



UNIVERSITAT POLITÈCNICA  
DE CATALUNYA

PhD Thesis

ROBUST SPEAKER DIARIZATION  
FOR MEETINGS

Author: Xavier Anguera Miró

Advisors: Dr. Francisco Javier Hernando Pericás (UPC)  
Dr. Chuck Wooters (ICSI)

Speech Processing Group  
Department of Signal Theory and Communications  
Universitat Politècnica de Catalunya

Barcelona, October 2006



*Als meus pares,  
(To my parents)*



# Abstract

This thesis shows research performed into the topic of speaker diarization for meeting rooms. It looks into the algorithms and the implementation of an offline speaker segmentation and clustering system for a meeting recording where usually more than one microphone is available. The main research and system implementation has been done while visiting the International Computer Science Institute (ICSI, Berkeley, California) for a period of two years.

Speaker diarization is a well studied topic on the domain of broadcast news recordings. Most of the proposed systems involve some sort of hierarchical clustering of the data into clusters, where the optimum number of speakers or their identities are unknown a priori. A very commonly used method is called bottom-up clustering, where multiple initial clusters are iteratively merged until the optimum number of clusters is reached, according to some stopping criterion. Such systems are based on a single channel input, not allowing a direct application for the meetings domain. Although some efforts have been done to adapt such systems to multichannel data, at the start of this thesis no effective implementation had been proposed. Furthermore, many of these speaker diarization algorithms involve some sort of model training or parameter tuning using external data, which impedes its usability with data different from what they have been adapted to.

The implementation proposed in this thesis works towards solving the aforementioned problems. Taking the existing hierarchical bottom-up mono-channel speaker diarization system from ICSI, it first uses a flexible acoustic beamforming to extract speaker location information and obtain a single enhanced signal from all available microphones. It then implements a train-free speech/non-speech detection on such signal and processes the resulting speech segments with an improved version of the mono-channel speaker diarization system. Such system has been modified to use speaker location information (when available) and several algorithms have been adapted or created new to adapt the system behavior to each particular recording by obtaining information directly from the acoustics, making it less dependent on the development data.

The resulting system is flexible to any meetings room layout regarding the number of microphones and their placement. It is train-free making it easy to adapt to different sorts of data and domains of application. Finally, it takes a step forward into the use of parameters that are more robust to changes in the acoustic data. Two versions of the system were submitted with excellent results in RT05s and RT06s NIST Rich Transcription evaluations for meetings, where data from two different subdomains (lectures and conferences) was evaluated. Also, experiments using the RT datasets from all meetings evaluations were used to test the different proposed algorithms proving their suitability to the task.



# Resum

Aquesta tesi doctoral mostra la recerca feta en l'àrea de la diarització de locutor per a sales de reunions. En la present s'estudien els algorismes i la implementació d'un sistema en diferit de segmentació i aglomerat de locutor per a grabacions de reunions a on normalment es té accés a més d'un micròfon per al processat. El bloc més important de recerca s'ha fet durant una estada al International Computer Science Institute (ICSI, Berkeley, Califòrnia) per un període de dos anys.

La diarització de locutor s'ha estudiat força per al domini de grabacions de ràdio i televisió. La majoria dels sistemes proposats utilitzen algun tipus d'aglomerat jeràrquic de les dades en grups acústics a on de bon principi no se sap el número de locutors òptim ni tampoc la seva identitat. Un mètode molt comunament utilitzat s'anomena "bottom-up clustering" (aglomerat de baix-a-dalt), amb el qual inicialment es defineixen molts grups acústics de dades que es van ajuntant de manera iterativa fins a obtenir el nombre òptim de grups tot i acomplint un criteri de parada. Tots aquests sistemes es basen en l'anàlisi d'un canal d'entrada individual, el qual no permet la seva aplicació directa per a reunions. A més a més, molts d'aquests algorismes necessiten entrenar models o afinar els paràmetres del sistema usant dades externes, el qual dificulta l'aplicabilitat d'aquests sistemes per a dades diferents de les usades per a l'adaptació.

La implementació proposada en aquesta tesi es dirigeix a solventar els problemes mencionats anteriorment. Aquesta pren com a punt de partida el sistema existent al ICSI de diarització de locutor basat en l'aglomerat de "baix-a-dalt". Primer es processen els canals de grabació disponibles per a obtenir un sol canal d'àudio de qualitat major, a més d'informació sobre la posició dels locutors existents. Aleshores s'implementa un sistema de detecció de veu/silenci que no requereix de cap entrenament previ, i processa els segments de veu resultant amb una versió millorada del sistema mono-canal de diarització de locutor. Aquest sistema ha estat modificat per a l'ús de l'informació de posició dels locutors (quan es tingui) i s'han adaptat i creat nous algorismes per a que el sistema obtingui tanta informació com sigui possible directament del senyal acústic, fent-lo menys dependent de les dades de desenvolupament.

El sistema resultant es flexible i es pot usar en qualsevol tipus de sala de reunions pel que fa al nombre de micròfons o la seva posició. El sistema, a més, no requereix en absolut dades d'entrenament, sent més senzill adaptar-lo a diferents tipus de dades o dominis d'aplicació. Finalment, fa un pas endavant en l'ús de paràmetres que siguin més robusts als canvis en les dades acústiques. Dos versions del sistema es van presentar amb resultats excel·lents a les evaluacions de RT05s i RT06s del NIST en transcripció rica per a reunions, a on aquests es van avaluar amb dades de dos subdominis diferents (conferències i reunions). A més a més, es fan experiments utilitzant totes les dades de les evaluacions RT per a demostrar la viabilitat dels algorismes proposats en aquesta tasca.



# Acknowledgements

It is always difficult to find a good balance on who to thank in order not to leave anyone important behind and not to become too long. Everyone's life changes constantly, influenced by the people in it. In my life there have been some people that have had an influence in who I have become, to whom I have to be thankful for.

First of all, to my parents and grandparents. I have always felt they were behind me, helping me to overcome any rock on my road, and giving me the opportunity to always pursue my dreams. This is to my parents to whom I specially dedicate the effort put into writing this thesis.

I would certainly not be writing this if it was not for Javier Hernando, my co-advisor, who became my friend a few years ago, then my co-worked, then my advisor. I thank him for always believing in me and for letting me find my path.

Although I met him later than Javier, Chuck Wooters, my second co-supervisor, has been an incredible source of knowledge, advice and friendship during my stay in Berkeley. He was always welcoming me with a smile and ready to help. It soon became clear we both shared a passion for speech processing and very similar ideals. It has been a pleasure and an honor being able to work with him.

The two years doing research at ICSI in Berkeley have been very important for me and for this PhD. They would not have been possible without the AMI project training program, the Spanish visitors program and ICSI for hosting me. Many thanks go to Barbara Peskin, Nelson Morgan and everyone responsible for both programs.

In my path through education there have been many good friends and partners of sacrifice in which I have looked upon for motivation, support and enjoyment. In these later PhD years I would like to always remember people like Pablo, Jan, Mireia, Pere, Andrey, Marta, Jordi and Josep Maria at UPC, and Manolo, Kofi, Marc, Arlo, James, Andy, Adam and Mathew at ICSI, just to name a few.

Over the years I started learning some dancing steps until now when it became not only a good exercise activity but also a relaxation method. Thanks to all my tango and ballroom partners for suffering my stumbles and grumpy days.

Finally, thank you for reading this thesis.

Tarragona, Spain  
October , 2006

Xavier Anguera Miró



# Contents

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Introduction</b>   | <b>1</b>  |
| 1.1      | Context and Motivations of this Thesis . . . . .            | 1         |
| 1.2      | Definition of the Thesis Objectives . . . . .               | 5         |
| 1.3      | Outline of the Thesis . . . . .                             | 6         |
| <b>2</b> | <b>State of the art</b>                                     | <b>9</b>  |
| 2.1      | Acoustic Features for Speaker Diarization . . . . .         | 11        |
| 2.2      | Speaker Segmentation . . . . .                              | 13        |
| 2.2.1    | Metric-Based Segmentation . . . . .                         | 14        |
| 2.2.2    | Non Metric-Based Segmentation . . . . .                     | 22        |
| 2.3      | Speaker Diarization . . . . .                               | 24        |
| 2.3.1    | Hierarchical Clustering Techniques . . . . .                | 25        |
| 2.3.2    | Other Clustering Techniques . . . . .                       | 31        |
| 2.3.3    | Use of Support Information in Diarization . . . . .         | 32        |
| 2.4      | Speaker Diarization in Meetings . . . . .                   | 33        |
| 2.4.1    | Current Meeting Room Research Projects . . . . .            | 33        |
| 2.4.2    | Databases . . . . .   | 35        |
| 2.4.3    | NIST RT Speaker Diarization Systems for Meetings . . . . .  | 36        |
| 2.5      | Multichannel Acoustic Enhancement . . . . .                 | 40        |
| 2.5.1    | Introduction to Acoustic Array Processing . . . . .         | 41        |
| 2.5.2    | Microphone Array Beamforming . . . . .                      | 45        |
| 2.5.3    | Time Delay of Arrival Estimation . . . . .                  | 47        |
| <b>3</b> | <b>Speaker Diarization: from Broadcast News to Meetings</b> | <b>51</b> |
| 3.1      | The ICSI Broadcast News System . . . . .                    | 52        |

|          |   |            |
|----------|---|------------|
| 3.1.1    | Speech/non-Speech Detection and Parameters Extraction . . . . .           | 53         |
| 3.1.2    | Clusters Initialization and Acoustic Modeling . . . . .                   | 56         |
| 3.1.3    | Clusters Comparison, Pruning and Clusters Merging . . . . .               | 58         |
| 3.1.4    | Stopping Criterion and System Output . . . . .                            | 60         |
| 3.2      | Analysis of Differences from Broadcast News to Meetings . . . . .         | 61         |
| 3.2.1    | Input Data Analysis: Broadcast News versus Meetings . . . . .             | 61         |
| 3.2.2    | Summary of Differences and Proposed Changes . . . . .                     | 72         |
| 3.3      | Robust Speaker Diarization System for Meetings . . . . .                  | 73         |
| 3.3.1    | Acoustic Signal Enhancement . . . . .                                     | 73         |
| 3.3.2    | Single Channel System Frontend . . . . .                                  | 75         |
| 3.3.3    | Speaker Clusters and Models Initialization . . . . .                      | 75         |
| 3.3.4    | Clusters Merging and System Output . . . . .                              | 82         |
| <b>4</b> | <b>Acoustic Modeling Algorithms for Speaker Diarization in Meetings</b>   | <b>85</b>  |
| 4.1      | Speech/Non-Speech Algorithm . . . . .                                     | 86         |
| 4.1.1    | Energy-Based Speech/non-Speech Detector with Variable threshold . . . . . | 87         |
| 4.1.2    | Model-based Speech/Non-Speech Decoder . . . . .                           | 90         |
| 4.1.3    | Hybrid Speech/non-Speech Detection . . . . .                              | 91         |
| 4.2      | Speaker Clusters Description and Modeling . . . . .                       | 92         |
| 4.2.1    | Friends-and-Enemies Initialization . . . . .                              | 92         |
| 4.2.2    | Clusters and Models Complexity Selection . . . . .                        | 96         |
| 4.2.3    | Acoustic Modeling without Time Restrictions . . . . .                     | 99         |
| 4.3      | Cluster Purification Algorithms . . . . .                                 | 101        |
| 4.3.1    | Frame-Level Cluster Purification . . . . .                                | 103        |
| 4.3.2    | Segment-Level Cluster Purification . . . . .                              | 107        |
| <b>5</b> | <b>Multichannel Processing for Meetings</b>                               | <b>109</b> |
| 5.1      | Multichannel Acoustic Beamforming for Meetings . . . . .                  | 110        |
| 5.1.1    | Meeting Room Microphone Array Characteristics . . . . .                   | 110        |
| 5.1.2    | Filter-and-Sum Beamforming . . . . .                                      | 111        |
| 5.2      | Multichannel Acoustic Beamforming System Implementation . . . . .         | 113        |
| 5.2.1    | Individual Channels Signal Enhancement . . . . .                          | 113        |
| 5.2.2    | Meeting Information Extraction . . . . .                                  | 115        |

|          |   |            |
|----------|---|------------|
| 5.2.3    | TDOA Values Selection . . . . .   | 120        |
| 5.2.4    | Output Signal Generation . . . . .  | 126        |
| 5.3      | Use of the Estimated Delays for Speaker Diarization . . . . .                     | 129        |
| 5.3.1    | TDOA Modeling and Features Fusion . . . . .                                       | 130        |
| 5.3.2    | Automatic Features Weight Estimation . . . . .                                    | 134        |
| <b>6</b> | <b>Experiments</b>  | <b>139</b> |
| 6.1      | Meetings Domain Experiments Setup . . . . .                                       | 139        |
| 6.1.1    | Baseline Systems . . . . .  | 140        |
| 6.1.2    | Databases . . . . .   | 141        |
| 6.1.3    | Evaluation Metrics . . . . .  | 142        |
| 6.1.4    | Reference Segmentation Selection and Calculation . . . . .                        | 144        |
| 6.2      | Experiments from Broadcast News to Meetings . . . . .                             | 147        |
| 6.3      | Speech/Non-Speech Detection Block . . . . .                                       | 148        |
| 6.4      | Acoustic Beamforming Experiments . . . . .  | 151        |
| 6.4.1    | Baseline System Analysis . . . . .  | 152        |
| 6.4.2    | Reference Channel Estimation Analysis . . . . .                                   | 153        |
| 6.4.3    | TDOA Post-Processing Analysis . . . . .   | 154        |
| 6.4.4    | Signal Output Algorithms Analysis . . . . .                                       | 158        |
| 6.4.5    | Use of the Beamformed Signal for ASR . . . . .                                    | 159        |
| 6.5      | Speaker Diarization Module Experiments . . . . .                                  | 159        |
| 6.5.1    | Individual Algorithms Performance . . . . .                                       | 160        |
| 6.5.2    | Algorithms Agglomeration Performance . . . . .                                    | 169        |
| 6.6      | Overall Experiments and Analysis of Results . . . . .                             | 176        |
| <b>7</b> | <b>NIST Evaluations in Speaker Diarization</b>                                    | <b>181</b> |
| 7.1      | NIST Rich Transcription Evaluations in Speaker Diarization for Meetings . . . . . | 182        |
| 7.1.1    | RT05s and RT06s Evaluation Conditions . . . . .                                   | 182        |
| 7.1.2    | Methodology of the Evaluations . . . . .  | 183        |
| 7.1.3    | Data used on the Speaker Diarization Evaluations . . . . .                        | 184        |
| 7.2      | ICSI Participation in the RT Evaluations . . . . .                                | 185        |
| 7.2.1    | Participation in the 2005 Spring Rich Transcription Evaluation . . . . .          | 185        |
| 7.2.2    | Participation in the 2006 Spring Rich Transcription Evaluation . . . . .          | 189        |

|          |  |            |
|----------|--|------------|
| 7.3      | Pros and Cons of the NIST Evaluations . . . . .    | 194        |
| <b>8</b> | <b>Conclusions</b>                                 | <b>197</b> |
| 8.1      | Overall Thesis Final Review . . . . .              | 197        |
| 8.2      | Review of Objectives Completion . . . . .          | 198        |
| 8.3      | Possible Future Work Topics . . . . .              | 200        |
| <b>A</b> | <b>BIC Formulation for Gaussian Mixture Models</b> | <b>203</b> |
| <b>B</b> | <b>Rich Transcription evaluation datasets</b>      | <b>207</b> |
|          | <b>Bibliography</b>                                |            |

# List of Tables

|      |  |     |
|------|--|-----|
| 3.1  | <i>DER improvement by using a speech/non-speech detector . . . . .</i>                                 | 55  |
| 3.2  | <i>Comparison of PLP12 and MFCC19 parameterizations on RT04f . . . . .</i>                             | 56  |
| 3.3  | <i>Comparison between BIC and Viterbi stopping criterions for RT04f data . . . . .</i>                 | 61  |
| 3.4  | <i>Estimated SNR for RT04f Broadcast News shows . . . . .</i>  | 63  |
| 3.5  | <i>Estimated SNR for RT06s Conference Meetings, MDM channels . . . . .</i>                             | 64  |
| 3.6  | <i>Estimated SNR for RT06s Conference Meetings, IHM channels . . . . .</i>                             | 64  |
| 3.7  | <i>Estimated SNR for RT06s Lecture Room Meetings, MDM channels . . . . .</i>                           | 65  |
| 3.8  | <i>Average total speaker duration in RT06s conference room data . . . . .</i>                          | 66  |
| 3.9  | <i>Average total speaker duration in RT06s lecture room data . . . . .</i>                             | 67  |
| 3.10 | <i>Average total speaker duration in RT04f broadcast news data . . . . .</i>                           | 68  |
| 3.11 | <i>Average number of speaker for Rt04f and RT06s . . . . .</i>   | 69  |
| 3.12 | <i>Speaker turn durations for RT04f and RT06s . . . . .</i>  | 69  |
| 3.13 | <i>Overlap analysis between hand and forced alignments in RT06s conference room meetings . . . . .</i> | 71  |
| 3.14 | <i>Main differences between Meetings and Broadcast News recordings . . . . .</i>                       | 72  |
| 6.1  | <i>Summary of datasets used in the experiments . . . . .</i>   | 142 |
| 6.2  | <i>Results for the CV-EM training algorithm in the agglomerate system . . . . .</i>                    | 147 |
| 6.3  | <i>Speech/non-speech errors on development and test data . . . . .</i>                                 | 150 |
| 6.4  | <i>DER using different speech/non-speech systems . . . . .</i>   | 150 |
| 6.5  | <i>RT06s system filter-and-sum performance . . . . .</i>   | 152 |
| 6.6  | <i>Reference channel estimation results . . . . .</i>  | 153 |
| 6.7  | <i>Post-processing algorithms results . . . . .</i>  | 154 |
| 6.8  | <i>Results for various percentages of noise thresholding . . . . .</i>                                 | 155 |
| 6.9  | <i>Results for alternative Viterbi weights . . . . .</i>   | 157 |

|      |  |     |
|------|--|-----|
| 6.10 | <i>Results for alternative Viterbi weights</i>   | 158 |
| 6.11 | <i>Results for relative channel weights and elimination of bad channels</i>  | 158 |
| 6.12 | <i>WER using RT06s ASR system including the presented beamformer</i>   | 159 |
| 6.13 | <i>DER for the development and evaluation sets for models complexity selection and initial number of clusters algorithms</i> | 163 |
| 6.14 | <i>Average DER for the development set using various possible distance metrics and initial segment selection criterions</i>  | 164 |
| 6.15 | <i>DER results for the segment and frame purification algorithms</i>   | 166 |
| 6.16 | <i>DER results for different weight selection algorithms</i>   | 168 |
| 6.17 | <i>Summary of DER improvements for the individual algorithms in the development set</i>                                      | 169 |
| 6.18 | <i>Results for the model complexity and number of initial clusters in the agglomerate system</i>                             | 171 |
| 6.19 | <i>Results for the friends-and-enemies algorithm</i>   | 172 |
| 6.20 | <i>Results for the CV-EM training algorithm in the agglomerate system</i>  | 174 |
| 6.21 | <i>Results for the frame purification algorithm in the agglomerate system</i>  | 175 |
| 6.22 | <i>Results for the segment purification algorithm in the agglomerate system</i>  | 176 |
| 6.23 | <i>Summary of average DER for the agglomerate system on development and evaluation data</i>                                  | 176 |
| 6.24 | <i>Results for the TDOA-MDM task using different algorithm settings</i>  | 177 |
| 6.25 | <i>Overall thesis scores comparison</i>  | 178 |
| 6.26 | <i>Overall thesis DER scores comparison</i>  | 179 |
| 7.1  | <i>Systems summary description and DER on the evaluation set for RT05s</i>   | 188 |
| 7.2  | <i>Results for RT06s Speaker Diarization, conference room environment</i>  | 191 |
| 7.3  | <i>Results for RT06s Speaker Diarization, lecture room environment</i>   | 191 |
| 7.4  | <i>Results for RT06s Speech Activity Detection (SAD). Results with * are only for a subset of segments</i>                   | 193 |
| B.1  | <i>Summary of datasets used in the experiments</i>   | 208 |

# List of Figures

|      |  |     |
|------|--|-----|
| 2.1  | <i>Graphic interpretation of the most common clustering techniques . . . . .</i>                               | 25  |
| 2.2  | <i>Example of a passive aperture response to different incoming signals . . . . .</i>                          | 43  |
| 3.1  | <i>ICSI Speaker Diarization for Broadcast News blocks diagram . . . . .</i>                                    | 54  |
| 3.2  | <i>Acoustic models for speaker clustering . . . . .</i>  | 57  |
| 3.3  | <i>Speaker turn duration histograms . . . . .</i>  | 70  |
| 3.4  | <i>Overlap histograms in RT06s conference room meetings . . . . .</i>  | 71  |
| 3.5  | <i>Main blocks involved in the meetings speaker diarization system . . . . .</i>                               | 73  |
| 3.6  | <i>single-channel speaker diarization for meetings block diagram . . . . .</i>                                 | 76  |
| 3.7  | <i>Speaker models initialization based on Gaussian splitting . . . . .</i>                                     | 78  |
| 3.8  | <i>Cross-validation EM training algorithm . . . . .</i>  | 80  |
| 4.1  | <i>Energy-based detector blocks diagram . . . . .</i>  | 87  |
| 4.2  | <i>Left, filter over <math>\tilde{e}[n]</math>. Decision of silence in red after the thresholding. . . . .</i> | 89  |
| 4.3  | <i>State machine used to apply time constraints. . . . .</i>   | 90  |
| 4.4  | <i>Hybrid Speech/non-speech detector blocks diagram . . . . .</i>  | 91  |
| 4.5  | <i>Clusters initialization blocks diagram . . . . .</i>  | 94  |
| 4.6  | <i>Friends-and-enemies clusters initialization process . . . . .</i>   | 96  |
| 4.7  | <i>Cluster models with Minimum duration and modified probabilities . . . . .</i>                               | 100 |
| 4.8  | <i>Possible Speaker clustering errors due to clusters purity problems . . . . .</i>                            | 102 |
| 4.9  | <i>Speech-silence histogram for a full meeting . . . . .</i>   | 103 |
| 4.10 | <i>Observed assignment of frames to Gaussian mixtures . . . . .</i>  | 104 |
| 4.11 | <i>Evaluation of metric 1 on two clusters given their models . . . . .</i>                                     | 105 |
| 4.12 | <i>Speech/non-speech histograms for different possible model complexities . . . . .</i>                        | 106 |
| 5.1  | <i>Linear microphone array with all microphones equidistant at distance <math>d</math> . . . . .</i>           | 111 |

|      |  |     |
|------|--|-----|
| 5.2  | <i>Filter and sum algorithm blocks diagram . . . . .</i>   | 113 |
| 5.3  | <i>filter-and-sum implementation blocks diagram . . . . .</i>  | 114 |
| 5.4  | <i>Cross-correlation values histograms for RT06s AMI meetings . . . . .</i>  | 121 |
| 5.5  | <i>Filter and Sum double-Viterbi delays selection . . . . .</i>  | 122 |
| 5.6  | <i>Two-step TDOA Viterbi decoding example, step 1 . . . . .</i>  | 124 |
| 5.7  | <i>Two-step TDOA Viterbi decoding example, step 1 for an individual channel . . .</i>  | 125 |
| 5.8  | <i>Two-step TDOA Viterbi decoding example, step 2 . . . . .</i>  | 126 |
| 5.9  | <i>Multichannel delayed signal sum using a triangular window . . . . .</i>   | 129 |
| 5.10 | <i>Locations information contained in the TDOA values . . . . .</i>  | 131 |
| 5.11 | <i>Fusion of TDOA values and acoustic features within the speaker diarization module</i>                                     | 132 |
| 5.12 | <i>First merge cluster-pair BIC values and histogram for acoustic and TDOA features</i>                                      | 135 |
| 5.13 | <i>Acoustic weight evolution with the number of iterations for meeting<br/>CMU_20050912-0900 . . . . .</i>                   | 137 |
| 6.1  | <i>Energy-based system errors depending on its segment minimum duration . . . . .</i>  | 148 |
| 6.2  | <i>Model-based system errors depending on its segment minimum duration . . . . .</i>   | 149 |
| 6.3  | <i>Individual meetings DER vs. SNR vs. number of microphones in the RT06s system</i>   | 152 |
| 6.4  | <i>Development set SNR modifying the percentage of noise threshold adjustment . .</i>  | 155 |
| 6.5  | <i>Development set SNR values modifying the Viterbi transition prob. weights in the<br/>F&amp;S algorithm . . . . .</i>      | 156 |
| 6.6  | <i>Development set SNR values modifying the number of N-best values used for<br/>TDOA selection . . . . .</i>                | 157 |
| 6.7  | <i>DER for the model complexity selection algorithm using different CCR values . .</i>                                       | 161 |
| 6.8  | <i>DER for the initial number of clusters algorithm using different CCR values . . .</i>                                     | 162 |
| 6.9  | <i>DER for the combination of complexity selection + initial number of clusters using<br/>different CCR values . . . . .</i> | 162 |
| 6.10 | <i>DER variation with the number of parallel models used in CV-EM training . . . .</i>                                       | 163 |
| 6.11 | <i>DER variation with the number of friends used in the friends-and-enemies ini-<br/>tialization . . . . .</i>               | 165 |
| 6.12 | <i>DER variation with the percentage of accepted frames and used Gaussians in<br/>frame purification . . . . .</i>           | 166 |
| 6.13 | <i>DER scores for the baseline system setting the relative weights by hand on devel-<br/>opment data . . . . .</i>           | 166 |
| 6.14 | <i>DER evolution with the weight computation iterations . . . . .</i>  | 167 |
| 6.15 | <i>DER evolution changing the initial feature stream weights . . . . .</i>   | 168 |

|      |   |     |
|------|---|-----|
| 6.16 | <i>DER variation with the number of Gaussian mixtures initially assigned to the TDOA models . . . . .</i> | 170 |
| 6.17 | <i>DER variation with the CCR parameter in the agglomerate system . . . . .</i>                           | 171 |
| 6.18 | <i>DER variation with the number of friends in the agglomerate system . . . . .</i>                       | 172 |
| 6.19 | <i>DER variation with the number of EM iterations of a standard EM-ML training algorithm . . . . .</i>    | 173 |
| 6.20 | <i>DER variation with the number CV-EM parallel models . . . . .</i>                                      | 173 |
| 6.21 | <i>DER variation with the frame % acceptance for frame purification algorithm . . .</i>                   | 174 |
| 6.22 | <i>DER variation with the Gaussian % used in the frame purification algorithm . .</i>                     | 175 |
| 7.1  | <i>DER Break-down by meeting for the RT05s conference data . . . . .</i>                                  | 188 |
| 7.2  | <i>DER break-down by show for the RT05s lecture data . . . . .</i>  | 189 |
| 7.3  | <i>DER break-down by show for the RT06s conference data . . . . .</i>                                     | 192 |
| 7.4  | <i>DER break-down by show for the RT06s lecture data . . . . .</i>  | 193 |



# Chapter 1

## Introduction

The purpose of this initial chapter is to present the problems and motivations that sparked and impelled the development of this thesis work and exposes what it aimed to achieve. Working towards this, section 1.1 introduces the topic of the thesis and the motivations behind it. Section 1.2 defines the set of objectives pursued with its development. Finally, section 1.3 outlines the contents to be found in each of the remaining chapters in this document.

### 1.1 Context and Motivations of this Thesis

#### People like to talk

Speech is still one of the most used way that humans have to communicate their ideas and to convey information to the world outside of ourselves. In fact, the quantity of available information by means of speech (telephone, radio, television, meetings, lectures, internet, etc) that is being stored is very big and rapidly increasing given the cheaper and cheaper ways of storage available nowadays. Following the two maxims that say "time is money" and "information is power", it becomes clear how desirable it is to have access to all this information, but as we only have two ears and limited time, we would like someone else to access it for us and to tell us only what is important, not wasting time in listening to multiple hours of contentless recordings. Some other times we might be interested in accessing some particular bit of this information which we do not know where it is, lost inside of our "Alexandria audio library". This is one area where speech technology can make a big contribution by means of techniques like audio indexing, where information is automatically extracted from the audio, which allows the processing, search and recovery of the desired content much easier. Considering a parallelism, acoustic indexing could be considered to an audio-based library what a good librarian is to a paper-based library.

## People do not like, usually, to talk alone

Most of the times when a person speaks, his/her speech is directed to someone or something else, which we expect to communicate with. In fact, even when we are talking to an animal, a machine or a little baby we are adapting our speech so that the message is conveyed to this outer entity. When dealing with information extraction from a recording, it becomes very important to answer questions like: “what was said?” as it conveys the message, but also “who said it?” as information varies depending on who utters the spoken words.

Within the speech technologies, The broad topic of *acoustic indexing* studies the classification of sounds into different classes/sources. Such classes could be as broad as [cats, dogs, humans] or more concrete like [pit bull, pug, German shepherd]. Algorithms used for acoustic indexing worry about the correct classification of the sounds, but not necessarily about the correct separation of them when more than one exist in the same audio segment. These purely classification techniques have sometimes been called *audio clustering*, which benefit from the broad topic of clustering, well studied in many areas.

When multiple sounds appear in the same audio signal one must turn his attention to techniques called as *audio diarization* to process them. As described in Reynolds and Torres-Carrasquillo (2004), audio diarization is known as the process of annotating an input audio signal with information that attributes (possibly overlapping) temporal regions of signal to their specific sources/classes (i.e. creating a “diary” of events in the audio document). These can include particular speakers, music, background noise sources, and other signal source/channel characteristics. It is very dependent on the application which particular classes are defined, becoming as broad or narrow as intended. In the simplest case, one could refer as audio diarization to the task of speech versus non-speech detection.

When the possible classes correspond to the different speakers in a recording these techniques are called *speaker diarization*. They aim at answering the question “Who spoke when?” given an audio signal. Algorithms doing speaker diarization need to locate each speaker turn and assign them to the appropriate speaker cluster. The output of the system is a set of segments with a unique ID assigned to each person that intervenes in the recording, leaving it to speaker identification systems to determine the person’s identity given each ID. Until the present time, the domains that have received most research attention within the speaker diarization community have been

- Telephone speech: Speaker diarization systems started being evaluated by NIST (*National Institute for Standards and Technology* 2006) using single channel telephone speech signals, within the speaker recognition evaluations in the late 1990’s.
- Broadcast News (radio and TV broadcasts): Mainly with the impulse of DARPA’s EARS

program (*DARPA Effective, Affordable, Reusable Speech-to-Text (EARS) 2004*) rich transcription of broadcasted news content became the primary research domain for speaker diarization roughly from 2002 to 2004. Rich transcription consists on the addition of extra information (generally called metadata, including speaker diarization information) to the speech-to-text transcriptions.

- Meetings (lectures and conferences): Mainly due to the impulse of the European CHIL and AMI projects (*Computers in the Human Interaction Loop (CHIL) website* (2006), *Augmented Multiparty Interaction (AMI) website* (2006)) the focus of research shifted from broadcast news to meetings around 2004. Although of its current prominence, many smaller projects had studied and recorded meetings previously in the 1990's.

When talking about speaker diarization is equivalent to saying *speaker segmentation and clustering* of an audio document as both these techniques are normally used together in diarization. On one hand, speaker segmentation (also called *speaker change detection*) aims at finding changes of speaker in an audio recording. It differs from *acoustic change detection* in that it does not consider changes in the background sounds during a single speaker segment to be a change to consider. On the other hand, speaker clustering agglomerates audio segments into homogeneous groups, coming from a similar/same source. In the general definition it does not constrain the process to a single file as all it requires is that each segment contain only a single speaker. When used in conjunction with speaker segmentation for speaker diarization, it clusters the segments created by the segmentation of one single recording.

Finally, also related to speaker diarization there are techniques regarding *speaker tracking* where the identity of one or more speakers is known a priori and the aim is to locate their interventions within the audio document.

In a general point of view, speaker diarization algorithms are a very useful part of many speech technology systems, for example:

- Speaker indexing and rich transcription: By indexing the audio according to the speakers and adding extra information to speech transcripts it becomes easier for humans to locate information and for machines to process it. typical automatic uses of such output might be speech summarization and translation.
- Speaker segmentation and clustering helping Automatic Speech Recognition (ASR) systems: Segmentation algorithms are used to split the audio into small segments (maintaining all acoustic units intact) for the ASR systems to process. Also, speaker diarization algorithms are used to cluster all the input data into speakers towards model adaptation. Sometimes the clustering is performed into broader speaker clusters (less than the actual amount of speakers) to maximize the amount of adaptation data.

- Preprocessing modules for speaker-based algorithms: Speaker diarization can be used before speaker tracking, speaker identification, speaker verification and other single speaker-based algorithms, to split the data into individual speakers.

This thesis verses about speaker diarization pursued on the meetings environment. While doing so, and following the guidelines proposed by NIST in the Rich Transcription (RT) evaluations, it processes the data without any prior information on the number of speakers present in the meeting or their identities. Furthermore, the system is intended for use without any assumption on the meeting room layout, which usually contains multiple microphones recording synchronously. These microphones are of different kinds and it is assumed that their exact location is unknown to the system.

This thesis is being presented in partial fulfillment of the requirements for the PhD in Theory of Signal and Communications in the UPC, where I have taken the necessary doctorate courses and previously prepared the thesis proposal. The proposed system was implemented at the International Computer Science Institute during a two years research stay with funding from the AMI project on the first year and the Spanish visitors program on the second. The implementation of speaker diarization for meetings takes into account all available prior knowledge in speaker diarization for the broadcast news environment present at ICSI at the start of this project. It is based on a modified version of the Bayesian Information Criterion (BIC) which does not require the tuning of any penalty term. It performs an agglomerative clustering of the initial acoustic data after it has been filtered with a speech/non-speech detector to eliminate all non-speech information. The clustering finished when no cluster pair is available for merging.

Improvements are proposed for the system in three main areas. To extend the applicability of the system to multiple microphone recordings it implements a filter&sum beamforming and adds several algorithms to improve the output signal when microphones are very dissimilar. The beamforming algorithm also started being used by the ASR system at ICSI in the meetings Rich Transcription evaluations with great results directly attributed to this module. Another area is the speech/non-speech detection where a new train-free system was implemented to allow for an accurate filtering of the silence segments in a meeting. Finally, within the inherited broadcast news system, several algorithms are either added or improved to increase the robustness of the system and to allow for it to extract as much information as possible from each recording allowing for fast adaptation to new domains. These include the automatic initial number of clusters and model complexity selection algorithms, two purification algorithms to allow better comparisons between clusters and a more robust training. Finally, the time delay of arrival between microphones in the beamforming module is successfully used in the diarization to increase the amount of information used to perform the diarization.

## 1.2 Definition of the Thesis Objectives

The main objective of this thesis is the development of a robust speaker diarization system towards its use in the meetings domain. In order to fully accomplish this, a set of concrete objectives is established (without any order of importance):

- The speaker diarization system is to be built using the expertise accumulated at ICSI in the research done in broadcast news. First, the differences between broadcast news and meetings need to be analyzed. Then the mono-channel speaker diarization system used in broadcast news is to be adapted to the meetings domain by first addressing the points where both domains differ, and then improving current algorithms to improve its performance.
- The resulting system is to be as independent as possible to any room distribution, number of microphones and placement, and kind of meeting. It should also be easy to be adapted to new domains with as little development time as possible. Within the meetings domain, algorithms should be able to obtain automatically all necessary parameters in each meeting and algorithms should work for all possible meeting conditions with acceptable performance. When porting the system to new domains it should perform well from the start.
- The algorithms implemented for the meetings system should reduce *show flakiness*, which accounts for sudden changes to the system performance, within the same set, upon slight modification of its parameter settings. It should also improve within sets *robustness*, with similar results when running the same system in different data than the development. This can be achieved by research in system parameters that focus on the particular characteristics of the individual audio excerpts instead of the whole set, thus becoming more robust to changes in the used set. These parameters need to have a flat performance response around the optimum to allow for small changes not to dramatically affect the outcome.
- In a similar fashion, the system is also aimed at being train-free (no external data is used to train acoustic models prior to the test). This allows both a quick adaptation to domains and a robust performance when new data within the same domain has a different acoustic content than the development data. This was already a goal of the broadcast news system, where only the speech/non-speech detector needed to be trained. The proposed system aims at replacing this module by a train-free alternative and to implement all new algorithms and improvements to be independent of any data outside of the test set for models training. Still development data will be used to set the system parameters.
- The system is developed for participation in the NIST Rich Transcription (RT) evaluations for 2005 and 2006 in order to benchmark the performance of the technology and algorithms

implemented in comparison to other systems given the same data. All decisions taken and parameter settings are in tune with the existing rules in these evaluations, which intend to measure general system performance, without emphasis in any particular application.

- Last but not least, emphasis is put at the publication of results and improvements made to the system to allow for other research groups to know the research progress made at ICSI in terms of speaker diarization. Furthermore, efforts are made into making the system available for people to use it, either entirely or some of its modules, and both internally or by external users, giving support when possible.

### 1.3 Outline of the Thesis

This thesis is split into seven main chapters on the topic of robust speaker diarization for meetings. A brief description follows of what is to be found in each chapter.

Chapter 2 takes a look into the proposed problem: how to robustly and optimally determine “who spoke when?” in a meeting domain where multiple microphones are usually available for recording. In order to address it, a review of what feature parameters have been previously used in speaker-related problems is followed by an analysis of the state of the art in speaker segmentation, which plays an important part in many speaker diarization algorithms. Then a review of previously proposed diarization algorithms and implementations sets the grounds for a description of the projects, databases and systems that, to the date, have had its main focus in the meetings domain. Finally, and given the multichannel nature of a meeting room, acoustic enhancement theory is introduced to process multiple microphones, and the main techniques are reviewed for the purpose of obtaining a single “enhanced” channel from multiple inputs.

Chapter 3 Leads the reader through the system implementation, basing it in the diarization system that existed for broadcast news prior to this thesis work. An initial review of the ideas behind the system and the implementation of the broadcast news speaker diarization system is followed by an analysis of the differences and needs in order to adapt it to the meetings domain. Finally, a description of the meetings implementation in comparison to the prior system is pursued. Each of the blocks and algorithms that have been reused, refurbished or created from scratch for the meetings domain are introduced, while leaving for later chapters the description in detail of the novel algorithms presented in this thesis.

Chapter 4 describes in detail all the novel techniques introduced in this thesis for the processing of single channel acoustic data. These include a new speech/non-speech decoder which improves the previous version by being totally train-free and more adapted to the diarization process. Also, several algorithms for speaker clusters description and modeling, including algorithms for description of the number of clusters, model complexity selection, a new training

algorithm and two cluster purification algorithms.

Chapter 5 describes in detail the particular characteristics of using multiple channels in a meeting room and how the diarization processing can benefit from them. A filter&sum beamforming algorithm is selected for the task. The algorithm basic description and its implementation is described, explaining both well known and novel algorithms used for it. Also in this chapter a description of the use of the Time Delay of Arrival (TDOA) between channels as a parallel feature stream for the diarization module is pursued. Finally, a novel algorithm for the weight determination between TDOA and acoustic features is described.

Chapter 6 describes the experiments to show the appropriateness of all techniques. First, it describes the setup for running the experiments and then shows and explains the results for each one, comparing it to a baseline derived from the original broadcast news system prior to this thesis work or from intermediate (well established) points.

Chapter 7 describes the content and motivations behind the NIST Rich Transcription evaluations, which has been the tool used to assess the quality and to compare the proposed diarization system to other research systems and is the source of all datasets used in the experiments. A description of ICSI's submissions for 2005 and 2006 is explained in detail and results for those evaluations are given.

Finally, chapter 8 summarizes the major contributions and results obtained in this thesis and proposes some improvements and future work.



## Chapter 2

# State of the art

In this chapter the main techniques used over the recent years on the task towards speaker diarization (i.e. speaker segmentation and clustering) and on acoustic beamforming are reviewed. Initially, the features that have been found suitable for speaker diarization are explained. Then, a look at the algorithms and systems to deal in general with the task at hand are introduced. Finally some ground is set on techniques oriented towards performing speaker diarization in meetings, being this the main domain of application of this thesis.

Speaker diarization can be defined in terms of being a subtype of audio diarization, where the speech segments of the signal are broken into the different speakers (Reynolds and Torres-Carrasquillo 2004). It generally answers to the question "Who spoke when?" and it is sometimes referred to as speaker segmentation and clustering. In the domain of application of this thesis it is performed without any prior knowledge of the identity of the speakers in the recordings or how many are there. This, though, is not a requirement to call it speaker diarization as partial knowledge on the identities of some people in the recordings, the number of speakers or the structure of the audio (what follows what) might be available and used depending on the application at hand. None of these informations is provided in the RT evaluation campaigns organized by NIST (*NIST Spring Rich Transcription Evaluation in Meetings website*, <http://www.nist.gov/speech/tests/rt/rt2005/spring> 2006) which is the task used to evaluate all the algorithms presented in this thesis.

According to Reynolds and Torres-Carrasquillo (2004), there are 3 main domains of application for speaker diarization that have received special attention over the years:

- Broadcast news audio: Radio and TV programs with various kinds of programming, usually containing commercial breaks and music, over a single channel.
- Recorded meetings: meetings or lectures where multiple people interact in the same room or over the phone. Normally recordings are made with several microphones.

- Phone conversations: single channel recordings of phone conversations between two or more people. It is very much used in the speaker recognition campaigns but in disuse in diarization.

Furthermore, one could consider other particular domains, like air traffic communications, dialog in the car, and others.

As part of speaker diarization, speaker segmentation and speaker clustering belong to the pattern classification family, where one tries to find categorical (discrete) classes for continuous observations of speech and, by doing so, it finds the boundaries between them. Speech recognition is also a pattern classification problem. As such, they all need to work on a feature set that represents well the acoustic data and define a distance measure/method to assign each feature vector to a class.

In general, clustering data into classes is a well studied technique for statistical data analysis, with applications in many fields, including machine learning, data mining, pattern recognition, image analysis, bioinformatics and others.

When using clustering techniques for speaker or acoustic clustering one needs to previously define the segments that are going to be clustered, which might be of different sizes and characteristics (speech, non-speech, music, noises). In creating the segments using segmentation techniques one needs to be able to separate the speech stream into speakers and not words or phones. Any speech segment is populated with voiced and unvoiced phones, and short pauses between some phones or when punctuating. A speaker segmentation and clustering algorithm needs to define the properties of the speaker's data to be assigned to a single speaker and define techniques to assign such data into a single cluster. To do so one needs to use the appropriate acoustic models, their size and training algorithms so that they identify differences correctly in the acoustics at the speaker level.

The first section in this chapter takes a look at the features that have been proven useful for speaker based processing (like speaker diarization). Emphasis is given to alternatives to the traditional features, to focus on speaker characteristics that better discriminate and help identify the speakers present in a recording.

Following the features review, an overview of the main techniques that have been used in the area of speaker segmentation and speaker diarization is pursued. Speaker segmentation is a first step in many speaker diarization systems and therefore it is found useful to review what techniques have been mainly used in the past and to create a ground theory for the speaker diarization review. After explaining the main speaker diarization systems focus will be geared towards speaker diarization for meetings, which is the focus of implementation in this thesis.

In meetings one usually encounters several available microphones for processing, are all

located inside the meetings room in several locations around the speakers. Although most of these microphones are not defined to form a microphone array in theory, in practice it is found useful to use microphone array beamforming techniques in order to combine the microphones data into one “enhanced” channel and then process only this channel using the diarization system. This has the advantage that the speaker diarization system stays totally transparent of the particularities of each meeting room setting and processes only one channel in any case, improving in speed, versus any other solutions involving some sort of processing of all channels in parallel.

In the last section of this state of the art review the main techniques currently available in acoustic beamforming will be covered, which have been applied in the implemented system in order to take advantage of the multiplicity of available microphones. First, an overview of the techniques used to obtain an “enhanced” signal as an output from multiple input signals is covered, and then possible ways to estimate the delay between each of these channels is explored, necessary in order to align the acoustic data, used the majority of beamforming algorithms.

## 2.1 Acoustic Features for Speaker Diarization

Speaker diarization falls into the category of the speaker-based processing techniques. Features extracted from the acoustic signal are intended to convey information about the speakers in the conversations in order to enable the systems to separate them optimally.

Likewise speaker recognition and speech recognition systems, well used parametrization features in speaker diarization are Mel Frequency Cepstral Coefficients (MFCC), Linear frequency cepstral coefficients (LFCC), Perceptual Linear Predictive (PLP), Linear Predictive Coding (LPC) and others.

Although the aforementioned parametrization techniques yield a good performance in current speaker diarization and recognition systems, they are usually not focused on representing the information relevant to distinguishing between speakers and to isolate such information from other interfering sources (like non-stationary noises, background music and others). Nevertheless speaker recognition and diarization systems like the one presented in this thesis use MFCC parameters with a higher number of coefficients as it is known that the higher coefficients do incorporate speaker information.

In this section some research is pointed out that propose alternative parameters focusing on the speaker characteristics and/or particular conditions of the tasks that they are applied to, all within the speaker-based area, which can constitute an advantage if used alone or in conjunction with the most common parametrization techniques. Although the use of these parameters is still not general, these should constitute the tip of the iceberg of parameters exploiting speaker

information to come.

In Yamaguchi, Yamashita and Matsunaga (2005) it proposes a speaker segmentation system using energy, pitch frequency, peak-frequency centroid and peak-frequency bandwidth, and adds three new features: temporal feature stability of the power spectra, spectral shape and white noise similarities; all three related to the cross correlation of the power spectrum of the signal.

In order to avoid the influence of background noises and other non-speaker related events, in Pelecanos and Sridharan (2001) and more recently in Ouellet, Boulianne and Kenny (2005), feature warping techniques are proposed to change the shape of the p.d.f. of the features to a Gaussian shape prior to their modeling. They have been applied with success in Sinha, Tranter, Gales and Woodland (2005) and Zhu, Barras, Lamel and Gauvain (2006) for speaker diarization in broadcast news and meetings respectively.

In the area of speech activity detection (SAD) there have been also several features proposed in the latter years. In Kristjansson, Deligne and Olsen (2005) some well known features and other new ones are proposed, based on autocorrelation of the signal or on the spectrum characteristics.

In Nguyen (2003) a new theoretical framework for natural isometric frontend parameters based on differential geometry is presented and applied to speaker diarization, improving performance when used in combination to standard MFCC parameters.

In Moh, Nguyen and Junqua (2003), Tsai, Cheng and Wang (2004) and Tsai, Cheng, Chao and Wang (2005) speaker diarization systems are proposed by constructing a speaker space from the data and projecting the feature vectors in it prior to the clustering step. Similarly, Collet, Charlet and Bimbot (2005) proposes the technique of anchor modeling (introduced in Sturim, Reynolds, Singer and J.P.Campbell (2001)) where acoustic frames are projected into an anchor model space (previously defined from outside data) and performs speaker tracking with the resulting parameter vectors. They show that it improves robustness against outside interfering signals and they claim it to be domain independent.

When more than one microphone is collecting the recordings (for examples in meeting rooms) Pardo, Anguera and Wooters (2006a), Pardo, Anguera and Wooters (2006b), *ICSI Meeting Recorder Project: Channel skew in ICSI-recorded meetings* (2006), Lathoud and McCowan (2003) show that it is useful for speaker diarization the use of the time-delays between microphones.

Finally, in Chan, Lee, Zheng and hua Ouyang (2006) they propose the use of vocal source features for the task of speaker segmentation using a system based on Delacourt and Wellekens (2000). Also in Lu and Zhang (2002b) a real-time 2-step algorithm is proposed by doing a bayesian fusion of LSP, MFCC and pitch features.

## 2.2 Speaker Segmentation

Speaker segmentation has sometimes been referred to as speaker change detection and is closely related to acoustic change detection. For a given speech/audio stream, speaker segmentation/change detection systems find the times when there is a change of speaker in the audio. On a more general level, acoustic change detection aims at finding the times when there is a change in the acoustics in the recording, which includes speech/non-speech, music/speech and others. Acoustic change detection can detect boundaries within a speaker turn when the background conditions change.

Although erroneously, the term “speaker segmentation” has sometimes been used instead of speaker diarization for systems performing both a segmentation into different speaker segments and a clustering of such segments into homogeneous groups. As it will be pointed out later on, many systems obtain a speaker diarization output by means of first performing a speaker segmentation and then grouping the segments belonging to the same speaker. Other times this distinction is not so clear as segmentation and clustering are mixed together. In this thesis a system will be called to perform speaker segmentation when all frames, assigned to any particular speaker ID, are contiguous in time. Otherwise the system will be said to perform speaker segmentation and clustering (or equivalently speaker diarization).

In a very general level, two main types of speaker segmentation systems can be found in the bibliography. The first kind are systems that perform a single processing pass of the acoustic data, from where the change-points are obtained. A second broad class of systems are those that perform multiple passes, refining the decision of change-point detection on successive iterations. This second class of systems include two-pass algorithms where in a first pass many change-points are suggested (more than there actually are, therefore with a high false alarm error) and in a second pass such changes are reevaluated and some are discarded. Also part of the second broad class of systems are those that use an iterative processing of some sort to converge into an optimum speaker segmentation output. Many of the algorithms to find the change-points reviewed in this section (including all of the metric-based techniques) can either work alone or in a two-step system together with another technique.

On another level, a general classification of the methods available for speaker segmentation will be used in this section to describe the different algorithms available. In the bibliography (Ajmera (2004), Kemp, Schmidt, Westphal and Waibel (2000), Chen, Gales, Gopinath, Kanvesky and Olsen (2002), Shaobing Chen and Gopalakrishnan (1998), Perez-Freire and Garcia-Mateo (2004)) three groups are defined: metric-based, silence-based and model-based algorithms. In this thesis this classification will be augmented with a fourth group (called “others”) to amalgamate all other techniques that do not fit any of the three proposed classes. In the next section the metric-based techniques are reviewed in detail and in 2.2.2 the other three groups are treated.

### 2.2.1 Metric-Based Segmentation

Metric based segmentation is probably the most used technique up to date. It relies on the computation of a distance between two acoustic segments to determine whether they belong to the same speaker or to different speakers, and therefore whether there exists a speaker change point in the audio at the point being analyzed. The two acoustic segments are usually next to each other (in overlap or not) and the change-point considered is between them. Most of the distances used for acoustic change detection can also be applied to speaker clustering in order to compare the suitability that two speaker clusters belong to the same speaker.

Let's consider two audio segments  $(i,j)$  of parameterized acoustic vectors  $\mathcal{X}_i$  and  $\mathcal{X}_j$  of lengths  $N_i$  and  $N_j$  respectively, and with mean and variance values  $\mu_i, \sigma_i$  and  $\mu_j, \sigma_j$ . Each one of these segments is modeled using Gaussian processes  $M_i(\mu_i, \sigma_i)$  and  $M_j(\mu_j, \sigma_j)$ , which can be a single Gaussian or a Gaussian Mixture Model (GMM). On the other hand, let's consider the agglomerate of both segments into  $\mathcal{X}$ , with mean and variance  $\mu, \sigma$  and the corresponding Gaussian process  $M(\mu, \sigma)$ .

In general, there are two different kinds of distances that can be defined between any pair of such audio segments. The first kind compares the sufficient statistics from the two acoustic sets of data without considering any particular model applied to the data, which from now on will be called *statistics-based* distances. These are normally very quick to compute and give good performances if  $N_i$  and  $N_j$  are big enough to robustly compute the data statistics and the data being modeled can be well defined using a single mean and variance.

A second group of distances are based on the evaluation of the likelihood of the data according to models representing it. These distances are slower to compute (as models need to be trained and evaluated) but can achieve better results than the statistics-based as bigger models can be used to fit more complex data. These will be referred as *likelihood-based* techniques. The following are the metrics that have been found of interest used in the literature for either case:

- **Bayesian Information Criterion (BIC):** The BIC is probably the most extensively used segmentation and clustering metric due to its simplicity and effectiveness. It is a likelihood criterion penalized by the model complexity (amount of free parameters in the model) introduced by Schwarz (1971) and Schwarz (1978) as a model selection criterion. For a given acoustic segment  $X_i$ , the BIC value of a model  $M_i$  applied to it indicates how well the model fits the data, and is determined by:

$$BIC(\mathcal{M}_i) = \log \mathcal{L}(\mathcal{X}_i, \mathcal{M}_i) - \lambda \frac{1}{2} \#(\mathcal{M}_i) \log(N_i) \quad (2.1)$$

Being  $\log \mathcal{L}(\mathcal{X}_i, \mathcal{M}_i)$  the log-likelihood of the data given the considered model,  $\lambda$  is a free design parameter dependent on the data being modeled;  $N_i$  is the number of frames in the

considered segment and  $\#(\mathcal{M}_i)$  the number of free parameters to estimate in model  $\mathcal{M}_i$ . Such expression is an approximation of the Bayes Factor (BF) (Kass and Raftery (1995), Chickering and Heckerman (1997)) where the acoustic models are trained via ML methods and  $N_i$  is considered big.

In order to use BIC to evaluate whether a change point occurs between both segments it evaluates the hypothesis that  $\mathcal{X}$  better models the data versus the hypothesis that  $\mathcal{X}_i + \mathcal{X}_j$  does instead, like in the GLR, by computing:

$$\Delta BIC(i, j) = -R(i, j) + \lambda P \quad (2.2)$$

The term  $R(i)$  can be written for the case of models composed on a single Gaussian as:

$$R(i, j) = \frac{N}{2} \log |\Sigma_{\mathcal{X}}| - \frac{N_i}{2} \log |\Sigma_{\mathcal{X}_i}| - \frac{N_j}{2} \log |\Sigma_{\mathcal{X}_j}| \quad (2.3)$$

where  $P$  is the penalty term, which is a function of the number of free parameters in the model. For a full covariance matrix it is

$$P = \frac{1}{2}(p + \frac{1}{2}p(p + 1)) \log(N)$$

The penalty term accounts for the likelihood increase of bigger models versus smaller ones. For cases where GMM models with multiple Gaussian mixtures are used, eq. 2.2 is written as

$$\Delta BIC(\mathcal{M}_i) = \log \mathcal{L}(\mathcal{X}, \mathcal{M}) - (\log \mathcal{L}(\mathcal{X}_i, \mathcal{M}_i) + \log \mathcal{L}(\mathcal{X}_j, \mathcal{M}_j)) - \lambda \Delta \#(i, j) \log(N) \quad (2.4)$$

where  $\Delta \#(i, j)$  is the difference between the number of free parameters in the combined model versus the two individual models. For a mathematical proof on the equality of equations 2.3 and 2.4 please refer to the appendix section.

Although  $\Delta BIC(i, j)$  is the difference between two  $BIC(i)$  criterions in order to determine which model suits better the data, it is usual in the speaker diarization literature to refer to the difference as BIC criterion. For the task of speaker segmentation, the technique was first used by Chen and Gopalakrishnan (Shaobing Chen and Gopalakrishnan (1998), Chen and Gopalakrishnan (1998), Chen et al. (2002)) where a single full covariance Gaussian was used for each of the models, as in eq. 2.3.

Although not existent in the original formulation, the  $\lambda$  parameter was introduced to adjust the penalty term effect on the comparison, which constitutes a hidden threshold to the BIC difference. Such threshold needs to be tuned to the data and therefore its correct setting has been subject of constant study. Several people propose ways to automatically selecting  $\lambda$ ,

(Tritschler and Gopinath (1999), Delacourt and Wellekens (2000), Delacourt, Kryze and Wellekens (1999a), Mori and Nakagawa (2001), Lopez and Ellis (2000a), Vandecatseye, Martens et al. (2004)). In Ajmera, McCowan and Bourlard (2003) a GMM is used for each of the models ( $M$ ,  $M_i$  and  $M_j$ ) and by building the model  $M$  with the sum of models  $M_i$  and  $M_j$  complexities, it cancels out the penalty term avoiding the need to set any  $\lambda$  value. The result is equivalent to the GLR metric where the models have the complexity constraint imposed to them.

In the formulation of BIC by Schwarz (1978) the number of acoustic vectors available to train the model were supposed to be infinite for the approximation to converge. In real applications this becomes a problem when there is a big mismatch between the length of the two adjacent windows or clusters being compared. Some people have successfully applied slight modification to the original formula, either to the penalty term (Perez-Freire and Garcia-Mateo 2004) or to the overall value (Vandecatseye and Martens 2003) to reduce this effect.

Several implementations using BIC as a segmentation metric have been proposed. Initially Shaobing Chen and Gopalakrishnan (1998) proposed a multiple changing point detection algorithm in two passes, and later Tritschler and Gopinath (1999), Sivakumaran, Fortuna and Ariyaeinia (2001), sian Cheng and min Wang (2003), Lu and Zhang (2002a), Cettolo and Vescovi (2003) and Vescovi, Cettolo and Rizzi (2003) followed with one or two-pass algorithms. They all propose a system using a growing window with inner variable length analysis segments to iteratively find the changing points. In Tritschler and Gopinath (1999) it proposes some ways to make the algorithm faster and to focus on detecting very short speaker changes. In Sivakumaran et al. (2001), Cettolo and Vescovi (2003) and Vescovi et al. (2003) speedups are proposed in ways of computing the mean and variances of the models. In Roch and Cheng (2004) a MAP-adapted version of the models is presented, which allows for shorter speaker change points to be found. By using MAP, this work opposes to the way the models are described to be trained in the original formula (which defines an ML criterion).

Even with the efforts to speed up the processing of BIC, it is computationally more intensive than other statistics-based metrics when used to analyze the signal with high resolution, but its good performance has kept it as the algorithm of choice in many applications. This is why some people have proposed BIC as the second pass (refinement) of a 2-pass speaker segmentation system. As described earlier, an important step in this direction is taken with DISTBIC (Delacourt and Wellekens (2000), Delacourt et al. (1999a), Delacourt, Kryze and Wellekens (1999b)) where the GLR is used as a first pass. Also in this direction are Zhou and Hansen (2000), Kim, Ertelt and Sikora (2005) and Tranter and Reynolds (2004), proposing the to use Hotelling's  $T^2$  distance, and Lu and Zhang (2002a) using KL2 (Kullback-Leibler) distance. In Vandecatseye et al. (2004) a normalized GLR

(called NLLR) is used as a first pass and a normalized BIC is used in the refinement step.

Some research has been done to combine alternative sources of information to help the BIC in finding the optimum change point. This is the case in Perez-Freire and Garcia-Mateo (2004) where image shot boundaries are used.

In sian Cheng and min Wang (2004) a two-pass algorithm using BIC in both passes is proposed. This is peculiar in that instead of producing a first step with high FA and a second step that merges some of the change-points, the first step tries to minimize the FA and the second step finds the rest of unseen speaker changes.

- **Generalize Likelihood Ratio (GLR):** The GLR (first proposed for change detection by Willsky and Jones (1976) and Appel and Brandt (1982)) is a likelihood-based metric that proposes a ratio between two hypotheses: on one hand,  $H_0$  considers that both segments are uttered by the same speaker, therefore  $\mathcal{X} = \mathcal{X}_i \cup \mathcal{X}_j \sim M(\mu, \sigma)$  represents better the data. On the other hand,  $H_1$  considers that each segment has been uttered by a different speaker, therefore  $\mathcal{X}_i \sim M_i(\mu_i, \sigma_i)$  and  $\mathcal{X}_j \sim M_j(\mu_j, \sigma_j)$  together suit better the data. The ratio test is computed as a likelihood ratio between the two hypotheses as

$$GLR(i, j) = \frac{H_0}{H_1} = \frac{\mathcal{L}(\mathcal{X}, M(\mu, \sigma))}{\mathcal{L}(\mathcal{X}_i, M_i(\mu_i, \sigma_i))\mathcal{L}(\mathcal{X}_j, M_j(\mu_j, \sigma_j))} \quad (2.5)$$

and determining the distance as  $D(i, j) = -\log(GLR(i, j))$  which upon using an appropriate threshold one can decide whether both segments belong to the same speaker or otherwise. The GLR differs from a similar metric called the standard likelihood ratio test (LLR) in that the p.d.f.'s for the GLR are unknown and must be estimated directly from the data within each considered segment, whereas in the LLR the models are considered to be known a priori. In speaker segmentation the GLR is usually used with two adjacent segments of the same size which are scrolled through the signal, and the threshold is either pre-fixed or it dynamically adapts.

In Bonastre, Delacourt, Fredouille, Merlin and Wellekens (2000) the GLR is used to segment the signal into speaker turns in a single step processing for speaker tracking. The threshold is set so that miss errors are minimized (at the cost of higher false alarms), as each segment is then independently considered as a potential speaker in the tracking algorithm.

in Gangadharaiah, Narayanaswamy and Balakrishnan (2004) a two-speaker segmentation is performed in two steps. On the first step GLR is used to over-segment the data. On a second step, “seed” segments are selected for both speakers and the rest are assigned to either speaker with a Viterbi decoding / ML approach without modifying the defined change-points.

On the same two-speaker detection task, in Adami, Kajarekar and Hermansky (2002) the first second of speech is considered to be from the first speaker and the second speaker is found determining the change-points via GLR. A second step assigns segments of speech to either speaker by comparing the GLR score of each of the two speakers computed across the recording and selecting the regions where either one is higher.

On the task of change detection for transcription and indexing in Liu and Kubala (1999) a penalized GLR is used as a second step, to accept/reject change-points previously found using a pre-trained phone-based decoder (where the ASR phone-set has been reduced into phone clusters). The penalty applied to the GLR is proportional to the amount of training data available in the two segments as

$$GLR'(i, j) = \frac{GLR(i, j)}{(N_1 + N_2)^\theta} \quad (2.6)$$

where  $\theta$  is determined empirically. On the same tone, Metze, Fugen, Pan, Schultz and Yu (2004) uses the GLR for a segmentation step in a transcription system for meetings.

Probably the most representative algorithm of the use of GLR for speaker segmentation is DISTBIC (Delacourt and Wellekens (1999), Delacourt et al. (1999a), Delacourt et al. (1999b), Delacourt and Wellekens (2000)) where GLR is proposed as the first step of a two-step segmentation process (using BIC as the second metric). Instead of using the GLR distance by itself, a low pass filtering is applied to it in order to reduce ripples in the computed distance function (which would generate false maxima/minima points) and then the difference between each local maxima and adjacent minima is used to assert the change-points.

- **Gish distance:** It is a likelihood-based metric obtained as a variation to the GLR presented in Gish, Siu and Rohlicek (1991) and Gish and Schmidt (1994). To derive it, the GLR function is split into two parts ( $\lambda_{cov}$  and  $\lambda_{mean}$ ) and the background dependent part is ignored, leading to the equation

$$D_{Gish}(i, j) = -\frac{N}{2} \log\left(\frac{|S_i|^\alpha |S_j|^{(1-\alpha)}}{|W|}\right) \quad (2.7)$$

where  $S_i$  and  $S_j$  represent the sample covariance matrices for each segment,  $\alpha = \frac{N_1}{N_1+N_2}$  and  $W$  is their sample weighted average  $W = \frac{N_1}{N_1+N_2}S_1 + \frac{N_2}{N_1+N_2}S_2$ .

In Kemp et al. (2000) the Gish distance is compared to other techniques for speaker segmentation.

- **Kullback-Leibler distance (KL or KL2):** The KL and KL2 distances (Siegler, Jain, Raj and Stern (1997), Hung, Wang and Lee (2000)) are well used due to their fast computation

and acceptable results. Given two random distributions  $X, Y$ , the K-L distance (also called divergence) is defined as

$$KL(X; Y) = E_X(\log \frac{P_X}{P_Y}) \quad (2.8)$$

Where  $E_X$  is the expected value with respect to the PDF of  $X$ . When the two distributions are taken to be Gaussian, one can obtain a close form solution to such expression (Campbell 1997) as

$$KL(X, Y) = \frac{1}{2}tr[(C_X - C_Y)(C_Y^{-1} - C_X^{-1})] + \frac{1}{2}tr[(C_Y^{-1} - C_X^{-1})(\mu_X - \mu_Y)(\mu_X - \mu_Y)^T] \quad (2.9)$$

For GMM models there is no close form solution and the KL distance needs to be computed using sample theory or one needs to use approximations as shown below. The KL2 distance can be obtained by symmetrizing the KL in the following way:

$$KL2(X; Y) = KL(X; Y) + KL(Y; X) \quad (2.10)$$

As previously, if both distributions  $X$  and  $Y$  are considered to be Gaussian one can obtain a closed form solution for the KL2 distance in function of their covariance matrices and means.

Given any two acoustic segments  $\mathcal{X}_1$  and  $\mathcal{X}_2$  can be considered as  $X$  and  $Y$  and therefore obtain the distance between them using these distances.

In Delacourt and Wellekens (2000) the KL2 distance is considered as a first of two steps for speaker change detection. In Zochova and Radova (2005) KL2 is used again in an improved version of the previous algorithm.

In Hung et al. (2000) the MFCC acoustic vectors are initially processed via a PCA dimensionality reduction for each of the contiguous scrolling segments (either two independent PCA or one applied to both segments) and then Mahalanobis, KL and Bhattacharyya distances are used to determine if there is a change point.

- **Divergence Shape Distance(DSD)**: In a very similar fashion as how the Gish distance is defined in Gish et al. (1991), the DSD is derived from the KL distance of two classes with n-variate normal pdfs by eliminating the part affected by the mean, as it is easily biased by environment conditions. Therefore, it corresponds to the expression

$$D(i, j) = \frac{1}{2}tr[(C_i - C_j)(C_j^{-1} - C_i^{-1})] \quad (2.11)$$

In Kim et al. (2005) it is used in a single-step algorithm and its results are compared to BIC.

The DSD is also used in Lu and Zhang (2002a) as a first step of a two step segmentation system, using BIC on the refinement step. In Lu and Zhang (2002b) some speed-ups are proposed to make previous the system real-time.

The same authors present in Wu, Lu, Chen and Zhang (2003b), Wu, Lu, Chen and Zhang (2003a) and Wu, Lu, Chen and Zhang (2003c) an improvement to the algorithm using DSD and a Universal Background Model (UBM) trained from only the data in the processed show. Evaluation of the likelihood of the data according to the UBM is used to categorize the features in each analysis segment and only the good quality speech frames from each one are compared to each other. They use an adaptive threshold (adapted from previous values) to determine change points.

