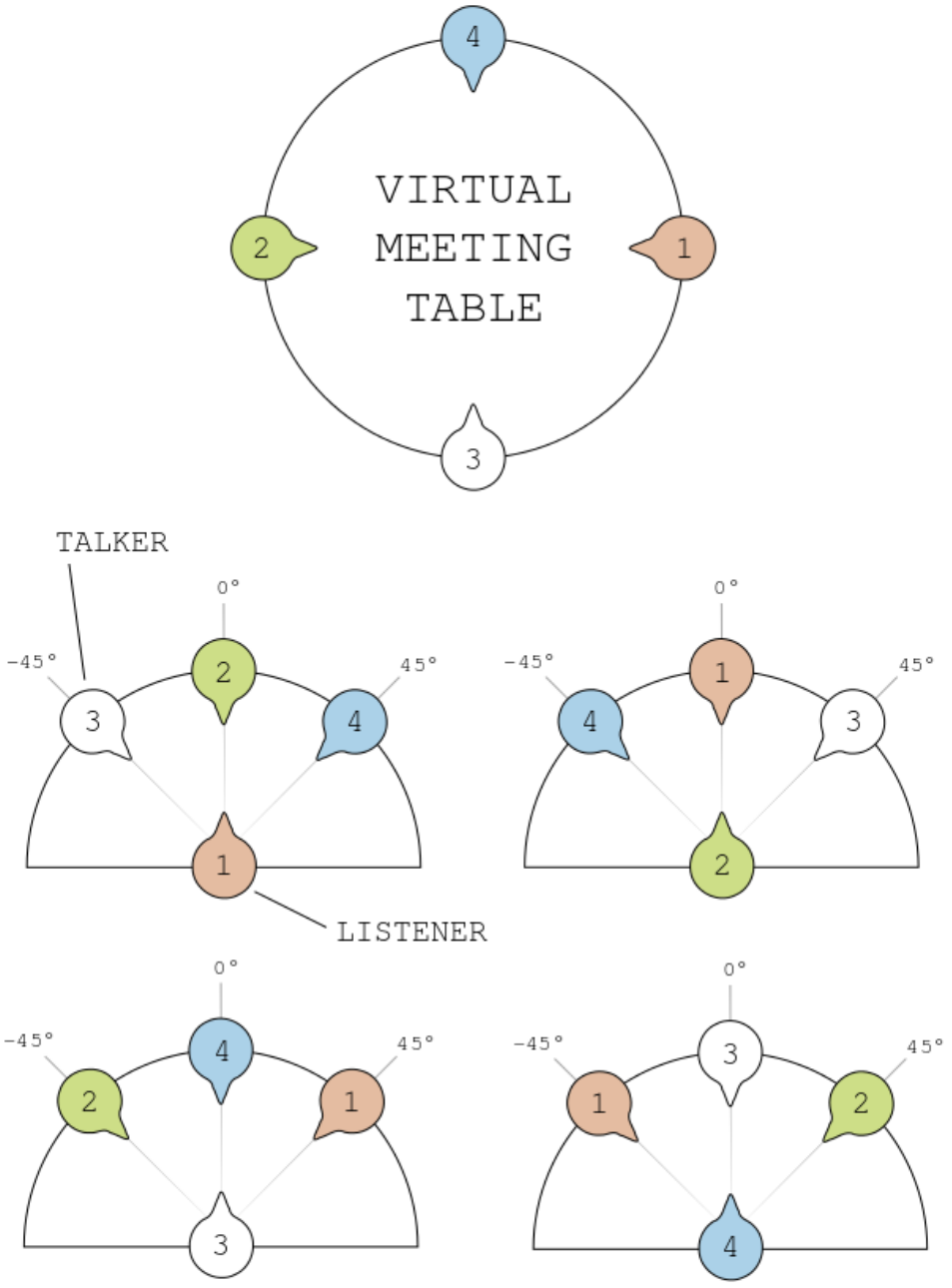


Figure 2.4: Graphical representation of a 4-person *virtual meeting table* (above) and the simplified linear prototype of the *individual algorithm* (beneath), devised for simulation.

$$
\begin{aligned}
\alpha_{sli} &= \text{the angle of talker } i \text{ in listener } l\text{'s setup after step } s, \\
&\quad \text{which corresponds with the } s^{th} \text{ arrival.} \\
&\quad \forall l \leq s, i \leq s, i \neq l, i, l = \{1,2,3,4,5\} \text{ and } s = \{2,3,4,5,6\} \\
d_{sli} &= \mid \alpha_{sli} - \alpha_{(s-1)li} \mid \\
&= \text{the absolute value of the difference between the angle of talker} \\
&\quad i \text{ before and after step } s \text{ for } l\text{'s setup} \\
&\quad \forall l < s, i < s, i \neq l, i, l = \{1,2,3,4,5\} \text{ and } s = \{3,4,5,6\} \\
\bar{d}_{sli} &= \frac{1}{(40)} \sum_{s=3}^{6} \sum_{l=1}^{s-1} \sum_{i=1, i\neq l}^{s-1} d_{sli} \\
VAR(d_{sli}) &= \frac{1}{(39)} \sum_{s=3}^{6} \sum_{l=1}^{s-1} \sum_{i=1, i\neq l}^{s-1} (d_{sli} - \bar{d}_{sli})^2
\end{aligned}
$$

$\bar{d}_{sli}$ denotes the average of all acoustical shifts in the entire set-up, and $VAR(d_{sli})$, the variance of this group. As it makes sense to experience as little angular shifts in the acoustic views of the talkers (in size and frequency) as possible, logically we would want to take end products, having minimal values of the two aforementioned variables. For demonstration, we work out one example for the *virtual meeting table* ordering of { 6 5 4 3 2 1}. All α_{sli}'s and d_{sli}'s are noted in Table 2.2.

We compute the following :

$$\bar{d}_{sli} = 17,775 \quad - \quad VAR(d_{sli}) = 61,82 \tag{2.2}$$

In hindsight, these variables will not help, as they produce the same values for every global ordering. So we need to compute other measurements, that can show us different perceptual

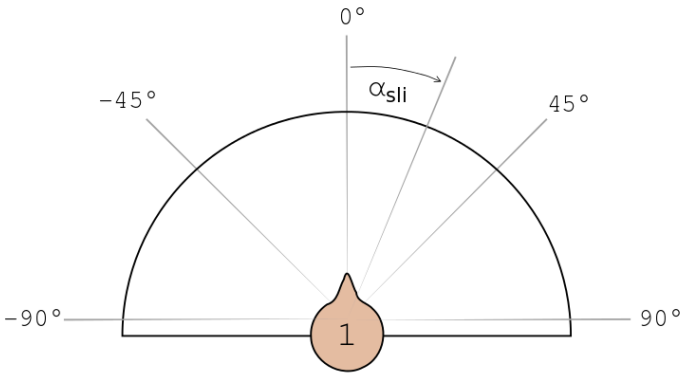


Figure 2.5: Graphical representation of α_{sli}

s	l	i	α_{sli}	d_{sli}
2	1	2	0	-
	2	1	0	-
3	1	2	-30	30
		3	30	-
	2	1	-30	30
		3	30	-
	3	1	-30	-
		2	30	-
4	1	2	-45	15
		3	0	30
		4	45	-
	2	1	-45	15
		3	0	30
		4	45	-
	3	1	-45	15
		2	0	30
		4	45	-
	4	1	-45	-
		2	0	-
		3	45	-
5	1	2	-54	9
		3	-18	18
		4	18	27
		5	54	-
	2	1	-54	9
		3	-18	18
		4	18	27
		5	54	-
	3	1	-54	9
		2	-18	18
		4	18	27
		5	54	-
	4	1	-54	9
		2	-18	18
		3	18	27

s	l	i	α_{sli}	d_{sli}
5	4	5	54	-
	5	1	-54	-
		2	-18	-
		3	18	-
		4	54	-
6	1	2	-60	6
		3	-30	12
		4	0	18
		5	30	24
		6	60	-
	2	1	-60	6
		3	-30	12
		4	0	18
		5	30	24
		6	60	-
	3	1	-60	6
		2	-30	12
		4	0	18
		5	30	24
		6	60	-
	4	1	-60	6
		2	-30	12
		3	0	18
		5	30	24
		6	60	-
	5	1	-60	6
		2	-30	12
		3	0	18
		4	30	24
		6	60	-
	6	1	-60	-
		2	-30	-
		3	0	-
		4	30	-
		5	60	-

Table 2.2: Example of angular positions (α_{sli}) and shifts (d_{sli}) for global ordering : {654321}.

performance values, for each of the 60 residual algorithms (cfr. Table 2.1). In the Matlab function *'calcAngularChanges.m'* (see Appendix B), the content of Table 2.2 is constructed for a given global ordering, upon which the mean and variance of d_{sli} is calculated for step, s, 3 to 6, and also for every listener l. The step-wise statistics are always the same, which tells us, as said before, that the overall average and mean are equal for all orderings. The listener-based computations give slightly varying outcomes though. However, because we find it hard to make a selection based on those hardly performance-revealing numbers, we design two additional measures to continue this analysis.

Upon hearing multiple shifts of a talker's position in an audio conference, the listener could experience a talker that is oscillating back and forth throughout the build-up, one that is always moving in the same direction, or any combination of both. On one hand, one might think that the absence of oscillation gives a constant moving trend, which the listener can get used to and is non-disturbing. On the other hand, the presence of oscillation causes a smaller total angular shift, which is also something we are trying to acquire. Anyway, we believe that the differences in these movements can cause varying perceptions of user experience.

Therefore an indicator of oscillation is created, o_{sl}. Firstly two 5x5-matrices are created. One counts the number of clockwise shifts (CW, corresponds with an angular increase, cfr. 2.5), the other enumerates the anti-clockwise (ACW, angular decrease) movements. The rows denote the listeners and the columns the talkers, which consequently explains that all diagonal elements are meaningless/zero. The dimensions don't exceed 5, because the sixth participant doesn't take part in any shifts. If we were to make these matrices for the example in Table 2.2, we would get the following :

$$(ACW): \begin{bmatrix} 0 & 4 & 3 & 2 & 1 \\ 0 & 0 & 3 & 2 & 1 \\ 0 & 0 & 0 & 2 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad (CW): \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 4 & 0 & 0 & 0 & 0 \\ 3 & 3 & 0 & 0 & 0 \\ 2 & 2 & 2 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \end{bmatrix}$$

Afterwards another 5x5 oscillation matrix is constructed where each element, o_{sl} (with exclusion of the diagonal ones, $s = l$), is the ratio of the corresponding value in the ACW matrix to the one in the CW matrix, or it's inverse if the ratio is bigger than one. Although for some global orderings a certain degree of oscillation is found, the resulting matrix of the current example has all-zero elements. From this oscillation matrix a single number, $\bar{o}^{w}_{sl}$, is extracted for comparison. Once more, we utilize the mean. However, it is important to calculate a weighted averaging, because some elements shift or oscillate more times than others.

$$\begin{aligned}\bar{o}_{sl}^{w} = \frac{1}{40}(&4 * o_{12} + 3 * o_{13} + 2 * o_{14} + 1 * o_{15} + \\ &4 * o_{21} + 3 * o_{23} + 2 * o_{24} + 1 * o_{25} + \\ &3 * o_{31} + 3 * o_{32} + 2 * o_{34} + 1 * o_{35} + \\ &2 * o_{41} + 2 * o_{42} + 2 * o_{43} + 1 * o_{45} + \\ &1 * o_{51} + 1 * o_{52} + 1 * o_{53} + 1 * o_{54})\end{aligned}$$

In addition we also have a look at the residual angular shift of all talkers. This is the absolute value of the difference between a talker's initial position and his or her final one, expressed in degrees. We summarize it in a matrix where the rows relate to listeners and the columns to talkers, giving for the current user case :

$$\begin{bmatrix} 0 & 60 & 60 & 45 & 24 \\ 60 & 0 & 60 & 45 & 24 \\ 30 & 60 & 0 & 45 & 24 \\ 15 & 30 & 45 & 0 & 24 \\ 8 & 12 & 18 & 24 & 0 \end{bmatrix}$$

Finally the same weighted averaging method as above is applied to deliver the mean, $\bar{r}_{sl}^{w}$, for direct comparison of global orderings. In this case we get 37.50 °.

We decided earlier to only use these last two measurements for the selection of the best *virtual meeting table* orderings and thus the global variants of the sequencing algorithm. The maxima and minima can be seen in Table 2.3.

Finally, we will select two end products. We want to minimize oscillation and the total residual angular shift. From the set of 11 orderings with minimal percentage of oscillation (0 %), we select those with minimal residual angular shift, being 30.7 ° :

6 4 5 3 2 1

6 4 5 3 1 2

6 4 5 2 1 3

6 4 5 1 2 3

In analogy, we select from the 4-element set of minimal $\bar{r}_{sl}^{w}$ (9, 30 °), the subset with lowest percentage of oscillation. We quickly find that they all have an equal percentage of oscillation, 7 %.

We end up with 8 end products. In order not to lose ourselves too much in pointless analysis, we decide to arbitrarily select one ordering from both arranged groups. Resulting in

Ordering	Min $\bar{o}_{sl}$	Max $\bar{o}_{sl}$	Min $\bar{r}^w_{sl}$	Max $\bar{r}^w_{sl}$
6 5 4 3 2 1				
6 5 4 3 1 2				
6 5 4 2 1 3				
6 5 3 2 1 4				
6 5 3 1 2 4				
6 5 2 1 3 4	0 %	-	-	-
6 5 1 2 3 4				
6 4 5 3 2 1				
6 4 5 3 1 2				
6 4 5 2 1 3				
6 4 5 1 2 3				
6 3 4 2 5 1				
6 3 4 1 5 2				
6 3 5 2 4 1				
6 3 5 1 4 2	-	22 %	-	-
6 2 4 3 5 1				
6 2 5 3 4 1				
6 4 2 5 3 1				
6 4 1 5 3 2				
6 3 2 5 4 1	-	-	$9,30\ ^\circ$	-
6 3 1 5 4 2				
6 5 4 3 2 1				
6 5 4 3 1 2				
6 5 4 2 1 3				
6 5 4 1 2 3				
6 5 3 2 1 4	-	-	-	$37,50\ ^\circ$
6 5 3 1 2 4				
6 5 2 1 3 4				
6 5 1 2 3 4				

Table 2.3: Orderings with maxima and minima for $\bar{o}_{sl}$ and $\bar{r}^w_{sl}$

the following two global variants :

6 4 5 3 2 1

6 4 2 5 3 1

2.3.3 Individual Considerations

Once again we investigate the sequencing algorithm, but this time applying an individual paradigm without taking the renditions of the other conferees into account. The experience of a *virtual meeting table* is abandoned, because every conferee receives his or her own optimal individual ordering, in contrary to Section 2.3.2, where all conferees are given the same global sequence. Nonetheless, the definition of orderings and sequences will still be utilized. Instead of having a single global ordering, the *sequencing algorithm* will then parse incoherent individual orderings.

To begin we employ a random user case. Let us say a participant joins a teleconference. The formerly present conferees, if any present, may be randomly ordered, because they enter his acoustical image simultaneously. The listener, however, only perceives a talker's presence once he speaks for the first time. To this extent we could implement a *Voice Activity Detector (VAD)* that alters the individual ordering, in favor of making the sequence of talk bursts appear in a certain pattern (from central to sideways for example). This, however, lies outside the scope of this study.

As was wondered in the previous section, whether the talkers should oscillate or move in the same direction over several transitions, we propose two procedures, defining the sequencing algorithm, that should be followed by each of the listeners :

- Upon arrival, the present participants are randomly distributed. The ordering contains the listener's tag in the first position, followed by the arbitrary allocation of the already-present participants. When somebody joins the conversation, he or she is concatenated to the end of the individual ordering. This implies that new arrivers always come in from the left side of the acoustical view, while all other conferees shift in the same direction towards the right shoulder of the listener. If $N = 6$ for example, the individual ordering would be { 1 2 3 4 5 6 } for listener 1, { 3 * * 4 5 6 } for listener 3, ... The asterisks are to be filled arbitrarily by all integers smaller than the listener's tag number. Hereafter this method will be referred to as the *Sideways Sequencing Algorithm.*

- Implementing a maximal effect of oscillation, the same initialization is done as above. Afterwards, the side of insertion is switched for each transition. At the first arrival, the talker is placed on the left, for the next to the right, and so on. This translates in concatenating the first arrivee's tag at the end of the sequence. The second one receives the second position, etc.

We call this the *Oscillating Sequencing Algorithm*. E.g., for $N = 6$, we get { 1 5 3 2 4 6 } for listener 1 and { 4 6 * * * 5 } for listener 5.

It might be interesting to clearly present newly arriving sources to the listener, as he or she must get used to the new presence and voice timbre. To that extent a third alternative is presented, that embeds new talkers centrally :

- Once more, the same initialization process is copied. Thereafter, incoming sources are allocated in the middle of the ordering with exclusion of the first element, in the odd case, and to the left of the middle element, in the even case. It gives rise to an effect where, 'fresh' participants are more centralized and older ones pushed to the side. In accordance with previous examples, we get { 2 3 5 6 4 1 } for listener 2 and { 4 * 6 5 * * } for listener 4. It is named the *Waterfall Sequencing Algorithm*, after the downwards shift movement.

Hereabove we arbitrarily chose for the left side, when an asymmetrical descision had to be taken. This could as well have been the right. Although psychological differences perceived between left- and right-inbound auditory events have been reported by [9], investigating which side would be best is somewhat too laborious.

In the above we solely delved into arrivals. Taking exiting events into account, adds a simple regulation to the procedure, as already described in Section 2.3.1. The departing participant is firstly omitted from the sequence, upon which the participant tags are reassigned in the same order, making sure that the created gap disappears. So { 1 3 5 6 2 (4) } becomes { 1 3 4 5 2 }.

2.3.4 Summary

To summarize, we visually presented the system as a *virtual meeting table*, relating it to ranked sets in mathematical terms, that specifies one output of the *sequencing algorithm* for all conferees. The total number of inherently different ranked sets was reduced substantially, by eleminating redundancy.

Trying to optimize the group experience, we simulated all possible simplified setup scenarios and extracted certain measurements, that served as performance indicators according to our estimations. This led to the selection of two end products, that determine the workings of the global *sequencing algorithm* for a set-up of six participants in the absence of intermediate exits in the light of a common table experience. That part was dedicated to the mutual advantage of all conferees.

For individual considerations, we abandoned the idea of the *virtual meeting table*. This implies that the conferees do not all receive the same global ordering, but different individual ones. Three variants of the *sequencing algorithm* were developed, of which the first two show

quite opposing features : *the Sideways, Oscillating and Waterfall Rendering Algorithm.*

2.4 Rendering Algorithm

Previously we adressed the issue of the chronological, spatial ordering of talkers experienced by each listener of a conference. There still are a lot of design choices that need to be made though, before rendering the final acoustical stereo signals. All sorts of questions arise, such as : *Which technique should be used to create the effect of spatial audio? Where are the talkers distributed in the acoustic environment? ...* Some literature study is done and reflection on the actual transitional dynamical aspects. Those determinations will allow us to define the workings of the *rendering algorithm*, which has a global or individual ordering as input and delivers the placement parameters needed for the final rendition.

2.4.1 Literature

3D audio conferences can be created by spatially distributing loudspeakers throughout the room, each playing a different mono source. On the other hand, it can be achieved with signal processing (HRTF or BRTF) and playback through headphones (Section 5.2.1). The latter is way more applicable than the former, as it only requires stereo transmission and playback (independently of the number of participants) and a single head-phone instead of multiple loudspeakers. Therefore, we apply headphone reproduction. Hyder et al. investigated in [5] and [6] the optimal participant placement in audio teleconferences. Listening-only-tests were conducted for sound localization, performance and the speech quality. The results showed that the best configuration was found when the talkers were placed as around a meeting table. That should not be confounded with the *virtual meeting table* concept. In this situation all talkers were positioned in the azimuth plane in front of the listener. Placing all sources in the frontal half space avoids the well-known front/back confusions reported by [10], also appearing in the analysis of [11] and [12]. The human ear shows better localization traits for interaural differences, instead of elevated diotic variations. Interaural time and level differences (ITD & ILD) are the basic cues responsible for increased localization performance ([13], [14] and [15]). Taking also into account that no head tracking is implemented, which provides improved perceived elevated spatial cues due to head movement ([4, 16]), we take the sensible decision of placing all talkers in the azimuth frontal half plane. To keep the perceptual loudness equal, they are localized on a circle with the listener as centre, so that the distance stays constant.

2.4.2 Auditory Transformation

In addition, research has shown that the localization resolution is the highest in the horizontal plane of the listener at zero degrees (nasal direction, Figure 2.5) and decreases towards $\pm$ 90°. From [17], we quote the following statement : *"Localization blur is the smallest change*

in the direction of the sound source that can be perceived. To measure the latest, we have to search for the minimum audible angle (MAA) or the just noticeable difference (JND), where subjects only have to compare two sound sources and identify only the change of the source direction [18–24].”

The variable property of the MAA suggests that the participants should not be linearly distributed over the $[-90°, 90°]$ angular interval. We justify this decision by the following example : *Talker A is positioned at minus eighty degrees and talker B at zero. Let us say they both undergo a positive angular shift of $x°$. Now x is chosen as such that it is smaller than the JND at minus eighty degrees, but lies above the one at zero. Perceptually, the listener will notice a displacement of talker B, but not of talker A.* In the previous analysis of Section 2.3.2, these shifts were regarded as equal, while their perceived auditory events are quite different. Therefore we devise a bijection between the linear angular positioning scale (seen in Figure 2.5) and the auditory image with non-constant localization resolution. Although Wersenyi enumerates an ample collection of study results in [17], we use a large-scale horizontal-plane localization blur experiment described in [25]. Those results are displayed in Table 2.4. Based on these numbers we will synthesize a mathematical relation between the two aforementioned ranges.

