KU LEUVEN

ARENBERG DOCTORAL SCHOOL
Faculty of Engineering Science

Signal Processing Algorithms for EEG-based Auditory Attention Decoding

Simon Geirnaert

Supervisors:
Prof. dr. ir. A. Bertrand
Prof. dr. ir. T. Francart

Dissertation presented in partial fulfillment of the requirements for the degree of Doctor of Engineering Science (PhD): Electrical Engineering

May 2022

Signal Processing Algorithms for EEG-based Auditory Attention Decoding

Simon GEIRNAERT

Examination committee:
em. Prof. dr. ir. J.-P. Celis, chair
Prof. dr. ir. A. Bertrand, supervisor
Prof. dr. ir. T. Francart, supervisor
em. Prof. dr. ir. S. Van Huffel
Prof. dr. ir. H. Van hamme
dr. M. Slaney
(Machine Hearing Group, Google Research, USA)
Prof. dr. B. Babadi
(Department of Electrical & Computer Engineering, University of Maryland, USA)

Dissertation presented in partial fulfillment of the requirements for the degree of Doctor of Engineering Science (PhD): Electrical Engineering

May 2022

Abstract

One in five experiences hearing loss. The World Health Organization estimates that this number will increase to one in four in 2050. Luckily, effective hearing devices such as hearing aids and cochlear implants exist with advanced noise suppression and speaker enhancement algorithms that can significantly improve the quality of life of people suffering from hearing loss. State-of-the-art hearing devices, however, underperform in a so-called 'cocktail party' scenario, when multiple persons are talking simultaneously. In such a situation, the hearing device does not know which speaker the user intends to attend to and thus which speaker to enhance and which other ones to suppress. Therefore, a new problem arises in cocktail party problems: determining which speaker a user is attending to, referred to as the *auditory attention decoding* (AAD) problem.

The problem of selecting the attended speaker could be tackled using simple heuristics such as selecting the loudest speaker or the one in the user's look direction. However, a potentially better approach is decoding the auditory attention from where it originates, i.e., the brain. Using neurorecording techniques such as electroencephalography (EEG), it is possible to perform AAD, for example, by reconstructing the attended speech envelope from the EEG using a neural decoder (i.e., the stimulus reconstruction (SR) algorithm). Integrating AAD algorithms in a hearing device could then lead to a so-called 'neuro-steered hearing device'. These traditional AAD algorithms are, however, not fast enough to adequately react to a switch in auditory attention, and are supervised and fixed over time, not adapting to non-stationarities in the EEG and audio data. Therefore, the general aim of this thesis is to develop novel signal processing algorithms for EEG-based AAD that allow fast, accurate, unsupervised, and time-adaptive decoding of the auditory attention.

In the first part of the thesis, we compare different AAD algorithms, which allows us to identify the gaps in the current AAD literature that are partly addressed in this thesis. To be able to perform this comparative study, we develop a new performance metric - the minimal expected switch duration (MESD) - to evaluate

AAD algorithms in the context of adaptive gain control for neuro-steered hearing devices. This performance metric resolves the traditional trade-off between AAD accuracy and time needed to make an AAD decision and returns a single-number metric that is interpretable within the application-context of AAD and allows easy (statistical) comparison between AAD algorithms. Using the MESD, we establish that the most robust currently available AAD algorithm is based on canonical correlation analysis, but that decoding the spatial focus of auditory attention from the EEG holds more promise towards fast and accurate AAD. Moreover, we observe that deep learning-based AAD algorithms are hard to replicate on different independent AAD datasets.

In the second part, we address one of the main signal processing challenges in AAD: unsupervised and time-adaptive algorithms. We first develop an unsupervised version of the stimulus decoder that can be trained on a large batch of EEG and audio data without knowledge of ground-truth labels on the attention. The unsupervised stimulus decoder is iteratively retrained based on its own predicted labels, resulting in a self-leveraging effect that can be explained by interpreting the iterative updating procedure as a fixed-point iteration. This unsupervised but subject-specific stimulus decoder, starting from a random initial decoder, outperforms a supervised subject-independent decoder, and, using subject-independent information, even approximates the performance of a supervised subject-specific decoder. We also extend this unsupervised algorithm to an efficient recursive time-adaptive algorithm, when EEG and audio are continuously streaming in, and show that it has the potential to outperform a fixed supervised decoder in a practical use case of AAD.