Such work is inspired by Beigi and Maes (1998) where each segment is clustered in three classes via a k-means and a global distance is computed by combining the distances between classes. There is no word in this work regarding to which particular distance is used between the classes.

- **Cross-BIC (XBIC):** This distance was introduced by the author in Anguera and Hernando (2004b) and Anguera (2005), which derives a distance between two adjacent segments by cross-likelihood evaluation, inspired on the BIC distance by comparison to a distance between HMM presented in Juang and Rabiner (1985):

$$XBIC(\mathcal{X}_1; \mathcal{X}_2) = \mathcal{L}(\mathcal{X}_1, \mathcal{M}_2(\mu_2, \sigma_2)) + \mathcal{L}(\mathcal{X}_2, \mathcal{M}_1(\mu_1, \sigma_1)) \quad (2.12)$$

In Malegaonkar, Ariyaeeinia, Sivakumaran and Fortuna (2006) they propose a similar metric and study different likelihood normalization techniques to make the metric more robust, achieving better results than BIC for speaker segmentation.

- **Other distances:** There are many other metrics that are able to define a distance between two sets of acoustic features or two models. Some of them have been applied to the speaker segmentation task.

In Omar, Chaudhari and Ramaswamy (2005) the CuSum distance (Basseville and Nikiforov 1993), the Kolmogorov-Smirnov test (Deshayes and Picard 1986) and BIC are first used independently to find putative change points and then fused at likelihood level to assert those changes.

In (Hung et al. 2000) the Malalanobis and Bhattacharyya distances (Campbell 1997) are used in comparison to the KL distance for change detection.

In Kemp et al. (2000) the entropy loss (Lee 1998) of coding the data in two segments instead of only one is proposed in comparison to the Gish and KL distances.

In Mori and Nakagawa (2001) applies VQ (Vector quantization) techniques to create a codebook from one of two adjacent segments and applies a VQ distortion measure (Nakagawa and Suzuki 1993) to compare its similarity with the other segment. Results are compared to GLR and BIC techniques.

In Zhou and Hansen (2000) and Tranter and Reynolds (2004) Hotelling's  $T^2$  distance is proposed, being it a multivariate analog of the t-distribution. It is applied for the first of a two-step segmentation algorithm. It finds the distance between two segments, modeling each one with a single Gaussian where both covariance matrices are set to be the same.

All of these metric-based techniques compute a function whose maxima/minima need to be compared with a threshold in order to determine the suitability of every change point. In many cases such threshold is defined empirically given a development set, according to a desired performance. Such proceeding leads to a threshold which is normally dependent on the data being processed and that needs to be redefined every time data of a different nature needs to be processed. This problem has been studied within the speaker identification community in order to classify speakers in an open set speaker identification task (see for example Campbell (1997)). In the area of speaker segmentation and clustering some publications propose automatic ways to define appropriate thresholds, for example:

- In Lu, Zhang and Jiang (2002), Lu and Zhang (2002b) and Lu and Zhang (2002a) an adaptive threshold is made dependent on the  $P$  previous as

$$Th_i = \alpha \frac{1}{P} \sum_{p=0}^P D(i-p-1, i-p) \quad (2.13)$$

where  $\alpha$  is an amplification coefficient (usually set close to 1).

The same adaptive threshold is used in Wu et al. (2003b), Wu et al. (2003a) and Wu et al. (2003c) to evaluate the difference between the local maxima and the neighboring minima distance points.

- In Rougui, Rziza, Aboutajdine, Gelgon and Martinez (2006) a dynamic threshold is defined in comparing speaker clusters (rather than speaker segments) where a population of clusters is used to decide on the threshold value. It is defined as

$$Th = \max(hist(d(M_i, M_j), \forall i \neq j)) \quad (2.14)$$

where  $hist$  denotes the histogram and  $d()$  is the distance between two models, which in that work is defined as a modified KL distance to compare two GMM models.

### 2.2.2 Non Metric-Based Segmentation

In this section the other three classes of speaker segmentation are reviewed, namely silence/decoder-based, model-based and other segmentation techniques.

#### Silence and Decoder-Based Segmentation

These techniques detect speaker changes hypothesizing that most changes between speakers will be through a silence segment. These have been traditionally implemented towards using the segments for speech recognition, as it is very important to obtain clean speaker changes without cutting any words in half. Systems falling into this category are energy-based and decoder-based systems.

The energy-based systems use an energy detector to find the points where it is most probable to exist a speaker change. The detector normally obtains a curve with minimum/maximum points in potential silences. A threshold is usually used to determine them (Kemp et al. (2000), Wactlar, Hauptmann and Witbrock (1996), Nishida and Kawahara (2003)). In Siu, Yu and Gish (1992) the MAD (Mean absolute deviation statistic), which measures the variability in energy within segments, is used instead in order to find the silence points.

In contrast, decoder-guided segmenters run a full recognition system and obtain the change points from the detected silence locations (Kubala, Jin, Matsoukas, Gnuyen, Schwartz and Machoul (1997), Woodland, Gales, Pye and Young (1997), Lopez and Ellis (2000b), Liu and Kubala (1999), Wegmann, Scattone, Carp, Gillick, Roth and Yamron (1998)) they normally constrain the minimum duration of the silence segments to reduce false alarms. Some of these systems use extra information from the decoder, such as gender labels (Tranter and Reynolds 2004) or wide/narrow band plus music detectors (Hain, Johnson, Turek, Woodland and Young 1998). The output has normally been used as an input to recognition systems, but not for indexing or Diarization as there is not a clear relationship between the existence of a silence in a recording and a change of speaker. In such systems they sometimes take these points as hypothetic speaker change points, and then using other techniques define which of them actually mark a change of speaker and which do not.

#### Model-Based Segmentation

Initial models (for example GMM's) are created for a closed set of acoustic classes (telephone-wideband, male-female, music-speech-silence and combinations of them) by using training data. The audio stream is then classified by ML (Maximum Likelihood) selection using these models (Gauvain, Lamel and Adda (1998), Kemp et al. (2000), Bakis, Chen, Gopalakrishnan and Gopinath (1997), Sankar, Weng, Stolcke and Grande (1998), Kubala et al. (1997)). The bound-

aries between models become the segmentation change points. One could also consider the decoder-guided systems to be model-based, as they model each phoneme and silence, but here they try to distinguish among broader models, instead of models derived from speech recognition and trained for individual phones.

This segmentation method resembles very closely the speaker clustering techniques where the identity of the different speakers (in this case acoustic classes) is known a priori and an ML segmentation is found. Both areas have a robustness problem given that they require initial data to train the models. As will be shown in the speaker clustering section, in the later years there has been research done on the topic of blind speaker clustering, where no initial information of the clusters is known. There is some of this research that applies these techniques to speaker segmentation, in particular some clustering systems make use of an ML decoding of evolutive models that look for the optimum acoustic change points and speaker models at the same time.

In Ajmera, Boulard and Lapidot (2002) and Ajmera and Wooters (2003) the iterative decoding is done bottom-up (starting with a high number of speaker changes as product of a first step processing and then eliminating them until obtaining the optimum amount) and in Meignier, Bonastre and Igournet (2001) and Anguera and Hernando (2004a) it is done top-down (starting with one segment and adding extra segments until the desired amount is reached).

In Meignier, Moraru, Fredouille, Besacier and Bonastre (2004) they analyze the use of evolutive systems where pretrained models are also used modeling background conditions, showing that in general the more prior information that can be given to the system the better performance it achieves.

All of these systems use Gaussian Mixture Models (GMM) to model the different classes and an ML/Viterbi decoding approach to obtain the optimum change points. In Lu, Li and Zhang (2001) SVMs (Support Vector Machines) are used as a classifier instead of GMM models and the ML decoding, training them using pre-labelled data.

### **Segmentation Using Other Techniques**

There are some speaker segmentation techniques proposed in the literature that are not a clear fit to any of the previous categories. These are therefore mentioned here.

In Vescovi et al. (2003) and Zdansky and Nouza (2005) dynamic programming is proposed to find the speaker change points. In Zdansky and Nouza (2005) BIC is used as marginal likelihood, solving the system via ML where all possible number of change points is considered. In Vescovi et al. (2003) they also use BIC and explore possible computation reduction techniques.

In Pwint and Sattar (2005) a genetic algorithm is proposed where the number of segments is estimated via the Walsh basis functions and the location of change points is found using a

multi-population genetic procedure.

In Lathoud, McCowan and Odobez (2004) segmentation is based on the location estimation of the speakers by using a multiple-microphone setting. The difference between two locations is used as a feature and tracking techniques are employed to estimate the change points of possibly moving speakers. Further work on using location cues for clustering will be presented in the next section.

## 2.3 Speaker Diarization

In some occasions the use of the term speaker diarization is confused with speaker clustering. One must refer as speaker clustering the techniques and algorithms that agglutinate together all segments that belong to the same speaker. This does not entail whether such segments come from the same acoustic file or different ones. It also does not say anything about how acoustically homogeneous segments within a single file are obtained. The term speaker diarization refers to the systems that perform a speaker segmentation of the input signal and then a speaker clustering of the created segments into homogeneous groups (or some hybrid mechanism doing both at the same time), all within the same file or input stream.

In the literature one can normally find two main applications for speaker diarization. On one hand, Automatic Speech Recognition (ASR) systems make use of the speaker homogeneous clusters to adapt the acoustic models to be speaker dependent and therefore increase recognition performance. On the other hand, speaker indexing and rich transcription systems use the speaker diarization output as one of (possibly) many information pieces extracted from a recording, which allow its automatic indexation and other further processing areas.

This section reviews the main systems present in the literature for both applications. It mainly focuses on systems that propose solutions to a blind speaker diarization problem, where no information is known a priori about the number of people or their identities. On one hand, it is crucial for systems oriented towards rich transcription of the data to accurately estimate the number of speakers present, as error measures penalize any incorrectly assigned speaker segment. On the other hand, in ASR systems it becomes more important to have sufficient data to accurately adapt the resulting speaker models, therefore several speakers with similar acoustic characteristics are preferably grouped together.

At a high level point of view one can differentiate between online and offline systems. The systems that process the data offline have access to all the recording before they start processing it. These are the most common in the bibliography and they are the main focus of attention of this review. The online systems only have access to the data that has been recorded up to that point. They might allow a latency in output to allow for a certain amount of data to become

available for processing, but in any case no information on the complete recording is available. Such systems usually start with one single speaker (whoever starts talking at the beginning of the recording) and iteratively increase the number of speaker as they intervene. The following are some representative systems used for online processing:

In Mori and Nakagawa (2001) a clustering algorithm based on the Vector Quantization (VQ) distortion measure (Nakagawa and Suzuki 1993) is proposed. It starts processing with one speaker in the code-book and incrementally adds new speakers whose VQ distortion exceeds a threshold in the current code-book.

In Rougui et al. (2006) a GMM based system is proposed, using a modified KL distance between models. Change points are detected as the speech becomes available and data is assigned to either speaker present in the database or a new speaker is created, according to a dynamic threshold. Emphasis is put into fast classification of the speech segments into speakers by using a decision tree structure for speaker models.

All systems presented below are based on offline processing, although some of the techniques presented could potentially be used also in an online implementation. These systems can be classified in two main groups, on one hand the hierarchical clustering techniques reach the optimum diarization by iterative processing of different number of possible clusters obtained by merging or splitting existing clusters. On the other hand, other clustering techniques first estimate the number of clusters and obtain a diarization output without deriving the clusters from bigger/smaller ones.

### 2.3.1 Hierarchical Clustering Techniques

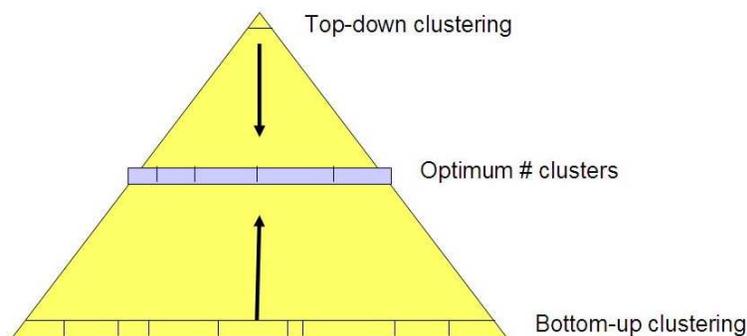


Figure 2.1: *Graphic interpretation of the most common clustering techniques*

Most of the reviewed offline clustering algorithms use hierarchical schemes, where speech segments or clusters are iteratively split or merged until the optimum number of speakers is reached. In figure 2.1 a pedantic abstraction of the two mostly used techniques in speaker clustering is shown. Bottom-up clustering systems are those which start with a big number of

segments/clusters and via merging techniques converge to the optimum amount of clusters. On the other hand, top-down systems usually start with one or very few clusters and work its way up (in the number of clusters, down in the figure) via splitting procedures to obtain the optimum amount. In the design of either system, two items need to be defined:

1. A distance between clusters/segments to determine acoustic similarity. Instead of defining an individual value pair, usually a distance matrix is described, which is created with the distance from any possible pair. In many cases the distance metrics used for speaker clustering resemble those used in speaker segmentation computed using the metrics presented in section 2.2.
2. A stopping criterion to stop the iterative merging/splitting at the optimum number of clusters (which might be different depending on the application).

Classified by the type of clustering, the following are the most representative techniques described in the literature:

### Bottom-up Clustering Techniques

This is by far the mostly used approach for speaker clustering as it welcomes the use of the speaker segmentation techniques to define a clustering starting point. It is also referred as agglomerative clustering and has been used for many years in pattern classification (see for example Duda and Hart (1973)). Normally a matrix distance between all current clusters (distance of any with any) is computed and the closest pair is merged iteratively until the stopping criterion is met.

One of the earliest research done in speaker clustering for speech recognition was proposed in Jin, Kubala and Schwartz (1997), using the Gish distance (Gish et al. 1991) as distance matrix, with a weight to favor neighbors merging. As stopping criterion, the minimization of a penalized version (to avoid over-merging) of the within-cluster dispersion matrix is proposed as

$$W_{Jin} = \left| \sum_{k=1}^K N_k \Sigma_k \right| \sqrt{k} \quad (2.15)$$

where  $K$  is the number of clusters considered,  $\Sigma_k$  is the covariance matrix of cluster  $k$ , with  $N_k$  acoustic segments and  $|\cdot|$  indicating the determinant.

Around the same time, in Siegler et al. (1997) the KL2 divergence distance was used as a distance metric and a stopping criterion was determined with a merging threshold. It shows that the KL2 distance works better than the Mahalanobis distance for speaker clustering. Also in Zhou

and Hansen (2000) the KL2 metric is used as a cluster distance metric. In this work they first split the speech segments into male/female and perform clustering on each one independently; this reduces computation (the number of cluster-pair combinations is smaller) and gives them better results.

In general, the use of statistics-based distance metrics (not requiring any models to be trained) is limited in speaker clustering as they implicitly define distances between single mean and covariance matrices from each set, which in speaker clustering falls short many times in modeling the amount of data available from one speaker. Some people have adapted these distances and obtained multi-Gaussian equivalents.

In Rougui et al. (2006) they propose a distance between two GMM models based on the KL distance. Given two models  $M_1$  and  $M_2$ , with  $K_1$  and  $K_2$  Gaussian mixtures each, and Gaussian weights  $W_1(i), i = 1 \dots K_1$  and  $W_2(j), j = 1 \dots K_2$ , the distance from  $M_1$  to  $M_2$  is

$$d(M_1, M_2) = \sum_{i=1}^{K_1} W_1(i) \min_{j=1}^{K_2} KL(\mathcal{N}_1(i), \mathcal{N}_2(j)) \quad (2.16)$$

where  $\mathcal{N}(i)$  is one of the Gaussians from the model.

In Beigi, Maes and Sorensen (1998) a distance between two GMM models is proposed by using the distances between the individual Gaussian mixtures. A distance matrix of  $d(i, j), \forall i, j$  between all possible Gaussian pairs in the two models is processed (distances proposed are the Euclidean, Mahalanobis and KL) and then the weighted minima for each row and column is used to compute the final distance.

In Ben, Betser, Bimbot and Gravier (2004) and Moraru, Ben and Gravier (2005) cluster models are obtained via MAP adaptation from a GMM trained on the whole show. A novel distance between GMM models is derived from the LK2 distance for the particular case where only means are adapted (and therefore weights and variances are identical in both models). Such distance is defined as

$$D(M_1, M_2) = \sqrt{\sum_{m=1}^M \sum_{d=1}^D W_m \frac{(\mu_1(m, d) - \mu_2(m, d))^2}{\sigma_{m,d}^2}} \quad (2.17)$$

where  $\mu_1(m, d)$  and  $\mu_2(m, d)$  are the mean  $d^{th}$  components for the mean vector for Gaussian  $m$ ,  $\sigma_{m,d}^2$  is the  $d^{th}$  variance component for Gaussian  $m$  and  $M, D$  are the number of mixtures and dimension of the GMM models respectively.

In Ben et al. (2004) a threshold is applied to such distance to serve as stopping criterion, while in Moraru et al. (2005) the BIC for the global system is used instead.

Leaving behind the statistics-based methods, in Gauvain et al. (1998) and Barras, Zhu, Meignier and Gauvain (2004) a GLR metric with two penalty terms is proposed, penalizing for large number of segments and clusters in the model, with tuning parameters. Iterative Viterbi decoding and merging iterations find the optimum clustering, which is stopped using the same metric.

Solomonov, Mielke, Schmidt and Gish (1998) also uses GLR and compares it to KL2 as distance matrices and iteratively merges clusters until it maximizes the estimated cluster purity, defined as the average over all segments and all clusters of the ratio of segments belonging to cluster  $i$  among the  $n$  closest segments to segment  $k$  (which belongs to  $i$ ). The same stopping criterion is used in Tsai et al. (2004), where several methods are presented to create a different reference space for the acoustic vectors that better represents similarities between speakers. The reference space defines a speaker space to which feature vectors are projected, and the cosine measure is used as a distance matrix. It is claimed that such projections are more representative of the speakers.

Other research is done using GLR as distance metric, including Siu et al. (1992) for pilot-controller clustering and Jin, Laskowski, Schultz and Waibel (2004) for meetings diarization (using BIC as stopping criterion).

The most commonly used distance and stopping criteria is again BIC, which was initially proposed for clustering in Shaobing Chen and Gopalakrishnan (1998) and Chen and Gopalakrishnan (1998). The pair-wise distance matrix is computed for each iteration and the pair with biggest  $\Delta\text{BIC}$  value is merged. The process finishes when all pairs have a  $\Delta\text{BIC} < 0$ . In some later research (Chen et al. (2002), Tritschler and Gopinath (1999), Tranter and Reynolds (2004), Cettolo and Vescovi (2003) for Italian language and Meinedo and Neto (2003) for Portuguese language) propose modifications to the penalty term and differences in the segmentation setup.

In Sankar, Beaufays and Digalakis (1995) and Heck and Sankar (1997) the symmetric relative entropy distance (Juang and Rabiner 1985) is used for speaker clustering towards speaker adaptation in ASR. This distance is similar to Anguera (2005) and equivalent to Malegaonkar et al. (2006), both used for speaker segmentation. It is defined as

$$D(M_1, M_2) = \frac{1}{2}[D_{\lambda_1, \lambda_2} + D_{\lambda_2, \lambda_1}] \quad (2.18)$$

where  $D_{\lambda_i, \lambda_j}$  is defined as

$$D_{\lambda_i, \lambda_j} = \log p(\mathcal{X}_i | M_i) - \log p(\mathcal{X}_i | M_j) \quad (2.19)$$

An empirically set threshold on the distance is used as a stopping criterion. Later on, the same

authors propose in Sankar et al. (1998) a clustering based on a single GMM model trained on all the show and the weights being adapted on each cluster. The distance used then is a weighted by counts entropy change due to merging two clusters (Digalakis, Monaco and Murveit 1996).

In Barras et al. (2004), Zhu, Barras, Meignier and Gauvain (2005), Zhu et al. (2006) and later Sinha et al. (2005) propose a diarization system making use of speaker identification techniques in the area of speaker modeling. A clustering system initially proposed in Gauvain et al. (1998) is used to determine an initial segmentation in Barras et al. (2004), Zhu et al. (2005) and Zhu et al. (2006), while a standard speaker change detection algorithm is used in Sinha et al. (2005). The systems then use standard agglomerative clustering via BIC, with a  $\lambda$  penalty value set to obtain more clusters than optimum (under-cluster the data). On the speaker diarization part, it first classifies each cluster for gender and bandwidth (in broadcast news) and uses a Universal Background Model (UBM) and MAP adaptation to derive speaker models from each cluster. In most cases a local feature warping normalization (Pelecanos and Sridharan 2001) is applied to the features to reduce non-stationary effects of the acoustic environment. The speaker models are then compared using a metric between clusters called cross likelihood distance (Reynolds, Singer, Carlson, O'Leary, McLaughlin and Zixxman 1998), and defined as

$$D(X_1, X_2) = \frac{1}{N_1} \log \frac{p(X_1|M_{2-UBM})}{p(X_1|UBM)} + \frac{1}{N_2} \log \frac{p(X_2|M_{1-UBM})}{p(X_2|UBM)} \quad (2.20)$$

where  $M_{i-UBM}$  indicates that the model has been MAP adapted from the UBM model. An empirically set threshold stops the iterative merging process.

The same cross-likelihood metric is used in Nishida and Kawahara (2003) to compare two clusters. In this paper emphasis is given to the selection of the appropriate model when training data is very small. It proposes a vector quantization (VQ) based method to model small segments, by defining a model called common variance GMM (CVGMM) where Gaussian weights are set uniform and variance is tied among Gaussians and set to the variance of all models. For each cluster BIC is used to select either GMM or CVGMM as the model to be used.

Some other people integrate the segmentation with the clustering by using a model-based segmentation/clustering scheme. This is the case in Ajmera et al. (2002), Ajmera and Wooters (2003) and Wooters, Fung, Peskin and Anguera (2004) where an initial segmentation is used to train speaker models that iteratively decode and retrain on the acoustic data. A threshold-free BIC metric (Ajmera et al. 2003) is used to merge the closest clusters at each iteration and as stopping criterion.

In Wilcox, Chen, Kimber and Balasubramanian (1994) a penalized GLR is proposed within an traditional agglomerative clustering approach. The penalty factor favors merging clusters which are close in time. To model the clusters, a general GMM is built from all the data in

the recording and only the weights are adapted to each cluster as in Sankar et al. (1998). A refinement stage composed of iterative Viterbi decoding and EM training follows the clustering, to redefine segment boundaries, until likelihood converges.

In Moh et al. (2003) a novel approach to speaker clustering is proposed using speaker triangulation to cluster the speakers. Given a set of clusters  $C_k, k = 1 \dots K$  and the group of non-overlapped acoustic segments  $X_s, j = s \dots S$  which populate the different subsets/clusters. The first step generates the coordinates vector of each cluster according to each segment (modeled with a full covariance Gaussian model) by computing the likelihood of each cluster to each segment. The similarity between two clusters is then defined as the cross correlation between such vectors as

$$C(k, j) = \sum_s p(C_k | X_s) p(C_j | X_s) \quad (2.21)$$

merging those clusters with higher similarity. This can also be considered as a projection of the acoustic data into a speaker space prior to the distance computation.

### Top-down Clustering Techniques

In the current literature there are fewer systems that start from one cluster and iteratively split until the stopping criterion is met than the previously presented systems, doing otherwise.

In Johnson and Woodland (1998) a top-down clustering method is proposed for speaker clustering towards ASR, and in Johnson (1999) and Tranter and Reynolds (2004) it is applied to speaker diarization. The algorithm splits the data iteratively into four sub-clusters and allows for merging clusters that are very similar to each other. In Johnson and Woodland (1998) it proposes two different implementations of the algorithm. On one hand it proposes an MLLR likelihood optimization technique to obtain resulting clusters well adapted to the ASR MLLR adaptation step. On the other hand it proposes the Arithmetic Harmonic Sphericity (AHS) metric (Bimbot and Mathan 1993) to assign speech segments to the created sub-clusters at each stage, and uses a minimum occupancy stopping criterion. The AHS is defined for single Gaussian models as

$$d(X_1, X_2) = \log[tr(\Sigma_{X_2} \Sigma_{X_1}^{-1}) \cdot tr(\Sigma_{X_1} \Sigma_{X_2}^{-1})] - 2 \log(D) \quad (2.22)$$

where  $D$  is the dimensionality of the data and  $tr$  is the trace function.

In Johnson (1999) and Tranter and Reynolds (2004) the AHS-based algorithm is used for speaker diarization and the stopping criterion is changed to be a cost-based function depending on several criteria.

In Meignier et al. (2001) and Anguera and Hernando (2004a) an initial cluster is trained with all the acoustic data available. Iterative decoding/MAP adaptation of new models is performed where new clusters are split using a likelihood metric averaged over a window. The variation of the overall likelihood of the data given all models is used as a stopping criterion. In Anguera and Hernando (2004a) a similar approach is followed and a repository model is further introduced to improve the purity of the created clusters.

### Combination of Clustering Methods

While bottom-up clustering is much more popular than top-down clustering, it is not clear which one can achieve better results and in which conditions. On the topic of broadcast news transcription, in Hain et al. (1998) both techniques were compared. On one hand, bottom-up clustering uses a divergence-like distance measure and a minimum cluster feature count as stopping criterion. On the other hand, top-down clustering uses the arithmetic harmonic sphericity distance and also the cluster count as stopping criterion.

Given that both Top-down and bottom-up techniques could eventually complement each other, some people have proposed systems that can combine multiple systems and obtain an improved speaker diarization.

In Tranter (2005) a cluster voting algorithm is presented to allow diarization output improvement by merging two different speaker diarization systems. Tests are performed using two top-down and two bottom-up systems.

In Moraru, Meignier, Besacier, Bonastre and Magrin-Chagnolleau (2004) and Fredouille, Moraru, Meignier, Besacier and Bonastre (2004) two different combination approaches are presented to combine top-down and bottom-up outputs and are applied to broadcast news and meetings processing. A first technique, called *hybridization*, proposes one system as initialization to the second system. The second technique is called *Fusion* and proposes a matching of common resulting segments followed by a resegmentation of the data to assign the non-common segments.

### 2.3.2 Other Clustering Techniques

There are some papers in the literature that do not fit into an hierarchical clustering context. The systems reviewed here all define an algorithm or metric to determine the optimum number of speakers and a method for finding the optimum speaker clustering given a that number.

In Tsai and Wang (2006) a genetic algorithm is proposed to obtain an optimum speaker clustering that optimizes the overall model likelihood by initial random cluster assignment and iterative evaluation of the likelihood and mutation. In order to select the optimum amount of

speakers they use BIC computed on the resulting models.

A relatively new learning technique called Variational Bayesian learning (VB) or Ensemble learning (Attias (2000), MacKay (1997)) is used in Valente and Wellekens (2004), Valente and Wellekens (2005) and Valente (2006) for speaker clustering. the VB training has the capacity of model parameter learning and model complexity selection, all in one algorithm. The models trained with this technique adapt their complexity to the amount of data available for training. In the proposed systems it computes the optimum clustering for a range of different number of clusters and uses a distance called free energy to determine the optimum.

In Lapidot (2003) self-organizing maps (SOM) (Lapidot, Gunterman and Cohen (2002), Kohonen (1990)) are proposed for speaker clustering given a known number of speakers. This is a VQ algorithm for training the code-books representing each of the speakers. An initial non-informative set of code-books is created and then SOM is iterated, retraining them until the number of acoustic frames switching between clusters is close to 0. In order to determine the optimum number of clusters a likelihood function is defined (derived from the code-words in the code-books by assigning a Gaussian pdf) and BIC is used.

### 2.3.3 Use of Support Information in Diarization

When used for certain applications it is feasible to obtain an improvement in speaker diarization by using information other than the acoustics. In this section the use of the transcripts from the recording and the time delays between channels in a multi-microphone setting are visited.

#### Helping Diarization Using the Spoken Transcripts

A very interesting area of study to improve speaker Diarization in certain conditions is the use of the transcripts from the acoustic signal in order to extract information that can help assigning each speaker turn to each cluster. Such transcripts can be obtained via an automatic speech recognition system.

In Canseco-Rodriguez, Lamel and Gauvain (2004a), Canseco-Rodriguez, Lamel and Gauvain (2004b) and Canseco, Lamel and Gauvain (2005) the use of such linguistic information is studied for the domain of broadcast news, where people normally present themselves and interact with the other speakers calling them by their names. In these, they propose a set of rules to identify the speaker presenting himself, and the speakers who he precedes and who speaks after him. The rules are applied to speaker turns generated with a decoder-based system which is the output of an ASR system, but no further speaker diarization techniques are proposed.

## Speaker Diarization Using Multi-Channel Information

One outstanding characteristic of the meetings domain is that multiple microphones are usually available for processing. The time differences between microphones can be used as a feature to identify the speakers in a room by their locations as the speech uttered by each speaker takes a different time to reach each of the microphones according to their position in the room. Such feature has two main drawbacks from the acoustic features. On one hand it is prone to errors when speakers are located in symmetry to the microphones. On the other hand, they become less tractable when two speakers move inside the room, which accounts then for tracking algorithms to be used.

For the task of speaker segmentation, in Lathoud, McCowan and Odobez (2004) a speaker tracking approach is proposed using only between channel differences. In Lathoud, Odobez and McCowan (2004) the same is extended to speaker clustering and algorithms are proposed for detection of concurrent events. Ellis and Liu (2004) and Pardo et al. (2006a) also use only delays for clustering.

Given the literature, the delays between channels can not outperform the acoustic features, although in Ajmera, Lathoud and McCowan (2004) it is shown that the combination of delays and MFCC parameters can improve clustering. In Pardo et al. (2006b) it reaches the same conclusion and further improves results by using a weighted combination of the delays and MFCC likelihoods.

## 2.4 Speaker Diarization in Meetings

On recent years there has been increasing emphasis on research for speech and video processing for the meeting room domain. Within the different projects that are interested in this area, two different alternative meeting settings have been proposed. On one hand, some consider a lecture environment where a single speaker gives a talk in front of an audience, which intervenes at different points of the lecture with questions and remarks. In this situation there is always a main speaker, facing an audience, and many people listening, facing the speaker. On the other hand, the conference room environment is a gathering of people where mostly everyone speaks and discussions are being carried on one or more common topics to all attendees.

### 2.4.1 Current Meeting Room Research Projects

There are many research institutions carrying out research on one topic or another related to meetings. In here some of the projects that have led the research efforts in the latest years are pointed out.

The Interactive Multimodal Information Management (IM2), ongoing program sponsored by the Swiss government, aims at the study of multimodal interaction, covering a wide range of activities and applications, including the recognition and interpretation of spoken, written and gestured languages, computer vision, and the automatic indexation and management of multimedia documents. Other important related themes are information content protection, data access control, and the structuring, retrieval and presentation of multimedia information (*Interactive Multimodal Information Management (IM2) website* 2006). Linked to IM2, the project named m4 (multimodal meeting manager) was supported by the EU IST Programme and ran from 2002 to 2005. It was concerned with the construction of an automated meeting browser to be able to browse, query and structure meetings taking place in a room equipped with multiple sensors (*Multimodal Meeting Manager (M4) website* 2006).

The project Computers in the Human Interaction Loop (CHIL) aims at the creation of computers to help in the normal human-human interaction, in a non obtrusive way. CHIL is an Integrated Project (IP) within the European Union (EU) sixth framework program which started in 2004 for three years. Within the many lines of research it opened, several intelligent meeting rooms with audio and video sensors were built where data is collected and research is performed on the lecture-type meetings (*Computers in the Human Interaction Loop (CHIL) website* 2006).

The project Augmented Multimodal Interaction (AMI) is focused on the use of advanced signal processing, machine learning models and social interaction dynamics to improve human-to-human communications, particularly during business meetings between local and remote (virtual) participants (*Augmented Multiparty Interaction (AMI) website* 2006). The AMI project is also an IP project within the European sixth framework program, focusing on the conference-type meetings. AMI has been granted a continuation project (called AMIDA) within the European Union seventh framework program.

There are other projects with emphasis on multimodal interaction and human-to-human communications. Some of them are the “Similar” network of excellence (*Similar Network of Excellence website* 2006), the Pascal network (*Pattern analysis, Statistical modeling and Computational learning (Pascal) website* 2006) and Humaine emotions research (*Humaine emotion research website* 2006) in Europe, and Video analysis and content extraction for defense intelligence (VACE) (*Video analysis and content extraction for defense intelligence (ARDA-VACE II)* 2006) and Cognitive Assistant that Learns and Organizes (CALO) (*Cognitive Assistant that Learns and Organizes (CALO) website* 2006) in the USA.

## 2.4.2 Databases

In order for research to be performed in speech technologies, there is a constant need for data collection and annotation. In this respect there have been several efforts over the years to collect data on the meeting environment. On the particular area of speaker diarization systems for Meetings, there needs to be meetings databases accurately transcribed into speaker segments. Nowadays a few databases are already available and a few more are currently being recorded and transcribed, some of them are:

- ICSI Meetings Corpus (*ICSI Meetings Recorder corpus* (2006), Janin, Baron, Edwards, Ellis, Gelbart, Morgan, Peskin, Pfau, Shriberg, Stolcke and Wooters (2003)): 75 meetings with about 72 hours in total. They were recorded in a single meeting room, with 4 omnidirectional tabletop and 2 electret microphones mounted on a mock PDA.
- CMU Meeting Corpus (*CMU Meetings Corpus website* (2006), Burger, Maclaren and Yu (2002)) : 104 meetings of an average duration of 60 minutes with 6.4 participants (in average) per meeting (only 18 meetings are publicly available through LDC). They are focused on a given scenario or topic, changing from meeting to meeting. Initial meetings have 1 omnidirectional microphone, newer ones have 3 omnidirectional tabletop microphones.
- NIST Pilot Meeting Corpus (*NIST Pilot Meeting Corpus website* 2006): Consists of 19 meetings with a total of about 15 hours. Several meeting types are proposed to the attendants. Audio recordings are done using 3 omnidirectional table-top microphones and one circular directional microphone with 4 elements.
- CHIL Corpus: Recordings were conducted in 4 different meeting room locations consisting on lecture type meetings. Each meeting room is composed of several distant microphones, as well as speaker localization microphones and microphone arrays. Each meeting also contains several video cameras.
- AMI corpus (*Augmented Multiparty Interaction (AMI) website* 2006): About 100 hours of meetings with generally 4 participants were recorded, transcribed and released through their website. These are split into two main groups: real meetings and scenario-based meetings (where people are briefed to talk about a particular topic). One or more circular arrays of 8 microphones each are centrally located in the table. no video was collected.
- M4 audio-visual corpus (McCowan, Gatica-Perez, Bengio, Lathoud, Barnard and Zhang 2005): Created within the auspices of the M4 project (EU sponsored), used multiple microphones and cameras to record each participant.
- VACE multimodal corpus (Chen, Rose, Parrill, Han, Tu, Huang, Harper, Quek, McNeill,

Tuttle and Huang 2005): Is a video and acoustics meeting database created within the ARDA VACE-II project recording mainly military related meetings.

- LDC meetings data: The Linguistic Data Consortium (LDC) has been in charge of transcribing and distributing most of the databases in this list. Also, in an effort to contribute to the NIST Meetings evaluation campaigns, it recorded a set of meetings (Strassel and Glenn 2004) within the SPINE/ROAR project (*Speech in noisy environments* 2006).

### 2.4.3 NIST RT Speaker Diarization Systems for Meetings

The National Institute for Standards and Technology (NIST) (*National Institute for Standards and Technology* 2006) has been organizing multiple evaluations over the years on many aspects of speech technologies. In the area of speaker diarization evaluations, they started in year 2000 with interest in telephone speech (2000, 2001, 2002), broadcast news (2002, 2003, 2004) and meetings (2002, 2004, 2005, 2006). In the latest two years, focus has been geared exclusively towards the meetings environment.

The datasets used in the meetings evaluations were hand-transcribed by LDC. This acoustic data constitutes the basis for the development and evaluation of the algorithms proposed in this thesis. Initially, in 2002, the speaker segmentation task was enclosed within the speaker recognition evaluation (SRE-02) and used data from the NIST meeting room research project. This changed for 2004-2006 when speaker diarization has been a part of the Rich Transcription (RT) evaluation (RT04s, RT05s and RT06s), grouping it with the speech-to-text evaluation (STT) on meetings data. The datasets used for these evaluations contain data from CMU, ICSI, LDC, NIST, CHIL and AMI.

In the following sections the main ideas in the systems presented to each of the NIST meetings evaluations are explained, together with the particular algorithms that were created explicitly for processing of meetings data.

#### NIST 2002 Speaker Recognition Evaluation

In 2002 NIST started the series of speaker diarization evaluations for meetings including them in the speaker recognition evaluation. In that occasion systems were evaluated for broadcast news recordings, telephone conversations and meetings recordings. In that case only one channel of audio data was provided for any of the cases, therefore multiple channel techniques were not necessary. The meetings data used was recorded by NIST.

There were four participants in that evaluation, namely CLIPS-IMAG, LIA, ELISA consortium and MITLL. The systems can be grouped in two:

- The ELISA consortium (Moraru, Meignier, Besacier, Bonastre and Magrin-Chagnolleau 2002) was formed at that time by three laboratories in France (LIA, CLIPS-IMAG and Lab. Dynamique du langage, DDL). They presented two systems (both based on hierarchical clustering, a top-down and a bottom-up system), which they presented independently (LIA and CLIPS-IMAG individual submissions) and then combined (ELISA submission) in the same way as in Moraru, Meignier, Besacier, Bonastre and Magrin-Chagnolleau (2004) and Fredouille et al. (2004). The ELISA consortium and/or its individual components have been constant and active participants in all the speaker diarization evaluations since 2002. In Moraru, Besacier, Meignier, Fredouille and francois Bonastre (2004) they describe the evolution of their system over these years.
- MIT Lincoln Labs (MITLL) presented a system inspired in speaker identification techniques (Dunn, Reynolds and Quatieri 2000). It first performs a speaker segmentation using a modified GLR metric like in Wilcox et al. (1994) and follows with a GMM-UBM modeling technique to cluster segments into the different speakers.

### **NIST 2004 Rich Transcription Spring Meeting Evaluation**

Within the NIST 2004 Spring Rich Transcription Evaluation (*NIST Spring Rich Transcription Evaluation in Meetings website*, <http://www.nist.gov/speech/tests/rt/rt2005/spring> 2006) speaker diarization was evaluated in meeting recordings in two different conditions: Multiple Distant Microphones (MDM) and Single Distant Microphone (SDM). The MDM condition uses multiple microphones located in the center of a meetings table, and the SDM case uses only one of these microphones, normally the most centrally located. This is the first time that this task was performed for meetings environment on the MDM condition. A full description of the different tasks evaluated and the results of such evaluation can be found in Garofolo, Laprun and Fiscus (2004). Following are the approaches (in brief) that the participants proposed for the MDM and SDM conditions:

- Macquarie University in Cassidy (2004) proposes the same system for SDM than for MDM, using always the SDM channel. A BIC based speaker segmentation step is followed by an agglomerative clustering using Mahalanobis distance between clusters and BIC as stopping criterion.
- The ELISA consortium in Fredouille et al. (2004) proposes a two-axis merging strategy. An horizontal merging consists on the collapse and resegmentation of the clustering output of their two expert systems (based on BIC and EHMM) as proposed in the RT03 and SRE02 evaluations (Moraru, Meignier, Fredouille, Besacier and Bonastre 2004). This is done for each individual MDM channel or for the SDM channel. The vertical merging is applied

when processing multiple channels and unifies all the individual channels into one single resulting output by merging all channels at the output level. It uses an iterative process that searches for the longest speaker interventions that are common to all outputs and finally assigns to the closest speaker those segments of short duration where the different channels do not agree on.

- Carnegie Mellon University (CMU) in Jin et al. (2004) presents a clustering scheme based on GLR distance and BIC stopping criterion. In order to obtain the initial segmentation of the data it does a three steps process, first a Speech Activity Detection (SAD) is done over all the channels, then the resulting segments for all channels are collapsed into a single segmentation and the best channel (according to an energy/SNR metric) is chosen for each segment. Finally GLR change detection is applied on segments  $>5s$  to detect any missed change point. The speaker clustering is done using a global GMM trained on all the meeting excerpt data and adapted to each segment and uses GLR to compute the cluster pair distances to be used in an agglomerative clustering processing with BIC stopping criterion.

### **NIST 2005 Rich Transcription Spring Meeting Evaluation**

The RT05s evaluation welcomed a different kind of meetings to be evaluated. These are the meetings in a lecture environment, where a speaker is giving a lecture in front of an audience and there are eventual questions and answer periods. In this evaluation systems could be presented for either or both subtasks (lecture room and conference room data). The sets of microphones used was extended from the previous evaluations due to the existence of two new kinds in the lecture room data (entirely recorded by the partners in the CHIL project). These were labelled as MM3A (Multiple Mark III microphone arrays) which consisted on one or several 64 elements microphone arrays developed by NIST and positioned on one of the walls of the meetings room; and MSLA (Multiple source localization microphones) which are four sets of four microphones each, used primarily for speaker localization, but available also for speaker diarization. For a more thorough description of the tasks and microphone types please refer to Fiscus, Radde, Garofolo, Le, Ajot and Laprun (2005). The following is a brief description of the approaches taken in this evaluation:

- The Macquarie University system (Cassidy 2004) participated only on SDM which expands its work from the RT04s system. In the RT05s submission it uses the KL distance between clusters and does a post-processing of the segments using speaker identification techniques to refine the segments-to-speakers assignment.
- The TNO speaker diarization system (van Leeuwen 2005) presents a system for MDM using a single channel. It first uses a Speech Activity Detector (SAD) to filter out non-speech

frames. Then it does a segmentation and clustering using an agglomerative clustering via BIC.

- The ICSI-SRI speaker diarization system (Anguera, Wooters, Peskin and Aguilo 2005) uses a filter&sum module to obtain an enhanced signal on the MDM condition, and then uses an iterative agglomerative clustering using a BIC-alike metric. This system and its improvements for RT06s are described in this thesis.
- The ELISA consortium system (Istrate, Fredouille, Meignier, Besacier and Bonastre 2005) is different from their system in RT04s in that a preprocessing step is performed on the MDM channels to obtain a single enhanced channel. It is based on a weighted sum of the individual channels, weighted by their relative Signal to Noise Ratio (SNR) without any relative delays estimation. Three different clustering systems are then proposed. The first system is based on EHMM (Meignier et al. 2001), doing a top-down clustering. The second and third systems are both bottom-up, one using speaker change detection via GLR and agglomerative clustering via BIC, and the other using BIC for change detection and UBM-BIC in the agglomerative clustering part. All systems use a resegmentation stage at the end in order to refine the speaker segments. For this evaluation either system was run individually, with no collapse of the different outputs.

### **NIST 2006 Rich Transcription Spring Meeting Evaluation**

The RT06s evaluation continues its parallel testing of conference room data and lecture room data. This year five laboratories participated in the evaluation, making it a very good evaluation in terms of new systems and ideas. For a full description refer to Fiscus, Ajot, Michet and Garofolo (2006). An overview of the systems in RT06s follows:

- The Athens Information Technology (AIT) system (Rentzeperis, Stergiou, Boukis, Pnevmatikakis and Polymenakos 2006) uses a speaker segmentation and then clustering steps. The classic BIC implementation (Shaobing Chen and Gopalakrishnan 1998) is used for speaker segmentation as their primary system. A contrastive system uses a silence-based method cutting segments in silence points. A first step of the clustering process it also uses BIC to merge adjacent segments believed to be from the same speaker. Finally, all segments are modeled with GMM and a likelihood based technique is used to cluster them.
- The LIMSI system (Zhu et al. 2006) adapts their high-performance system presented for RT04f (Zhu et al. 2005) in order to process lecture room data. It is based on a 2-stage processing where a BIC agglomerative clustering precedes a speaker identification module where cross likelihood (Reynolds et al. 1998) is used to finish the clustering. In this system the speech activity detection module is reworked to adapt it to the lecture acoustics by

using a likelihood ratio between pretrained speech and silence models. The MDM condition is processed by randomly selecting one of the channels in the set and running the system in that one alone.

- The LIA system (improvements of the E-HMM based speaker diarization system for meetings records 2006) presents a single system based on the EHMM top-down hierarchical clustering that has been presented in previous evaluations. In this submission there are a few improvements to the system. One improvement deals with the selection of new speakers added to the system, which is modified to take into account all currently selected speakers to make it more robust and allow for all speakers to fall at least in one cluster. Also, a segment purification algorithm is proposed following Anguera, Wooters, Peskin and Aguilo (2005) in order to purify the existing clusters from segments belonging to other speakers. Furthermore, some feature normalization techniques were applied at the frontend level. Finally, an algorithm to detect overlapping speech was proposed, although it did not succeed in lowering the final diarization error rate.
- The AMI team (Leeuwen and Huijbregts 2006) was formed by TNO and University of Twente. They presented three systems to the evaluation. The first system is very similar to what was presented by TNO in RT05s (van Leeuwen 2005). The other two systems use a hierarchical clustering following the work at ICSI and presented in Anguera, Wooters, Peskin and Aguilo (2005). One of the two systems improves in runtime by considering a Viterbi-based clusters merging criterion. Each cluster is taken out of the ergodic HMM model (one at a time) and a Viterbi decoding gives the likelihood of the rest modeling the data. The cluster which causes the least loss in likelihood is eliminated and merged with the rest. The system iterates while the overall likelihood increases.
- The ICSI system (Anguera, Wooters and Pardo 2006b) is based on the system for RT05s (Anguera, Wooters, Peskin and Aguilo 2005) and includes many new ideas which will be covered in the rest of this thesis. The main step forward is the total independence from training data achieved by the creation of a new hybrid speech/non-speech detector (Anguera, Aguilo, Wooters, Nadeu and Hernando 2006) and the inclusion of delays as an independent feature stream.

## 2.5 Multichannel Acoustic Enhancement

Possibly the most noticeable difference when performing speaker diarization in the meetings environment versus other domains (like broadcast news or telephone speech) is the availability, at times, of multiple channels which are laid out inside the meetings room, synchronously recording what occurs in the meeting.

In order to take advantage of this fact one needs to explore an area of signal processing that differs from standard speech modeling techniques pointed out in previous sections and which constitutes a complex topic of research by itself. This is the area of microphone array beamforming for speech/acoustic enhancement (see for example Veen and Buckley (1988), Krim and Viberg (1996)). Although the task at hand differs from some of the assumptions taken in the beamforming theory, it will be found beneficiary to take it as a background for the use of all the microphones available.

Microphone array beamforming techniques usually take advantage of the fact that the same acoustic signal arrives to each of the microphones (forming the shape decided for the array) at a slightly different time due to the delay of propagation of the signal through the air. By combining the signals of all microphones (in different ways) one can simulate a directional microphone whose acoustic beam focuses on the speaker or acoustic event which is predominant, at each instant, in the meetings room. There are multiple acoustic beamforming techniques which require different degrees of knowledge on the microphone characteristics and the location of the speakers.

First, a theoretical overview of acoustic signal beamforming is given in 2.5.1, followed by a look at the most predominant acoustic beamforming techniques in 2.5.2. In the task at hand one does not know a priori how many speakers there are, or their locations in the room, therefore several possible techniques to find the Time Delay of Arrival (TDOA) between microphone pairs will also be reviewed in 2.5.3.

### 2.5.1 Introduction to Acoustic Array Processing

One singular trait of meeting rooms is the existence in some settings of multiple microphones recording the meeting synchronously. This is taken advantage of in this thesis to obtain a better signal to be further processed by the speaker diarization system. In this section the basic concepts behind microphone array processing are introduced to serve as a background on the developed techniques for this thesis.

#### Acoustic Signal Propagation

In general, in terms of signal propagation, an active speaker can be considered as an acoustic source which emits an acoustic signal that propagates through the air until it reaches each of the microphones in the room. In order to define an equation for such acoustic signal one can consider it as a longitudinal wave that propagates generating areas of compression and expansion. Using Newton's equations of motion of the volume in a fluid, and considering a semi-ideal case (McCowan (2001), Brandstein and Ward (2001)) one obtains:

$$\nabla^2 x(t, \mathbf{r}) - \frac{1}{c^2} \frac{\delta^2}{\delta t^2} x(t, \mathbf{r}) = 0 \quad (2.23)$$

were  $\nabla^2$  is the Laplacian operator,  $x(\cdot)$  is the wave field (of any sort) as a function of time and space,  $\mathbf{r}$  is the 3D position of the wave and  $c$  is the speed of sound (about 330 m/s in air).

In microphone array processing this equation can be solved for two particular cases. On one hand, when the acoustic wave field is considered monochromatic and plane (for far-field conditions) it is solved as

$$x(t, \mathbf{r}) = A(t)e^{j(\omega t - \mathbf{k}\mathbf{r})} \quad (2.24)$$

where  $\omega = 2\pi f$  is the considered frequency (in radians per second),  $A(t)$  is the wave field amplitude and  $\mathbf{k}$  is the wavenumber and is defined as

$$\mathbf{k} = \frac{2\pi}{\lambda} [\sin \theta \cos \phi \quad \sin \theta \sin \phi \quad \cos \theta]$$

where  $\lambda$  is the wavelength ( $\lambda = c/f$ ),  $\theta$  and  $\phi$  are the polar coordinates for elevation and azimuth (respectively) of the waveform position in space.

On the other hand, when the wave is considered spherical (propagating in all directions), as used in near-field conditions, it is solved as

$$x(t, \mathbf{r}) = \frac{-A(t)}{4\pi r} e^{j(\omega t - kr)} \quad (2.25)$$

where now  $r = |\mathbf{r}|$  determines the scalar distance to the source in any direction and  $k$  is the scalar version of the wavenumber,  $k = 2\pi/\lambda$  for all directions.

From these formulas one can observe how any acoustic wave can be sampled both in time and space in a similar way (both dimensions being in the exponential). Time sampling is done to obtain a digital signal and space sampling is done by a microphone array. In both cases one can reconstruct the original signal as long as it complies with the Nyquist rule (Ifeachor and Jervis 1996) (or else there will be spacial/temporal aliasing).

## Passive Apertures

In order to describe the effect of the signal when received by a microphone array, the theory behind transmission/reception of propagating waves needs to be reviewed. An *aperture* is defined as a spacial region designed to emit (active) or receive (passive) propagating waves. The concept of aperture is very broad and is used for many different kinds of waves.

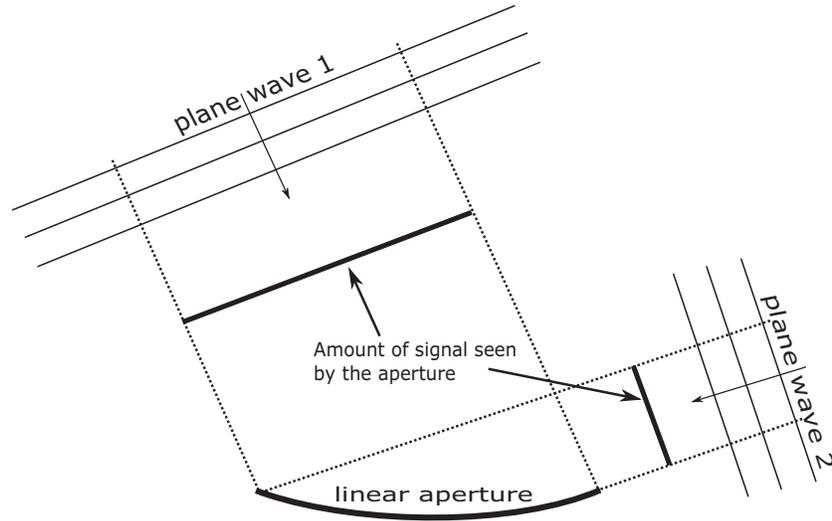


Figure 2.2: Example of a passive aperture response to different incoming signals

As can be seen in figure 2.2, a passive aperture has a particular spacial orientation in space and therefore alters the receiving signal in a different way for each frequency and location. In this context the *aperture function* or *sensitivity function* ( $A(f, \mathbf{r})$ , with impulsive response  $\alpha(t, \mathbf{r})$ ) is defined as such response of the aperture to the incoming signal  $x(\tau, \mathbf{r}) \xrightarrow{\mathcal{F}} X(f, \mathbf{r})$ , resulting on  $X_R(f, \mathbf{r})$  as

$$x_R(t, \mathbf{r}) = \int_{-\infty}^{\infty} x(\tau, \mathbf{r}) \alpha(t - \tau, \mathbf{r}) d\tau \xleftrightarrow{\mathcal{F}} X_R(f, \mathbf{r}) = X(f, \mathbf{r}) A(f, \mathbf{r}) \quad (2.26)$$

The aperture function is defined for a particular direction of arrival. In order to measure and characterize the response of an aperture for all directions, the directivity pattern (or beam pattern) is defined as the aperture response to each frequency and direction of arrival. It is given by:

$$D_R(f, \boldsymbol{\alpha}) = \mathcal{F}_{\mathbf{r}}\{A(f, \mathbf{r})\} = \int_{-\infty}^{\infty} A(f, \mathbf{r}) e^{j2\pi\boldsymbol{\alpha}\mathbf{r}} d\mathbf{r} \quad (2.27)$$

where  $\mathcal{F}_{\mathbf{r}}$  is the 3D fourier transform,  $\mathbf{r}$  now indicates a point along the aperture and  $\boldsymbol{\alpha}$  is the direction vector of the wave

$$\boldsymbol{\alpha} = \frac{1}{\lambda} [\sin \theta \cos \phi \quad \sin \theta \sin \phi \quad \cos \theta]$$

## Linear Apertures Theory

Over the general directivity pattern in eq. 2.27 one can apply some simplifications, oriented towards array processing:

- The aperture has a linear shape:  $\mathbf{r} = [x, 0, 0]$
- A far-field signal is received (therefore  $|\mathbf{r}| > \frac{2L^2}{\lambda}$ , with  $L$  being the total length of the aperture.
- The aperture function is constant for all frequencies.

In this case the directivity function simplifies to

$$D_R(f, \alpha_x) = L \text{sinc}(\alpha_x L) \quad (2.28)$$

which contains zeros of reception at  $\alpha_x = \frac{m\lambda}{L}$  with  $m$  being a scalar value.

At all effects, a linear sensor array can be considered as a sampled version of a continuous linear aperture. One can obtain the aperture function of the array as the superposition of all individual element functions ( $e_n(\cdot)$ ) which are equivalent to the array function and measure the element's response for a particular direction of arrival. The aperture function is now written as:

$$A(f, x) = \sum_{n=-\frac{N-1}{2}}^{\frac{N-1}{2}} w_n(f) e_n(f, x - x_n) \quad (2.29)$$

for an array with  $N$  elements, where  $e_n$  is the element function for element  $n$ ,  $w_n(f)$  is the complex weighting for element  $n$  and  $x_n$  is the position of such element in the  $x$  axis.

For the far-field case, and considering all elements with identical element function, the directivity function can be computed as

$$D(f, \alpha_x) = \sum_{n=-\frac{N-1}{2}}^{\frac{N-1}{2}} w_n(f) e^{j2\pi\alpha_x x_n} \quad (2.30)$$

In where the complex weighting can be expressed as module and phase in the following way:

$$w_n(f) = a_n(f) e^{j\varphi_n(f)} \quad (2.31)$$

where  $a_n(f)$  can be used to control the shape of the directivity and  $\varphi_n(f)$  to control the angular location of the main lobe, being both scalar functions.

Beamforming techniques that use a microphone array for acoustic enhancement of the signal play with these two parameters to obtain the desired shaping and steering of the lobes of the directivity pattern to certain locations in the space. Some of these techniques use the approximation of far-field signals done in here and others (fewer) consider near-field waves, with different directivity pattern development.

## 2.5.2 Microphone Array Beamforming

The application of the general signal beamforming theory to the case of acoustic beamforming has some peculiarities and has been broadly studied in the past. In general it is considered that the acoustic signal is generated by a far-field source (therefore it arrives at the microphones as a flat wave) and that it has usually been considered as having a narrow-band frequency response (not taking into consideration the different behavior of the arrays to multiple frequencies).

There are two main groups of beamforming techniques that can be found in the bibliography. These are data-independent (or fixed) and data-dependent (or adaptive). The techniques that are data-independent fix their parameters and maintain them throughout the processing of the input signal. Data dependent techniques update their parameters to better suit the input signal, adapting to changing noise conditions. Moreover, there are several postprocessing techniques that are applied after the beamforming, some of them very linked to the beamforming process.

Fixed beamforming techniques are simpler to implement than the adaptive ones, but are more limited in their ability to eliminate highly directive (and sometimes changing) noise sources. The simplest beamforming technique in this group is the delay&Sum (D&S) technique (Flanagan, Johnson, Kahn and Elko (1994), Johnson and Dudgeon (1993)). The output signal  $y[n]$  is defined as:

$$y(n) = \frac{1}{M} \sum_{m=1}^N x_m(n - \tau_m) \quad (2.32)$$

given a set of  $M$  microphones, where each microphone has a delay of  $\tau_m$  relative to the others. In this technique all channels are equally weighted at the output. The D&S beamforming is a particular case of a more general definition of a filter&sum beamforming where an independent filter is applied to each channel:

$$y(n) = \sum_{m=1}^N w_m[n] x_m(n - \tau_m) \quad (2.33)$$

One application of such techniques is the Superdirective Beamforming (SDB) (Cox, Zeskind and Kooij (1986), Cox, Zeskind and Owen (1987)), where the channel filters (or also called superdirective beamformers) are defined to maximize the array gain (or directivity factor), which is defined as the improvement in signal to noise ratio between the reference channel and the “enhanced” system output.

For the case of near-field signals (like when a microphone array is located right in front of the speaker in a workstation) the SDB has been reformulated (Tager (1998a), Tager (1998b), McCowan, Moore and Sridharan (2000)) by using the near-field propagation functions for the acoustic waves, where waves are not considered planar anymore.

Considering the speech signal to be narrow-band simplifies the design of beamforming systems but does not represent well the reality. To deal with broadband signals in an effective manner, several sub-array beamforming techniques have been proposed (Fischer and Kammeyer (1997), Sanchez-Bote, Gonzalez-Rodriguez and Ortega-Garcia (2003)) where the set of microphones is split into several sub-arrays which focus their processing in a particular band, collapsing all the information into the “enhanced” signal at the end.

The adaptive beamforming techniques present a higher capacity at reducing noise interference but are much more sensitive to steering errors due to the approximation of the channel delays.

The Generalized Sidelobe Canceller (GSC) technique (Griffiths and Jim 1982) aims at enhancing the signal that comes from the desired direction while cancelling out signals coming from other sources. This is achieved by creating a double path for the signal in the algorithm. A standard beamforming path is modified by an adaptive path consisting of a blocking matrix and a set of adaptive filters that aim at minimizing the output noise power. The blocking matrix blocks the desired signal from the second path. At the end both paths are subtracted to obtain the output signal. In order to find the optimum coefficients for the lower part, an algorithms like the Least Mean Squares (LMS) can be used.

Although widely used, in practice the GSC can suffer from distortion of the output signal normally called signal leakage. This is due to the inability of the blocking matrix to completely eliminate the desired signal from the adaptive path (which is very common in speech due to its broadband properties). This problem is treated in Hoshuyama, Sugiyama and Hirano (1999) where the blocking matrix is designed with control of the allowed target error region.

A different kind of adaptive beamforming techniques are those that allow a small amount of distortion of the desired signal as it is considered not to affect the quality of the signal as perceived by human ears. One of such techniques is named the AMNOR (Adaptive Microphone-array system for Noise Reduction), introduced by Kaneda and Ohga (1986), Kaneda (1991) and Kataoka and Ichirose (1990). It introduces a known fictitious desired signal during noise-only periods in order to adapt the filters to cancel such signal and therefore improve the quality of

the speech parts. One drawback of this technique is the need for accurate speech/non-speech detection.

Some efforts have been reported applying the adaptive beamforming techniques to the near-field case. In McCowan, Marro and Mauuary (2000) and McCowan, Pelecanos and Sridharan (2001) adaptive beamforming and super-directive beamforming are combined for this effect.

In real applications none of the previously described beamforming techniques achieves the levels of improvement on the signal set theoretically. In practice a post-processing of the acoustic signal is necessary in order to obtain the optimum output quality. In Zelinski (1988) a Wiener post-filtering is applied where time delays information is used to further enhance the signal in the filter. In Marro, Mahieux and Simmer (1998) it does a very thorough analysis of the interaction of Wiener filtering with a filter&sum beamforming, showing that the post-filter can cancel incoherent noise and allows for slight errors in the estimated array steering. Other post-filtering approaches applied to microphone arrays beamforming are proposed in Cohen and Berdugo (2002) and Valin, Rouat and Michaud (2004).

There are many post-processing techniques aimed to the “enhanced” single channel signal resulting from the beamforming. Some of them take into account acoustic considerations (Rosca, Balan and Beaugeant (2003), Zhang, Hansen and Rehar (2004)) or acoustic models (Brandstein and Griebel 2001) to better enhance the signal.

### 2.5.3 Time Delay of Arrival Estimation

In order to apply almost any of the array beamforming techniques in an acoustic signal, and given that the location of the acoustic source is not given, one needs a way to estimate the TDOA between channels or the Direction of Arrival (DOA) of the signal. In practice the DOA estimation is much less used for signal enhancement in this domain as its requirements and computational cost are normally higher than for TDOA. DOA estimation has also been considered less suitable than TDOA for broadband signals.

There have been many techniques proposed in the past in order to estimate the TDOA between a pair of sensors, like the use of LMS adaptive filters used in sonar (F. Reed and Bershad (1981), Schmidt (1986)).

However, the approaches that have become more popular on recent years have been those based on the cross-correlation of the signals. Given two real signals,  $x_1$  and  $x_2$ , the cross-correlation between them is defined as:

$$R_{x_1x_2}(m) = E[x_1(n) \cdot x_2^*(n - m)] \quad (2.34)$$

although, as in practice one cannot work with infinite signals, it is estimated as:

$$\hat{R}_{x_1x_2} = x_1(n) * x_2(n - m) = \sum_{n=-N}^N x_1(n) \cdot x_2^*(n - m) \quad (2.35)$$

where each signal has length  $N$ . In order to do this computation in a more efficient way, both signals are first Fourier transformed, the product is computed and then the inverse Fourier transform is applied.

When the cross-correlation between two signals is computed where one of the signals is a (similar) delayed version of the other by a time  $T$ , the main peak of the cross-correlation will be located at either time  $\pm T$  (depending on which signal is  $x_1$  and  $x_2$ ). In real applications though, there are many disturbing factors that will affect the position of the peak or will mask it. These factors can be noise, reverberation and others. The case of reverberation has been greatly studied in the literature (Champagne, Bedard and Stephenne (1996), Brandstein and Silverman (1997)).

Addressing this problem, the Generalized Cross Correlation (GCC) was introduced (Knapp and Carter 1976). It implements a frequency domain weighting of the cross correlation according to different criteria, in order to make it more robust to external disturbing factors. The general expression for the GCC is:

$$R_{x_1x_2}^{GCC}(m) = \mathcal{F}^{-1}(X_1(w) \cdot X_2^*(w) \cdot \psi(w)) \quad (2.36)$$

where  $\psi$  is a weighting function. If  $\psi = 1$  for all  $w$  the standard cross correlation formula is obtained.

The first weighting function that will be considered is the *Roth correlation* (Roth 1971), which weights the cross correlation according to the Signal to Noise Ratio (SNR) value of the signal. Its results approximate an optimum linear Wiener-Hopf filter (Trees 1968). Frequency bands with a low SNR obtain a poor estimate of the cross correlation and therefore are attenuated versus high SNR bands.

$$\psi_{ROTH}(w) = \frac{1}{X_1(w) \cdot X_1^*(w)} \quad (2.37)$$

A variation of the ROTH weight is the Smoothed Coherence Factor (*SCOT*) (Carter, Nuttall and Cable 1973) which acts upon the same SNR-based weighting concept, but allows both signals being compared to have a different spectral noise density function.

$$\psi_{SCOT}(w) = \frac{1}{\sqrt{X_1(w) \cdot X_1^*(w) \cdot X_2(w) \cdot X_2^*(w)}} \quad (2.38)$$

In environments with high reverberation, the Phase Transform (*PHAT*) weighting function (Knapp and Carter 1976) is the most appropriate as it normalizes the amplitude of the spectral density of the two signal and uses only the phase information to compute the cross correlation. It is applied to speech signals in reverberant rooms by Brandstein and Silverman (1997).

$$\psi_{PHAT}(w) = \frac{1}{|X_1(w) \cdot X_2^*(w)|} \quad (2.39)$$

The GCC-PHAT achieves very good performance when the SNR of the signal is high, but deteriorates when the noise level increases. This is the solution used as weighting function in the beamforming implementation proposed in this thesis.

Another weighting function of interest is the *Hannan & Thomson* (Knapp and Carter (1976), Brandstein, Adcock and Silverman (1995)), also known as Maximum Likelihood (ML) correlation, which also tries to maximize the SNR ratio of the signal. For speech applications, Brandstein et al. (1995) proposed the approximation:

$$\psi_{ML}(w) = \frac{|X_1(w)||X_2(w)|}{|N_1(w)|^2|X_2(w)|^2|N_2(w)|^2|X_1(w)|^2} \quad (2.40)$$

where  $N_1(w)$  is the noise power spectra.

Finally, the Eckart filter (Eckart 1952) maximizes the deflection criterion, i.e. the ratio of the change in mean correlation output due to signal present compared to the standard deviation of correlation output due to noise alone. The weighting function achieving this is:

$$\psi_{eckart} = \frac{S_1(w)S_1^*(w)}{N_1(w)N_1^*(w) \cdot N_2(w)N_2^*(w)} \quad (2.41)$$

where  $S_1(w)$  is the speech power spectra.



## Chapter 3

# Speaker Diarization: from Broadcast News to Meetings

When applying a known technique to a new task it is preferable to do it starting from some well rooted theory and some implementation that has been proven to be successful in a task similar to the proposed one, while analyzing its shortcoming on this new domain and proposing improvements to it. This is the case of the diarization system presented for the meetings environment, which is based on the system previously developed at the International Computer Science Institute (ICSI) for the task of broadcast news. It has been developed by proposing alternatives to the algorithms that had some room for improvement or that needed to be adapted to better fit the new domain. Also, given that the broadcast news system is designed to run only on a single-channel recording, the necessary algorithms have also been implemented to adapt the signals from multiple channels/microphones to be able to process them with the presented system.