First we take the average of the numbers in Table 2.4 for the positive (right) and negative (left) orientation, resulting in : $5,3°$ Front, $10,5°$ Front Side, $16,93°$ Side. We will denote the linear analytic values (used in Section 2.3.2), as x and the angles, effectively used for the final rendering and taking into account the non-constant spatial auditory resolution, as y. An equally perceived angular displacement around zero and ninety degrees, corresponds to a constant Δx, but Δy should be $\frac{16,93}{5,3}$ times bigger at 90° than 0° or $\frac{10,5}{5,3}$ % of 0° at 45°. This auditory transformation can be expressed in a progressive function under following conditions :

Angle	90° Right	45° Front Right	0° Front	−45° Front Left	−90° Left
Mean localization inaccuracy ($\Delta\bar{\phi}$)	$+4,55°$	$-1,15°$	$-1,33°$	$-4,17°$	$-6,47°$
Standard deviation of $\Delta\phi$	$15,17°$	$10,37°$	$5,3°$	$10,63°$	$18,69°$

Table 2.4: Results of a single-source localization experiment with 900 test subjects in the horizontal plane [25].

$$
\begin{aligned}
&f(x) = y \\
&f(0) = 0, \quad f(90) = 90 \\
&\frac{\delta f(0)}{\delta x} = \frac{5,3}{10,5} * \frac{\delta f(45)}{\delta x} \\
&\frac{\delta f(0)}{\delta x} = \frac{5,3}{16,93} * \frac{\delta f(90)}{\delta x}
\end{aligned}
$$

Four conditions can be fulfilled in a third degree polynomial $ax^3 + bx^2 + cx + d$. This solution is only applicable for the $[0, 90]$ range. So for negative x values, the absolute value must be taken for parsing it through the auditory transformation. We calculated and plotted the solution in *Maple* (Figure **??**).

$$y = 9,45676 * 10^{-6}x^3 + 0,00475894x^2 + 0,495096x$$

2.4.3 Dynamical Aspects

Here, the essential aspect of this thesis is tackled. *How do we alter the acoustical view after conference-specific transitions?* The corresponding dynamical or transient actions, can be categorized in three forms : discrete movement (balanced method), no movement (fixed method) and gradual movement (gradual method).

Contemplating about what would be experienced as smooth and non-disturbing, we come to the two following guidelines :

1) It is interesting to have the participants well-balanced throughout the entire 180 degree horizon, which maximizes the displacement between talkers and, thus, offers the best discernation.

2) We want to avoid as much sudden angular shifting of talkers as possible. Sources that suddenly move around, give rise to confusion and increased talker identification difficulties.

Balanced Method

An optimal balancing of conferees would consist of an equal distribution over the positioning circle of the frontal azimuth plane. We decide to exclude the borders ($\pm 90°$) as available positions, due to the diminished side resolution and front-back confusion. So in case we have an even number of participants (n), which corresponds to an odd number of talkers ($n - 1$), they are placed at $\{\frac{180i}{n}, i = \frac{-n+2}{2}..\frac{n-2}{2}\}$. In the other situation, where n is odd, the conferees

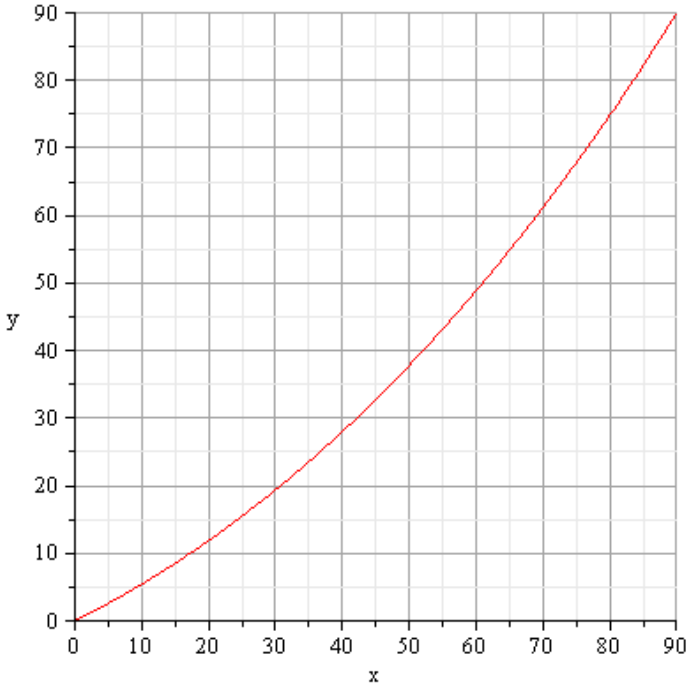


Figure 2.6: The auditory transformation $f(x) = y$

are placed at $\{\frac{180}{2n} - \frac{180i}{n}, i = \frac{-n+1}{2}..\frac{n-3}{2}\}$. Afterwards, we can parse these values through the auditory transformation function, before rendering the signals. This brings us to the first rendering method. The *balanced rendering algorithm*, redistributes all talkers after each arrival or exit, continuously maintaining a balanced setup. This solution entirely respects abovementioned guideline 1. The second is less respected, as each transition is accompagnied by discrete, step-wise movements.

Fixed Method

Answering to the second guideline, angular shifting will be minimized at the expense of balance. We should however think about trying to achieve balance during a maximal part of the conference. Because we imagine that a greater part of the meeting is spent with all participants present, we pursue the final balance. At the start of the conference, a maximum number of participants should be given or estimated. By use of the full ordering provided by the *sequencing algorithm*, we construct the balanced view for when all conferees are present. After the auditory transformation, these positions are held onto. When a conferee arrives, he or she is immediately placed at that saved position. When someone exits, all sources maintain their position, except the one that is dropped. Subsequently, new conferees will get appointed to the positions that were once occupied, but now empty, before resuming the 'planned' set-up (this might imply a violation of the sequencing algorithm). We call this the *fill-up rendering*

algorithm. The importance of this method lies in the fact that no shifts occur at all. Once a talker receives a position, it will not move at any point during the conference. The flaw of this method, lies in the fact that some listeners might start off having all talkers on one side, which could be confusing. If more than the estimated number of participants end up joining the conference, two options present themselves. One consists of placing these extra talkers in between the fixed ones, the other would be to switch over to the *balanced rendering algorithm*.

Gradual Method

Here, a *rendering algorithm* is proposed, that will conduct transitions in a slow, possibly less noticeable manner. After the arrival or exit of a conferee, this method gradually moves the participants from their old position to the new balanced one over a certain amount of time. Actually it is a duplicate of the *balanced rendering algorithm*, with the only differences that shifting is spread over time. It is important to note that it only makes sense to shift a talker if his or her voice is active. Otherwise, there is just silence and a discrete shift when he starts talking at a later point. Additionally, as in general the conferees talk one by one, all alterations are - in the ideal situation - sequential and don't take place at the same time, which would could be too hard to grasp. So, to correctly implement this algorithm, the use of a *Voice Activity Detector* (VAD) is indispensable. The next question to pose is, over what time interval should this shift take place. It is preferable to have all movements take place at the same speed. So the time interval will vary in function of the angular shift, which is maximum 30 degrees. Intuitively we went for 10 degrees per second. This seems not to be too quick, so that the gradual movement is not perceived as a discontinuous one. Neither is it too long, making the shift take place over multiple talk bursts, which is to be avoided.

The procedure can be described as follows :

When a participant joins the conversation, the new balanced configuration is constructed - but not yet implemented - for all conferees. The newly arriving partaker immediately receives the rendering of his balanced configuration. The other, already-present listeners start out with the previous view with the new talker, receiving his final position of the balanced configuration. Once one of the remaining sources is flagged as active by the VAD, it starts shifting towards it's new position, given by the balanced configuration, by a constant speed s. If it stops producing speech before the movement is complete, it retains the position and the shifting is resumed, when the the talker in question speaks again.

When a participant exits the conversation, the new balanced configuration is constructed - but not yet implemented - for all remaining conferees. The source of the departed partaker is omitted in all the individual views and the same shifting technique is carried out as described above.

For clarity, we give a more technical description :

GRADUAL METHOD

---

s represents the predetermined
α_{li} represents the angle of talker i for listener l
t is the current time stamp

ARRIVAL of new n_{th} participant

α^*_{li} = Angle of talker i for listener l before arrival of conferee n,
$\forall i, l \in \{1, 2, ..., n-1\}, i \neq l$
α^{**}_{li} = Balanced configuration angle of talker i for listener l after arrival of n,
$\forall i, l \in \{1, 2, ..., n\}, i \neq l$
$\alpha_{ni} = \alpha^{**}_{ni}, \forall i \in \{1, 2, ..., n-1\}$
$\alpha_{ln} = \alpha^{**}_{ln}, \forall l \in \{1, 2, ..., n-1\}$
$r_c = 0, \forall c \in \{1, 2, ..., n-1\}$
WHILE VAD$_c$ ON $\forall c \in \{1, 2, ..., n-1\}$ and $\alpha_{lc} \neq \alpha^*_{lc}$,$\forall l \in \{1, 2, ..., n-1\} \backslash \{c\}$

$t_x = t$
$\alpha_{lc} = \alpha_{lc} + (t - t_x) * s, \forall l \in \{1, 2, ..., n-1\} \backslash \{c\}$

EXIT of a participant

First we reassign all indices, as to make sure it is the n_{th} conferee that is leaving
α^*_{li} = Angle of talker i for listener l before departure of conferee n,
$\forall i, l \in \{1, 2, ..., n-1\}, i \neq l$
α^{**}_{li} = Balanced configuration angle of talker i for listener l after departure of n,
$\forall i, l \in \{1, 2, ..., n-1\}, i \neq l$
No new angles have to be assigned, before voice activity
$r_c = 0, \forall c \in \{1, 2, ..., n-1\}$
WHILE VAD$_c$ ON $\forall c \in \{1, 2, ..., n-1\}$ and $\alpha_{lc} \neq \alpha^*_{lc}$,$\forall l \in \{1, 2, ..., n-1\} \backslash \{c\}$

$t_x = t$
$\alpha_{lc} = \alpha_{lc} + (t - t_x) * s, \forall l \in \{1, 2, ..., n-1\} \backslash \{c\}$

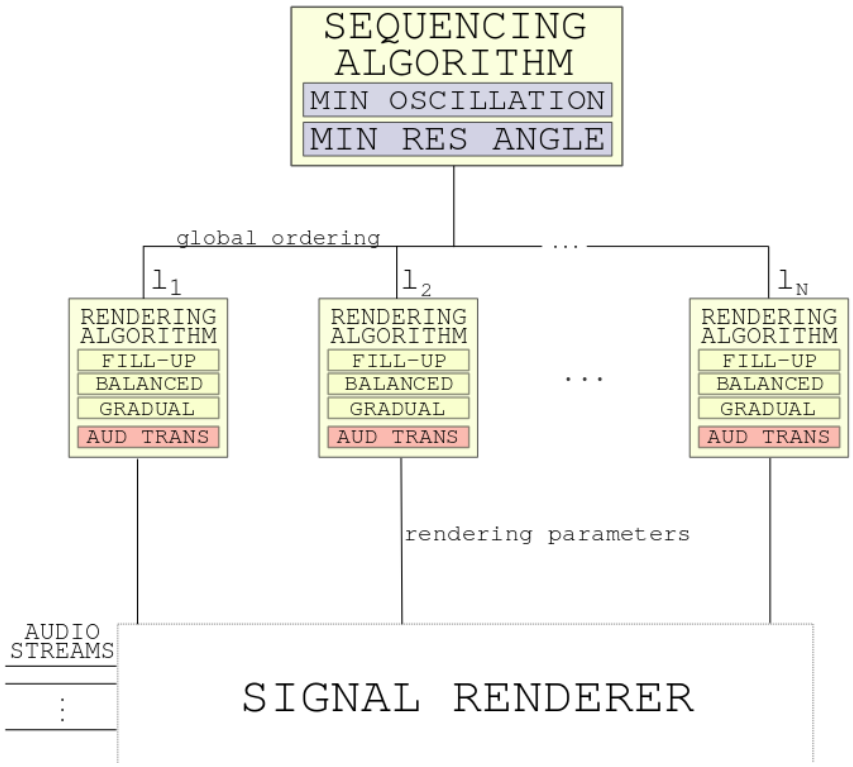
Finally, we remark that it probably is interesting to investigate hybrids and combinations of aforementioned solutions, that depend on whether people are joining or leaving the conference or on the number of participants, etc.

2.5 Summary

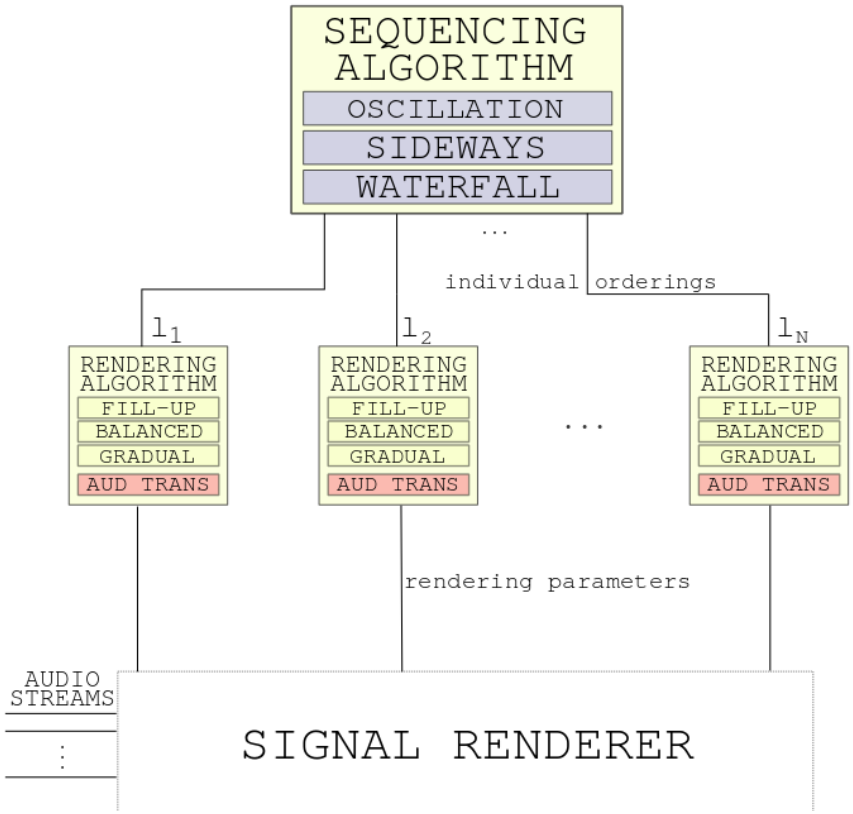
In the beginning of this chapter, the entire regulation system of the spatial allocation throughout a conference was split into two parts. One was conceived in light of the *virtual meeting table*, called the *sequencing algorithm*, that delivers ranked sets concerning the order of the participants. That output changes as conferees arrive or leave. We delivered two variants of the *sequencing algorithm* in the form of a six participant global sequence, offering, according to our estimations, an optimal experience for the entire group of conferees, when a conference build-up occurs without any intermediate exits. In individual considerations, we abandoned the idea of a common table experience and thus make the *sequencing algorithm* deliver several individual orderings, instead of a single global one. Here three products were devised : Waterfall, Sideways and Oscillation.

Each conferee also has a *rendering algorithm* running, which computes for the listener the rendering parameters of the talkers, based on the ordering provided by the *sequencing algorithm*. The transitions (i.e. arrival or departure events) can be handled in three different ways : with discontinuous movement (Balanced), no movement (Fill-Up) or continuous movement (Gradual). In addition, a possible auditory transformation was proposed, taking the variable MAA into account.

Finally, an abstract presentation of the entire system can be found in Figure 2.7a. Both cases are shown. We see that the *rendering algorithm* does not execute the actual signal processing, but calculates the parameters, that specify the positioning for each talker and hence how the rendition should be performed. Consequently some kind of digital signal processing (DSP) unit handles the signal transformations and mixing. Finally we point out, that an extra kind of feedback loop should be installed, if the *rendering algorithm* utilizes the fill-up method, due to the need of a predeveloped ordering containing the information of the final positions.



(a) Summary of the entire system when run as a *virtual meeting table*



(b) Summary of the system with individual orderings

Chapter 3

Implementation

This chapter gives a brief and non-extended overview about implementation options. We shortly look over the tools that are available to put the algorithms into action and propose some - to a certain extent, generic - code, written in *C++*, that handles a few elements of the rendering techniques, established in the previous chapter.

3.1 Soundscape Renderer

Geier et al. [26] developed a real-time spatial audio software framework, that can be used for numerous applications, called the *The Soundscape Renderer (SSR)*. Among others, the environment provides a *binaural renderer* [27], which is interesting for this thesis, as it allows us to work with a simple stereo headphone reproduction. It uses Head-Related Impulse Responses (HRIR) from the *Fabian* mannequin [28] with a resolution of one degree, which is more than sufficient for human perception. One HRIR contains 512 samples ($f_s = 44,1kHz$) and no room reflections. One option to control the acoustical set-up in the SSR would be by use of configuration files, which are loaded in at the start-up. As our algorithms change over time - in the case of *gradual*, a lot -, this is not a feasible solution. Additionally the environment provides a network interface, allowing control using XML-message over TCP/IP. It was recommended to implement the communication in Python. However, in the current version 0.3.3, they declared that this module was still under heavy development.

The SSR also allows the use of customized HRTFs. Begault concluded in [4], that room reverberation provides better out-of-the-head externalization. We tried out the use of the Wittek Binaural Room Impulse Responses (BRIR) [29] and evaluated that they provide better localization than the defaults.

3.2 C++

With the purpose of creating a generic environment and bearing in mind that we do not know how much conferees the session will accommodate beforehand, we write some C++ classes that are able to handle - in theory - an unlimited number of participants. This is efficiently done by the use of doubly-linked lists, that can dynamically add and eliminate elements by simple techniques and efficient memory usage. Only the *balanced* rendering algorithm is implemented. The header files can be found in Appendix C. Additionally, Figure 3.1 shows a scheme of how the classes are connected with doubly-linked lists. For *individual* and *global considerations* (Section 2.3), we created two separate classes. The first one, *GlobTeleConference*, has two elements, *SeqAlg* and *GlobRendAlg* that represent the *sequencing* and *rendering algorithm* respectively. For the *individual considerations* no single sequencing algorithm was needed, so the individual ordering process, being rather simple to program in this case, was included in *IndRendAlg*, the *individual rendering algorithm*. We restrained ourselves from coding the two other rendering methods (*Gradual & Fill-Up*), as it demands more complex and advanced programming - especially when it comes to the input and output methods for the *gradual* option.

3.3 Summary

To conclude this short chapter, we repeat that the surface of the design choices were touched. The motivation lied in the attempt to devise a system that could be used for the subjective tests. However, due to other obligations, this development was traded off for others. Except for the C++ doubly linked lists, we do not provide recommendations for the implementation of a working software environment. Nonetheless, we realized that the *Balanced rendering algorithm* requires much less computing power and code than the two other variants. A back-end framework was proposed for that method, although we expect it has a lot of room for improvement.

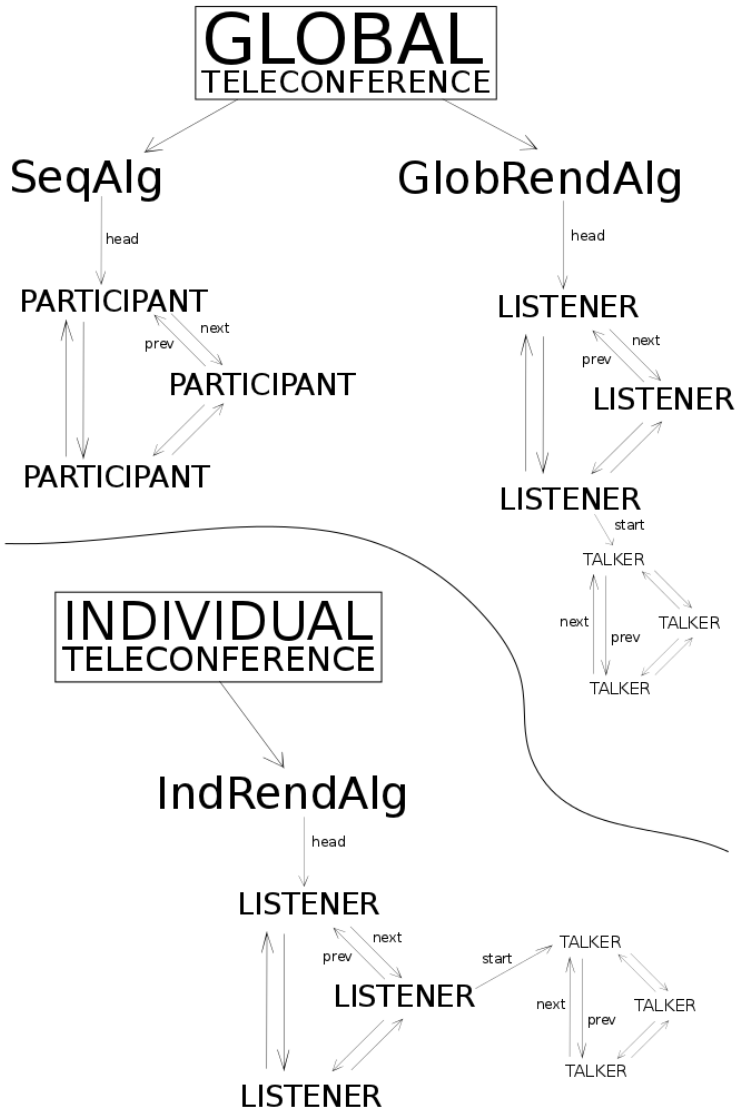


Figure 3.1: Structure of dynamically allocated memory and classes (an arrow corresponds to a pointer).

Chapter 4

Conversational Test Scenarios

This chapter describes how the data was established, used to implement the different algorithms on in our experiments of Chapter 5. Exploring literature, Section 4.1 gives an explanation of what is exactly needed and proposes an architecture to devise the blueprints of the material. Based on that we create five different conversation scenarios used for this thesis. In Section 4.2 we simulate conferences using these conversation scenarios to give rise to recordings. Descriptions are given concerning the recording session set-up, but mainly about the post-processing of the audio files. Finally we evaluate the quality of our scenarios through objective analysis of the uttered speech patterns in Section 4.3.

4.1 Design & Methodology

4.1.1 Introduction

A significant requirement of the experiment is the content. Material is needed to apply the different rendering techniques on. In order to obtain a sufficient amount of data per subject, he or she must be presented more than one rendering technique, using a different recording for each conference. If the same recording is used, subjects would increasingly alter their interpretation as they hear the equal content repeatedly. In other words, it is important to grasp the partaker's attention, by constantly surprising him with new material. Naturally, we can present the same material to separate subjects, as no repetitive judgement or interference takes place. Nonetheless, it is of crucial importance that the different conversations hold the same structure. They should rest on one fundamental pattern, while only differing in descriptive content. This entails that the conferences will seem unique for the listener and that, due to the similar anatomies, the rendering techniques will be reflected to an equal degree. Moreover, in the act of maximizing that degree, we shall search for an ideal pattern, conform with a conventional conference, making the rendering technique influence the subject to the fullest.