In the third part, we develop novel AAD algorithms that decode the spatial focus of auditory attention to provide faster and more accurate decoding. To this end, we use both a linear common spatial pattern (CSP) filtering approach and its nonlinear extension using Riemannian geometry-based classification (RGC). The CSP method achieves a much higher accuracy compared to the SR algorithm at a very fast decision rate. Furthermore, we show that the CSP method is the preferred choice over a similar convolutional neural network-based approach, and is also applicable on different directions of auditory attention, in a three-class problem with different angular domains, using only EEG channels close to the ears, and when generalizing to data from an unseen subject. Lastly, the RGC-based extension further improves the accuracy at slower decision rates, especially in the multiclass problem.

To summarize, in this thesis we have developed crucial building blocks for a plug-and-play, time-adaptive, unsupervised, fast, and accurate AAD algorithm that could be integrated with a low-latency speaker separation and enhancement algorithm, and a wearable, miniaturized EEG system to eventually lead to a neuro-steered hearing device.

Beknopte samenvatting

Een op vijf ervaart gehoorverlies. De Wereldgezondheidsorganisatie schat dat dit aantal zal toenemen tot een op vier in 2050. Gelukkig bestaan er effectieve hoortoestellen zoals hoorapparaten en cochleaire implantaten met geavanceerde ruisonderdrukkingsalgoritmen die de levenskwaliteit van mensen met gehoorverlies significant kunnen verbeteren. State-of-the-art hoortoestellen presteren echter ondermaats in een zogenaamd '*cocktail party*' scenario, waarin meerdere personen tegelijkertijd aan het spreken zijn. In een dergelijke situatie weet het hoortoestel niet naar welke spreker de gebruiker wil luisteren. Bijgevolg weet het niet welke spreker versterkt en welke andere sprekers onderdrukt moeten worden. Dit leidt tot een nieuw probleem in het cocktail party scenario: bepalen naar welke spreker een gebruiker luistert. Dit heet het *auditieve aandachtsdecodering* (AAD) probleem.

De spreker waarvoor aandacht is kan geïdentificeerd worden met behulp van eenvoudige heuristieken zoals het selecteren van de luidste spreker of de spreker in de kijkrichting van de gebruiker. Een mogelijk betere oplossing is het decoderen van de auditieve aandacht daar waar het ontstaat of plaatsvindt, namelijk de hersenen. De auditieve aandacht kan gedecodeerd worden door, bijvoorbeeld, de omhullende van het spraaksignaal waarvoor aandacht is te reconstrueren uit het elektro-encefalogram (EEG) met een neurale decoder (dit is het stimulusreconstructie (SR)-algoritme). De integratie van AAD-algoritmen in een hoortoestel zou dan kunnen leiden tot een zogenaamd 'neurogestuurd hoortoestel'. Traditionele AAD-algoritmen werken helaas niet snel genoeg om adequaat te reageren op een verandering in auditieve aandacht. Bovendien zijn ze gesuperviseerd en stationair in de tijd, waardoor ze zich niet kunnen aanpassen aan niet-stationariteiten in de EEG- en audiodata. Daarom is de globale doelstelling van deze thesis om nieuwe signaalverwerkingsalgoritmen te ontwikkelen voor EEG-gebaseerde AAD die snelle, nauwkeurige, niet-gesuperviseerde en tijdsadaptieve decodering van de auditieve aandacht mogelijk maken.

In het eerste deel van de thesis vergelijken we verschillende AAD-algoritmen, wat ons toelaat de hiaten in de huidige AAD-literatuur te identificeren die gedeeltelijk in deze thesis worden opgevuld. Om deze vergelijkende studie te kunnen uitvoeren, ontwikkelen we een nieuwe performantiemetriek - de minimaal verwachte omschakelduurtijd (MESD) - om AAD-algoritmen te evalueren in de context van adaptieve versterkingsregeling voor neurogestuurde hoortoestellen. Deze performantiemetriek vindt een balans in de traditionele afweging tussen de nauwkeurigheid van het AAD-algoritme en de tijd die nodig is om een AAD-beslissing te nemen. Er wordt één getal berekend dat interpreteerbaar is in de context van AAD en een gemakkelijke (statistische) vergelijking tussen AAD-algoritmen faciliteert. Met behulp van de MESD tonen we aan dat het meest robuuste AAD-algoritme dat momenteel beschikbaar is gebruikmaakt van canonieke correlatieanalyse, maar dat het decoderen van de spatiale focus van auditieve aandacht uit het EEG veelbelovender is voor snelle en nauwkeurige AAD. Bovendien stellen we vast dat resultaten van AAD-algoritmen gebaseerd op diepe neurale netwerken moeilijk te repliceren zijn op verschillende onafhankelijke AAD-datasets.