This chapter covers the description of both the broadcast news system and the new meetings domain system, bridging the gap between both by analyzing the differences that have been observed during development.

In the first part, the broadcast news system is described in detail, pointing out the main ideas behind it and its implementation, and baseline results are shown regarding its performance for the NIST Rich transcription evaluations for broadcast news (RT03s and RT04f) in which ICSI participated.

Following the broadcast news description, a comparison on some of the parameters measurable on both domains (meetings and broadcast news) is offered. The differences between them are pointed out, as well as the areas where this thesis proposes improvements in converting a system from one task to the other.

Finally, a description of the meetings domain speaker diarization system is given. The detailed

description of all the novel algorithms involved in the new system is split between the current and next two chapters. In this chapter a detailed description is given of those algorithms that have been adapted from different sources but that are not considered a novelty of this thesis by themselves. It also gives an overview description of the rest of the algorithms (novel in this thesis) to obtain a complete view of the overall system.

The techniques considered to be the primary contribution of this thesis will be described in chapters 4, focusing in those algorithms within the single-channel speaker diarization system, and 5 which deals with the use of the multiple channels in a meeting room to further improve the system.

### 3.1 The ICSI Broadcast News System

The broadcast news (BN) system currently used at ICSI and which has been used as a base for the meetings system, was originally created by Jitendra Ajmera circa 2003. He built the system while he was a PhD student at EPFL (Lausanne, Switzerland) and IDIAP (Martigny, Switzerland) and implemented it at ICSI while visiting for 6 months. During Ajmera's stay, ICSI participated in the NIST 2003 Rich transcription of broadcast news spring evaluation with the developed system, and soon afterwards in the RT03f ("who spoke the words" evaluation). The diarization system was then improved and ICSI participated again in the RT04f evaluation (Wooters et al. 2004), also in broadcast news.

The system is a bottom-up agglomerative clustering approach that uses a modified version of the BIC distance (Ajmera et al. 2003) in order to iteratively merge the closest clusters until the same BIC distance determines the system to stop. Speaker segmentation of the data is not done explicitly before the clustering part, but it is done via Viterbi decoding of the data given the current speaker models at every iteration. For a thorough description of the system refer to Ajmera (2004).

The philosophy behind the system and all research that has been done towards implementation of the meetings system is based on these key concepts:

1. Make the system as robust as possible to data within the same domain which the system has not been adapted to.
2. Allow for a fast adaptation of the system to use it in new domains (i.e. broadcast news, meetings, telephone speech, and others).

These key concepts were put into practice by imposing the following guidelines:

- Use as few training data as possible so that the system can be easily adapted to new domains and is not over-tuned to the data it is trained on.
- Avoid as much as possible the use of thresholds and tuning parameters. If not possible, try to define parameters that once tuned can achieve good performance in different kinds of data.

The implementation of the broadcast news system used as a baseline for the meetings domain was presented to the RT04f broadcast news evaluation (Wooters et al. 2004), which is the latest broadcast news evaluation conducted by NIST within the EARS (Effective Affordable Reusable Speech-to-Text) program. It differs from the original diarization system created by Ajmera (Ajmera and Wooters 2003) in four main points. First the inclusion of a speech/non-speech detector to filter out the non-speech segments prior to doing any further processing to the data and the discontinuation of use of a speech/music classifier used in the RT03s evaluation. Also, the parameterization used was MFCC, instead of PLP used until then. Finally, the inclusion of an iterative segmentation-training loop in the algorithm to allow models to converge to the clusters data.

It can be seen in figure 3.1 the main blocks constituting the system. In the following sections a detailed description of the different blocks is given.

### 3.1.1 Speech/non-Speech Detection and Parameters Extraction

The use of a speech/non-speech detector in speaker diarization is important to ensure that acoustic models for each of the clusters correctly represents the speech data and is not “contaminated” by non-speech information. In the ICSI system each cluster is initially modeled using a small number of Gaussian mixtures (usually 5) given that they are trained using ML over a small amount of data. This causes that the inclusion of non-speech data into the training makes clusters resemble each other much more and make the system prone to clustering errors, together with non-speech errors.

The speech/non-speech detector for the BN system is a two-class detector, in which each class is modeled by a three-state HMM, with a minimum duration of 30 msec. The non-speech model includes both music and silence. The features used in the SNS detector (MFCC12) are different from the features used for clustering. This detector that was initially used for ICSI-SRIs BN STT system on RT04f. It was trained on 80 hours of 1996 HUB4 BN acoustic data. No tuning was made to adapt the detector to speaker diarization for the RT04f evaluation.

In order to illustrate the advantages of using a speech/non-speech (spnsp) detector (also sometimes referred as Speech Activity Detection, SAD) in table 3.1 (taken from Wooters et al. (2004)) diarization error rates are shown on the RT04f data set using different kinds of spnsp

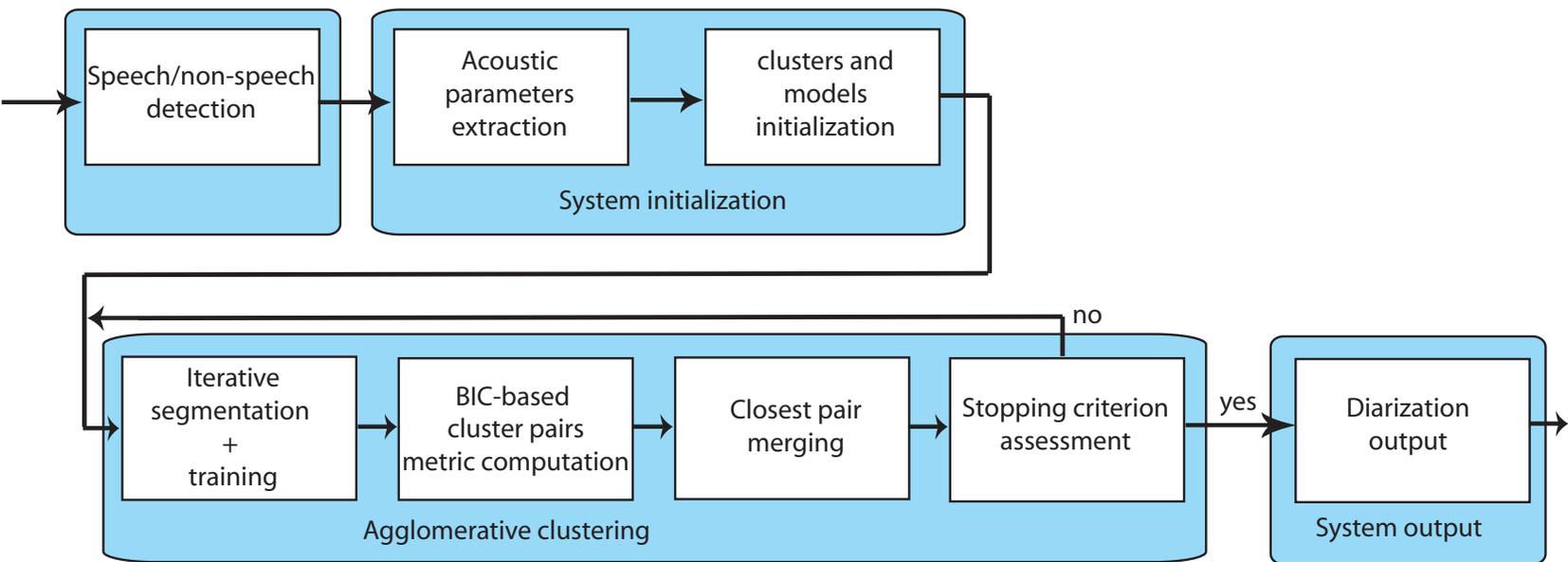


Figure 3.1: ICSI Speaker Diarization for Broadcast News blocks diagram

detectors. The Diarization Error Rate (DER) is the percentage of time that the system mis-attributes speakers/non-speech segments. It can be broken down into speaker errors, which accounts for miss-attributed speaker segments, false alarms (FA) and missed speech errors (MISS), which account for non-speech labelled as speech, and viceversa. For an exhaustive definition of each one of these types of error refer to section 6.1.3.

The first column shows the baseline system composed of the RT03f system. It has an overall non-speech error of 5.1% and a speaker error of 17.8%. By adding the speech/non-speech detector proposed for broadcast news it not only improved the non-speech errors but also reduces the speaker error, due to the reduction in clustering errors as noted above. Finally, it is interesting to see how much can be achieved in terms of DER if a perfect spnsp detector was built. Such detector is obtained by extracting the speaker segments from the reference segmentation and running the diarization with those as spnsp input. It can be seen that the proposed spnsp detector is still about 1.2% worse than the perfect detector. The speaker error is lower in the proposed spnsp detector than in the ideal one. This could indicate that some non-speech data can still be beneficiary to train discriminant speaker models. In this implementation the system obtained a 0.2% and 0.1% MISS errors in the perfect spnsp and baseline systems which was later reduced to 0%.

| System used  | %MISS | %FA | %SPKR | %DER  |
|--------------|-------|-----|-------|-------|
| RT03f system | 0.1   | 5.0 | 17.8  | 22.95 |
| +SRI/spnsp   | 1.5   | 1.2 | 15.4  | 18.17 |
| +ideal spnsp | 0.2   | 0.0 | 16.8  | 16.98 |

Table 3.1: *DER improvement by using a speech/non-speech detector*

With respect to the parameters used in the system, as it happens with other speech processing areas, acoustic modeling for speaker diarization is performed based of acoustic features extracted from the input signal. For the broadcast news system at ICSI the features used have been modified over the years finally settling down into the use of MFCC features with 19 coefficient, without any deltas or double deltas and without the zeroth cepstral coefficient, linked to the energy of the signal. For broadcast news these features were computed over a 60 millisecond analysis window in 20 milliseconds intervals. Multiple tests were done resulting on the selection of these features. On one hand, the increase in computation involved in using the delta and double delta coefficients was considered unacceptable given that the system gave mixed results when using them. On the other hand, MFCC19 were chosen as opposed to PLP12, which were used on RT03f, due to a slightly better performance when using them together with the spnsp detector.

As can be seen in table 3.2 also from Wooters et al. (2004), the baseline system using PLP and no spnsp detector produces better overall results than the counterpart MFCC system, but this second one is better when spnsp is added. In the diarization system for meetings a possible

combination to use delays as features is proposed which is also applicable to all other kinds of feature vectors.

| System used | %MISS | %FA | %SPKR | %DER  |
|-------------|-------|-----|-------|-------|
| RT03f PLP   | 0.1   | 5.0 | 15.8  | 20.93 |
| +SRI spnsp  | 1.6   | 1.2 | 15.5  | 18.36 |
| RT03f MFCC  | 0.1   | 5.0 | 17.8  | 22.95 |
| +SRI spnsp  | 1.5   | 1.2 | 15.4  | 18.17 |

Table 3.2: Comparison of PLP12 and MFCC19 parameterizations on RT04f

### 3.1.2 Clusters Initialization and Acoustic Modeling

The system implemented at ICSI for broadcast news is based on an agglomerative clustering process that iteratively merges clusters until getting to the optimum number of clusters. In order to initialize the system one needs to obtain an initial set of clusters  $K$  (where  $K > K_{opt}$ , the optimum number of clusters representing the number of speakers in the recording). During the implementation of the original system two alternatives were considered, on one hand, a k-means algorithm was tested in order to split the data into  $K$  clusters containing homogeneous frames. Another alternative was to split the data into even sized pieces. The second was finally selected due to its simplicity and the good results that it achieved on most data.

The linear initialization of the data into clusters is a simple algorithm that clusters the data according to its temporal proximity rather than acoustic proximity, allowing for models to be trained with acoustic data of very different acoustic characteristics that belongs to the same speaker.

In order to create  $K$  clusters the clusters initialization algorithm first splits the show into  $P$  partitions (where  $P = 2$  for Broadcast news). Then for each partition the data is split into  $K$  segments of the same size and labelled  $1 \dots K$ . The initial data for cluster  $k$  (where  $k \in 1 \dots K$ ) is the union of the data labelled  $k$  for each of the partitions. This technique is thought to work better than a more elaborate frame-level k-means algorithm because it takes into account the possible acoustic variation of the speech belonging to a single speaker. By clustering the data with k-means one cannot ensure that the resulting clusters contain frames from the same speaker, but maybe it contains acoustic frames that belong to the same phonetic class from several speaker.

Each initial cluster obtained via linear initialization it will most certainly have data belonging to more than one source/speaker. In order for the clusters to achieve some speaker homogeneity before stating the merging iterations the algorithm performs three iterations of models training and Viterbi segmentation of the data. Next section goes into more detail how clusters are modeled. The resulting clusters tend to contain data from a single speaker or at least a majority of

it.

There are some occasions when using linear initialization that it creates clusters with acoustic segments from more than one speaker, causing them potential merging errors and therefore a decrease in performance. In the improvements for the meetings room data a new initialization algorithm and a segment purification algorithm, that detects and splits such clusters, will be proposed.

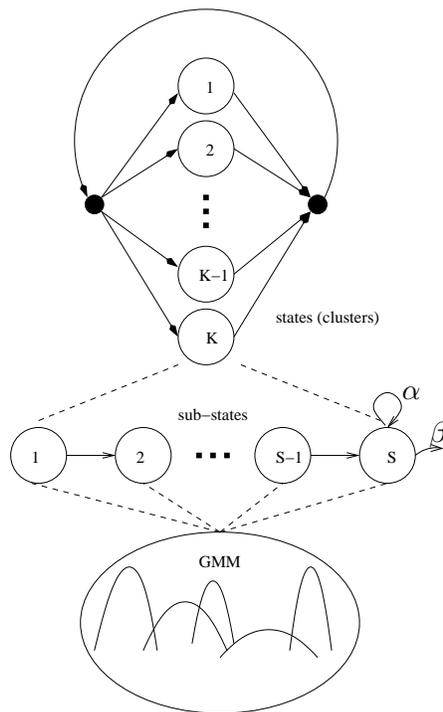


Figure 3.2: *Acoustic models for speaker clustering*

The broadcast news clustering algorithm models the acoustic data using an ergodic hidden Markov model (HMM) topology, as seen in figure 3.2, where each initial state corresponds to one of the initial clusters. Upon completion of the algorithm's execution, each remaining state is considered to represent a different speaker. Each state contains a set of  $M\_D$  sub-states, imposing a minimum duration of staying in any model. Each one of the sub-states has a probability density function modeled via a Gaussian mixture model (GMM). The same GMM model is tied to all sub-states in any given state. Upon entering a state at time  $n$ , the model forces a jump to the following sub-state with probability 1.0 until the last sub-state is reached. In that sub-state, we can remain in the same sub-state with transition weight  $\alpha$ , or jump to the first sub-state of another state with weight  $\beta/M$ , where  $M$  is the number of active states/clusters at that time. The diarization system for broadcast news used values  $\alpha = 0.9$  and  $\beta = 0.1$  with the intention of favoring the system to stay in the same cluster and therefore model speaker turns bigger than  $M\_D$  frames. As will be shown, this implicitly models the maximum length for a speaker term.

In the meetings system it is modified to reduce such constraint.

Each of the GMM models initially has a complexity of 5 Gaussian mixtures, which was optimized using development data from previous evaluations. Upon deciding that two clusters belong to the same speaker, one of the clusters/models is eliminated from the HMM topology,  $M$  is reduced by 1 and the resulting model is trained from scratch with a complexity being the sum of the previous two models. This ensures that the complexity of the overall system after any particular iteration remains constant and therefore the overall likelihood of the data given the overall HMM model can be compared between iterations.

### 3.1.3 Clusters Comparison, Pruning and Clusters Merging

Given  $M$  clusters with their corresponding models, the matrix of distances between every cluster pair is created and the closest pair is merged if it is determined that both clusters contain data from the same speaker. In order to obtain a measure of similarity between two clusters modeled by a GMM a modified version of the Bayesian Information Criterion (BIC) is used, as introduced by Ajmera in Ajmera, McCowan and Bourlard (2004) and Ajmera and Wooters (2003). As explained in the state of the art chapter, the BIC value quantifies the appropriateness of a model given the data. It is a likelihood-based metric that introduces a penalty term (in the standard formulation) to penalize models by their complexity.

Given two clusters  $C_i$  and  $C_j$  composed of  $N_i$  and  $N_j$  acoustic frames respectively, they are modeled with two GMM models  $\mathcal{M}_i$  and  $\mathcal{M}_j$ . The data used to train each of the models is the union of the data belonging to each one of the segments labelled as belonging to cluster  $C$ . In the same manner a third model  $\mathcal{M}_{i+j}$  is trained with  $C_{i+j} = C_i \cup C_j$ .

In the standard BIC implementation to compare two clusters (Shaobing Chen and Gopalakrishnan 1998) the penalty term adds a factor  $\lambda$  that is used to determine the effect of the penalty on the likelihood. The equation of the standard  $\Delta$ BIC for GMM models is

$$\begin{aligned} \Delta BIC(C_i, C_j) &= \mathcal{L}(C_{i+j} | \mathcal{M}_{i+j}) - (\mathcal{L}(C_i | \mathcal{M}_i) + \mathcal{L}(C_j | \mathcal{M}_j)) \\ &\quad - \frac{1}{2} \lambda (\#M_{i+j} - \#M_i - \#M_j) \log(N_i + N_j) \end{aligned} \quad (3.1)$$

where  $\#M$  is the number of free parameters to be estimated for each of the models, i.e. relative to the topology and complexity of the model.

It is considered that two clusters belong to the same speaker if they have a positive  $\Delta$ BIC value. Such value is affected by the penalty term (including the  $\alpha$ , which acts as a threshold determining which clusters to merge and which not to). The penalty term also modifies the order in which the cluster pairs are merged in an agglomerative clustering system, as each pair

will have a different number of total frames and/or models complexities, which will cause the penalty term to be different. In systems based on  $\Delta\text{BIC}$  this penalty term ( $\lambda$ ) is usually tuned based on development data and it always takes values greater than 0. In some cases two different values are defined, one for the merging criterion and the other one for the stopping criterion.

When training models to be used in  $\Delta\text{BIC}$ , these need to be trained using an ML approach, but there is no constraint in the kind of model to use. Ajmera's modification to the traditional BIC formula comes with the inclusion of a constraint to the combined model  $M_{i+j}$ :

$$\#M_{i+j} = \#M_i + \#M_j \quad (3.2)$$

This is an easy to follow rule as model  $M_{i+j}$  is normally built exclusively for the comparison.

By Applying this rule to the  $\Delta\text{BIC}$  formula one avoids having to decide on a particular  $\lambda$  parameter and therefore the real threshold which is applied to consider if two clusters are from the same speaker becomes 0. The formula of the modified-BIC becomes equivalent to the GLR, but with the condition that 3.2 applies.

The lack of an extra tuning parameter makes the system more robust to changes in the data to be processed, although, as the BIC formula is just an approximation of the Bayesian Factor (BF) formulation, sometimes the robustness increase comes with a small detriment on performance.

In the broadcast news system that has been described, the model  $M_{i+j}$  is generated directly from the two individual models  $i$  and  $j$  by pooling all the Gaussian mixtures together. Then the data belonging to both parent models is used to train the new model via ML. Training is always performed using an Expectation Maximization (EM-ML) algorithm and performing 5 iterations on the data.

Once the  $\Delta\text{BIC}$  metric between all possible cluster pairs has been computed, it searches for the biggest value and if  $\Delta\text{BIC}_{max}(C_i, C_j) > 0$  the two clusters are merged into a single cluster. In this case, the merged cluster is created in the same way as  $M_{i+j}$  is created, although it is not a requirement. The total complexity of the system remains intact through the merge as the number of Gaussian mixtures representing the data is the same, clustered in M-1 clusters.

The computation of  $\Delta\text{BIC}$  for all possible combinations of clusters is by far the most computationally intensive step in the agglomerative clustering system. Given that the models are retrained and the data is resegmented after each merge it obtains models at each iteration that are quite dissimilar to the models in the previous iteration, therefore it is recommended to compute all values again. Some techniques were tested to speedup this process by looking at the behavior of the  $\Delta\text{BIC}$  values. Some are:

- In every iteration merge more than one cluster pair, selecting them to be the pairs with highest BIC value and positive. This generates mixed results as modifies the way that the models are grouped.
- Compute the BIC value only for clusters which have changed considerably between iterations, maintaining the same value as previous iterations for those with almost the same segments assigned. This also resulted in mixed results, depending on the shows evaluated.
- Do not compute the BIC value for clusters pairs that obtain a negative value at any given iteration. This reduces the computation as only positive BIC values will be considered in subsequent iterations. It was implemented with success in the broadcast news system.

### 3.1.4 Stopping Criterion and System Output

Introduced in the system presented for the RT04f evaluation, after each cluster pair merge a set of three iterations of models training and Viterbi segmentation of the data given the models is performed. This achieved a small improvement on the RT04f evaluation data but proved to be positive and to increase robustness in the system.

After any cluster pair merges the cluster structure changes, with one less cluster in the system. When performing a Viterbi segmentation many segment boundaries change and some segments are reassigned to different clusters. Such new clusters are used to retain the models which are used again to segment the data. After three iterations the segmentation has usually converged and a new merging step is started.

In order to stop the clustering processing at the optimum number of clusters, two different alternatives were proposed as stopping criterion in the RT04f evaluation system. On one hand, the clustering can be performed while there is any positive  $\Delta$ BIC distance between any two clusters, it is called the BIC stopping criterion. On the other hand, the overall likelihood of the data given all the acoustic models can be compared between iterations and stop the processing when it starts decreasing (and revert to the previous segmentation), it is called Viterbi stopping criterion.

It must be noted that the Viterbi criterion can be applied only because the overall system complexity remains constant between iterations and therefore overall likelihoods are comparable. Note also that the Viterbi stopping criterion is in fact the BIC criterion applied over the overall model, comparing a model with  $M$  clusters and a model with  $M-1$  clusters and stopping when  $M-1$  is better than  $M$ .

Table 3.3 shows the resulting scores when using either BIC or Viterbi stopping criterions on the RT04f dataset. Although Viterbi stopping criterion achieves an absolute 1.55% improvement over BIC, the breakout by shows indicated mixed results and overall results are the opposite for

other sets.

| Criterion | %MISS | %FA | %SPKR | %DER  |
|-----------|-------|-----|-------|-------|
| BIC       | 1.5   | 1.2 | 15.1  | 17.91 |
| Viterbi   | 1.5   | 1.2 | 13.6  | 16.36 |

Table 3.3: Comparison between BIC and Viterbi stopping criterions for RT04f data

The final system for broadcast news uses a BIC stopping criterion as it does not require an extra clustering iteration for the stopping point to be found.

Once the system stops merging, the segmentation is output into a file. At this stage all removed non-speech regions are taken into account and inserted into the output where appropriate so that the output file is synchronous with the reference file used to evaluate its performance.

## 3.2 Analysis of Differences from Broadcast News to Meetings

Taking the broadcast news speaker diarization system as a backbone to build a meetings diarization system requires the adaptation of the previous system to the new requirements. While doing so, it was considered important to keep as much as possible the same structure and to make all the changes adaptable to obtain a system that could run with a correct performance in both broadcast news and meeting domains.

In this section, first an analysis is performed of some of the parameters that might differ from meetings to broadcast news, by looking at the input signals and reference segmentation files. This is intended to be a practical comparison between the two domains in order to identify the strengths and weaknesses of each one. Then, a more theoretical comparison is proposed between both domains and some high level changes are proposed to adapt the original broadcast news system to the meeting domain.

### 3.2.1 Input Data Analysis: Broadcast News versus Meetings

In this section some parameters are computed both in meetings and broadcast news shows in order to draw some conclusions on the nature of the input data to the speaker diarization system. In order to constraint the analysis to a known set of data, it has been performed on the RT04f broadcast news evaluation set and on the RT06s meetings evaluation set.

The RT04f set is composed of 12 shows, both from radio and television programs. The evaluation region in each of the shows is approximately 40 minutes, although the recording might be longer. The RT06s set is composed of two subsets, for the lecture data and conference data subdomains. The conference room data is composed of 8 meeting excerpts, with a length of

around 15 minutes each. The Lecture room set is composed of 28 lecture excerpts with varying times.

### Signal to Noise Ratio

The first parameter obtained from the input signals is the signal to noise ratio (SNR). It was computed using the *stnr* tool from the NIST Speech Quality Assurance Package (SPQA) (*NIST Speech tools and APIs 2006*) which is also used in the acoustic beamforming system evaluated in the experiments section. This program estimates the SNR of a file, defined as

$$10\log\frac{\text{peak\_speech\_power}}{\text{mean\_noise\_power}}$$

where power refers to the Root Mean Square (RMS) of the signal over a sliding window of 20ms, with a scrolling size of 10ms. A histogram is created using the RMS values and then the noise and speech values are computed.

To determine the noise average power, a raised cosine function is fitted to the peak in the left hand side of the histogram (lowest values) using a search algorithm to minimize the Chi-Square distance between the histogram and the function. The midpoint of such function is considered the mean noise power. Then the obtained raised cosine is subtracted from the histogram in order to estimate the speech power distribution. The peak speech power is defined as the histogram bin midpoint where the 95% of the power falls below it. Given that the speech power contains additive noise, the computed noise power is subtracted from it to use it in the SNR formula.

As speech and noise do not exist independently in the recorded signal, this method is only an approximation of the SNR. The result from this tool might not be comparable to the result from other tools, but according to the authors it is consistent to results using the same tool and therefore adequate to compare the quality difference of several signals as it is intended in this section. It must be noted that for a few cases this algorithm is known to give erroneous results, therefore it should be taken as an information source and the average should be taken to avoid misreadings.

In order to compute the SNR values, both for the meetings and for the broadcast news recording, only the regions determined to be part of the evaluation were considered. As pointed out before, some of the recordings contain more acoustic data than the evaluated region, which sometimes is excluded due to problems with the microphones (in meetings) or because it contains commercials or very noisy acoustics (in broadcast news).

First of all, the SNR is computed for the files in the RT04f data set. As it can be seen in table 3.4 the speech peak power remains constant at a very high value, with an average of 65db, while the noise average power is very variable and ranges from around 15db to around 62db.

Such averages are taken over the SNR values in log domain, and aim at indicating the overall quality of the dataset.

Some of the shows contain news material where reporters give their chronicles from the field, with a high level of background noise, while the conductor is in the studio, with a very good quality microphone in a controlled environment. The shows that contain less or none of the field recordings achieve a very good SNR (around 50db) while others perform very poorly (for example the CNN headline news, ABC and CNBC shows).

| Show_ID                   | Speech Peak Power | Noise ave. Power | SNR          |
|---------------------------|-------------------|------------------|--------------|
| 20031202.050216_CNN_ENG   | 66.5              | 14.75            | 51.75        |
| 20031202.203013_CNBC_ENG  | 60                | 57.75            | 2.25         |
| 20031203.183814_ABC_ENG   | 66.75             | 60.75            | 6            |
| 20031204.130035_CNN_ENG   | 67.25             | 53.75            | 13.5         |
| 20031206.163852_CSPAN_ENG | 61                | 30.5             | 30.5         |
| 20031209.193152_ABC_ENG   | 65                | 61.25            | 3.75         |
| 20031209.193946_PBS_ENG   | 63.25             | 20.75            | 42.5         |
| 20031215.204057_CNNHL_ENG | 68.25             | 61.75            | 6.5          |
| 20031215.231058_WBN_ENG   | 67                | 16.5             | 50.5         |
| 20031217.184122_ABC_ENG   | 66.75             | 22.75            | 44           |
| 20031218.004126_PBS_ENG   | 65                | 55.5             | 9.5          |
| 20031219.202502_CNBC_ENG  | 62                | 32               | 30           |
| average                   | <b>64.89</b>      | <b>40.66</b>     | <b>24.22</b> |

Table 3.4: *Estimated SNR for RT04f Broadcast News shows*

In the meetings domain the SNR is computed separately for the conference and lecture room sets. On the conference room set each of the rooms contains a variable number of microphones, mostly separated into 2 groups: the microphones situated in the middle of the table (labelled MDM) and the head-mounted microphones, worn by some of the participants (labelled IHM). Although the speaker diarization system presented in this thesis does not analyze the IHM case, the SNR for these microphones is also computed for comparison purposes.

Tables 3.5 and 3.6 show average SNR for the MDM and IHM channels in the RT06s meetings in the conference room. In both tables the number of microphones available is indicated in the second column. Then, the third through fifth columns indicate the average (in the linear domain) of the SNR values for all channels in each meeting. As the variety of microphones causes them to have very diverse quality levels, the last two columns indicate the maximum and minimum SNR values to give an idea of how disperse these are. Finally, the averages (in the log domain, as done in the broadcast news results) are computed for all meetings.

The speech quality for all cases is approximately the same (around 65db). The noise level for the MDM channels is much higher than the Broadcast news channels, which causes a decrease in SNR of almost 5db. The Average noise level is lower for the IHM channels than for the MDM

or the broadcast news shows which leads to an overall better SNR. This is due to the proximity of these microphones to the speakers and that a meeting room contains less noise than some broadcast news shows. It is interesting to point out the outstanding quality of the IHM channels used in the Edimburgh recordings (within the AMI project), but at the same time these meetings have some of the worse quality MDM microphones.

In overall, the MDM channels in the conference room are of less quality than the average in the broadcast news, but they remain more constant in quality across meetings.

| Show_ID            | # channels | Ave. Speech Power | Ave. Noise Power | Ave. SNR     | Min. SNR | Max. SNR |
|--------------------|------------|-------------------|------------------|--------------|----------|----------|
| CMU_20050912-0900  | 2          | 67.95             | 49.22            | 18.52        | 18       | 18.75    |
| CMU_20050914-0900  | 2          | 61.72             | 47.56            | 14.02        | 13.5     | 14.25    |
| EDI_20050216-1051  | 16         | 56.06             | 39.45            | 18.30        | 7.25     | 19.5     |
| EDI_20050218-0900  | 16         | 60.56             | 41.88            | 18.43        | 16.5     | 19       |
| NIST_20051024-0930 | 7          | 73.07             | 56.41            | 25.15        | 15.75    | 26       |
| NIST_20051102-1323 | 7          | 72.02             | 55.48            | 19.47        | 15.5     | 20.25    |
| VT_20050623-1400   | 4          | 59.72             | 36.51            | 26.96        | 21.25    | 27.5     |
| VT_20051027-1400   | 4          | 63.64             | 37.89            | 25.34        | 22.25    | 25.75    |
| average            |            | <b>64.34</b>      | <b>45.55</b>     | <b>20.77</b> | –        | –        |

Table 3.5: *Estimated SNR for RT06s Conference Meetings, MDM channels*

| Show_ID            | # channels | Ave. Speech Power | Ave. Noise Power | Ave. SNR     | Min. SNR | Max. SNR |
|--------------------|------------|-------------------|------------------|--------------|----------|----------|
| CMU_20050912-0900  | 4          | 70.14             | 42.39            | 40.39        | 16       | 41       |
| CMU_20050914-0900  | 4          | 72.89             | 40.89            | 46.39        | 32       | 47       |
| EDI_20050216-1051  | 4          | 57.39             | 13.89            | 63.14        | 43.5     | 63.75    |
| EDI_20050218-0900  | 4          | 60.89             | 23.39            | 61.39        | 33.5     | 62       |
| NIST_20051024-0930 | 9          | 70.59             | 41.36            | 38.79        | 18.5     | 39.76    |
| NIST_20051102-1323 | 8          | 68.91             | 41.79            | 39.59        | 16       | 40.5     |
| VT_20050623-1400   | 5          | 68.24             | 42.30            | 36.06        | 25.5     | 36.75    |
| VT_20051027-1400   | 4          | 58.19             | 25.40            | 41.12        | 31       | 41.75    |
| average            |            | <b>65.91</b>      | <b>33.93</b>     | <b>45.86</b> | –        | –        |

Table 3.6: *Estimated SNR for RT06s Conference Meetings, IHM channels*

Finally, table 3.7 shows the computed averages for the RT06s meetings in the lecture room dataset. In the same way as in 3.5, for each recording several distant microphones are available. The Average between microphones is done in the linear domain while the average over all recordings is done in the log domain.

Although all meeting recordings were done within the CHIL project, the specifications on the room layout and on the acoustic environment change within each lecture room. The speech average peak power changes immensely among the rooms (from around 50db on AIT recordings to around 80db on UKA recordings) remaining stable within the same lecture room. The same

| Show_ID                 | # channels | Ave. Speech Power | Ave. Noise Power | Ave. SNR     | Min. SNR | Max. SNR |
|-------------------------|------------|-------------------|------------------|--------------|----------|----------|
| AIT_20051010/Segment1   | 4          | 51.15             | 23.14            | 27.90        | 24.25    | 28.5     |
| AIT_20051011_B/Segment1 | 4          | 51.65             | 22.30            | 28.91        | 25.5     | 29.5     |
| AIT_20051011_C/Segment1 | 4          | 48.22             | 22.81            | 26.66        | 22.25    | 27.25    |
| AIT_20051011_D/Segment1 | 4          | 49.90             | 22.64            | 27.16        | 24       | 27.75    |
| IBM_20050819/Segment1   | 3          | 58.84             | 46.75            | 12.09        | 9.75     | 12.5     |
| IBM_20050822/Segment1   | 3          | 61.77             | 47.57            | 13.77        | 10       | 14.25    |
| IBM_20050823/Segment1   | 3          | 59.03             | 48.02            | 13.02        | 9        | 13.5     |
| IBM_20050830/Segment1   | 3          | 60.59             | 47.41            | 13.30        | 11       | 13.75    |
| ITC_20050503/Segment1   | 5          | 59.25             | 42.78            | 16.85        | 14.75    | 17.5     |
| ITC_20050607/Segment1   | 5          | 60.54             | 45.28            | 36.30        | 15       | 37       |
| UKA_20050420_A/Segment1 | 5          | 77.55             | 58.55            | 22.30        | 13.25    | 23       |
| UKA_20050420_A/Segment2 | 5          | 77.55             | 58.55            | 20.70        | 14       | 21.25    |
| UKA_20050427_B/Segment1 | 5          | 81.30             | 64.05            | 18.30        | 13       | 19       |
| UKA_20050504_A/Segment1 | 5          | 82.80             | 67.30            | 17.25        | 14.5     | 17.75    |
| UKA_20050504_B/Segment1 | 5          | 82.80             | 69.05            | 13.63        | 13       | 14       |
| UKA_20050504_B/Segment2 | 5          | 81.55             | 67.30            | 14.83        | 12.25    | 15.5     |
| UKA_20050525_A/Segment1 | 5          | 82.55             | 67.30            | 15.49        | 6.25     | 16       |
| UKA_20050525_A/Segment2 | 5          | 82.80             | 67.80            | 14.84        | 8.75     | 15.25    |
| UKA_20050525_B/Segment1 | 5          | 78.05             | 67.80            | 10.07        | 4.25     | 10.5     |
| UKA_20050525_C/Segment1 | 5          | 82.55             | 68.80            | 14.24        | -2.5     | 14.5     |
| UKA_20050615_A/Segment1 | 5          | 79.30             | 66.30            | 13.60        | 12.25    | 14.25    |
| UKA_20050622_B/Segment1 | 5          | 67.64             | 59.33            | 11.30        | 8.25     | 12.0     |
| UKA_20050622_C/Segment1 | 5          | 74.07             | 60.35            | 16.55        | 13.0     | 17.25    |
| UKA_20050622_C/Segment2 | 5          | 75.06             | 61.31            | 15.31        | 12.25    | 16       |
| UPC_20050706/Segment1   | 4          | 79.66             | 51.65            | 31.89        | 27.5     | 32.5     |
| UPC_20050720/Segment1   | 4          | 71.15             | 56.40            | 16.65        | 6.25     | 17.25    |
| UPC_20050722/Segment1   | 4          | 71.15             | 53.90            | 17.07        | 16.75    | 17.25    |
| UPC_20050727/Segment1   | 4          | 67.90             | 50.42            | 17.18        | 16.75    | 17.5     |
| average                 |            | <b>69.87</b>      | <b>53.03</b>     | <b>18.47</b> | –        | –        |

Table 3.7: *Estimated SNR for RT06s Lecture Room Meetings, MDM channels*

thing happens with the noise average power, which is the lowest for the AIT recordings and the highest for the UKA. This indicates that the recording settings were not set equally for all settings, being such difference possibly due solely to the amplification applied to the signal by the recording equipment.

Regarding the SNR over all the channels, the AIT recordings are constantly achieving SNR values on the twenties, while the other shows are usually on the tens, with a global average of 18.47, which is slightly lower in average than the meetings in the conference room subdomain. The differences between minimum and maximum SNR values remain in the same line as in 3.5.

### Average Total Speaking Time

Apart from looking at the acoustic signal, the reference transcriptions were also analyzed. The first parameter computed both on the meetings and broadcast news is the speaking time per speaker in each of the shows. This is important as it is necessary to create models that can train optimally to the data, and therefore need to be adjusted if the amount of data per speaker changes across domains. Tables 3.8, 3.9 and 3.10 show the number of speakers, average time per speaker and maximum and minimum speaking times.

| Show_ID            | #<br>spkr. | ave. time | max. time | min. time | ave. time | max. time | min. time |
|--------------------|------------|-----------|-----------|-----------|-----------|-----------|-----------|
|                    |            | manual    | manual    | manual    | FA        | FA        | FA        |
| CMU_20050912-0900  | 4          | 373.01    | 620.49    | 169.77    | 283.71    | 487.14    | 105.96    |
| CMU_20050914-0900  | 4          | 368.04    | 544.88    | 131.66    | 277.61    | 444.69    | 102.46    |
| EDL_20050216-1051  | 4          | 301.65    | 432.01    | 184.03    | 224.01    | 306.88    | 111.21    |
| EDL_20050218-0900  | 4          | 318.01    | 489.42    | 210.46    | 238.46    | 361.15    | 162.5     |
| NIST_20051024-0930 | 9          | 182.45    | 503.49    | 32.83     | 118.46    | 384.29    | 1.15      |
| NIST_20051102-1323 | 8          | 179.25    | 336.05    | 46.04     | 121.47    | 257.0     | 2.76      |
| VT_20050623-1400   | 5          | 258.16    | 438.16    | 155.11    | 195.60    | 342.06    | 118.84    |
| VT_20051027-1400   | 4          | 244.27    | 581.27    | 103.53    | 184.21    | 457.39    | 70.19     |

Table 3.8: Average total speaker duration in RT06s conference room data

In all cases the average values vary greatly even within the same domain. For example, the CSPAN show in broadcast news contains an average speaker length several orders of magnitude higher than any of the other shows. This is due to the rather common total show lengths imposed by NIST for the evaluations but the variability in the number of speakers existent in each recording. From these results it is clear that an automatic way of selecting the speaker models complexity is necessary in order to be able to model each of the possibilities correctly, as the more data available from a speaker, the more complex the models need to be to be able to represent the same level of detail for that speaker compared to others.

Another observation is on the minimum and maximum speaking times columns. The maximum speaking time indicates how long has the main speaker in the recording spoken. In both the lecture room and broadcast news recordings this column tends to contain values very much higher than the average speaking time. In lectures it is the case when the excerpt mainly contains the lecturer giving his talk (sometimes filling the entire excerpt and sometimes with a small question and answers section). In the broadcast news shows it is usual when the show contains an anchor speaker that directs the flow of the program. In the conference room meetings the NIST shows also tend to have a dominant speaker.

The minimum speaking time column indicates the length of time that the speaker with less interventions speaks. In many of the lecture room meetings this is nonexistent as the lecturer speaks for the whole time. In the other cases, many of the recordings in lectures and broadcast

| Show_ID                 | # speakers | Ave. time | max. time | min. time |
|-------------------------|------------|-----------|-----------|-----------|
| AIT_20051010/Segment1   | 4          | 52.68     | 202.06    | 2.00      |
| AIT_20051011_B/Segment1 | 4          | 65.01     | 233.91    | 0.66      |
| AIT_20051011_C/Segment1 | 4          | 57.94     | 160.63    | 5.67      |
| AIT_20051011_D/Segment1 | 5          | 52.80     | 228.07    | 0.48      |
| IBM_20050819/Segment1   | 4          | 63.07     | 156.09    | 28.83     |
| IBM_20050822/Segment1   | 2          | 127.47    | 253.48    | 1.46      |
| IBM_20050823/Segment1   | 4          | 70.86     | 172.02    | 30.65     |
| IBM_20050830/Segment1   | 1          | 251.27    | –         | –         |
| ITC_20050503/Segment1   | 4          | 66.81     | 234.06    | 6.16      |
| ITC_20050607/Segment1   | 4          | 64.64     | 207.49    | 2.82      |
| UKA_20050420_A/Segment1 | 3          | 80.53     | 228.30    | 3.69      |
| UKA_20050420_A/Segment2 | 2          | 121       | 178.31    | 63.68     |
| UKA_20050427_B/Segment1 | 1          | 157.51    | –         | –         |
| UKA_20050504_A/Segment1 | 1          | 219.16    | –         | –         |
| UKA_20050504_B/Segment1 | 1          | 253.86    | –         | –         |
| UKA_20050504_B/Segment2 | 3          | 86.74     | 222.55    | 6.13      |
| UKA_20050525_A/Segment1 | 1          | 240.71    | –         | –         |
| UKA_20050525_A/Segment2 | 4          | 60.09     | 153.77    | 3.53      |
| UKA_20050525_B/Segment1 | 2          | 114.11    | 226.36    | 1.87      |
| UKA_20050525_C/Segment1 | 2          | 119.76    | 238.52    | 1.00      |
| UKA_20050615_A/Segment1 | 3          | 55.28     | 147.78    | 6.00      |
| UKA_20050622_B/Segment1 | 1          | 211.22    | –         | –         |
| UKA_20050622_C/Segment1 | 3          | 80.38     | 214.86    | 8.38      |
| UKA_20050622_C/Segment2 | 1          | 230.59    | –         | –         |
| UPC_20050706/Segment1   | 5          | 43.91     | 182.58    | 5.96      |
| UPC_20050720/Segment1   | 5          | 56.16     | 255.72    | 3.17      |
| UPC_20050722/Segment1   | 5          | 57.71     | 234.39    | 2.88      |
| UPC_20050727/Segment1   | 5          | 87.78     | 139.46    | 62.73     |

Table 3.9: Average total speaker duration in RT06s lecture room data

news, and the NIST recordings in conference room, contain very short durations. These speakers are difficult to model as not much data is available and could create many problems and errors when comparing their models with the longer speaking ones. This is why it is sometimes desirable to talk of agglomerative clustering systems (like the one presented in this thesis) as having the goal of obtaining the optimum number of final clusters, instead of the exact number of existing speakers. Although detecting these short speakers and labelling them as independent clusters is always desirable, it can normally lead to other errors and therefore should be considered as a secondary priority.

In Table 3.8 two different transcriptions were used to compute these parameters. On one hand, one set of transcriptions were generated by hand, distributed by NIST and used in the evaluations. On the other hand, another set of reference transcriptions were generated automatically via forced-alignment of the reference speech-to-text transcriptions to the IHM channels.

| Show_ID                   | # speakers | Ave. time | max. time | min. time |
|---------------------------|------------|-----------|-----------|-----------|
| 20031202_050216_CNN_ENG   | 15         | 65.92     | 462.31    | 5.7       |
| 20031202_203013_CNBC_ENG  | 13         | 69.41     | 322.5     | 1.07      |
| 20031203_183814_ABC_ENG   | 28         | 38.67     | 196.86    | 2.06      |
| 20031204_130035_CNN_ENG   | 13         | 99.79     | 479.12    | 13.73     |
| 20031206_163852_CSPAN_ENG | 4          | 345.62    | 941.21    | 7.76      |
| 20031209_193152_ABC_ENG   | 30         | 33.84     | 292.49    | 1.26      |
| 20031209_193946_PBS_ENG   | 15         | 86.52     | 349.40    | 0.47      |
| 20031215_204057_CNNHL_ENG | 10         | 110.62    | 312.53    | 8.15      |
| 20031215_231058_WBN_ENG   | 29         | 39.52     | 536.04    | 0.44      |
| 20031217_184122_ABC_ENG   | 25         | 42.56     | 181.38    | 1.87      |
| 20031218_004126_PBS_ENG   | 27         | 47.52     | 353.56    | 0.4       |
| 20031219_202502_CNBC_ENG  | 25         | 38.48     | 351.59    | 1.10      |

Table 3.10: Average total speaker duration in RT04f broadcast news data

The forced-alignments are the ones used in this thesis in the experiments section. For a more detailed description on the differences and motivation behind the forced-aligned transcriptions refer to the experiments chapter 6.

### Average Number of Speakers

Another parameter that describes the different subdomains of application is the number of expected speakers to be clustered. Given that the number of initial clusters needs to be higher than the optimum number of clusters, it is important to define an upper boundary on the number of speakers so that systems are ensured to be able to reach the optimum point. Although an optimal speaker diarization system using hierarchical agglomerative clustering should be able to start at a very high number of clusters and work its way down, in reality it makes a difference in the resulting performance the correct estimation of an appropriate upper limit for the number of clusters. This is explained more in detail in section 4.2.2.

The average number of speakers and their minimum and maximum values are represented for the three datasets in table 3.11. One can observe how in general the broadcast news shows contain a vast amount of speakers (averaging 19), although in the shows considered there was one case (CSPAN) with 4 speakers. This creates a very big variation (or standard deviation) between the values. The system processing the broadcast news data needs to ensure a good performance both when many speakers are present (with smaller speaking time) and when less are available. Without any automatic initial number of speakers detection algorithm, the system starts at 40 clusters.

In the case of meetings, the lecture room data contains many recordings where only the lecturer speaks, and other with several people, going to a maximum number of four speakers. The standard deviation is therefore smaller compared to broadcast news. On conference rooms

the number of speakers range between 4 and 9, with an average of 5.25 speakers present. Without automatic detection the systems start at 10 or 16 clusters for meetings and at 5 or 10 for lecture room.

| Domain              | # excerpts | ave. spkrs | min. spkrs | max. spkrs | std spkrs |
|---------------------|------------|------------|------------|------------|-----------|
| Meetings conference | 8          | 5.25       | 4          | 9          | 2.05      |
| Meetings lecture    | 28         | 3          | 1          | 5          | 1.49      |
| Broadcast news      | 12         | 19         | 4          | 29         | 8.35      |

Table 3.11: Average number of speaker for *Rt04f* and *RT06s*

### Average Speaker Turn Duration

Given the HMM acoustic modeling presented in the previous section, a minimum duration is applied to the segment length when performing the acoustic decoding of the data using the Viterbi algorithm. Such segments constitute the speaker turns and therefore it is important to analyze how long in average these are in the different subdomains, in order to adapt (if necessary) this parameter of the system to allow for smaller/larger turns.

In Table 3.12 the average, maximum and standard deviation of the speaker turn duration is given both for conference room meetings data and for the lecture room data. In the case of the conference room data, both the manual transcriptions and the forced-alignments are analyzed. When analyzing this property, two speaker turns from the same speaker but separated with a silence are considered different turns, and their durations counted separately.

| Domain                               | Average duration | max. duration | std duration |
|--------------------------------------|------------------|---------------|--------------|
| Meetings conference hand alignment   | 3.96             | 85.92         | 5.74         |
| Meetings conference forced alignment | 1.89             | 14.64         | 1.90         |
| Broadcast news                       | 8.61             | 107.54        | 10.08        |

Table 3.12: Speaker turn durations for *RT04f* and *RT06s*

It is clear that in the lecture room data the speaker turns are in average of much greater length, given that many times a lecturer speaker for elongated amounts of time. In some cases though it was seen that the transcriptions contained errors when small silence segments needed to be transcribed as such and were included within the speaker segment adjacent to it. The maximum speaker turn length for these is of 1:45 minutes approx.

On the conference room data there is a difference between the two transcriptions sources. This is mainly due to the discrepancies on the transcription of small silences. According to NIST rules for the evaluations, any silence segment of length greater than 0.3 seconds needs to be considered as such. This can be implemented efficiently in the forced-alignment transcriptions but it is more difficult to be followed by the human transcribers, leading to longer segments being annotated.

To further illustrate the distribution of the speaker turn durations, Figure 3.3 shows the histogram of all three analyzed cases, showing the durations histogram of the first 10s, with a resolution of 0.1 second. It can be observed that both transcriptions for the meetings conference room recordings have a similar shape, being very pointy around 0.3s, while the broadcast news shows reflect a broader shape. Such a small peak duration in the conference data has a reason to be in that overlap segments were also used when computing the speaker turn duration. These segments occur when two or more people are speaking at the same time and can just refer to people uttering affirmative/negative responses or short sentences.

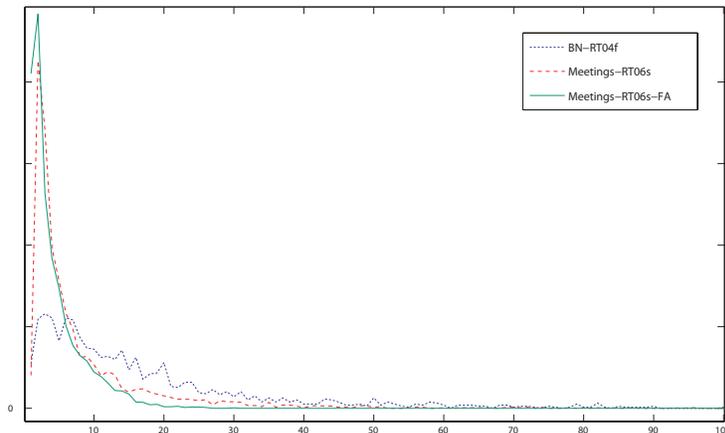


Figure 3.3: *Speaker turn duration histograms*

In overall, the speaker turn length of meetings is much smaller than the average in broadcast news, which is modeled by the system with a minimum duration of 3 seconds. Such duration would be enough if the meetings system was evaluated using the hand-alignments, but needs to be reduced when evaluated with forced-alignments.

### Meetings Domain Overlap Regions

Given the analysis performed in the previous subsection, it was found interesting to look at the overlap regions in more detail. These are found with less frequency in the broadcast news data and it only started being evaluated with the start of the meetings domain evaluations. Nowadays overlap is considered an important feature of the meeting data and therefore is included in the main metric in the NIST RT evaluations. An analysis of the overlap was performed for both forced-alignments and hand-alignments in the conference room data, and it is shown in table 3.13. In it, the average, maximum and standard deviation segment length is computed for the overlap regions alone and for the regions without any overlap.

From the average duration of the overlap regions one can see how much difference in average

| type             | average overlap | max length overlap | std. overlap | average non-ovl | max. length non-ovl | std. non-ovl |
|------------------|-----------------|--------------------|--------------|-----------------|---------------------|--------------|
| hand alignment   | 1.19            | 10.97              | 1.032        | 2.77            | 74.10               | 4.22         |
| forced alignment | 0.54            | 6.23               | 0.468        | 1.52            | 14.63               | 1.71         |

Table 3.13: *Overlap analysis between hand and forced alignments in RT06s conference room meetings*

length there is between both transcription sources. The hand alignments are double the length than the forced-alignments in average, probably due to the difference in how the transcriptions are created. A human transcriber upon listening to an overlap region might have labelled it grossly, allowing for a few extra milliseconds in either side. The forced-alignments are based on the uttered words, which are tightly aligned by the ASR system. One drawback of the forced-alignments on overlap regions comes when the transcribers that wrote down the words miss the words or sounds existing in the overlaps, and therefore the transcription is not aligned correctly. Finally, on the overlap results, note that the values on the hand-alignments have a much bigger standard deviation than the automatically generate ones.

Regarding the analysis of values in the regions without any overlap, the same observation as in the previous subsection can be made. The average length of the speaker turns is bigger in the hand-alignments, probably due to the consistent miss of small silence regions, shown also by the values of the maximum segment lengths.

To further analyze the duration of the overlaps, in figure 3.4 the histograms of the lengths of the overlap segments in both forced-alignments and hand-alignments is shown. As hinted by the averages, the peak of the forced-alignment overlaps falls around 0.5 seconds, while the peak of the hand-alignments is around 1 second and has a broader range of bigger values than the forced-alignments.

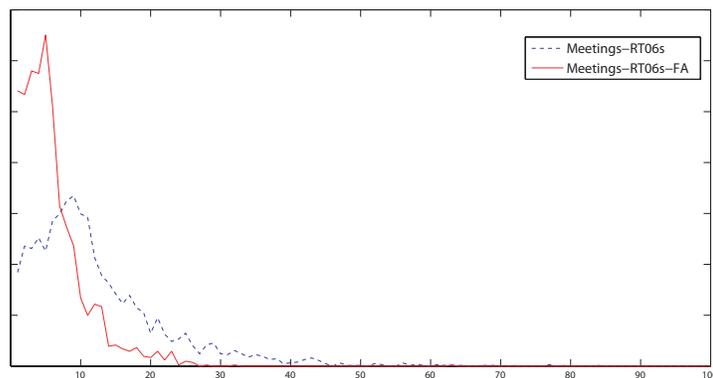


Figure 3.4: *Overlap histograms in RT06s conference room meetings*

### 3.2.2 Summary of Differences and Proposed Changes

In a more theoretical point of view there are many other differences between meetings and broadcast news that need the system builder's attention when converting a system to the other domain. In table 3.14 some of these differences are pointed out (some of them already studied in this section) and in some cases a proposed solution, as described in this thesis, is given.

| Meetings Environment   | Broadcast News Environment  | Proposed solution                                      |
|--|---|--|
| Reduced amount of speakers, limited by the capacity of the room, but unknown                             | Totally unknown amount of speakers  | Automatic number of initial clusters estimation        |
| There are neither music or commercials   | There can be commercials and background music with speech                       | Changed speech/non-speech detector                     |
| There are impulsive noises (doors shut down, pens fall, speakers touch their mics...)                    | Different background conditions occur when reporting from the field             | Changed speech/nonspeech detector                      |
| All recordings take place in the same setting (there could be people call into the meeting with a phone) | Recordings alternate between studio and field (different bandwidth conditions). |  |
| Different meetings can take place in different settings (rooms, microphones positions/number,...)        | Recordings for the same program take place in the same studio.                  | Acoustic beamforming without layout constraints        |
| Major use of spontaneous speech, with more silences and filling words/sounds                             | Much more scripted speech with professional narrators.                          | Frame and segment purification algorithms              |
| The average speaker turn can be very small (for example yes/no answers)                                  | The average speaker turn is longer  | Reduced minimum duration in decoding                   |
| Normal existence of overlapping regions where two or more people speak at the same time                  | Normally there is no (or very little) overlapping speech                        |  |
| The recordings are performed using several microphones   | Only one channel is available   | Acoustic beamforming to collapse all channels into one |
| The far-field channels (microphones in the meeting table) regularly have worse quality than closer mics  | The speech quality is the regular broadcasting quality.                         | Acoustic beamforming tries to enhance the signal       |

Table 3.14: *Main differences between Meetings and Broadcast News recordings*

### 3.3 Robust Speaker Diarization System for Meetings

The proposed speaker diarization system for meetings can be broken up into three/four main blocks, as shown in figure 3.5. These are the single channel Signal-to-Noise Ratio (SNR) enhancement block, the multi-channel acoustic fusion and the segmentation and clustering system, which itself could be broken up into the speech/non-speech detection and the single-channel speaker diarization system.

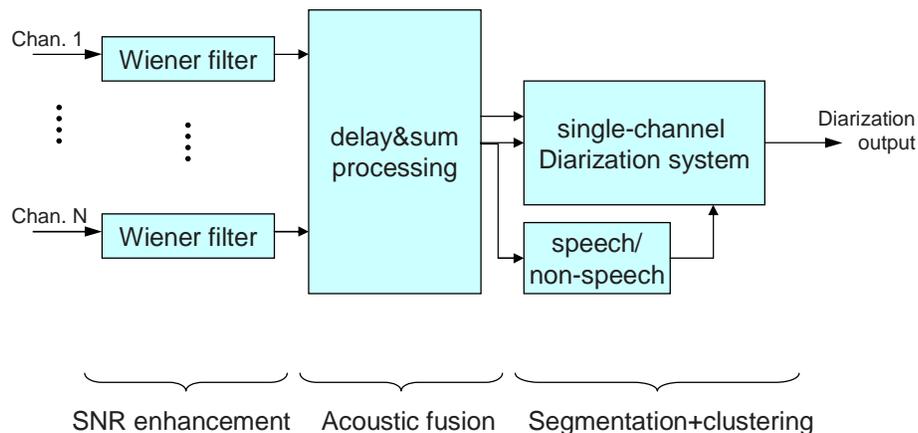


Figure 3.5: Main blocks involved in the meetings speaker diarization system

When only one channel is to be processed, the acoustic fusion block is bypassed and the individual, SNR enhanced, signal is processed directly by the segmentation and clustering blocks. The advantage of this architecture is obvious, there is no need to generate a different system or techniques depending on the number of microphones available as both the acoustic fusion and the segmentation and clustering blocks accept an acoustic channel as an input and can be simply turned on/off depending on the characteristics of the data. The following sections describe each of the blocks in more detail.

#### 3.3.1 Acoustic Signal Enhancement

Both the individual channel enhancement block and the acoustic fusion block aim at obtaining a signal with a better quality than the original in order to improve the performance of the diarization system.

The individual channels are first Wiener-filtered (Wiener and Norbert 1949) to improve the SNR with the same algorithm as in the ICSI-SRI-UW Meetings recognition system (Mirghafori, Stolcke, Wooters, Pirinen, Bulyko, Gelbart, Graciarena, Otterson, Peskin and Ostendorf 2004), which uses a noise reduction algorithm developed for the Aurora 2 front-end, proposed originally in Adami, Burget, Dupont, Garudadri, Grezl, Hermansky, Jain, Kajarekar, Morgan and

Sivadas (2002). The algorithm performs Wiener filtering with typical engineering modifications, such as a noise over-estimation factor, smoothing of the filter response, and a spectral flooring. The original algorithm was modified to use a single noise spectral estimate for each meeting waveform. This was calculated over all the frames judged to be non-speech by the voice-activity detection component within the module. As observed in Figure 3.5, the algorithm is independently applied to each meeting channel and uses overlap-add resynthesis to create noise-reduced output waveforms, which then are either fed into the acoustic fusion block (multi-channel) or directly into the segmentation and clustering block (single-channel).

The acoustic fusion module makes use of standard beamforming techniques in order to obtain an “enhanced” version of the signal as a combination of the multiple channel input signals. It considers the multiple channels to form a microphone array. Neither the microphone positions nor their acoustic properties are known. Given these constraints, a variation of the simple (yet effective) delay&sum beamforming technique is applied as it does not require any information from the microphones in order to operate. As the different microphones have a different acoustic directivity pattern and are located in places in the room where the noise level is different, a dynamic weighting of the individual channels and a triangular filtering is used to reduce its negative effects. By using such channel filtering the system will be referred to as filter&sum from now on.

The filter&sum beamforming technique involves estimating the relative time delay of arrival (TDOA) of the acoustic signal with respect to a reference channel. The GCC-PHAT (Generalized Cross Correlation with Phase Transform) is used to find the potential relative delays regarding each of the speakers in the meeting. In order to avoid impulsive noise, short-term events and overlap speech from tainting the correct approximation of the TDOA, multiple TDOA values are computed for each time and a double post-processing algorithm is implemented to select the most appropriate value. On one hand, noise is detected by measuring the quality of the computed cross-correlation values at each point with respect to the rest of the meeting and the computed TDOA values are substituted by the previous (more reliable) values when considered too low. On the other hand, impulsive events and overlap is dealt with by using a double-step Viterbi decoding of the delays in order to obtain the optimum set of TDOA values that are both reliable and stable. A more in depth explanation of these and other steps involved in the acoustic fusion block is given in chapter 5.

Apart from using the post-processed estimated delays for the filter&sum beamforming, they are also used in the segmentation and clustering block as they can convey information about the speaker through his/her location in the room. Such information is orthogonal to the acoustic information and therefore adds useful information to the diarization system. In section 5.3 the combination of both features is presented in detail.

### 3.3.2 Single Channel System Frontend

The speaker segmentation and clustering block of the overall speaker diarization system contains two main blocks, the speech/non-speech detector and the single-channel speaker diarization system. The speech/non-speech detector is different from the one used for broadcast news and does not require any training data for the acoustic models. It is a hybrid energy-based/model-based system which considers that most non-speech to be detected in a meeting, which can harm the diarization, is silence. It will be further described in 4.1.

The single-channel speaker diarization module has evolved from the broadcast news system by adding new algorithms and proposing improvements to existing ones. In figure 3.6 a block diagram of the diarization process is shown with the newly proposed algorithms and changes to the baseline system in a darker box. There are also other improvements in various steps of the algorithms that are not reflected in the figure, these are the modification in duration modeling within the models and the models initialization algorithm. In the following sections a description of each one of these new modules and algorithm improvements is described in detail. For those not changing from the baseline refer to section 3.1 for complete details.

As mentioned earlier, the meetings speaker diarization system makes use of the TDOA values (when available) as an independent feature stream. These features contain N-1 dimension vectors, where N is the number of channels available, computed at the same rate as the MFCC parameters for synchronous operation. They are reused without any further processing, just converting them to HTK format to be properly read by the system.

The acoustic features continue to be Mel Frequency Cepstrum Coefficients (MFCC) but with an analysis window of 30ms (instead of 60ms) and computed every 10ms (instead of 20ms). The increase in computation due to having double the amount of features is explained by an increase in performance.

### 3.3.3 Speaker Clusters and Models Initialization

In order to initialize the hierarchical bottom-up agglomerative clustering one needs to first define an initial number of clusters  $K_{init}$ , bigger than the optimum number of clusters  $K_{opt}$ . The system defined for broadcast news used  $K_{init} = 40$  clusters, value chosen empirically given some development data. It was found that even though the optimum number of clusters in a recording is independent of the length of such recording, in terms of selecting an initial number of clusters for the agglomerative system the total length of the available data has to be considered to allow for clusters to be well trained and best represent the speakers. By making the  $K_{init}$  constant for any kind of data used in the system makes some recordings do not perform as well since the initial models either contain too much or too few acoustic data. In the system

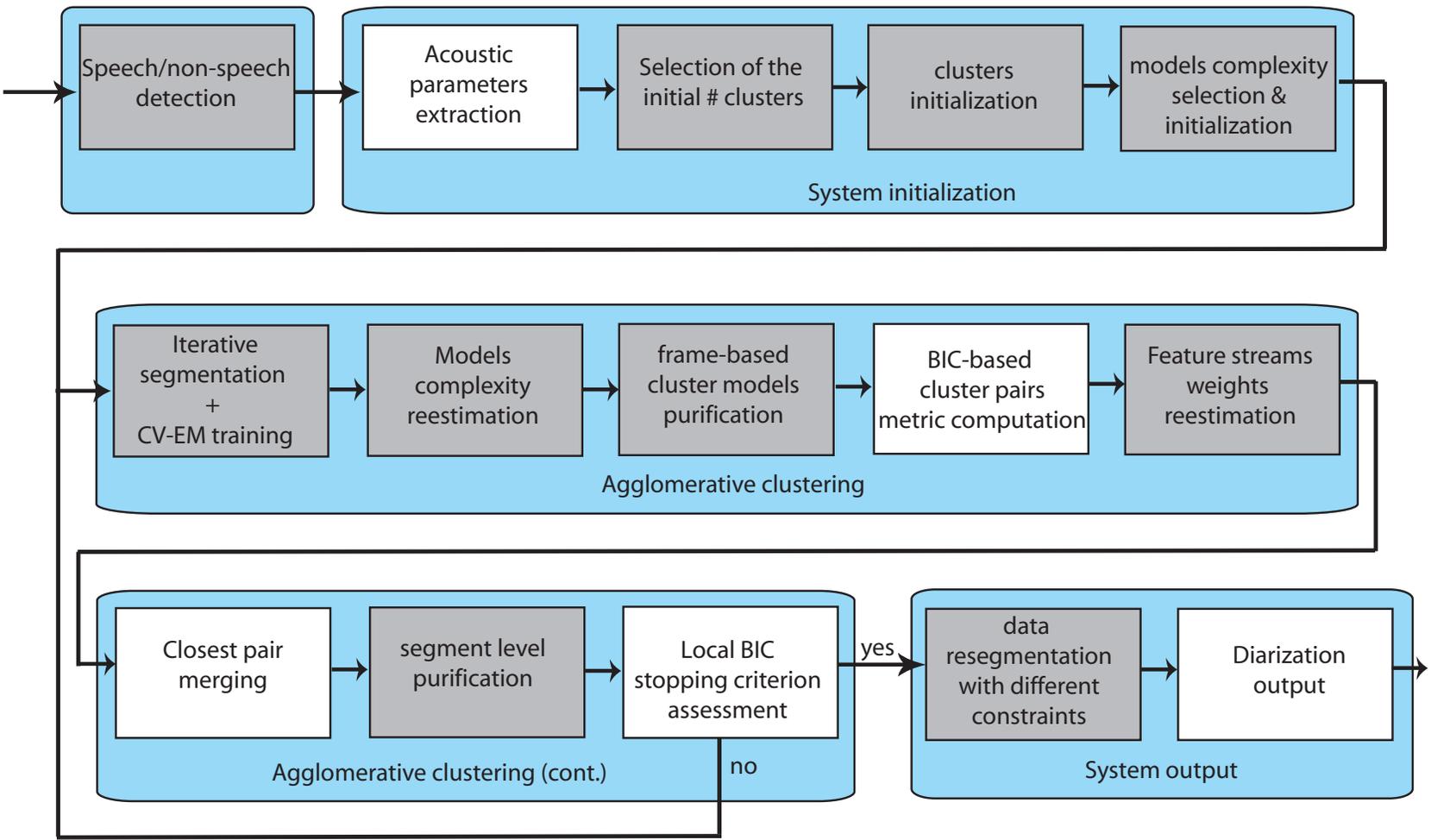


Figure 3.6: single-channel speaker diarization for meetings block diagram

presented here for meetings, this initial number is made dependent on the amount of data after the speech/non-speech detection. A new parameter called Cluster Complexity Ratio (CCR) represents the relationship between data and cluster complexity. The algorithm used is further described in detail in 4.2.2.