From [8] by A. Raake et al. we quote : *'For classical two-person conversations, different types of conversation scenarios have been described in the literature (see [30] for a summary). The main shortcoming of many of these scenarios is that they reduce the naturalness of the assessment situation. Similarly, some of the existing multiparty communication scenarios represent unnatural tasks, and others employ free conversations about pre-defined topics [1, 26, 30–32] that cannot easily be compared with each other. ... In order to reduce some of the drawbacks of (dialogue-type) conversation tests, the SCTs (Short Conversation Test scenarios) developed by S. Möller [33] represent real-life telephone scenarios like ordering a pizza or reserving a plane ticket. They lead to natural but semi-structured, comparable and balanced conversations of approximately 2 to 3 minutes duration.'* They applied the three-person conference test scenarios, loosely contingent on [33], on conversational tests to among other things provide recordings of conferences for later use in listening tests. For our LoT's we need six-party (actually five as will be explained hereunder) conference recordings, having a constructive pattern, of which all phases need to contain speech of all interlocutors. We quickly came to the realization, we would need to record these ourselves, as no published work has been found for these sort of tests. However, the 3CTs (3-person conversation test scenarios) created in [8], presents fruitful groundwork to base ourselves on.

Finally, we mention that, according to our simulations in the second chapter, we want to test six-party set-up conferences, with no intermediate exits. The test subject needs to be one of those conferees. However, since we conduct pre-recorded LoTs, he or she can not interact with the others. This means that the recordings only need to be five-party scenarios, rendered and presented to the test subject, as if he or she were a member of a six-party session. To have as many dynamical transitions as possible we choose our test subject to be the first arrivee of the conference.

4.1.2 Architecture

We define three architectural layers, forming a framework to build and devise the test scenarios : the log layer, the function layer and the content layer.

Log Layer

The log layer defines the fundamental structure of the conference and must be identical for each conversation. The objective here is to create a structure that will erect the rendering technique to the listener in an optimal way. Firstly, the conference is segregated in phases, that are separated by transitional events, being exits or arrivals. Every phase is split up in one or several logs. In these experiments there are five phases, incrementing from one to five participants. The more conferees are present, the longer a phase should last, in the interest of letting everyone talk sufficiently for the change in configuration to be heard. In pursuance of making the recordings realistic, it is probable that there also simply is more to say among

numerous interlocuters. Each phase consists of as many logs as the number of participants. The topics are tagged by the number of people engaged in the discussion (monologue, dialogue, trialogue, quadrilogue and pentalogue). The first phase starts with a monologue. Afterwards, each log is incremented by one talker until all speakers are involved. The newly arrivee takes part in all logs of a phase. Finally, we add another pentalogue at the end of the last phase, reserved to close down the conference. A schematic presentation of the log structure can be viewed in Appendix D. We make sure that this exact pattern is answered to by all test scenarios.

Function Layer

The function layer will specify the interactivity of logs. It allocates speech bursts to specific participants that collectively form a log and describes by use of a keyword the conversational function of these speech bursts. In the functional layer, five different kinds of logs are described. A *default log* consists of following speech bursts, in more or less consecutive order : a *demand*, *constraints* and *conflicts*, *solutions* and *conclusions*. In theory, the *default log* can be as long as needed, involving an unlimited amount of interlocuters. The *out-of-topic log* starts with a *launch* and is followed by *free talk*. The latter form can be allocated to several speakers at the same time. This type of log permits some more randomized, less controlled opinion-influenced talk, hence the name. It assists the transitions, making them smoother and less forced, because there is no conclusion. Otherwise it could be considered fake if a participant arrives, just at the time a topic has been resolved. So as a rule, we proclame that each event in the scenarios are to be preceded by a *out-of-topic log*. Additionally, there is also an *interrupted default log*, which differs from the *default log* in the fact that it is split up over two phases by use of *interruption and pick-up*. It's existence is due to the inability to resolve a discussion by absence of a conferee, that will join at a later point in time. The *ciao* log is a a simple mean to close down the conference, where all talkers exchange a brief farewell expression. Finally, while all other logs need at least two interlocuters, a monologue will always be a *hello log*, a stand-alone speech burst, where the person in question gives a formal introduction about his function and greets the other participant. Although this is a non-realistic announcement in a conference, it is necessary for the contextual integration of the listener.

Content Layer

In the content layer, we essentially produce distinct conversational test scenarios. Based on the residue of the function layer, descriptive content is added to the whole. We start off by describing the context of the conference and the roles of each participant. Subsequently for all topics, each statement (the content layer equivalent of a speech burst) receives meaning, according to it's function described in the layer above, by abbreviated bullet-point description. As in [8], we do not want to produce a written script, so no exact formulations have to be made.

Just a minimal amount of instruction, from which no informational conflicts are expected, should suffice. This will make the actors (which are the people who will execute the scenarios) think for themselves, adding naturalness to their speech interaction. Given the difficulty to generate suitable content for a function layer product, altering small positions and functions of speech bursts is allowed and recommended, as long as the log layer pattern is still respected.

The entire architecture is graphically summarized in Figure 4.1.

4.1.3 Products

For the experiments of this thesis, we create five different scenarios. It is complicated to simulate personal relations via scenario bullet-points, so we attempt to make the content as formal as possible. To this extent, a business context seems as the right choice. As the tests should be conducted in mother tongue, the applied language was German. Here, we shortly describe the context of the five scenarios in English :

- **Purchase of land** : A steel processing business plans to expand by building a remote factory. It is in the process of searching for a parcel of land to construct their second affiliate. Members of the management corps discuss several options with two real estate agents. In the end they agree on a date for visiting the properties.

- **Acoustics Conference** : Several professors of German-speaking universities discuss on how to organize a day at an acoustics conference in Geneva about automotive noise reduction. After agreeing on who of them takes which time slots, they talk about different speakers they might invite to fill up the afternoon.

- **Travel Magazine** : A travel magazine urgently needs a region to cover in their next edition. Several employees meet with an independent experienced traveller to make a decision. In addition, the planning and first arrangements concerning the excursion for making the report.

- **Festival** : The organisational committee of a large-scale festival gets together to share detailed information and make small decisions concerning the overall progress and preparation.

- **Product Presentation** : A car rental company is interested in purchasing a new piece of informational hardware from a start-up to install in their cars. Both parties make a conference call to get acquainted, answer some questions and fix a meeting for an integral product presentation and Q&A.

For a detailed insight on how these scenarios took shape, we refer to Appendix D.

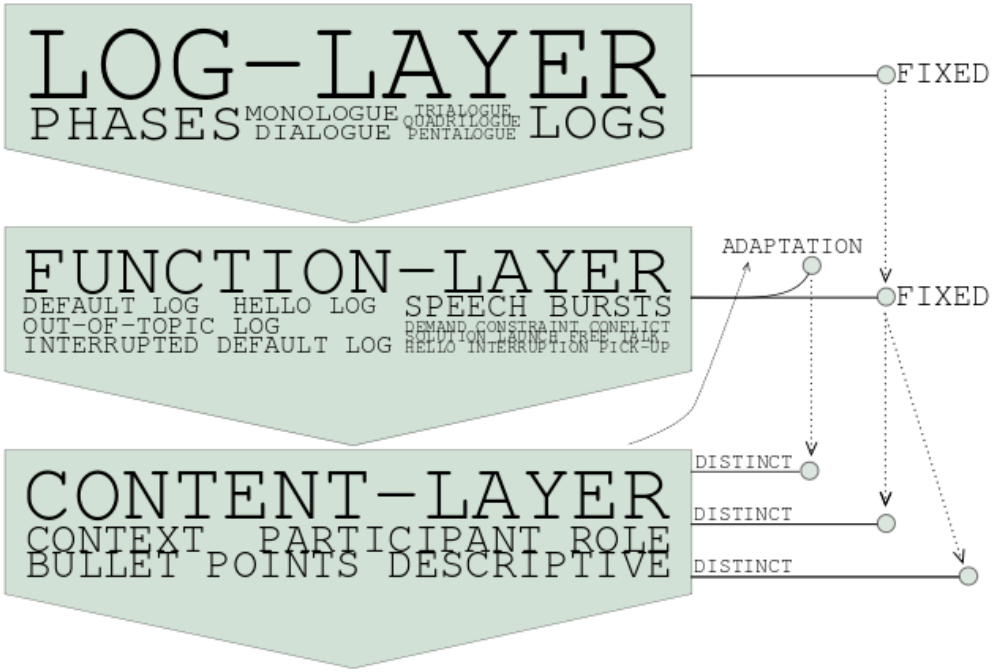


Figure 4.1: Summary of conversational test scenario architecture

4.2 Recordings

The next step consists of executing these scenarios to make recordings for the LoT's (see Section 5.1.3). After post-processing these conversations, we can make measurements of similarity, expressing the effectiveness of how well the different stimuli gave rise to the desired structure.

4.2.1 Set-up

The recordings were made in a isolated noise-attenuated experiment chamber. For them to be valid, it is important that subjects do not have visual, nor acoustical contact, except for the conference system. Therefore all seats were directed towards the outside, big sound-damping boards were placed in between speakers and headsets were used for audio reproduction and recording. These three arrangements also limit cross-talk, which could be easily removed by silence insertion if the actors don't speak simultaneously.

Two different types of headsets were used : two *Beyerdynamic DT 790* and three *Beyerdynamic DT 290*. Each headset's microphone was amplified with a *4-channel SM Pro Audio Q-Pre*. The analogue output signals were amplified by a *Millenium HA4*. Five male colleagues served as voice actors. Each of them were handed the scenario for the first time. They were asked to quickly go through it, just before performance. While the arrival order of the roles were attached to the seats, the colleagues were asked to move one seat clock-wise after each scenario, so that all of them received each position and role once.

All five recordings took place in one session and were completed in a single trial. At the end the first recording (*Purchase of Land*) was reperformed, as the initial one was of significant longer duration than the others. In retrospect, we assert that it served as a training session for the actors to develop their sense of interactivity and get accustomed to the scenario formulations. To that end we take the second execution of the *Purchase of Land* into practice and omit the initial 'training' recording. Additionally, two interruptions or retakes were done in the other scenarios, due to mistake and humorous aspects.

4.2.2 Post-Processing

Before employing these recordings for subjective testing, some audio processing and editing must be done, to obtain purer sound files. All audio files are stored in the uncompressed, raw wav-format. The operations are all done using the *Audacity 2.0.1* free-ware package and in following order:

1. Cutting & Editing

In this step, begin and end silences are cut out, split audio files concatenated and retakes left out. Additionally, we shortened awkward silences, of which we felt they were caused by the

task distraction, until they, according to our intuition, were not bothering any more. Two seconds of silence were added in between transitions for clarity and the playback of an arrival signaling tone.

2. Normalization I

Firstly, a manual and intuitive normalization process is performed. Due to the use of two different headsets, the signals are contaminated by a variation in bandwidth. This statement is based on a clear auditory impression, which is surprising as the frequency responses of both headsets' data sheets seem equal [34, 35]. Signals of similar intensity with different bandwidths often have other perceptual loudnesses [36–38]. So all tracks are amplified until they have a seemingly equal overall loudness. In case clipping might occur, we attenuate the high-peaked phonemes. This should not alter the perceptual event of the word too much, as loudness tends to be assessed over the duration of several utterances of speech [39].

3. Noise Reduction

A non-negligible amount of noise was present, especially in the silent parts. Therefore the *Noise Removal* effect of *Audacity* developed by *Dominic Mazzoni*, was applied. For each individual recording we searched through visual analysis of highly amplified speech-inactive segments for a noise profile, which was then used for the noise removal. Some high-frequency cut-off was noticed in the residue, but we estimated the signal being of better quality then the noise-contaminated original. In the process, all processing parameters were shifted towards their highest value.

4. Silence Insertion

Most non-speech parts contained disturbances, caused by the voice actors, such as coughing, breathing and paper flipping. Also, some acoustical interference was noticed from the head-set's playback into it's microphone. For those reasons, we decide to replace the speech-inactive parts by silence. This is easily done by the software's *Silence Insertion* function. The start of the speech bursts are often preceded by a talker-induced breathing inhalation, that can last up to one second. As this can be seen as some kind of charging for air, it is considered as speech activity and never replaced by silence.

5. Normalization II

As the previous normalization phase, was done separately for each conference, we wish now to bring all recordings to the same level. This process is done in *Matlab* using *Lu Huo's* implementation of the *Active Speech Level Measurement following* [40] delivered by [41]. For

each conference we compute the speech level (given in dB) and select the minimum. Afterwards, all signals are attenuated to that level by following factor :

$$x = 10^{(ASL_{minimum} - ASL_{current})/20}$$

We chose to attenuate to the lower bound, instead of amplifying to the upper bound, so that digital clipping would be avoided.

4.3 Evaluation

4.3.1 Quality Categorization

The five scenarios, established by the given framework, gave rise to five recordings of simulated conferences. The quality and usability of these products is determined by three main factors. One is related to it's capability of erecting the effect of the rendering technique in the listener. We discussed this aspect in the *Log Layer* in Section 4.1.2. The second concerns the reality and genuineness of the content. It is important that the subject believes what is being said and is emerged into the situation as he or she would be in everyday life. Non-credible content or utterances that remind the test listener of the circumstances' simulating nature, must be avoided as much as possible. This task, depending on the *Content Layer* and the actor's abilities, is not uncomplicated and has to be assessed by intuition, as no ad hoc objective evaluation exists for that matter. The third factor, which can be measured objectively, regards to which degree the speech-activity patterns of the conferences are similar. We focus on the third aspect to determine how well our scenario designs lead to consistent recordings.

4.3.2 Graphical Interpretation

A first and simple feature is the overall conference duration (Figure 4.2). The variation interval is enclosed by 357 and 430 seconds.

Another duration measure that can be looked into is the phase duration. As defined in the *Log Layer*, the phases are separated by conference-specific transitions. In Figure 4.3 we can see that the scenarios lead to more or less similar phase durations. The higher the order of the phase, the longer it's duration, which makes sense due to the increased amount of participants and logs. We clearly notice the increasing variation, as a consequence of the raised amount of content material to be discussed. An alternative way to represent this data, would be to create box plots (Figure 4.4).

The next step is identifying the speech bursts for all recordings. The variety of their duration tells us how the listener will be exposed to the multi-directional sources and can illustrate (dis)similarities between the different conversations. The identification contains the duration of the talk burst, the actor's identity, the corresponding recording, the participant's

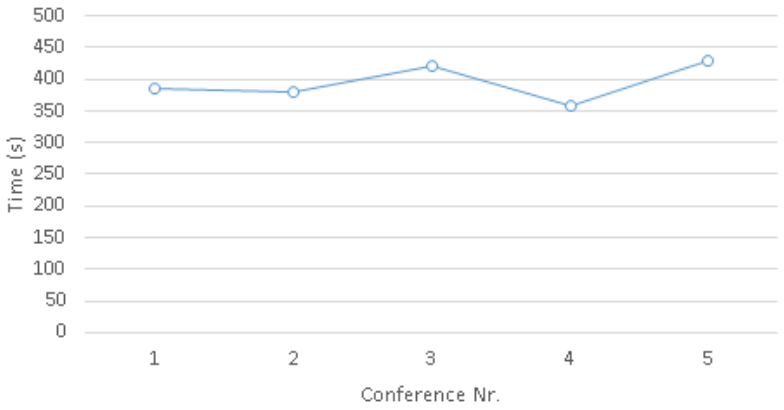


Figure 4.2: The total duration of each conference - Acoustic Conference (1), Purchase of Land (2), Festival (3), Product Presentation (4), Travel Magazine (5)

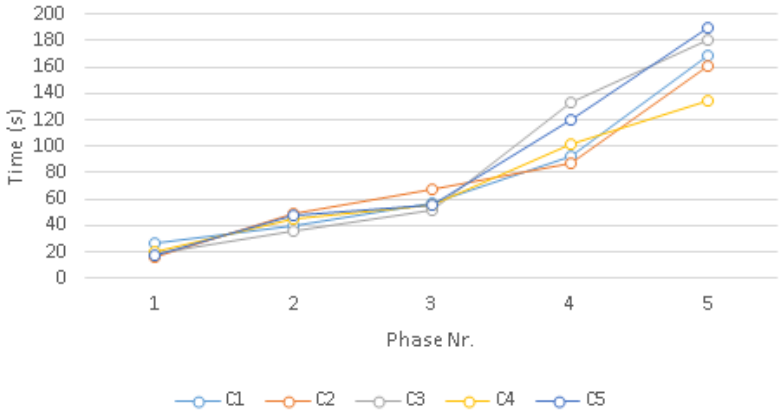


Figure 4.3: The duration of the consecutive phases for each conference - Acoustic Conference (C1), Purchase of Land (C2), Festival (C3), Product Presentation (C4), Travel Magazine (C5)

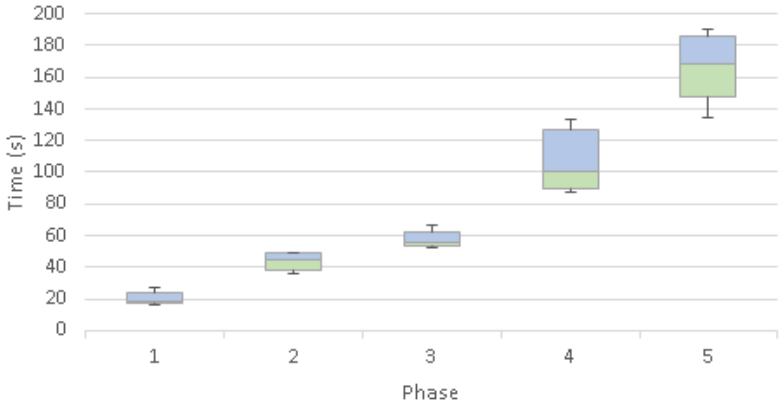


Figure 4.4: The duration of the consecutive phases for each conference in box plots - Acoustic Conference (C1), Purchase of Land (C2), Festival (C3), Product Presentation (C4), Travel Magazine (C5)

identity (in accordance with the arrival order of the conference) and a time stamp. We note here that due to their brevity, acknowledgements are not considered. Hereby, we mean statements containing only one or two words such as *'Hallo!'*, *'Alles Klar!'*, *'Ok!'*,... The rule of thumb is that speech bursts, shorter than one second, are not considered. We first give a quick look at a histogram (frequency distribution) containing all elements, having a mean of about 7 seconds (Figure 4.5). We notice a positive skew, meaning that relatively short statements are compressed around 2-4 seconds. The longer utterances are more spread out in time. This is a such, because the former is limited by a lower bound of about one second as a necessity to express the shortest sentences. The latter has no real limit and can thus spread as much as needed. The far outlier at 27 seconds is an introduction burst of a conference, where the first conferee gives some explanation concerning the content of the scenario. This can take highly varying amounts of time, depending on the contextual situation. At this point, the rendering technique hasn't really done anything yet, so we can safely omit that value.

Now we want to get an idea whether the talk burst duration is strongly influenced by the actors or not. As each actor took up all possible posts in the arrival order once, the effects of the behavioural differences due to the relative arrival position, are averaged out. Figure 4.6 shows us that in general the actors talk for a fairly similar amount of time. In addition we see again the skewness from Figure 4.5. We can also have a look at the total amount of time the actors spoke throughout the five conferences. In Figure 4.7 it is normal to find variance per conference for each actor, however the spreads seem to overlap reasonably well. This leads us to stating intuitively, that the third quality feature of a conversational recording (in correspondence to the description given at the beginning of this section) does not depend too much on the actor.

Another important representation are the box plots of the talk burst duration sample for each conference, that can be found in Figure 4.8. Just as for the actors, we find the histogram's skewness again. For the rest these distributions seem quite similar, which is an extra indication for structural scenario resemblance.

Finally, some attention is given to the different participants. As explained before, the first arrivee of a scenario is participant nr. 1, the second arrivee is participant nr. 2, etc. Figure 4.9 and 4.10, respectively represent line plots for the quantity and cumulative duration of speech bursts per participant. The first figure shows a slight downward tendency. The content of the second however seems to be more randomly scattered, while we do expect to find that negative slope, because it sounds logical that the more times someone speaks, the longer time he or she speaks. This indicates some room for improvement.