In het tweede deel gaan we in op een van de belangrijkste uitdagingen op het gebied van signaalverwerking in AAD: niet-gesuperviseerde en tijdsadaptieve algoritmen. We ontwikkelen eerst een niet-gesuperviseerde versie van de stimulusdecoder, die getraind kan worden op een grote hoeveelheid EEG- en audiodata zonder kennis van de juiste labels over de auditieve aandacht. De niet-gesuperviseerde stimulusdecoder wordt iteratief hertraind op basis van zijn eigen voorspelde labels, wat resulteert in een zelfversterkend effect dat verklaard kan worden door de iteratieve updatingsprocedure te interpreteren als een vaste-punt iteratie. Deze niet-gesuperviseerde maar subject-specifieke stimulusdecoder, startende van een willekeurige decoder, presteert beter dan een gesuperviseerde subject-onafhankelijke decoder en benadert zelfs de performantie van een gesuperviseerde subject-specifieke decoder wanneer we gebruikmaken van subject-onafhankelijke informatie. We breiden dit niet-gesuperviseerde algoritme ook uit naar een efficiënt recursief tijdsadaptief algoritme voor in het geval dat EEG- en audiodata continu binnenstromen, en laten zien dat het het potentieel heeft om in een praktische toepassing van AAD beter te presteren dan een stationaire, gesuperviseerde decoder.

In het derde deel ontwikkelen we nieuwe AAD-algoritmen die de spatiale focus van auditieve aandacht decoderen om een snellere en nauwkeurigere decodering te bekomen. Hiertoe gebruiken we zowel een lineaire gemeenschappelijke spatiale patronen (CSP)-filtering methode als de niet-lineaire uitbreiding ervan gebruikmakende van riemann-geometriegebaseerde classificatie (RGC). De CSP-methode bereikt aan een zeer hoge beslissingssnelheid een veel hogere nauwkeurigheid in vergelijking met het SR-algoritme. Verder tonen

we aan dat de CSP-methode de voorkeur wegdraagt ten opzichte van een vergelijkbare methode gebaseerd op een convolutioneel neuraal netwerk en ook toepasbaar is wanneer er verschillende richtingen van auditieve aandacht zijn, in een drieklassenprobleem met verschillende spatiale domeinen, wanneer alleen EEG-kanalen dicht bij de oren worden gebruikt en bij generalisatie naar data van een nieuwe gebruiker. Ten slotte verbetert de RGC-gebaseerde uitbreiding de nauwkeurigheid aan tragere beslissingssnelheden, vooral in het meerklassenprobleem.

In deze thesis hebben we cruciale bouwstenen ontwikkeld voor een plug-and-play, tijdsadaptief, niet-gesuperviseerd, snel en nauwkeurig AAD-algoritme dat geïntegreerd kan worden met een snel ruisonderdrukkingsalgoritme en een draagbaar, geminiaturiseerd EEG-systeem om uiteindelijk te leiden tot een neurogestuurd hoortoestel.

Contents

List of Acronyms

A	**AAD**	Auditory Attention Decoding
	ADMM	Alternating Direction Method of Multipliers
B	**BCI**	Brain-Computer Interface
C	**CCA**	Canonical Correlation Analysis
	CI	Cochlear Implant
	CNN	Convolutional Neural Network
	CSP	Common Spatial Pattern
	CV	Cross-Validation
D	**DNN**	Deep Neural Network
	DOA	Direction Of Arrival
E	**ECoG**	ElectroCorticoGraphy
	EEG	ElectroEncephaloGraphy
	ESD	Expected Switch Duration
F	**FB-CSP**	FilterBank Common Spatial Pattern
	FC	Fully Connected
	FIFO	First In, First Out
	fMRI	functional Magnetic Resonance Imaging
G	**GEVc**	Generalized EigenVector
	GEVD	Generalized EigenValue Decomposition
	GEVl	Generalized EigenValue
H	**HA**	Hearing Aid
	HRTF	Head-Related Transfer Function