The same CCR parameter is also used throughout the agglomerative clustering process to determine the complexity (number of Gaussian mixtures) of the speaker models. Such mechanism ensures that all models remain at a complexity relative to the amount of data that they are trained with, and therefore remain comparable to each other. This is further explained in section 4.2.2.

Given the data assigned to each cluster, in order to obtain an initial GMM model with a certain complexity the technique used in the baseline system has been replaced by another one in order to obtain better initialized models. It was seen in experiments that the initial models play an important role in the overall performance of the system as the initial position for the mixtures is an important factor in how well the model can be trained using EM-ML and therefore how representative it will be of the data. This is particularly crucial in speaker diarization where small models (initially 5 Gaussians) are used due to little training data.

The broadcast news system uses a method that resembles the *HCompV* routine in the HTK toolkit (Young, Kershaw, Odell, Ollason, Valtchev and Woodland 2005) for initialization without a reference transcription. Given a set of acoustic vectors  $X = \{x[1] \dots x[N]\}$  and a desired GMM with complexity  $M$  Gaussians, the first Gaussian is computed via the sufficient statistics of the data  $X$  as

$$\mu_1 = \frac{1}{\text{size}(X)} \sum_{i=1}^N x[i]$$

$$\sigma_1^2 = \frac{1}{M} \left( \frac{1}{N} \sum_{i=1}^N x^2[i] - \mu_1^2 \right)$$

For the rest of the Gaussian mixtures, equidistant points in  $X$  are chosen as means and the same variance as in Gaussian 1 is used:

$$\mu_i = X \left[ i \cdot \frac{N}{M} \right]$$

$$\sigma_i^2 = \sigma_1^2$$

with Gaussian weights kept equal for all mixtures,  $W_i = \frac{1}{M}$ .

This method has two obvious drawbacks. On one hand, as pointed out above, this technique does not consider a global ML approach and therefore Gaussian mixtures can easily end up in

local maxima. On the other hand, it does not ensure that all the acoustic space of the acoustic data is covered by the positioned Gaussians.

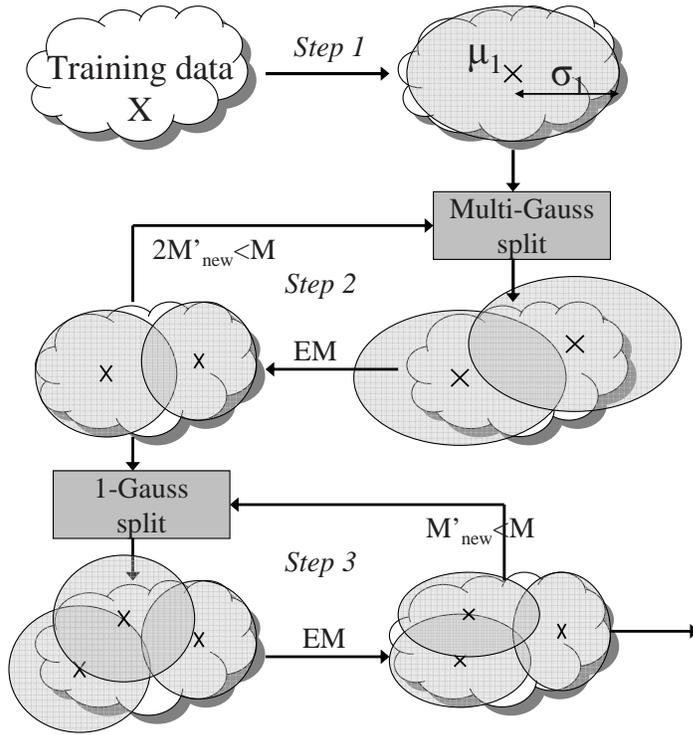


Figure 3.7: *Speaker models initialization based on Gaussian splitting*

The introduced technique is inspired on the split and vanish techniques used in the GMTK toolkit (Bilmes and Zweig 2002) and the mixture incrementing function in HTK. As seen in figure 3.7, the initial mean and variance of data  $X$  are computed in the same way as in the previous technique (step 1). Then the algorithm iteratively splits each of the  $M'_{prev}$  Gaussian mixtures into two mixtures, obtaining a total of  $M'_{new}$  mixtures, while  $2M'_{new} < M$ , the desired model complexity. The  $M'_{new}$  Gaussian mixtures are computed from their previous counterpart by

$$\sigma_{new1}^2 = \sigma_{new2}^2 = \sigma_{prev}^2$$

$$\mu_{new1} = \mu_{prev} + 0.2\sigma_{prev} \quad \mu_{new2} = \mu_{prev} - 0.2\sigma_{prev}$$

$$W_{new1} = W_{new2} = \frac{W_{prev}}{2}$$

After each split, a single step EM training of the current models given data  $X$  is performed to allow for the Gaussian mixtures to adapt mean and variance to the data.

Once an extra splitting iteration would overpass the desired number of desired Gaussian mixtures, the algorithm moves into a single Gaussian split mode (step 3). In it the Gaussian selected to split is the one with the highest weight, and it is split in the same way as shown before. Some experiments were performed with different alternative splitting/vanishing procedures but to initialize GMM models with a small number of Gaussian mixtures it was seen that performance would diminish any time that vanishing was applied, therefore the technique applied here only uses a splitting procedure. Also, the defunct function implemented by HTK to discard Gaussians with low weigh was seen to be perjudicial for the GMM models grown here.

Once the number of initial cluster  $K_{init}$  is defined, in the broadcast news system it was explained how speaker clusters were initialized by evenly assigning the available data into the different clusters and doing several segmentation-training iterations to allow for homogeneous data to cluster together. While this mechanism is very simple and gives surprisingly good results, it does not ensure that the final clusters contain only data from one cluster (i.e. with a high purity).

In order to improve on the linear initialization technique, several alternative methods were tested, including K-means at the segment level, E-HMM top-down clustering (Meignier et al. 2001) and others, finally designing a brand new algorithm that has been called the friends-and-enemies initialization and is further explained in section 4.2.1.

### Models Training Using CV-EM and Clusters Segmentation

In order to train the speaker models used throughout the processing a standard EM-ML algorithm was used by the broadcast news system. It performed a five iterations EM-ML algorithm regardless of the data or the models being trained. The use of EM in small training datasets has two potential problems. On one hand the models can suffer from overfitting to the available data, becoming not general enough to represent the speaker at hand. On the other hand there is no guarantee that the models will converge to the best possible parameters that maximize the likelihood of the data given such model. The use of  $k = 5$  iteration of EM training is a parameter that needs to be defined for the system in order to avoid overfitting but to allow the models to be correctly trained to the data. It was seen that modifying the value of the parameter  $k$  would considerably alter the final performance, and therefore it was found desirable to find a more robust algorithm.

For these reasons a new training algorithm has been implemented. The choice of implementation has been the cross-validation EM training algorithm (CV-EM for short), recently proposed by T. Shinozaki in Shinozaki and Ostendorf (2007). It introduces a cross-validation technique, in use for decision tree design, to the iterative process of the EM, addressing the problems of overfitting and potential local maxima.

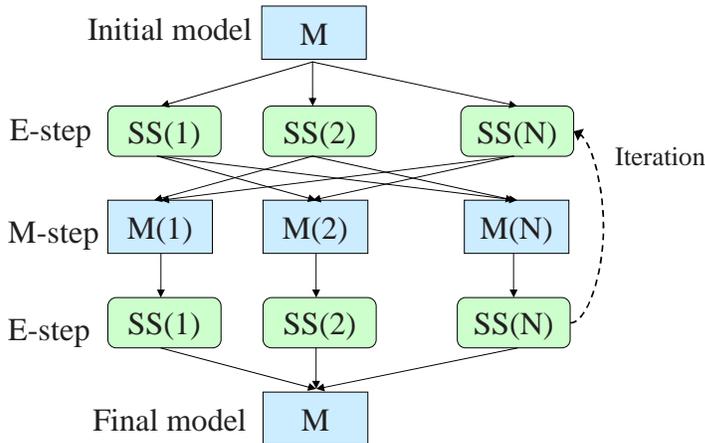


Figure 3.8: *Cross-validation EM training algorithm*

Figure 3.8 shows the CV-EM procedure. The system starts from an initial single model to be trained and finishes also with a single model. On the initial E-step of the EM processing the training data is split into  $N$  partitions as homogeneously as possible (in the implementation each consecutive frame is assigned to a different partition sequentially until all frames have been assigned). Then the conditional probability of each frame to each Gaussian mixture in the initial model is computed. This process is identical to the initial E-step in a similar technique called parallel EM training (Young et al. 2005).

In the following M-step, each model  $M_i$  is reestimated using the sufficient statistics computed for all partitions except for  $SS_i$ , which is kept as cross-validation data. This differs from the parallel EM technique, which collapses all the statistics into creating a single model, losing the cross-validation properties. In the CV-EM algorithm, once all the  $N$  models have been approximated, new conditional probabilities are computed for the frames in each partition  $SS_i$  using model  $M_i$ . As data in partition  $SS_i$  has not been involved in the reestimation of the parameters in  $M_i$ , the accumulated likelihood from all partitions can be used as a cross-validation to check for convergence, avoiding the possible overfitting to the data. In the implementation a  $\Delta\mathcal{L}_{inc} = 0.1\%$  likelihood increase criterion is used.

In Shinozaki and Ostendorf (2007) it proposes a 5 iterations step when training models towards speech recognition, although in speaker diarization a likelihood relative increase stopping criterion is preferred in order to bound the likelihood variation between iterations.

Given two clusters  $A$  and  $B$ , with data  $X_A$  and  $X_B$  and their respective models,  $M_A$  and  $M_B$ , when training such models let us consider the variation in likelihood between two EM iterations as  $\Delta\mathcal{L}(X_A|M_A)$  and  $\Delta\mathcal{L}(X_B|M_B)$ . Within the diarization system we want to use the  $\Delta\text{BIC}$  metric to determine whether they belong to the same speaker or not. By using the modified

$\Delta BIC$  constrained by 3.2, and expanding terms, we obtain:

$$\begin{aligned} \Delta BIC(A, B) = & \log \mathcal{L}(X_A|M_{A+B}) + \log \mathcal{L}(X_B|M_{A+B}) \\ & - \log \mathcal{L}(X_A|M_A) - \log \mathcal{L}(X_B|M_B) \end{aligned} \quad (3.3)$$

In the usual proceeding of the algorithm, by comparing the resulting  $\Delta BIC$  value to a threshold 0 it will be determined whether both clusters are the same speaker or not. If each of the models is trained an extra EM iteration, and using the notation introduced before, one can express the resulting  $\Delta' BIC$  in terms of the one just computed in equation 3.3 as

$$\begin{aligned} \Delta BIC'(A, B) = & \Delta BIC(A, B) + \Delta \mathcal{L}(X_A|M_{A+B}) + \Delta \mathcal{L}(X_B|M_{A+B}) \\ & - \Delta \mathcal{L}(X_A|M_A) - \Delta \mathcal{L}(X_B|M_B) \end{aligned} \quad (3.4)$$

In order for the system to be robust and results consistent it is desired that  $BIC'(A, B) = BIC(A, B)$  which leads to having the likelihood variation terms to cancel out. While it is not possible to control the exact likelihood variations between iterations, by using a minimum relative likelihood variation as a stopping criterion for the CV-EM training makes these terms upper bounded and the BIC more stable. Furthermore, by forcing these variations to be small will result in  $BIC(A, B) \simeq BIC'(A, B)$  as desired.

According to Shinozaki and Ostendorf (2007), since N cross-validation models are reestimated from different subsets of the data it could potentially create a problem where the Gaussian mixtures would behave differently to the data and obtain totally different parallel models, in which case the CV-EM algorithm would not be usable. In reality the difference in number of samples between any two models is  $\frac{1}{N-1}$ , which becomes very small when N is large, and therefore prevents this divergence from happening.

Once the CV-training stopping criterion is reached, the current sufficient statistics computed for each of the subsets are used to derive a single output model. The increase in computation for this parallel training technique is small as only in the M-step the number of operations is increased. When the size of the training data is big, the most costly part of the EM algorithm is the E-step, which takes the same time to be computed as by the CV-EM algorithm.

In order to avoid quick changes in the speaker turns in both the baseline and the current system, a minimum duration of 3 seconds is imposed when performing Viterbi segmentation of the data. This is imposed in the speaker model by using multiple consecutive states with transition probability 1 between them, and tied Gaussian mixture models, as seen in figure 3.2.

On the contrary, it was observed that the maximum turn duration for the speaker turn is

artificially constrained by the  $\alpha$  and  $\beta$  parameters in figure 3.2. As explained in detail in section 4.2.3 these were changed to  $\alpha = 1$  and  $\beta = 1$  to allow the maximum duration to be solely decided by the acoustics. This is an important change given that conference room data is very different in terms of average speaker turn length to broadcast news and to lecture room data.

As mentioned earlier, when processing multiple microphones the system creates an independent feature stream to the acoustic stream composed of the TDOA values between microphones. As explained in section 5.3, each one of the feature streams is represented by different models and the total likelihood of the data at any instant is obtained as the weighted sum of the log-likelihood of the respective feature vectors according to their models. The resulting log-likelihood affects the decisions made in the Viterbi segmentation module and in the  $\Delta$ BIC computation between two clusters, which otherwise are identical to the broadcast news system.

In order for the different independent feature streams to be combined at the log-likelihood level a relative weight has to be assigned for each one depending on their reliability to contribute to the diarization. Although an initial weight is set for all meetings using development data, each particular meeting will respond differently to the use of the TDOA values and therefore an automatic system of reestimating these initial weights is desirable. An effective way was found using a metric derived from the  $\Delta$ BIC values computed between all pairs for all feature streams. It is described in section 5.3.2.

### 3.3.4 Clusters Merging and System Output

When computing the BIC metric between two clusters it was observed that small amounts of non-speech data affect negatively the speaker models and therefore could create errors when deciding whether to merge them or not. A new technique called frame-based cluster purification is introduced to modify the cluster models for the BIC comparison step in order to obtain more discriminant models. It is explained in detail in section 4.3.1.

It has also been observed that some clusters contain speaker segments from more than one speaker. The models associated with these clusters are able to model both speaker correctly and therefore cause a problem when comparing with other clusters containing either one of those speakers, leading to potentially serious decrease of performance due to erroneous cluster merges. For this reason, and during the initial iterations of the segmentation and clustering algorithm, a segment-level cluster purification algorithm aims at detecting speaker segments that are very dissimilar to the cluster they belong to, and assign them to a new cluster. A further description of the algorithm is given in section 4.3.2.

In order to save some computation when computing the  $\Delta$ BIC metric among all possible pairs, a pruning algorithm was implemented in broadcast news that would not compute the  $\Delta$ BIC for those pairs that had previously obtained a negative value. For the meetings system this

was revisited and it was observed on development data that, specially during the initial iterations of the algorithm, the  $\Delta\text{BIC}$  metric would oscillate between small positive and negative values for some clusters until they would finally stabilize its assigned data. By using such a restrictive pruning, the system does not allow such clusters to eventually merge, even though they might be from the same speaker.

For this reason the pruning algorithm was relaxed to eliminate a cluster pair from further comparisons only if its  $\Delta\text{BIC}$  value falls below a certain threshold ( $< 0$ ), much safer to use. Such threshold was set to -100 as it was seen that  $\Delta\text{BIC}$  values below this threshold would always remain negative throughout the process and therefore there is no chance of eliminating any potential merge pair.

As in the broadcast news system, the cluster pair with the biggest  $\Delta\text{BIC}$  value is merged into a common cluster. The resulting model is the union of both merged models. If none of the cluster pairs obtains a positive  $\Delta\text{BIC}$  value, no merging takes place and the system prepares to finish, as it failed the stopping criterion.

In the Meetings system only the local  $\Delta\text{BIC}$  stopping criterion is used as tests using a likelihood criterion (in the same way as in the broadcast news system) resulted in worse performance.

When the system's stopping point is reached, the algorithm does a final post-processing in order to output a final clustering. During the iterative merging process the minimum speaker turn duration is set to be 3 seconds. This is necessary when there are many clusters, each containing small amounts of data, as the corresponding models can fluctuate a lot and not model the speaker appropriately.

Once the system determines to stop merging, the optimum amount of clusters has been reached and the models are expected to contain enough data to model each speaker appropriately. At this point, a single Viterbi segmentation iteration is performed where the minimum duration is set to 1.5 seconds in order to allow the output segmentation to contain smaller speaker turns, given that in meetings the average speaker turn duration is smaller than in broadcast news, as seen in section 3.2.



## Chapter 4

# Acoustic Modeling Algorithms for Speaker Diarization in Meetings

This chapter covers the main contributions of this thesis in the area of acoustic modeling for speaker diarization in the meeting domain. As pointed out earlier, these algorithms were defined either to improve an existing algorithm in the baseline system or created new to solve problems detected in the system.

This chapter is structured into three main sections. The first section introduces a new speech/non-speech detector that does not require any training data while achieving similar performance to the prior pre-trained system on non-speech detection, and better diarization performance.

The second section covers four algorithms used in the definition of the speaker clusters and the related models. The first algorithm automatically defines a number of initial clusters for the agglomerative clustering to start with. The second algorithm obtains an initial clustering by classifying the acoustic data into the desired number of initial clusters. On the topic of speaker modeling, the third algorithm is used to determine the complexity of each model in the system given the amount of data available for training. Finally, a modification to the baseline duration modeling is proposed to avoid any artificial constraints imposed previously to the speaker turn duration.

The third section explores the problems derived of clusters containing data other than a single-speaker. When comparing two speaker models an erroneous decision can be made depending on the amount of such misplaced data. This section presents two algorithms to purify the clusters and avoid such problems. On one hand, the frame level purification modifies the speaker models only in the comparison step by filtering out acoustic frames that might harm the comparison. On the other hand, the segmentation level purification detects full segments that do not match the cluster they belong to and assigns it to a new cluster.

## 4.1 Speech/Non-Speech Algorithm

In the broadcast news domain it was shown in Wooters et al. (2004) that the speaker diarization performance can be improved by the use of a speech/non-speech detector as a first step to the agglomerative clustering process. The speech/non-speech system used in this previous work was based on acoustic models that needed to be trained on data as similar as possible to the test data. This poses a robustness problem when one intends to use the diarization system on “unseen” data, and slows down the portability of the system to new environments, where new training data needs to be labelled/located and new speech/non-speech models need to be trained. For this purpose an alternative was sought that would not require any training.

Among the systems that do not use acoustic models for speech/non-speech detection, the most widely used, when non-speech is considered mainly of a silence/noise nature, always include energy as a feature. The performance of such systems is dependent on setting appropriate thresholds which are typically tuned using some development data. Tests done with the energy-based decoder, that is part of the hybrid system presented below, showed that the optimum threshold depends on the meeting acoustics and therefore it should be tuned whenever data from a different source needed to be processed, falling in the same trap as with model-based detectors.

A novel system was designed and is presented in this thesis to perform speech/non-speech detection, and its application to speaker diarization in the meetings environment. Such system takes advantage of the fact that most non-speech in meetings is silence. It first performs an energy-based detection of the silence portions in the input data using energy derivative filtering based on Li, Zheng, Tsai, and Zhou (2002). This system only needs a coarse setting of a threshold, which is then iteratively modified until hypothesizing a reasonable number of silence segments. The second stage of the system models speech and silence given the output from the first stage by using GMM models, and creates a final speech/non-speech segmentation to be used in the diarization system. By running this two-stage system, it is avoided to use any external training data to obtain an initial set of acoustic models. From these initial models several iterations are performed between segmenting the data and retraining the models to obtain the final speech/non-speech segmentation.

The introduced hybrid system therefore attempts at solving some of the problems from both model-based and energy-based speech/non-speech detectors. On one hand, it is avoided to accurately tune the energy threshold by using an iterative search of a rough speech/non-speech segmentation used to initialize the model-based decoder. On the other hand, by using such initialization on the model-based decoder, it is avoided having to train its models with pre-labelled data, resulting in a system that is freed of the need for training data.

In the following sections both the energy-based decoder and the model-based decoder used in the hybrid system are described. Finally, the combination of both systems into the hybrid decoder is explained.

#### 4.1.1 Energy-Based Speech/non-Speech Detector with Variable threshold

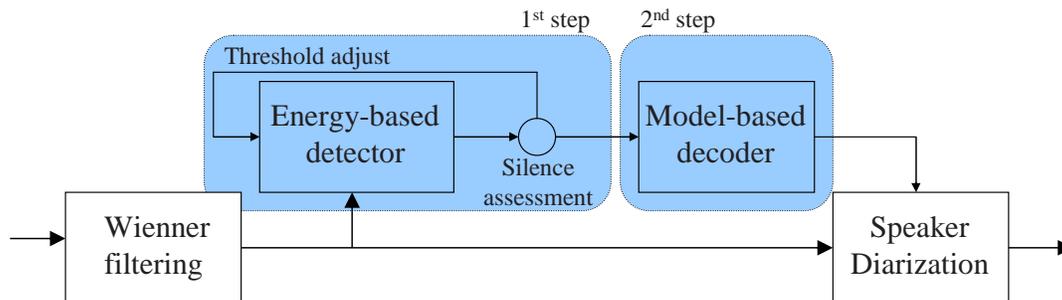


Figure 4.1: *Energy-based detector blocks diagram*

The first stage of the process consists on an energy-based speech/non-speech detector which can be divided into three major blocks as seen in figure 4.1. Each of these blocks are explained below. First of all, the data is preprocessed using common engineering techniques with the purpose of increasing the quality of the speech signal. Then a derivative filter is applied over the energy signal. Finally we use a thresholding method together with a minimum duration enforcement via a Finite State Machine (FSM) to detect silences. This work was initiated by M. Aguilo while visiting ICSI, and was assembled into the current system by the author. For a deeper explanation of each individual module refer to Aguilo's master's thesis in (Aguilo 2005).

#### Data Preprocessing

Due to the different sources and recording setups, the average amplitude of the signal to be processed can vary over a large range. Therefore it needs to be normalized to be able to bring consistency in the follow-on processing. A standard energy average over all the recording would not be plausible due to the existence of extended silence regions and of sudden noise bursts. In order to compute the normalization constant  $\mu$  was chosen, which is more robust to these effects as shown in equation 4.1. This same expression is used in the filter&sum processing to obtain the overall channels weighting factor of the input signals 5.2.2.

$$\mu = \frac{1}{P} \sum_{p=1}^P \max(s[p \cdot TF_s], \dots, s[(p+1) \cdot TF_s]) \quad (4.1)$$

where  $P$  is the total amount of non overlapped blocks of duration  $TF_s$  (with  $F_s$  being the sampling rate in samples/second, and  $T$  is the analysis segment size in seconds) in the recording. Each block of samples ranges from  $p \cdot TF_s$  to  $(p + 1) \cdot TF_s$

Finally a low-pass Butterworth filter deletes all high band noises leaving only information of the signal below  $4kHz$ . This is done because the major part of the energy of the signal is contained in this band and no information is needed but the energy at this point of the non-speech detection process. This Butterworth filter has been implemented using its IIR form.

### Derivative Filtering

Given the normalized and filtered energy signal ( $\tilde{e}[n]$ ) a derivative filter is used in order to enhance the speech/non-speech change-points. This processing helps prevent degradation due to low signal-to-noise ratios or nonstationary environments and was first introduced by Li et al. (2002). Such filter is defined via the following impulse response,

$$h[n] = \{-f[-W \leq n \leq 0], f[1 \leq n \leq W]\} \quad (4.2)$$

Where,

$$\begin{aligned} f[n] = & e^{An}[K_1 \sin(An) + K_2 \cos(An)] \\ & + e^{-An}[K_3 \sin(An) + K_4 \cos(An)] \\ & + K_5 + K_6 e^{sn} \end{aligned} \quad (4.3)$$

And,

$$\begin{aligned} A &= 0.41s \\ s &= \frac{7}{W} \\ W &= \text{Half of the window length.} \end{aligned} \quad (4.4)$$

And the values of the coefficients  $[K_1 \dots K_6] = [1.583, 1.468, -0.078, -0.036, -0.872, -0.56]$ , for a chosen window length  $W = 31$ . The selection of an appropriate value for the  $W$  parameter is important as it sets the temporal resolution of the detector.

As shown in fig. 4.2 the result of the convolution of  $\tilde{e}[n]$  and  $h[n]$ ,  $\hat{e}[n]$  is thresholded and labelled, each sample, as *speech* or *non-speech*.

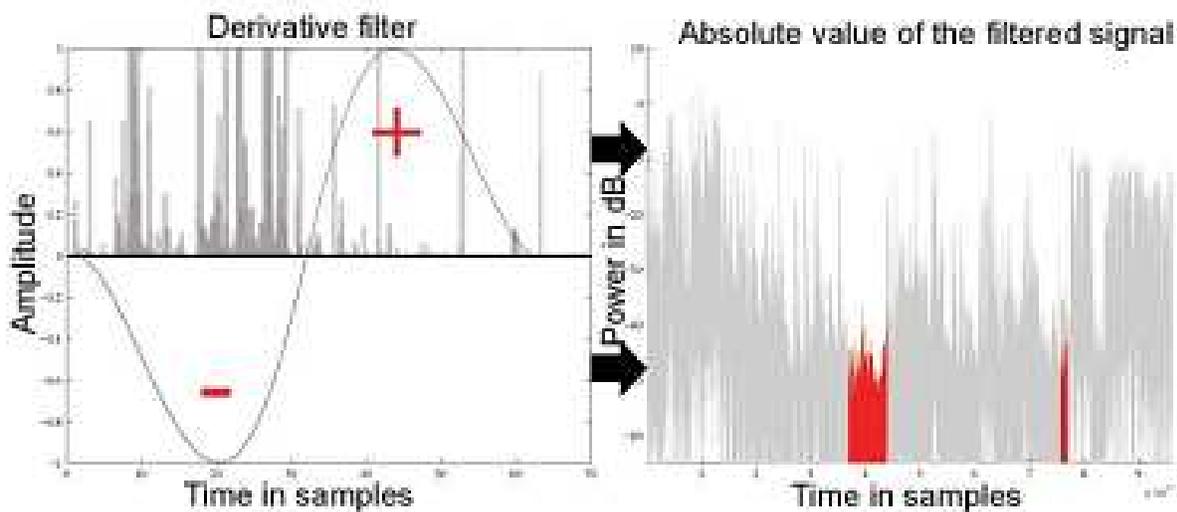


Figure 4.2: Left, filter over  $\tilde{e}[n]$ . Decision of silence in red after the thresholding.

### Time Constraints on Speech/non-Speech

After the energy is filtered for the third time one needs to impose some time constraints to avoid changing too quickly between *speech* and *non-speech*. A finite state machine (FSM) has been implemented for this purpose. In this FSM described in figure 4.3 the time constraints are forced through *enter times* and *leave times* according to the values of  $\hat{e}[n]$  using two thresholds (*enter thrld*,  $\Theta_{enter}$  and *leave thrld*,  $\Theta_{leave}$ ) on each sample. The selection of the right thresholds is crucial to the correctness of the detector and, although the energies have been initially normalized, might differ from meeting to meeting. The threshold *enter thrld* is defined to be an order of magnitude bigger than *leave thrld* and its value is iteratively defined by the hybrid system described below. As for the appropriate minimum time of either speech or non-speech states it must be estimated using development data, but as it will be shown, it is more independent to meeting room variations than the threshold values.

Inside the FSM, the conditions to go from *non-speech* to *speech* are the same to go from *speech* to *non-speech*. This way to go from *speech* to *non-speech*,  $\hat{e}[n]$  has to be higher than the threshold to enter ( $\Theta_{enter}$ ), and vice versa:

$$\begin{aligned} \hat{e}[n_1] \geq \Theta_{enter} \ \& \ State_t = NSP \ \rightarrow \ State_{t+1} = SP \\ \hat{e}[n_2] \leq \Theta_{exit} \ \& \ State_t = SP \ \rightarrow \ State_{t+1} = NSP \end{aligned} \quad (4.5)$$

where NSP is a non-speech state and SP is a speech state.

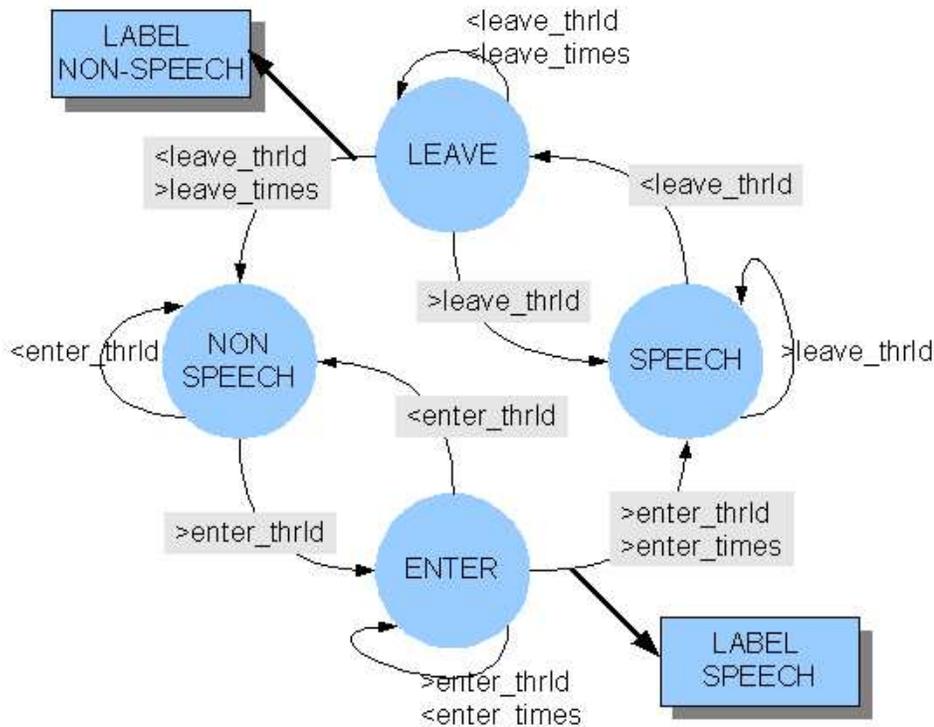


Figure 4.3: State machine used to apply time constraints.

#### 4.1.2 Model-based Speech/Non-Speech Decoder

The second stage of the process consists of a model-based speech/non-speech detector which obtains an initial segmentation (used for training its models) from the output of the energy-based detector. It then produces the speech/non-speech labels that are used for the speaker diarization task. By training the models from the output of the energy-based detector, it avoids the need for any external training data (or pretrained models).

The model-based decoder is composed of a two states ergodic HMM (following the same architecture as the speaker Diarization system), where one state models silence using a single Gaussian model, and the speech state uses a GMM with  $M$  mixtures ( $M > 1$ ). In each state a minimum duration  $M\_D$  is imposed which is allowed to be different from the duration set in the energy-based detector. EM-ML is used to train the models and Viterbi to decode the acoustic data. An iterative segmentation-training is performed until the overall meeting likelihood stops increasing, then the system outputs the speech/non-speech labels.

In order for the speech and silence models to represent well the acoustic information, there needs to be enough frames of data in the input segmentation for each model. As seen in Anguera, Wooters and Hernando (2006d) and Anguera, Wooters and Hernando (2006b) the silence data

can be modeled with a single Gaussian with a very narrow variance. On the other hand, the speech information is much “broader” and dependant on the speakers present in the meeting. It is therefore important for the data used in training the silence model to contain as little speech data as possible. This translates into a very small “missed speech” rate requirement in the energy based detector.

### 4.1.3 Hybrid Speech/non-Speech Detection

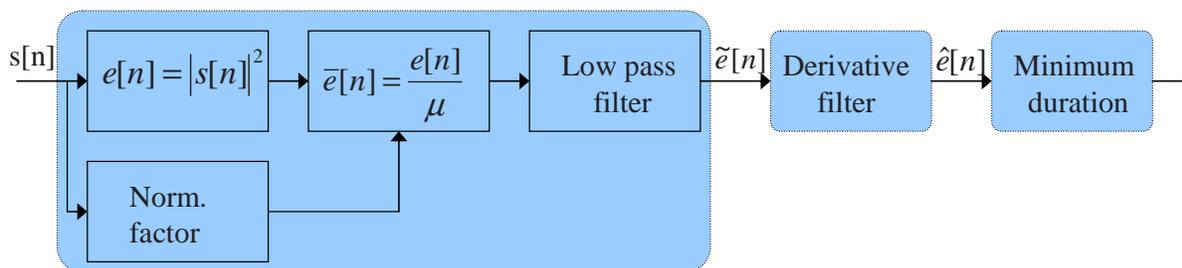


Figure 4.4: *Hybrid Speech/non-speech detector blocks diagram*

The hybrid Speech/non-Speech detector introduced here is composed of a 2 step process, as seen in figure 4.4, combining the energy-based detector and the model-based decoder presented above. The output of the energy detector is used exclusively to initialize the model-based decoder, whose output is used as the speaker diarization speech/non-speech input.

As described above, the functioning of the energy detector depends on setting a threshold value properly. In an exclusively energy-based system such threshold has to be defined using a development set as close as possible to the test set to obtain optimum results. By using a model-based decoding as a second step one can relax the need for a perfectly tuned threshold since the aim now is to obtain a rough distinction between speech and non-speech. The Energy detector is initially run with a very low threshold pair (1e-5/1e-6). While the number of non-speech segments found ( $N_{sil}$ ) is smaller than 10 the threshold pair is raised by an order of magnitude and the energy system is rerun (the system’s computational requirement’s are minimal).

This is done iteratively until  $N_{sil} > 10$ ). At that point, if  $N_{sil} > 100$  it is considered that there are too many silence segments and a refinement step lowers the threshold pair, using a smaller threshold step size, until obtaining between 10 and 100 non-speech segments. The selection of the range (10 - 100) is defined *a grosso modo* in order to obtain a sufficient amount of silence frames to train the silence model in the model-based decoder with a low percentage of speech labelled as silence.

Such speech/non-speech segments are used to train the two models in the model-based

decoder, which performs iterative Viterbi decodings and EM-ML training on the data until reaching likelihood convergence.

The use of two well known speech/non-speech detection techniques back-to-back allows for the creation of a more robust system than using either of them alone. On one hand, on a system totally energy-based it is found that the optimum thresholds defining the speech and non-speech segments are different from one recording type to another (as it depends on the room, microphones used, distance of the people to them, etc.) and therefore they need to be optimized using data from the same source, becoming very dependent on it. On the other hand, in a totally cluster-based system, there is a need for pre-labelled data in order to train the models (or somehow generated initial models), which is also very dependent on the type of recording. By using both systems any kind of data can be processed on its own, without the burden of similar data collection or annotation.

The proposed system is not parameter free. There are three main parameters that need to be determined in order to obtain optimum results. These are the minimum duration of the speech/non-speech in the energy-based detector, the number of Gaussian mixtures assigned to speech in the model-based decoder and the minimum duration of speech and non-speech in such decoder. These are though more robust to changes in the recording acoustics.

## 4.2 Speaker Clusters Description and Modeling

### 4.2.1 Friends-and-Enemies Initialization

In this section a new cluster initialization algorithm is presented (see Anguera, Wooters and Hernando (2006c)), which has been called “friends-and-enemies” initialization due to the way that single-speaker segments are grouped with those closer to them, and new clusters are created as enemies of the existing clusters. A cross-likelihood metric is used to determine “friendliness”. This algorithm is aimed at improving the prior linear initialization algorithm, explained in section 3.1.2.

The clusters initialization block has often been considered to be of less importance in the past, as many segmentations and models retraining iterations take place later in the process that should allow any “pseudo-optimal” initializations to perform as well as any other in the end. In this respect it has been considered that the best initialization is that which does not introduce any computational burden to the overall system.

With a marked reduction of the error in the current system, it has been seen that the linear initialization *does* cause a problem on the final score, since some initial clustering errors are propagated all the way to the end of the agglomerative clustering and show up in the final

result. It has also been observed that a linear initialization without any acoustic constraints on the initially created clusters introduces a random effect in the system which could be one of the sources of per-show “flakiness”, as presented in Mirghafori and Wooters (2006).

When designing an initialization algorithm for speaker diarization there is an additional problem beyond the standard problem of acoustic clustering. It is important to constrain the classification of every acoustic frame according to its context, in the same way as it will be classified within the rest of the system, which uses a minimum duration for a speaker turn to avoid instabilities and very short segments. For this reason it is important to separate into two different initial clusters, for example, data from a speaker in a solo presentation and data from the same speaker in an overlap region, or in a region with a lot of non-detected silence segments as the models used initially are small and can cause problems when modeling different kinds of data. These clusters should be merged in the agglomerative clustering, using more complex models.

The proposed “friends-and-enemies” initialization algorithm aims at creating  $K_{init}$  initial clusters ensuring that each cluster contains data only from one speaker. It intends to maximize the cluster purity, as introduced in Gauvain et al. (1998), which accounts for the percentage of frames in any given cluster that come from the most represented speaker in that cluster. It differs from the DER in that it does not pretend to find the optimum number of clusters, but rather obtain each cluster with only one speaker worth of data in it.

### Initialization Algorithm Description

The proposed initialization algorithm, called friends-and-enemies initialization, is designed to split the acoustic data to be processed into  $K_{init}$  clusters, where  $K_{init}$  is determined beforehand by the algorithm presented in 4.2.2 or manually set by the user. In the agglomerative clustering scheme presented for the meeting domain, it corresponds to the initial number of clusters used to start the agglomerative process. Each of the resulting initial clusters has a duration which is not restricted to be equal to any other cluster.

The complete initialization is composed of three distinct blocks, as shown in Figure 4.6. The first block performs a speaker-change detection on the acoustic data to identify segments with a high probability of containing only one acoustic event. Such acoustic events can either be silence, various noises, an individual speaker or various speakers overlapping each other. This first step is performed using the modified Bayesian Information Criterion (BIC) metric (introduced by Ajmera and Wooters (2003)) computed between two models created from the data in two adjacent windows of size  $W$ , connected at the evaluated possible change point. The modified  $\Delta$ BIC metric is computed over all the acoustic data every  $S$  frames. A possible change point is selected if  $BIC < 0$ , it corresponds to a local minimum of the  $\Delta$ BIC values around

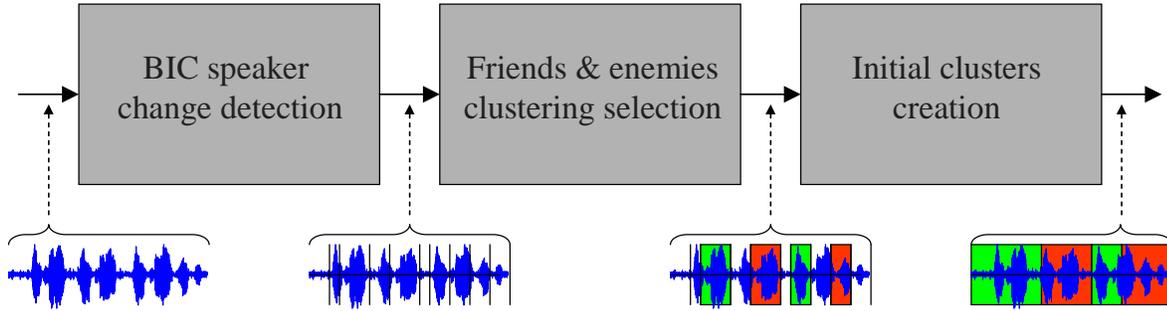


Figure 4.5: *Clusters initialization blocks diagram*

it, and there is no other possible change point with smaller  $\Delta\text{BIC}$  value which is closer than  $MD$  frames to it. In the implementation  $W = 2$  second windows are used, with a scroll  $S = 0.5$  seconds. Each window is modeled using a model with 5 Gaussian mixtures (therefore with 10 Gaussians for the combined model) and a  $MD$  of 3 seconds, equal to the minimum speaker turn duration used in the following agglomerative-clustering process.

The second block in the initialization algorithm creates clusters by identifying the segments defined in the first part as friends or enemies of each other. It is considered that two given acoustic segments are friends if they contain acoustically homogeneous data; only the best friends are brought together to form a cluster. In the same way, it is considered that two segments are enemies if they contain very dissimilar acoustic data. It is intended to obtain  $N$  final enemy groups (the desired final number of clusters) consisting of  $F$  segments each, which are friends of each other. Three different similarity metrics were experimented with to compare each segment pair  $S_1$  and  $S_2$ . On the first place a geometric mean of the frame cross-likelihood as

$$d_1(S_1, S_2) = \frac{\log \mathcal{L}(S_1|\Theta_{S_2}) + \log \mathcal{L}(S_2|\Theta_{S_1})}{(N_{S_1} + N_{S_2})} \quad (4.6)$$

where  $N_{S_i}$  is the number of frames in segment  $i$ , and  $\Theta_{S_i}$  is a model with 5 Gaussian Mixtures trained with  $S_i$ .

The second metric normalizes each term by the number of frames in the linear domain instead, resulting in a penalty to the cross-likelihood as

$$d_2(S_1, S_2) = \log \text{lkld}(S_1|\Theta_{S_2}) + \log \text{lkld}(S_2|\Theta_{S_1}) - \log L_{S_1} - \log L_{S_2} \quad (4.7)$$

The third metric does a full cross-likelihood as introduced by Rabiner in Juang and Rabiner (1985)

$$d_3(S_1, S_2) = \log \text{lkld}(S_1|\Theta_{S_2}) + \log \text{lkld}(S_2|\Theta_{S_1}) - \log \text{lkld}(S_1|\Theta_{S_1}) - \log \text{lkld}(S_2|\Theta_{S_2}) \quad (4.8)$$

All three metrics are bigger the closest the segments are to each other. In order to initiate the process one needs to define an initial segment. Again, three criteria have been considered:

1. Select the segment which is most representative of the meeting, which might indicate that it belongs to the speaker with most participation. In order to find it, a global GMM with 16 Gaussian mixtures is trained using all data in the meeting and the segment with biggest likelihood (normalized using geometric mean) is chosen.
2. Select the segment which is the least representative of the meeting, which might indicate that it belongs to the speaker with smallest participation. Using the same GMM model as before, the segment with smallest averaged likelihood is chosen.
3. Select the segment with the closest average distance to all other segments. Using one of the distances presented above, the average of distances between each segment and all other segments is computed and the maximum is chosen.

Figure 4.6 shows an example case on how the algorithm works. In the horizontal axis the speaker segments as found by the first block, are represented. The vertical axis shows the distance value associated to each segment. In step (0) the initial segment needs to be determined. In this example the criterion 2 is used to find the segment with smallest averaged likelihood,  $S_1$ .

Then, in step (1a) the data in  $S_1$  is used to train a model with 5 Gaussian mixtures ( $\Theta_{S_1}$ ) and compute either metric  $d_{1..3}$  between itself and all other segments. The  $F - 1$  segments with bigger value are its friends. In this example,  $F = 3$  and the selected friends for  $S_1$  are chosen to be  $S_5$  and  $S_7$ . In step (1b), a new model is trained from all data in this first cluster ( $\Theta_1$ ) and the same metric as before is computed, except that now it is measured between all segments in the model and each of the remaining segments.

A new enemy  $S_2$  is selected as the segment with smaller value to the first cluster. Also in the same way, in step (2a)  $F - 1$  friends are chosen for  $S_2$  and in (2b) we select a new enemy for both previously established clusters. This is done by computing the sum of the used metric for each segment given all predefined groups. The processing continues until the desired number of initial clustering  $N$  is reached or it runs out of free segments.

At that point in the third block all created models are used to reassign the acoustic data into the  $K_{init}$  classes. This is done using a Viterbi decoding where the resulting segmentation is not constrained to the predefined speaker changes, therefore any previous speaker change detection errors can be corrected. All data gets assigned to its closest cluster, classifying any acoustic

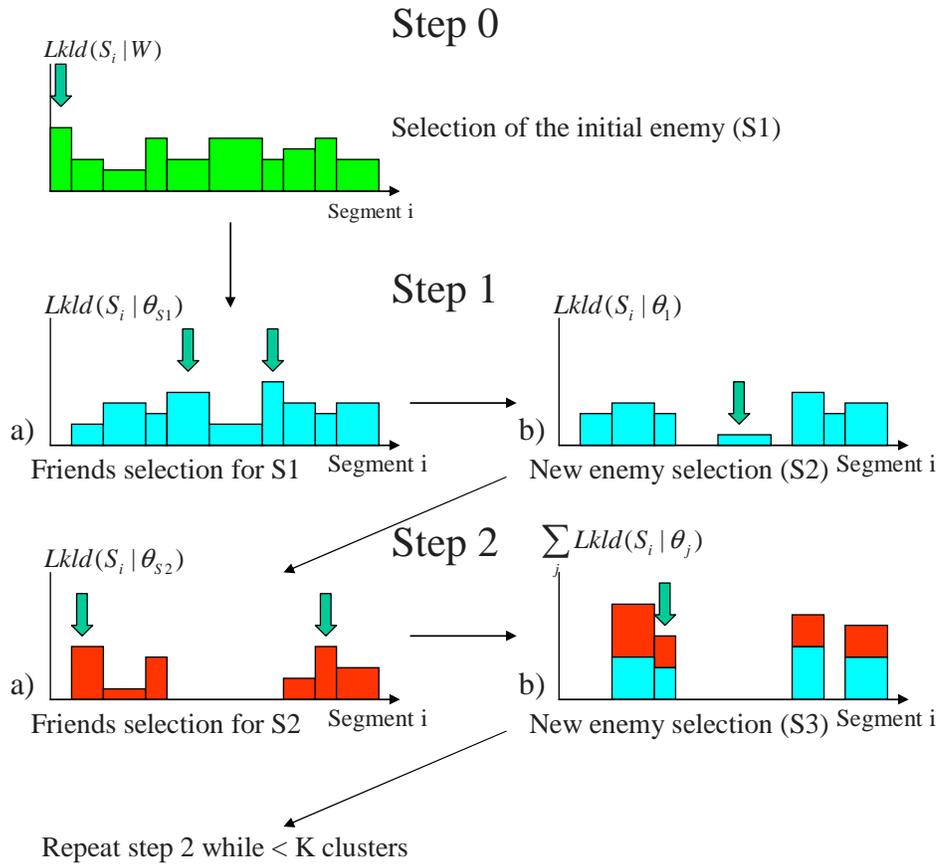


Figure 4.6: *Friends-and-enemies clusters initialization process*

frames not assigned in the previous block. Finally, one cluster model is trained from each of the resulting clusters.

#### 4.2.2 Clusters and Models Complexity Selection

This section describes two algorithms that are used to automatically determine the number of Gaussian mixtures per model and the number of initial clusters to be used in the system. In the baseline system these values were tuned using development data. This approach though was considered deficient as it assumes that both development and test data perform the same way. It was seen that the appropriate number of clusters and the complexity of each model at each stage are strongly dependent on the amount of data available, therefore any difference in the length of the data to be clustered between development and test was seen to harm the performance. Furthermore, each meeting contains a different amount of data after the speech/non-speech detection, which makes any defined parameters not tuned to the particular meeting's properties.

In order to determine the number of Gaussian mixtures and the number of initial clusters,

the algorithms presented below base their selection on information on each particular recording rather than defining a pre-fixed value for all recordings of a certain type. In order to do this, a new parameter is defined which is called Cluster Complexity Ratio (CCR), and which defines a ratio between the amount of data being modeled and the mixtures needed to represent it. The CCR ratio is defined using development data, and it is used to define recording-specific values for the above mentioned parameters. Although the CCR still needs tuning, it allows for individual parameters to be determined for each show, adding robustness to the system.

### Model Complexity Selection

The acoustic models used to represent each cluster are a key part of the agglomerative clustering process. On one hand, comparing their likelihood given the data is how it is decided whether two models belong to the same cluster or not. On the other hand, they are used in the decoding process to redistribute the acoustic data into the different clusters on every iteration.

When designing their size, an important decision is whether to use fixed models (meaning a fixed number of Gaussian mixtures from start to finish), or if it allows the number of Gaussian mixtures to vary according to time or occupancy. Using fixed models is a viable alternative, but runs into the problem of having sufficient training data when the number of Gaussian mixtures is set to be high, or being too general a model when it is set to be small.

Furthermore, when comparing two models via  $\Delta\text{BIC}$ , if they are too general they tend to over-merge, and when they are too specific to the data they under-merge. Therefore it is important to find a tradeoff on the number of mixtures used (model complexity). This has been addressed in the systems presented by ICSI to the RT evaluations for meetings and broadcast news (Anguera, Wooters, Peskin and Aguilo (2005) and Wooters et al. (2004)) by using variable complexities as the merging process progresses. In such systems, all cluster models (regardless of their size) are initially trained using a fixed number of Gaussian mixtures. Upon merging any two clusters, the data from both original clusters are merged and a new cluster model is created as the sum of both parents' Gaussian mixtures.

Such an approach has a drawback that is addressed with the proposed technique. Models with the same complexity are modeling different amounts of data (sometimes very different), therefore their focus is very different. When doing a  $\Delta\text{BIC}$  comparison of such models one cannot expect to obtain coherent results, therefore system performance can degrade.

An algorithm is presented that selects the number of mixtures to be used when modeling each cluster according to its occupancy count. This could be referred to as an **occupancy driven approach**. After each important change in the amount of data assigned to each cluster (normally due to a segmentation step), the number of acoustic frames that are assigned to each of the models is used to determine the number of mixtures by:

$$M_i^j = \text{round}\left(\frac{N_i^j}{CCR_{gauss}}\right) \quad (4.9)$$

The number of Gaussian mixtures to model cluster  $i$  at iteration  $j$  ( $M_i^j$ ) is determined by rounding the number of frames belonging to that cluster at that time ( $N_i^j$ ) divided by a constant value ( $CCR_{gauss}$ ) that is defined as Cluster Complexity Ratio, fixed across all meetings.

In both approaches, the previous and this new one, the total number of mixtures used over all models remains constant in average, being distributed between the different cluster models as described above. This allows tracking of the system evolution by inspection of the Viterbi decoding total likelihood, which can be compared across merging iterations.

The model complexity selection algorithm is executed in the places described in Figure 3.6. The desired complexity of each model is computed using the equation described above and when it is different than the current complexity of the model it is readjusted in one of two possible ways:

- When the final complexity is bigger than the current one, the model is grown, one Gaussian at a time, as described in step 3 of section 3.3.3.
- When the final complexity is smaller than the current one, models are trained from scratch following the procedure in section 3.3.3. As it is explained in that section, eliminating/vanishing Gaussian mixtures from small models is not desirable and leads to a decrease in performance.

### Automatic Selection of the Initial Number of Clusters

In order to perform an agglomerative clustering on the data an initial number of clusters  $K_{init}$  needs to be defined. This value needs to be higher than the actual number of speakers to allow the system to perform some iterations before finding the optimum number of clusters  $K_{opt}$ . It also cannot be too big, as each model needs a minimum cluster occupancy to be trained properly, and to avoid unnecessary computation.

In prior work (Anguera, Wooters, Peskin and Aguilo (2005) for the meetings domain and Wooters et al. (2004) for broadcast news data), the number of initial clusters was fixed within each domain. In the meetings domain, it was set to either 10 or 16 initial clusters, and in the broadcast news domain it was set to 40 initial clusters. The selection of these values had to be tuned to be greater than the possible number of speakers in any given recording while maximizing the performance. As pointed out earlier, this leads to suboptimal results when conditions change.

With the following method, the number of initial clusters is defined on a per recording basis by taking into account the total amount of data available for clustering:

$$K_{init} = \frac{N_{total}}{G_{clusinit}CCR_{gauss}} \quad (4.10)$$

The number of initial clusters is a function of the amount of data available for clustering  $N_{total}$ , the number of Gaussian mixtures to initially assign per cluster  $G_{clusinit}$  (as in prior work,  $G_{clus} = 5$ ) and the Cluster Complexity Ratio  $CCR_{gauss}$  presented in the previous section. This initializes the system using an average complexity of  $G_{clusinit}$  and the amount of data per cluster as defined by  $CCR_{gauss}$ . This technique does not try to guess the real number of speakers present in a recording, but rather sets an upper boundary to the number of speakers that is closely coupled with the complexity selection algorithm and which allows a correct modeling of each initial cluster for each particular recording by determining the optimum amount of data it should be trained with.

### 4.2.3 Acoustic Modeling without Time Restrictions

In this section a small change to the cluster models is proposed which leads to the elimination of the dependency of the acoustic models on the average speaker turn length. This is achieved by modifying the acoustic modeling topology by changing the probabilities of self-loop and transition in the last state. By doing so, a minimum duration for a speaker turn can be implemented like in the past while not influencing the final duration of a speaker turn. While setting a minimum duration for speaker turns is advantageous for the processing of the recordings and can be set to be independent of the kind of recording encountered, the average speaker turn duration is quite variable between individual recordings and domains. It is therefore better to let the acoustic data alone define when the speaker turn finishes once it achieves a minimum length.

In the cluster models each state contains a set of  $MD$  sub-states, as seen in figure 4.7, imposing a minimum duration of each model. Each one of the sub-states has a probability density function modeled via a Gaussian mixture model (GMM). The same GMM model is tied to all sub-states in any given state. Upon entering a state, at time  $n$  the model forces a jump to the following sub-state with probability 1.0 until the last sub-state is reached. In that sub-state, it can remain in the same sub-state with transition weight  $\alpha$ , or jump to the first sub-state of another state with weight  $\beta/M$ , where  $M$  is the number of active states/clusters at that time. In the baseline system these were set to  $\alpha = 0.9$  and  $\beta = 0.1$  (summing to 1).

One disadvantage of using these settings is that it creates an implicit duration model on the data beyond the minimum duration  $MD$ , set as a parameter. Let us consider a sequence of  $N$  feature vectors  $X = \{x[1] \dots x[N]\}$ . Let us also consider a set of  $K$  cluster models  $\Theta = \{\Theta_1 \dots \Theta_K\}$ .

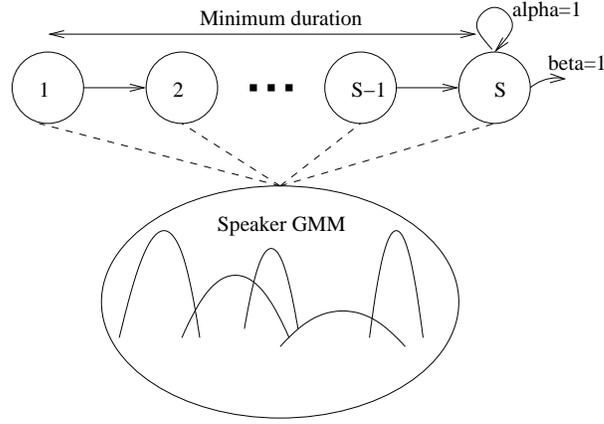


Figure 4.7: Cluster models with Minimum duration and modified probabilities

The system imposes an equal probability to choose either cluster once it outputs a prior cluster and has a minimum duration  $MD$  inside either cluster.

In order to study the interaction between  $\alpha$ ,  $\beta$  and  $MD$  parameters, the likelihood of the data given the models is analyzed. In equation 4.11 the likelihood is written when the system selects model 1 as the initial model and stays in it for the whole  $N$  acoustic frames, therefore creating 0 model changes as

$$\begin{aligned} \mathcal{L}_0(X|\Theta) &= \mathcal{L}(x[1]|\Theta_1) \prod_{i=2}^{MD} (1 \cdot \mathcal{L}(x[i]|\Theta_1)) \\ &\cdot \prod_{i=MD+1}^N (\alpha \cdot \mathcal{L}(x[i]|\Theta_1)) \end{aligned} \quad (4.11)$$

In equation 4.12 the likelihood is computed for the case when one cluster change occurs within the decoded  $N$  frames. The decoding used imposes that the second model will contain at least  $MD$  acoustic frames. Considering models 1 and 2 it can be written as:

$$\begin{aligned} \mathcal{L}_1(X|\Theta) &= \mathcal{L}(x[1]|\Theta_1) \prod_{i=2}^{MD} (1 \cdot \mathcal{L}(x[i]|\Theta_1)) \cdot \prod_{i=MD+1}^{N_1} (\alpha \cdot \mathcal{L}(x[i]|\Theta_1)) \\ &\cdot \frac{\beta}{K} \prod_{i=N_1}^{N_1+MD} (1 \cdot \mathcal{L}(x[i]|\Theta_2)) \cdot \prod_{i=N_1+MD+1}^N (\alpha \cdot \mathcal{L}(x[i]|\Theta_2)) \end{aligned} \quad (4.12)$$

where  $N_1$  indicates a random point in the  $N$  frames, as long as  $N_1 > MD$  and  $N_1 < N - MD$ .

The transition probabilities from these equations are the terms not affected by the acoustic models. By extending the number of changes to  $C$ , the transition probability can be proven that takes the expression:

$$Tr(C) = \left(\frac{\beta}{K}\right)^C \alpha^{(N-(C+1)MD)} \quad (4.13)$$

It is composed of two parts. On one hand, the left side depends on the  $\beta$  parameter and depends exclusively on the number of cluster changes and the number of possible clusters to go to. On the other hand, the right side is dependent on the  $\alpha$  parameter and encodes the duration modeling of each of the acoustic models. This duration model depends on the number of speaker changes  $C$  and the minimum duration  $MD$ .

On the broadcast news system the parameters were set as  $\alpha = 0.9$ ,  $\beta = 0.1$  and  $MD = 3$  seconds. This led to a transition probability which is dependent on  $C$  and  $MD$ , which for many cases created segments that in average were very close to duration  $MD$ . This was because on most cases when evaluating on  $N$  frames of data,  $\mathcal{L}_{i \neq 0}(X|\Theta) > \mathcal{L}_0(X|\Theta)$ . In order to avoid cluster changes every  $MD$  seconds a lower boundary for  $\alpha$  must be set by ensuring that  $tr_{i \neq 0} < tr_0$  computed for a hypothetic case when all models are the same (i.e.  $\Theta_i = \Theta_j, \forall i, j$ ). Applying this condition to the transition probabilities for all possible  $C$  values gives:

$$\alpha^{MD} > \frac{\beta}{K} \quad (4.14)$$

In order to remove the dependency of the  $MD$  on duration modeling, and agreeing with equation 4.14, the parameters were set as  $\alpha = 1.0$  and  $\beta = 1.0$ . Thus, once a segment exceeds the minimum duration, the HMM state transitions no longer influence the speaker turn length; it is solely governed by acoustics. This creates a non-standard (but valid) HMM topology as  $\alpha + \beta$  no longer sums to 1.

### 4.3 Cluster Purification Algorithms

Given the speaker clustering algorithm presented in this thesis, there are usually acoustic frames assigned to a cluster which do not belong to the modeled speaker. These frames are either non-speech or frames from another speaker. In this thesis this phenomenon is referred as cluster “impurity”. It is very important to ensure that the clusters only contain one speaker and therefore the merging decision and stopping point criterion don’t suffer from cluster impurity. Such cluster impurity has been studied separating it into two levels of detail (relative to two sources of error) and two algorithms are presented to detect and purify the clusters.

One source of error occurs when a cluster is created from speech segments from multiple speakers. In standard agglomerative systems there is no mechanism to split a cluster when segments from different speakers are assigned to the same cluster. This effect causes an increase in the final speaker error as seen in the example in Figure 4.8(a) for the case of two misplaced

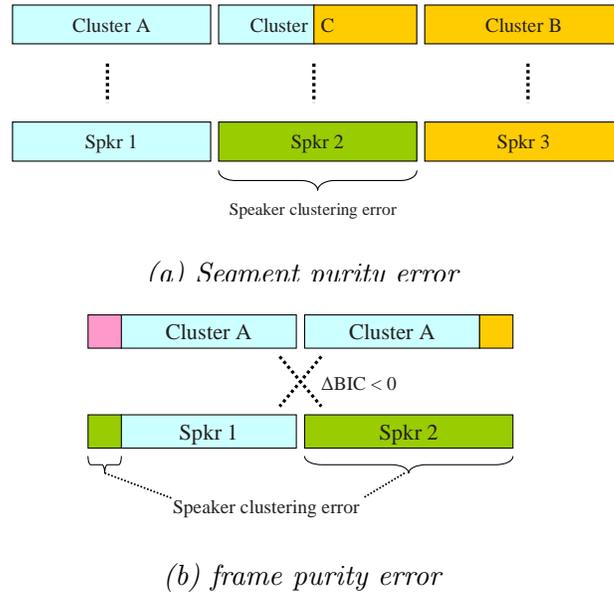


Figure 4.8: Possible Speaker clustering errors due to clusters purity problems

segments of two existing speakers. It is very possible that the speaker model for the mixed cluster is able to represent both speakers' data and therefore Viterbi segmentation does not achieve to homogenize the cluster classifying the acoustic frames into their respective clusters. At the end of the processing, the mixed cluster is likely to be assigned to a non-existent speaker (or to either of the speaker present in it), causing a large increase on the Diarization Error Rate (DER).

The second source of error comes from the interference of non-speech frames in both clusters during cluster comparison. This is particularly true for short silences and short acoustic events that belong to the modeled speaker but do not discriminate one speaker from another. This can affect the final clustering in two ways, as seen in Figure 4.8(b). First, when comparing two clusters belonging to the same speaker, the confounding frames can cause  $\Delta\text{BIC} < 0$  to decide to keep them separate. Second, false alarm errors are produced when non-speech frames are assigned to one of the speakers.

Both sources of error are interrelated and are caused by frames that are assigned to the wrong acoustic model. The difference is the unit that is considered is miss-assigned (segment or frame). In next subsections the algorithms are proposed towards solving both problems. The first algorithm identifies the segments that acoustically deviate most from their cluster, and splits them into a new cluster. This is referred to as “segment-level” purification. The second algorithm locates the individual frames within a cluster that can cause problems in the merging state and avoids using them when computing the distance between the cluster pair. It is referred to as “frame-level” purification.

### 4.3.1 Frame-Level Cluster Purification

Due to the use of a minimum duration in the acoustic modeling, speech segments that legitimately belong to a particular cluster can be “infected” with sets of non-speech frames and frames belonging to other sources. Such sets are too short to be taken into account by the segment-based decoding as independent clusters or eliminated by the model-based speech/non-speech detector without an important increase in miss speech error. They cause the models to diverge from their acoustic modeling targets, which is particularly important when considering whether to merge two clusters. The frame level purification presented here focuses on detecting and eliminating the non-speech frames that do not help to discriminate between speakers (e.g. short pauses, occlusive silences, low-information fricatives, etc).

#### Speech and Non-Speech Modeling

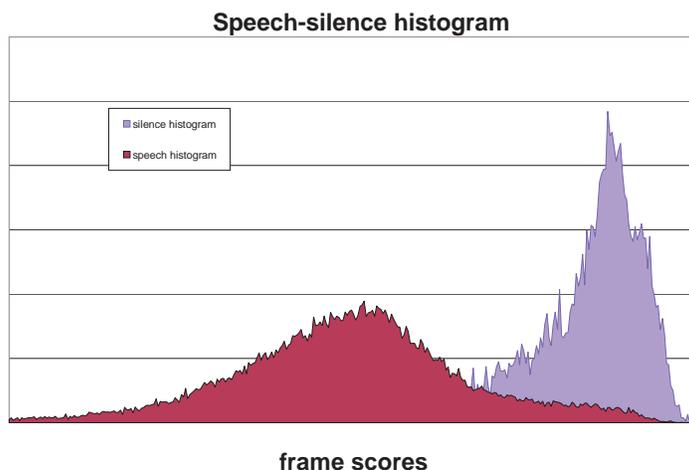


Figure 4.9: *Speech-silence histogram for a full meeting*

In order to see the effect of typical acoustic speaker models with non-speech data an experiment was performed on all the data belonging to an ICSI meeting used in the RT04s evaluation. All the acoustic frames  $X$  from that meetings were split into speech frames  $X_0$  and non-speech frames  $X_1$  according to the reference segmentation file provided by NIST. A speaker model with 5 Gaussian Mixtures was trained using only the speech-labelled frames  $X_1$ . Then both speech and non-speech frames were evaluated using such model and two normalized histograms were created from the resulting likelihood scores, as can be seen in Figure 4.9.

The scores of the non-speech frames  $X_0$  are mainly located in the higher part of the histogram, indicating that  $X_0$  usually obtains higher likelihood scores than  $X_1$  even when evaluating it on a model trained only with  $X_1$  data. Part of the  $X_1$  frames are also in the upper part of the histogram, which are most probably non-speech frames that are labelled as speech in the

reference file. Even with the use of a speech/non-speech detector, a residual error of around 5% of non-speech data enters the clustering system. In order to purify a cluster both the non-speech (undetected) data and the speech-labelled non-speech data needs to be eliminated while maintaining the rest of acoustic frames that discriminate between speakers. It is clear that likelihood can be used to detect and filter out these frames.

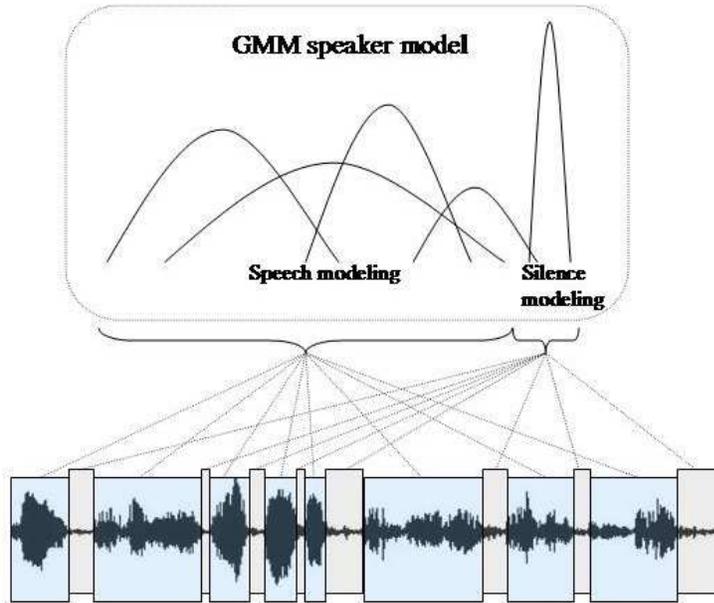


Figure 4.10: *Observed assignment of frames to Gaussian mixtures*

A possible explanation for this behavior is illustrated in Figure 4.10 where a cluster model  $\Theta_A$ , using  $M$  Gaussian mixtures, is trained using acoustic data  $X_1$  labelled as speech by the speech/non-speech detector. After training the model, a group of Gaussian mixtures  $M_1$  adapt their mean and variances to model the subset of the speaker data  $X_{1,1}$ , while another group of Gaussians  $M_2$  appears to model the subset of data  $X_{1,2}$  which are non-speech frames remaining in  $X_1$ . Since the number of frames in  $X_{1,1}$  is typically much larger than those of  $X_{1,2}$ , the number of Gaussian mixtures associated to each subgroup are  $|M_1| \gg |M_2|$  and, at times,  $|M_2|$  could be 0 if the non-speech data is minimal. Furthermore, the variance of the non-speech Gaussian mixtures in  $M_2$  is always much smaller than  $M_1$ . This is the reason why any non-speech frame evaluated by the model gets a higher score than a speech frame. This is taken advantage of in the frame level purification algorithm.