4.3.3 Suggestion for Improvement

The number of times someone speaks can easily be controlled by the *function layer*. It differs from the desirable value, because alterations were made to provide suitable content and/or the

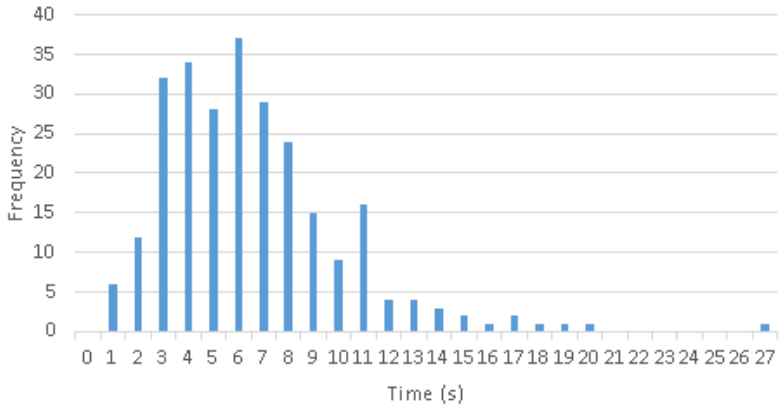


Figure 4.5: Frequency distribution of all speech burst durations (the x-value of a bar denotes that the bin lies in between x and x+1)

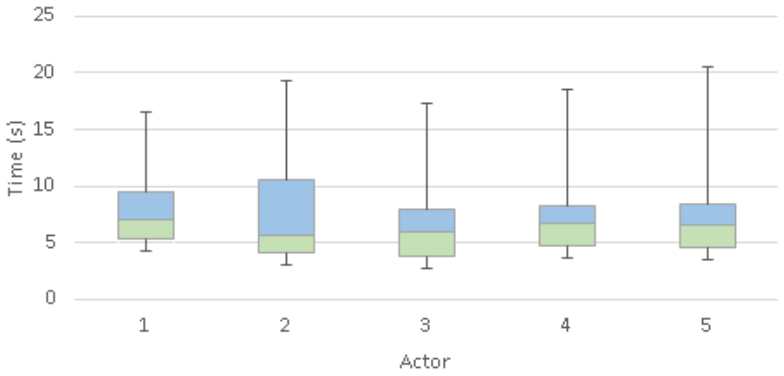


Figure 4.6: Boxplots of the talk burst durations for each actor

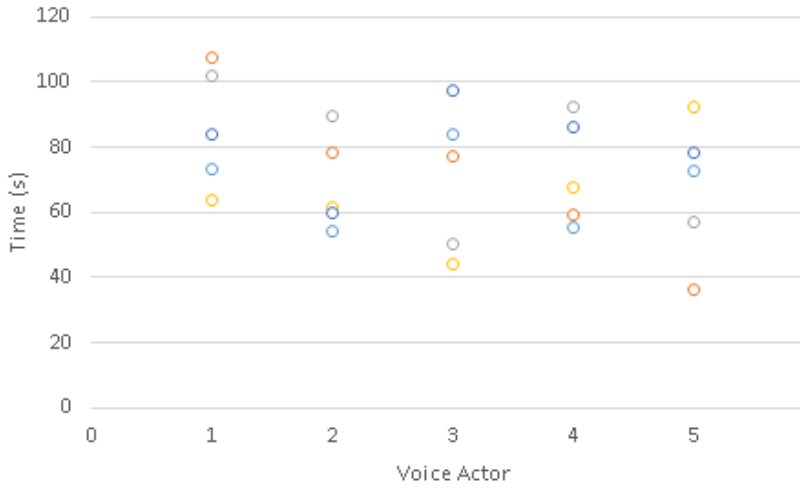


Figure 4.7: Total amount of uttered speech per voice actor and conference

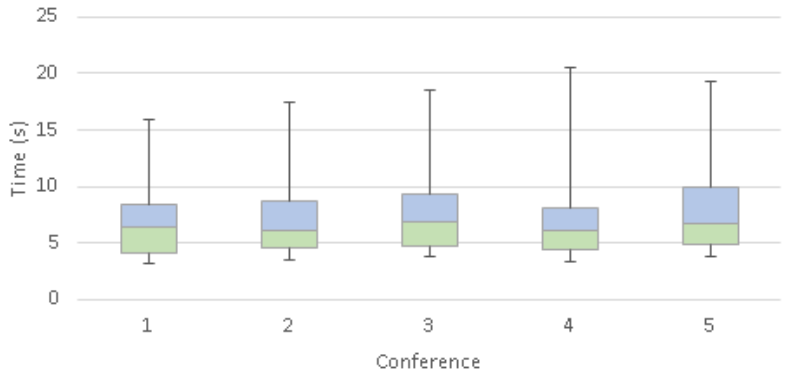


Figure 4.8: Boxplot of speech burst durations for each conference

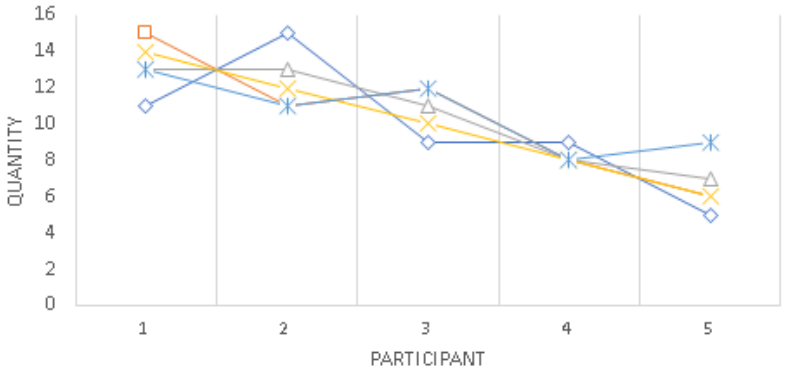


Figure 4.9: Number of speech bursts per participant per conference

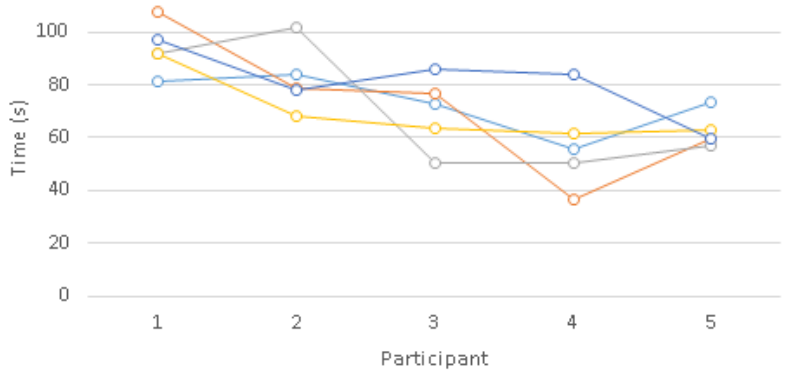


Figure 4.10: Cumulative speech duration per participant for each conference

actors change it up a bit. The length of the talk bursts, which is dependent on the *Content Layer*, were not really anticipated in an attentive manner. One suggestion to do this though, is to devise the scenario word for word in the form of a script up to the point that a requested distribution of words or letters has been achieved and then form the bullet-point notations, based on the written sentences.

4.4 Conclusion

In this chapter we created the content for our subjective tests (Chapter 5). Conversational test scenarios were created as stimuli for voice actors to simulate conferences, saved in recordings. As the structural similarity of these conferences is an important feature to obtain good and valid results, we presented an architectural framework to build these scenarios. Afterwards the recordings undergo a series of digital signal processes in order to polish them for experimental use. To get a notion of how useful that structural production scheme really is, we objectively analysed the morphology through measurement of speech bursts. Looking at some plots we realize that the actors do not influence the patterns too much, but there is some room for improvement when it comes to the speech duration, that might be a bit too cumbersome though. Overall we believe these recorded conversations will serve well for the subjective tests and will quite probably not be a cause of poor results.

Chapter 5

Subjective Testing

In this chapter we put our previous developments to the test. Several procedures were devised that should have different influences on the user experience. As it is not feasible to determine their qualitative worth through objective methods, we turn to subjective assessment. The challenge lies in delivering well-devised experimental work, in order to extract a valid notion about the effect that different techniques exert on the end user. Section 5.1 will address the issue of the objects that should be focussed on. Section 5.2 describes in detail the set-up of the experiment. Before summarizing the entirety, Section 5.3 is devoted to result analysis.

5.1 Introduction

5.1.1 QoE vs. QoS

Due to the increasing importance of overall user quality in technological applications, we dedicate a brief segment to *Quality of Experience* and *Quality of Service*. In the field of multimedia assessment there exists somewhat uncertainty about these two terms, that are getting coined more often than ever. Therefore, before use, clear definitions are cited :

Quality of Service (QoS) *The collective effect of service performance which determines the degree of satisfaction of a user of the service.* - [42]

Quality of Experience (QoE) *Degree of delight of the user of a service. In the context of communication services, it is influenced by content, network, device, application, user expections and goals, and context of use.* - [43]

Although the analysis of Chapter 2 was based on metrical estimations, we can not evaluate each algorithm in a single comparable performance coefficient. These proceedings lie thus in the domain QoE. To that extent, we search for the influence of different dynamical spatial

audio algorithms on the degree of user delight in a multi-participant acoustic teleconference service. The following experiments are to be categorized as utilitarian and subject-related. The quality is assessed by the subject/user's opinion, to whom the object, a rendering technique, is exposed to. He or she must provide a description of the quality by inserting ratings to a fixed series of questions. This corresponds with extracting b_0 in Figure 5.1, representing an overview of the human quality assessment procedure.

5.1.2 Goal

The objective is to simulate conference sessions that are as realistic and similar as possible, for multiple test subjects. By approximately reproducing identical test conditions and solely adapting the rendering techniques, the different QoE-reports should give insight about our proposals. Three key aspects for gaining valid results, are :

- **The total number of test subjects** : the more people we include in the subjective tests, the more our samples will, statistically speaking, approach the actual distribution - provided that the other two aspects are fulfilled. With fewer research subjects, the conclusions have a higher probability of being randomly biased.

- **The ability of recreating the same test conditions** : the subject is (non-)subconsciously influenced by everything in his or her surrounding. The unattainability to perfectly reproduce a test environment (due to, e.g. varying participant's mood), should partly be compensated by leaving the controllable conditions unaltered, such as signal loudness for example.

- **Asking the right questions** : it is crucial to construct a good questionnaire and rating scale for extracting the effective user experience. Whether their exists a 'perfect' survey is a philosophical question, what matters is that we approach it as much as possible.

In the experiment set-up it is important to keep those three elements in mind and look for the right trade-off between their acknowledgement and practical feasibility.

5.1.3 Test Methodology

Basing ourselves on the works of A. Raake in [7] and [8], the first decision to be taken is whether conversation or listening tests should be used. The first one consists of simulating an on-line conference with several test subjects, interacting with one and other guided by protocols and descriptions described by the researcher. This method produces a more pragmatic and valid simulation, but is very complex to conduct. The listening tests, having a lot less hazards for complications than the previous, are a simplified off-line assessment method, where a single test subject listens to and evaluates pre-rendered recordings. Other advantages of conversation over listening tests are summarized e.g. in [33]. Due to time, test subject quantity and organisational constraints we chose the listening-only test methodology (LoT).

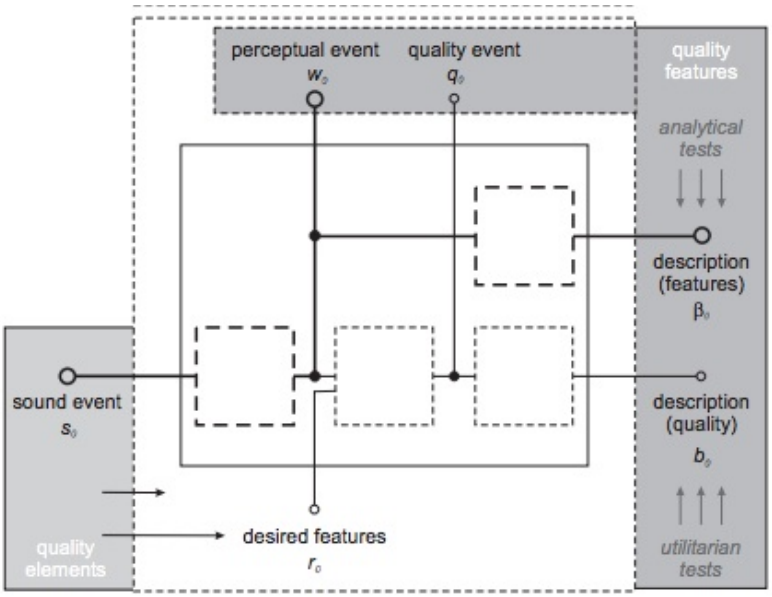


Figure 5.1: Block scheme of quality assessment from Raake's [44] based on [45].

5.1.4 Object

The choice of conducting LoT's has an important consequence : the global effects of hosting a *virtual meeting table* can not be put to the test. The only way to test a common table experience, is by running conversational tests, which is not feasible as mentioned in previous Section. This unfortunately entails that we cannot submit the methods devised in 2.3.2 to experimental work. In the LoT we focus on the individual experience of the user and, thus, when it comes to the *sequencing algorithm*, we employ the techniques defined in Section 2.3.3, namely the (individual) *Sideways, Oscillation and Waterfall Sequencing Algorithm*. In addition, the *rendering algorithm* also has three different options : the *Fill-Up, Balanced and Gradual Rendering Algorithm*. So we have two test conditions or dimensions, each having three possibilities. Every combination of those is a unique, applicable rendering technique, worth testing independently. There are $3 * 3 = 9$ alternatives.

In Chapter 4 only five conference simulations were recorded, meaning we shall have to select a subset out of all 9 possibilities. Per experiment session, one recording will be used to train the test participant, so that he or she can get used to the feeling of (static) spatial audio. Afterwards, only four rendering techniques can be investigated. There are two test conditions per algorithm. The **first** dimension addresses the shifting behaviour given by the *rendering algorithm* : we want to research whether it is better to have no, gradual or sudden movement. [1] The **second** dimension accords to the individual *sequencing algorithm*. For feasibility, the two extreme conditions, when it comes to shift orientation, are selected, namely *Sideways & Oscillating*. This gives rise to the four objects, given by Table 5.1. The first three combinations

[1] Obviously it is better to have no movement, but is it worth the balancing trade-off ?

allow us to study the shifting attitude, as the first dimension condition is kept constant. The third and fourth rendering technique can be compared for the movement orientation. In conclusion, four products are selected for testing wherein two aspects or dimensions are researched.

5.2 Experiment Design

5.2.1 Rendition

A non-negligible task comprises the rendering of the residual stereo headphone signals. The rendering techniques must be combined with the recorded conferences to provide pre-rendered test stimuli. The first arising question is which combinations of algorithm and recording are to be selected and in what order. This is highly important as one can imagine that in general the subject will experience the first stimulus quite differently than the last one. Therefore, we use one of the many 4x4 Graeco-Latin square arrangements.[2]

$$\begin{array}{cccc} A1 & B2 & C3 & D4 \\ C4 & D3 & A2 & B1 \\ D2 & C1 & B4 & A3 \\ B3 & A4 & D1 & C2 \end{array}$$

Let us say the letters denote the rendering algorithm and the ciphers the recording. Then the columns represent consecutive stimuli, exposed to a single subject. Each row identifies a session. As more than four sessions will be directed, this arrangement will be used several times. The first subject will hear combination A1, B2, C3 and D4 consecutively. The seventh participant shall hear configuration D2, C1, B4 and A3.[3] Explaining the importance of the Graeco-Latin square, we see that each conference and algorithm receives each position in the column-based stimuli order exactly once and that each combination occurs a single time. From

Nr.	**Sequencing Alg.**	**Rendering Alg.**	Name
0	Waterfall	Gradual	GRADWATER
1	Oscillating	Fill-Up	FILLUPOSC
2	Oscillating	Balanced	BALOSC
3	Oscillating	Gradual	GRADOSC
4	Sideways	Gradual	GRADSIDE

Table 5.1: The four test combination objects. For the training session (nr. 0) the Waterfall is chosen, because it is not used in the other combinations, and Gradual was chosen arbitrarily.

[2]It does not really matter which one of the more than 1000 possibilities is applied, even though the indices are not assigned.

[3]It is important to mention that before these significant sequences, the subjects listen to a training session. It is not taken into account in the GL-square, because it has no experimentation value. It serves merely as preparation, not making it less necessary.

that, the ordering and interdependent effects will statistically be averaged out. Finally, we must select one of the five recordings for the training. Naturally we want to take the worst one. Instead of basing ourselves on Section 4.3, we chose the *Acoustics Conference* for it's inferior quality of credibility. Assigning the indices randomly we get following groups of stimuli in Table 5.2.

In Chapter 3 implementation proposals were given, though a complete, ready-to-use system was not provided. Initially we planned to work with the *Soundscape Renderer* [26], however the implementation of the Gradual Rendering Technique would be too cumbersome without the use of a network interface, which was not set up due to a know-how limitation. We decided to implement *Matlab* code for the rendering of the final stereo audio files. After attentive listening and thoughtful consideration, we decided not to apply the auditory transformation, as seemed a bit too radical. In Appendix E the code can be found, together with a short explanation.

5.2.2 Assessment

Now that the stimuli are properly devised, the next step consists of suitably extracting the QoE-perception of the subjects. Their experience satisfaction is assessed by probing them with concise questionnaires. It speaks for itself that at least one interrogation is needed after each conference, that will be referred to as the *Final Questionnaire*. We are interested in the dynamical spatial audio aspects, however it is best to poll about other effects as well. There are three different kinds : overall impression, cognitive load and technical quality. One question will be from the first category and is quite self-explanatory. Two cognitive load questions will be posed :

- one about the concentration effort
- the other concerning the ease of identifying the talkers

Most of the queries will be of a technical quality nature. This should not be taken too literally though, as they are very intuitive :

- 1 about the signal quality (in the ideal case the answers of this control question are constant, as the signal quality does not change)
- another control question about the **static** spatial audio aspects
- polling about the new assigned positions
- querying their opinion about the changes in positions
- two extra redundant versions of the two above-mentioned **dynamical** spatial-audio-related questions.

This amounts to a total of 9 questions. As the transitions contain the facets we wish to

Session Nr.	Stimuli Nr.	Algorithm Name	Conference
1	0	GRADWATER	Acoustics Conference
	1	GRADSIDE	Purchase of Land
	2	GRADOSC	Festival
	3	BALOSC	Product Presentation
	4	FILLUPOSC	Travel Magazine
2	0	GRADWATER	Acoustics Conference
	1	FILLUPOSC	Product Presentation
	2	BALOSC	Travel Magazine
	3	GRADOSC	Purchase of Land
	4	GRADSIDE	Festival
3	0	GRADWATER	Acoustics Conference
	1	GRADOSC	Travel Magazine
	2	GRADSIDE	Product Presentation
	3	FILLUPOSC	Festival
	4	BALOSC	Purchase of Land
4	0	GRADWATER	Acoustics Conference
	1	BALOSC	Festival
	2	FILLUPOSC	Purchase of Land
	3	GRADSIDE	Travel Magazine
	4	GRADOSC	Product Presentation

Table 5.2: Features for each group of stimuli

learn about and each changeover differs in shift amplitude and quantity, we want to examine each phase separately. Therefore, two questions (nr. 3 and 6)[4] of the *Final Questionnaire* are duplicated to form an *Intermediate Questionnaire*. A transition also includes the time spent in the residual state, so the first intermediate questionnaire should be posed just before the arrival of the third talker, in contemplation of the transition of one to two speakers. Another three should be submitted just before the arrival of the fourth talker, the fifth and after the conference is finished. The probing structure and it's goal is once more summarized in Figure 5.2.

When it comes to the answers, an *Absolute Category Rating* scale is used, ranging between zero and six, each representing opposing extreme adjectives. The questionnaires can be seen in Appendix F.

5.2.3 Test

Subjects

A total of 20 subjects took part in the experiment. Being a multiple of four, this ensures that all stimuli are tested an equal number of times. Only one participant had previous experience

[4]We limit it to two only, as not to defocus the subject too much, so that he or she remains immersed in the experience

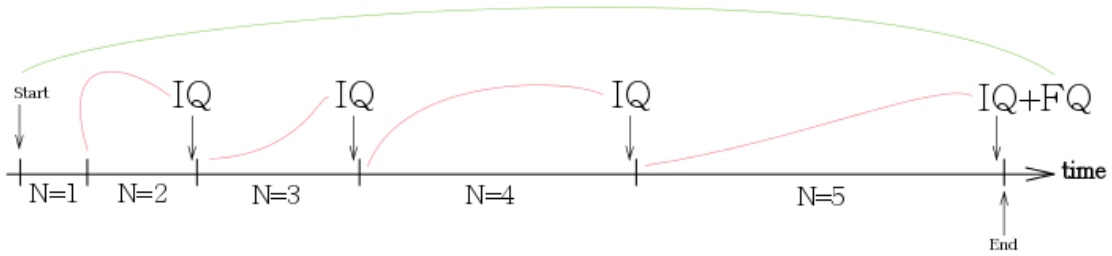


Figure 5.2: Chronological representation of the questioning method. The coloured lines indicate which events the questionnaire should be investigating.

with spatial audio. The group consisted of 11 males and 9 females and was aged between 21 and 41 with an average of 26. We additionally asked whether they had done conferences with multiple participants.[5] Four of them reported negatively.

Set-Up & Process

Figure 5.3 depicts an overview of how the experiment was set-up. The experimenter resided in a different room than the subject, so that the latter one was not bothered and could reach a maximal level of relaxation. The experimenter controlled the playback of the stimuli via the laptop. Both parties communicated by use of a simple intercom system (headset on one end, microphone + speaker on the other). The exact used hardware components are identified in the diagram.

The participants were given a cash effort compensation of € 10. At the beginning they were handed a two page introduction and explanation document about spatial audio and the dynamical effects we were investigating. Afterwards, the experimenter explained the structure of the stimuli and questionnaires. Additionally, during the entire experiment, a form was laid out in front of the subject describing the process structure and showing a graphical representation of spatial audio as a reminder. The end of each partial extract of a recording is indicated by a distinct beeping noise, upon which the subject fills in the *Intermediate Questionnaire* and gives brief notice, when the next phase can start.

The tests went by without complications, except for the fact that the speaker sometimes produced some clicks and pulsating noise. As this was only noticed after a while, we decided not to alter the set-up.[6] Some participants reported after the test, that they took these noises

[5] A singular connection with more than one person on the one end of the line is also considered as a multi-party conference.

[6] These noises really resembled the sound of a pen being tapped on a table. So although it was clearly hearable from the beginning, we interpreted it as a subject-induced tick.

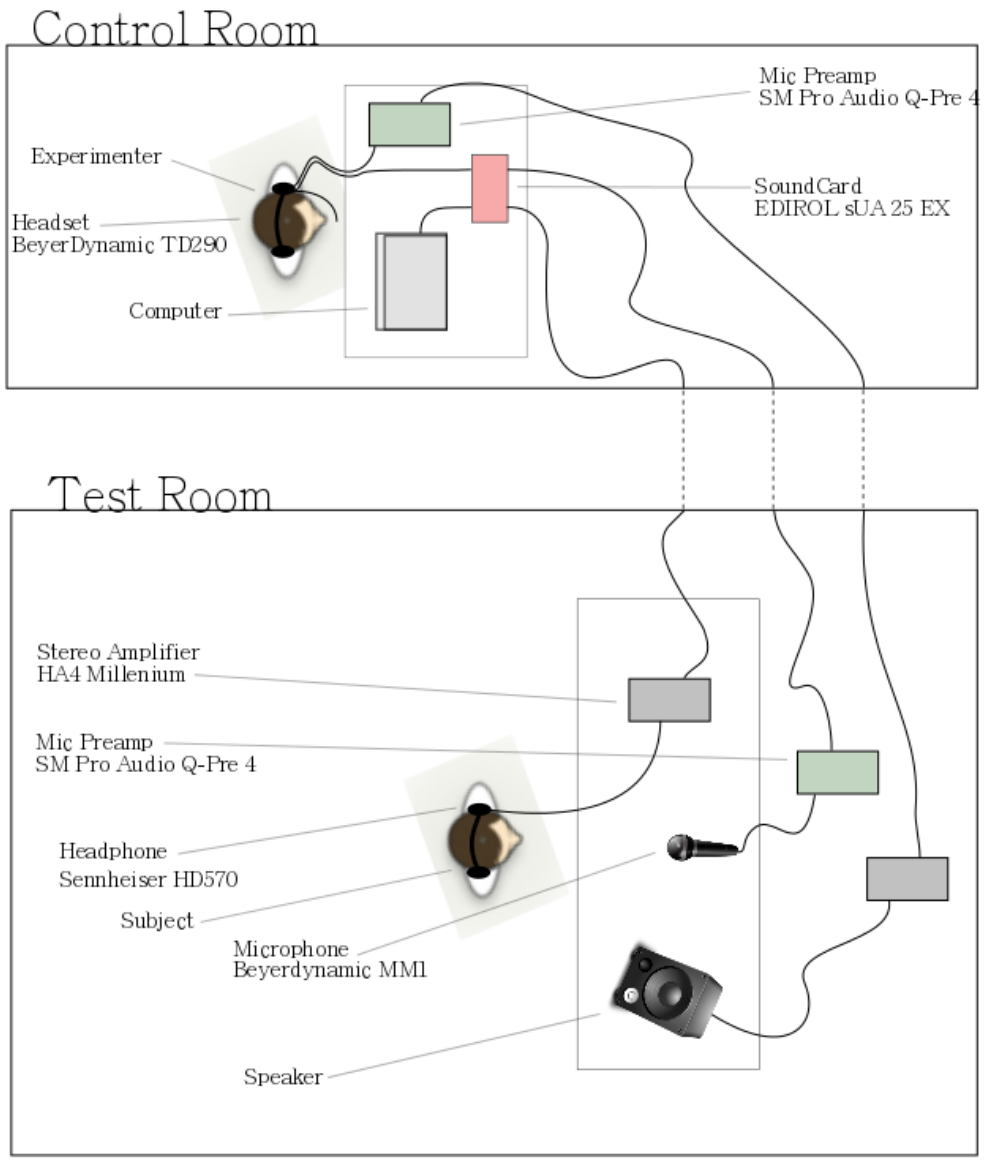


Figure 5.3: Diagram of the experiment set-up.

into account for the evaluation of the sound quality. For that reason, the final control question on signal quality can not be used for the result analysis.