I	**ICA**	Independent Component Analysis
	ITR	Information Transfer Rate
L	**LASSO**	Least Absolute Shrinkage and Selection Operator
	LCMV	Linearly Constrained Minimum Variance
	LDA	Linear Discriminant Analysis
	LMM	Linear Mixed-effects Model
	LOSO-CV	Leave-One-Segment-Out Cross-Validation
	LOSpO-CV	Leave-One-Speaker-Out Cross-Validation
	LOSuO-CV	Leave-One-Subject-Out Cross-Validation
	LS	Least Squares
	LSTM	Long Short-Term Memory
M	**MEG**	MagnetoEncephaloGraphy
	MESD	Minimal Expected Switch Duration
	MHT	Mean Hitting Time
	MISO	Multi-Input Single-Output
	MMSE	Minimum Mean Squared Error
	M-NICA	Multiplicative Non-negative Independent Component Analysis
	MWF	Multi-channel Wiener Filter
N	**NIRS**	Near-InfraRed Spectroscopy
P	**PCA**	Principal Component Analysis
R	**RGC**	Riemannian Geometry-based Classification
	RMOE	Ratio of Median Output Energies
S	**SI**	Subject-Independent
	SNR	Signal-to-Noise Ratio
	SPD	Symmetric Positive Definite
	SR	Stimulus Reconstruction
	SRT	Speech Reception Threshold
	SS	Subject-Specific
	SVM	Support Vector Machine
T	**TRF**	Temporal Response Function
	TSM	Tangent Space Mapping
V	**VAD**	Voice Activity Detection

List of Symbols

General

a	scalar
$\mathbf{a}$	vector
$\mathbf{A}$	matrix
$\mathbb{R}$	real field
$\mathcal{M}$	manifold
$P(\cdot)$	probability
$\mathcal{N}(\mu, \sigma^2)$	normal distribution with mean μ and standard deviation σ
$\mathbb{E}\{\cdot\}$	expectation
$\mathcal{O}(\cdot)$	big-O notation
$\text{sign}(\cdot)$	sign
$\lfloor \cdot \rfloor$	floor (largest integer smaller than or equal to)
$\lceil \cdot \rceil$	ceil (smallest integer larger than or equal to)
$\triangleq$	equal by definition
$\|\|\cdot\|\|_1$	ℓ_1-norm (sum of absolute values)
$\|\|\cdot\|\|_2$	ℓ_2-norm (square root of sum of squared elements)
$\|\|\cdot\|\|_\infty$	ℓ_∞-norm (maximum absolute value)
$.^{\text{T}}$	transpose
$.^{-1}$	inverse
$\text{Tr}\,(\cdot)$	trace
$\text{vech}(\cdot)$	half-vectorization
$\mathbf{I}$	identity matrix

Frequently used notations

τ	decision window length (decision time)
$p(\tau)$	accuracy-decision window length performance curve
$\mathbf{x}(t)$	multi-channel EEG signal
$\mathbf{X}$	multi-channel EEG segment

C	number of EEG channels
L	number of time lags
$\mathbf{s}_\mathrm{a}(t)$	(attended) speech envelope
$\mathbf{d}$	stimulus decoder
$\hat{\mathbf{s}}_\mathrm{a}(t)$	reconstructed speech envelope
$\rho(\cdot, \cdot)$	Pearson correlation coefficient
$\hat{\mathbf{R}}_{xx}$	(estimated) EEG covariance/autocorrelation matrix
$\hat{\mathbf{r}}_{xs}$	(estimated) EEG-speech envelope crosscorrelation vector
$\mathbf{f}$	feature vector
f_s	sampling frequency

List of Figures

List of Tables

1 Introduction

1.1 The story of Herman on Christmas Day

On Christmas Day 2021, we were able to celebrate Christmas with family once again after we needed to skip various family dinners due to the COVID-19 pandemic. We were 15 people, and everyone was happy and relieved that we could again celebrate together. The pater familias is Herman, 85 years old and the grandfather of my partner. He seemed cheerful, especially once one of his great-granddaughters entered the stage. Sometimes, however, he appeared to be a bit off, and we needed to specifically address him to involve him in the conversations. Nevertheless, we had a wonderful day, and everyone went home satisfied.