To further prove that the acoustic frames with a higher likelihood are those which are less suitable to discriminate between speaker models another experiment was performed taking two speaker clusters trained with acoustic data for two different speakers according to the reference segmentation. Figure 4.11 illustrates the relationship between the likelihood scores of the data used in training each of the two models and evaluated on both models. It is possible to determine

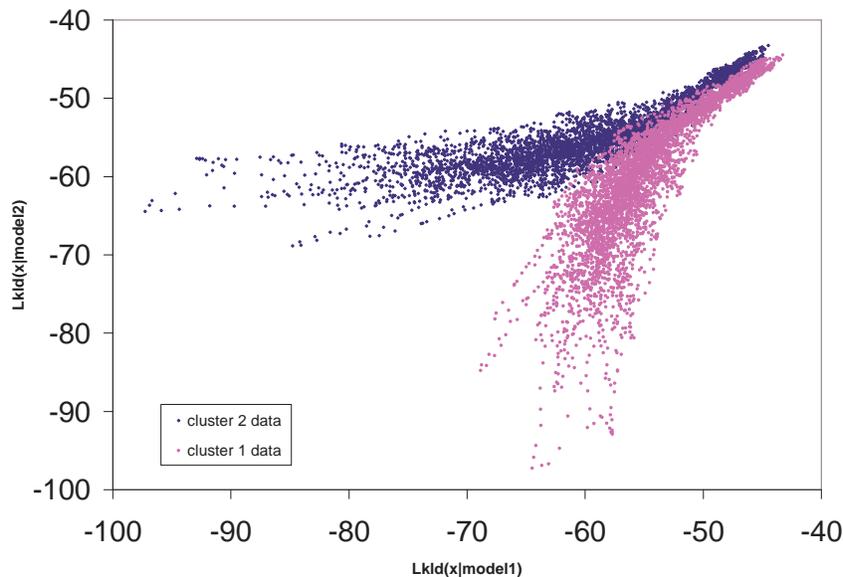


Figure 4.11: *Evaluation of metric 1 on two clusters given their models*

an axis between the likelihood values of the two models. The distance to this axis indicates the discriminative power of the data from each cluster. Frames from both clusters with the highest likelihood values are grouped together on this axis, indicating how badly they can differentiate between speakers.

### Frame-Based Cluster Purification Metrics

In order to detect and filter out the non-speech frames using the detected likelihood property of the non-speech data, two variants of a likelihood-based metric are proposed.

$$\bar{\mathcal{L}}(x[i]|\Theta_A) = \frac{1}{Q} \sum_{j=-Q/2}^{Q/2-1} \sum_{m=1}^{\tilde{M}} \log(W_A[m] \mathcal{N}_{A,m}(x[i+j])) \quad (4.15)$$

The two metrics are based in equation 4.15 where  $Q$  defines the length of an average window and is used to average the measure around the desired value to avoid noisy values;  $\tilde{M}$  is the number of Gaussian mixtures used to compute the likelihood (where  $\tilde{M} < M$ , the number of mixtures in the model);  $W_A[m]$  is the mixture weight and  $\mathcal{N}_{A,m}(x[i+j])(x[\cdot])$  is the result of evaluating  $x[\cdot]$  on the Gaussian mixture  $\mathcal{N}_{A,m}(x[i+j])$ :

**Metric 1** A standard smoothed likelihood over 100ms of data ( $Q = 5$  with 10ms acoustic frames) around each acoustic frame, with  $\tilde{M} = M$  (all mixtures in model  $\Theta_A$ ).

**Metric 2** The same smoothed likelihood (over 100ms) given a model formed by a subset of

all Gaussian mixtures in the speaker model, which include the mixtures assigned to non-speech. The mixtures used are selected by computing the sum of variance over all dimensions and selecting those with smaller accumulated variance,  $\widetilde{M} = M_{non-speech}$ . This second metric is equivalent to metric 1 when 100% of the Gaussian mixtures are selected.

### Frame-Based Purification Implementation

When running the Speaker Diarization algorithm, each cluster is modeled with a variable number of Gaussian mixtures according to the amount of data it contains. It is necessary to analyze at what cluster complexity this behavior is present and the presented metrics can be used. In figure 4.12, the histograms of speech and non-speech (according to the reference file) are shown of metric 1 evaluated using models ranging from 1 to 8 mixtures. All model complexities have been trained with the same data and used to evaluate metric 1 on all the meeting in the same way as in figure 4.9.

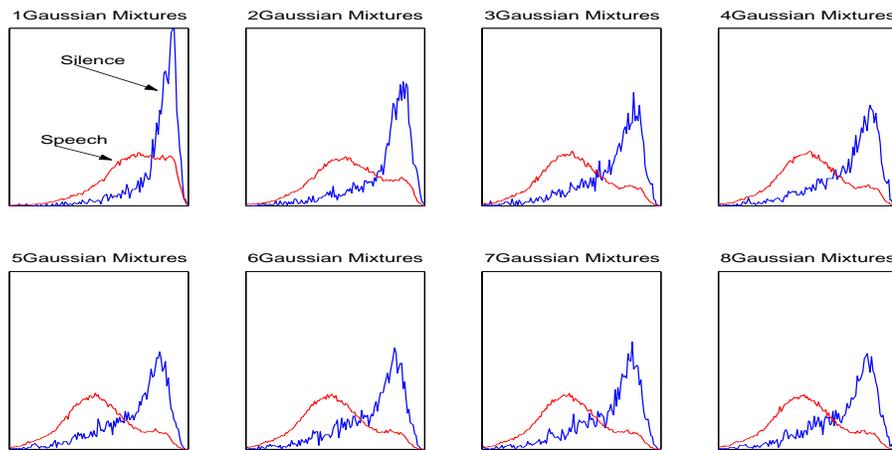


Figure 4.12: *Speech/non-speech histograms for different possible model complexities*

It is seen that only the case of 1 Gaussian mixture shows a bigger overlap between the speech and non-speech histograms, while after 3 mixtures all plots seem identical (in fact, running the same experiments from 1 to 20 mixtures/model gives identical results from 9 to 20). The frame-level purification algorithm is therefore applied whenever the number of Gaussian mixtures is greater than one.

The algorithm is used when gathering the data to compare two clusters using the  $\Delta\text{BIC}$  metric in the following way:

1. Retrieve all frames assigned to each of two clusters and use either metric for each frame in both clusters.
2. If  $M_i > 1$ , eliminate the  $P\%$  of frames in each cluster with the highest computed metric,

where  $P$  is a value to be optimized according to the data.

3. Train two new models with the remaining data and use them for computing the  $\Delta\text{BIC}$  metric.

### 4.3.2 Segment-Level Cluster Purification

There are some situations where a cluster retains speaker segments from more than one speaker; the segment-level cluster purification algorithm is a proposed mechanism used to force splitting these cluster into two parts. The algorithm detects the segments in each cluster that are likely to belong to another speaker and reassigns one of them to a new cluster in each iteration of the agglomerative clustering algorithm. The algorithm works as follows:

1. Find the segment that best represents each model (highest normalized likelihood). This is done to isolate the effect of a big speaker model when trying to determine if it contains any segments from more than one speaker. The most representative segment is very probable to contain only data from one speaker and it is more reliable to compare it with other segments of similar size.
2. Compute, within each cluster, the  $\Delta\text{BIC}$  value between the best segment (found in step 1) and each of the other segments. If all pairs have a value greater than a minimum purity (empirically set to -50) that model is labelled as “pure” and is not checked again in subsequent iterations.
3. The segment that most differs from its model’s best segment is assigned to a new model. All models are retrained and the data is resegmented with Viterbi.

In order to avoid instability, the algorithm is run at most  $K_{init}$  times ( $K_{init}$  being the number of initial clusters). Doing so avoids clusters continuously split and merge the same segments over and over.



## Chapter 5

# Multichannel Processing for Meetings

In meeting room recordings there is normally access to more than one microphone that recorded what occurred in the room synchronously, bringing spacial diversity. It is desirable to take advantage of such signal multiplicity by using multichannel algorithms like acoustic beamforming techniques. In section 2.5 the basic microphone array theory and the main techniques covered by the literature in the topic of acoustic beamforming were reviewed, as well as its use for speech enhancement and the methods previously applied to the meetings environment to this effect.

In order to use multichannel beamforming techniques in the meetings domain one needs to consider the set of multiple microphones to constitute a microphone array. The characterization and use of this array is not done in a classical manner as the locations and characteristics of the available microphones can be non-conventional. The system to be used is required to be robust and not require much prior information as the microphones can be located anywhere (with varying distances between them) and can be of very different quality and characteristics (both in directivity and type). By applying the appropriate techniques, in most cases it is possible to obtain a signal quality gain and to improve speaker diarization and speech recognition results.

One of the necessary steps to perform acoustic beamforming on this environment, with the selected techniques, is the estimation of delays between channels using cross-correlation techniques. Such delays output can be also used as an input to the speaker diarization system to cluster the different speakers in a meeting room by their locations (derived from the delays). Although by themselves they do not carry as much information as the acoustic signal, when combined with it using a multi-stream diarization system (presented in this section) important gains are observed with respect to using acoustic alone.

First, in section 5.1 the real issues encountered in a meeting room multichannel layout are exposed and a filter-and-sum algorithm is proposed and described to process the data. Then in

section 5.2 the full acoustic beamforming implementation developed and used for the speaker diarization and Automatic Speech Recognition (ASR) tasks is covered. Finally, in section 5.3 the use of the delays obtained from the speaker location estimation is explained, describing how they improve the acoustic diarization performance by combining both types of features and how the weighting between features is automatically computed.

## 5.1 Multichannel Acoustic Beamforming for Meetings

Although linear microphone arrays theory sets a solid theoretical background for acoustic microphone array beamforming, its assumptions differ many times from real life applications. In this section the practical characteristics that are encountered in the beamforming implementation and the basic theory behind the implemented system are explained.

### 5.1.1 Meeting Room Microphone Array Characteristics

When using multichannel enhancement techniques to improve speaker diarization in the meetings environment one usually encounters a set of characteristics in the meeting room data that will condition the practical implementation of the system.

1. Microphone array definition: In a meeting the microphones are set in different positions of the room. Some microphones are in the meetings table, some are in the room walls and in some occasions the attendees are wearing head-mounted or lapel microphones. Such multiplicity of types and positions defies the standard concept of microphone array as analyzed in the theory. The lack of a single/optimum microphone, which obtains a clean signal from all participants, makes acoustic beamforming using all available microphones a feasible and worthwhile application for these microphones. Such implementation needs to be sufficiently general to fit such loose microphone array definition.
2. Number of elements: The number of acoustic channels (microphones) available for processing varies from meeting to meeting, not necessarily being kept constant for meetings coming from the same source. The implementation cannot impose any constraint in the number of channels it requires for processing, and should optimally obtain an “enhanced” signal with better quality than either one of the individual signals alone.
3. Different microphone qualities: The frequency response of the different microphones in the meeting room cannot be considered equal as these can be of multiple types. One needs to consider possible differences in the contribution of each of the microphones according to their signal quality, either known a priori or computed automatically by the system.

4. Microphone locations: The exact location of the microphones in the room is unknown. In some cases one can know the relationship between the positions of certain microphone groups (for example microphones within a circular microphone array or a linear array). The microphone settings change for each meeting room and it should not be necessary to know them a priori.

### 5.1.2 Filter-and-Sum Beamforming

The filter-and-sum beamforming is one of the simplest beamforming techniques but still gives a very good performance. It is based on the fact that applying different phase weights to the input channels the main lobe of the directivity pattern can be steered to a desired location, where the acoustic input comes from. It differs from the simpler delay-and-sum beamformer in that an independent weight is applied to each of the channels before summing them.

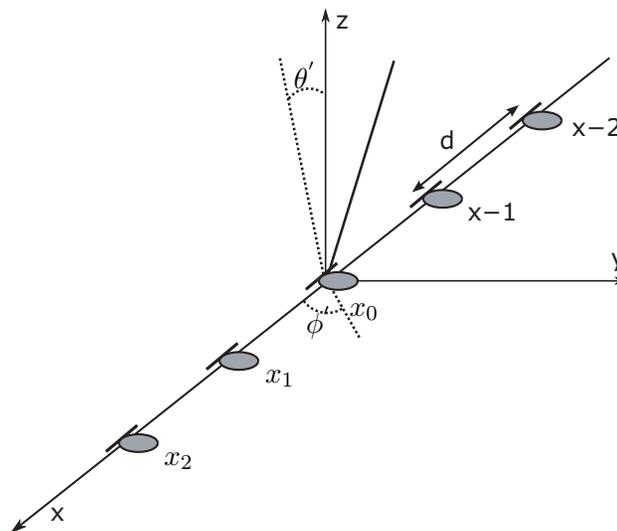


Figure 5.1: *Linear microphone array with all microphones equidistant at distance  $d$*

Let us consider the hypothetical case of a microphone array composed of  $N$  microphones, as seen in figure 5.1, with identical frequency response and equal distance between any two adjacent microphones, of value  $d$  meters. If only the horizontal directivity pattern is considered ( $\theta = \pi/2$ ) and have microphones with equal amplitude weights  $a_n(f) = 1/N$ , the main lobe can be steered to the direction  $\phi'$  using a basic delay-and-sum beamforming by applying the following phase weights to the channels:

$$\varphi_n = \frac{-2\pi(n-1)d\cos(\phi')f}{c} \quad (5.1)$$

Thus obtaining the following directivity pattern:

$$D(f, \phi) = \frac{1}{N} \sum_{n=1}^N e^{j \frac{2\pi f(n-1)d(\cos\phi - \cos\phi')}{c}} \quad (5.2)$$

where the term  $\cos\phi'$  forces the main lobe to move to the direction  $\phi = \phi'$ .

Such steering can be applied in real applications by inserting time delays to the different microphone inputs. In this case the delay to be applied to each microphone to steer at angle  $\phi'$  is:

$$\tau_n = \frac{(n-1)d\cos\phi'}{c} \quad (5.3)$$

Each of the microphone inputs is delayed a time  $\tau_n$  and then all signals are summed to obtain the delay-and-sum output, as it can be seen in figure 5.2. The physical interpretation when the waveform front is considered flat is that  $\tau_n$  is the time that takes the same signal wave to reach each of the microphones.

By using such time delays equivalence the delay-and-sum output  $y(t)$  can be written as

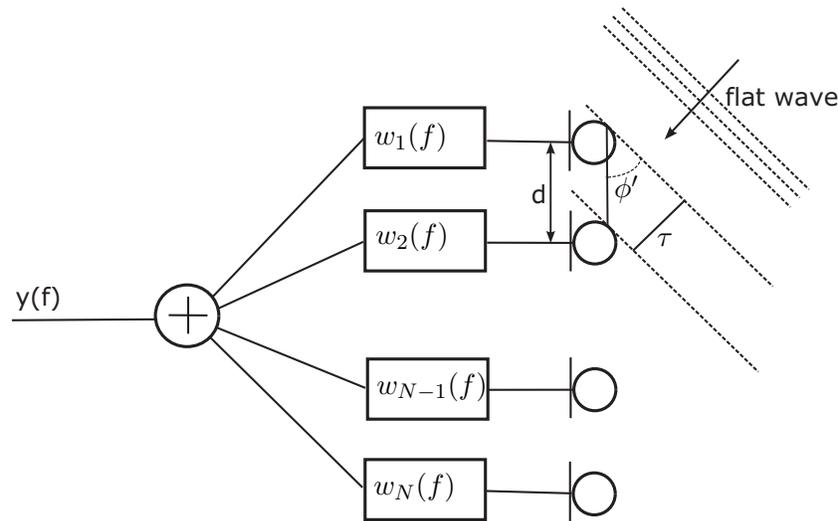
$$y(t) = \frac{1}{N} \sum_{n=1}^N x_n(t - \tau_n) \quad (5.4)$$

The basic delay-and-sum beamforming considers all channels to have an identical frequency response and then uses equal amplitude weights ( $a_n$ ) to all channels. In the application for meetings it will be considered that microphones have different (and unknown) frequency responses. This problem can be addressed by adding a non-uniform amplitude weight and making both amplitude and phase weights frequency dependent. Therefore obtaining a filter-and-sum beamforming system output as

$$y(f) = \sum_{n=1}^N w_n(f)x_n(f) \equiv \sum_{n=1}^N a_n(f)x_n(f)e^{-2\pi f \frac{(n-1)d\cos\phi'}{c}j} \iff y(t) = \sum_{n=1}^N a_n(t)x_n(t - \tau_n) \quad (5.5)$$

where  $w(f)$  is determined by eq. 2.31.

Figure 5.2 represents what is seen in equation 5.5. The input signal (considered to be coming from a distant source and flat) arrives to each microphone from an angle  $\phi'$  at a different time instant. The signals from the different microphones are passed through a filter  $w_i$ , independent for each microphone (1 through  $N$ ), which accounts for an amplitude and time delay (as seen in eq. 5.5). The output or “enhanced” signal is the sum of all filtered individual signals.

Figure 5.2: *Filter and sum algorithm blocks diagram*

This type of beamforming technique was selected for the implementation of the meetings system because it agrees with all desired characteristics. Furthermore, its simplicity allows for a fast implementation, normally under real-time, that allows it to eventually be used in a real-time system.

## 5.2 Multichannel Acoustic Beamforming System Implementation

This section describes the implementation of the multichannel acoustic beamforming system for meetings based of the filter-and-sum system presented in section 5.1 and also described in Anguera, Wooters and Hernando (2005). This involves the processing of the signal from each available microphone until obtaining the final “enhanced” channel output and other related information useful for further processing in the mono-channel speaker diarization system presented in the next section.

Figure 5.3 shows the different blocks involved in the filter-and-sum process. The system is able to handle from 2 microphones to as many microphones as memory allows in the computer where it is executed. Each processing stage is either performed on each individual microphone or to all microphones in combination. Each processing block is described in detail below.

### 5.2.1 Individual Channels Signal Enhancement

Prior to doing any multichannel beamforming each individual channel is Wiener filtered (Wiener and Norbert 1949). It aims at cleaning the signal from corrupting noise, which is considered to

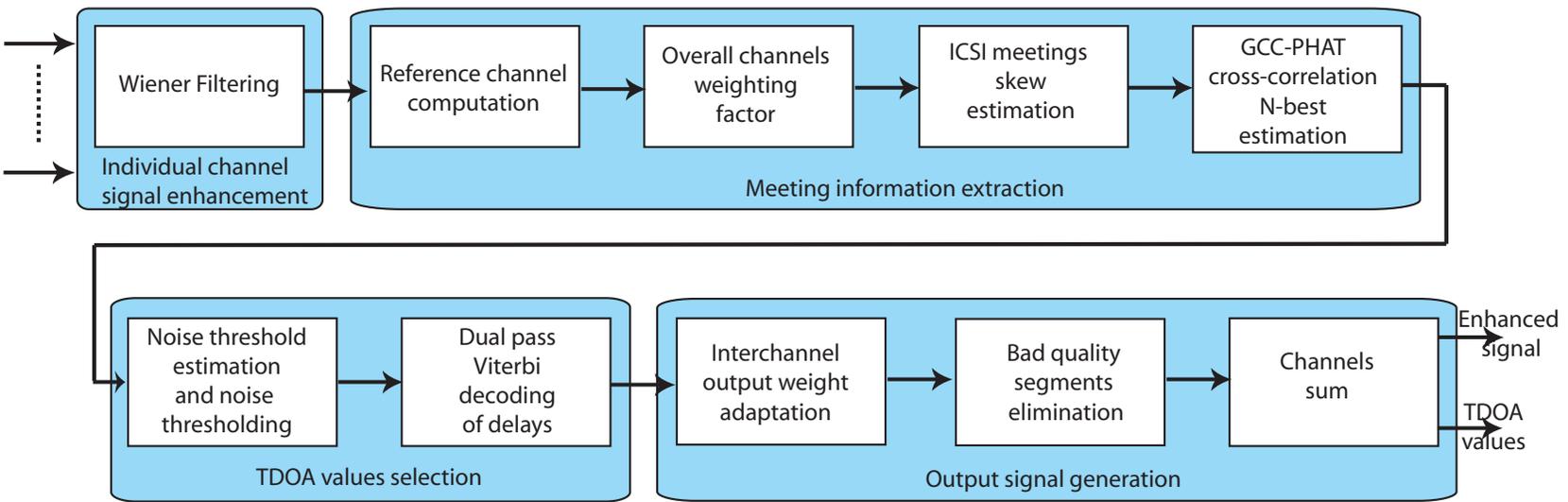


Figure 5.3: filter-and-sum implementation blocks diagram

be additive and of a stochastic nature. The Wiener filter parameters  $w(t)$  are chosen so that the mean square error between the clean signal  $x(t)$  and the resulting output signal  $s(t)$  is minimized. Considering an additive noise  $n(t)$  it can be written as:

$$x_n[k] = w_n[k] * (s_N[k] + n_n[k]) \quad (5.6)$$

where  $s_n[k]$  and  $n_n[k]$  are the discrete speech and noise recorded by each of the  $N$  channels in the room, and  $x_n[k]$  is the cleaned signal which will be further processed by the system.

In this implementation Wiener filtering is applied to each channel independently, not taking advantage of the multichannel properties of the speech or noise being recorded as in Rombouts and M. Moonen (2003) and Doclo and Moonen (2002). Being that the microphones are located in unknown places in the room it is considered that no assumptions can be made on the noise or speech properties at this level. The Wiener filtering implementation is taken from ICSI-SRI-UW and used in the ASR system as explained in Mirghafori et al. (2004).

### 5.2.2 Meeting Information Extraction

The algorithms in this block extract information from the input signals, which will be used further on in the process to construct the signal output. It is composed of four algorithms and which described below.

#### Reference Channel Computation

In a typical implementation of a time-delay based beamforming system one needs to select one of the channels as the reference channel. This channel is compared to all others and the time delay of arrival (TDOA) is estimated for each pair. It is important for this channel to be the best representative of the acoustics in the meeting, as the correct estimation of the delays of each of the channels depends on the chosen reference.

In the meetings transcribed by NIST to be used for the Rich Transcription evaluations (*NIST Rich Transcription evaluations, website: <http://www.nist.gov/speech/tests/rt> 2006*) there is one microphone indicated to be the most centrally located in the room. Such microphone is chosen empirically given the room layout and the prior knowledge of the microphone types. This module overpasses that decision and selects one microphone automatically given a criterion based on acoustics. This is intended for system robustness in cases where absolutely no information on the room layout and the microphone placements is available. Two possible acoustic criterions were investigated to select such channel:

- A selection based on **Signal to Noise ratio** (SNR). A simple speech/non-speech de-

tection based on energy is applied to each of the channels independently and the SNR is computed. The channel with better SNR is chosen to be the reference channel. This poses a problem on how accurate is the speech/non-speech detection and how it correlates between channels. The algorithm implementation computed speech/non-speech for each channel independently and then computed the SNR for each one, giving mixed results. An SNR computation using some combined speech/non-speech technique where all channels could be taken into account to come up with one single segmentation could have improved this selection algorithm.

- A selection based on **average cross-correlation** between channels: The cross-correlation (GCC-PHAT) is computed for all possible channel combinations for a block of duration 1s. This is repeated for  $M = 200$  blocks linearly spaced along the recording. For each channel  $i$  the average cross-correlation is computed as:

$$\overline{\text{cross\_correlation}_i} = \frac{1}{MN} \sum_{m=1}^M \sum_{j=1, j \neq i}^N \text{xcorr}(i, j) \quad (5.7)$$

where  $N$  is the number of channels and  $M$  indicates the number of blocks used in the average. In the implementation GCC-PHAT cross-correlation was used as described below. The channel with the highest average cross-correlation was chosen as reference channel. By using this metric it takes into account the amount of time each speaker speaks in total and the quality of each microphone. In the case where all microphones were the same and all speakers spoke the same amount of time, the chosen microphone should be the most physically centrally located one, coinciding with what NIST reports in the RT evaluations.

### Overall Channels Weighting Factor

The input signal to the filter-and-sum module is typically a 16bit, 16KHz signal, and the output being treated by the diarization system is of the same characteristics. By using 16 bits it can represent values from -32767 to +32768 in a single channel in steps of 1 (resolution of the input). Such resolution gets modified when performing the weighted sum of  $N$  signals as the resolution becomes smaller than 1 (the range of possible values of the summed signal depends of the weights of the individual signals, it would be  $\frac{1}{N}$  for equal weighing). Although a higher resolution is available after the sum, the signal needs to be quantized to steps of unit value to fit it into the 16bit output channel, therefore getting a quantization error at each frame.

As the use of a signal output using more bits (like using floating points) creates an inconsistency with the standard signals used in the system and therefore was not considered as feasible, two simple modifications were done in order to minimize the amount of quantization error whenever possible. These are:

- The input signals usually does not cover all the dynamic range used by the 16 bits available (or only a few instants in the meeting do). A scaling factor was defined for all signals so that the sum of them will have a dynamic range closer to the available output, minimizing the quantization errors of the output signal.

There are several alternatives in signal processing to find maximum values of a signal in order to normalize it. Some alternatives are to compute the absolute maximum amplitude over all the show, or the Root Mean Square (RMS) value, or other variations of it involving a histogram of the signal (for example, taking the maximum as the 80% of such histogram).

It was observed that the processed signal contains very low energy areas (silence regions) with short duration in average, and very high energy areas (impulsive noises, like door slams, or common laughs or discussions), with even shorter duration. By using the absolute maximum or RMS it would saturate the normalizing factor to the highest possible value or bias it according to the amount of silence in the show. A windowed maximum averaging was implemented instead in blocks of  $T=10$  seconds to ensure that every block is highly probable to contain some speech. In each block the maximum value is found and averaged over all the recording. Such average is used to obtain the overall weighting factor for the signal in terms of the average maximum of each of the channels as

$$W_{all} = \frac{1}{N} \sum_{n=1}^N \frac{1}{M} \sum_{m=1}^M \max\{x[n + \frac{T(m-1)}{f_s}], \dots, x[n + \frac{Tm}{f_s}]\} \quad (5.8)$$

- The quantization of the output signal is necessary to convert from a floating point value (obtained from the sum of all delayed-weighted-summed signals) to a 16bit signal. It is quantized to the closest integer value within the range  $\pm 32767$ , allowing a maximum quantization error of value  $\pm 0.5$  instead of using the standard functions “int” or “floor” in C, which considers a maximum error of 1.

### ICSI Meetings Skew Estimation

This module was created to deal with all meetings that come from the ICSI Meeting Corpus which have a error in the synchronization of the channels. This was originally detected and reported in *ICSI Meeting Recorder Project: Channel skew in ICSI-recorded meetings* (2006), indicating that the hardware used for the recordings was found not to keep an exact synchronism between the different acoustic channels, having a skew between channels of multiples of 2.64ms. It is not possible to know beforehand the amount of skew of each of the channels as they did not follow a consistent ordering in their connections to the hardware being used, therefore it is needed to automatically detect such skew for it not to affect in the beamforming processing.

The artificially generated skew does not affect the general processing of the channels by an

ASR system as it does not need exact time alignment between the channels (in terms of ms). It does though pose a problem when computing the delays between channels as it introduces an artificial delay between channel pairs which forces to use a bigger analysis window for the ICSI meetings than with other meetings in order to compute such delays accurately, increasing the possibility of delay estimation error and reducing the precision of such values. This module is therefore used to estimate the skew between each channel and the reference channel (in the case of ICSI meetings) and use it as a constant bias in the rest of the delay computations from then on.

In order to estimate the bias a similar technique was used as when estimating the reference channel and the weighting factor. Given signal  $x_i[n]$  to compute the skew for, the cross-correlation is computed of it with the reference signal for  $P = 25$  blocks of 20 seconds each, evenly spaced along the recording. Such segment's length has been determined in order to ensure that there is some speech within the windows being compared. The average skew is obtained for that channel by averaging the time delays of arrival (TDOA) obtained for each of the segments when their cross-correlation function is maximized. The process can be summarized in:

$$Skew_i = \frac{1}{P} \sum_{m=1}^P TDOA_m(x_m^i, x_m^{ref}) \quad (5.9)$$

### GCC-PHAT Cross-Correlation

The computation of the time delay of arrival (TDOA) between each of the considered channels and the reference channel is repeated along the recording in order for the beamforming to respond to changes in the speaker. In this implementation it is computed every 250ms (called segment size or analysis scroll) over a window of 500ms (called the analysis window) which covers the current analysis segment and the next. The size of the analysis window and of the segment size constitute a tradeoff. A big analysis window or segment window lead to a reduction in the resolution of changes in the TDOA. On the other hand, using a very small analysis window reduces the robustness of the cross-correlation estimation, as less acoustic frames are used to compute it. The reduction of the segment size also increases the computational cost of the system, while not increasing the quality of the output signal.

In order to compute the TDOA between the reference channel and any other channel for any given segment it is usual to estimate it as the delay that causes the cross-correlation between the two signals segments to be maximum. In order to improve robustness against reverberation it is normal practice to use the Generalized Cross Correlation with Phase Transform (GCC-PHAT) as presented by Knapp and Carter (1976) and Brandstein and Silverman (1997).

Given two signals  $x_i(n)$  and  $x_j(n)$  the GCC-PHAT is defined as:

$$\hat{G}_{PHAT}(f) = \frac{X_i(f)[X_j(f)]^*}{|X_i(f)[X_j(f)]^*|} \quad (5.10)$$

Where  $X_i(f)$  and  $X_j(f)$  are the Fourier transforms of the two signals and  $[ ]^*$  denotes the complex conjugate. The TDOA for these two microphones is estimated as:

$$\hat{d}_{PHAT}(i, j) = \underset{d}{argmax} (\hat{R}_{PHAT}(d)) \quad (5.11)$$

Where  $\hat{R}_{PHAT}(d)$  is the inverse Fourier transform of Eq. 5.10.

Although the maximum value of  $\hat{R}_{PHAT}(d)$  corresponds to the estimated TDOA for that particular segment, there are three particular cases for which it was considered not appropriate to use the absolute maximum from the cross-correlation function. On one hand, the maximum can be due to a spurious noise or event not related to the speaker active at that time in the surrounding acoustic region, being the speaker of interest represented by another local maximum of the cross-correlation.

On the other hand, when two or more speakers are overlapping each other, each speaker will be represented by a maximum of the cross-correlation function, but the absolute maximum might not be constantly assigned to the same speaker, resulting on artificial speaker switching. In order to effectively enhance the signal it would be optimum to first detect when more than one speaker is speaking at the same time and then obtain a filter-and-sum signal for each one, stabilizing the selected delays and avoiding them from constant speaker switching. Due to a lack of an efficient overlap detector, this was not implemented in this thesis and remains as future work.

Also, when the segment that has been processed is entirely filled with non-speech acoustic data (either noise or random acoustic events) the GCC-PHAT function obtained will not be at all informative. In such case no source delay information can be extracted from the signal and the delays ought to be discarded and substituted by something more informative.

In the system implementation, to deal with such issues, the top M relative maximums in eq. 5.11 are computed and several delay post-processing techniques are implemented to stabilize and choose the appropriate delay before aligning the signals for the sum. These are described below:

### 5.2.3 TDOA Values Selection

Once the TDOA values of all channels have been computed, we have seen above that it is desirable to apply a TDOA post-processing to obtain the set of delay values to be applied to each of the signals when performing the filter-and-sum as proposed in eq. 2.33. We propose and implement two different filtering steps: Noisy TDOA detection and elimination (TDOA continuity enhancement), and 1-best TDOA selection from the M-best computed vector.

#### TDOA Post-Processing

The first filtering proposed intends to detect those TDOA values that are not reliable. A TDOA value does not show any useful information when it is computed over a silence (or mainly silence) region or when the SNR of the signals being compared (either one) are very low, making them very dissimilar. The first problem could be addressed by using a speech/non-speech detector prior to any further processing. Initial experiments indicated that further errors were introduced due to the detector used. An improvement was obtained by applying a simple continuity filter on the TDOA values based on their GCC-PHAT values by using a “noise threshold”:

$$TDOA_i[n] = \begin{cases} TDOA_i[n-1] & \text{if } GCC\text{-}PHAT_i[n] < Thr_{noise} \\ TDOA_i[n] & \text{if } GCC\text{-}PHAT_i[n] \geq Thr_{noise} \end{cases} \quad (5.12)$$

where  $Thr_{noise}$  is the “noise threshold”, defined as the minimum correlation value at which it can be considered that the correlation is returning feasible results. It should be considered independently in every meeting as the correlation values are dependent not only on the signal itself but also on the microphone distribution in the different meeting rooms. In order to find an appropriate value for it, the distribution of computed correlation values needs to be evaluated for each meeting. In the diarization system presented for RT05s (Anguera, Wooters, Peskin and Aguilo 2005) a constant threshold was fixed for all meetings. This caused the system to filter out a high amount of delays in some meetings while keeping non-speech segments unmodified from others. For RT06s a threshold was computed for each meeting as the value that filters out the 10% of lower cross-correlation values. This considers that in each meeting there are 10% of frames that either are non-speech or unreliable in terms of TDOA estimation.

Figure 5.4 shows the histogram of the two AMI meetings present in RT05s to illustrate this change. Such histograms are generated taking the output values from the GCC-PHAT for the used TDOA values, placing them in bins with minimum value 0 and maximum 1, and normalizing it. Most of the meetings present a bimodal histogram like for AMI\_20041210-1052, where the relative minimum between the modes falls around the 10% of the values. In such case selecting a noise threshold at 10% absolute (0.1 value applied to the GCC-PHAT output) or finding the threshold at 10% of all computed values (0.0842 over 1 for AMI\_20041210-1052) gives almost

the same result. On the other hand, some meetings, like AMI\_20050204-1206, obtain a poor distribution of GCC-PHAT values, concentrating them in the lower part of the histogram. In this case there is a big difference between the two kinds of thresholding (0.1 versus 0.0532 for AMI\_20050204-1206).

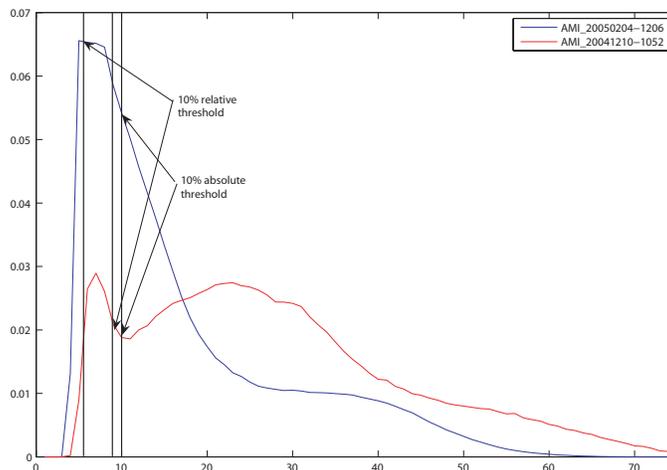


Figure 5.4: *Cross-correlation values histograms for RT06s AMI meetings*

Within each meeting there is also a slight difference in the distributions of each of the channel's correlations. It was found that there was no difference whether to compute an individual threshold for each channel or one global threshold for all channels, therefore a global threshold was used for all channels in the system.

### Dual-Pass Viterbi Post-Processing

The second post-processing technique applied to the computed delays is used to select the appropriate delay to be used among the M-best GCC-PHAT values are computed at each step. As pointed out previously, the aim here is to maximize speaker continuity avoiding constant delay switching in the case of multiple speakers, and to filter out undesired steering towards spurious noises present in the room.

As seen in figure 5.5 a 2-level Viterbi decoding of the M-best TDOA computed was applied. The first level consists of a local individual-channel decoding where the 2-best delays are chosen from the M-best delays computed for that channel at every segment. The second level of decoding considers all combinations of such 2-best across all channels and selects the final single TDOA that are more consistent across all. For each step one needs to define the topology of the Viterbi algorithm and the emission and transition probabilities to be used. The selection of a 2-step algorithm is due in part to computational constraints as an exhaustive search over all

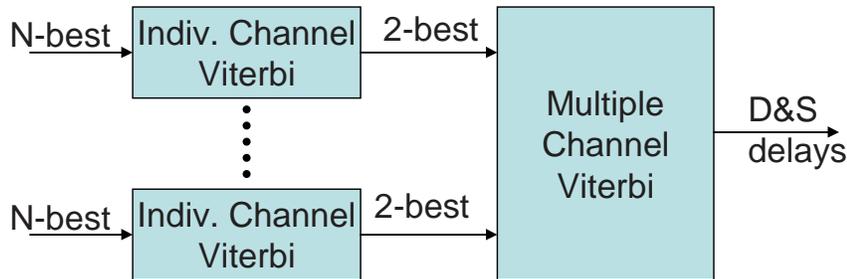


Figure 5.5: *Filter and Sum double-Viterbi delays selection*

possible combinations of all  $M$ -best values for all channels would easily become computationally prohibitive.

Both steps choose the most probable (and second most probable) sequence of hidden states where each item is related to the TDOA values computed for one segment. In the first step the set of possible states at each instant is given by the computed  $M$ -best values. Each possible state has an emission probability for each processed segment, equal to the GCC-PHAT value for each delay ( $P_1^m[c]$ , where  $m$  is the  $m$ -best value being considered and  $c$  is the current segment).

The transition probability between two states is taken as the inverse proportional to the distance between its delays. Given two nodes,  $i$  and  $j$  at segments  $c$  and  $c - 1$ , respectively, the transition probability between them is

$$Tr_1(i, j)[c] = \frac{\max\_diff(i, j) - |\text{TDOA}_i[c] - \text{TDOA}_j[c - 1]|}{\max\_diff(i, j)} \quad (5.13)$$

where  $\max\_diff(i, j) = \max(|\text{TDOA}_i[c] - \text{TDOA}_j[c - 1]|, \forall i, j)$ . This way all transition probabilities are locally bounded between 0 and 1, assigning a 0 probability to the furthest away delays pair.

This first Viterbi level aims at finding the best two TDOA values that represent the meeting's speakers at any given time. By doing so it is considered that the system will be able to choose the most appropriate/stable TDOA for that segment and a secondary delay, which can come from interfering events, other speakers or the same speaker's echoes. Such TDOA values are any two (not allowing the paths to collapse) of the  $M$ -best computed previously by the system, and are chosen exclusively based on their distance to surrounding TDOA values and their GCC-PHAT values.

The second level Viterbi decoding finds the best possible path given the set of hidden states generated by all possible combinations of delays from the 2-best delays obtained earlier for each channel. Given the vector of dimension  $N - 1$  (same as the number of channels for which

TDOA values are computed) describing for each channel which TDOA value is being used  $\mathbf{g}^i[c] = [g_1^i[c] \dots g_{N-1}^i[c]]$  with each  $g_n^i[c]$  indicating the  $i$  position among the 2-best list of TDOA values considered for channel  $n$  at segment  $c$ . Given also that any given  $\text{xcorr\_phat}_n^{g_n^i[c]}[c]$  value (the GCC-PHAT value associated to the  $g_n^i[c]$ -best TDOA value for channel  $n$  at segment  $c$ ) will take values  $[0, 1]$ , the emission probabilities are considered as the product of the individual GCC-PHAT values of each considered TDOA combination  $\mathbf{g}^i[c]$  at segment  $c$  as

$$P_2(i)[c] = \sum_{n=1}^N \log(\text{xcorr\_phat}_n^{g_n^i[c]}[c]) \quad (5.14)$$

which can be considered as the extension of  $P_1(i)[c]$  to the case of multiple TDOA values where we consider that the different dimensions are independent from each other (interpreted as independence of the TDOA values obtained for each channel at segment  $c$ , not their relationship with each other in space along time).

On the other hand, the transition probabilities are also computed in a similar way as in the first step, but in this case they introduce a new dimension to the computation, as now a vector of TDOA values needs to be taken into account. As it was done with the emission probabilities, the total distance is considered as the sum of the individual distances from each element. Given  $\text{TDOA}(g_n^i, n)[c]$  as the TDOA value for the  $g_n^i[c]$ -best element in channel  $n$  for segment  $c$ , the transition probability between TDOA position vectors  $i$  and  $j$  is determined by

$$\text{Tr}_2(i, j)[c] = \sum_{n=1}^N \frac{\text{max\_diff}(i, j, n) - |\text{TDOA}(g_n^i, n)[c] - \text{TDOA}(g_n^j, n)[c]|}{\text{max\_diff}(i, j, n)} \quad (5.15)$$

where now  $\text{max\_diff}(i, j, n) = \max(|\text{TDOA}(g_n^i, n)[c] - \text{TDOA}(g_n^j, n)[c]|, \forall i, j, n)$ .

This second level of processing considers the relationship in space present between all channels, as they are presumably steering to the same point in space. By performing a decoding over time it selects the TDOA vector elements according to their distance to the vectors in its surroundings.

In both cases the transition probabilities are weighted to emphasize its effect in the decision of the best path in the same way as in the ASR systems (by product in the log domain). It will be shown in the experiments section that a value of weight 25 for both cases is what optimized the diarization error given the development set.

To illustrate how the two-step Viterbi decoding works on the TDOA values let us consider figure 5.6. It shows a situation where four microphones are used in a room where two speakers are talking to each other, with some overlap speech. There is also one or more noisy events of short duration and noise room in general, both represented by a “noise” source. Given one of

the microphones as a reference, the delay to each of the other microphone is computed, resulting in delays from speech coming from either speaker ( $D(s[1, 2], m)$ ) or from any of the noisy events ( $D(nx, m)$ ) with  $m = 1 \dots 3$ .

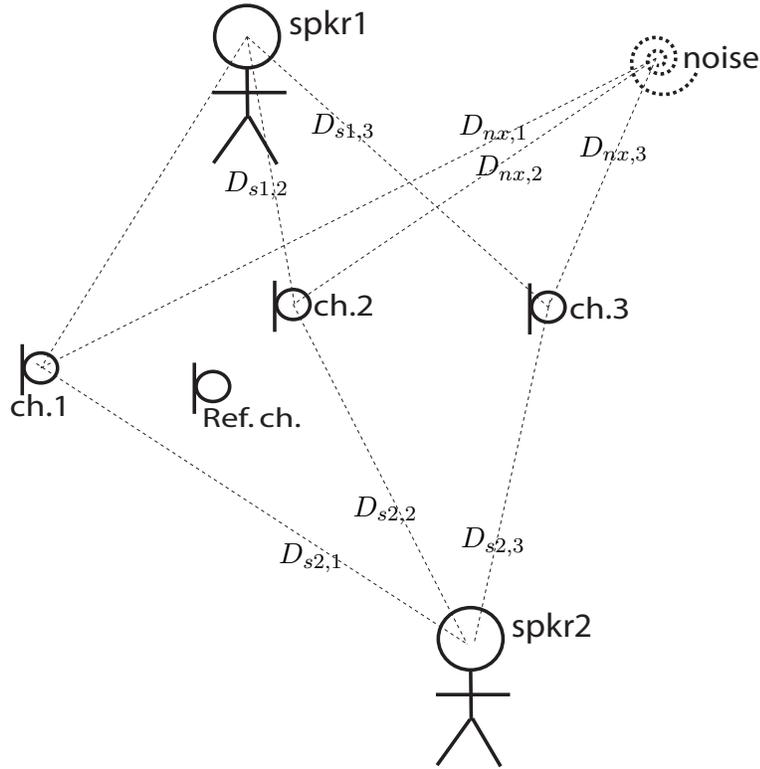


Figure 5.6: *Two-step TDOA Viterbi decoding example, step 1*

For a particular segment the M-best TDOA values from the GCC-PHAT cross correlation function are computed. The first Viterbi step determines for each individual channel the 2-best paths across time for all the meeting. Figure 5.7 shows a possible Viterbi trellis for the first step for channel 1, where each column represents the M-best TDOA values computed for one segment. In this example four segments were considered where two speakers are overlapping each other, and there is also some eventual noisy events. For any given segment the Viterbi algorithm finds the two-best paths (forced not to overlap with each other) according to their distance of the delays to the chosen delays in the neighboring segments (transition probability) and to their cross-correlation values (emission probability). The resulting paths could be:

1. best path:  $D_{s1,1}, D_{s2,1}, D_{s1,1}, D_{s1,1}$
2. second path:  $D_{s2,1}, D_{s1,1}, D_{s2,1}, D_{s2,1}$

The third computed segment contains a noisy event that is well detected by channel 1 and the reference channel, and therefore it appears as the first in the M-best computed TDOA list.

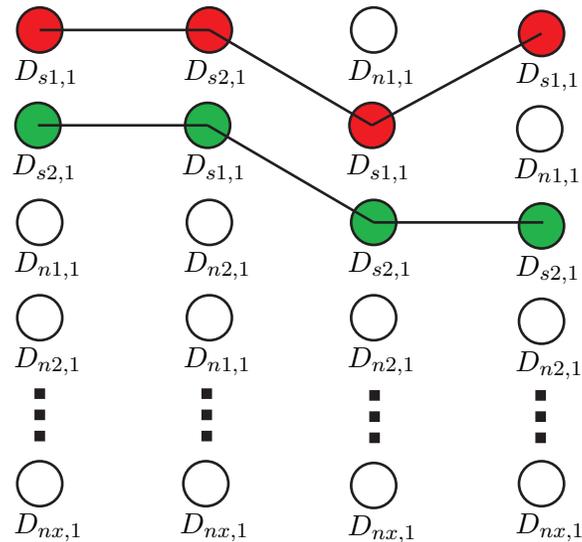


Figure 5.7: *Two-step TDOA Viterbi decoding example, step 1 for an individual channel*

The effect of the Viterbi decoding can avoid selecting this event as its delay differs too much from the best delays in its surroundings and both speakers also appear with high correlation. On the other hand, the first and second segments contain the delays referring to the true speakers in the first and second-best positions, although alternated in both segments. This example illustrates a possible case where they cannot be correctly ordered and therefore there is a quick speaker change in the first and second-best delay paths in that segment.

The second step Viterbi decoding is intended to add an extra layer of robustness for the selection of the appropriate delays by considering all the possible delay combinations from all channels. Figure 5.8 shows the trellis formed by considering for each segment (in columns) all possible combinations of  $m$ -best delays ( $\mathbf{g}_n^i[c]$ ) for the 3 channels case.

In this step only the best path is selected according to the overall combined distances and correlation values among all possible combinations. In this example the algorithm is able to solve the order mismatch from the previous step, selecting the delays relative to speaker 1 for all the segments. The current computes the 2-best path also in this step and output a signal steering at the two sets of TDOA values, although the diarization algorithm only use the first of them. In order to take advantage of the second (or more) delays steering at the overlap speakers in the meeting it is necessary to achieve some more progress in reliable speaker overlap detection algorithms, which remains as future work at the end of this thesis.

In the implementation of the second level Viterbi decoding a big burden in computation time could be faced depending on the amount of microphones to be processed. In the second level Viterbi the amount of possible states for each instant  $k$  is defined by

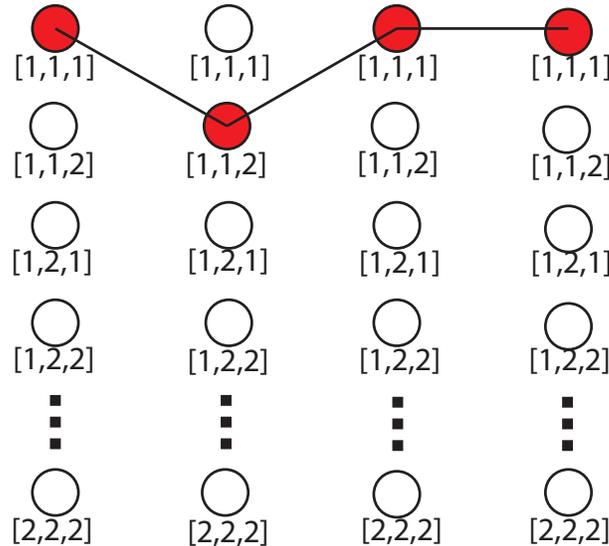


Figure 5.8: *Two-step TDOA Viterbi decoding example, step 2*

$$S[k] = M_2^{N-1} \quad (5.16)$$

where  $M_2$  is the number of best TDOA values extracted from the  $M$ -best values in the first Viterbi level (in this implementation  $M_2 = 2$ ). As the amount of states grows exponentially when increasing  $D$ , it becomes computationally prohibitive for meetings with 16 or more microphones available (for  $N = 17$ ,  $M_2 = 2$ ,  $S[k] = 65536$ ). For a feasible implementation, when  $N > 5$  the pool of microphones are split in blocks of 5 and the Viterbi is computed in each block independently. This is a suboptimal solution as not all microphones are used to optimize the delays and therefore it is not certain that all blocks will converge to the same solution. It is though much faster in processing time and it was not observed to degrade the overall performance compared to using all microphones together.

In conclusion, this newly-introduced double-Viterbi technique aims at finding a good tradeoff between reliability (cross-correlation) and stability (distance between contiguous delays). The second is cherished the most as the aim is to obtain an optimally improved signal, avoiding quick changes of the beamforming between acoustic events.

#### 5.2.4 Output Signal Generation

Once all information is computed from the input signals and the optimum TDOA values are selected, it is time to output the enhanced signal and any accompanying information to be used by the subsequent systems. In this module several algorithms were used to account for the differences between the standard linear microphone array theory and the implementation in this

module.

### Automatic Channel Weight Adaptation

In the typical formulation of the filter&sum processing, the additive noise components on each of the channels are expected to be random processes with very similar power density distributions. This allows the noise on each channel to be statistically cancelled and the relevant signal enhanced when the delay-adjusted channels are summed. In standard beamforming systems, this noise cancellation is achieved through the use of identical microphones placed only a few inches apart one from each other.

In the meetings room it is considered that all of the distant microphones form a microphone array. However, by having different types of microphones there is a change in the characteristics of the signal being recorded and therefore a change in the power density distributions of the resulting additive noises. Also when two microphones are far from each other, the speech they record will be affected by noise of a different nature, due to the room's impulse response, and will have different amplitude depending on the position of the speaker talking.

This issue is addressed by weighting each channel in the filter&sum processing. The weights are adapted continuously during the meeting. This is inspired by the fact that the different channels will have different signal quality depending on their relative distance to the person speaking, which probably changes constantly during a recording.

The weight for channel  $n$  at segment  $c$  ( $\mathcal{W}_n[c]$ ) is computed in the following way:

$$\mathcal{W}_n[c] = \begin{cases} \frac{1}{N} & c = 0 \\ (1 - \alpha) \cdot \mathcal{W}_n[c - 1] + \alpha \cdot \text{avecorr}_n[c] & \text{otherwise} \end{cases} \quad (5.17)$$

where  $\alpha$  is the adaptation ratio, which was empirically set to  $\alpha = 0.05$ ,  $c$  is the segment being processed, and  $\text{avecorr}_n[c]$  is the average of the cross-correlation between channel  $n$  and all other channels being all delayed using the selected  $TDOA_n[c]$  value for that channel:

$$\text{avecorr}_n[c] = \frac{1}{N - 1} \sum_{j=1, j \neq n}^{N-1} \sum_{k=0}^W x_n[cS - TDOA_n[c] - k] x_j[cS - TDOA_j[c] - k] \quad (5.18)$$

where  $S$  and  $W$  are respectively the scroll/segment size and window size of the filter&sum processing

### Automatic Adaptive Channel Elimination

Although efforts are made to ensure that the TDOA values assigned to each of the channels are correct, in some cases the signal of one of the channels at a particular segment is itself of such low quality that its use in the sum would only degrade the overall quality. This usually happens when the quality of one or more microphones is very different from the others (for example the PDA microphones in the ICSI meeting room recordings as explained in Janin, Ang, Bhagat, Dhillon, Edwards, Macias-Guarasa, Morgan, Peskin, Shriberg, Stolcke, Wooters and Wrede (2004)).

In the filter&sum processing all available microphones in the room are used and a dynamic selection and elimination of the microphones that could harm the overall signal quality at every particular segment is performed. The previously defined  $avecorr_n[c]$  is used to determine the channel quality. If  $avecorr_n[c] < \frac{1}{4N}$  then  $\mathcal{W}_n[c] = 0$ . After checking all the channels for any elimination the weights are readapted to sum up to 1.

### Channels Sum and Output

Once the output weight has been determined for each channel at a particular segment, all the signals are summed up to form the output “enhanced” signal. Such output signal needs to be ensured acoustic continuity at all times. In the theoretical filter&sum equation as shown in eq. 2.33 it will causes discontinuities in the signal on the segments edges due to the mismatch between the summed-up signals on the edges between segments.

A triangular window is therefore used to smooth and reduce the discontinuity between any two segments, as seen in figure 5.9. At every segment the triangular filter smooths the signal delayed using that segment’s selected TDOA value with the signals delayed using the TDOA values from the previous segment. By using the triangular window the system obtains a constant total value without discontinuities. The actual implementation follows equation 5.19.

$$y[cS + k] = W_{overall}(\alpha[k] \sum_{n=1}^N w_i[c] x_i[cS + k - TDOA_i[c]] + (1 - \alpha[k]) \sum_{n=1}^N w_i[c] x_i[cS + k - TDOA_i[c - 1]]) \quad (5.19)$$

where  $S$  is the segment sample length,  $c$  is the segment being processed and  $k$  is the sample within that segment being processed.

In the standard implementation the analysis window overlaps 50% with the segment window, which agrees with the triangular overlap of 50% overlap done here. After all samples from both

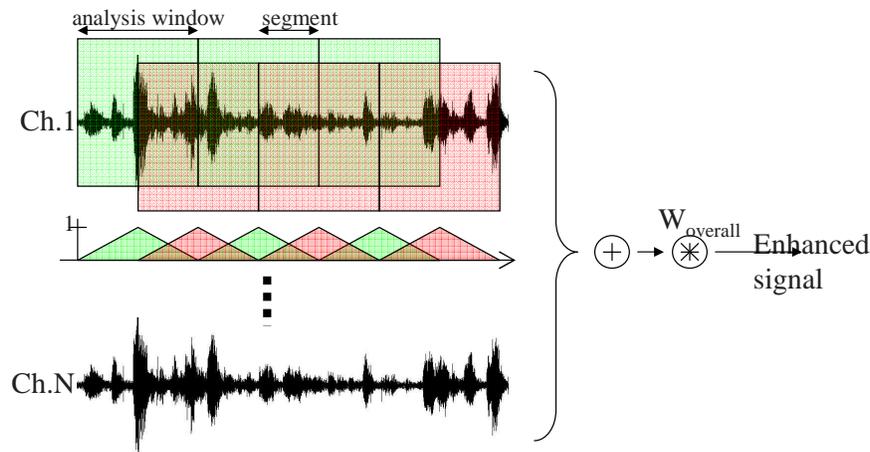


Figure 5.9: Multichannel delayed signal sum using a triangular window

overlapping windows are summed the overall weighting factor computed earlier is applied to ensure that the dynamic range of the filter&summed signal is optimally matched with the available dynamic range of the output file. The resulting signal is further processed by the speaker diarization system which is described in the next chapter. Together with the acoustic signal, also the TDOA values used in the channel delays are written to an ASCII file for use by the Diarization system as features and used as explained in the next section.

### 5.3 Use of the Estimated Delays for Speaker Diarization

Performing an acoustic beamforming of the multiple input signals has multiple advantages, including the simplicity of the following speaker diarization system, which can be reused from the broadcast news system as it only needs to compute the output for a single acoustic channel. Another advantage is the independence of the proposed system to the room layout and number of microphones.

By doing an acoustic beamforming there comes a drawback in that all spatial information about the speaker location, which is carried by the multiple microphones in the room, is lost in the process. For this reason when multiple microphones are available (and therefore a beamforming is performed) the speaker location information is reused for the speaker diarization module. Such information comes from the Time Delay of Arrival (TDOA) values between each microphone and the reference channel. Although extensive research has gone into speaker localization using multiple microphones (including the identification of each speaker from the others), this is only possible when the topology and exact location of all microphones is known in advance. This is not considered in the current implementation of the system as multiple room topologies are to be processed and, for some of them, the microphones locations are not known.

Apart from the TDOA values, there are other features that could be useful to determine the difference between speakers. One such possibility is the relative amplitude between the different channels, which should be able to identify whoever is closer to what microphone, being therefore an indicator of location of the speakers. This metric is though very correlated to the TDOA values, suffering from the same problems, and therefore has not been considered in this thesis. Further study should be done to indicate whether using both information streams could lead to further improved speaker diarization.

### 5.3.1 TDOA Modeling and Features Fusion

For any given set of  $N$  channels at frame  $n$  ( $x_1[n] \dots x_N[n]$ ) the beamforming system determines a single vector  $TDOA[n]$  with dimension  $N - 1$  obtained from the best TDOA values between each microphone and the reference. The dimension of the TDOA vector will change depending on the number of microphones available. This does not indicate a priori that TDOA vectors with lower dimension will be able to discriminate worse between speakers, as it depends not only on the number of microphones but also on the acoustic properties of the room (influencing how accurate the TDOA values are), the microphones topology and the location of the speakers.

For any particular frame, the  $TDOA[n]$  vector will contain a set of TDOA values that identify the location of the main acoustic source towards where the acoustic beamforming is steering. To exemplify this, figure 5.10 shows the histograms and X-Y plot of the first two dimensions of the TDOA vectors extracted for all speech frames in the show ICSI\_20000807-1000 containing six speakers (which is the actual number of participants in that meeting). The histograms show the existence of around 6 speakers, being some of them closer together than others. In the X-Y plot the higher density places indicate higher probability of speakers. The points that fall far from any of the speakers are due to silence regions not eliminated by the post-processing step in the beamforming, or by acoustic events other than speakers (pen drops, door slams, etc). There are also some vectors falling along one of the speaker axes indicating that a speaker is most probably active during that frame instant but the different dimensions do not agree. This can be due to overlap regions where each microphone points at a different speaker or errors in the TDOA approximation. This problem is common when computing TDOA values and is one of the issues addressed by the double-Viterbi post-processing algorithm. The remaining TDOA vectors not detected by the post-process tend to cause errors in the diarization algorithm by causing models to fit such data as an independent speaker.

The use of delays for speaker diarization using the presented diarization system was initiated by J.M. Pardo and presented in Pardo et al. (2006a). Later, the same proposed in Pardo et al. (2006b) the combination of TDOA values and acoustics in order to improve results even more. Also at ICSI some work by Gallardo-Antolin, Anguera and Wooters (2006) shows other features

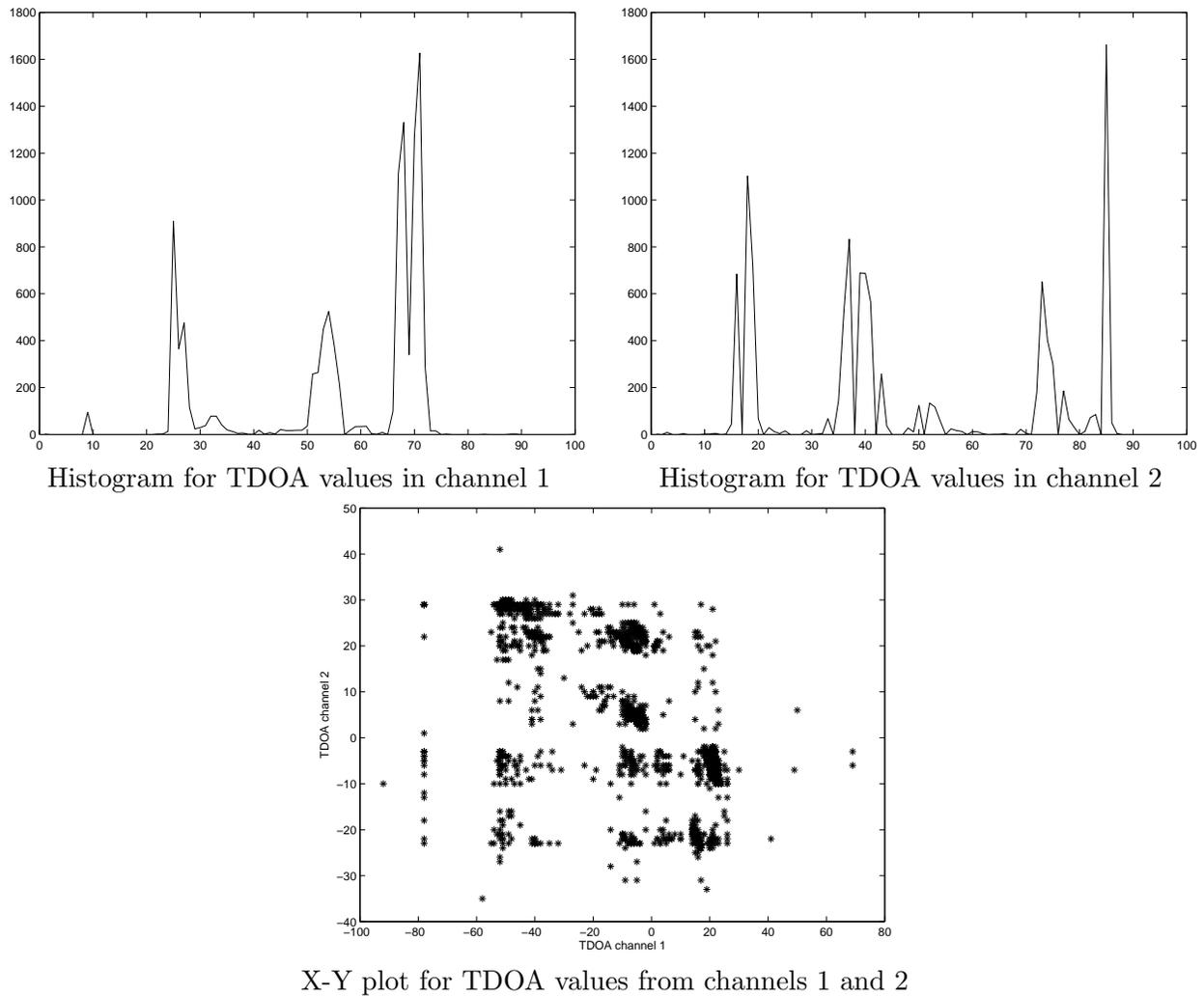


Figure 5.10: *Locations information contained in the TDOA values*

fusion alternatives other than the TDOA values.

In order to use these delays vectors to add extra information in the speaker diarization module they are treated as a feature vector and modeled by a GMM. As used in Lathoud, McCowan and Odobez (2004), a single Gaussian is used to model the clusters initially created in the diarization system. In figure 5.11 it indicates the way that features computed from the acoustic signal and the TDOA values are fused.

Upon starting the diarization two feature streams are available for processing, the acoustic stream (which is composed of 19 MFCC features, computed every 10ms) and the TDOA stream, computed in the beamforming module. In theory the same TDOA values that are used for the beamforming process can be reused in this module, but in practice, in order to obtain synchrony between acoustics and TDOA values, they are recomputed every 10ms. Use of the same TDOA values was also tested by repeating the same values several times (25 times for 250ms scroll)

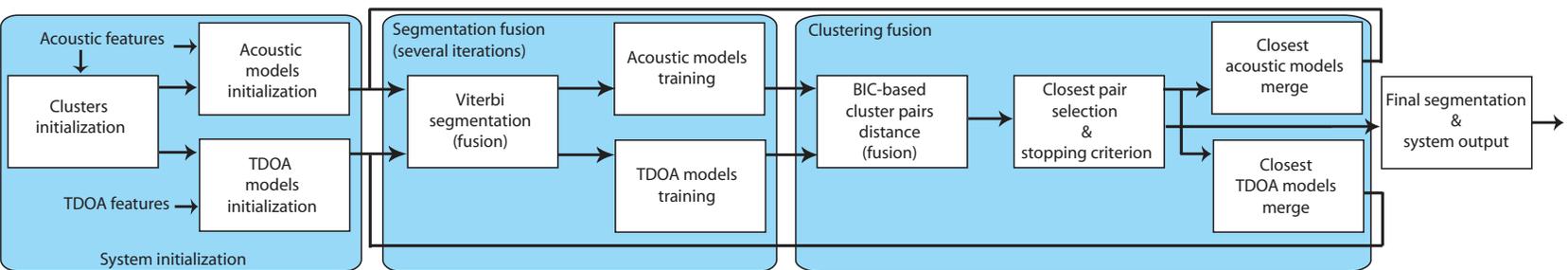


Figure 5.11: Fusion of TDQA values and acoustic features within the speaker diarization module

with slightly worse (but acceptable) results, showing its feasibility in case of computational constraints.

In order to process the signal using both feature streams the system maintains two different/independent HMM speaker model sets and keeps the same speaker clustering, which gets defined using both streams. The speaker models use the same structure as in the standard system (an ergodic HMM) and share the number of speakers, defining a model pair for each speaker cluster, but can be represented using a different complexity, depending on the optimum way that the data in each stream should be modeled.