Finally, from brief informal conversations with the test subjects after the experiment we noticed that a majority preferred the method where talkers were shifted in from the side. They however didn't report any perceptual differences between the gradual and discrete *rendering algorithms*. These statements have no concluding worth whatsoever, but they can help us to look in the right direction.

5.3 Results

Plots and statistical tests will be the main tools in our attempt to extract findings and recommendations about the proposed rendering techniques. The analysis of the questionnaires' answers is divided into eleven steps and were all done using the software package *SPSS Statistics*.

1. Error bar plots of final ratings

In this step the error bar plots (mean + 95% confidence intervals) are constructed for each question of the *final questionnaire* comparing each different algorithm (not stimulus). No big differences were found in the diagrams, from this we expect no statistically significant results. The plots for question 3.1 and 3.3 (Appendix F) are shown in Figures 5.4 & 5.5.

2. Error bar plots of z-values

When computing the z-values of the results, to improve sensitivity, we don't see any noticeable improvements (Figures 5.6 & 5.7).

3. Repeated-measures ANOVA of final ratings

Repeated-measures analysis of variance (rANOVA) [46] is the equivalent of the one-way ANOVA, but for related, not independent groups. It is a commonly used statistical test for analysing different test conditions - in this case rendering techniques -, that are repeatedly exposed to several subjects. Applying it to the ratings of each final question separately, we find no significant differences. Looking at all those questions, with exception of the one about *signal quality* as explained before, the Mauchly's test evaluates valid sphericity for all. The rANOVA test results in values between 0,147 and 0,884, which corresponds to no significant differences (¡0,05) between the algorithms.

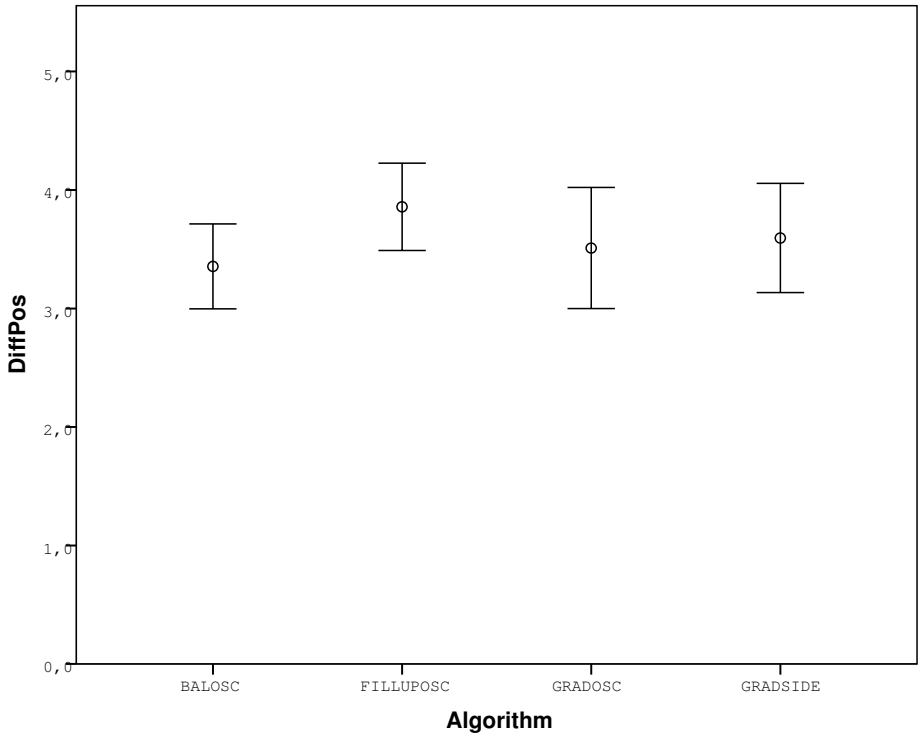


Figure 5.4: Error bar plot for the *differences in positions* rating.

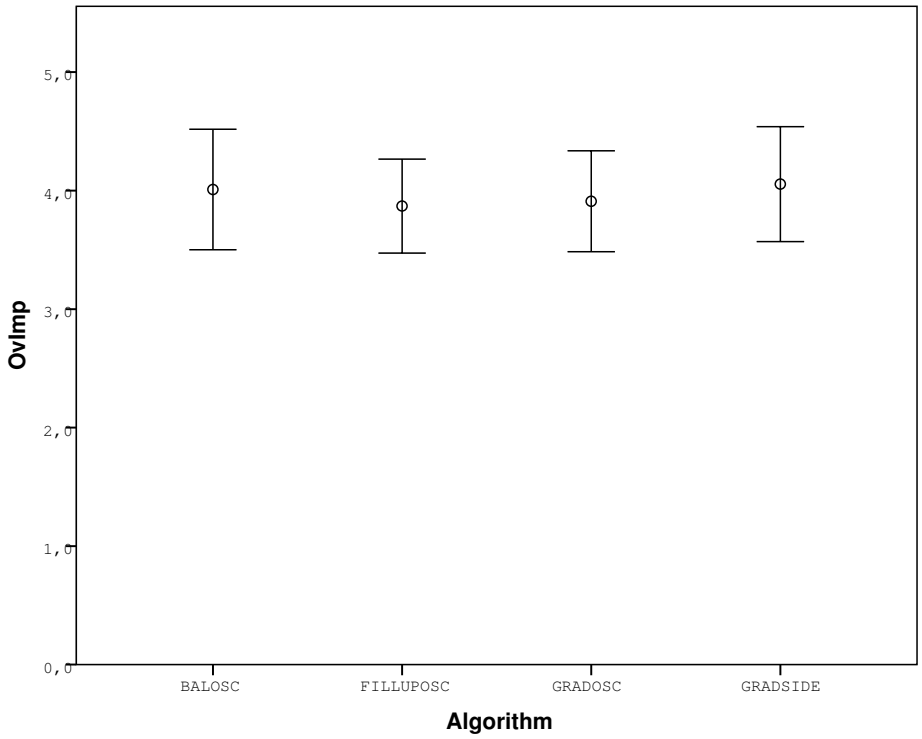


Figure 5.5: Error bar plot for the *overall impression* rating.

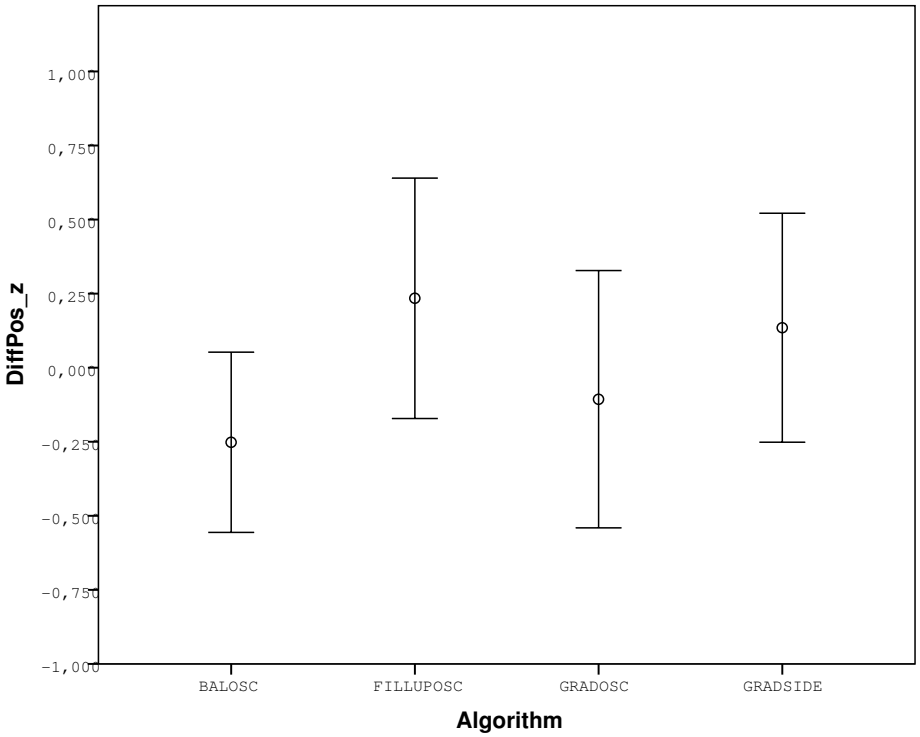


Figure 5.6: Error bar plot for the *differences in positions* z-rating.

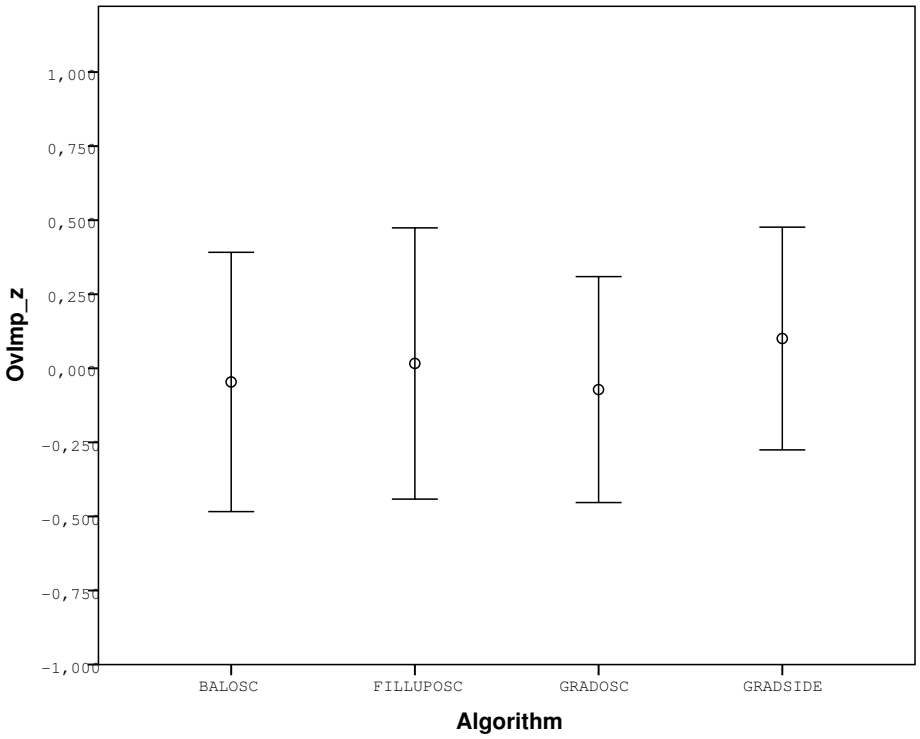


Figure 5.7: Error bar plot for the *overall impression* z-rating.

4. Repeated-measures ANOVA of z-values

The same tests are applied for the z-values calculated before. Again all samples are spherical, but none significant. The rANOVA test gives values between 0,243 and 0,985, proving that the transformation to z-values does not enhance differences in the data-set.

5. Error bar plots for personal preference

As the previous procedures did not reveal any valid differences, the next step is to divide the sample in groups. We thought the algorithm preference, if any, could be based on individual preferences. If that is the case we should make groups that assemble people with the same inclinations and then see whether significant differences are to be found within those groups. Four groups are formed based on the preferred algorithm established by the highest rating for the *overall impression* question. Looking at the other final questions we notice that in general the favored rendering technique of the corresponding group receives the highest mean rating. This indicates that the subjects were consistent in their evaluations throughout the final questionnaire. However, the confidence intervals are still highly overlapping, which hints towards insignificance in the following statistical tests. Figures **??** to **??** show the error bar plots for each group of the *new positions* question 3.2.

6. rANOVA for personal preference

8 questions for 4 groups give rise to 32 data sets to be analysed. Two of them did not answer to the sphericity condition according to Mauchly's test, to which the Greenhouse-Geisser correction was applied. We found 4 times significant differences for the *overall impression* question. This is not surprising however, as we formed the groups based on that question. So these differences can not be taken into account, because the group splitting is an adaptation that statistically forces significance to a certain extent. Besides that, all groups have one more statistical significant (¡0.05) result for varying questions ($0.021 \leq p \leq 0.043$). In conclusion, making groups for personal preference showed of more trends in the results. However, we believe these are not significant enough to really explain the indecisiveness in the first four steps.

7. Error bar plots of intermediate questions I

Now we look at the intermediate ratings. In this step the consecutive questions are plotted for each algorithm separately. Two questions (see Appendix F) for four rendering techniques amount to eight plots. In all of them a clear trend of decreased rating can be noticed (e.g. Figure 5.12). This speaks for itself, because the more conferees are present in the acoustical view, the harder it is to identify and accept the allocated positions.

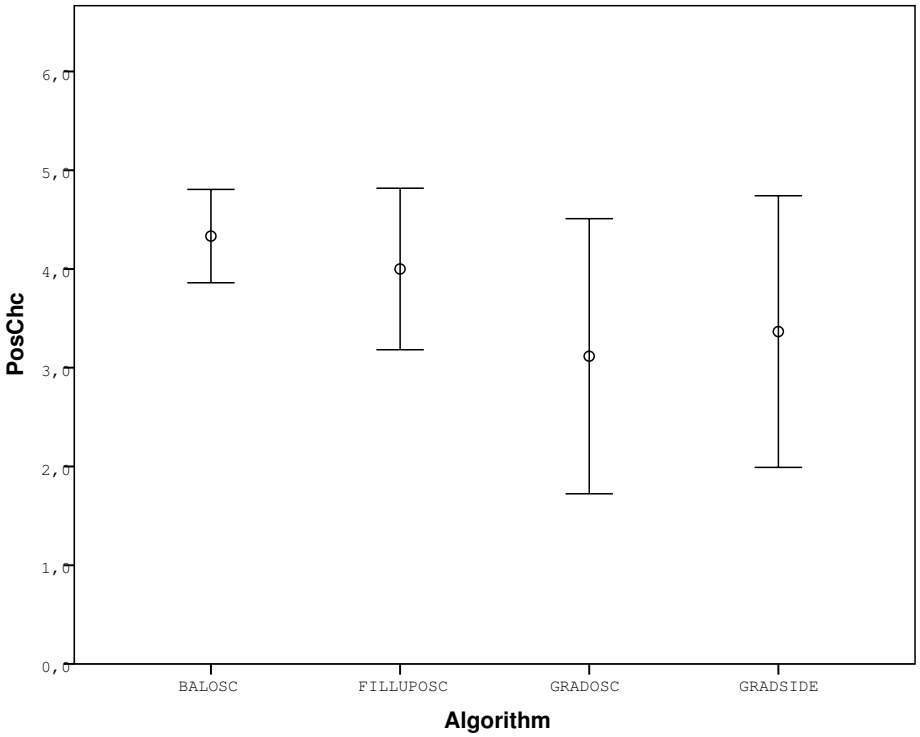


Figure 5.8: Error bar plot for the *new positions* rating of the group preferring 'BALOSC'.

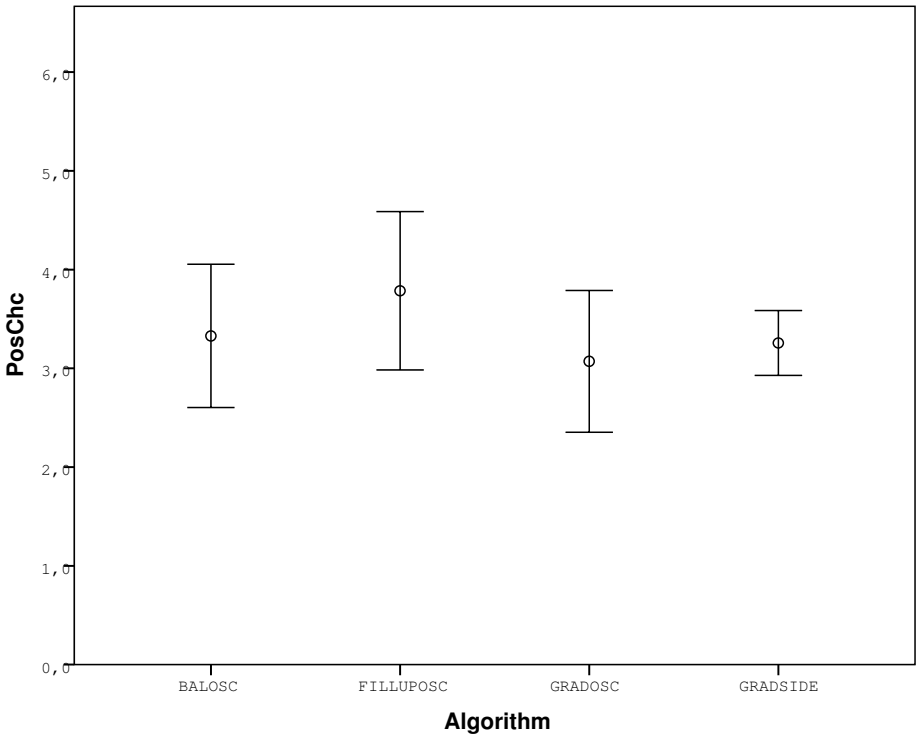


Figure 5.9: Error bar plot for the *new positions* rating of the group preferring 'FILLUPOSC'.

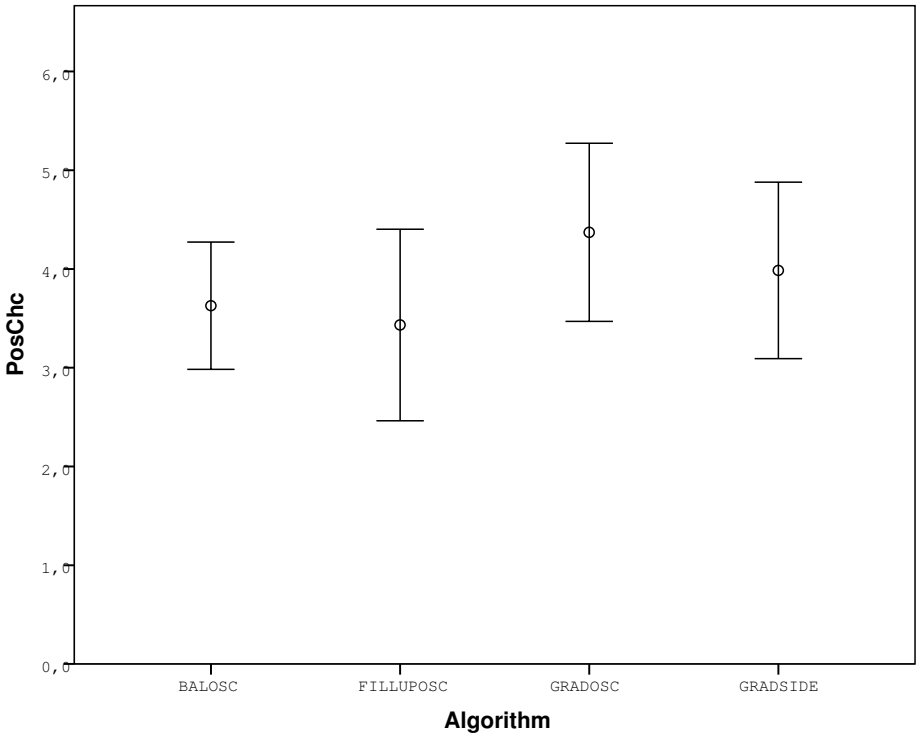


Figure 5.10: Error bar plot for the *new positions* rating of the group preferring 'GRADOSC'.

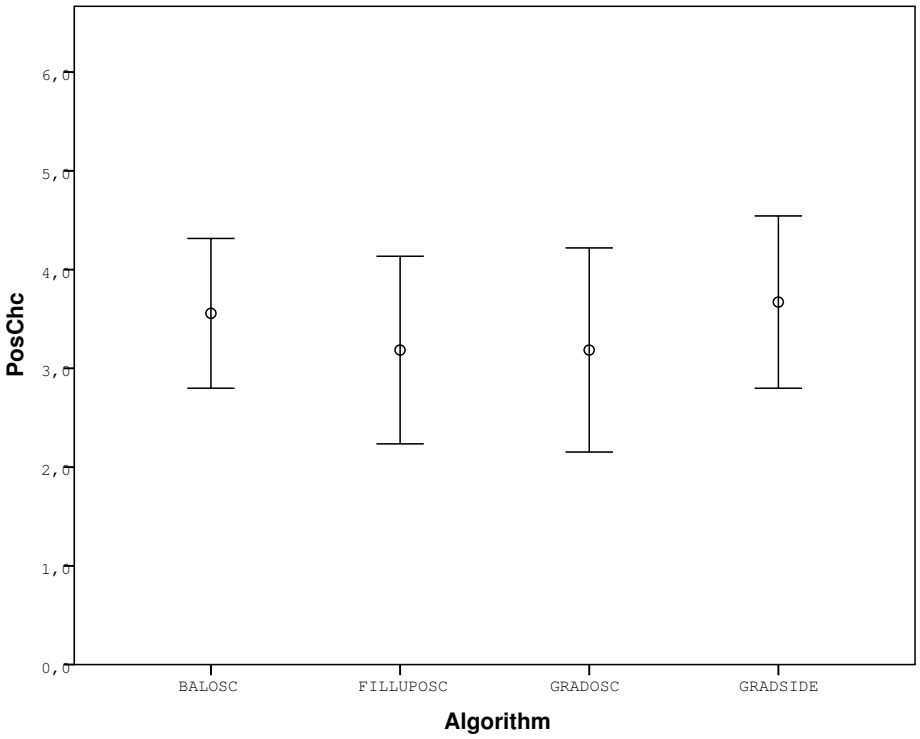


Figure 5.11: Error bar plot for the *new positions* rating of the group preferring 'GRADSIDE'.