Two days later, we visited Herman at home. He welcomed us by telling us how happy he was that we were visiting him at home, without other family members - except his wife - present. He appeared much more cheerful than on Christmas Day. Herman suffers from hearing loss. For almost ten years now, he has a hearing aid that amplifies sound. As he is describing the feeling when he could hear birds singing again, it becomes clear that hearing aid technology has a significant impact on his quality of life. After a while, he admitted that, while he was happy to see the whole family together on Christmas Day, he was feeling sad himself. He estimated that he could only follow 25% of the conversations. Herman's hearing aid did not know which person or conversation he wanted to listen to. He experienced a chaotic mix of voices that he could not disentangle. He shut himself off and felt lonely as he was unable to participate in any conversation, while he saw his family engaged in all these lively talks around him. It explains why, *every single time*, he stresses how grateful he is when we are visiting him at home without other family members.

1.2 The auditory attention decoding problem

The World Health Organization estimates that one in five people worldwide experiences some form of hearing loss, of which, like Herman, around 27% requires rehabilitation [6]. Due to population growth and aging, they expect that 2.5 billion people, or one in four, will experience hearing loss in 2050, and more than 7% of the world's population will require rehabilitation. Hearing loss has a tremendous impact on society, both on an individual and economic level. On an individual level, unaddressed hearing loss hampers the ability to communicate. For children, this can severely impact their language development, which negatively impacts their cognitive and social development. Furthermore, it increases the probability of unemployment and generally induces social isolation and loneliness, heavily impacting mental health. On an economic level, the World Health Organization estimates that the annual global cost of unaddressed hearing loss amounts up to 980 billion dollars across all sectors of society [6].

Assistive hearing devices, such as hearing aids (HA) and cochlear implants (CI), try to rehabilitate people with hearing loss by restoring communication. They effectively improve speech intelligibility and therefore significantly improve the quality of life of people suffering from hearing loss, and, moreover, in a cost-effective way [6]. While these hearing devices have improved in the past decades by including more advanced speech enhancement, directional beamforming, and noise suppression technology, current state-of-the-art hearing devices still lack a fundamental piece of knowledge in so-called 'cocktail party' scenarios, i.e., when multiple persons are talking simultaneously (Figure 1.1). In such a cocktail party scenario, normal-hearing people have the remarkable ability to focus on one specific speaker, even in very challenging acoustic scenarios (for example, in the presence of a lot of reverberation or noise) [7, 8]. While advanced speech enhancement algorithms exist to suppress acoustic background noise and enhance one speaker out of a speech mixture, they generally do not know what speaker to target. In other words, current hearing devices do not know which speaker should be treated as the attended speaker (i.e., the person the hearing device user wants to listen to) and which other speaker(s) should be treated as acoustic background noise and should thus be suppressed. We refer to this problem as the *auditory attention decoding* (AAD) problem:

Section 1.2 is partly based on the introduction of [5].

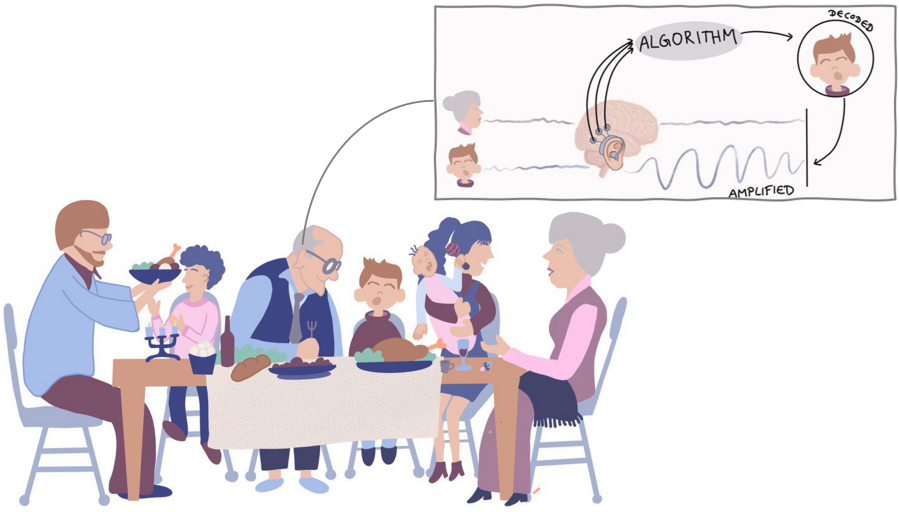


Figure 1.1: An illustration of a typical cocktail party scenario, such as a family dinner. This thesis tackles the AAD problem using brain signals to enhance the correct, attended speaker. This illustration is kindly provided by Debora Fieberg.