The first step in the system is to initialize the  $K$  initial speaker clusters. This entails splitting the input data among these  $K$  clusters. This is currently done in the same way as in the standard system, using solely the acoustic data stream. Once an initial clustering is defined, the initial models are created both for the acoustics and the TDOA values and the system enters the segmentation/training step. Both speaker models are used in the Viterbi decoding to determine the optimum path among the different speaker clusters by considering the joint log-likelihood for any given frame as

$$\mathcal{L}(x_{aco}[n], x_{del}[n]|\Theta_{aco}, \Theta_{del}) = W_1 \cdot \mathcal{L}(x_{aco}[n]|\Theta_{aco}) + W_2 \cdot \mathcal{L}(x_{del}[n]|\Theta_{del}) \quad (5.20)$$

where  $\Theta_{aco}$ ,  $x_{aco}[n]$  is the acoustic model and data,  $\Theta_{del}$ ,  $x_{del}[n]$  is the delay model and data, and  $W_1, W_2$  weight the effect of each stream in the decoding, given that  $W_1 + W_2 = 1$ . In this formulation it is considered each stream to be statistically independent from each other, which is a plausible consideration given that acoustic and TDOA information convey very different information. If more feature streams are available, this formulation can be expanded with each feature likelihood being weighted by a different  $W_i$ . When running the Viterbi decoding a minimum duration for a speaker segment is set to  $MD = 3$  seconds (optimized for the development data) for both models in order to avoid constant changes in the clustering. Once a new speaker clustering is defined, the models are retrained independently.

The second step where the feature streams fusion takes place is in the clustering step where the closest cluster pair is selected and the clusters and models are merged (or the processing finishes if the stopping criterion decides so). As explained in 3.1 the cluster pair comparison metric of choice is a variant of the BIC metric where the penalty term is eliminated by constraining the complexity of the different models being compared. In this particular case the formulation for the BIC contemplating the fusion between both streams can be defined directly from equation 5.20 as

$$\Delta BIC(A, B) = W_1 \Delta BIC_{aco}(A, B) + W_2 \Delta BIC_{del}(A, B) \quad (5.21)$$

where  $A, B$  are two clusters we want to compute the distance for, and  $W_1, W_2$  are the same weights as in eq. 5.20. This can also be directly expanded to use more than 2 streams.

If frame or segment purification are to be applied, these are done so only using the acoustic frames. This is so in the case of frame purification because the TDOA models react in a different way to the non-speech data than the acoustic models.

The same stopping criterion as in the regular system is used. While the system does not determine to stop the clustering process, the closest cluster pair is selected and merged, together with the models belonging to such cluster. In the case of the TDOA models the merging is done by overlapping both existing models and retraining the overall model using all the data from both clusters. In the case of the acoustic models it is either done in the same way as just explained or it is modified according to the determined new complexity for the resulting model.

Whenever the system determines to stop clustering, a final Viterbi decoding is performed using again both frame streams, with a smaller minimum duration, as explained in the meetings system in section 3.3.

### 5.3.2 Automatic Features Weight Estimation

As seen in equations 5.21 and 5.20, in order to combine the acoustic and TDOA features one needs to determine an optimum set of weights  $W_i$  that define how relevant each one is. Without an automatic way to determine such value it needs to be found using development data and performing a sweep of the  $W_i$  parameters optimizing the Diarization Error Rate (DER) score. This constitutes a problem of robustness due to the possible big differences between the development and test sets in terms of the relative importance between features. It also becomes a tedious job if the number of parallel feature streams is big. Some of the factors that reduce the ability of a feature set to optimally represent the speakers in a recording (and therefore its relevance should be reduced) are:

- Acoustics from speakers with very similar voices.
- TDOA values for people very close together.
- TDOA values for people in symmetry with most of the microphones.
- TDOA values in noisy environments (where approximation of the correct TDOA value is difficult) or with multiple impulsive noises.
- TDOA values when the speakers are moving around.

When setting the values by hand they are normally defined for all meetings equally and therefore they do not account for peculiarities due to the meeting room (noisier rooms) or to

the nature of the meetings (kind of usual attendees or whether they move from their seats). The automatic weight setting algorithm presented here is able to compute the optimum values for each meeting independently.

Prior art in weights selection for features fusion needs to be searched for in areas other than speaker diarization, like in speaker verification and biometric fusion techniques (Fierrez-Aguilar, Ortega-García and González-Rodríguez (2003), Ross, Jain and Qian (2001), Verlinde, Chollet and Acheroy (2000)) and in speech recognition (Misra, Boulard, and Tyagi (2003), Iqbal, Misra, Sivadas, Hermansky, and Boulard (2004), Li (2005)). Throughout the literature a well used technique for automatic weighting of different feature streams is based on the feature vectors entropy.

Initial tests were performed using the inverse entropy as relative weight to see how discriminant each feature stream was. This was done by obtaining the weights in a frame-basis via the inverse entropy of the posterior probabilities of the cluster models given the data. For MFCC, PLP and other acoustic features these entropies were comparable to each other and could therefore determine a correct relative weight between features, as shown in Misra et al. (2003). When using it with TDOA values their GMM models are such that low entropy values are obtained for almost every frame, regardless of how accurate the TDOA values can represent a real speaker position.

The proposed technique in this thesis uses the Bayesian Information Criterion (BIC) to compare how well each feature stream differentiates between clusters in order to determine an appropriate stream weighting. The  $\Delta$ BIC values are independent of the complexity and topology of the models being used and are a good indication of how close two clusters are.

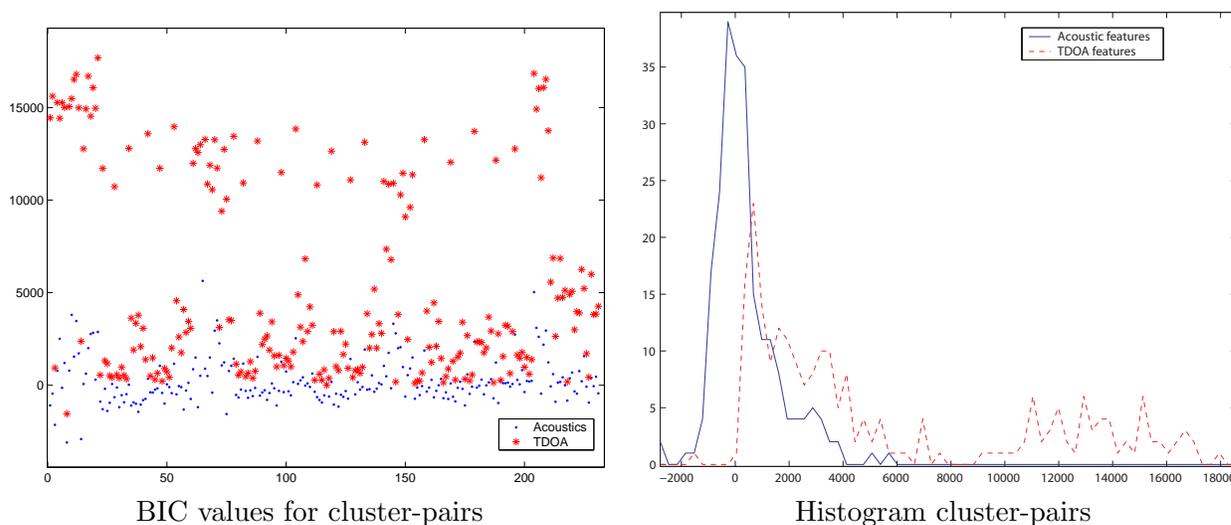


Figure 5.12: First merge cluster-pair BIC values and histogram for acoustic and TDOA features

Given the  $\Delta\text{BIC}$  values between all cluster pairs for the acoustic and TDOA models, figure 5.12 shows the values and their histograms for the meeting EDI\_20050216-1051 from the RT06s evaluation data set, computed for all pairs (given 22 initial clusters) for the first iteration of the clustering. The TDOA values are much bigger in average and contain more positive values than the acoustic values. If a weight  $W_1 = 0.5$  (equal relevance) is considered, the TDOA BIC values would mask the acoustics and decide which pair to merge, possibly leading to errors as not all the information is considered. In order to allow for different feature streams to contribute in equal conditions in the merging decision it is needed to transform both  $\Delta\text{BIC}$  value sets to have the same scale using the  $W_1$  weight. This way the TDOA values with overall high  $\Delta\text{BIC}$  are penalized versus the acoustic values in order to be comparable to each other. For a general case of  $M$  feature streams, the weight  $W_i$  assigned to each stream  $i$  is defined as

$$W_i = \frac{\frac{1}{\sqrt{P_i}}}{\sum_{j=1}^M \frac{1}{\sqrt{P_j}}} \quad (5.22)$$

where  $P_i$  is computed from the  $N$   $\Delta\text{BIC}$  values computed for all cluster pairs  $x_j, x_k$  from each feature stream as

$$P_i = \frac{1}{N} \sum_{j=1}^{j=N-1} \sum_{k=j+1}^{k=N} \Delta\text{BIC}_i^2(x_j, x_k) \quad (5.23)$$

This process is equivalent to a variance normalization of single Gaussians modeling each feature stream with zero mean. Setting the mean to zero avoids moving the decision threshold in the  $\Delta\text{BIC}$  comparison, as defined by the BIC theory.

The automatic computation of the  $W_i$  weight is performed at the first clustering step, when the  $\Delta\text{BIC}$  values are computed. At the initial segmentation step, no weight has been automatically defined and therefore some initial weight still needs to be determined by hand, or it can be set to an uninformative  $W_1 = W_2 = 0.5$ .

On subsequent clustering iterations the models usually represent the clusters better and obtain  $\Delta\text{BIC}$  values which are more accurate. In order to allow the system to refine the weight as the merging iterations progress, the  $\Delta\text{BIC}$  values are kept for all cluster pairs that disappeared during previous iterations and existing pairs are recomputed. Then a new weight is computed taking into account both old and updated values in order to allow for a weight adaptation, containing enough samples for a robust computation.

To illustrate the effect of the weight adaptation as the system iterates, figure 5.13 shows the evolution of the  $W_i$  weight over the initial 10 iterations of the algorithm for meeting CMU\_20050912-0900 (in the RT06s data set). It is common on all meetings to start with big-

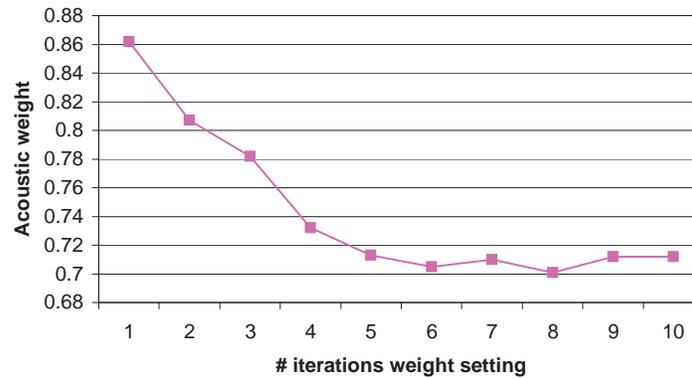


Figure 5.13: *Acoustic weight evolution with the number of iterations for meeting CMU\_20050912-0900*

ger values for the acoustic part and to see it reduced overtime and converging to a final value ( $W_1 = 0.71$  in this case, converging after 5 iterations). The optimum weight always enhances the acoustic values versus the TDOA values for all shows, both when computed automatically or manually. By doing it automatically each show obtains its own optimum value, which would had been set to  $W_1 = 0.9$  manually for RT06s set (including this meeting).

In the experiments section the effect of the number of iterations in which the weight is computed versus the final DER score is computed. It is found that weights always converge to constant values with optimum DER values, therefore leading to a robust solution with one less tuning parameter.



## Chapter 6

# Experiments

This chapter verses about the experimentation of the different proposed techniques in order to evaluate its suitability in the task of speaker diarization for meetings. It is done by first defining a baseline system to compare all algorithms to. Such baseline system is derived from the broadcast news mono-channel system with several improvements that were considered standard and necessary to the system as adapted to meetings.

Then a set of metrics used in the evaluation of the different techniques are described in detail. Next, the databases that are used to compare the algorithms performance with that of the baseline and the reference segmentations which are used in the experiments are explained and reasoned. Finally, the different experiments with the proposed algorithms are performed and results are explained.

### 6.1 Meetings Domain Experiments Setup

When comparing the results of new speech-related algorithms it is usual to always face some sort of “flakiness”. This term started being used for speaker diarization during the RT04f workshop (*NIST Fall Rich Transcription Evaluation website 2006*) in order to account for two phenomena that were common to all diarization systems presented in that evaluation. These were the big variance of the scores among all evaluated shows and the extreme susceptibility of the scores to experience big changes upon small modifications of their tuning parameters.

Alike some other disciplines within the speech technologies, it makes a difference, when comparing the performances of algorithms compared to a baseline, to select the optimum baseline databases and test conditions to be able to show when the proposed algorithms preform the best. In many cases, due to flakiness, testing the same algorithms with two different databases or baseline systems derives into two very different results, one proving the validity of the proposed algorithm and one otherwise.

In order to run meaningful and fair experiments using the algorithms proposed in this thesis one needs to define:

- A baseline system, which acts as the comparison ground to all systems proposed and tested.
- A common development and test datasets, based on the NIST RT evaluations datasets, in order for results to be comparable between experiments and to systems outside of the thesis.
- A set of metrics in order to evaluate such systems with commonly used and available techniques.

In the following subsections each of these items is described as it has been used in this thesis for most of the experiments with the system's main blocks.

### 6.1.1 Baseline Systems

Taking as a reference the blocks diagram in figure 3.5, experiments were conducted on three of the main blocks, namely the filter&sum module, the speech/non-speech module and the mono-channel speaker diarization module. For each block a baseline was defined to suit its characteristics and to allow for the development of its optimum parameters selection. The initial Wiener filtering of the signal was not analyzed as it was used without modification from its original implementation outside of the scope of this thesis.

The baseline system used for the experiments in the diarization module and in the speech/non speech module corresponds to a modified version of the broadcast news system presented for the NIST RT04f evaluation as described in section 3.1. This corresponds to a mono-channel system (or Single Distant Microphone, SDM, in the meetings domain) with the following main differences from RT04f:

- The speech/non-speech (spnsp) detector used in the experiments for the hybrid spnsp algorithm is composed of a two-states HMM model trained with meetings data, as it was used in the RT05s evaluation and explained in section 3.1. For the speaker diarization module the proposed hybrid spnsp detector was used instead, with parameters equal to the values used for the RT06s evaluation (see Anguera, Wooters and Pardo (2006b), (Anguera, Wooters and Pardo 2006a)). These use the parameter values optimized in the spnsp experiments section.
- During the agglomerative clustering processing the same speaker turn minimum duration is applied as in the broadcast news system (3 seconds). Before the output of the resulting

segmentation, a final segmentation step is performed using the same speaker models but reducing the minimum duration to 1.5 seconds to allow for smaller speaker turns to be properly detected.

- The HMM acoustic models used in the segmentation of the data do not have any maximum time constraint, as explained in section 4.2.3, to allow the speaker segments to be as long as the acoustics dictate. As shown in Anguera, Wooters and Hernando (2006a) it does not change much the DER of the systems but allows for longer speaker segments to be created.
- A few bug fixes regarding floating point values inexactitudes were resolved which slightly changed the system outputs.
- The BIC-based stopping criterion is used in all experiments in order to stop clustering when the optimum number of clusters is reached.

The baseline system used for experiments on the beamforming module is composed of the submission to RT06s NIST evaluation campaign. This contains all the modules as explained in section 5.2 and their parameters optimized using a subset of 10 meetings from the development data available for RT06s.

### 6.1.2 Databases

In the experiments in this thesis the datasets used were obtained from the data available for the Rich Transcription (RT) evaluations for the meetings domain. So far the evaluations on meetings have been RT02, RT04s, RT05s and RT06s. On the later two years only the conference room type data has been used as it contains a richer variety of speakers and with characteristics matching more closely the aim of the algorithms presented in the thesis.

From all available datasets, two groups have been defined as development and test. The RT02, RT04s and RT05s sets form the development set, with a total of 24 meeting excerpts, ranging from 10 to 12 minutes in duration each. The RT06s set has been used as a test set (with 8 meeting excerpts), to compare the system improvements on data not used to tune its parameters. Figure 6.1 summarizes the data available in each one of the RT sets used. For a complete list of the individual files refer to appendix B.

These sets contain a few special characteristics that need to be taken into account. On one hand, the development set contains four meetings that only contain one available microphone. These are two pairs of two CMU meetings recorded for the RT02 and RT04s evaluations. These are not suitable to evaluate the beamforming performance but are left in the development data to obtain fair and comparable results.

| Dataset | # excerpts | ave. duration | meeting sources |      |     |      |     |     |    |
|---------|------------|---------------|-----------------|------|-----|------|-----|-----|----|
|         |            |               | CMU             | ICSI | LDC | NIST | AMI | EDI | VT |
| RT02    | 8          | 674.286       | 2               | 2    | 2   | 2    |     |     |    |
| RT04s   | 8          | 593.185       | 2               | 2    | 2   | 2    |     |     |    |
| RT05s   | 10         | 722.210       | 2               | 2    |     | 2    |     | 2   | 2  |
| RT06s   | 8          | 1080.442      | 2               |      |     | 2    |     | 2   | 2  |

Table 6.1: Summary of datasets used in the experiments

On the other hand, the meeting NIST\_20050412-1303 from RT05s dataset contains one speaker which was participating in the meeting through a telephone device. As will be described later on, using forced alignments to robustly evaluate the data leads to results where this speaker was not included in the reference files and therefore causes a big bias in the scores. Depending on the test performed this meeting will be eliminated to allow for a fair comparison (when doing so it will be clearly stated).

### 6.1.3 Evaluation Metrics

The two main metrics used to evaluate the algorithms presented in this thesis are the Diarization Error Rate (DER) and the Signal-to-Noise Ratio (SNR).

#### Diarization Error Rate

The main metric that is used for speaker diarization experiments is the Diarization Error Rate (DER) as described and used by NIST in the RT evaluations (*NIST Fall Rich Transcription on meetings 2006 Evaluation Plan 2006*). It is measured as the fraction of time that is not attributed correctly to a speaker or to non-speech. To measure it, a script names MD-eval-v12.pl (*NIST MD-eval-v21 DER evaluation script 2006*), developed by NIST, was used.

As per the definition of the task, the system hypothesis diarization output does not need to identify the speakers by name or definite ID, therefore the ID tags assigned to the speakers in both the hypothesis and the reference segmentation do not need to be the same. This is unlike the non-speech tags, which are marked as non labelled gaps between two speaker segments, and therefore do implicitly need to be identified.

The evaluation script first does an optimum one-to-one mapping of all speaker label ID between hypothesis and reference files. This allows the scoring of different ID tags between the two files. The Diarization Error Rate score is computed as

$$DER = \frac{\sum_{s=1}^S \text{dur}(s) \cdot (\max(N_{ref}(s), N_{hyp}(s)) - N_{correct}(s))}{\sum_{s=1}^S \text{dur}(s) \cdot N_{ref}} \quad (6.1)$$

where  $S$  is the total number of speaker segments where both reference and hypothesis files contain the same speaker/s pair/s. It is obtained by collapsing together the hypothesis and reference speaker turns. The terms  $N_{ref}(s)$  and  $N_{sys}(s)$  indicate the number of speaker speaking in segment  $s$ , and  $N_{correct}(s)$  indicates the number of speakers that speak in segment  $s$  and have been correctly matched between reference and hypothesis. Segments labelled as non-speech are considered to contain 0 speakers. When all speakers/non-speech in a segment are correctly matched the error for that segment is 0.

The DER error can be decomposed into the errors coming from the different sources, which are:

- Speaker error: percentage of scored time that a speaker ID is assigned to the wrong speaker. This type of error does not account for speakers in overlap not detected or any error coming from non-speech frames. It can be written as

$$E_{Spkr} = \frac{\sum_{s=1}^S \text{dur}(s) \cdot (\min(N_{ref}(s), N_{hyp}(s)) - N_{correct}(s))}{T_{score}} \quad (6.2)$$

where  $T_{score} = \sum_{s=1}^S \text{dur}(s) \cdot N_{ref}$  is the total scoring time, in the denominator in eq. 6.1.

- False alarm speech: percentage of scored time that a hypothesized speaker is labelled as a non-speech in the reference. It can be formulated as

$$E_{FA} = \frac{\sum_{s=1}^S \text{dur}(s) \cdot (N_{hyp}(s) - N_{ref}(s))}{T_{score}} \quad \forall (N_{hyp}(s) - N_{ref}(s)) > 0 \quad (6.3)$$

computed only over segments where the reference segment is labelled as non-speech.

- Missed speech: percentage of scored time that a hypothesized non-speech segment corresponds to a reference speaker segment. It can be expressed as

$$E_{MISS} = \frac{\sum_{s=1}^S \text{dur}(s) \cdot (N_{ref}(s) - N_{hyp}(s))}{T_{score}} \quad \forall (N_{ref}(s) - N_{hyp}(s)) > 0 \quad (6.4)$$

computed only over segments where the hypothesis segment is labelled as non-speech.

- Overlap speaker: percentage of scored time that some of the multiple speakers in a segment do not get assigned to any speaker. This errors usually fuses either into the  $E_{MISS}$  or  $E_{FA}$ , depending on whether it is the reference or the hypothesis containing non assigned speakers. If multiple speakers appear in both the reference and the hypothesis the error produced belongs to  $E_{spkr}$ .

Given all possible errors one can rewrite equation 6.1 as

$$DER = E_{spkr} + E_{MISS} + E_{FA} + E_{ovl} \quad (6.5)$$

When evaluating performance, a collar around every reference speaker turn can be defined which accounts for inexactitudes in the labelling of the data. It was estimated by NIST that a  $\pm 250$ ms collar could account for all these differences. When there is people overlapping each other in the recording it is stated so in the reference file, with as many as 5 speaker turns being assigned to the same time instant. As pointed out in the denominator of eq. 6.1, the total evaluated time includes the overlaps. Errors produced when the system does not detect any or some of the multiple speakers in overlap count as missed speaker errors.

Once the performance is obtained for each individual meeting excerpt, the time weighted average is done among all meetings in a given set to obtain an overall average score. The scored time is the one used for such weighting, as it indicates the total (overlapped speaker included) time that has been evaluated in each excerpt.

### Signal-to-Noise Ratio

The SNR is a metric based on the power ratio between the signal and the noise. It is mainly used in signal processing applications to evaluate how the desired signal stands out from the background noise. In this thesis it was thought useful to measure how much quality does the resulting speech signal has after the beamforming module. In speech signals it is not clear what part belongs to silence and what to speech, therefore several methods can be applied to compute an SNR approximation. Each computation method can be very independent from every other and therefore comparisons should only be made using the same estimation algorithm. In this thesis the method used in the NIST Speech Quality Assurance Package (SPQA) (*NIST Speech tools and APIs 2006*) and described in detail in section 3.2.1.

#### 6.1.4 Reference Segmentation Selection and Calculation

The use of predefined reference segmentations is necessary to compute the DER given the system hypotheses. The data used in this chapter all comes from the NIST evaluations, which defined a set of rules on how the transcription should be made. In the latest evaluation (*NIST Fall Rich Transcription on meetings 2006 Evaluation Plan 2006*) they were:

- Within a speaker's turn, pauses lasting less than 0.3 seconds are to be considered to belong to that speaker turn. Pauses with more than 0.3 seconds or in between different speaker turns are to be considered non-speech. This value was determined in 2003 by NIST as the minimum duration for a pause that would indicate an utterance boundary.

- Vocal noises, such as laugh, cough, sneeze, breath and lipsmack are to be considered non-speech, and take this into account when considering segment boundaries.
- Although not a rule in creating the transcriptions, it is worth mentioning again the collar of  $\pm 0.25$  seconds to be considered around each reference segment boundary when comparing it to the hypothesis in order to account for inexactitudes in computing the real segment boundary.

Within the NIST evaluation campaigns all data sent out for development and test was carefully transcribed by hand by the Linguistic Data Consortium (LDC). Such transcription was usually done listening to the channel with the best quality possible (which usually is the Individual Headphone Channel, IHM, when available) for each participant, and then the transcriptions are collapsed into a main reference file for all participants.

Prior to the RT06s evaluation it was under consideration by NIST and by some of the participants (including ICSI) the use of forced alignments of the acoustic data. Although in RT06s still hand alignments were used, it is the intention of NIST to change the reference transcriptions to be forced alignments in the near future. The need for such change became strong when areas in overlap started being scored as part of the main metric for system performance. In chapter 3.2 a quantitative comparison is done between forced and hand alignments. In brief, the main drawbacks found in the hand-aligned references are:

- Transcriptions time inconsistency due to the gap of 1 year between each of the transcriptions for each evaluation, which leads to a change in transcription criteria, human transcriber, transcription tools, etc. Leading to consistent differences between the reference files to which the systems try to learn from.
- Inability, at times, to detect short speaker pauses when these are around 0.3 seconds. This leads to problems for systems which are trained to this data and which are impeded to determine when a speaker pause has to be a silence and when it does not.
- Existence of extended durations when labelling the overlap speech. As seen in chapter 3.2 the average length of speech in overlap is bigger in the hand-alignments, usually so as the human transcribers added some arbitrary padding to either side of some overlap regions, leading to greater overlap errors. Such difference varies from evaluation to evaluation and was detected only in RT06s data when overlap became part of the main metric.
- The inability, at times, to identify in the distant microphones (the ones actually used in the evaluations) some sounds or artifacts that are heard and transcribed in the IHM channels (much closer to each speaker's mouth).

It was decided at ICSI that development for the RT06s evaluation had to be done using forced alignments in order to avoid these problems. In order to obtain the forced alignment of a meeting recording a two steps process was followed:

1. The human words transcription for each one of the IHM channels was used to do a forced alignment of the audio in each of the IHM channels to such transcription, obtaining a time-aligned word transcription for each speaker with a headset on. To do so, the ICSI-SRI ASR system (Janin, Stolcke, Anguera, Boakye, Cetin, Frankel and Zheng 2006) was used. Experiments pursued by NIST after the RT06s evaluation Fiscus, Garofolo, Ajoy and Michet (2006) indicated that very similar behaviors for all participants could be obtained using either ICSI-SRI transcriptions or LIMSI's ASR system transcriptions.
2. The transcriptions from each individual speaker were collapsed into a single file and the transcription rules were applied to determine when two words were to be joined into a single speaker segment or two speaker segments needed to be created.

By using forced alignments there are also several drawbacks to point out:

- In the way that these were done, an IHM channel needed to be provided for each participant in the meetings in order to obtain that channel's alignment. One meeting in RT05s (named NIST\_20050412-1303) contained a speaker through a telephone speaker which was not considered, therefore creating a transcript lacking of some of the data. This could be avoided by using other channels instead, trying to always select the optimum quality source.
- Errors in the transcription of the words (which is done so by human transcribers) propagates into the forced-alignments. These errors were measured to be much smaller than transcribing the speaker turns directly.
- Each ASR system does their own systematic errors/decisions which translate into systematic segmentation issues. These are thought to be the difference between every ASR forced-alignment output that can be used. Although such difference is very small, in order to create good quality transcripts, reducing this variability, they could be derived from the output of multiple systems.

All results reported in this thesis were computed using the forced alignments obtained using the ICSI-SRI ASR system, unless otherwise stated.

## 6.2 Experiments from Broadcast News to Meetings

This section covers the experiments done to assess the performance of the system given the different improvements proposed in the previous chapters. To do so, the following structure will be followed:

- This section sets the baseline using the system described in section 6.1.1.
- Section 6.3 verses about the speech/non-speech detector applied to the baseline system in the SDM case.
- Section 6.4 explores the different algorithms introduced in the beamforming block. It quantifies their performance both in SNR and DER.
- Section 6.5 takes both the speech/non-speech detector and the beamforming system and analyzes the performance of the diarization block with the introduced algorithms. It first analyzes each individual algorithm contribution separate from the others and then iteratively agglomerates them into the final optimized system using all blocks.

As explained above, there are several baseline systems that are considered to test the different modules proposed by this thesis. By doing so every module’s performance can be evaluated independently.

The system used to evaluate the speech/non-speech detector can be considered as the baseline of this thesis as it is directly derived from the broadcast news (BN) system found at ICSI at the time of this thesis work start. Such system already contains a few improvements from the BN initial system but these are considered core and will not be evaluated.

The other baseline used (although it is in reality an intermediate system) uses the beamforming system submitted to the RT06s evaluation and the hybrid speech/non-speech detector together with the initial baseline. This is used to evaluate the algorithms in the acoustic beamforming module and in the diarization module.

| System                 | DER Development set |        |          |        |
|------------------------|---------------------|--------|----------|--------|
|                        | SDM                 | MDM    | TDOA-MDM | Ave    |
| RT04f                  |                     |        |          |        |
| Baseline(24 shows)     | 20.6%               | 19.04% | 16.49%   | 18.71% |
| RT06s system(20 shows) | 19.45%              | 17.65% | 14.70%   | 17.26% |
|                        | DER Evaluation set  |        |          |        |
| RT04f                  |                     |        |          |        |
| Baseline               | 24.54%              | 26.5%  | 18.65%   | 23.23% |

Table 6.2: Results for the CV-EM training algorithm in the agglomerate system

Table 6.2 shows the baseline scores to compare to through the following sections. The difference between the baseline and the RT06s system is the inclusion or not of 4 CMU meetings from the development set with a single channel. The version with 20 meeting excerpts is used to develop the beamforming system, while the complete baseline is used to evaluate it (all meetings contain more than one channel).

### 6.3 Speech/Non-Speech Detection Block

Experiments for the speech/non-speech module were obtained for the SDM case to make it directly comparable with the baseline system results shown in the previous section. Although in this case two slightly different development and test sets were used. The development set consisted on the RT02 + RT04s datasets (16 meeting excerpts) and the test set was the RT05s set (with exception of the NIST meeting with faulty transcriptions). Forced alignments were used to evaluate the DER, MISS and FA errors.

In the development of the proposed hybrid speech/non-speech detector there are three main parameters that need to be set. These are the minimum duration for the speech/non-speech segments in both the energy block and the models block, and the complexity of the models in the models block.

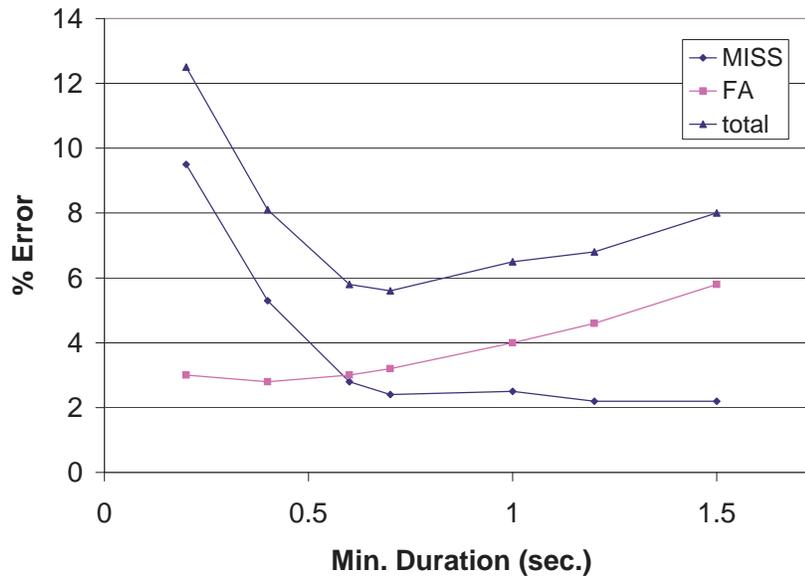


Figure 6.1: *Energy-based system errors depending on its segment minimum duration*

The development set was used to first estimate the minimum duration of the speech and non-speech segments in the energy-based detector. In figure 6.1 one can see the MISS and FA scores for various durations (in # frames). While for a final speech/non-speech system one would choose the value that gives the minimum total error, in this case the goal is to obtain enough

non-speech data to train the non-speech models in the second step. It is very important to choose the value with smaller MISS so that the non-speech model is as pure as possible. This is so because the speech model is usually assigned more Gaussian mixtures in the modeling step, therefore a bigger FA rate does not influence it as much. It can be observed how in the range between duration 1000 and 8000 the MISS rate remains quite flat, which indicates how robust the system is to variations in the data. In any new dataset, if it does not contain a minimum value for the MISS rate at the same value are in the development set, it will most probably still be a very plausible solution. A duration = 2400 (150ms duration) is chosen with MISS = 0.3% and FA=9.5% (total 9.7%).

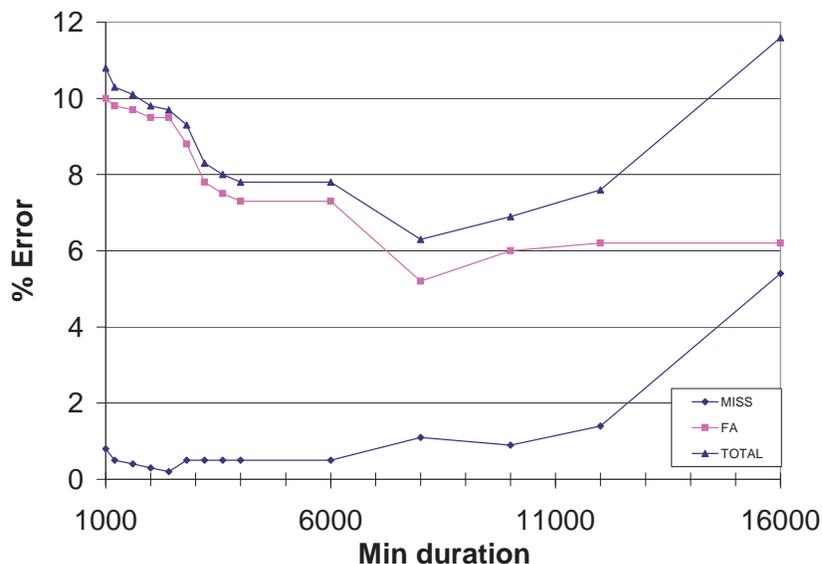


Figure 6.2: *Model-based system errors depending on its segment minimum duration*

The same procedure is followed to select the minimum duration for the speech and non-speech segments decoded using the model-based decoder, using the minimum duration determined by the previous analysis of the energy-based detector. In figure 6.2 one can see the FA and MISS error rates for different minimum segment sizes (the same for speech and non speech); such curve is almost identical when using different # mixtures for the speech model, a complexity of 2 Gaussian mixtures for the speech model and 1 for silence is chosen. In contrast to the energy-based system, this second step does output a final result to be used in the diarization system, therefore it is a need to find the minimum segment duration that minimizes the total percent error. An minimum error of 5.6% was achieved using a minimum duration of 0.7 seconds. If the parameters in the energy-based detector that minimize the overall speech/non-speech error had been chosen (which is at 8000 frames, 0.5 seconds) instead of the current ones, the obtained scores would have had a minimum error of 6.0% after the cluster-based decoder step.

In table 6.3 results are presented for the development and evaluation sets using the selected

| sp/nsp system      | RT02+RT04s |       |             | RT05s |       |             |
|--------------------|------------|-------|-------------|-------|-------|-------------|
|                    | MISS       | FA    | total       | MISS  | FA    | total       |
| All-speech system  | 0.0%       | 11.4% | 11.4%       | 0.0%  | 13.2% | 13.2%       |
| Pre-trained models | 1.9%       | 3.2%  | <b>5.1%</b> | 1.9%  | 4.6%  | 6.5%        |
| hybrid (1st part)  | 0.4%       | 9.7%  | 10.1%       | 0.1%  | 10.4% | 10.5%       |
| hybrid system(all) | 2.4%       | 3.2%  | 5.6%        | 2.8%  | 2.1%  | <b>4.9%</b> |

Table 6.3: *Speech/non-speech errors on development and test data*

parameters, taking into account only the MISS and FA errors from the proposed module. Used as comparison, the “all-speech” system shows the total percentage of data labelled as non-speech in the reference (ground truth) files. After obtaining the forced alignment from the STT system, there existed many non-speech segments with a very small duration due to the strict application of the 0.3s minimum pause duration rule to the forced alignment segmentations. The second row shows the speech/non-speech results using SRI speech/non-speech system (Stolcke, Anguera, Boakye, Cetin, Grezl, Janin, Mandal, Peskin, Wooters and Zheng 2005) which is was developed using training data coming from various meeting sources and its parameters optimized using the development data presented here and the forced alignment reference files. If tuned using the hand annotated reference files provided by NIST for each data set, it obtains a much bigger FA rate, possibly due to the fact that it is more complicated in hand annotated data to follow the 0.3s silence rule. The third and forth rows belong to the results for the presented algorithm. The third row shows the errors in the intermediate stage of the algorithm, after the energy-based decoding. These are not comparable with the other systems as the optimization in here is done regarding the MISS error, and not the TOTAL error. The forth row shows the result of the final output from both systems together.

Although the speech/non-speech error rate obtained for the development set is worse than what is obtained using the pre-trained system, it is almost a 25% relative better in the evaluation set. This changes when considering the final DER. In order to test the usability of such speech/non-speech output for the speaker diarization of meetings data the baseline system was used interposing either of the three speech/non-speech modules shown in table 6.3.

| sp/nsp system      | Development   | evaluation    |
|--------------------|---------------|---------------|
| All-speech         | 27.50%        | 25.17%        |
| Pre-trained models | 19.24%        | 15.53%        |
| hybrid system      | <b>16.51%</b> | <b>13.97%</b> |

Table 6.4: *DER using different speech/non-speech systems*

It is seen in 6.4 that the use of any speech/non-speech detection algorithm improves the performance of the speaker diarization system. Both systems perform much better than just using the diarization system alone. This is due to the agglomerative clustering technique, which starts with a large amount of speaker clusters and tries to converge to an optimum number of

clusters via cluster-pair comparisons. As non-speech data is distributed among all clusters, the more non-speech they contain, the less discriminative the comparison is, leading to more errors.

In both the development and evaluation sets the final DER of the proposed speech/non-speech system outperforms by a 14% relative (development) and a 10% relative (evaluation) the system using pre-trained models. It can be seen how the DER on the development set is much better than the pretrained system, even though the proposed system has a worse speech/non-speech error. This indicates that the proposed system obtains a set of speech/non-speech segments that are more tightly coupled with the diarization system.

## 6.4 Acoustic Beamforming Experiments

In this section an analysis is made on the appropriateness of the different techniques implemented for the acoustic beamforming of the multiple available signals into an “enhanced” signal. The experiments were conducted using both development and evaluation sets as described in 6.1.2 where 4 meetings from CMU were taken out of the development set as they only contained a single microphone.

The experiments use as a comparison system the filter&sum (F&S) beamforming used in the RT06s NIST evaluation, which contains all the modules and algorithms described in section 5.2. This implementation is the one used in the following section to test the appropriateness of all the algorithms in the single-channel diarization module. Each module is evaluated by comparing the performance of the system with and without it, maintaining all other modules in place.

The metrics used in the experiments process in this section are the Signal-to-Noise ratio (SNR) and the Diarization Error Rate (DER) as described in 6.1.3. In order to conduct a fair comparison, the F&S was obtained for each considered beamformed signal and the SNR was computed. After this, the DER was obtained by running the diarization module on that signal (previously parameterized) using the optimum diarization parameters according to the results in section 6.5. The TDOA values were not used in this analysis and the speech/non-speech labels were kept constant to those of the RT06s system (used as the baseline system) as computed and explained in section 4.1.3. This was done in order to focus on the changes in DER only from the change in the beamforming module.

The modules within the beamforming that were analyzed are:

- Reference channel estimation
- Use of a noise threshold
- Use of TDOA stability algorithms

- Use of bad frames elimination in the output
- Use of output weights adaptation

### 6.4.1 Baseline System Analysis

| F&S system   | SNR     |         | DER    |        |
|--------------|---------|---------|--------|--------|
|              | Dev     | Eval    | Dev    | Eval   |
| RT06s system | 30.20db | 45.28db | 17.15% | 22.92% |

Table 6.5: *RT06s system filter-and-sum performance*

Table 6.5 shows the SNR and DER results for the development and test sets. The SNR values are obtained in the same way as in section 3.2, doing a lineal average of the values from each meeting source. The first thing to observe is that although the SNR for the test set is much higher than the development set, the DER values are otherwise, which raises a warning on how uncorrelated these two metrics are. This phenomenon will be repeated throughout the experiments in this section.

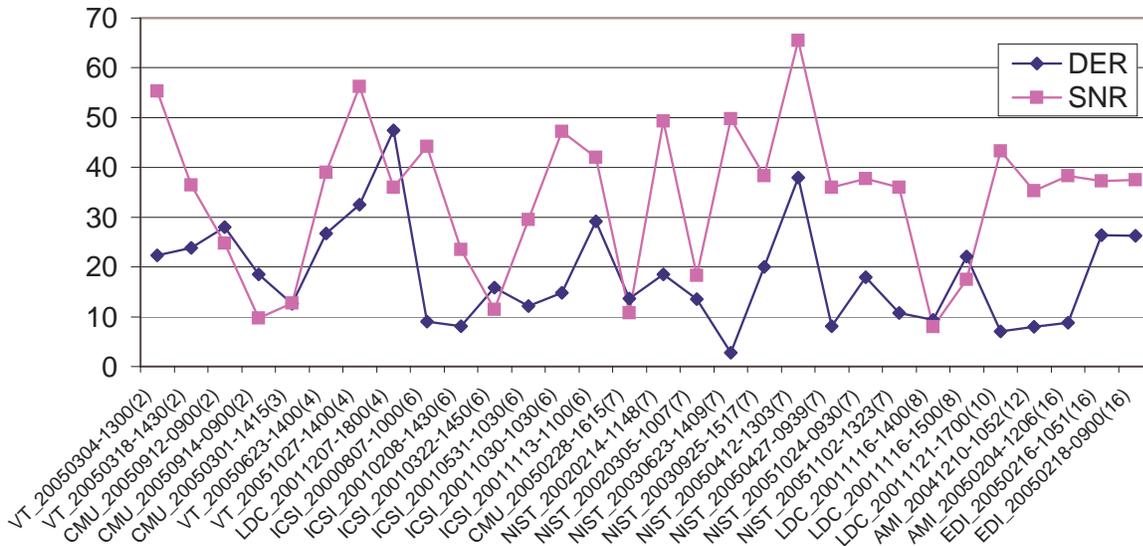


Figure 6.3: *Individual meetings DER vs. SNR vs. number of microphones in the RT06s system*

To further show the lack of correlation of the SNR vs. DER values, figure 6.3 shows the individual values for all shows (22 dev + 8 eval) used in the experiment in table 6.5. The meetings in the X axis are sorted according to the number of available microphones (shown in parenthesis). Both the DER and SNR values share the same Y axis, although SNR is better the higher it goes and DER otherwise. No correlation can be observed neither between SNR and DER values nor with SNR and the number of channels in the meetings.

As for the SNR values it totally depends on the particular rooms, time of day of the recordings and the type of microphones being used. Two cases where values are stable are the AMI project meetings (including AMI and EDI recordings) which keep a very constant SNR value around 37db in average. The DER results depend on these and many other factors. In Mirghafori and Wooters (2006) some of these factors are studied, referring to the high variability of the DER values as show flakiness.

Therefore, it becomes clear that SNR and DER do measure and are affected by different phenomena. A signal output with higher SNR (therefore higher signal quality) does not necessarily lead to a better DER. Given that the aim of this thesis is to improve the diarization output, the DER is the metric that will be most observed (and minimized) but the average SNR will still be shown for all cases as comparison. For other applications, like using the output signal for ASR, the SNR is still the metric to be maximized. Results for ASR using the presented filter&sum system are shown in section 6.4.5.

### 6.4.2 Reference Channel Estimation Analysis

As explained in section 5.2.2, a cross-correlation based algorithm was proposed and implemented in the RT06s system in order to select the optimum channel to act as reference channel in the beamforming system. By automatically selecting this channel the system becomes independent of the human assignment of the SDM channel, which NIST selects in the RT datasets and which was used as reference in previous versions of the system.

| F&S system                    | SNR     |         | DER    |        |
|-------------------------------|---------|---------|--------|--------|
|                               | Dev     | Eval    | Dev    | Eval   |
| RT06s system                  | 30.20db | 45.28db | 17.15% | 22.92% |
| Hand-picked reference channel | 30.72db | 45.53db | 17.09% | 22.49% |

Table 6.6: *Reference channel estimation results*

Table 6.6 shows the results of the RT06s system (with automatic selection) or the same system but using the SDM channel as reference. By automatic selection of the optimum channel the results become slightly worse both in SNR and DER. Although the DER of the development set is almost equal to the hand-picked reference channel case, in the test set there is a decrease in DER performance of 1.87% relative. By considering the development results, it is preferable and more robust to use the automatic selection of the reference channel, as it then becomes possible to use this system in areas other than the RT evaluation data, where there might not be any prior information on which microphone to select as the reference channel.

### 6.4.3 TDOA Post-Processing Analysis

The post-processing module (explained in 5.2.3) includes the noise thresholding algorithm and the TDOA values stability algorithm using Viterbi decoding. These do a post-processing of the computed TDOA values to select the final delays to be applied to each signal prior to doing the sum of the different channels. The noise thresholding algorithm detects those TDOA values that most probably come from a silence region and substitutes its value by the previous, more stable, delay value. It does it by finding the threshold that cuts 10% of the TDOA values as noise. The TDOA stability algorithm uses a double-pass Viterbi algorithm to select the optimum among all possible combinations of N-best computed TDOA values.

These modules are the second version of the initial algorithms implemented and presented in the RT05s system explained in Anguera, Wooters, Peskin and Aguilo (2005). On one hand, the initial algorithm for the noise thresholding used a fixed threshold set by hand (using development data) for all meetings. This caused problems in very noisy shows, as it will be shown in the results, where it is used as a comparison with a threshold = 0.1 over the GCC-PHAT value for each frame (whose values can range from 0 to 1). On the other hand, the first version of the stability 1-best selection algorithm used a simple distance based rules algorithm to either use the 1st-best value or some other value in the N-best list. If the difference between the TDOA value of the first element in the N-best TDOA list for a particular frame and the selected value for the previous frame was greater than a threshold it was searched in the N-best list if there was any other element which was closer, in which case that was the one selected.

| F&S system                  | SNR     |         | DER    |        |
|-----------------------------|---------|---------|--------|--------|
|                             | Dev     | Eval    | Dev    | Eval   |
| RT06s system                | 30.20db | 45.28db | 17.15% | 22.92% |
| RT05s continuity algorithm  | 29.42db | 42.90db | 18.44% | 23.17% |
| No continuity algorithm     | 32.81db | 44.71db | 17.16% | 23.63% |
| RT05s fixed noise threshold | 28.23db | 42.96db | 17.36% | 22.36% |
| No noise Threshold          | 28.79db | 44.87db | 18.31% | 23.38% |
| No post-process             | 31.38db | 45.25db | 18.72% | 22.25% |

Table 6.7: *Post-processing algorithms results*

Table 6.7 shows results for the RT06s system with the latest version of both algorithms presented, and compares them to using the RT05s versions or not using any algorithm (for either algorithm and in overall).

The development set SNR using the RT06s version of both algorithms is better than the RT05s algorithms combination, but is worse than not using any continuity algorithm or not using post-processing at all. The evaluation set SNR for RT06s, though, outperforms all other cases. As for the DER, on the development set, the RT06s system outperforms all other combinations. On the test set results are slightly worse on the proposed RT06s algorithms than not using anything

or doing a fixed noise threshold. In overall, the RT06s post-processing algorithm outperforms the lack of postprocessing in an 8.3% relative on the development set, while it gets a 2.9% worse in the evaluation set.

The noise thresholding algorithm intends to be a simple speech/non-speech detector at the beamforming level. Initial tests were performed to include a more sophisticated detector but it was finally not used in any of the systems. On one hand the results did not reflect the improvements expected. On the other hand the beamforming module was pretended to be independent of any training data, so that it could be applied to any circumstance, which the speech/non-speech detector at the time did not allow.

The change of the noise thresholding to a percentage-based threshold in RT06s versus using a fixed threshold slightly benefits the results of DER in the development set while it gets worse in the test set. Such new algorithm was implemented given the problems obtained in processing very noisy meetings (as it was the case for the LDC meetings) which the RT05s algorithm labelled many frames as non-speech.

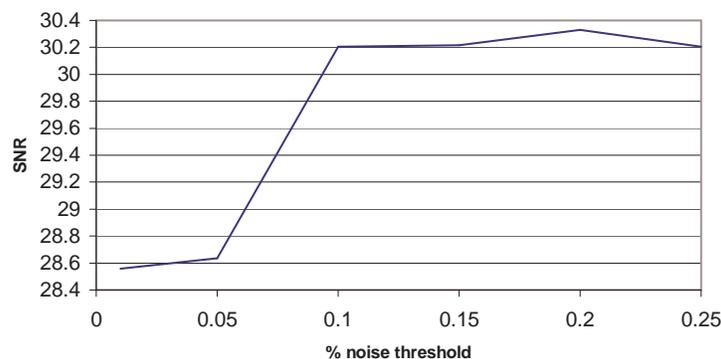


Figure 6.4: *Development set SNR modifying the percentage of noise threshold adjustment*

Figure 6.4 shows the SNR obtained for the development set by sweeping the percentage of frames with lowest GCC-PHAT values considered as noise. After 10% of the frames are selected the SNR results are very stable. This was the selected value for the RT06s system as it modifies as few frames as possible while achieving good performance. If instead of a 10% threshold, the maximum SNR point has been selected (20%) the results for the SNR and DER are shown in table 6.8:

| F&S system                        | SNR     |         | DER    |        |
|-----------------------------------|---------|---------|--------|--------|
|                                   | Dev     | Eval    | Dev    | Eval   |
| RT06s system(10% noise threshold) | 30.20db | 45.28db | 17.15% | 22.92% |
| RT06s system(20% noise threshold) | 30.32db | 44.09db | 16.62% | 22.00% |

Table 6.8: *Results for various percentages of noise thresholding*

By using the optimum value for the SNR in the development set it is observed that the evaluation set obtains a slightly worse SNR but both the dev and eval sets obtain an important improvement in DER. This optimum value was not found during the development of the RT06s system as the development set was slightly different than the one used in these experiments. These scores demonstrate the better performance of the noise thresholding using a percentage instead of using a fixed threshold or no thresholding, shown in table 6.7 above.

The use of the TDOA-selection continuity algorithm is justified by the results in table 6.7. Results on DER comparing it to the RT05s algorithm results show a 6.9% relative improvement on the development set and a more modest 1% relative on the test set. Comparing it to not doing anything obtains similar results in the development set but a 3% relative improvement in the test set. This algorithm though requires the computation of a double Viterbi decoding of the multiple TDOA values, which can take a long time to compute, depending on the number of microphones to process. Although results are beneficiary to the system, it is doubtful if it is feasible to be used in a realtime application.

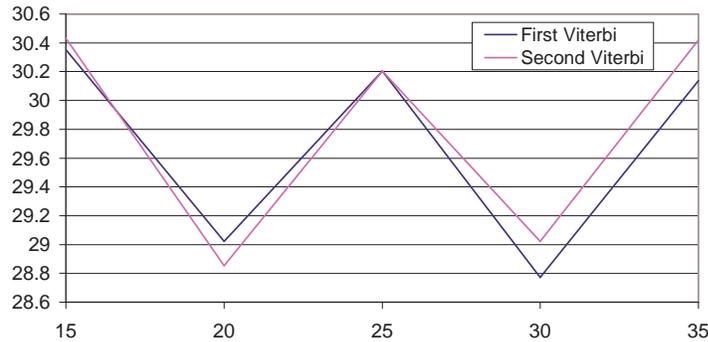


Figure 6.5: *Development set SNR values modifying the Viterbi transition prob. weights in the F&S algorithm*

To further study the effect of the double-Viterbi decoding, the behavior of three parameters in the algorithm are studied with respect to the SNR in the development set. These are the weights used to enhance the transition probabilities in each of the 2 Viterbi decodings levels (see section 5.2.3 for more details) and the number of initial N-best TDOA values given to the algorithm to select from. Table 6.5 shows the SNR for the development set by changing the relative weights of either the first or second Viterbi steps. The default value for both variables taken at 25 is a relative maximum of both curves (very similar to each other). Other values around the default ones obtain better SNR. For both cases in table 6.9 the SNR and DER for the default RT06s case and for cases when either weight is set to 15 are shown.

By selecting an alternative value for the weights in the first Viterbi decoding it obtains an improvement in the SNR of both devel and eval sets and DER in the development set, but the DER gets worse for the evaluation set. On the contrary, by using the alternative weight in the

| F&S system                             | SNR     |         | DER    |        |
|--|---------|---------|--------|--------|
|  | Dev     | Eval    | Dev    | Eval   |
| RT06s system(All weights to 25)        | 30.20db | 45.28db | 17.15% | 22.92% |
| RT06s system(First Viterbi weight 15)  | 30.35db | 45.50db | 16.52% | 23.29% |
| RT06s system(Second Viterbi weight 15) | 30.43db | 42.93db | 17.77% | 23.82% |

Table 6.9: Results for alternative Viterbi weights

second Viterbi only the SNR in the development set improves. It is desirable to maintain both weights to the same value to avoid over-tuning to the data, which in average is better to be at 25, as set for the RT06s evaluation.

The third parameter to analyze in the algorithm is the number of N-best values to be considered by the algorithm when selecting the optimum TDOA value. The first Viterbi step does a local selection within each channel from the N-best possible TDOA values to the 2-best, which then are considered by the second Viterbi in a global decoding among all TDOA values from all channels. The number of possible initial TDOA values is a parameter that describes how many possible peaks in the GCC-PHAT function have to be considered by the first Viterbi. These maxima might represent cases with multiple speakers in overlap or single speakers with impulsive noises. The selection of the right number of initial N-best values needs to account for concurrent acoustic events while avoiding false peaks in the GCC-PHAT function to be selected.

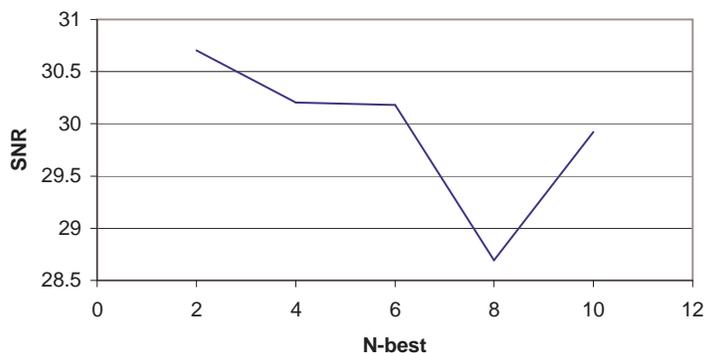


Figure 6.6: Development set SNR values modifying the number of N-best values used for TDOA selection

Figure 6.6 shows the SNR for the development set by selecting from 2 to 10 initial N-best values. The RT06s system default value of 4 obtains a stable SNR behavior (not much change is seen in SNR around it). Choosing only the 2-best peaks of the TDOA values gives a slightly better SNR in the development set, and therefore it was compared to 4 by computing the SNR and DER on the test set, and shown in table 6.10:

Although the SNR for the development set is slightly better using 2-best than using 4-best, the DER for such set behaves very poorly compared to the RT06s system. The SNR for the eval

| F&S system             | SNR     |         | DER    |        |
|------------------------|---------|---------|--------|--------|
|                        | Dev     | Eval    | Dev    | Eval   |
| RT06s system(N-best 4) | 30.20db | 45.28db | 17.15% | 22.92% |
| RT06s system(N-best 2) | 30.70db | 43.43db | 18.86% | 22.21% |

Table 6.10: *Results for alternative Viterbi weights*

set is worse than in the default system, although the DER slightly outperforms the default. The optimum value for diarization is therefore left at 4-best TDOA values.

#### 6.4.4 Signal Output Algorithms Analysis

The signal output module includes the relative channel weight estimation algorithm (explained in 5.2.4) and the elimination of frames from low quality channels (seen in 5.2.4).

The relative channel weight algorithm is necessary when the different microphones are of very diverse type and therefore the levels and kinds of noises being recorded are different. In such cases the standard delay-and-sum theory does not apply as the noise from the different channels cannot cancel itself out. One needs to find the appropriate weights of each of the channels that is able to reduce the effect of a channel when the quality of the signal is poor and magnify it when it is very good. In the RT06s system the relative weights are computed once the channel delays are known, and is a function of the average correlation between the signal of all channels.

The elimination of certain channels when their quality is too low uses also the correlation information to determine when a channel in a particular frame is of too low quality that it is better not to use it in the output as it would degrade the output quality. This is done automatically and in a dynamic way (only certain frames are eliminated, not all the data in that recording). Table 6.11 shows the results comparing the RT06s system with the same without the use of the relative channels weighting and the automatic channels elimination algorithms. When no channel weights are used, a  $\frac{1}{N}$  constant weight is applied to all channels.

| F&S system              | SNR     |         | DER    |        |
|-------------------------|---------|---------|--------|--------|
|                         | Dev     | Eval    | Dev    | Eval   |
| RT06s system            | 30.20db | 45.28db | 17.15% | 22.92% |
| No relative weights     | 30.34db | 44.06db | 17.48% | 24.06% |
| No channels elimination | 31.5db  | 44.96db | 17.14% | 23.91% |

Table 6.11: *Results for relative channel weights and elimination of bad channels*

The SNR of the development set improves when not using either one of the techniques, but gets worse in the test set. The DER improves a 1.8% relative by using the relative channel weights on the development set and a 4.7% relative on the evaluation set. By eliminating the bad channels from the processing the DER does not change in the development set but is improves

a 4.1% relative in the evaluation set.

### 6.4.5 Use of the Beamformed Signal for ASR

The beamforming system presented for this thesis was also used to obtain an enhanced signal for the ASR systems at ICSI presented to the RT NIST evaluations. For RT05s the same beamforming system was used for ASR than for diarization. As explained in Stolcke et al. (2005), evaluated in the RT04s eval set, and not considering the CMU mono-channel meetings, the new beamforming outperformed in 2.8% absolute (from 42.9% word error rate to 40.1%) the previous beamforming system in use at ICSI, which was based on delay&sum of full speech segments.

For the RT06s system the beamforming module was tuned separately from the diarization module to optimize for Word Error Rate (WER) with is a word-based metric (not as the DER, which is time-based). This lead to a system which was more robust than the RT05s beamformer.

| dataset | SDM   | MDM   | ADM   | MM3A |
|---------|-------|-------|-------|------|
| RT05s   | 47.7% | 45.8% | 38.6% | –    |
| RT06s   | 57.3% | 55.5% | 51%   | 56%  |

Table 6.12: *WER using RT06s ASR system including the presented beamformer*

As seen in Janin et al. (2006) and reproduced in table 6.12 the RT05s and RT06s datasets were used to evaluate the RT06s ASR submission in terms of WER. In both datasets there is an improvement of almost 2% absolute by using a single channel or the MDM beamformed signal, where the ASR system only differs in the F&S algorithm use and minor tuning parameters, optimized for each case.

This improvement becomes much larger between the MDM and ADM cases, where the improvement is exclusively due to the increase of microphones available in the ADM case and therefore to the improvement in signal quality due to the beamforming processing.

The mark III microphone arrays (MM3a) were available for the RT06s evaluation. Tests performed comparing results with other state of the art beamforming systems showed that the proposed beamformer achieves an excellent performance.

## 6.5 Speaker Diarization Module Experiments

This section analyzes each of the proposed improvements to the mono-channel diarization module proposed in this thesis. The algorithms that are analyzed in this section are:

- Number of initial clusters (section 4.2.2)

- Cluster complexity selection (section 4.2.2)
- Cross-Validation EM training (section 3.3.3)
- Friends-and-enemies clusters initialization (section 4.2.1)
- Frame purification (section 4.3.1)
- Segment purification (section 4.3.2)
- Multiple feature streams automatic weighting (section 5.3.1)

All experiments use the same baseline system as described in 6.1.1. This uses the new hybrid speech/non-speech detector and the RT06s beamforming system when necessary. In order to evaluate all possible diarization working conditions, three different tasks are run:

- The SDM system uses only the single SDM channel as defined by NIST for the RT datasets used in the experiments.
- The MDM mono-stream system uses the beamforming of the multiple available channels to obtain a single acoustic channel for use by the system. The same configuration is used for the diarization of this channel as for the SDM channel.
- The MDM multi-stream system uses the beamformed signal as well as the extracted TDOA values to add information to the diarization on the location of the speakers. Although this system uses the same acoustic channels as the pure MDM task, this evaluation condition is added as it behaves differently to the data than any of the other tasks.

Experiments were performed in two steps to find the optimum parameters for all algorithms by using the development set. First, each algorithm was tested alone against the proposed baseline. One or two parameters for each algorithm were searched for the optimum configuration. Then, the different algorithms in the order of individual improvement (from most to least) were used together and the optimum parameters were searched for again, obtaining the optimum system according to the development. Finally, the evaluation set was used to test how the system performs with unseen data.

The optimization of the parameters is done using the arithmetic average of the three considered systems, with the exception of the multi-stream weight computation algorithm which only affects the system using the TDOA feature stream.

### 6.5.1 Individual Algorithms Performance

In this chapter each algorithm is tested independently, compared to the baseline system.

## Number of Initial Clusters and Cluster Complexity selection

As pointed out during the system description chapters, the speaker diarization system via agglomerative clustering in the way that it is implemented in this thesis uses very small GMM models trained with small amounts of speaker data. These allow the system to be computationally faster than systems based on UBM adaptation techniques (like Zhu et al. (2006)) but it requires of a more accurate selection of the number of initial clusters in the system, the complexity of the cluster models, and how each model is trained so that data is equally well represented in each model and comparisons between them yield better decisions in the agglomerative process.

For both the number of initial clusters to be used in the system and for the complexity of the cluster models the Cluster Complexity Ratio (CCR) is defined as the parameter to be optimized, as described in section 4.2.2. In the experiments performed for both algorithms an initial analysis studied the effect of such parameter in the final DER when using either algorithm alone compared to the baseline system. As both algorithms use the same parameter, a join experiment using both algorithms was used to tune the optimum CCR value according to the average DER between the three systems considered in the experiments.

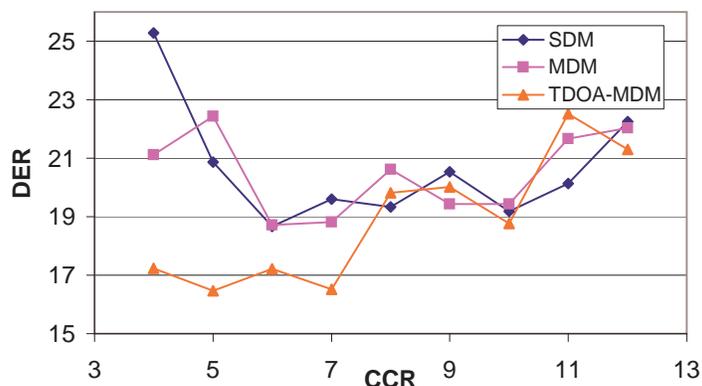


Figure 6.7: *DER for the model complexity selection algorithm using different CCR values*

Figure 6.7 shows the DER for the development set data using the three system implementations for model complexity selection. While SDM and MDM systems obtain a similar behavior on the different CCR values, the TDOA-MDM system tends to obtain the best scores for the lower set of evaluated CCR. Given that only the acoustic feature models are affected by the algorithm (not the TDOA values) this indicates that when using the TDOA values the system becomes more robust to complexity selection errors, given that bigger models (obtained by smaller CCR values) do not overfit as much to the data than when using acoustics alone.

The same behavior is observed when evaluating the number of initial clusters algorithm in figure 6.8. When using the TDOA-MDM system the DER is very stable from CCR=5 to 10. Both algorithms show also an increase in DER when the CCR values are higher that 10, which

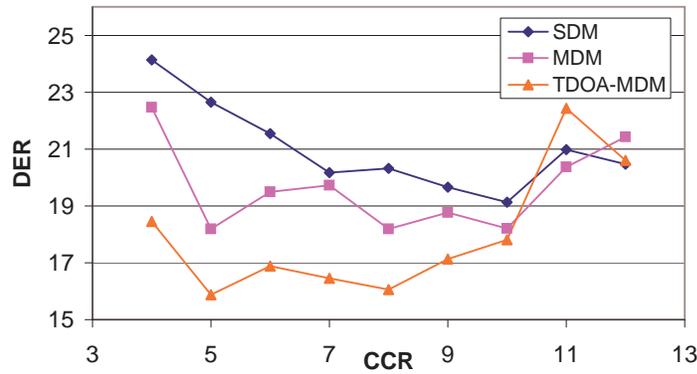


Figure 6.8: *DER for the initial number of clusters algorithm using different CCR values*

indicates that models are too small to well represent the speaker data and too few models are initially created to allow the system to distribute the data appropriately among the speakers.

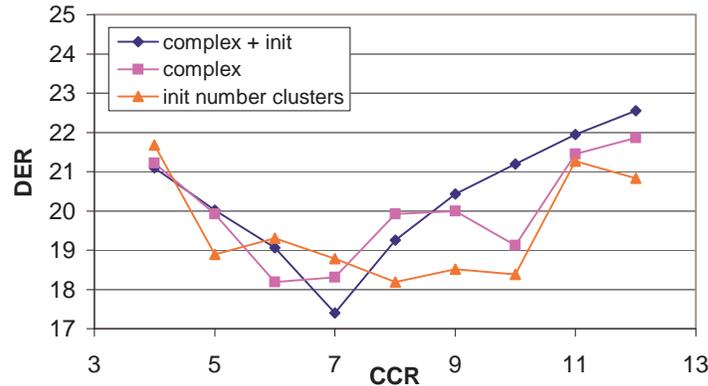


Figure 6.9: *DER for the combination of complexity selection + initial number of clusters using different CCR values*

In both algorithms the CCR value that obtains the optimum average DER is within the range of CCR=6 to 8. In order to use a single CCR value both algorithms are combined and the DER is shown in figure 6.9. In it the average DER between the three systems when using complexity selection or initial number of clusters is compared to the average DER when using both combined. It becomes very clear the existence of an average minimum value at CCR = 7.

Table 6.13 summarizes the results shown in the figures for the development set and shows the evaluation set scores obtained using the optimum parameters for each algorithm. While combining the two algorithms using a new CCR optimum value gives an improvement versus either development results, in the evaluation set the combined score is very similar to the worse of the individual algorithms. In any case, an improvement of 7.05% relative is shown in the development set and only a 0.2% relative in the evaluation set.

| System                       | optimum CCR | DER Devel | DER Test |
|------------------------------|-------------|-----------|----------|
| Baseline system              | n/a         | 18.71%    | 23.23%   |
| Base + complexity select.    | 6           | 18.19%    | 23.2%    |
| Base + number init clusters  | 8           | 18.19%    | 22.24%   |
| Base + complex + init number | 7           | 17.39%    | 23.18%   |

Table 6.13: *DER for the development and evaluation sets for models complexity selection and initial number of clusters algorithms*

### Cross-Validation EM Training

As seen in the previous section, it makes a big difference in the speaker diarization system the correct training of the speaker models. the EM-ML algorithm is appropriate for such endeavor, but in defining the number of iterations it normally undertrains or overfits to the training data. To solve this problem, the Cross-Validation EM (CV-EM) algorithm does EM training iterations over a set of parallel models that allow for a robust validation set to determine when to stop training (defined when the total likelihood of the validation set between two iterations increases less than 0.1%).