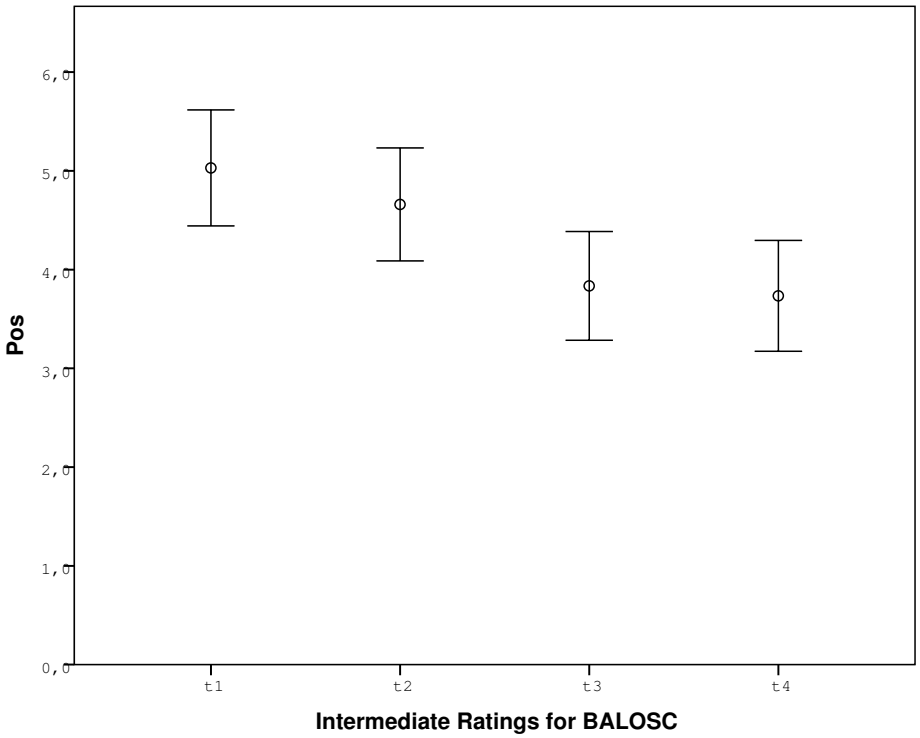


Figure 5.12: Error bar plot of the four consecutive *new positions* ratings of the intermediate questionnaire for 'BALOSC'.

8. Error bar plots of intermediate questions II

It is more interesting to look at each question separately and compare the different rendering techniques. In this case we tend to look more at the third and fourth question as the algorithm has not had the chance to shown many dynamical traits yet when only three or less participants are present. Nonetheless, as for the final questionnaires, not much variation is to be found. An example is shown in Figure 5.13.

9. rANOVA for intermediate questions I

The statistical tests corresponding with step nr. 7 reveal statistical differences for all eight sets ($0.000 \leq p \leq 0.001$). Looking into the post-hoc tests we see that the valid disparities are to be found between the outlying questions, which seems evident in correspondence with the downward trend. These findings are not useful though, as they do not reveal any divergence in algorithm perception. They do provide confirmation that the subjects were rather consistent in the rating process, instead of giving arbitrary responses.

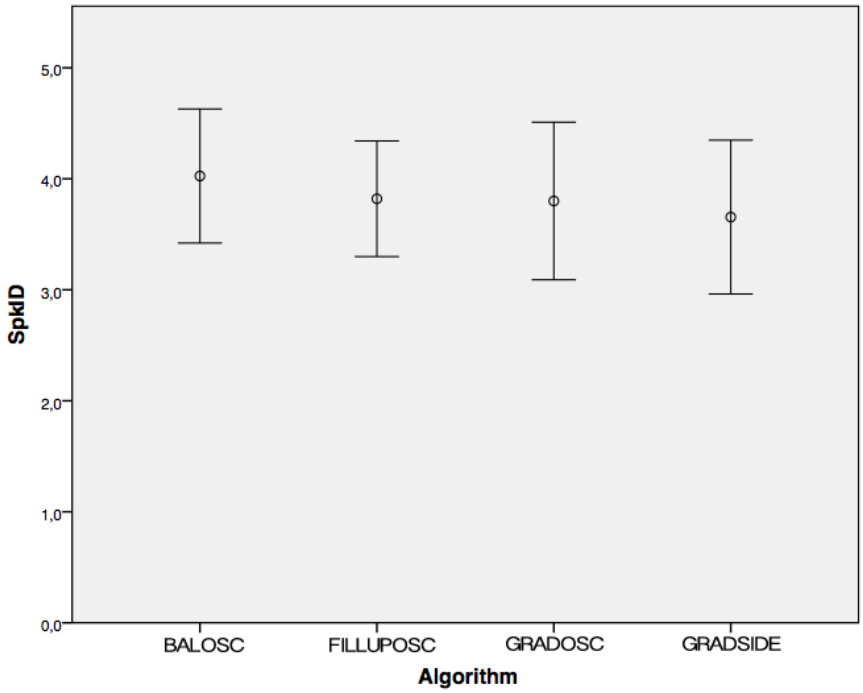


Figure 5.13: Error bar plot of the four algorithms for the fourth *speaker identification* intermediate question.

10. rANOVA for intermediate questions II

Performing the statistical tests from nr. 8, no significance is encountered, as expected. Focusing especially on the results of the third and fourth version of the intermediate questionnaire, we find $p \geq 0.500$, which denotes that the post-hoc results are far from usable.

11. Pearson correlation between final & intermediate ratings

Finally, in an attempt to assess whether it was worth having the intermediate ratings or not, we perform a Pearson correlation test[7] between the intermediate and final ratings. The results are given in Table 5.3. We clearly see that the higher the repetition order of the intermediate question the more it correlates to it's equivalent in the final questionnaire. This most likely has to do with the fact that the subject will pass it's final and most recent opinion on from the intermediate to the final ratings. Although they are not close to one, there is definitely some mutual connection, making the fourth intermediate question possibly redundant. The first and second intermediate questions are much less correlated, but also less worthy, as mentioned before, because the rendering technique has only been exposed to the subject to a very small extent. Taking the fact into account that no valid conclusions could be made about differences

[7]The Pearson correlation coefficient is the covariance between the two variables divided by the product of their standard errors.

in the algorithms, we lean towards the thought that the intermediate questions are redundant. This discussion will be readdressed in the next chapter.

5.4 Summary

After giving a brief introduction on the setting of these experiments in the research environment, we thoroughly explained which rendering techniques would be put to the test and why. In the second section we plunged into detail about how to expose these algorithms to the test subjects through a distinct set of stimuli, and the set-up of the entire operation. Consequently, the results were parsed through a block of statistical analysis in order to learn as much as possible - in the perspective of dynamical aspects - about the four proposed algorithms.

Pos		SpkID	
Pos1	0,323	SpkID1	0,348
Pos2	0,508	SpkID2	0,527
Pos3	0,667	SpkID3	0,820
Pos4	0,792	SpkID4	0,848

Table 5.3: Results of Pearson test between intermediate and final questions. The first row denotes the final questions, the others the four consecutive intermediate ones. *Pos* refers to question 1.2 and *SpkID* to 2.1 in Appendix F

Chapter 6

Conclusion & Future Work

6.1 Conclusion

Our initial prospects concerning the subjective tests were that one or several variants of the tested algorithms would prove to be superior to others in terms of QoE. As we got closer to the experiment, we realized throughout the development that it would be more interesting to investigate two aspects as research dimensions : the type of movement and the insertion order. After the analysis of the results, we came to the statistical conclusion that no version performed significantly better then others. This can bear two meanings :

1) All rendering techniques are equally good, as the ratings averaged positive on the rating scale. The subjects were content with all 4 presented methods that fixate the dynamical aspects of spatial audio.

2) The subjective experiment was not devised well enough to really grasp the perceptual differences in test conditions. Maybe some disturbance or distraction, we don't know about, deviated the participants' attention from the task at hand. Or the stimuli and questionnaires were not effective enough to extract the actual QoE. These complications could be the cause for the high overlap of variances in the ratings.

This brings us to a twofold conclusion when it comes to the tested algorithms, that is described as follows. The second statement only holds under the assumption that there is a perceivable difference in QoE. In theory, we may state that there definitely is one, as humans are subconsciously influenced by every alteration in their environment. However it might be so tiny, that there is no use in finding it, as users will not really experience or notice the difference. To successfully deliver valid 'winners', the precision of the subjective tests probably needs to be much closer to perfection. So in a theoretical way point 2 holds true. However, it might just be so impractical and fruitless, that delving into it by improving the experiment is

useless. Taking into account that we strongly believe the experiment was rightly executed to a reasonable extent, we aim more towards point 1.

Spatial audio in teleconferences is a novel technique. Researching the dynamical aspects is pioneering work. As we handled inexperienced subjects with only slight training, these new technological methodologies might not appear into their considerations at this point in time. Maybe more attention will be pointed towards this in a later state, when consumer products reach the market. In that light, we pass the general message to potential product developers to use whichever of the tested rendering techniques. Currently we think it is more sensible to base the selection on implementation benefits, rather than QoE. Our recommendations are to use the balancing option for the *rendering algorithm*, which will require the least computing effort if implemented efficiently.

Finally, we summarize the three things this work potentially contributes to the community. In the first chapter a theoretical framework was presented that unambiguously categorizes and defines multiple algorithms, that control the dynamical aspects of spatial audio teleconferences. Additionally, an architecture was devised that can be used to create conversational test scenarios for conversational tests or the recording of material for listening tests. It has been objectively evaluated to provide quite consistent products. Thirdly, subjective tests provided us with insights on the QoE perception of the rendering techniques.

6.2 Future Work

To end this thesis we give some opinions and recommendations about future work. First, if we were to do the tests again, we would change the following, based on our experience :

- Use spatial audio experts or at least heavily trained subjects. We predict that they might be more capable of assessing diversity.

- Omit the *intermediate questionnaire*, as the answers were highly correlated with the final one. The disadvantage is that it pulls the subject out of the experience and probably defocusses his or her overall assessment.

- Pose questions of a more technical and less intuitive nature. This implies extracting β_0 from Figure 5.1, instead of b_0. In practice this corresponds to asking about perceptual events, such as *"Did you hear a talker move?"*, rather then quality impressions, like *"Were the reallocations good?"*.

- Quite some subjects reported in informal discussion that they noticed the different *sequencing algorithms*, but not the way talkers were shifted. Considering this, it might be

worthy only looking into the first dimension, because if the second dimension does not matter, it's variance is just another factor that is contaminating the results.

In the case of concatenating research one might look into the effect of a **??**. We are of the opinion that this is only worth investigating with head tracking and visual representation. Especially with that combination, as colleagues reported to have some reference coordinate system confusion in the past when only using head tracking.

Lastly, when seating themselves around a big table, people have the reflex to sit close to those with whom they have most in common. E.g. when a meeting about a juridical matter is organized, usually the two parties sit at opposing ends. It might be interesting to give this possibility in teleconferences. Or a participant rearranges the positions manually according to the contextual role of the talkers, or the seats are allocated around the *virtual meeting table* based on the context. Either way, this concept superposes our work and can be researched independently. We only mention it, as it was reported by a few test subjects as appealing.

Appendix A

Nomenclature

acoustical view the perceived virtual acoustical space, fed by the effects of spatial audio, together with the rendered elements in it

BRIR binaural room impulse response : HRIR with the effect of room reflections

BRTF binaural room transfer function : frequency transformation of BRIR

diotic stereo playback, where the left and right signal do not differ

global ordering/sequence when applying the *virtual meeting table*, the unique ordering/sequence distributed to all components

conversational test scenarios a unique set of rules and descriptions, that provides the instructions to simulate conferences in a controlled fashion

head tracking the technique where head movement is measured by use of a sensor and incorporated in the real-time rendering process. perceptually this changes the spatial coordinate axis from originating inside the head to in the real physical environment.

HRIR head-related impulse response : the direction dependent response pair, measured with a mannequin head, used for the rendition of spatial audio

HRTF head-related transfer function : the spectral transformation of HRIR

individual ordering/sequence the incoherent orderings/sequences that are individually optimized for each listener separately

JND just noticeable difference, see MAA

listener the role of a conferee related to his or her listening perspective

LoT listening-only-test, see Chapter 5

MAA minimal audible angle, this is the minimal angular shift that needs to take place for a subject to notice a spatial displacement. this is subject- and location-related.

order tag a tag given to a conferee that specifies the relative point of arrival

ordering the output format of the *sequencing algorithm*, containing the order of participants (=sequence)

QoE Quality of Experience, see 5.1.1

QoS Quality of Service, see 5.1.1

rendering algorithm component of the allocation system that determines the absolute acoustical placement of talkers for each listener, based on the output of the *sequencing algorithm*

sequence the output format of the *sequencing algorithm*, containing the order of participants (=ordering)

sequencing algorithm component of the allocation system that determines the order by which talkers are distributed in the acoustical view

talker referring to a conferee's role in the acoustical view of a listener

virtual meeting table refers to the concept of organising the spatial placement as if the conference would take place around a real circular table

voice actor a person, used for the conference simulations, whose utterances are recorded. he or she is not necessarily a professional voice actor.

Appendix B

Matlab Simulation

```
function [ variance_c, u_c, variance_p, u_p, osc_f, res_f ]
= calcAngularChange( GA )

maxi = 6;

%% CREATION OF INDIVIDUAL VIEWS
% 'indView' : Odd rows represent ID, even columns represent their degrees.
% For each of the different listeners. So listener 1 has row 1 and 2,
% listener 2 has row 3 and 4, etc...

indView = zeros(2*maxi,5,6); % Multidimensional matrix keeping track
% of all individual setups after each arrival.
for t = 2:6 % t represents the t'th arrival
        ind = ceil(t/2); % index
        % Form the ordening at moment t, based on the final sequence.
        GA_t = zeros(1,t);
        x = 1;
        for i = 1:6
            if GA(i) <= t
                GA_t(1,x) = GA(i);
                x = x+1;
            end
        end
        for i = 1:t % construct view for i
            location = find(GA_t(:)==i); % find the location of i in GA
            if mod(t,2)==0 % if the number of part. is even, the number
```

```
% of listeners is odd and thus we place someone on zero degrees
                mid = mod(location+t/2,t);
                mid(mid==0) = t; % value zero corresponds to t
            end
            for l = 1:(ind-1) % find vector locations for i's neighbours
                min(ind-l) = mod(location-l,t);
                plus(ind-l) = mod(location+l,t);
                min(min==0) = t;
                plus(plus==0) = t;
            end
            if mod(t,2)==0 % even number of part. : one central speaker
% and an equal amount to it's left and right
                indView(2*i-1,3,t) = GA_t(mid);
                indView(2*i,3,t) = 0;
                for l = 1:(ind-1)
                    indView(2*i-1,3-l,t) = GA_t(min(l));
                    indView(2*i,3-l,t) = -90+(ind-l)*180/t;
                    indView(2*i-1,3+l,t) = GA_t(plus(l));
                    indView(2*i,3+l,t) = 90-(ind-l)*180/t;
                end
            else  % odd number of part. : no central speaker, but equal
% amount on the sides
                for l = 1:(ind-1)
                    indView(2*i-1,4-l,t) = GA_t(min(l));
                    indView(2*i,4-l,t) = -90+(ind-l)*180/t;
                    indView(2*i-1,3+l,t) = GA_t(plus(l));
                    indView(2*i,3+l,t) = 90-(ind-l)*180/t;
                end
            end
        end
end

%% COMPUTATION OF ANGULAR CHANGES

% per step
variance_c = zeros(1,6);
u_c = zeros(1,6);
for t = 3:6
```

```
    clearvars chg;
    ind = 1;
    for i = 1:(t-1)
        for j = 1:(t-1)
            if i~=j
                p1 = find(indView(2*i-1,:,t-1)==j);
                p2 = find(indView(2*i-1,:,t)==j);
                chg(ind) = abs(indView(2*i,p1,t-1)-indView(2*i,p2,t));
                ind = ind+1;
            end
        end
    end
    variance_c(t) = var(chg);
    u_c(t) = mean(chg);
end

% per participant

variance_p = zeros(1,5);
u_p = zeros(1,5);
for t = 1:5 % listener
    clearvars chg;
    ind = 1;
    for i = max([t+1,3]):6 % cycle
        for j = 1:(i-1) % speaker
            if t~=j
                p1 = find(indView(t*2-1,:,i-1)==j);
                p2 = find(indView(t*2-1,:,i)==j);
                chg(ind) = abs(indView(t*2,p1,i-1)-indView(t*2,p2,i));
                ind = ind+1;
            end
        end
    end
    variance_p(t) = var(chg);
    u_p(t) = mean(chg);
end

%% COMPUTATION OF DIRECTION OF ANGULAR SHIFTS
```

```
dir = zeros(5,5,2); % first matrix clockwise, second matrix anti-clockwise
for t = 1:5 % listener
    for i = max([t+1,3]):6 % cycle
        for j = 1:(i-1) % speaker
            if t~=j
                p1 = find(indView(t*2-1,:,i-1)==j);
                p2 = find(indView(t*2-1,:,i)==j);
                x = indView(t*2,p1,i-1)-indView(t*2,p2,i);
                if x/abs(x) > 0 % negative clockwise, positive anti-clockwise
                    dir(t,j,2) = dir(t,j,2) + 1;
                elseif x/abs(x) < 0
                    dir(t,j,1) = dir(t,j,1) + 1;
                end
            end
        end
    end
end

osc = zeros(5,5);
% calculate precentages of oscillation
for i = 1:5
    for j = 1:5
        if i~=j
            if (dir(i,j,2)~=0 && dir(i,j,1) < dir(i,j,2))
                osc(i,j) = dir(i,j,1)/dir(i,j,2);
            elseif (dir(i,j,1)~=0 && dir(i,j,2) < dir(i,j,1))
                osc(i,j) = dir(i,j,2)/dir(i,j,1);
            end
        end
    end
end

% calculate means
osc_m = zeros(1,5);
osc_m(1) = (4*osc(1,2)+3*osc(1,3)+2*osc(1,4)+1*osc(1,5))/10;
osc_m(2) = (4*osc(2,1)+3*osc(2,3)+2*osc(2,4)+1*osc(2,5))/10;
osc_m(3) = (3*osc(3,1)+3*osc(3,2)+2*osc(3,4)+1*osc(3,5))/9;
```

```
osc_m(4) = (2*osc(4,1)+2*osc(4,2)+2*osc(4,3)+1*osc(4,5))/7;
osc_m(5) = (1*osc(5,1)+1*osc(5,2)+1*osc(5,3)+1*osc(5,4))/4;
osc_f = (osc_m(1)*10+osc_m(2)*10+osc_m(3)*9+osc_m(4)*7+osc_m(5)*4)/40;

%% COMPUTATION OF RESIDUAL ANGULAR CHANGE

res = zeros(5,5); % first matrix clockwise, second matrix anti-clockwise
for t = 1:5 % listener
        for j = 1:5 % speaker
            if t~=j
                i = max([t,j,2]);
                p1 = find(indView(t*2-1,:,i)==j);
                p2 = find(indView(t*2-1,:,6)==j);
                res(t,j) = indView(t*2,p2,6)-indView(t*2,p1,i);
            end
        end
end

% calculate means (now we do take the absolute value!)
res = abs(res);
res_m = zeros(1,5);
res_m(1) = (4*res(1,2)+3*res(1,3)+2*res(1,4)+1*res(1,5))/10;
res_m(2) = (4*res(2,1)+3*res(2,3)+2*res(2,4)+1*res(2,5))/10;
res_m(3) = (3*res(3,1)+3*res(3,2)+2*res(3,4)+1*res(3,5))/9;
res_m(4) = (2*res(4,1)+2*res(4,2)+2*res(4,3)+1*res(4,5))/7;
res_m(5) = (1*res(5,1)+1*res(5,2)+1*res(5,3)+1*res(5,4))/4;
res_f = (res_m(1)*10+res_m(2)*10+res_m(3)*9+res_m(4)*7+res_m(5)*4)/40;

end
```

Appendix C

C++ Code

"globTeleConference.h"

```
#ifndef TELECONFERENCE_H_
#define TELECONFERENCE_H_
#include "seqAlg.h"
#include "globRendAlg.h"

class GlobTeleConference {
private:
    int seqalg; // MinOsc(645321) -> 1; MinResAng(642531) -> 2;
    int N;
    int going;
    int AT;
    SeqAlg A;
    GlobRendAlg B;

public:
    GlobTeleConference();
    GlobTeleConference(int SA, int audTrans);
    int GetSA() { return seqalg; }
    int GetN() { return N; }
    int ATFlag() { return AT; }
    int addPart(int ID);
    int remPart(int ID);
};

#endif /* TELECONFERENCE_H_ */
```

"globRendAlg.h"

```
#ifndef RENDALG_H_
#define RENDALG_H_
#include "listener.h"
#include "seqAlg.h"
#include "talker.h"

class GlobRendAlg // here we implement only the balanced rendering method
{
private:
    int N;
    Listener* head;
    int AT;
public:
    GlobRendAlg();
    GlobRendAlg(int AT);
    ~GlobRendAlg();
    int GetN() { return N; }
    int addConferee(int ID, SeqAlg * A);
    int removeConferee(int ID);
    int audTrans(int a, int aT);
};

#endif /* RENDALG_H_ */
```

"indTeleConference.h"

```
#ifndef INDTELECONFERENCE_H_
#define INDTELECONFERENCE_H_
#include "indRendAlg.h"

class IndTeleConference {
private:
    int seqalg; // Sideways -> 1; Oscillating -> 2; Waterfall -> 3
    int N;
    int going;
```

```
    int AT;
    IndRendAlg B;

public:
    IndTeleConference();
    IndTeleConference(int SA, int audTrans);
    int GetSA() { return seqalg; }
    int GetN() { return N; }
    int ATFlag() { return AT; }
    int addPart(int ID);
    int remPart(int ID);
};

#endif /* INDTELECONFERENCE_H_ */
```

"indRendAlg.h"

```
#ifndef INDRENDALG_H_
#define INDRENDALG_H_
#include "listener.h"
#include "talker.h"

class IndRendAlg // here we implement only the balanced rendering method
{
private:
    int N;
    Listener* head;
    int mode;
    int AT;
public:
    IndRendAlg();
    IndRendAlg(int seq, int AT);
    ~IndRendAlg();
    int GetN() { return N; }
    int addConferee(int ID);
    int removeConferee(int ID);
    int audTrans(int ID, int AT);
};
```

```
#endif /* INDRENDALG_H_ */
```

"seqAlg.h"

```
#ifndef SEQALG_H_
#define SEQALG_H_

class SeqAlg
{
private:
    int mode; // MinOsc(645321) -> 1; MinResAng(642531) -> 2;
    int N;
    struct Participant
       {
           int ID;
           int order;
           Participant* prev;
           Participant* next;
           Participant(int iden, int ord, Participant* p, Participant* n) :
           ID(iden), order(ord), prev(p), next(n) {}
       };
    Participant* head;
public:
    SeqAlg();
    SeqAlg(int m);
    ~SeqAlg();
    int GetMode() { return mode; }
    int GetN() { return N; }
    int addPart(int ID);
    int removePart(int ID);
    int *extractOrder(int function);
    int getID(int ord);
};

#endif /* SEQALG_H_ */
```

"listener.h"

```
#ifndef LISTENER_H_
#define LISTENER_H_
#include "talker.h"

class Listener {
public:
        int ID;
    Listener* next;
    Listener* prev;
    Talker* start;
    Listener();
    Listener(int iden, Listener* n, Listener* p);
    ~Listener();
    int addTalker(int ID, int ang);
    int removeTalker(int ID);
};

#endif /* LISTENER_H_ */
```

"talker.h"

```
#ifndef TALKER_H_
#define TALKER_H_

struct Talker {
        int ID;
        int angle;
        Talker* next;
        Talker* prev;
        Talker(int iden, int ang, Talker* n, Talker* p) :
        ID(iden), angle(ang), next(n), prev(p) {}
};

#endif /* TALKER_H_ */
```

Appendix D

Conversational Scenario Layers

The schematic representations of the three layers for the development of the conversational test scenarios are shown in this appendix. As an example the *Product Presentation* conference is used. Beginning with the log layer, the structure of logs can be viewed in Figure **??**. This holds for all scenarios.