The auditory attention decoding problem

The auditory attention decoding (AAD) problem consists of determining to which sound source out of multiple simultaneously active sound sources a listener intends to attend to.

In this thesis, we consider these sound sources to be speech signals, although other sources, such as music [9–16], are also studied in this context. Furthermore, for the sake of an easy exposition, we assume that there are two competing speakers. Generally, the presented algorithms and technology are generalizable to more than two competing speakers, unless mentioned or discussed otherwise.

The problem of selecting the attended speaker can be addressed using simple heuristics, e.g., based on look direction, or by selecting the loudest speaker or the speaker in front of the listener. However, these simple heuristics often fail in several practical scenarios, as it would, for example, require the user to always (uncomfortably) turn towards the target. This is, moreover, not always possible, for example, when listening to a public address system or a passenger when driving a car. In those scenarios, the heuristic could select the wrong speaker, thereby enhancing speech of a speaker the listener does not want to listen to.

A potentially better and more ideal strategy would be to extract the attention-related information from the brain, where the auditory attention originates by focusing sensory and cognitive resources towards a specific stimulus, driven both by bottom-up, stimulus-driven factors and top-down, goal-directed factors [17]. Recent neuroscientific insights have confirmed that this is possible, for example, by showing that certain characteristics of the speech signal of the attended speaker are (better) encoded in the brain (see Sections 1.5 and 1.6). Following these groundbreaking advances in auditory neuroscience, research towards *AAD from the brain* has gained traction in the neural engineering community in the past ten years, which could lead to a new assistive solution for the hearing impaired: *a neuro-steered hearing device* (see Section 1.7). Decoding the auditory attention from the brain is the central problem that is tackled in this thesis (Figure 1.1).

1.3 The brain and neurorecording techniques

1.3.1 The brain

The human brain largely consists of three parts: the brainstem, the cerebellum, and the cerebral cortex or cerebrum, which is divided into four lobes: the frontal, parietal, occipital, and temporal lobe (Figure 1.2). Each of these lobes is typically connected to different functionalities. The frontal lobe is related to cognitive functions such as reasoning and control of movements. The parietal lobe integrates sensory information, such as touch, taste, etc. The occipital lobe is mainly dedicated to visual information processing, while the temporal lobe is dedicated to auditory information processing as it contains the primary auditory cortex.

On a microscopic level, the most important cells of the cerebral cortex are neurons, the basic information processing units of our brain. A neuron consists of three parts: the soma or cell body, a long axon, and the dendrites (Figure 1.2). Each neuron is via its dendrites connected to the axon terminals of several other neurons via synapses, and communicates with these other neurons through electrical signals. If the dendrites of a neuron receive enough inputs (via neurotransmitters) from neighboring neurons, an electrical impulse, called an 'action potential', is triggered and travels via the neuron's axon to the axon's terminal where it can excite other neurons. This action potential traveling down the neuron is also referred to as the neuron 'firing'. Typically, the axon's body is covered by a myelin sheath, which acts as an insulator and enables the very high speed with which the action potentials travel down the axon. While the neuron is a quite basic information processing unit that communicates via a

binary all-or-nothing signal, the strength of the human brain comes from the connectivity of *billions* of neurons. It is estimated that the adult male human brain contains on average 86 billion or 86 000 000 000 neurons, of which 19% is located in the cerebral cortex [18]. Given that each neuron can be connected to thousands of other neurons, there is an astounding number of approximately 0.15 quadrillion (10^{15}) connections in the cerebral cortex [19]. These connections are, however, not fixed. One of the most crucial features of the human brain is its plasticity. The ability of neurons to adapt and make new connections, for example, via dendritic growth, form the basis of this neuroplasticity.