When using the CV-EM algorithm initial models use in average more EM iterations than the 5 iterations that were set for the standard EM-ML system. Once the models are retrained using almost the same data as in previous iterations, the CV-EM algorithm stops at 1 or 2 iterations, while the standard EM keeps doing 5 iterations. This reduced in average the computation of the system. the use of multiple parallel models in the CV-EM algorithm does not pose a computational burden as the increase in computation is minimal, which comes from the multiple accumulation of statistics for each model.

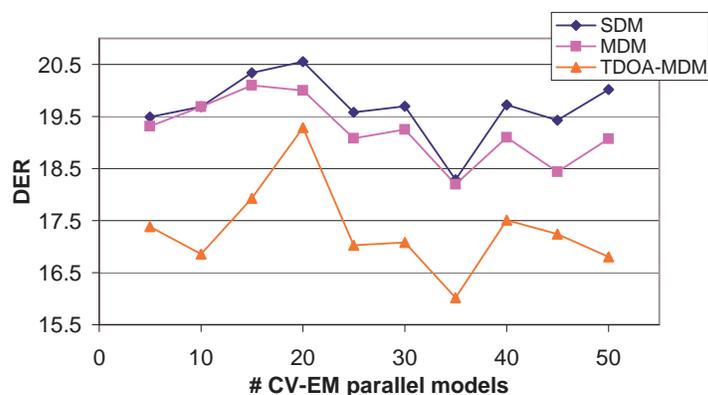


Figure 6.10: *DER variation with the number of parallel models used in CV-EM training*

In the proposed CV-EM algorithm the number of parallel models used needs to be defined a priori. Figure 6.10 shows the evolution of the DER for the three considered systems by modifying the number of parallel models used in the algorithm. In the three cases the minimum DER value

is found at 35 parallel models. At this optimum value, the CV-EM training obtains a 17.50% DER, a 6.46% relative improvement over the baseline.

### Friends-and-Enemies Clusters Initialization

Once the initial number of desired clusters is defined, the algorithm called friends-and-enemies was proposed to cluster the available acoustic data among these clusters so that the cluster purity was maximized and there were no speakers (regardless of the length of their intervention in the recording) that were left without any exclusive cluster. The proposed algorithm is an iterative process where initial single speaker segments are clustered together with the closest segments (friends) and new clusters are derived with the most dissimilar resulting segments(enemies).

In the definition of the algorithm there are three levels of freedom that need to be evaluated. On one hand, a way of selecting the initial speaker segment to start the iterative processing and a metric of “closeness” between segments need to be defined. On the other hand, one needs to determine the optimum number of friend-segments to group into a cluster so that it is well represented but only with data from one speaker.

In the description of the algorithm, in section 4.2.1, three alternatives are proposed for both the selection of the initial segment and the distance metric between segments. Table 6.14 shows the average DER of the considered systems using the different combinations of initial segment selection and distance metric as numbered in section 4.2.1 for the development set. These values were computed using 2 friends for each cluster (i.e. total of 3 segments per initial cluster).

| distance metric/init segment | init 1 | init 2        | init 3        |
|------------------------------|--------|---------------|---------------|
| metric 1 (eq. 4.6)           | 19.8%  | 19.51%        | 19.08%        |
| metric 2 (eq. 4.7)           | 19.74% | <b>18.34%</b> | 20.34%        |
| metric 3 (eq. 4.8)           | 19.58% | 19.12%        | <b>18.10%</b> |

Table 6.14: *Average DER for the development set using various possible distance metrics and initial segment selection criterions*

From the results in table 6.14, the combinations 2-2 and 3-3 are the ones with better DER values, compared to the baseline with 18.71% DER they represent a 1.9% and a 3.2% relative improvement respectively. Taking the combination 2-2, figure 6.11 shows the evolution of the DER for each system with the number of friends chosen.

For values greater than 1 the system is very robust as the three systems obtain very stable results. The minimum average DER pertains to using 3 friends, with a 17.47% DER, a 6.62% relative improvement over the baseline.

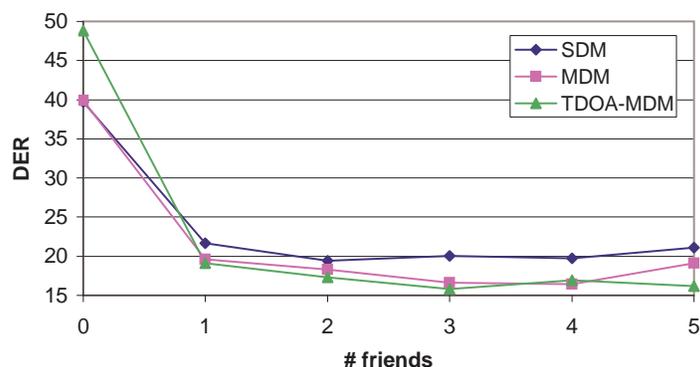


Figure 6.11: *DER variation with the number of friends used in the friends-and-enemies initialization*

### Frame and Segment Purification

The frame and segment purification algorithms deal with the problem of cluster impurity at two different levels. The segment level purification locates the segments within a cluster that are most probable to belong to a different speaker and places them in a different cluster. The frame level purification locates those frames that impede the proper discrimination between clusters and eliminates them when conducting the cluster pairs comparison.

First, experiments were performed on the frame purification algorithm. For all experiments either metric 1 or 2 were used (see section 4.3.1). Two main parameters were selected that determine the behavior of the algorithm given the data. On one hand the percentage  $P\%$  of data with the highest metric values to be eliminated from the models comparison. On the other hand, the percentage of Gaussian mixtures with smallest average variance that are used to compute metric 2. In fact, metric 1 can be considered equivalent to metric 2 when 100% of Gaussians are used.

Given the development data, figure 6.12 shows the average DER for several  $P\%$  of used frames and using 50% and 75% of the available Gaussians. The optimum value is found for 50% Gaussians and using  $P = 70\%$  of the frames. While using a 75% of Gaussians shows a clear minimum point with much higher values around it, with the optimum 50% values oscillate around the minimum, showing a more robust selection of the optimum point.

In the segment purification algorithm no parameters were tuned. Table 6.15 compares the selected systems to the baseline both in the development set and the evaluation set. While segment purification obtains an improvement of 2.5% relative on the development set, it obtains worse results than the baseline in the evaluation set. The frame purification algorithm obtains an improvement of 3.6% and 2.8% relative improvement in development and evaluation sets respectively.

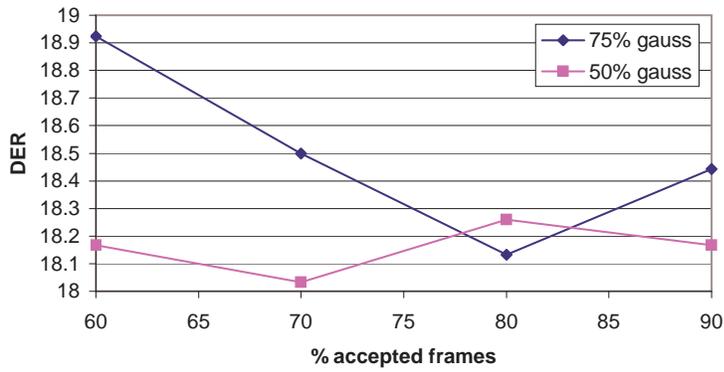


Figure 6.12: *DER variation with the percentage of accepted frames and used Gaussians in frame purification*

| System               | DER Devel | DER Test |
|----------------------|-----------|----------|
| Baseline system      | 18.71%    | 23.23%   |
| Frame purification   | 18.03%    | 22.58%   |
| Segment purification | 18.23%    | 24.23%   |

Table 6.15: *DER results for the segment and frame purification algorithms*

### Multiple Feature Streams Automatic Weighting

In order to test the effectiveness of the automatic weighting scheme for multi-stream feature sets only the MDM system with TDOA values was used. By setting the weights automatically each meeting can compute the optimum relationship between the acoustic and TDOA features. In fact, given that TDOA features determine the identity of the speaker by his/her physical location in the room, it can suffer from modeling errors whenever the speakers move around the room, which in the RT meetings only happens in a few cases.

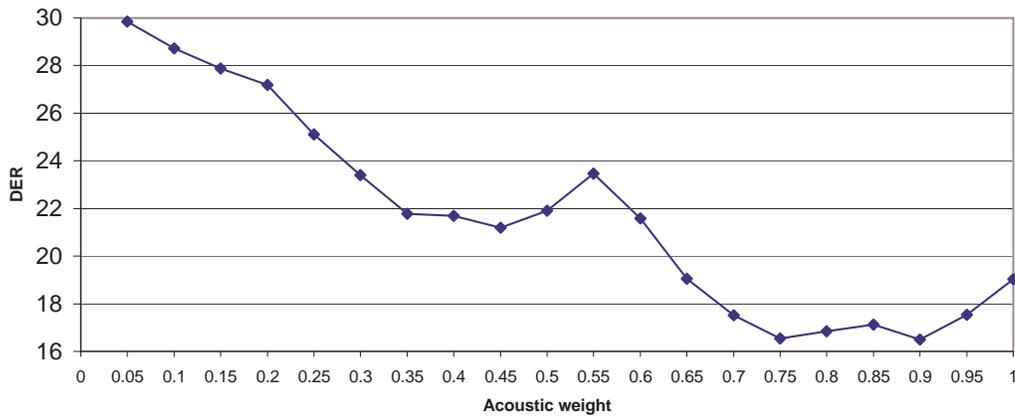


Figure 6.13: *DER scores for the baseline system setting the relative weights by hand on development data*

The alternative to automatically setting the weights is to define a relative weight by using a development set and applying it to all meetings in the test set. This alternative lacks from the flexibility to consider all meetings different, and can diverge from the development to the test set, lacking on robustness to changes. Figure 6.13 shows the DER on the development set for the TDOA-MDM system, where the relative weight between TDOA and acoustic features has been set a priori. It is easily observable the big dynamic range of DER scores obtained along the various possible weights. Given the computed values, the optimum working weight is at  $W_1 = 0.9$  with a DER of 16.49%.

By using the proposed automatic weighting it computes the weights after the  $\Delta\text{BIC}$  values are obtained for all cluster pairs. This could be done after each iteration of the agglomerative clustering and therefore a decision must be made whether the initial iteration weights are kept throughout the process or they are allowed to readapt by using the new  $\Delta\text{BIC}$  pairs.

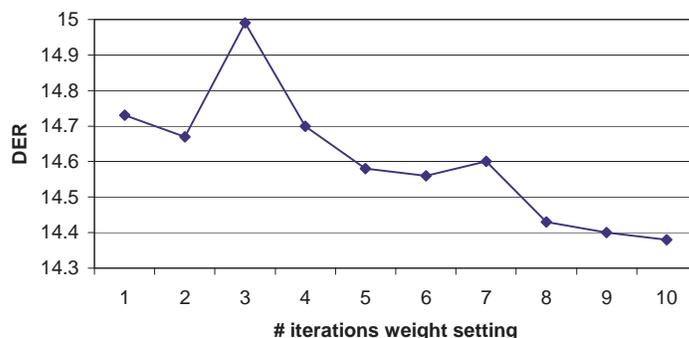


Figure 6.14: *DER evolution with the weight computation iterations*

It was observed in figure 5.13 in section 5.3.2 that the weights usually converge to a stable value after several agglomerative clustering iterations. It must be seen whether these weights obtain an appropriate DER result. Figure 6.14 shows the evolution of the DER computed using the development set by changing the number of iterations in which the weights are reestimated. The DER decreases as the number of iterations increase, with the exception of iteration 3, and stabilizing around iteration 9. This indicates that the system tends to obtain better values for the weight as it progresses, and therefore there is no need in the final system to tune for the number of iterations. Instead, it was allowed to adapt a new weight as long as the stopping criterion did not stop the system.

On final parameter of the weighting algorithm is the initial weights to be used in the system initialization and the initial segmentation, before the first clustering occurs. In order to study the effect of the chosen initial weight in figure 6.15 the DER variation is computed for the development set using the automatic weighting algorithm not limiting the number of adaptation iterations and setting the initial weights. The final system DER for the development data changes

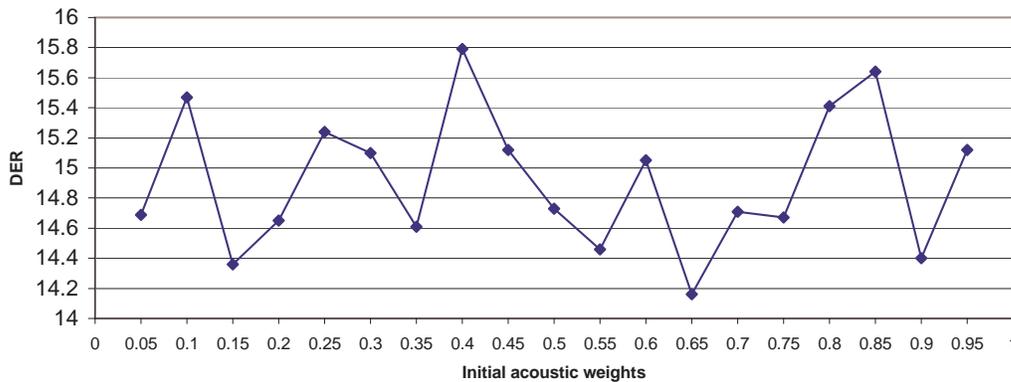


Figure 6.15: *DER evolution changing the initial feature stream weights*

depending on the initial weight setting, being the optimum setting at  $W_{1,init} = 0.65$ . This variation is thought minimal compared to the DER variation observed when using a manual setting and shown in figure 6.13. In fact, by selecting a non informative initial weight  $W_i = 0.5$  there is only a 0.57% absolute DER loss, which might be acceptable for many applications where the nature of the data changes rapidly and it becomes a burden to tune the system every time.

In table 6.16 the DER is computed for several implementations. The mono-stream system uses only acoustic features, the other systems use both acoustics and TDOA values, differing in the way that the weights are found. The system “inv-entropy” performs a frame-wise inverse entropy weight estimation as described in Misra et al. (2003). The “manual weights” system finds the optimum weights using a development set and is set to  $W_1 = 0.9$ . The other two lines show results using the automatic weighting with different initial weights,  $W_1 = 0.9$ , optimum in the development set for the manual case and  $W_1 = 0.65$  optimum in the development set for the automatic weight setting.

| System                   | weight           | DER Devel     | DER Test      |
|--------------------------|------------------|---------------|---------------|
| mono-stream MDM baseline | n/a              | 19.04%        | 26.50%        |
| inverse entropy          | auto             | 24.94%        | 28.57%        |
| manual weights           | optimum(0.9)     | 16.49%        | <b>18.65%</b> |
| auto-weights             | 0.9 + auto       | 14.4%         | 20.07%        |
| auto-weights             | opt(0.65) + auto | <b>14.16%</b> | 20.87%        |

Table 6.16: *DER results for different weight selection algorithms*

Given these results it is seen that using inverse entropy does not achieve good results. In average the entropy method assigns higher weight to the TDOA values while all optimum performance points do otherwise. Also, observe that all the multi-stream methods (except inverse entropy) greatly outperform the mono-stream baseline system.

Automatic weighting obtains, in its optimum point, a relative 14.1% improvement versus manual weighting in the development set. Manually setting the weight achieves the best per-

formance in the test set, although values for the DER around that point obtain much higher errors (DER = 22.85 for  $\mathcal{W}_1 = 0.85$  and DER = 22.29 for  $\mathcal{W}_1 = 0.95$ ) which makes it doubtful of its robustness in other data sets. On the other hand, the values for the automatic weighting algorithm in the test set remain stable (DER = 20.5% in average) for most observed weights.

Such system could be expanded to compute the weights for more than 2 streams, becoming it much easier to automatically do it than having to perform a sweep of possible values using a development set. Even in performance is not improved in all cases, by using the automatic algorithm it becomes much easier to adapt the system to new domains quickly, which follows one of the thesis objectives.

Given these results, the automatic setting of the relative weights between features is set to use an unlimited number of adaptation iterations, with an initial weight  $W_i = 0.65$ . This is the first algorithm to be added to the baseline system in the following chapter as it only affects one of the systems evaluated.

### 6.5.2 Algorithms Agglomeration Performance

In the previous section each algorithm proposed has been tested on its own, against the baseline system described in 6.1.1. Table 6.17 summarizes the results of each algorithm as applied independently to the baseline system, either being only the TDOA-MDM system for the weights computation, or the average of all three systems. The last column shows the rank in improvement over the baseline obtained by each system. This rank is used to determine the order of application of the algorithms in agglomerate to conform the final system.

| System                       | DER Devel | Improvement | rank |
|------------------------------|-----------|-------------|------|
| Baseline TDOA-MDM            | 16.49%    | –           | –    |
| Automatic weighting          | 14.16%    | 14.12%      | 1    |
| Baseline average             | 18.71%    | –           | –    |
| # init clusters + complexity | 17.39%    | 7.05%       | 2    |
| CV-EM Training               | 17.50%    | 6.46%       | 4    |
| Friends-and-enemies init     | 17.47%    | 6.62%       | 3    |
| Frame purification           | 18.03%    | 3.63%       | 5    |
| Segment purificaton          | 18.23%    | 2.56%       | 6    |

Table 6.17: *Summary of DER improvements for the individual algorithms in the development set*

Given the previous section procedure, emphasis must be given to the big difference in some cases between the three systems that are being tested (SDM, MDM and TDOA-MDM), therefore it occurs that some of the algorithms obtain better results with one system than with the other. Results obtained as an average are lower in improvement percentage than what could be obtained in a targeted application. By using an average of three systems and an extended development

set (20-24 meeting excerpts) the level of uncertainty of the results (very typical in speaker diarization, being called “flakiness”) is reduced.

In this section the baseline system is iteratively augmented with the different individual algorithms and further analysis on the parameters studied in the previous section is performed to assess that the same settings (or others) are the optimum in each step. Optimally a full system should be built and a full search done over all parameters space to find the optimum set, but the big number of dimensions of this space and number of algorithms disallow this procedure. Instead, a greedy algorithm iteratively adapts each algorithm to perform optimally within the system.

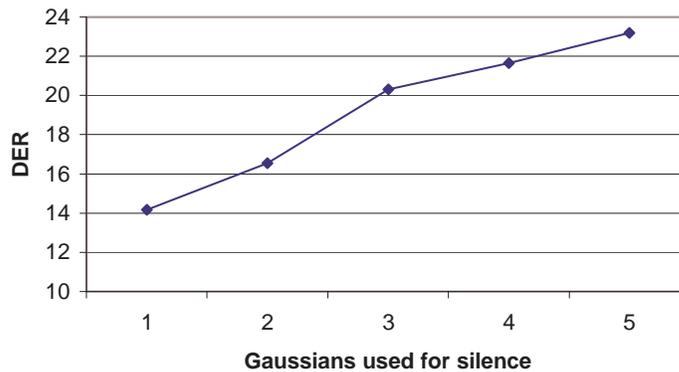


Figure 6.16: *DER variation with the number of Gaussian mixtures initially assigned to the TDOA models*

The first algorithm to be included into the system is the automatic weighting of the TDOA and acoustic streams using an initial weight  $W_1 = 0.65$ . At this point, and given that the second algorithm to include is the model complexity selection for the acoustic features, it is interesting to see how the complexity of the TDOA models in the TDOA-MDM system affect the behavior of the system. Figure 6.16 shows how 1 Gaussian mixture is the optimum complexity to be used for the TDOA models initially. Such complexity increases for the TDOA models as the different clusters merge by creating models that contain the sum of both parents complexities.

As hinted above, the algorithms to include in the next step are the model complexity selection and number of initial clusters. Figure 6.17 shows the evolution of the DER when changing the values of the CCR parameter for the three considered systems. As it happened when studying the individual system, the TDOA-MDM system performs better with lower CCR values than the other two. The optimum working point remains at CCR=7.

Table 6.18 shows the development and evaluation sets DER scores for all systems considered up to this point, and the average. It compares the current results with those of the baseline and the system at the prior agglomerative step, so that improvements in overall can be observed

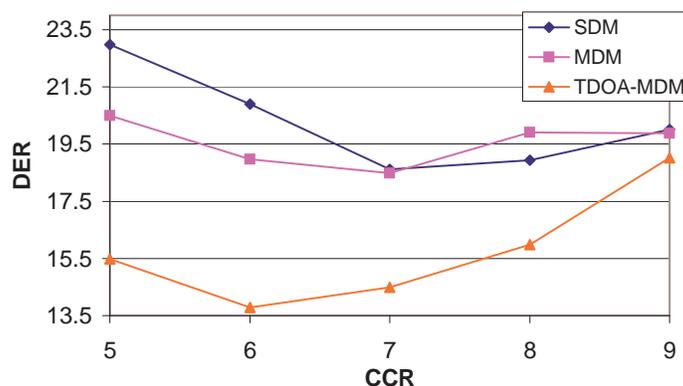


Figure 6.17: *DER variation with the CCR parameter in the agglomerate system*

as well as relative improvements of using this technique in the agglomerate. While the average DER are all better than the baseline for all cases, in the evaluation set the SDM system performs much worse than the baseline. Compared to the prior system (the stream-weight selection) an improvement is seen mostly in the test set where the prior system got worse results than the baseline, but combined with this algorithm obtains a 17.03% DER versus a 18.65% DER, a net gain of 1.62%.

| System                 | DER Development set |        |          |        |
|------------------------|---------------------|--------|----------|--------|
|                        | SDM                 | MDM    | TDOA-MDM | Ave    |
| Baseline               | 20.6%               | 19.04% | 16.49%   | 18.71% |
| prior system           | 20.6%               | 19.04% | 14.16%   | 17.93% |
| Complex + init. number | 18.61%              | 18.49% | 14.49%   | 17.19% |
| DER Evaluation set     |                     |        |          |        |
| Baseline               | 24.54%              | 26.5%  | 18.65%   | 23.23% |
| Prior system           | 24.54%              | 26.5%  | 20.87%   | 23.97% |
| Complex + init. number | 28.84%              | 23.68% | 17.03%   | 23.18% |

Table 6.18: *Results for the model complexity and number of initial clusters in the agglomerate system*

Next, the friends-and-enemies algorithm is evaluated when used in conjunction with the previous systems. In the previous section it was determined that the combinations of metric and init segment selection corresponding to 2-2 and 3-3 gave the best results. Figure 6.18 show the average DER when using either set of parameters. Although the optimum number of friends was 3 when evaluating the algorithm by itself, it is clear in this case that the minimum for both combinations goes to 4 friends.

In Table 6.19 results are shown for the development and evaluation sets for both metric sets with 4 friends per cluster, comparing them to the prior best system and to the baseline.

Although in the development set the MDM system is improved in both metric alternatives

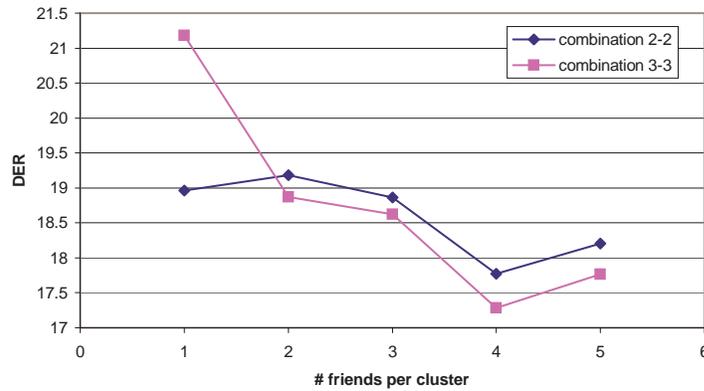


Figure 6.18: *DER variation with the number of friends in the agglomerate system*

| System                    | DER Development set |        |          |        |
|---------------------------|---------------------|--------|----------|--------|
|                           | SDM                 | MDM    | TDOA-MDM | Ave    |
| Baseline                  | 20.6%               | 19.04% | 16.49%   | 18.71% |
| Prior system              | 18.61%              | 18.49% | 14.49%   | 17.19% |
| Friends-and-Enemies (1-1) | 19.72%              | 16.58% | 17.02%   | 17.77% |
| Friends-and-Enemies (2-2) | 19.37%              | 17.57% | 14.9%    | 17.28% |
|                           | DER Evaluation set  |        |          |        |
| Baseline                  | 24.54%              | 26.5%  | 18.65%   | 23.23% |
| Prior system              | 28.84%              | 23.68% | 17.03%   | 23.18% |
| Friends-and-Enemies (1-1) | 30.84%              | 25.16% | 15.38%   | 23.79% |
| Friends-and-Enemies (2-2) | 29.31%              | 30.65% | 14.09%   | 24.68% |

Table 6.19: *Results for the friends-and-enemies algorithm*

with respect to the prior and baseline systems, this is not enough to obtain an average better result. In the evaluation set the same thing happens, being this time the TDOA-MDM system which obtains much better results than previously, but they are masked by the bad performance in SDM and MDM conditions. Even though the system shows that can be useful for certain tasks and conditions, in average in the agglomerate system it shows unable to improve the average performance at this point, therefore it is not included for the next step.

Following this algorithm, the next one in succession is the CV-EM training algorithm. The function of the CV-EM algorithm as used in this thesis is to execute at each training step the optimum number of iterations that allow the cluster models to optimally model the data without overfitting to it or undertraining. In order to compare this algorithm to the standard EM-ML training algorithm in figure 6.19 the average DER is evaluated in terms of the number of iterations of EM training for the system at this point using standard EM-ML. The optimum amount of iterations is 5, as has been used in the baseline system.

When using the CV-EM algorithm at this point in the system, figure 6.20 shows the DER

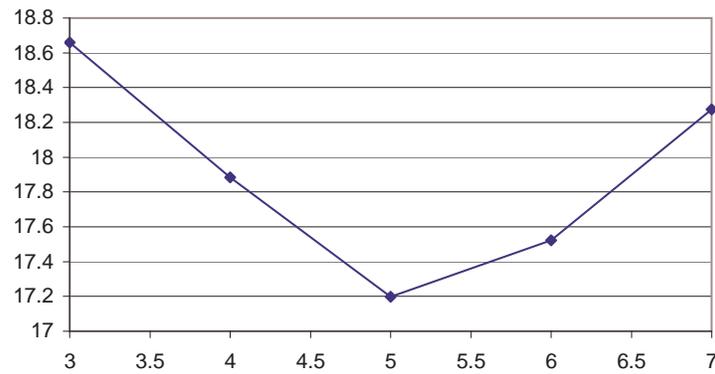


Figure 6.19: *DER variation with the number of EM iterations of a standard EM-ML training algorithm*

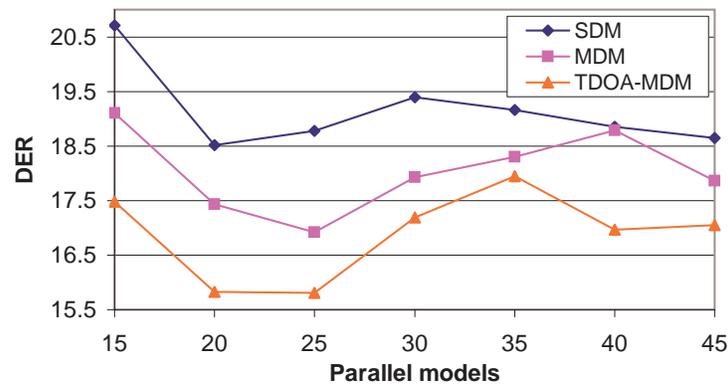


Figure 6.20: *DER variation with the number CV-EM parallel models*

for all three considered systems when selecting different number of CV-EM parallel models. contrary to the same test performed with the algorithm in isolation, in this case all three systems observe a very similar behavior to the number of used models. This can be explained with the increased robustness of the system by using the previously applied algorithms, which makes its performance much more stable and less flaky to small changes in the parameters. In the current CV-EM application the optimum number of models is 25, although 20 would be a good choice too, as results are very stable in that region. Using 15 or less models increases the DER, probably because the different models contain data that starts to be too different to each model's and therefore leads the EM steps to obtain divergent parameters for the models.

Table 6.20 shows the results comparing the inclusion to the CV-EM training algorithm to the prior system (which does not include the friends-and-enemies initialization) and to the baseline system. In the development set the main improvement comes from the MDM system, leading to a final slight gain over the prior system. In the evaluation set results are much improved and a 5.9% relative improvement is observed.

| System             | DER Development set |        |          |        |
|--------------------|---------------------|--------|----------|--------|
|                    | SDM                 | MDM    | TDOA-MDM | Ave    |
| Baseline           | 20.6%               | 19.04% | 16.49%   | 18.71% |
| Prior system       | 18.61%              | 18.49% | 14.49%   | 17.19% |
| CV-Em training     | 18.78%              | 16.92% | 15.81%   | 17.17% |
| DER Evaluation set |                     |        |          |        |
| Baseline           | 24.54%              | 26.5%  | 18.65%   | 23.23% |
| Prior system       | 28.84%              | 23.68% | 17.03%   | 23.18% |
| CV-EM training     | 25.0%               | 23.68% | 16.69%   | 21.79% |

Table 6.20: Results for the CV-EM training algorithm in the agglomerate system

The next algorithm to introduce, according to the order of individual improvement, is the frame purification algorithm. When used in isolation, the algorithm achieved optimum performance when using 50% of the possible Gaussians and keeping 70% of the frames (eliminating the 30% with highest evaluated metric). In order to test both parameters in this setting the first sweep is pursued on the % of frames while keeping the 50% Gaussians fixed.

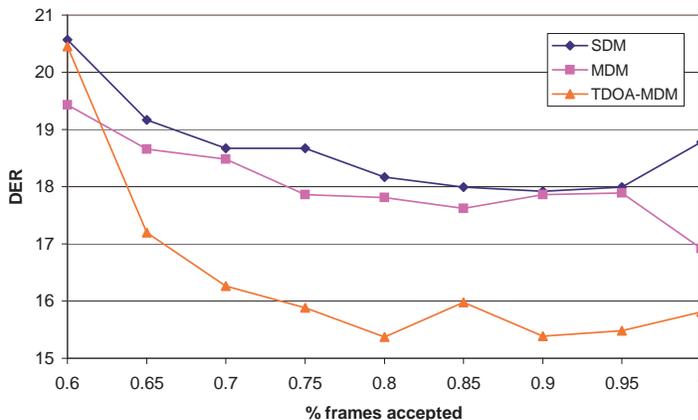


Figure 6.21: DER variation with the frame % acceptance for frame purification algorithm

Figure 6.21 shows the DER for the three compared systems for different frame acceptance percentages. The case of 100% corresponds to the prior algorithm, without any frame purification. In all cases the curves show a high DER at 60% which have a stable improvement as the percentage of accepted frames increases (with exception of 80% in TDOA-MDM). Both SDM and TDOA-MDM obtain several values with better DER than the 100% case, but in MDM this always behaves better. The optimum working point according to the average DER is at 90% of frames accepted.

Fixing now the % of frames at 0.9, the different values for the % of Gaussians used is studied. Figure 6.22 shows the DER when the percentage goes from 20% to 100%. SDM and MDM systems have a very flat behavior, which is disrupted in the TDOA-MDM system from

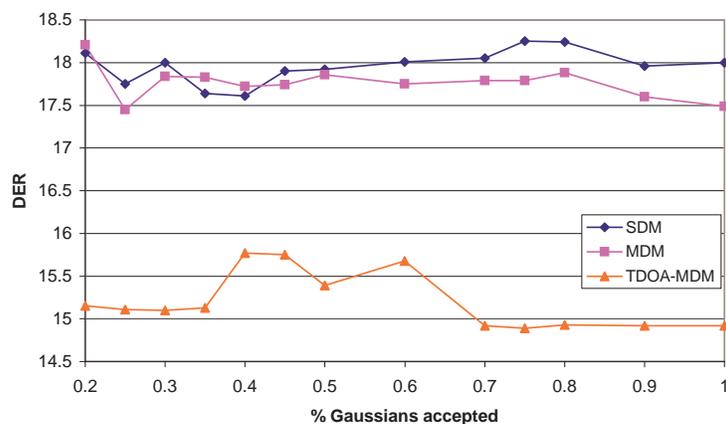


Figure 6.22: *DER variation with the Gaussian % used in the frame purification algorithm*

40% to 60%. The optimum points are at 25% and 100%, which are equivalent to the 2nd and 1st metrics shown in the algorithm description.

These results have a double interpretation. On one hand, by the success of the algorithm in improving the DER it is proven that acoustic frames with high likelihood are more prone at conveying information that is not useful at discriminating between speakers. This could be used in other fields, like speaker identification, where techniques based on frame bagging are already in use to omit those frames with the lowest likelihoods.

| System                          | DER Development set |        |          |        |
|---------------------------------|---------------------|--------|----------|--------|
|                                 | SDM                 | MDM    | TDOA-MDM | Ave    |
| Baseline                        | 20.6%               | 19.04% | 16.49%   | 18.71% |
| Prior system                    | 18.78%              | 16.92% | 15.81%   | 17.17% |
| Frame purification (25% Gauss)  | 17.75%              | 17.45% | 15.11%   | 16.77% |
| Frame purification (100% Gauss) | 18.0%               | 17.49% | 14.92%   | 16.80% |
|                                 | DER Evaluation set  |        |          |        |
| Baseline                        | 24.54%              | 26.5%  | 18.65%   | 23.23% |
| Prior system                    | 25.0%               | 23.68% | 16.69%   | 21.79% |
| Frame purification (25% Gauss)  | 21.57%              | 23.09% | 15.82%   | 20.16% |
| Frame purification (100% Gauss) | 24.24%              | 23.24% | 15.55%   | 21.01% |

Table 6.21: *Results for the frame purification algorithm in the agglomerate system*

Table 6.21 shows the DER for the frame purification algorithm (both using 25% and 100% of Gaussians). All algorithms show an excellent performance both in the development set and in the evaluation set, outperforming the baseline and the prior algorithm (agglomerate system with CV-EM training). This is a 7.4% and 3.5% relative on the evaluation set using either the 25% or 100% of Gaussians (respectively).

Finally, the segment purification algorithm is evaluated using the system composed of all

successful prior systems. Table 6.22 shows the results in comparison with the prior system (selected with 25% Gaussians used) and the baseline.

| System               | DER Development set |        |          |        |
|----------------------|---------------------|--------|----------|--------|
|                      | SDM                 | MDM    | TDOA-MDM | Ave    |
| Baseline             | 20.6%               | 19.04% | 16.49%   | 18.71% |
| Prior system         | 17.75%              | 17.45% | 15.11%   | 16.77% |
| Segment purification | 17.65%              | 17.94% | 14.89%   | 16.82% |
| DER Evaluation set   |                     |        |          |        |
| Baseline             | 24.54%              | 26.5%  | 18.65%   | 23.23% |
| Prior system         | 21.57%              | 23.09% | 15.82%   | 20.16% |
| Segment purification | 23.07%              | 23.47% | 15.12%   | 20.55% |

Table 6.22: Results for the segment purification algorithm in the agglomerate system

Results are mixed, improving in certain systems and being worse in others. In average both the development and evaluation sets obtain worse results than the prior system. An interesting effect is also noticed in that for both the development and evaluation the MDM system performs worse than the SDM, which indicates that somehow the segment purification system can identify better the segments from alien speakers when only one channel is used. Given the results and the high computational load that the segment purification poses on the system, this is taken out of the experiments system.

## 6.6 Overall Experiments and Analysis of Results

In the previous section a majority of the algorithms proposed in this thesis for use in speaker diarization have been analyzed, first by themselves comparing them to the baseline and then as part of an agglomerate system in order to obtain an optimum final system.

| System                       | DER devel | Improv. vs. prior | Improv. vs. baseline | DER eval | Improv. vs. prior | Improv. vs. baseline |
|------------------------------|-----------|-------------------|----------------------|----------|-------------------|----------------------|
| Baseline                     | 18.71%    | –                 | –                    | 23.23    | –                 | –                    |
| multi-stream weights         | 17.93%    | 4.16%             | 4.16%                | 23.97%   | -3.18%            | -3.18%               |
| # init clusters + complexity | 17.19%    | 4.12%             | 8.12%                | 23.18%   | 3.29%             | 0.21%                |
| Friends-and-enemies init     | 17.77%    | -3.37%            | –                    | 23.79%   | -2.63%            | –                    |
| CV-EM Training               | 17.17%    | 0.11%(*)          | 8.23%                | 21.79%   | 5.99%(*)          | 6.19%                |
| Frame purification           | 16.77%    | 2.32%             | <b>10.36%</b>        | 20.16%   | 7.48%             | <b>13.21%</b>        |
| Segment purificaton          | 16.82%    | -0.29%            | –                    | 20.55%   | -1.93%            | –                    |

Table 6.23: Summary of average DER for the agglomerate system on development and evaluation data

Table 6.23 shows a summary of the results analyzed in the previous section and computes the relative improvement of each algorithm with respect to the previous one and to the baseline (accumulating all improvements). Both the friends-and-enemies and segment purification algorithms obtain bad results in this experiment and therefore are not included in the agglomerate

system result (results with \* indicate that the relative improvement to the prior system computes it not taking them into account). The algorithms not included in this final experiment are still valid and obtain good results in certain situations, but not in the average of all cases.

Taking into account the average DER between the SDM, MDM and TDOA-MDM system outputs, and tuning all the algorithms to them, the proposed algorithms in the diarization module improve up to a **10.36%** relative in the development set and up to a **13.21%** relative in the evaluation set.

While such approach of optimizing the average DER obtains systems that will perform well for all the different tasks considered. In some applications where multiple microphones are available it is interesting to find the best result possible. To obtain it the TDOA-MDM system has been selected and the parameters optimized to it according to the parameter sweeps performed in the previous section. The parameters selection was then:

- Automatic multi-stream weighting
- model complexity selection and number of initial clusters with CCR=6
- CV-EM training with 25 CV-EM parallel models
- Friends-and-enemies initialization using 2-2 metric-init combination and 5 friends
- Frame purification with 80% frames accepted, 75% Gaussians used
- Segment purification enabled

| System                  | DER development | DER evaluation |
|-------------------------|-----------------|----------------|
| Optimum average system  | 14.92%          | 15.55%         |
| Optimum TDOA-MDM system | 14.64%          | 14.76%         |
| best TDOA-MDM devel     | 13.78%          | 17.22%         |

Table 6.24: *Results for the TDOA-MDM task using different algorithm settings*

The obtained results are shown in table 6.24 where DER results are shown only for the TDOA-MDM task, which is usually the one with best performance. By using the optimum parameters according to the development sweeps in the previous section, and including all algorithms into the system, the resulting optimum TDOA-MDM system obtains an improvement of 5.08% on the evaluation set versus the optimum average system optimized for the average DER for all three systems.

The final system has a robust performance over changes in the data, sometimes at the cost of not obtaining the absolute minimum DER in all cases. To illustrate this, let us take the system labelled as best TDOA-MDM on the development, which is a system built only using

the automatic weighting algorithm and the definition of number of clusters and complexity of the models. This system outperforms vastly the optimum systems in the development set but when changing to a different set it returns a poor performance. By using the optimum systems once all algorithms are in place, the results are even in both sets at the expense of some increase in the DER in the development.

| System                  | DER development | DER evaluation |
|-------------------------|-----------------|----------------|
| BN baseline (SDM)       | 24.88%          | 19.80%         |
| Meetings baseline (MDM) | 19.04%          | 26.5%          |
| Optimum TDOA-MDM system | 14.64%          | 14.76%         |

Table 6.25: Overall thesis scores comparison

Table 6.25 shows the DER scores to illustrate the overall improvement achieved by the system while transforming from broadcast news speaker diarization to diarization for meetings. The BN baseline system shows the DER of the described baseline which uses model-based speech/non-speech detection. As pointed out, even though this system is already a step forward from the system at the start of this thesis work, it acts as a good baseline for all the work done in beamforming and speaker diarization.

The meetings baseline is the same baseline as in the BN system, but using the hybrid speech/non-speech detection and the baseline RT06s beamforming. Finally, the optimum TDOA-MDM system has been presented earlier in this section and shows the optimum/robust results obtained by using all proposed algorithms.

The optimum TDOA-MDM system obtains an outstanding 41.15% relative improvement on the development set compared to the BN baseline, and a 25.45% relative improvement on the evaluation set. As shown over the experiment sections, these improvements are due to all the new and refurbished algorithms proposed for the system, being the MDM beamforming and the inclusion of TDOA features into the diarization the two most outstanding components.

One Interesting result in the BN baseline system is the outstanding difference between development and evaluation results. While the system performs rather poorly on the development set, for that particular combination of parameters it obtains a very good result on the evaluation set. Running any acoustic beamforming using more microphones than just the SDM results always in an increase on the DER. In fact, an experiment using the RT06s beamforming on top of the BN baseline achieves a 23.63% DER on dev set (a slight improvement) and 26.3% DER on the evaluation set (much worse, similar to the result for the meetings baseline).

This is another example of show flakiness and lack of robustness on the baseline system. On one hand, while one set of data performs well, another set can perform very poorly for the same parameters setting. On the other hand, when doing changes to the system, not all datasets perform the same way, achieving an improvement on one set does not mean that it translates

to others. This problem is the keystone of research in speaker diarization and has been a main concern and issue during all the thesis development, as random results jeopardize the assessment of new techniques that, although beneficiary to the system, might have been discarded due to poor performance.

Given each independent recording (meetings, broadcast news or other sources), the speaker diarization algorithm processes it and obtains an output segmentation. Such segmentation might show a slight improvement due to the applied systems but could also obtain a very high DER due to factors like a badly chosen stopping point. By considering the final DER as the time-weighted sum of all excerpts, if a few of them experienced such bad behavior then the final score is worse than previous runs, misleading to thinking that the tested algorithm is not correct. When computing the DER for a small set of excerpts (8 to 10), these errors cause a big impact in the final score.

While the DER score is a standard way of measuring diarization performance as used in the NIST RT evaluations, which is the ultimate metric to reduce to show improvements in the systems, in order to avoid the problems posed earlier there are several alternatives. On one hand the use of bigger development and evaluation sets so as to reduce the effect of these outliers. To work with such datasets there must be accurate and consistent transcriptions to test against, which should be obtained using automated mechanisms like forced-alignments. On the other hand, the DER metric could be altered to eliminate the outlier scores from the average during development. Although this would solve the occasional excerpts with big errors, it does not help improve those that are considered “hard nuts” (shows that always perform very badly) and it is therefore difficult to define the outlier boundaries that describe the system correctly.

| System / DER            | RT02s  | RT04s  | RT05s  | all dev. | eval (RT06s) |
|-------------------------|--------|--------|--------|----------|--------------|
| BN baseline (SDM)       | 35.20% | 28.26% | 23.26% | 27.82%   | 37.67%       |
| Optimum TDOA-MDM system | 27.86% | 25.05% | 17.47% | 22.27%   | 31.75%       |

Table 6.26: *Overall thesis DER scores comparison*

Finally, for comparison purposes, table 6.26 shows the initial and final systems evaluated this time using the hand-alignments proposed by NIST during the evaluation campaigns and splitting results into evaluation sources. These are used for comparison only as development using these references was stopped and switched to the forced-alignment references, much more robust across years.



## Chapter 7

# NIST Evaluations in Speaker Diarization

The National Institute of Standards and Technology (*National Institute for Standards and Technology* 2006) (NIST) is an agency of the U.S. Commerce Department's Technology Administration that was created to provide standards and measurements for the U.S. industry. Within NIST, the speech group's mission is to contribute to the advancement of spoken language processing technologies (both speech recognition and understanding) so that spoken language can reliably serve as an alternative modality for the human-computer interface of the future. This is done by defining performance measurement methods, providing reference materials, performing benchmark tests within the research and development community and building prototype systems to show the latest speech technology advances and future applications.

In the last decade NIST has been performing a series of evaluations in the topic of speaker diarization in order to empower research institutions to evaluate their research systems using a common framework, including data and specifications to follow. Each evaluation in speaker diarization has been included within a more general framework including other research areas. From 2000 to 2002 speaker diarization was run within the speaker recognition evaluation (SRE, see *NIST Speech Recognition Evaluation* (2006)) and from then on it has been included in the Rich Transcription Evaluation (RT eval, see *NIST Rich Transcription evaluations, website: <http://www.nist.gov/speech/tests/rt>* (2006)).

ICSI has been participated regularly to the speaker diarization evaluations and also to the speaker recognition and speech to text evaluations. During my stay at ICSI I have been a part of the team entering the latest broadcast news speaker diarization evaluation and on the latest two evaluations of speaker diarization for meetings. In this chapter an overview is first given of the evaluation campaigns in speaker diarization for meetings for the last two years and then the systems that ICSI participated with are explained, as well as which performance they achieved. Finally some personal insight is offered about my opinion of pros and cons of these evaluations.

## 7.1 NIST Rich Transcription Evaluations in Speaker Diarization for Meetings

The Rich transcription evaluations conducted by NIST started with the RT02s in 2002 until the latest one (RT06s). According to NIST (*Spring 2005 (RT-05S) Rich Transcription Meeting Recognition Evaluation Plan* (n.d.), *Spring 2006 (RT-06S) Rich Transcription Meeting Recognition Evaluation Plan* (n.d.)) the Rich Transcription (RT) of a spoken document addresses the need for information other than the set of words that have been said (extracted with a Speech-to-Text, STT, system). When obtaining a transcription of the words that have been spoken in a recording it is difficult to receive all the information that the speakers tried to convey. This is because spoken language is much more than just the spoken words; it contains information about the speakers, prosodic cues and intonation, and much more.

The goal of future RT systems is for transcripts to be created with all sorts of metadata to allow the user to fully understand the content of an audio recording without listening to it. In the recent RT evaluations NIST has focused on three core technologies that are important elements of the metadata content. These are Speech-to-Text (STT), Speaker Diarization (SPKR) and Speech Activity Detection (SAD). In the last two years (RT05s and RT06s) evaluations have been focusing on the meetings domain.

### 7.1.1 RT05s and RT06s Evaluation Conditions

This section focuses on the two latest evaluations performed on the meetings domain, namely RT05s and RT06s. These two are similar in that two different subdomains were proposed, with different microphone configurations within each subdomain. All systems were allowed to run with unlimited runtime speed so that they could be comparable within the same metrics. The speed of each system was reported as part of the system description.

In brief, the two proposed subdomains were:

- **Conference room meetings:** These are conducted around a meetings table with several participants involved in an active conversation among them. It contains various amounts of speaker overlap (depending on the nature of the meeting). These have been the focus of research of several projects including the European AMI project.
- **Lecture room meetings:** These are conducted in a lecture setting where a lecturer gives a presentation in front of an audience, which normally interrupts with questions during the talk. In these meetings the lecturer normally speaks for most of the time during the talk, and it becomes more balanced during question and answer sections. It has been the focus of research of the European CHIL project.

In each one of the meeting rooms there are multiple microphones available which record the signal synchronously. In some settings there are also cameras, but these fall outside of the scope of the speaker diarization evaluation. The microphones are clustered in different groups to determine different conditions/evaluation subtasks. The following list points out the terminology used for each of the possible groups and whether it is used in the speaker diarization evaluation and in which domain:

**SDM** (Single Distant Microphone): This is defined as one of the centrally located microphones in the room, located on the meetings table. This microphone is always part of the bigger MDM group. Both lecture and conference room subdomains run this task.

**MDM** (Multiple Distant Microphones): These are a set of microphones situated on the meeting table. All participants in the conference room subdomain sit around the table as well as participants on the lecture room subdomain except for the lecturer. This task also exists in both subdomains.

**MM3A** (Multiple Mark III Microphone Arrays): The lecture meetings contain one or two of these arrays, which were built by NIST and contain 64 microphones setup linearly. Diarization could be run on either 64 channels or a beamformed version of it distributed by Karlsruhe University for RT06s.

**MSLA** (Multiple Source Localization Microphone Arrays): These are four groups of four microphones positioned into a “T” shape array which were originally defined for speaker localization. They are only found in the lecture subdomain.

**ADM** (All Distant Microphones): In lecture room recordings this task allows the system to use all possible microphones previously explained (all except for the IHM microphones). The conference room subdomain does not usually define this task as all distant microphones are of MDM type.

**IHM** (Individual Headphone Microphone Arrays): Although not evaluated in the diarization evaluations, these microphones are worn by some of the participants in the meetings. They are a task in the STT evaluation and are also used when creating the forced-alignment reference segmentations for speaker diarization.

### 7.1.2 Methodology of the Evaluations

Each one of the NIST evaluations start one year in advance during the workshop organized to share results from the previous RT evaluation. In there all the participants are able to make comments on the different tasks and propose changes in the evaluation or possible new eval-

uations. A schedule is then set for each of the necessary deadlines to follow towards the next evaluation.

During the months following the workshop normally a set of conference calls occur where further details are polished in terms of available databases, metrics used or changes in the tasks. A deadline is normally set for research groups to commit to run the evaluations. This is normally about one month before the evaluation starts.

On the months prior to the evaluation period some development and training data is distributed and for STT there are limits set on the sources of the data that can be used for training so that what differentiates the systems is their algorithms and not the amount of data they are trained on.

The evaluation data is handed to all sites at the same time and they have normally about three weeks to process it. In RT05s and RT06s the conference room results were due a week earlier than the lecture room results, to allow labs with fewer resources to process all. By participating in the evaluation all sites make a pledge not to do any development using the evaluation data, so that results from their systems are a realistic indication of performance on unseen data.

Once the results have been turned in to NIST, scores are normally delivered to the participating sites within a week. The scores computed for each entry are the Diarization Error Rate (DER), as explained in the experiments chapter. For the task of speech activity detection the same score is used but considering any speaker segment as speech, wether if it is one or multiple speakers talking.

After results are made public each participant then prepares a paper describing the systems they used in order to share the knowledge acquired during the evaluation. These papers are presented in a Workshop where all participants can meet each other and start planning for the following year's evaluation. On both RT05s and RT06s the Workshop has coincided with the MLMI workshop, and the papers of the evaluation participants have been published in Lecture Notes in Compute Science from Springer jointly with the workshop's papers.

### 7.1.3 Data used on the Speaker Diarization Evaluations

The test datasets used in both RT05s and RT06s evaluations were composed of conference and lecture type data. The conference data is composed of ten and nine meeting excerpts of 12 minutes each. One meeting was eliminated from RT06s after the evaluation finished for technical issues. These datasets have been used in this thesis to evaluate the different proposed techniques and are covered in mode detail in the experiments chapter and in appendix B.

The lecture room data for test was composed of excerpts of different sizes contributed by the different partners in the CHIL project and corresponding to different instants in a lecture

meeting. In particular:

- RT05s test data was composed of 29 excerpts, all recorded at Karlsruhe University. Up to three excerpts were selected from each meeting, but systems were not expected to process the data from each meeting together. The majority of data corresponded to the lecturer, resulting in many excerpts where only one person was speaking. The shortest excerpt was 69 seconds and the longest 468 seconds.
- RT06s test data was composed of 38 excerpts of five minutes each, recorded in 5 different CHIL meeting rooms: 4 at AIT, 4 at IBM, 2 at ITC, 24 at Karlsruhe and 4 at UPC. This year the excerpts were chosen to contain a bigger variety of speakers and situations. After the evaluation finished, the set was reduced to 28 excerpts for technical reasons.

The development data used in these evaluations was usually a compilation of the data sets from previous evaluation campaigns. The used sets for conference room data were from RT02s and RT04s evaluations for RT05s, and a subset of RT02s through RT05s for the RT06s evaluation. For the lecture room evaluations, as this subdomain was first included in the evaluation in RT05s, there was no prior datasets available and therefore NIST distributed a set of transcribed lecture recordings similar to those in RT05s. For RT06s development was done using a subset of the original development set plus the RT05s evaluation set.

Although the diarization system does not use any training data, the speech/non-speech detector used in RT05s needed to be trained. It used around 80 hours of meetings data extracted from the ICSI meeting corpus.

## 7.2 ICSI Participation in the RT Evaluations

In this section an overview the ICSI participation in the NIST RT evaluations in meetings for 2005 and 2006 is given, which served as a test for the techniques and algorithms presented in this thesis, which were created and evolved during this time.

### 7.2.1 Participation in the 2005 Spring Rich Transcription Evaluation

For the RT05s evaluation ICSI presented several systems combining different alternative algorithms. All combinations have in common:

- Frontend composed of a single acoustic stream with 19<sup>th</sup> order MFCC, no deltas, 30 msec analysis window, 10 msec step size.
- Each initial cluster is modeled with a GMM with five Gaussian mixtures.

- Iterative segmentation/training.
- Segment level cluster purification.

ICSI participated in the speaker diarization task on conference room and lecture room data. The speech activity detection (SAD) algorithm was also ran on the data but it did not compete in the official SAD evaluation. Next the systems presented for each domain are presented.

### Conference Room Systems

For the conference room environment the submission consisted on one primary system in each of the MDM and SDM conditions. The MDM system uses filter&sum to acoustically fuse all the available channels into one enhanced channel. Then it applies the speaker diarization to this enhanced channel. The SDM condition skips the filter&sum processing, as the system's input is already a single channel (from the most centrally located microphone according to NIST). The filter&sum processing lacks some of the delay post-processing improvements presented in RT06s.

### Lecture Room Systems

In the lecture room environment the submission consisted on primary systems for the tasks MDM, SDM and MSLA, and contrastive systems for MDM (two systems), SDM and MSLA (two systems).

Following is a brief description for each of these systems and their motivation:

- MDM, SDM and MSLA primary condition (MDM/SDM/MSLA\_p-omnion): It was observed in the development data that on many occasions it was possible to obtain the best performance by just guessing one speaker for the whole duration of the lecture. This is particularly true when the meeting excerpt consists only of the lecturer speaking, but is often also achieved in the question-and-answer section since many of the excerpts in the development data consisted of very short questions followed by long answers by the lecturer. They were therefore presented as the primary submissions, serving also as a baseline score for the lecture room environment. Contrary to what was observed in the development data, the contrastive ("real") systems outperformed the primary ("guess one speaker") submissions on the evaluation data. Depending on what data is to be processed (the length of the lecturer turn and the amount of silence in the recordings) it might not be feasible to improve upon a "dummy" system with the current state of the art diarization systems.
- MDM using speech/non-speech detection (mdm\_c-spnspone): This differs from the primary submission only on the use of the speech/non-speech (spnsp) detector to eliminate the areas

of non-speech. On the development data it was observed that non-speech regions were only labelled (in the hand-made references) when there was a change of speakers, which never happened for the “all lecturing” sections. In a real system though it is important to detect these silences and not attribute them to speakers. This submission is meant to complement the previous one by trying to improve performance where between-speech silences are marked.

- MDM using only the TableTop microphone (mdm\_c-ttoppur): From the available five microphones in the lecture room, one microphone (labelled as “TableTop” microphone) is clearly of much better quality than all the others (which can be found via an SNR comparison among the channels). It is located in a different part of the room and is of a different kind, which could be the reason for its better performance. In the evaluation data it was found by using an SNR estimator and the standard diarization is used on it. No spnsp detection was used in this system.
- SDM using the SDM channel with a minimum duration of 12 seconds for each cluster (sdm\_c-pur12s): This uses the clustering system on the SDM channel. It didn’t use the spnsp detector either. It was observed that using a minimum duration of 12 seconds, the issue of silences marked as speech in the reference files could be bypassed, and force the system to end with fewer clusters.
- MSLA with standard filter&sum (msla\_c-nwsdpur12s): In order to combine the various available speaker-localization arrays, we used the filter&sum processing, using a random channel from one of the arrays as the reference channel. The enhanced channel obtained was then clustered using the 12 second minimum duration system.
- MSLA with weighted filter&sum (msla\_c-wsdpur12s): In the time between the conference room and lecture room submissions, experiments were performed with a first version of the weighted filter&sum algorithm as presented in this thesis. It was applied to the MSLA channels in this system.

## RT05s Official Performance Scores

The main metric used for the RT05s evaluation was the Diarization Error Rate (DER) not taking into account the speaker overlap regions. The DER scores as they were released by NIST are shown in the ninth column of table 7.1, together with a summary of each system’s characteristics. The numbers in the tenth column reflect improvements after small bug fixes right after the evaluation, mainly coming from problems in two of the meetings.

---

<sup>1</sup>This system uses a weighted version of filter&sum using correlations (slightly different from the one presented in this thesis).

| System ID    | room type | Task | Submit type | Delay &sum       | # Initial clusters | Acoustic min. dur. | Mics used | DER    | post-eval DER |
|--------------|-----------|------|-------------|------------------|--------------------|--------------------|-----------|--------|---------------|
| p-dspursys   | Conf.     | MDM  | Primary     | YES              | 10                 | 3 sec              | All       | 18.56% | 16.33%        |
| p-pursys     | Conf.     | SDM  | Primary     | NO               | 10                 | 3 sec              | SDM       | 15.32% | —             |
| p-omnione    | Lect.     | MDM  | Primary     | NO               | n/a                | n/a                | n/a       | 12.21% | —             |
| c-spnspone   | Lect.     | MDM  | Contrast    | NO               | n/a                | n/a                | n/a       | 12.84% | —             |
| c-ttoppur    | Lect.     | MDM  | Contrast    | NO               | 5                  | 5 sec              | Tabletop  | 10.41% | 10.21%        |
| p-omnione    | Lect.     | SDM  | Primary     | NO               | n/a                | n/a                | n/a       | 12.21% | —             |
| c-pur12s     | Lect.     | SDM  | Contrast    | NO               | 5                  | 12 sec             | SDM       | 10.43% | 10.47%        |
| p-omnione    | Lect.     | MSLA | Primary     | NO               | n/a                | n/a                | n/a       | 12.21% | —             |
| c-nwsdpur12s | Lect.     | MSLA | Contrast    | YES              | 5                  | 12 sec             | All       | 9.98%  | 9.66%         |
| c-wsdpur12s  | Lect.     | MSLA | Contrast    | YES <sup>1</sup> | 5                  | 12 sec             | All       | 9.99%  | 9.78%         |

Table 7.1: Systems summary description and DER on the evaluation set for RT05s

In figures 7.1 and 7.2 the DER scores are shown for each one of the excerpts used in the evaluations for conference and lecture room data. The different excerpts are shown in the horizontal axis and the DER in the vertical axis, showing one curve for each one of the presented systems as described before. In the lecture room data the table omits the full meeting names and just show the terminations, which indicates the content of the meeting. Excerpts terminated with “E1” or “E3” only contain the lecturer and therefore it is easier for the system to obtain a perfect diarization.

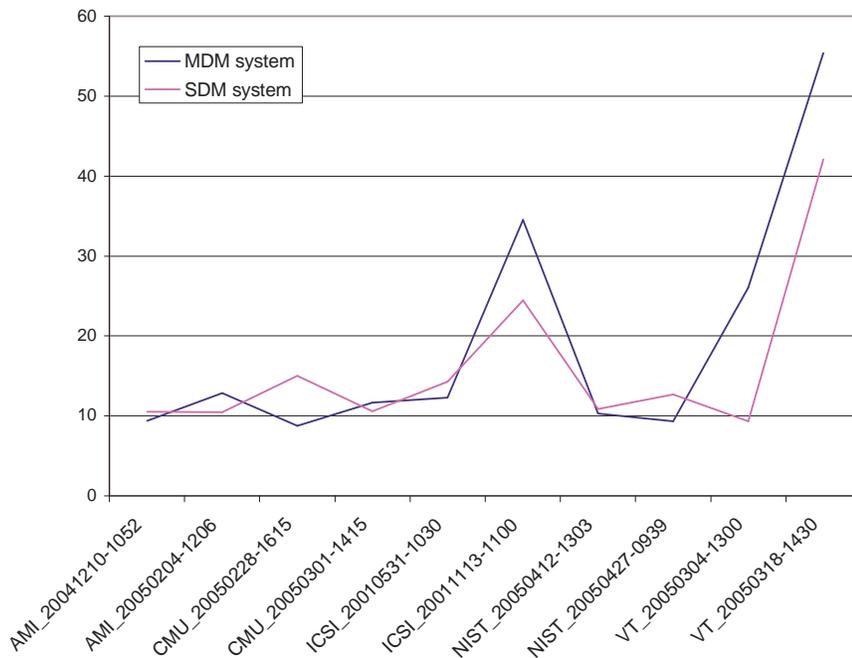


Figure 7.1: DER Break-down by meeting for the RT05s conference data

The use of filter&sum to enhance the signal before doing the clustering turned out to be a bad choice for the conference room systems, as the SDM DER is smaller than the MDM. This was explained due to the big difference between the quality of the signal of the different microphones. When using the best quality microphone as the SDM channel it is difficult to



In the following list the main characteristics of the systems presented are explained. Across tasks, systems with the same ID are equal or very similar, just differing on a few parameters. Their characteristics are:

**p-wdels:** This is the primary system presented this year for all multi-microphone conditions. It uses most of the proposed improvements of this thesis, and all changes in the diarization code from last year's evaluation.

**c-newspnspdelay:** This system is presented for the multi-microphone cases and is composed of RT05s evaluation code using the new filter&sum algorithm, this year's hybrid speech/non-speech detector and taking advantage of the delays for clustering. It uses a minimum duration of 3 seconds, 1/5 initial Gaussian mixtures for delays/acoustics and a split weight of 0.1/0.9 between the streams fixed for all meetings. It is intended to compare the improvements of using delays in the system compared to last year's performance.

**c-wdelsfix:** This system is identical to p-wdels in all parts except the decision of the initial number of clusters, which is fixed to 16 and 10 clusters for conference and lecture rooms, respectively. It intends to compare the robustness of the initial number of clusters selection.

**c/p-nodels:** This system contains all of RT06s improvements with respect to filter&sum (when available, in MDM), speech/non-speech detection and other diarization algorithms except the inclusion of the delays as an extra feature stream.

**c-oldbase:** This system uses all improvements in filter&sum (when available, in MDM) and speech/non-speech detection while using the RT05s core speaker diarization system. It is meant to serve as a baseline result for RT06s systems.

**c-guessone:** This system guesses one speaker for all of the show. In RT05s this was presented as the primary system for lecture room data, showing the need to beat this system in order to think of speaker diarization in the lecture data as a reasonable task. In RT06s it is also presented as a baseline lecture-room system to be compared with the other lecture-room systems.

## RT06s Official Performance Scores

In this section the NIST official scores are shown for all of the ICSI systems presented in the RT06s evaluation in the speaker diarization (SPKR) task and the speech activity detection (SAD) task. In RT06s the main metric used was DER including the speaker overlap regions. In tables 7.2 and 7.3, the SPKR results are shown both for conference and lecture room data, and in table 7.4 the SAD results are shown. During the development of the systems for RT06s focus was switched at using forced-alignments as reference segmentations instead of hand-alignments,

which were believed to be less reliable. In all cases in the results tables they show both the official hand-made references and the forced-alignment references.

In general, results for RT06s using hand-alignments were much worse than in previous years for conference room, which was not so pronounced when evaluating results using the forced alignments. This might be due to the increased complexity of the data and of a decrease in the quality of the hand-generated transcriptions for RT06s evaluation.

| Cond. | System ID      | %DER MAN | %DER FA |
|-------|----------------|----------|---------|
| MDM   | p-wdels        | 35.77%   | 19.16%  |
|       | c-newspnsdelay | 35.77%   | 20.03%  |
|       | c-wdelsfix     | 38.26%   | 23.32%  |
|       | c-nodels       | 41.93%   | 27.46%  |
|       | c-oldbase      | 42.36%   | 27.01%  |
| SDM   | p-nodels       | 43.59%   | 28.25%  |
|       | c-oldbase      | 43.93%   | 28.21%  |

Table 7.2: Results for RT06s Speaker Diarization, conference room environment

| Cond. | System ID  | %DER MAN | %DER MAN(subset) | %DER FA(subset) |
|-------|------------|----------|------------------|-----------------|
| ADM   | p-wdels    | 12.36%   | 11.54%           | 10.56%          |
|       | c-nodels   | 10.43%   | 10.60%           | 9.71%           |
|       | c-wdelsfix | 11.96%   | 12.73%           | 11.58%          |
|       | c-guessone | 25.96%   | 23.36%           | 24.51%          |
| MDM   | p-wdels    | 13.71%   | 11.63%           | 10.97%          |
|       | c-nodels   | 12.97%   | 13.80%           | 13.09%          |
|       | c-wdelsfix | 12.75%   | 12.95%           | 12.34%          |
|       | c-guessone | 25.96%   | 23.36%           | 24.51%          |
| SDM   | p-nodels   | 13.06%   | 12.47%           | 11.69%          |
|       | c-guessone | 25.96%   | 23.36%           | 24.51%          |
| MSLA  | p-guessone | 25.96%   | 23.36%           | 24.51%          |

Table 7.3: Results for RT06s Speaker Diarization, lecture room environment

In the SPKR task for conference room a substantial improvement can be seen between the first three systems in MDM and the last two due to using delays as features in diarization. In lecture room data (Table 7.3, third column) the use of delays affects negatively the performance, possibly due to the existence of people moving around the room (delays consider a different speaker for each location).

Figure 7.3 shows the DER per meeting for each of the presented systems. It is interesting to observe that the primary MDM system (mdm\_p-wdels) obtains flatter scores for all the shows than using last year's system, labelled as mdm\_c-newspnsdelay. Both are shown in dashed lines in figure 7.3.