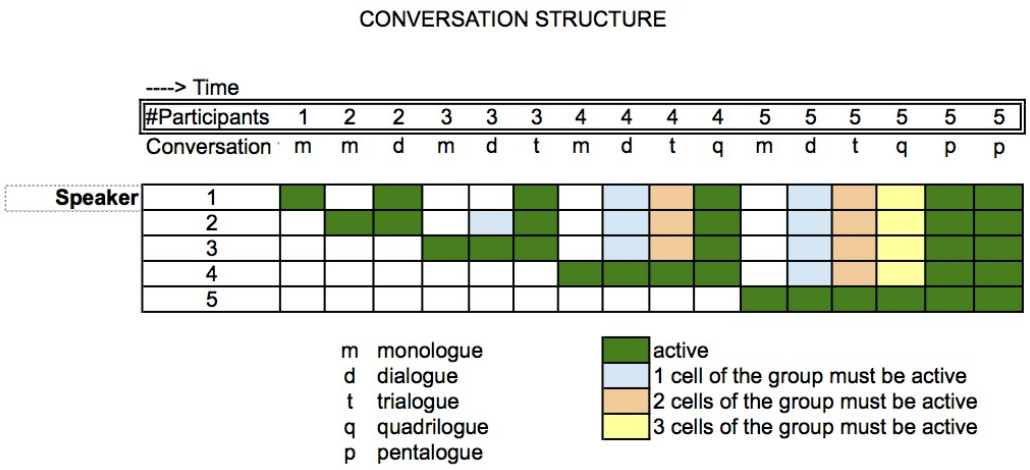


Figure D.1: Graphical representation of the log layer.

In the design of the first conference the function layer was created in function of the content. That function structure was then copied for the other scenarios and slightly adapted here and there to provide sensible content. Figure D.2 shows the function structure for the case at hand (coloured boxes) together with some preliminary content (conferee roles and keywords describing the speech bursts). At this point the context of the conference was already worked out (see D.3).

Finally, the final scenario forms are created (Figure D.3 to D.7), that contain all the

content. These are furthermore the hand outs administered to the voice actors for the recorded simulations of the scenarios. Furthermore, we highlighted the topics for each participant separately. In order to inform the actors well, the instruction document in Figure D.8 was also distributed.

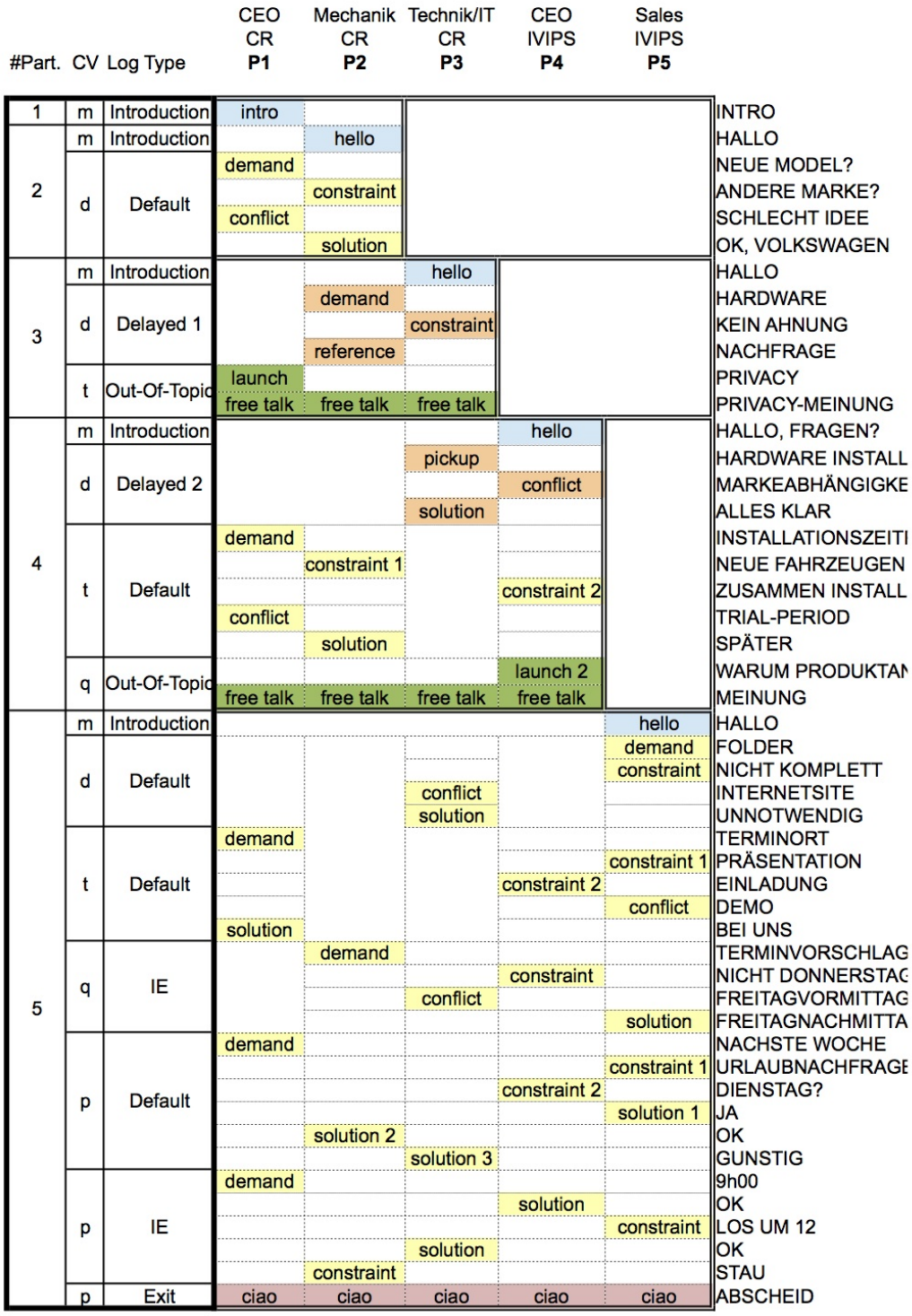


Figure D.2: Graphical representation of the function layer for the *Product Presentation* scenario with preliminary content description.

Szenario 1 – IVIPS

Information zu den Rollen

Teilnehmer

Gesprächspartner 1	Name Abteilung	Herr Keller (P1) CEO Carco
Gesprächspartner 2	Name Abteilung	Herr Huber (P2) Fleet Manager/Mechaniker Carco
Gesprächspartner 3	Name Abteilung	Herr Zimmel (P3) ERP/IT Manager Carco
Gesprächspartner 4	Name Abteilung	Herr Klühn (P4) CEO IVIPS
Gesprächspartner 5	Name Abteilung	Herr Röntsch (P5) Sales Manager IVIPS

Ihre Rolle

Information zum Thema der Konferenz

Grund des Anrufs

Die Firma Carco, ein Autoverleihbetrieb, ist interessiert, eine neue ERP-Technologie von IVIPS zu kaufen. Es geht um eine Hardwarekomponente, die in jedes Fahrzeug installiert wird und verschiedene technische Daten des Autos in die Cloud schickt. Zusätzlich bekommt man einen Cloud-Service, der das ganze gebrauchsfreundlich macht.
Drei Personen von Carco telefonieren mit dem CEO und dem Sales Manager von IVIPS, um allgemeine Fragen zu beantworten und einen umfangreichen Präsentationstermin zu verabreden.

Gesprächsleitfaden

⇒ nächste Seite

Figure D.3: Hand out of the content for *Product Presentation* - page 1.

Szenario 1 – IVIPS

1 Gesprächspartner (+ Herr Keller)

Einleitung	**Erzählen** Sie kurz - an einen imaginären Zuhörer - den **Grund** und das **Ziel** des **Gespräches**.	**P1**	• Carco, Autoverleihfirma • Kauf einer neuen ERP-Technologie, von IVIPS • Termin machen
Vorstellung	**Stellen Sie sich kurz vor** und erklären Sie Ihre Funktion bei dieser Konferenz.	**P1**	• Herr Keller, CEO Carco

2 Gesprächspartner (+ Herr Huber)

Vorstellung	**Stellen** Sie sich kurz **vor** und **fragen** Sie **wer** schon **da ist**.	**P2**	• Herr Huber, Fleet Manager • Organisiert die gesamte Flotte und kontrolliert die Mechanik
Neufahrzeuge	Carco muss **neue Autos kaufen**, sie **überlegen welche Modelle/Marke** sie nutzen sollten.	**P1**	• Dieselben Modelle wie bisher von Neufahrzeugen kaufen?
		P2	• Volvo hat ein gutes Modell für uns • Hab schon mit Vertretern auf einer Messe gesprochen
		P1	• Lass uns bei Volkswagen bleiben • Guter Service und es ist aufwendig mehrere Lieferanten zu haben
		P2	• Ok, ich rede mit dem VW Vertreter
		P1	• Prima, Sie sagen dann Bescheid über deren Empfehlungen

3 Gesprächspartner (+ Herr Zimmel)

Vorstellung	**Stellen** Sie sich kurz **vor** und **fragen** Sie **wer** schon **da ist**.	**P3**	• Herr Zimmel, IT Manager Carco • Unterhält u.a. die ERP-Systeme
Installierung	Man bespricht kurz ob es schwierig ist, die **Hardwarekomponente** zu **installieren**.	**P2**	• Kompliziert das Hardwareteil zu installieren?
		P3	• Hab schon das Prospekt durchgeschaut, steht leider nicht drin
		P2	• Ok, wir fragen später die Kollegen von IVIPS
Off-Topic	Sie reden über eventuelle **Privacy-Probleme** mit diesen **GPS-Modulen** des Produktes. Dann weis man immer wo der Kunde ist.	**P1**	• Gute Idee, denn ich weiß es auch nicht • Ansonsten, vielleicht Privacy-Probleme, mit den GPS-Modulen
		P3	• Ja, aber diese Daten sind verschlüsselt • Dritte haben kein Zugang dazu
		P2	• Nee, aber es geht um das Prinzip
		P1	• Wir sehen ob wir das überhaupt nutzen • Wenn ja, muss man dem Kunden Bescheid sagen

Seite 2

Figure D.4: Hand out of the content for *Product Presentation* - page 2.

Szenario 1 – IVIPS

4 Gesprächspartner (+ Herr Klühn)

Vorstellung	**Stellen Sie** sich kurz **vor**.	**P4**	• Herr Klühn, CEO IVIPS
Installierung	Jetzt befragt Herr Zimmel Herr Klühn über die **Installation** der **Hardwarekomponente**	**P3**	• Herr Klühn, kompliziert diese Hardwarekomponente in Fahrzeug zu installieren?
		P4	• Grundsätzlich nicht, aber auch abhängig von der Automarke • VW, Audi, Seat,… geht einfach Volvo z.B. ist bisschen schwieriger
		P3	• Gut, arbeiten zur Zeit nur mit VW
Installations-frist	Sie besprechen den **frühsten Installationstermin**. Vielleicht probiert Carco einen Versuchslauf.	**P1**	• Wann am frühsten installieren?
		P2	• Genau, weil wir kaufen neue Wagen
		P4	• Am besten alle Autos in einem Durchgang • Am frühsten, ungefähr Anfang nächsten Jahres
		P1	• Carco denkt an einen Versuchslauf mit den Geräten
		P2	• Das können wir später überlegen und danach einen Installationszeitraum verabreden
Off-Topic	Herr Klühn fragt, **warum Carco** genau **IVIPS gewählt** hat und welche ERP-Software sie jetzt nutzen.	**P4**	• Was für ERP-Software jetzt benutzt? • Warum für IVIPS entschieden?
		P1	• Jetzt ein einfache Software, heißt CarRent
		P2	• Zu wenig Kontrollmöglichkeiten und Daten
		P3	• Suchen etwas, das den Verleihprozess vereinfacht • Eco-Modus ist ganz interessant
		P4	• Ja, die meisten Kunden bestätigen das

Seite 3

Figure D.5: Hand out of the content for *Product Presentation* - page 3.

Szenario 1 – IVIPS

5 Gesprächspartner (+ Herr Röntsch)

Vorstellung	**Stellen Sie** sich kurz **vor.**	**P5**	• Herr Röntsch, Sales Manager IVIPS

Prospekt	Herr Röntsch fragt, ob Herr Zimmel **das Prospekt** bekommen hat.	**P5**	• Herr Zimmel, haben Sie das Prospekt bekommen?
		P3	• Nein, noch nicht • Aber hab es schon auf der Website durchgelesen
		P5	• Leider ist es nicht komplett up-to-date
		P3	• Es ist ok, ich verstehe das Prinzip

Tagungsort	Sie besprechen **den Termin**, an dem eine **ausführliche Vorstellung des Produktes** stattfinden soll.	**P1**	• Zu dem geplanten Meeting, wo machen wir das?
		P5	• Wo meine Präsentation stattfindet ist für mich egal
		P1	• Dann sind sie herzlich willkommen bei uns • Unsere Firmenzentralen sind nicht weit weg von einander?
		P4	• Ja stimmt • Gut, wenn wir die Halle und Fahrzeugflotte besichtigen können, dann können wir direkt praktische Vorschläge unterbreiten
		P1	• Ok, dann wird es bei uns stattfinden

Dauer	Man bespricht, wie lange das **Treffen** dauern sollte.	**P2**	• Wie lange das Treffen?
		P4	• 1 Stunde Präsentation & Fragen zum Produkt
		P3	• 1 Stunde Vorstellung und Besichtigung der Fahrzeugflotte
		P5	• 30 Minuten besprechen der Finanziellen Bedingungen
		P2	• Also planen 3 Std. inklusive Empfang und Pause

Seite 4

Figure D.6: Hand out of the content for *Product Presentation* - page 4.

Zeitpunkt	Man schlägt ein Zeitpunkt für das Treffen vor.	**P1**	• Treffen Am liebsten nächste Woche
		P5	• Ok, Montag haben wir eine Vorstandsveranstaltung • Ich fahre Dienstag in Urlaub
		P4	• Herr Röntsch, sie sind nur Dienstagabend weg?
		P5	• Ja • Dienstagmorgen würde gehen
		P2	• Ok für mich
		P3	• Ok, aber meine Frau hat die Woche ihren errechneten Geburtstermin • In diesen Fall, würde ich das halt verpassen

Besichtigungs-termin	**Herr Keller bestätigt** den **Terminzeitpunkt**. **Alle** sind **einverstanden**.	**P1**	• Gut also bei uns am kommenden Dienstag um 9 Uhr. Ok für alle?

Verabschiedung	**P1; P2; P3; P4; P5**

Figure D.7: Hand out of the content for *Product Presentation* - page 5.

Allgemeine Hinweise

Ziel ist es, dass das Gespräch so natürlich wie möglich klingt. Versuchen sie daher, keine besonders betonte, sondern eine natürliche Aussprache zu haben. Versetzen sie sich in die Situation, einen echten Anruf zu tätigen.

Alle Situationen spielen im geschäftlichen Umfeld und die Gesprächspartner kennen sich nicht oder nur flüchtig. Nehmen sie eine offene freundliche Haltung an.

Die folgenden Konferenzen beinhalten kleine „Konflikte", Probleme oder Besprechungen, die aber immer durch Alternativen oder Ideen eines Gesprächspartners gelöst werden können. Zeigen sie eine konstruktive Haltung; vermeiden sie, dass die Konflikte emotional zu sehr aufschaukeln.

Der Gesprächsleitfaden ist in folgender tabellarischer Form

Gesprächs-punkt 1	Kurze Umschreibung über den Inhalt des Gesprächspunkts	**P1**	• Man redet dann in dieser Reihenfolge über
		P2	• die hier beschriebenen Stichworte. Versuchen
		P3	• Sie, die Umschreibung dabei zu beachten.
		P1	• Jeder Gesprächspunkt wird mit einer kleinen Zusammenfassung abgerundet.
Gesprächs-punkt 2	Kurze Umschreibung über den Inhalt des Gesprächspunkts	**P1**	• Man redet dann in dieser Reihenfolge über
		P2	• die hier beschriebenen Stichworte. Versuchen
		P3	• Sie, die Umschreibung dabei zu beachten.
		P1	• Eine leichtblaue Zelle bedeutet, dass man eine kurze Atempause braucht, nach dem Gesprächspunkt

Figure D.8: The instruction document for the voice actors.

Appendix E

Signal Rendition

To render a combination of an algorithm with a scenario, the function *"renderScenario.m"* should be used. The current folder should contain *"hrir_fabian.wav"* containing each HRIR as a channel and all the recordings in conformance with *"S5_Cx_Py.wav"* with x,y, being a number between 1 and 5, denoting, the conference and participant respectively. The function firstly calculates the balanced views and then the stereo signal is created of each participant consecutively. To not overload the memory they are written in temporary wav-files. Once all rendered, the five signals are added together. In the main for-loop, the progress of acoustical angles is calculated for the entire conference and then parsed with the signal to *"renderSignal.wav"*. There, the convolutions are conducted using a window that counts 2025 samples ($f_s = 44.100kHz$), which is double the length of the HRIR's, that have been shortened to a sufficient 512 samples.

"renderScenario.m"

```
function [ rend_sign ] = renderScenario(s,c,m,GA,AT)
%
% INPUT
%
% 's' : this value determines which scenario should be used (1:5)
% 'c' : vector containing the timing cues of the scenario,
%       5 steps in total so, containing 5 elements (in seconds)
% 'm' : specifies which mode should be used to complete the transitions
%       1) FILL-UP
%       2) BALANCING
%       3) GRADUAL
% 'GA' : contains the ordering of the virtual meeting table, vector of
%        elements 1 to 6
```

```
% 'AT' : flag indicating whether the auditory transformation should be
%        applied or not
%
% OUTPUT
%
% 'rend_signal' : the rendered stereo signal containing
% the entire conference
%

% add voicebox functions
addpath('%path of voicebox%');

if s ~= intersect(s,[1 2 3 4 5]);
    fprintf('s must be between 1 and 5!');
    return
end

indView = calcBalancedViews(GA);
l_hrtf = 512;
ws = 2*l_hrtf+1; % window size in samples

% load HRTFs and restructure them
[HRTF,fsh] = wavread('hrirs_fabian');
HRTF = HRTF';
HRTFs = zeros(360,l_hrtf,2);
for i = 1:720
    HRTFs(ceil(i/2),:,mod(i-1,2)+1) = HRTF(i,:);
end

for i = 1:5 % (i+1)th participant/arrival according to GA
    ss = strcat('S5_C',int2str(s),'_P',int2str(i),'.wav');
    [P,fs] = wavread(ss); % fs must be equal for each track
    cc = c*fs; % convert the cues from seconds to samples
    l = length(P);
    hws =(ws-1)/2; % half of ws
    nf = floor((l-1)/hws)-1;

    % check sampling frequencies
```

```
    if fsh ~= fs
        fprintf('The sampling frequencies are not equal!');
        return
    end
    if m == 1 % fill-up
        index = find(indView(11,:)==(i+1));
        angle = indView(12,index);
        if AT == 1
            angle = audTrans(angle);
        end
        angle = angConv(angle);
        ac = angle*ones(nf,1);
        out = renderSignal(P,ac,ws,HRTFs);
    else
        if m == 2 % balanced
            ac = 360*ones(nf,1);
            for j = 1:5 % j'th step
                if find(indView(2*(j+1)-1,:)==(i+1)) % only assign 'ac'
% when the conferee (i+1) already arrived/has
% a position in the individual view
                    index = find(indView(2*(j+1)-1,:)==(i+1));
                    angle = indView(2*(j+1),index);
                    if AT == 1
                        angle = audTrans(angle);
                    end
                    angle = angConv(angle);
                    if j == 1
                        ac(1:floor(cc(1)/hws)-1)=
angle*ones(floor(cc(1)/hws)-1,1);
                    elseif j == 5
                        ac(floor(cc(j-1)/hws):end)=
angle*ones(length(ac(floor(cc(j-1)/hws):end)),1);
                    else
                        ac(floor(cc(j-1)/hws):floor(cc(j)/hws)-1)=
angle*ones(floor(cc(j)/hws)-floor(cc(j-1)/hws),1);
                    end
                end
            end
```

```
    out = renderSignal(P,ac,ws,HRTFs);
elseif m == 3 % gradual
    v = 10; % rotational speed in degrees per second,
                                    % - variable for audTrans!
    x = v*hws/fs; % angular increment per half frame box
    ac = ones(nf,1); % creation of ac
    vad = VAD(P,fs,nf,hws); % denotes if a frame is speech active
    angle1 = indView(2*(i+1),find(indView(2*(i+1)-1,:)==(i+1)));
    ac(1:floor(cc(i)/hws)-1) = angle1*ones(floor(cc(i)/hws)-1,1);
    index = floor(cc(i)/hws);
    for j = max(2,i+1):5 % j'th step
        angle2 = indView(2*(j+1),find(indView(2*(j+1)-1,:)==(i+1)));
        r = floor(cc(j)/hws)-floor(cc(j-1)/hws);
        if angle2 >= angle1
            for t = 1:r
                if vad(index) && angle2 > ac(index-1)
                    ac(index)=ac(index-1)+x;
                else
                    ac(index)=ac(index-1);
                end
                index = index+1;
            end
        elseif angle2 < angle1
            for t = 1:r
                if vad(index) && angle2 < ac(index-1)
                    ac(index)=ac(index-1)-x;
                else
                    ac(index)=ac(index-1);
                end
                index = index+1;
            end
        end
        angle1 = ac(index-1);
    end
    for h = 1:nf
        if AT == 1
            ac(h) = audTrans(ac(h));
        end
```

```
                ac(h) = angConv(ac(h));
                ac(h) = rounding(ac(h));
            end
            out = renderSignal(P,ac,ws,HRTFs);
        end
    end
    out = normalize(out);
    wavwrite(out',fs,strcat('temp/temp',int2str(i)));
    clearvars out P ac
end

P1 = wavread(strcat('temp/temp',int2str(1)));
P2 = wavread(strcat('temp/temp',int2str(2)));
P3 = wavread(strcat('temp/temp',int2str(3)));
P4 = wavread(strcat('temp/temp',int2str(4)));
P5 = wavread(strcat('temp/temp',int2str(5)));
rend_sign = P1+P2+P3+P4+P5;
end

function [ out ] = normalize(in)
% Makes sure no values of in are one or larger, all files must be equally
% normalized : Wittek 6 - Fabian 1.5

out = in/1.5;

end

function [ out ] = angConv(in)
% Converts the angle to the corresponding HRTF index
if in < 0 && in >= -90
    out = -in;
elseif in >= 0 && in < 90
    out = 360-in;
end
end

function [ out ] = audTrans(in)
% Performs the auditory transformation
```

```
if in == 0
    out = 0;
elseif in < 0
    in = -in;
    out = round(-(9.45676*10^(-6)*in^3+0.00475894*in^2+0.495096*in));
elseif in > 0
    out = round(9.45676*10^(-6)*in^3+0.00475894*in^2+0.495096*in);
end
end

function [ out ] = rounding(in)
% Rounds without bumping into zero

out = round(in);
if out == 0
    out = 360;
end
end
```