In general the more microphones available for processing, the better the results. As the diarization system is the same, the improvement is thanks to the filter&sum processing. This is

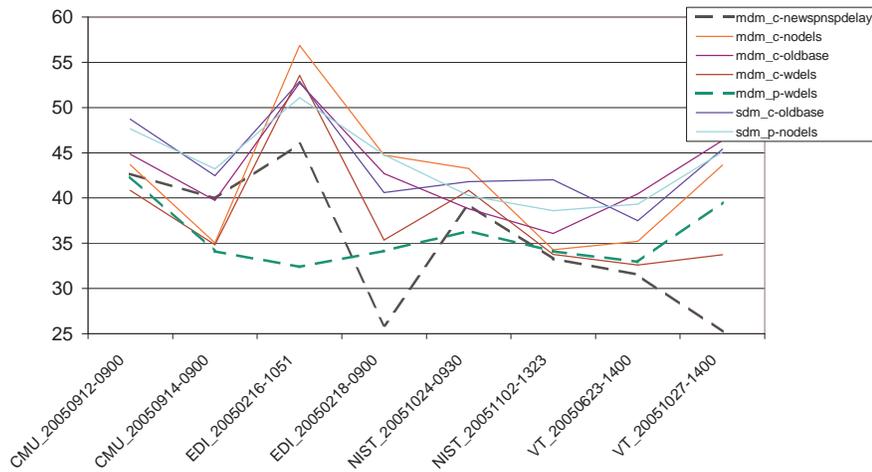


Figure 7.3: *DER break-down by show for the RT06s conference data*

clear in the conference room data, while in the lecture room data results are mixed. It is believed that this is due to the big difference in quality between the microphone used in SDM and all others.

In the lecture room results shown in Table 7.3 a comparison is made between the manual and forced-alignment DER for all systems submitted. The third column shows the results using the latest release of the manual reference segmentations (18 meeting segments). When generating the forced-alignments using the IHM channels from each individual speaker we could not produce them for the meeting segments containing speakers not wearing any headset microphone. The last column shows results using forced-alignment references for a subset of 17 meeting segments containing all speakers who wore a headset microphone. The second to last column shows results using this same subset and using hand-alignments for comparison purposes.

Results using FA references are much better than using hand-alignments in the conference room, while they remain similar in lecture room (with a constant improvement of 0.5% to 1% for FA). It is believed that the conference room manual references contain many human-created problems, which were filtered out in the lecture room references after several redistributions of references.

Figure 7.4 shows the break-down of the DER for all presented systems for the lecture room data. Some meetings are much harder to process than others, creating spikes in the DER curves, more or less pronounced depending on the system. In some cases the ADM systems perform as well in these “hard” meetings as in the easier ones.

On the other hand, in table 7.4 results are shown for systems on conference and lecture room data for the SAD task, using the new speech/non-speech detector developed for RT06s evaluation.



phones used, thanks to the filter&sum module. When comparing the forced-alignment with the hand-alignment subset the first group keeps a better balance between misses and false alarms, indicating that parameters defined in development translate robustly to the evaluation data.

Overall, for RT06s there was a big improvement with the use of delays between microphones as a feature in the diarization process for conference room data, while mixed results were obtained in lecture room. Also, a general improvement was observed using filter&sum on as many microphone signals as possible.

### 7.3 Pros and Cons of the NIST Evaluations

I strongly believe in the advantages behind any evaluation where different independent research groups work towards solving a common problem. But as much as I think that they are beneficiary, there are some issues that could be improved.

Participating in an evaluation campaign constitutes a wonderful opportunity for a research group to get in touch with people that work in the same topic of research, and therefore establish links and collaborations afterwards in other projects or in following evaluations. For example, in RT05s and RT06s the AMI team participated in the Speech-to-text (STT) task with contributions from multiple labs affiliated to the AMI project. Another good example is the ELISA consortium, constituted by four labs located in France which have shared expertise and built systems together for speaker diarization for years.

It is also a good framework to be able to share resources between research groups that allows for better systems to be created and for more systems to be at the top performance possible. This was the case in RT06s when Karlsruhe University shared the output of their beamforming system in order allow other labs to obtain results and perform research with the MM3A microphone array.

By participating in the evaluation campaigns it is beneficial for researchers as it sets a deadline for the systems to be ready, and allows a post-evaluation period when “almost-done” research can be finished and presented at the evaluation workshop. This although, can be seen as prejudicial for research groups involved in too many overlapping evaluations and other projects.

One drawback of the current rich transcription evaluations is the reduced number of participants in some of the tasks. This has been tried to address by setting up smaller tasks like Speech Activity Detection (SAD) in which many groups participated in RT06s.

By repeating the evaluations in successive years it allows technology and new ideas brought in by one group to be used by another with the purpose of solving/improving the problem at hand. Baseline tools and systems should be made available to research groups with a willingness

to participate in order to allow them to obtain competitive results without the need to building a whole system.



# Chapter 8

## Conclusions

In this chapter, first a general review is given on the improvements achieved at the end of this thesis. Then all the objectives proposed in the introduction are reviewed and their success is analyzed. Finally, future work still to be done is proposed.

### 8.1 Overall Thesis Final Review

This PhD thesis verses about the topic of speaker diarization for meetings. While answering to the question “Who spoke when?”, the presented speaker diarization system is able to process a variable number of microphones spread around the meeting room and determine the optimum output without any prior knowledge of the number of speakers or their identities.

The presented system uses as baseline the technology in speaker diarization for broadcast news existent at the International Computer Science Institute (ICSI) and adapts it to the meetings domain by developing new algorithms and improving existent ones to adapt the system to the desired meetings environment. While prior work in the topic of speaker diarization for meetings proposed some sorts of parallel diarization processing of the acoustics and a fusion of the multiple channel outputs, the proposed system uses acoustic beamforming to obtain an “enhanced” single channel and information about the speaker positions in order to use them combined in a single-channel speaker diarization process.

Then the system discards non-speech segments using a new hybrid speech/non-speech detector and processes both acoustics and speaker position information. Algorithms include automatic algorithms for models complexity selection, initialization and training, number of initial clusters and their initial segments, frame and segment purification algorithms and others.

The development of the system was closely linked to participation in the speaker diarization evaluations in Rich Transcription (RT) for meetings proposed by NIST in 2005 and 2006. In both submissions the systems proposed by ICSI both for lecture and conference room data, with

various numbers of microphones, obtained consistently good results.

Experiments were done using the NIST Rich Transcription evaluations datasets to analyze the suitability of each individual module, obtaining results that can be easily compared with other systems and implementations. A 41.15% relative improvement is reported for the development set comparing the system at the start of the thesis to the optimum system proposed. A 25.45% relative improvement is reported for the evaluation set.

## 8.2 Review of Objectives Completion

Upon the thesis start (as stated in the introduction) a set of objectives was set. At this point these were all successfully completed and will be reviewed in the following paragraphs.

In general, a successful system was implemented basing it on the broadcast news technology available at ICSI at the start of the thesis. During this process most of the differences between broadcast news and meetings were analyzed and algorithms were proposed to bridge the gap between both. These were, for example, the multi-channel setup in the meetings versus a single broadcast news channel, the different nature of the non-speech data to be detected and the existence of shorter (in average) speaker. When developing the system it was made modular so that the acoustic beamformer, the speech/non-speech detector and the speaker diarization modules were independent from each other and passing the information between them in files. This allowed the use of the beamforming module for the automatic speech recognition system for meetings, with good results.

Two main ideals that were already in place at ICSI at the start of the thesis were followed at heart. These were making the system be easy to adapt to new domains and that parameters should be robust and not flaky. In terms of easy adaptation, as has been already mentioned, the system was developed using separate blocks to allow for an easy recombination depending on the necessities. In fact, already within the meetings domain, the same speaker diarization module and the speech/non-speech module were reused for SDM and MDM conditions, either using the beamforming as an initial step or not. The core speaker diarization module was kept very similar to the broadcast news system, therefore it could be readapted to that domain with little effort.

In terms of robustness and lack of flakiness, they are problems present in many current speaker diarization systems under research nowadays. Mostly with the inclusion of the proposed new diarization module algorithms it has been shown in the experiments that final results on development and test follow each other closely, showing an increase in robustness from the start of the thesis work. Regarding flakiness, some parameters were defined to substitute others which experienced important differences in Diarization Error Rate (DER) when slightly modifying its

value. The DER value accounts for the percentage of incorrectly assigned time. With the new parameters, in many cases, it was shown that the DER curves were flatter, reducing therefore the flakiness. In some other cases there is still work to do.

Although the system needs the use of development data in order to tune some of the parameters in it, with the development of a hybrid speech/non-speech detector it does not require anymore the use of any external training data. This speaks also in favor of self-sufficiency of the system, which is another ideal followed during its implementation, very much in tune with the capability for fast adaptation to new domains and robustness to changes in the data being tested.

Both in 2005 and 2006 the speaker diarization system entered the NIST Rich Transcription (RT) evaluations where a common task and common datasets were processed by multiple research laboratories. In both entries the ICSI system performed very well. This was established as a goal or milestone in order to push the research and development of the system to be available for the evaluations. In 2005 the main improvement consisted on the development of the initial version of the beamforming system. In 2006 it was a set of improvements to the beamforming and many changes made to the speaker diarization module, as well as a totally new speech/non-speech detector.

As important as the technical improvements and innovations are the tasks to increase public awareness on the system and algorithms being proposed. To this respect, the RT evaluations are a wonderful way to meet people from the same research area and to expose one's research to the community. Another very important way is the publication of articles in conferences and technical magazines. From the start of this thesis work more than 15 papers have been accepted for publication which explain the different improvements and capabilities of the system.

Yet another way is the transfer of technology or knowledge between research labs, which allows other researcher to build on top of pre-established research from other researchers. This was the case of TNO-Twente research group (within the AMI project group) which implemented part of their RT06s contribution based on ICSI's system, or LIA (Avignon) which experimented with the segment purification algorithm originally proposed in RT05s ICSI's submission. Also in this group is the direct transfer of resources by means of system source code, as it was originally done by IDIAP to bring to ICSI the initial speaker diarization system (thanks to Jitendra Ajmera), and was followed recently by the author to take it to the University of Washington (UW). Finally, recently the diarization system has been adapted for speaker tracking used in a Spanish evaluation task within UPC ??.

### 8.3 Possible Future Work Topics

As with everything else in life, there is more that could be done, and it could not be otherwise in this thesis.

One topic within the meeting domain processing that has received quite some attention recently is speaker overlap detection. As such, it refers to the detection of the segments where more than one speaker is talking at the same time, and the output of an appropriate ID for each participant. In NIST RT06s evaluation the main metric included overlap for the first time and multiple research labs (including the author) researched techniques for its detection without any success in reducing the overall error. There is still work to be done in detecting when more than one person is speaking, which should come from the beamforming module (where speakers are well determined by their location) and from the diarization module (where data with multiple overlapping speakers has special acoustic properties compared to single speakers). Also in ASR systems for meetings it would be very beneficiary to create multiple signals in overlap regions, each one derived from the steering to every speaker.

Another area where there research should be directed is the creation of strong links between ASR transcription output and diarization. Although it is well established the use of diarization algorithms to help the ASR systems in model adaptation, it has only been briefly studied the use of ASR for diarization. It could be useful in a number of areas, like the definition of possible speaker change-points (ranging from boundaries at word level to discourse level), or in the assignment of speaker ID's (or the correct name) based on the transcription content (in which LIMSI has done some research). Also, both areas could benefit by the combination of plausible speaker labels with ASR N-best words for each instant, it being useful at decoding level to reduce the errors in both the ASR and diarization tasks.

In the topic of discourse modeling, speaker diarization could benefit from research on ways to model the turn-taking between the speakers. By using information at a higher level than pure acoustics, the transition probabilities between speakers could be appropriately set to help the decoding. One of such possible high level information is easily noticeable in broadcasted news where an anchor speaker is very probable to speak after every other speaker. Similar analysis could be made in the meetings domain and possibly classify the meetings into several types (more fine grained than the current lecture/conference classification) to apply different models to. Possible types could be: moderated meetings (with one person acting as anchor), structured meetings (people following an order, without anchor) and unstructured (where everyone intervenes at random, supposedly with higher amount of overlap regions). Also, it could be considered to split the meeting into several parts with each person's participation dependent on which part/topic the meeting is in.

One of the objectives of this thesis was to increase the robustness of the system to mismatches between the development data and the test data, and to make the system parameters less sensitive to the processed data by obtaining parameters more linked to the acoustics and by eliminating any model training step from the system. It has been shown in the experiments section that an important step forward has been taken in that direction. There is still more that can be done in this topic towards eliminating as many tuning parameters as possible, letting the algorithms select such parameters solely from the data. Also important is to better understand the underlying processes that lead the system to score very differently depending on every particular meeting (leading to “easy” and “difficult” meetings).

Finally, although current systems in the RT evaluations are defined with no application in mind, trying to be adaptable to any possible application, this poses a burden in the capacity of such systems to obtain the optimum score and makes them more computationally intensive as most of the algorithms used for diarization are iterative. It would be interesting to explore particular areas of application where, for example, the number of speakers in a meeting are known. This particular information would probably change the way that speaker diarization algorithms are designed and would allow for lower DER scores, most probably in the region where speaker identification techniques are nowadays.



## Appendix A

# BIC Formulation for Gaussian Mixture Models

The purpose of this appendix is to show the equivalence between two different representations of the Bayesian Information Criterion (BIC), one based on the likelihood of the data given the models, which allows the models to be arbitrary and as complex as necessary given the task at hand, and another representation only dependent on the sufficient statistics of the data, which considers the case of a single Gaussian modeling the data. These two representations are used alternatively in the bibliography with various modifications, which sometimes cause the results not to be comparable between each other.

Given an acoustic segment  $X_i$  with  $N_i$  acoustic frames, modeled by  $M_i$  which is an arbitrary model with a certain number of free parameters to estimate from the data, given by  $\#(\mathcal{M}_i)$ , which accounts for the complexity of such model. The general BIC expression of such model using the likelihood of the data is given by

$$BIC(\mathcal{M}_i) = \log \mathcal{L}(X_i, \mathcal{M}_i) - \lambda \frac{1}{2} \#(\mathcal{M}_i) \log(N_i) \quad (\text{A.1})$$

Being  $\log \mathcal{L}(X_i, \mathcal{M}_i)$  the log-likelihood of the data given the considered model. The parameter  $\lambda$  is a design parameter which is not part of the original BIC formulation but which is used to change the effect of the penalty term in the formula. Such formula allows the model  $\mathcal{M}_i$  to be of any kind.

If instead it is considered that the model is created by a single Gaussian, eq. A.1 can be rewritten as

$$BIC(\mathcal{M}_i) = -\frac{1}{2} N_i \log(|S_i|) - \frac{N_i}{2} d(1 + \log(2\pi)) - \lambda \frac{1}{2} \#(\mathcal{M}_i) \log(N_i) \quad (\text{A.2})$$

where  $S$  is the covariance matrix representing the data and  $d$  is its dimension. Such formulation

only depends on the sufficient statistics of the data, and therefore its computation is very fast.

Let us progress from equation 2.1 into obtaining equation A.2. Considering that the used model is a single Gaussian with full covariance, one can rewrite eq. 2.1 as

$$BIC(\mathcal{M}_i) = \sum_{n=1}^{N_i} \log p(x_i[n]|\mathcal{M}_i) = \log \prod_{n=1}^{N_i} \left[ \frac{1}{(2\pi)^{d/2} |S|^{1/2}} \exp^{-\frac{1}{2}(x_i[n]-\bar{x})' S^{-1} (x_i[n]-\bar{x})} \right] \quad (\text{A.3})$$

by doing the  $N_i$  products one obtains a sum of terms in the exponential, where each terms is a scalar value. One can use mathematical properties of the trace in order to obtain a closer form for it. As the  $\text{trace}(\text{scalar}) = \text{scalar}$ , it does not change the result.

Let us then consider only the trace of the numerator in the exponent

$$Tr[(x_i[1] - \bar{x})' S^{-1} (x_i[1] - \bar{x}) + \dots + (x_i[N_i] - \bar{x})' S^{-1} (x_i[N_i] - \bar{x})]$$

1. Applying the property that  $Tr[\sum] = \sum Tr[]$

$$\dots = Tr[(x_i[1] - \bar{x})' S^{-1} (x_i[1] - \bar{x})] + \dots + tr[(x_i[N_i] - \bar{x})' S^{-1} (x_i[N_i] - \bar{x})]$$

2. For each trace element applying the circularity of the trace property

$$\dots = Tr[(x_i[1] - \bar{x})' (x_i[1] - \bar{x}) S^{-1}] + \dots + tr[(x_i[N_i] - \bar{x})' (x_i[N_i] - \bar{x}) S^{-1}]$$

3. Applying the property of matrix algebra  $AB+CB = (A+C)B$  one can isolate the inverse covariance matrix. At this point, given the definition of covariance matrix one can identify it in the equation and therefore we obtain

$$\dots = Tr[N_i S S^{-1}] = N_i Tr[\mathbb{I}] = Nd$$

Given this result and going back to the BIC formulation in eq. A.3 and using the log properties

$$\begin{aligned} BIC(\mathcal{M}_i) &= \log \left[ \frac{1}{(2\pi)^{dN_i/2} |S|^{N_i/2}} e^{-N_i d/2} \right] \\ &= -\log(2\pi)^{dN_i/2} - \log(|S|)^{N_i/2} - \frac{N_i d}{2} - \lambda \frac{1}{2} \#(\mathcal{M}_i) \log(N_i) \end{aligned}$$

Obtaining finally

$$BIC(\mathcal{M}_i) = -\frac{1}{2}N_i \log(|S_i|) - \frac{N_i}{2}d(1 + \log(2\pi)) - \lambda \frac{1}{2}\#(\mathcal{M}_i)\log(N_i) \quad (\text{A.4})$$

Which is in fact equation A.2. Note that a factor  $\frac{1}{2}$  applies to each term in the expression. Such factor is sometimes omitted, causing the optimum  $\lambda$  factor to differ in the different implementations.



## Appendix B

# Rich Transcription evaluation datasets

In this appendix a complete list of the data used for the development and test sets in this thesis is listed. This data forms the datasets used by NIST in the RT evaluations, in the conference room recordings subdomain.

Table B.1 shows the complete meeting names and some relevant information about each meeting. The *total time* column indicates the length of the excerpt extracted from each meeting to be used for the evaluation, in seconds. The column titled *effective duration* indicates the length of the speech regions in each one of the meetings as indicated by the forced-alignment reference segmentation files.

| Filename           | Dataset | total duration | effective duration | # speakers | # channels |
|--------------------|---------|----------------|--------------------|------------|------------|
| CMU_20030109-1530  | RT02s   | 661.2          | 428.93             | 4          | 1          |
| CMU_20030109-1600  | RT02s   | 666            | 425.09             | 4          | 1          |
| ICSL_20000807-1000 | RT02s   | 682.15         | 443.70             | 6          | 6          |
| ICSL_20011030-1030 | RT02s   | 689.97         | 411.73             | 10         | 6          |
| LDC_20011121-1700  | RT02s   | 661.5          | 426.51             | 3          | 10         |
| LDC_20011207-1800  | RT02s   | 697.4          | 413.21             | 3          | 4          |
| NIST_20030623-1409 | RT02s   | 674            | 423.42             | 6          | 7          |
| NIST_20030925-1517 | RT02s   | 662.07         | 336.72             | 4          | 7          |
| CMU_20020319-1400  | RT04s   | 602.09         | 274.91             | 6          | 1          |
| CMU_20020320-1500  | RT04s   | 503.63         | 259.27             | 4          | 1          |
| ICSL_20010208-1430 | RT04s   | 599.85         | 369.66             | 7          | 6          |
| ICSL_20010322-1450 | RT04s   | 607.53         | 385.61             | 7          | 6          |
| LDC_20011116-1400  | RT04s   | 601.7          | 411.69             | 3          | 8          |
| LDC_20011116-1500  | RT04s   | 601.5          | 340.50             | 3          | 8          |
| NIST_20020214-1148 | RT04s   | 612.31         | 303.08             | 6          | 7          |
| NIST_20020305-1007 | RT04s   | 616.67         | 386.41             | 7          | 7          |
| AMI_20041210-1052  | RT05s   | 730.802        | 474.97             | 4          | 12         |
| AMI_20050204-1206  | RT05s   | 714.385        | 408.56             | 4          | 16         |
| CMU_20050228-1615  | RT05s   | 721.5          | 428.87             | 4          | 7          |
| CMU_20050301-1415  | RT05s   | 718.479        | 418.79             | 4          | 3          |
| ICSL_20010531-1030 | RT05s   | 731.033        | 442.07             | 7          | 6          |
| ICSL_20011113-1100 | RT05s   | 719.65         | 448.77             | 9          | 6          |
| NIST_20050412-1303 | RT05s   | 727.018        | 352.76             | 10         | 7          |
| NIST_20050427-0939 | RT05s   | 715.65         | 431.06             | 4          | 7          |
| VT_20050304-1300   | RT05s   | 718.968        | 511.54             | 5          | 2          |
| VT_20050318-1430   | RT05s   | 724.619        | 311.78             | 5          | 2          |
| CMU_20050912-0900  | RT06s   | 1071.191       | 686.02             | 4          | 2          |
| CMU_20050914-0900  | RT06s   | 1078.913       | 626.60             | 4          | 2          |
| EDI_20050216-1051  | RT06s   | 1080.065       | 578.00             | 4          | 16         |
| EDI_20050218-0900  | RT06s   | 1090.262       | 604.43             | 4          | 16         |
| NIST_20051024-0930 | RT06s   | 1089.009       | 680.43             | 9          | 7          |
| NIST_20051102-1323 | RT06s   | 1086.517       | 673.01             | 8          | 7          |
| VT_20050623-1400   | RT06s   | 1082.2         | 509.85             | 5          | 4          |
| VT_20051027-1400   | RT06s   | 1065.376       | 511.06             | 4          | 4          |

Table B.1: *Summary of datasets used in the experiments*

# Bibliography

- Adami, A., Burget, L., Dupont, S., Garudadri, H., Grezl, F., Hermansky, H., Jain, P., Kajarekar, S., Morgan, N. and Sivasdas, S.: 2002, Qualcomm-icsi-ogi features for asr, *Proc. International Conference on Speech and Language Processing*.
- Adami, A. G., Kajarekar, S. S. and Hermansky, H.: 2002, A new speaker change detection method for two-speaker segmentation, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Orlando, Florida.
- Aguilo, M.: 2005, *Deteccion de actividad oral en un sistema de diarizacion*, Master's thesis, UPC.
- Ajmera, J.: 2004, *Robust Audio Segmentation*, PhD thesis, Ecole Polytechnique Federale de Lausanne.
- Ajmera, J., Boulard, H. and Lapidot, I.: 2002, Improved unknown-multiple speaker clustering using HMM, *Technical report*, IDIAP.
- Ajmera, J., Lathoud, G. and McCowan, I.: 2004, Clustering and segmenting speakers and their locations in meetings, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Vol. 1, pp. 605–608.
- Ajmera, J., McCowan, I. and Boulard, H.: 2003, Robust speaker change detection, *Technical report*, IDIAP.
- Ajmera, J., McCowan, I. and Boulard, H.: 2004, Robust speaker change detection, *IEEE Signal Processing Letters* **11**(8), 649–651.
- Ajmera, J. and Wooters, C.: 2003, A robust speaker clustering algorithm, *IEEE Automatic Speech Recognition and Understanding Workshop*, US Virgin Islands, USA.
- Anguera, X.: 2005, Xbic: Real-time cross probabilities measure for speaker segmentation, *Technical Report TR-99-2004*, ICSI.

- Anguera, X., Aguilo, M., Wooters, C., Nadeu, C. and Hernando, J.: 2006, Hybrid speech/non-speech detector applied to speaker diarization of meetings, *Speaker Odyssey 06*, Puerto Rico, USA.
- Anguera, X. and Hernando, J.: 2004a, Evolutive speaker segmentation using a repository system, *Proc. International Conference on Speech and Language Processing*, Jeju Island, Korea.
- Anguera, X. and Hernando, J.: 2004b, XBIC: nueva medida para segmentacion de locutor hacia el indexado automatico de la senal de voz, *III Jornadas en Tecnologia del Habla*, Valencia, Spain.
- Anguera, X., Wooters, C. and Hernando, J.: 2005, Speaker diarization for multi-party meetings using acoustic fusion, *IEEE Automatic Speech Recognition and Understanding Workshop*, Puerto Rico, USA.
- Anguera, X., Wooters, C. and Hernando, J.: 2006a, Automatic cluster complexity and quantity selection: Towards robust speaker diarization, *MLMI'06*, Washington DC, USA.
- Anguera, X., Wooters, C. and Hernando, J.: 2006b, Frame purification for cluster comparison in speaker diarization, *MMUA'06*, Toulouse, France.
- Anguera, X., Wooters, C. and Hernando, J.: 2006c, Friends and enemies: A novel initialization for speaker diarization, *Proc. International Conference on Speech and Language Processing*, Pittsburgh, USA.
- Anguera, X., Wooters, C. and Hernando, J.: 2006d, Purity algorithms for speaker diarization of meetings data, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Toulouse, France.
- Anguera, X., Wooters, C. and Pardo, J. M.: 2006a, Robust speaker diarization for meetings: ICSI RT06s evaluation system, *Proc. International Conference on Speech and Language Processing*, Pittsburgh, USA.
- Anguera, X., Wooters, C. and Pardo, J. M.: 2006b, Robust speaker diarization for meetings: ICSI RT06s meetings evaluation system, *RT06s Meetings Recognition Evaluation*, Washington DC, USA.
- Anguera, X., Wooters, C., Peskin, B. and Aguilo, M.: 2005, Robust speaker segmentation for meetings: The ICSI-SRI spring 2005 diarization system, *RT05s Meetings Recognition Evaluation*, Edinburgh, Great Britain.
- Appel, U. and Brandt, A.: 1982, Adaptive sequential segmentation of piecewise stationary time series, *Inf. Sci.* **29**(1), 27–56.

- Attias, H.: 2000, A variational bayesian framework for graphical models, *Advances in Neural information processing systems*. MIT Press, Cambridge.
- Augmented Multiparty Interaction (AMI) website*: 2006.  
\*<http://www.amiproject.org>
- Bakis, R., Chen, S., Gopalakrishnan, P. and Gopinath, R.: 1997, Transcription of broadcast news shows with the IBM large vocabulary speech recognition system, *Speech Recognition Workshop*, pp. 67–72.
- Barras, C., Zhu, X., Meignier, S. and Gauvain, J.-L.: 2004, Improving speaker diarization, *Fall 2004 Rich Transcription Workshop (RT04)*, Palisades, NY.
- Basseville, M. and Nikiforov, I.: 1993, *Detection of abrupt changes-theory and application*, Prentice-Hall.
- Beigi, H. S. and Maes, S. H.: 1998, Speaker, channel and environment change detection, *World Congress on Automation*.
- Beigi, H. S., Maes, S. H. and Sorensen, J. S.: 1998, A distance measure between collections of distributions and its application to speaker recognition, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Detroit, USA.
- Ben, M., Betsler, M., Bimbot, F. and Gravier, G.: 2004, Speaker diarization using bottom-up clustering based on a parameter-derived distance between adapted GMMs, *Proc. International Conference on Speech and Language Processing*, Jeju Island, Korea.
- Bilmes, J. and Zweig, G.: 2002, The graphical models toolkit: an open source software system for speech and time-series processing, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Orlando, FL, USA.
- Bimbot, F. and Mathan, L.: 1993, Text-free speaker recognition using an arithmetic-harmonic sphericity measure, *Eurospeech'93*, Berlin, Germany, pp. 169–172.
- Bonastre, J.-F., Delacourt, P., Fredouille, C., Merlin, T. and Wellekens, C.: 2000, A speaker tracking system based on speaker turn detection for NIST evaluation, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Istanbul, Turkey, pp. 1177–1180.
- Brandstein, M., Adcock, J. and Silverman, H.: 1995, A practical time-delay estimator for localizing speech sources with a microphone array, *Comput. Speech Lang.* **9**, 153–159.
- Brandstein, M. and Griebel, S.: 2001, *Explicit Speech Modeling for Microphone Array Applications*, Springer, chapter 7.

- Brandstein, M. S. and Silverman, H. F.: 1997, A robust method for speech signal time-delay estimation in reverberant rooms, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Munich, Germany.
- Brandstein, M. and Ward, D.: 2001, *Microphone Arrays*, Springer.
- Burger, S., Maclaren, V. and Yu, H.: 2002, The ISL meeting corpus: The impact of meeting type on speech style, *Proc. International Conference on Speech and Language Processing*, Denver, USA.
- Campbell, J. P.: 1997, Speaker recognition: a tutorial, *Proceedings of the IEEE* **1.85**(9), 1437–1462.
- Canseco, L., Lamel, L. and Gauvain, J.-L.: 2005, A comparative study using manual and automatic transcriptions for diarization, *IEEE Automatic Speech Recognition and Understanding Workshop*, San Juan, Puerto Rico.
- Canseco-Rodriguez, L., Lamel, L. and Gauvain, J.-L.: 2004a, Speaker Diarization from Speech Transcripts, *Proc. International Conference on Speech and Language Processing*, Jeju Island, S. Korea, pp. 1272–1275.
- Canseco-Rodriguez, L., Lamel, L. and Gauvain, J.-L.: 2004b, Towards using STT for Broadcast News Speaker Diarization, *Proc. DARPA RT04*, Palisades NY.
- Carter, G., Nuttall, A. H. and Cable, P. G.: 1973, The smoothed coherence transform, *Proc. IEEE (Lett.)* **61**, 1497–1498.
- Cassidy, S.: 2004, The macquarie speaker diarization system for rt04s, *NIST 2004 Spring Rich Transcription Evaluation Workshop*, Montreal, Canada.
- Cettolo, M. and Vescovi, M.: 2003, Efficient audio segmentation algorithms based on the BIC, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*.
- Champagne, B., Bedard, S. and Stephenne, A.: 1996, Performance of time-delay estimation in the presence of room reverberation, *IEEE Transactions on Speech and Audio Processing* .
- Chan, W., Lee, T., Zheng, N. and hua Ouyang: 2006, Use of vocal source features in speaker segmentation, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Toulouse, France.
- Chen, L., Rose, R. T., Parrill, F., Han, X., Tu, J., Huang, Z., Harper, M., Quek, F., McNeill, D., Tuttle, R. and Huang, T.: 2005, Vace multimodal meeting corpus, *MLMI*, Edimburgh, UK.

- Chen, S. S., Gales, M. J. F., Gopinath, R. A., Kanvesky, D. and Olsen, P.: 2002, Automatic transcription of broadcast news, *Speech Communication* **37**, 69–87.
- Chen, S. S. and Gopalakrishnan, P.: 1998, Clustering via the bayesian information criterion with applications in speech recognition, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Vol. 2, Seattle, USA, pp. 645–648.
- Chickering, D. M. and Heckerman, D.: 1997, Efficient approximations for the marginal likelihood of bayesian networks with hidden variables, *Machine Learning* **29**, 181–212.
- CMU Meetings Corpus website*: 2006.  
\*[http://penance.is.cs.cmu.edu/meeting\\_room](http://penance.is.cs.cmu.edu/meeting_room)
- Cognitive Assistant that Learns and Organizes (CALO) website*: 2006.  
\*<http://caloproject.sri.com/>
- Cohen, I. and Berdugo, B.: 2002, Speech enhancement based on a microphone array and log-spectral amplitude estimation, *22nd Convention of Electrical and Electronics Engineers in Israel*.
- Collet, M., Charlet, D. and Bimbot, F.: 2005, A correlation metric for speaker tracking using anchor models, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Philadelphia, USA.
- Computers in the Human Interaction Loop (CHIL) website*: 2006.  
\*<http://chil.server.de>
- Cox, H., Zeskind, R. and Kooij, I.: 1986, Practical supergain, *IEEE Transactions on Acoustics, Speech and Signal Processing* **34**(3), 393–397.
- Cox, H., Zeskind, R. and Owen, M.: 1987, Robust adaptive beamforming, *IEEE Transactions on Acoustics, Speech and Signal Processing* **35**(10), 1365–1376.
- DARPA Effective, Affordable, Reusable Speech-to-Text (EARS)*: 2004.  
\*<http://www.darpa.mil/ipto/programs/ears>
- Delacourt, P., Kryze, D. and Wellekens, C. J.: 1999a, Detection of speaker changes in an audio document, *Eurospeech-1999*, Budapest, Hungary.
- Delacourt, P., Kryze, D. and Wellekens, C. J.: 1999b, Speaker-based segmentation for audio data indexing, *ESCA Workshop on accessing Information in Audio Data*.
- Delacourt, P. and Wellekens, C. J.: 1999, Audio data indexing: Use of second-order statistics for speaker-based segmentation, *IEEE International Conference on Multimedia, Computing and Systems*, Florence, Italy.

- Delacourt, P. and Wellekens, C. J.: 2000, DISTBIC: A speaker-based segmentation for audio data indexing, *Speech Communication: Special Issue in Accessing Information in Spoken Audio* **32**, 111–126.
- Deshayes, J. and Picard, D.: 1986, *Off-line statistical analysis of change-point models using non-parametric and likelihood methods*, Springer-Verlag.
- Digalakis, V., Monaco, P. and Murveit, H.: 1996, Genones: generalized mixture tying in continuous hidden markov model-based speech recognizers, *IEEE transactions on speech and audio processing* **4**(4), 281–289.
- Doclo, S. and Moonen, M.: 2002, Gsvd-based optimal filtering for single and multimicrophone speech enhancement, *IEEE Trans. Signal Processing* **50**, 2230–2244.
- Duda, R. and Hart, P.: 1973, *Pattern classification and Scene analysis*, John Wiley & Sons.
- Dunn, R. B., Reynolds, D. and Quatieri, T. F.: 2000, Approaches to speaker detection and tracking in conversational speech, *Digital signal processing* **10**, 93–112.
- Eckart, C.: 1952, Optimal rectifier systems for the detection of steady signals, *Technical Report Rep SIO 12692, SIO Ref 52-11,1952*, Univ. California, Scripps Inst. Oceanography, Marine Physical Lab.
- Ellis, D. and Liu, J. C.: 2004, Speaker turn detection based on between-channels differences, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*.
- F. Reed, P. F. and Bershada, N.: 1981, Time delay estimation using the lms adaptive filter - static behavior, *IEEE Transactions on Acoustics, Speech and Signal Processing* .
- Fierrez-Aguilar, J., Ortega-García, J. and González-Rodríguez, J.: 2003, Fusion strategies in multimodal biometric verification, *IEEE International Conference on Multimedia and Expo*.
- Fischer, S. and Kammeyer, K.-D.: 1997, Broadband beamforming with adaptive postfiltering for speech acquisition in noisy environments, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*.
- Fiscus, J. G., Ajot, J., Michet, M. and Garofolo, J. S.: 2006, The rich transcription 2006 spring meeting recognition evaluation, *NIST 2006 Spring Rich Transcription Evaluation Workshop*, Washington DC, USA.
- Fiscus, J. G., Garofolo, J., Ajot, J. and Michet, M.: 2006, Rt-06s speaker diarization results and speech activity detection results, *NIST 2006 Spring Rich Transcription Evaluation Workshop*, Washington DC, USA.

- Fiscus, J. G., Radde, N., Garofolo, J. S., Le, A., Ajot, J. and Laprun, C. D.: 2005, The rich transcription 2005 spring meeting recognition evaluation, *NIST 2005 Spring Rich Transcription Evaluation Workshop*, Edimburgh, UK.
- Flanagan, J., Johnson, J., Kahn, R. and Elko, G.: 1994, Computer-steered microphone arrays for sound transduction in large rooms, *Journal of the Acoustic Society of America* **78**, 1508–1518.
- Fredouille, C., Moraru, D., Meignier, S., Besacier, L. and Bonastre, J.-F.: 2004, The NIST 2004 spring rich transcription evaluation: Two-axis merging strategy in the context of multiple distant microphone based meeting speaker segmentation, *NIST 2004 Spring Rich Transcription Evaluation Workshop*, Montreal, Canada.
- Gallardo-Antolin, A., Anguera, X. and Wooters, C.: 2006, Multi-stream speaker diarization systems for the meetings domain, *Proc. International Conference on Speech and Language Processing*, Pittsburgh, USA.
- Gangadharaiyah, R., Narayanaswamy, B. and Balakrishnan, N.: 2004, A novel method for two-speaker segmentation, *Proc. International Conference on Speech and Language Processing*, Jeju, S. Korea.
- Garofolo, J. S., Laprun, C. D. and Fiscus, J. G.: 2004, The rich transcription 2004 spring meeting recognition evaluation, *NIST 2004 Spring Rich Transcription Evaluation Workshop*, Montreal, Canada.
- Gauvain, J.-L., Lamel, L. and Adda, G.: 1998, Partitioning and transcription of broadcast news data, *Proc. International Conference on Speech and Language Processing*, Vol. 4, Sidney, Australia, pp. 1335–1338.
- Gish, H. and Schmidt, M.: 1994, Text-independent speaker identification, *Signal Processing Magazine, IEEE* pp. 18–32.
- Gish, H., Siu, M.-H. and Rohlicek, R.: 1991, Segregation of speakers for speech recognition and speaker identification, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Vol. 2, Toronto, Canada, pp. 873–876.
- Griffiths, L. and Jim, C.: 1982, An alternative approach to linearly constrained adaptive beamforming, *IEEE Trans. on Antenas and Propagation* .
- Hain, T., Johnson, S., Turek, A., Woodland, P. and Young, S. J.: 1998, Segment generation and clustering in the HTK broadcast news transcription system, *DARPA Broadcast News Transcription and Understanding Workshop*, pp. 133–137.

- Heck, L. and Sankar, A.: 1997, Acoustic clustering and adaptation for robust speech recognition, *Eurospeech-97*, Rhodes, Greece.
- Hoshuyama, O., Sugiyama, A. and Hirano, A.: 1999, A robust adaptive beamformer for microphone arrays with a blocking matrix using coefficient-constrained adaptive filters, *IEEE Trans. on Signal Processing*.
- Humaine emotion research website*: 2006.  
\*<http://emotion-research.net/>
- Hung, J., Wang, H. and Lee, L.: 2000, Automatic metric based speech segmentation for broadcast news via principal component analysis, *Proc. International Conference on Speech and Language Processing*, Beijing, China.
- ICSI Meeting Recorder Project: Channel skew in ICSI-recorded meetings*: 2006.  
\*<http://www.icsi.berkeley.edu/dpwe/research/mtgrcdr/chanskew.html>
- ICSI Meetings Recorder corpus*: 2006.  
\*<http://www.icsi.berkeley.edu/Speech/mr>
- Ifeachor, E. and Jervis, B.: 1996, *Digital signal processing: a practical approach*, Addison-Wesley.
- Ikbal, S., Misra, H., Sivadas, S., Hermansky, H., and Boulard, H.: 2004, Entropy based combination of tandem representations for noise robust asr, *Proc. International Conference on Speech and Language Processing*, South Korea.
- improvements of the E-HMM based speaker diarization system for meetings records, T.: 2006, The rich transcription 2006 spring meeting recognition evaluation, *NIST 2006 Spring Rich Transcription Evaluation Workshop*, Washington DC, USA.
- Interactive Multimodal Information Management (IM2) website*: 2006.  
\*<http://www.im2.ch>
- Istrate, D., Fredouille, C., Meignier, S., Besacier, L. and Bonastre, J.-F.: 2005, NIST RT05S evaluation: Pre-processing techniques and speaker diarization on multiple microphone meetings, *NIST 2005 Spring Rich Transcription Evaluation Workshop*, Edinburgh, UK.
- Janin, A., Ang, J., Bhagat, S., Dhillon, R., Edwards, J., Macias-Guarasa, J., Morgan, N., Peskin, B., Shriberg, E., Stolcke, A., Wooters, C. and Wrede, B.: 2004, The icsi meeting project: Resources and research, *ICASP*, Montreal.
- Janin, A., Baron, D., Edwards, J., Ellis, D., Gelbart, D., Morgan, N., Peskin, B., Pfau, T., Shriberg, E., Stolcke, A. and Wooters, C.: 2003, The ICSI meeting corpus, *ICASP*, Hong Kong.

- Janin, A., Stolcke, A., Anguera, X., Boakye, K., Cetin, O., Frankel, J. and Zheng, J.: 2006, The ICSI-SRI spring 2006 meeting recognition system, *Proceedings of the Rich Transcription 2006 Spring Meeting Recognition Evaluation*, Washington, USA.
- Jin, H., Kubala, F. and Schwartz, R.: 1997, Automatic speaker clustering, *DARPA Speech Recognition workshop*, Chantilly, USA.
- Jin, Q., Laskowski, K., Schultz, T. and Waibel, A.: 2004, Speaker segmentation and clustering in meetings, *NIST 2004 Spring Rich Transcription Evaluation Workshop*, Montreal, Canada.
- Johnson, D. and Dudgeon, D.: 1993, *Array signal processing*, Prentice Hall.
- Johnson, S.: 1999, Who spoke when? - automatic segmentation and clustering for determining speaker turns, *Eurospeech-99*, Budapest, Hungary.
- Johnson, S. and Woodland, P.: 1998, Speaker clustering using direct maximization of the MLLR-adapted likelihood, *Proc. International Conference on Speech and Language Processing*, Vol. 5, pp. 1775–1779.
- Juang, B. and Rabiner, L.: 1985, A probabilistic distance measure for hidden markov models, *AT&T Technical Journal* 64, AT&T.
- Kaneda, Y.: 1991, Directivity characteristics of adaptive microphone-array for noise reduction (amnor), *Journal of the Acoustical Society of Japan* 12(4), 179–187.
- Kaneda, Y. and Ohga, J.: 1986, Adaptive microphone-array system for noise reduction, *IEEE Trans. on Acoustics, Speech, and Signal Processing* .
- Kass, R. E. and Raftery, A. E.: 1995, Bayes factors, *Journal of the American Statistics association* 90, 773–795.
- Kataoka, A. and Ichirose, Y.: 1990, A microphone array configuration for anmor (adaptive microphone-array system for noise reduction), *Journal of the Acoustical Society of Japan* 11(6), 317–325.
- Kemp, T., Schmidt, M., Westphal, M. and Waibel, A.: 2000, Strategies for automatic segmentation of audio data, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Istanbul, Turkey, pp. 1423–1426.
- Kim, H.-G., Ertelt, D. and Sikora, T.: 2005, Hybrid speaker-based segmentation system using model-level clustering, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Philadelphia, USA.
- Knapp, C. H. and Carter, G. C.: 1976, The generalized correlation method for estimation of time delay, *IEEE Transactions on Acoustics, Speech and Signal Processing* ASSP-24(4), 320–327.

- Kohonen, T.: 1990, The self-organizing map, *Proceedings of the IEEE* **78**(9), 1464–1480.
- Krim, H. and Viberg, M.: 1996, Two decades of array signal processing research, *IEEE Signal Processing Magazine* pp. 67–94.
- Kristjansson, T., Deligne, S. and Olsen, P.: 2005, Voicing features for robust speech detection, *Proc. International Conference on Speech and Language Processing*, Lisbon, Portugal.
- Kubala, F., Jin, H., Matsoukas, S., Gnuyen, L., Schwartz, R. and Machoul, J.: 1997, The 1996 BBN byblos HUB-4 transcription system, *Speech Recognition Workshop*, pp. 90–93.
- Lapidot, I.: 2003, SOM as likelihood estimator for speaker clustering, *Eurospeech*, Geneva, Switzerland.
- Lapidot, I., Gunterman, H. and Cohen, A.: 2002, Unsupervised speaker recognition based on competition between self-organizing-maps, *IEEE Transactions on Neural Networks* **13**(4), 877–887.
- Lathoud, G. and McCowan, I. A.: 2003, Location based speaker segmentation, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*.
- Lathoud, G., McCowan, I. and Odobez, J.: 2004, Unsupervised location-based segmentation of multi-party speech, *ICASSP-NIST Meeting Recognition Workshop*.
- Lathoud, G., Odobez, J.-M. and McCowan, I.: 2004, Short-term spatio-temporal clustering of sporadic and concurrent events, *Technical Report IDIAP-RR 04-14*, IDIAP.
- Lee, K.-F.: 1998, *Large vocabulary speaker-independent continuous speech recognition: the SPHINX system*, PhD thesis, Carnegie Mellon University, Pittsburgh, PA, USA.
- Leeuwen, D. A. V. and Huijbregts, M.: 2006, The AMI speaker diarization system for NIST RT06s meeting data, *NIST 2006 Spring Rich Transcription Evaluation Workshop*, Washington DC, USA.
- Li, Q., Zheng, J., Tsai, A., and Zhou, Q.: 2002, Robust endpoint detection and energy normalization for real-time speech and speaker recognition, *IEEE Transactions on Speech and Audio Processing* **10**(3).
- Li, X.: 2005, *Combination and Generation of Parallel Feature Streams for Improved Speech Recognition*, PhD thesis, ECE Department, CMU.
- Liu, D. and Kubala, F.: 1999, Fast speaker change detection for broadcast news transcription and indexing, *Eurospeech-99*, Vol. 3, Budapest, Hungary, pp. 1031–1034.

- Lopez, J. F. and Ellis, D. P. W.: 2000a, Using acoustic condition clustering to improve acoustic change detection on broadcast news, *Proc. International Conference on Speech and Language Processing*, Beijing, China.
- Lopez, J. F. and Ellis, D. P. W.: 2000b, Using acoustic condition clustering to improve acoustic change detection on broadcast news, *Proc. International Conference on Speech and Language Processing*, Beijing, China.
- Lu, L., Li, S. Z. and Zhang, H.-J.: 2001, Content-based audio segmentation using support vector machines, *ACM Multimedia Conference*, pp. 203–211.
- Lu, L. and Zhang, H.-J.: 2002a, Real-time unsupervised speaker change detection, *ICPR'02*, Vol. 2, Quebec City, Canada.
- Lu, L. and Zhang, H.-J.: 2002b, Speaker change detection and tracking in real-time news broadcasting analysis, *ACM International Conference on Multimedia*, pp. 602–610.
- Lu, L., Zhang, H.-J. and Jiang, H.: 2002, Content analysis for audio classification and segmentation, *IEEE Transactions on Speech and Audio Processing* **10**(7), 504–516.
- MacKay, D. J. C.: 1997, Ensemble learning for hidden Markov models. <http://www.inference.phy.cam.ac.uk/mackay/abstracts/ensemblePaper.html>.
- Malegaonkar, A., Ariyaeeinia, A., Sivakumaran, P. and Fortuna, J.: 2006, Unsupervised speaker change detection using probabilistic pattern matching, *IEEE Signal Processing Letters* **13**(8), 509–512.
- Marro, C., Mahieux, Y. and Simmer, K.: 1998, Analysis of noise reduction and dereverberation techniques based on microphone arrays with postfiltering, *IEEE Trans. on Speech and Audio Processing*.
- McCowan, I.: 2001, *Robust Speech Recognition using microphone arrays*, PhD thesis, Queensland University of Technology, Australia.
- McCowan, I. A., Pelecanos, J. and Sridharan, S.: 2001, Robust speaker recognition using microphone arrays, *IEEE Speaker Odyssey recognition workshop*.
- McCowan, I., Gatica-Perez, D., Bengio, S., Lathoud, G., Barnard, M. and Zhang, D.: 2005, Automatic analysis of multimodal group actions in meetings, *IEEE Trans. on Pattern Analysis and Machine Intelligence* **27**, 305–317.
- McCowan, I., Marro, C. and Mauuary, L.: 2000, Robust speech recognition using near-field superdirective beamforming with post-filtering, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Vol. 3, pp. 1723–1726.

- McCowan, I., Moore, D. and Sridharan, S.: 2000, Speech enhancement using near-field superdirectivity with an adaptive sidelobe canceler and post-filter, *Australian International Conference on Speech Science and Technology*, pp. 268–273.
- Meignier, S., Bonastre, J.-F. and Igournet, S.: 2001, E-HMM approach for learning and adapting sound models for speaker indexing, *A speaker Odyssey*, Chania, Crete, pp. 175–180.
- Meignier, S., Moraru, D., Fredouille, C., Besacier, L. and Bonastre, J.-F.: 2004, Benefits of prior acoustic segmentation for automatic speaker segmentation, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Montreal, Canada.
- Meinedo, H. and Neto, J.: 2003, Audio segmentation, classification and clustering in a broadcast news task, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Hong-Kong, China.
- Metze, F., Fugen, C., Pan, Y., Schultz, T. and Yu, H.: 2004, The ISL RT-04S meetings transcription system, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Montreal, Canada.
- Mirghafori, N., Stolcke, A., Wooters, C., Pirinen, T., Bulyko, I., Gelbart, D., Graciarena, M., Otterson, S., Peskin, B. and Ostendorf, M.: 2004, From switchboard to meetings: Development of the 2004 ICSI-SRI-UW meeting recognition system, *Proc. International Conference on Speech and Language Processing*, Jeju Island, Korea.
- Mirghafori, N. and Wooters, C.: 2006, Nuts and flakes: A study of data characteristics in speaker diarization, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Toulouse, France.
- Misra, H., Bourlard, H., and Tyagi, V.: 2003, New entropy based combination rules in hmm/ann multi-stream asr, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Hong Kong.
- Moh, Y., Nguyen, P. and Junqua, J.-C.: 2003, Towards domain independent speaker clustering, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Hong Kong.
- Moraru, D., Ben, M. and Gravier, G.: 2005, Experiments on speaker tracking and segmentation in radio broadcast news, *Proc. International Conference on Speech and Language Processing*, Lisbon, Portugal.
- Moraru, D., Besacier, L., Meignier, S., Fredouille, C. and francois Bonastre, J.: 2004, Speaker diarization in the elisa consodrium over the last 4 years, *NIST 2004 Spring Rich Transcription Evaluation Workshop*, Montreal, Canada.

- Moraru, D., Meignier, S., Besacier, L., Bonastre, J.-F. and Magrin-Chagnolleau, I.: 2002, The ELISA consortium approaches in speaker segmentation during the NIST 2002 speaker recognition evaluation, *NIST 2002 Spring Rich Transcription Evaluation Workshop*.
- Moraru, D., Meignier, S., Besacier, L., Bonastre, J.-F. and Magrin-Chagnolleau, I.: 2004, The ELISA consortium approaches in speaker segmentation during the NIST 2002 speaker recognition evaluation, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Montreal, Canada.
- Moraru, D., Meignier, S., Fredouille, C., Besacier, L. and Bonastre, J.-F.: 2004, The ELISA consortium approaches in broadcast news speaker segmentation during the NIST 2003 rich transcription evaluation, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Montreal, Canada.
- Mori, K. and Nakagawa, S.: 2001, Speaker change detection and speaker clustering using VQ distortion for broadcast news speech recognition, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Vol. 1, Salt Lake City, USA, pp. 413–416.
- Multimodal Meeting Manager (M4) website*: 2006.  
\*<http://www.m4project.org>
- Nakagawa, S. and Suzuki, H.: 1993, A new speech recognition method based on VQ-distortion and hmm, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Vol. 2, Minneapolis, USA, pp. 676–679.
- National Institute for Standards and Technology*: 2006.  
\*<http://www.nist.gov/speech>
- Nguyen, P.: 2003, SWAMP: An isometric frontend for speaker clustering, *NIST 2003 Rich Transcription Workshop*, Boston, USA.
- Nishida, M. and Kawahara, T.: 2003, Unsupervised speaker indexing using speaker model selection based on bayesian information criterion, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Hong Kong.
- NIST Fall Rich Transcription Evaluation website*: 2006.  
\*<http://www.nist.gov/speech/tests/rt/rt2004/fall>
- NIST Fall Rich Transcription on meetings 2006 Evaluation Plan*: 2006.  
\*<http://www.nist.gov/speech/tests/rt/rt2006/spring/docs/rt06s-meeting-eval-plan-V2.pdf>
- NIST MD-eval-v21 DER evaluation script*: 2006.  
\*<http://www.nist.gov/speech/tests/rt/rt2006/spring/code/md-eval-v21.pl>

*NIST Pilot Meeting Corpus website*: 2006.

\*[http://www.nist.gov/speech/test\\_beds/mr\\_proj/meeting\\_corpus\\_1](http://www.nist.gov/speech/test_beds/mr_proj/meeting_corpus_1)

*NIST Rich Transcription evaluations, website*: <http://www.nist.gov/speech/tests/rt>: 2006.

\*<http://www.nist.gov/speech/tests/rt>

*NIST Speech Recognition Evaluation*: 2006.

\*<http://www.nist.gov/speech/tests/spk/index.htm>

*NIST Speech tools and APIs*: 2006.

\*<http://www.nist.gov/speech/tools/index.htm>

*NIST Spring Rich Transcription Evaluation in Meetings website*,  
<http://www.nist.gov/speech/tests/rt/rt2005/spring>: 2006.

\*<http://www.nist.gov/speech/tests/rt/rt2005/spring>

Omar, M. K., Chaudhari, U. and Ramaswamy, G.: 2005, Blind change detection for audio segmentation, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Philadelphia, USA.

Ouellet, P., Boulianne, G. and Kenny, P.: 2005, Fravors of gaussian warping, *Proc. International Conference on Speech and Language Processing*, Lisbon, Portugal.

Pardo, J. M., Anguera, X. and Wooters, C.: 2006a, Speaker diarization for multi-microphone meetings using only between-channel differences, *MLMI 2006*.

Pardo, J. M., Anguera, X. and Wooters, C.: 2006b, Speaker diarization for multiple distant microphone meetings: Mixing acoustic features and inter-channel time differences, *Proc. International Conference on Speech and Language Processing*.

*Pattern analysis, Statistical modeling and Computational learning (Pascal) website*: 2006.

\*<http://www.pascal-network.org/>

Pelecanos, J. and Sridharan, S.: 2001, Feature warping for robust speaker verification, *ISCA Speaker Recognition Workshop odyssey*, Crete, Grece.

Perez-Freire, L. and Garcia-Mateo, C.: 2004, A multimedia approach for audio segmentation in TV broadcast news, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Montreal, Canada, pp. 369–372.

Pwint, M. and Sattar, F.: 2005, A segmentation method for noisy speech using genetic algorithm, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Philadelphia, USA.

- Rentzeperis, E., Stergiou, A., Boukis, C., Pnevmatikakis, A. and Polymenakos, L. C.: 2006, The 2006 athens information technology speech activity detection and speaker diarization systems, *NIST 2006 Spring Rich Transcription Evaluation Workshop*, Washington DC, USA.
- Reynolds, D. A., Singer, E., Carlson, B. A., O'Leary, G. C., McLaughlin, J. J. and Zixxman, M. A.: 1998, Blind clustering of speech utterances based on speaker and language characteristics, *Proc. International Conference on Speech and Language Processing*, Sidney, Australia.
- Reynolds, D. and Torres-Carrasquillo, P.: 2004, The MIT Lincoln Laboratories RT-04F diarization systems: Applications to broadcast audio and telephone conversations, *Fall 2004 Rich Transcription Workshop (RT04)*, Palisades, NY.
- Roch, M. and Cheng, Y.: 2004, Speaker segmentation using the MAP-adapted bayesian information criterion, *Odyssey-04*, Toledo, Spain, pp. 349–354.
- Rombouts, G. and M. Moonen: 2003, Qrd-based unconstrained optimal filtering for acoustic noise reduction, *IEEE Trans. Signal Processing* **83**(9), 1889–1904.
- Rosca, J., Balan, R. and Beaugeant, C.: 2003, Multi-channel psychoacoustically motivated speech enhancement, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*.
- Ross, A., Jain, A. K. and Qian, J. Z.: 2001, Information fusion in biometrics, *3rd International Conference on Audio and Video-Based Person Authentication*.
- Roth, P.: 1971, Effective measurements using digital signal analysis, *IEEE Spectrum* **8**, 62–70.
- Rougui, J., Rziza, M., Aboutajdine, D., Gelgon, M. and Martinez, J.: 2006, Fast incremental clustering of gaussian mixture speaker models for scaling up retrieval in on-line broadcast, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Toulouse, France.
- Sanchez-Bote, J., Gonzalez-Rodriguez, J. and Ortega-Garcia, J.: 2003, A real-time auditory-base microphone array assessed with e-rasti evaluation proposal, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*.
- Sankar, A., Beaufays, F. and Digalakis, V.: 1995, Training data clustering for improved speech recognition, *Eurospeech-95*, Madrid, Spain.
- Sankar, A., Weng, F., Stolcke, Z. R. A. and Grande, R. R.: 1998, Development of SRI's 1997 broadcast news transcription system, *DARPA Broadcast News Transcription and Understanding Workshop*, Landsdowne, USA.

- Schmidt, R.: 1986, Multiple emitter location and signal parameter estimation, *IEEE Transactions on Antennas and Propagation* .
- Schwarz, G.: 1971, A sequential student test, *The Annals of Statistics* **42**(3), 1003–1009.
- Schwarz, G.: 1978, Estimating the dimension of a model, *The Annals of Statistics* **6**, 461–464.
- Shaobing Chen, S. and Gopalakrishnan, P.: 1998, Speaker, environment and channel change detection and clustering via the bayesian information criterion, *Proceedings DARPA Broadcast News Transcription and Understanding Workshop*, Virginia, USA.
- Shinozaki, T. and Ostendorf, M.: 2007, Cross-validation EM training for robust parameter estimation, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing* . submitted.
- sian Cheng, S. and min Wang, H.: 2003, A sequential metric-based audio segmentation method via the bayesian information criterion, *Eurospeech'03*, Geneva, Switzerland.
- sian Cheng, S. and min Wang, H.: 2004, METRIC-SEQDAC: A hybrid approach for audio segmentation, *Proc. International Conference on Speech and Language Processing*, Jeju, South Korea.
- Siegler, M. A., Jain, U., Raj, B. and Stern, R. M.: 1997, Automatic segmentation, classification and clustering of broadcast news audio, *DARPA Speech Recognition Workshop*, Chantilly, pp. 97–99.
- Similar Network of Excellence website*: 2006.  
\*<http://www.similar.cc/cms/default.asp?id=0>
- Sinha, R., Tranter, S. E., Gales, J. J. F. and Woodland, P. C.: 2005, The cambridge university march 2005 speaker diarisation system, *European Conference on Speech Communication and Technology (Interspeech)*, Lisbon, Portugal, pp. 2437–2440.
- Siu, M.-H., Yu, G. and Gish, H.: 1992, An unsupervised, sequential learning algorithm for the segmentation of speech waveforms with multiple speakers, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Vol. 2, San Francisco, USA, pp. 189–192.
- Sivakumaran, P., Fortuna, J. and Ariyaeinia, A.: 2001, On the use of the bayesian information criterion in multiple speaker detection, *Eurospeech'01*, Scandinavia.
- Solomonov, A., Mielke, A., Schmidt, M. and Gish, H.: 1998, Clustering speakers by their voices, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Vol. 2, Seattle, USA, pp. 757–760.

*Speech in noisy environments*: 2006.

\*<http://www.speech.sri.com/projects/spine/>

*Spring 2005 (RT-05S) Rich Transcription Meeting Recognition Evaluation Plan*: n.d.

\*<http://www.nist.gov/speech/tests/rt/rt2005/spring/rt05s-meeting-eval-plan-V1.pdf>

*Spring 2006 (RT-06S) Rich Transcription Meeting Recognition Evaluation Plan*: n.d.

\*<http://www.nist.gov/speech/tests/rt/rt2006/spring/docs/rt06s-meeting-eval-plan-V2.pdf>

Stolcke, A., Anguera, X., Boakye, K., Cetin, O., Grezl, F., Janin, A., Mandal, A., Peskin, B., Wooters, C. and Zheng, J.: 2005, Further progress in meeting recognition: The icsi-sri spring 2005 speech-to-text evaluation system, *RT05s Meetings Recognition Evaluation*, Edinburgh, Great Britain.

Strassel, S. and Glenn, M.: 2004, Shared linguistic resources for human language technology in the meeting domain, *ICASSP-DARPA Meetings Diarization Workshop*, Montreal, Canada.

Sturim, D., Reynolds, D., Singer, E. and J.P.Campbell: 2001, Speaker indexing in large audio databases using anchor models, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Salt Lake City, USA.

Tager, W.: 1998a, *Etudes en traitement d'antenne pour la prise de son*, PhD thesis, Universite de Rennes.

Tager, W.: 1998b, Near field superdirectivity (nfsd), *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 2045–2048.

Tranter, S.: 2005, Two-way cluster voting to improve speaker diarization performance, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Montreal, Canada.

Tranter, S. and Reynolds, D.: 2004, Speaker diarization for broadcast news, *ODYSSEY'04*, Toledo, Spain.

Trees, H. V.: 1968, *Detection Estimation and Modulation Theory*, Vol. 1, Wiley.

Tritschler, A. and Gopinath, R.: 1999, Improved speaker segmentation and segments clustering using the bayesian information criterion, *Eurospeech'99*, pp. 679–682.

Tsai, W.-H., Cheng, S.-S., Chao, Y.-H. and Wang, H.-M.: 2005, Clustering speech utterances by speaker using eigenvoice-motivated vector space models, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Philadelphia, USA.

- Tsai, W.-H., Cheng, S.-S. and Wang, H.-M.: 2004, Speaker clustering of speech utterances using a voice characteristic reference space, *Proc. International Conference on Speech and Language Processing*, Jeju Island, Korea.
- Tsai, W.-H. and Wang, H.-M.: 2006, On maximizing the within-cluster homogeneity of speaker voice characteristics for speech utterance clustering, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Toulouse, France.
- Valente, F.: 2006, Infinite models for speaker clustering, *Proc. International Conference on Speech and Language Processing*, Pittsburgh, USA.
- Valente, F. and Wellekens, C.: 2004, Variational bayesian speaker clustering, *Speaker Odyssey*, Toledo, Spain.
- Valente, F. and Wellekens, C.: 2005, Variational bayesian adaptation for speaker clustering, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Lisbon, Portugal.
- Valin, J., Rouat, J. and Michaud, F.: 2004, Microphone array post-filter for separation of simultaneous non-stationary sources, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*.
- van Leeuwen, D.: 2005, The TNO speaker diarization system system for NIST RT05s for meeting data, *NIST 2005 Spring Rich Transcription Evaluation Workshop*, Edinburgh, UK.
- Vandecatseye, A. and Martens, J.-P.: 2003, A fast, accurate and stream-based speaker segmentation and clustering algorithm, *Eurospeech'03*, Geneva, Switzerland, pp. 941–944.
- Vandecatseye, A., Martens, J.-P. et al.: 2004, The cost278 pan-european broadcast news database, *LREC'04*, Lisbon, Portugal.
- Veen, B. V. and Buckley, K.: 1988, Beamforming: A versatile approach to spacial filtering, *IEEE Transactions on Acoustics, Speech and Signal Processing* .
- Verlinde, P., Chollet, G. and Acheroy, M.: 2000, Multi-modal identity verification using expert fusion, *Information Fusion* 1(1), 17–33.
- Vescovi, M., Cettolo, M. and Rizzi, R.: 2003, A DP algorithm for speaker change detection, *Eurospeech'03*.
- Video analysis and content extraction for defense intelligence (ARDA-VACE II)*: 2006.  
\*<http://www.informedia.cs.cmu.edu/arda/vaceII.html>
- Wactlar, H., Hauptmann, A. and Witbrock, M.: 1996, News on-demand experiments in speech recognition, *ARPA STL Workshop*.

- Wegmann, S., Scattone, F., Carp, I., Gillick, L., Roth, R. and Yamron, J.: 1998, Dragon system's 1997 broadcast news transcription system, *DARPA Broadcast News Transcription and Understanding Workshop*, Landsdowne, USA.
- Wiener and Norbert: 1949, *Extrapolation, Interpolation, and Smoothing of Stationary Time Series*, Wiley.
- Wilcox, L., Chen, F., Kimber, D. and Balasubramanian, V.: 1994, Segmentation of speech using speaker identification, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Vol. 1, Adelaide, Australia, pp. 161–164.
- Willsky, A. S. and Jones, H. L.: 1976, A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems, *IEEE Transactions on Automatic Control* **AC-21**(1), 108–112.
- Woodland, P., Gales, M., Pye, D. and Young, S.: 1997, The development of the 1996 HTK broadcast news transcription system, *Speech Recognition Workshop*, pp. 73–78.
- Wooters, C., Fung, J., Peskin, B. and Anguera, X.: 2004, Towards robust speaker segmentation: The ICSI-SRI fall 2004 diarization system, *Fall 2004 Rich Transcription Workshop (RT04)*, Palisades, NY.
- Wu, T., Lu, L., Chen, K. and Zhang, H.-J.: 2003a, UBM-based incremental speaker adaptation, *ICME'03*, Vol. 2, pp. 721–724.
- Wu, T., Lu, L., Chen, K. and Zhang, H.-J.: 2003b, UBM-based real-time speaker segmentation for broadcasting news, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*.
- Wu, T., Lu, L., Chen, K. and Zhang, H.-J.: 2003c, Universal background models for real-time speaker change detection, *International Conference on Multimedia Modeling*.
- Yamaguchi, M., Yamashita, M. and Matsunaga, S.: 2005, Spectral cross-correlation features for audio indexing of broadcast news and meetings, *Proc. International Conference on Speech and Language Processing*.
- Young, S., Kershaw, D., Odell, J., Ollason, D., Valtchev, V. and Woodland, P.: 2005, *The HTK Book*, Cambridge University Engineering Department.
- Zdansky, J. and Nouza, J.: 2005, Detection of acoustic change-points in audio records via global BIC maximization and dynamic programming, *Proc. International Conference on Speech and Language Processing*, Lisbon, Portugal.

- Zelinski, R.: 1988, A microphone array with adaptive post-filtering for noise reduction in reverberant rooms, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Vol. 5, pp. 2578–2581.
- Zhang, X., Hansen, J. and Rehar, K.: 2004, Speech enhancement based on a combined multi-channel array with constrained iterative and auditory masked processing, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*.
- Zhou, B. and Hansen, J. H.: 2000, Unsupervised audio stream segmentation and clustering via the bayesian information criterion, *Proc. International Conference on Speech and Language Processing*, Vol. 3, Beijing, China, pp. 714–717.
- Zhu, X., Barras, C., Lamel, L. and Gauvain, J.-L.: 2006, Speaker diarization: from broadcast news to lectures, *NIST 2006 Spring Rich Transcription Evaluation Workshop*, Washington DC, USA.
- Zhu, X., Barras, C., Meignier, S. and Gauvain, J.-L.: 2005, Combining speaker identification and bic for speaker diarization, *Proc. International Conference on Speech and Language Processing*, Lisbon, Portugal.
- Zochova, P. and Radova, V.: 2005, Modified DISTBIC algorithm for speaker change detection, *Proc. International Conference on Speech and Language Processing*, Lisbon, Portugal.