"calcBalancedViews.m"

```
function [ indView ] = calcBalancedViews( GA )
%
% I N P U T
%
% 'GA' : The global ordering is a vector containing the elements 1 to 6
%
% O U T P U T
%
% 'indView' : Each pair of rows corresponds with a new arrival. The first
% one indicates the identity, the second the actual actual angle. It starts
% with the first conferee, so with two zero vectors and starts presenting
% views from then on.
%

indView = zeros(12,5);
for t = 2:6 % t represents the t'th arrival
    ind = ceil(t/2); % index
```

```
% Form the ordening at moment t, based on the final sequence.
GA_t = zeros(1,t);
x = 1;
for i = 1:6
    if GA(i) <= t
        GA_t(x) = GA(i);
        x = x+1;
    end
end

location = find(GA_t(:)==1); % find the location of i in GA
if mod(t,2)==0 % if the number of part. is even, the number of
    % listeners is odd and thus we place someone on
    % zero degrees
    mid = mod(location+t/2,t);
    mid(mid==0) = t; % value zero corresponds to t
end

for l = 1:(ind-1) % find vector locations for i's neighbours
    min(ind-l) = mod(location-l,t);
    plus(ind-l) = mod(location+l,t);
    min(min==0) = t;
    plus(plus==0) = t;
end

if mod(t,2)==0 % even number of part. : one central speaker and
    % an equal amount to it's left and right
    indView(2*t-1,3) = GA_t(mid);
    indView(2*t,3) = 0;
    for l = 1:(ind-1)
        indView(2*t-1,3-l) = GA_t(min(l));
        indView(2*t,3-l) = -90+(ind-l)*180/t;
        indView(2*t-1,3+l) = GA_t(plus(l));
        indView(2*t,3+l) = 90-(ind-l)*180/t;
    end
else % odd number of part. : no central speaker, but equal
    % amount on the sides
    for l = 1:(ind-1)
```

```
                indView(2*t-1,4-l) = GA_t(min(l));
                indView(2*t,4-l) = -90+(ind-l)*180/t;
                indView(2*t-1,3+l) = GA_t(plus(l));
                indView(2*t,3+l) = 90-(ind-l)*180/t;
            end
        end
end
end
```

"renderSignal.m"

```
function [ out ] = renderSignal(s,ac,wsl,HRTF)
%
% INPUT
%
% 's' : the mono input signal
%
% 'ac' : the vector containing the angle for each frame, which thus
%        determines which HRTFs should be used for each of the frames
%
% 'wsl' : window/frame sample length (has to be odd!)
%
% 'HRTF' : a 2x360xhsl matrix with following structure :
%          (degree one to 360)x(signal)x(L, R channel)
%
% OUTPUT
%
% 'out' : the rendered signal is slightly smaller than the input signal
%         (not more than (wsl-1)/2)
%

ssl = length(s); % signal sample length
hwsl =(wsl-1)/2; % half of wsl
nf = floor((ssl-1)/hwsl)-1; % number of frames in the signal
hsl = size(HRTF,2);

% checking length of 'ac'
if nf ~= size(ac,1)
```

```
    fprintf('Make sure that ac has as
many rows as the number of shifting frames that fit into s \n');
end

% window
w = hann(wsl)';

% creation, windowing and convolution of frames
frames = zeros(nf,wsl);
framesL_c = zeros(nf,wsl+hsl-1);
framesR_c = zeros(nf,wsl+hsl-1);

for i = 1:nf
    i1 = 1+(i-1)*hwsl;
    i2 = wsl+(i-1)*hwsl;
    frames(i,:) = s(i1:i2,1)';
    frames(i,:) = w.*frames(i,:);
    hrtf_L = HRTF(ac(i),:,1);
    hrtf_R = HRTF(ac(i),:,2);
    framesL_c(i,:) = conv(hrtf_L,frames(i,:));
    framesR_c(i,:) = conv(hrtf_R,frames(i,:));
end

% reconstruction of signal
out = zeros(2,wsl+(nf-1)*hwsl+hsl-1);
for i = 1:nf
    i1 = 1+(i-1)*hwsl;
    i2 = wsl+(i-1)*hwsl+hsl-1;
    out(1,i1:i2) = out(1,i1:i2)+framesL_c(i,:);
    out(2,i1:i2) = out(2,i1:i2)+framesR_c(i,:);
end
end
```

Appendix F

Questionnaires

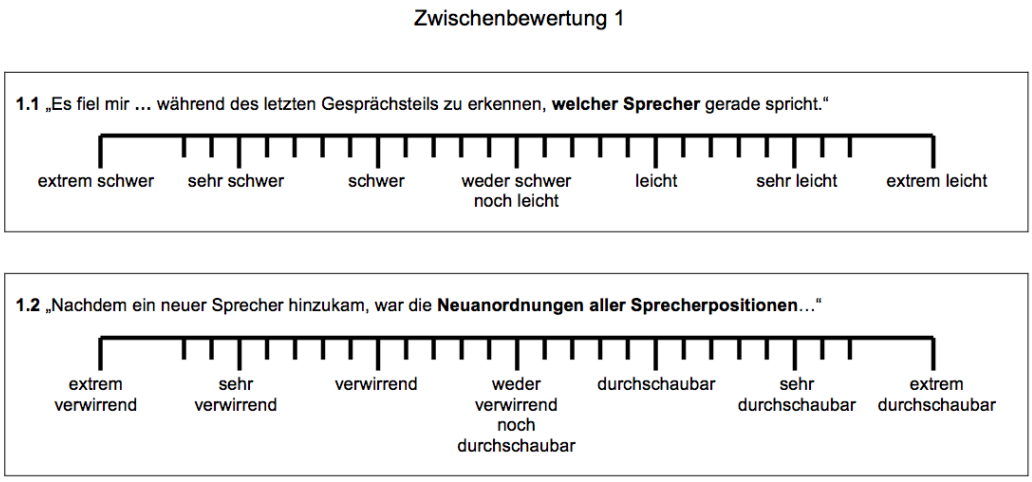


Figure F.1: Intermediate questionnaire

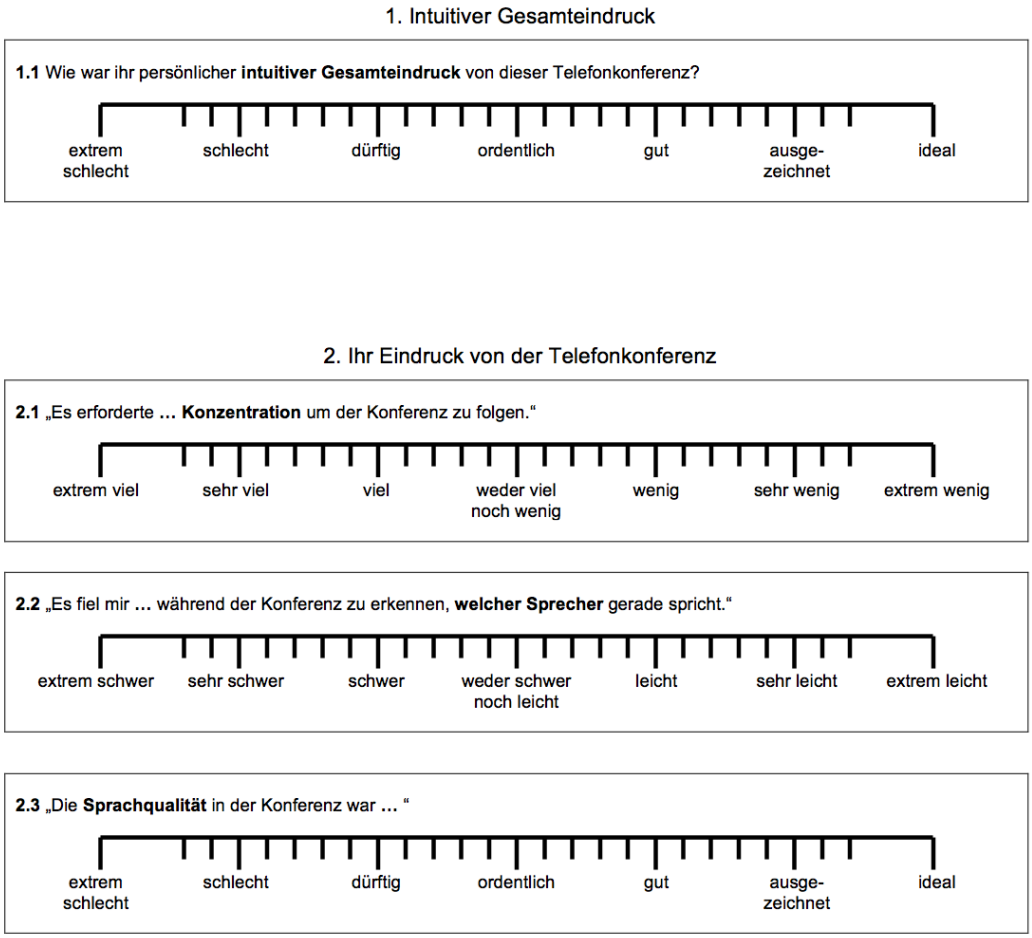


Figure F.2: Final questionnaire - page 1

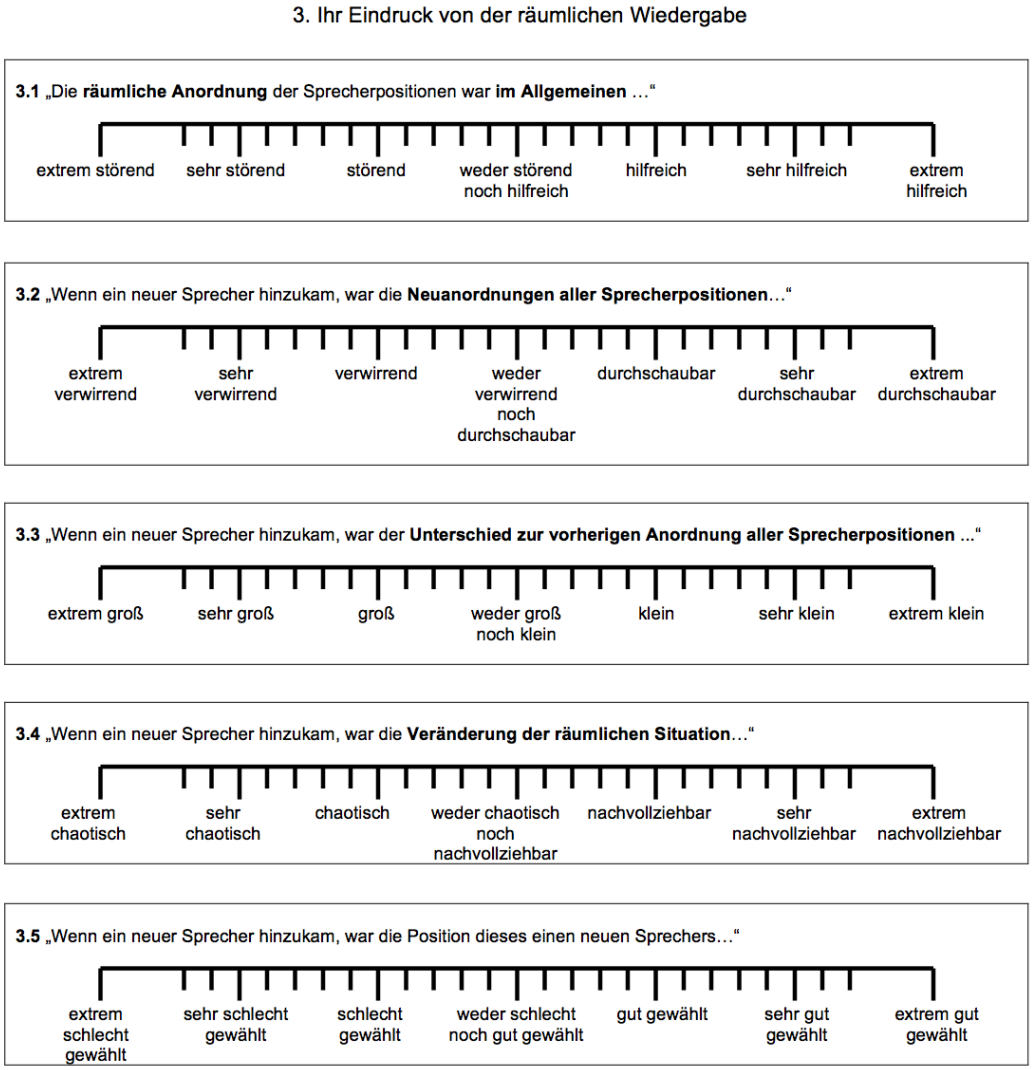


Figure F.3: Final questionnaire - page 2

Bibliography

[1] J. Baldis. Effects of spatial audio on memory, comprehension, and preference during desktop conferences. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '01, pages 166–173, New York, NY, USA, 2001. ACM.

[2] R. Kilgore, M. Chignell, and P. Smith. Spatialized audioconferencing: what are the benefits? In *Proceedings of the 2003 conference of the Centre for Advanced Studies on Collaborative research*, CASCON '03, pages 135–144. IBM Press, 2003.

[3] K. Crispien and T. Ehrenberg. Evaluation of the -cocktail-party effect- for multiple speech stimuli within a spatial auditory display. *J. Audio Eng. Soc*, 43(11):932–941, 1995.

[4] D. R. Begault, E. M. Wenzel, and M. R. Anderson. Direct comparison of the impact of head tracking, reverberation, and individualized head-related transfer functions on the spatial perception of a virtual speech source. *J. Audio Eng. Soc*, 49(10):904–916, 2001.

[5] M. Hyder, M. Haun, and C. Hoene. Measurements of sound localization performance and speech quality in the context of 3d audio conference calls. In *Internation Conference on Acoustics. Rotterdam, Netherlands: NAG/DAGA*, Mar. 2009.

[6] M. Hyder, M. Haun, and C. Hoene. Placing the participants of a spatial audio conference call. In *Consumer Communications and Networking Conference (CCNC), 2010 7th IEEE*, pages 1–7, 2010.

[7] A. Raake and C. Schlegel. Auditory assessment of conversational speech quality of traditional and spatialized teleconferences. In *Voice Communication (Sprachkommunikation), 2008 ITG Conference on*, pages 1–4, 2008.

[8] A. Raake, C. Schlegel, M. Geier, and J. Ahrens. Listening and conversational quality of spatial audio conferencing. In *Audio Engineering Society Conference: 40th International Conference: Spatial Audio: Sense the Sound of Space*, 2010.

[9] Doreen Kimura. Left-right differences in the perception of melodies. *Quarterly Journal of Experimental Psychology*, 16:355–358, 1964.

[10] J. Blauert. Spatial hearing : the psychophysics of human sound localization. *MIT, Cambridge, MA*, 1983.

[11] F. L. Wightman and D. J. Kistler. Psychoacoustical aspects of synthesized vertical locale cues. *J. Acoust. Soc. Am.*, 85:868–878, 1989b.

[12] F. L. Wightman and D. J. Kistler. Resolution of front-back ambiguity in spatial hearing by listener and source movement. *The Journal of the Acoustical Society of America*, 105(5):2841–2853, 1999.

[13] S. K. Roffler and R. A. Butler. Factors that influence the localization of sound in the vertical plane. *J. Acoust. Soc. Am.*, 43:1255–1259, 1968.

[14] L. R. Bernstein, C. Trahiotis, M. A. Akeroyd, and K. Hartung. Sensitivty to brief changes of interaural time and interaural intensity. *J. Acoust. Soc. Am.*, 109:1604–1616, 2001.

[15] A. J. Watkins. Psychoacoustical aspects of synthesized vertical locale cues. *J. Acoust. Soc. Am.*, 63:1152–1165, 1978.

[16] David A. Burgess. Techniques for low cost spatial audio. In *Proceedings of the 5th annual ACM symposium on User interface software and technology*, UIST '92, pages 53–59, New York, NY, USA, 1992. ACM.

[17] G. Wersenyi. Localization in a hrtf-based minimum-audible-angle listening test for guib applications. *Electronic Journal, Technical Acoustics*, 1, 2007.

[18] J. C. Middlebrooks. Narrow-band sound localization related to external ear acoustics. *J. Acoust. Soc. Am.*, 92:2607–2624, 1992.

[19] H. Fisher and S. J. Freedman. The role of the pinna in auditory localization. *J. Audiol. Research*, 8:15–26, 1968.

[20] W. M. Hartmann and B. Rakerd. On the minimum audible angle - a decision theory approach. *J. Acoust. Soc. Am.*, 85:2031–2041, 1989.

[21] T. Z. Strybel, C. L. Manlingas, and D. R. Perrott. Minimum audible movement angle as a function of azimuth and elevation of the source. *Human Factors*, 34(3):267–275, 1992.

[22] D. R. Perrott and A. D. Musicant. Minimum auditory movement angle: binaural localization of moving sources. *J. Acouts. Soc. Am.*, 62:1463–1466, 1977.

[23] J. Zwislocki and R. S. Feldman. Just noticeable differences in dichotic phase. *J. Acouts. Soc. Am.*, 28:860–864, 1956.

[24] P. A. Campbell. Just noticeable differences of changes of interaural time differences as a function of interaural time differences. *J. Acouts. Soc. Am.*, 31:917–922, 1959.

[25] B. G. Haustein and W. Schirmer. Messeinrichtung zur untersuchung des richtungslokalisationsvermoegens. *Hochfrequenztechnik und Elektroakustik*, 79:96–101, 1970.

[26] M. Geier, J. Ahrens, and S. Spors. The soundscape renderer: A unified spatial audio reproduction framework for arbitrary rendering methods, 2008.

[27] M. Geier, J. Ahrens, and S. Spors. Introduction to the soundscape renderer (ssr), 2012.

[28] Alexander Lindau and Stefan Weinzierl. Fabian - schnelle erfassung binauraler raumimpulsantworten in mehreren freiheitsgraden. *Fortschritte der Akustik, DAGA Stuttgart*, 2007.

[29] Helmut Wittek. University of surrey.

[30] A. Raake. Speech quality of voip : Assessment and prediction, 2006.

[31] A. Raake, S. Spors, J. Ahrens, and J. Ajmera. Concept and evaluation of a downward-compatible system for spatial teleconferencing using automatic speaker clustering.

[32] K. Inkpen, Hedge Rajesh, M. Czerwinski, and Z. Zhengyou. Exploring spatialized audio & video for distributed conversations, 2010.

[33] S. Moeller. Assessment and prediciton of speech quality in telecommunications. *Kluwer Academic Publishers, USA-Boston*, 2000.

[34] Beyerdynamic dt 290.

[35] Beyerdynamic dt 790.

[36] H. Fletcher and W. A. Munson. Loudness, its definition, measurement and calculation, 1933.

[37] H. Fletcher and W. A. Munson. Relation between loudness and masking, 1937.

[38] Normal equal-loudness-level contours. *Acoustics International Organization for Standarization (ISO) 226*, 2003.

[39] G. Seshadriaa and B. Yegnanarayana. Perceived loudness of speech based on the characteristics of glottal excitation source. *Acoustical Society of America*, pages 2061–2071, 2009.

[40] Objective measurement of active speech level. *ITU-T Series P: Terminals and Subjective and Objective Assessment Methods*, 2011.

[41] Sebastian Möller, Klaus-Peter Engelbrecht, Michael Pucher, Peter Fröhlich, Lu Huo, Ulrich Heute, and Frank Oberle. TIDE: A Testbed for Interactive Spoken Dialogue System Evaluation. In *Proceedings of the 12th International Conference on Speech and Computer (SPECOM 2007)*, Moskow, Russia, October 2007.

[42] ITU. *Definitions of terms related to quality of service (ITU-T Recommendation E.800).* International Telecommunications Union, 2008/2009.

[43] *Ergebnis Dagstuhl Seminar 2009, zit. nach Moeller 2010.*

[44] Alexander Raake. Speech quality of voip: Assessment and prediction. 2006.

[45] U. Jekosch. Sprache hoeren und beurteilen: Ein ansatz zur grundlegung der sprachqualitaetsbeurteilung. 2000.

[46] Andy Field. *Discovering Statistics using SPSS, Third Edition.* 2009.

List of Figures

List of Tables