The computational power of the human brain. Because of its billions of neurons and quadrillions of connections, the human brain is a very powerful biological computer. Although hard to compare and quantify, it is estimated that the human brain operates at around one exaFLOPS (10^{18} floating-point operations per second) [20]. As the result of the exponential increase in computational power of (super)computers, as observed by Moore's law, in theory, this computational power can be matched by the most powerful currently available supercomputers, such as the Fugaku supercomputer [21]. Cutting many corners, one could thus say that one human (brain) is as powerful as the most powerful current supercomputer (supercomputers are, evidently, scarce). Note, however, that the fact that a supercomputer matches the human brain in FLOPS does not mean that it can perform the same tasks as a human. Apart from the hardware, this additionally requires the correct software, which is very challenging to create (for example, as it would require a tremendous amount of training data/information in real-world conditions). Moreover, the human brain is highly adaptive and flexible because of the neuroplasticity, and is a very volume- and power-efficient biological computer compared to supercomputers, making it even more remarkable.

1.3.2 Neurorecording techniques

Electroencephalography

As explained in Section 1.3.1, the neurons in the brain are continuously firing, generating electrical action potentials. Using electroencephalography (EEG) sensors on the scalp, the electrical activity generated within the brain can be measured as a voltage between two electrodes (Figure 1.2). The difference between two electrode voltages is called an 'EEG channel'. However, as these

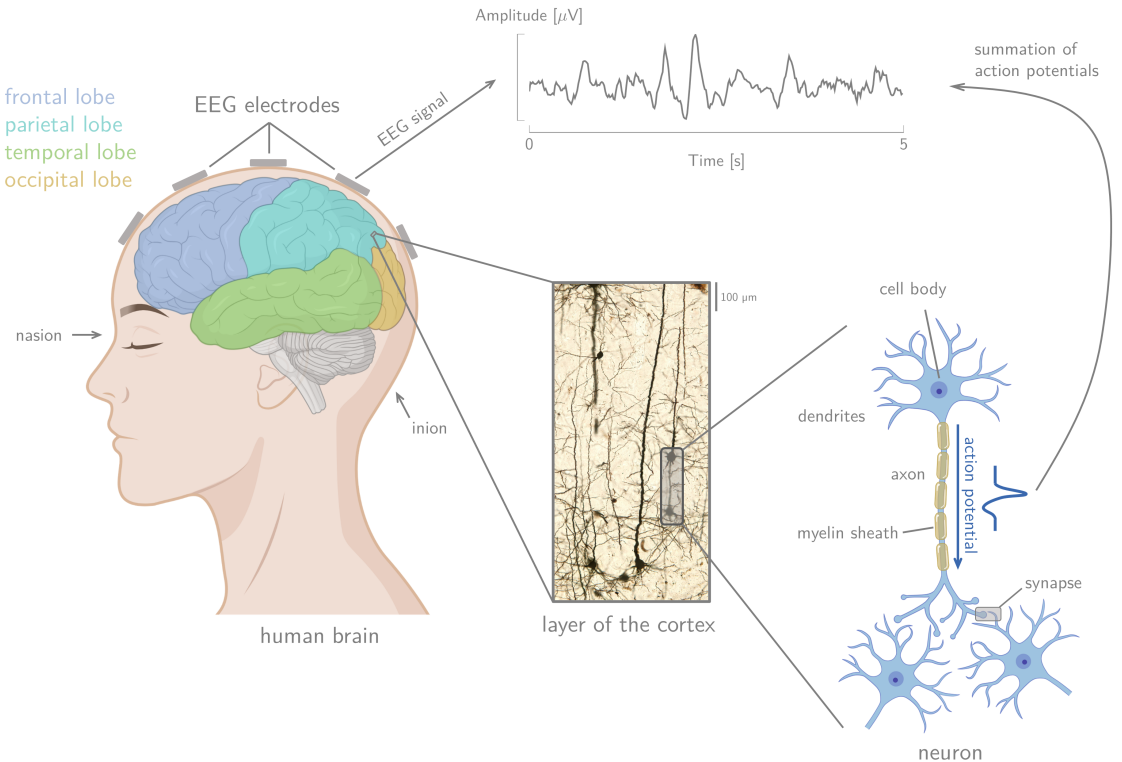


Figure 1.2: EEG measures the summation of action potentials of neurons that are synchronously firing. The brain is divided into four lobes, containing neurons as the basic information processing units. A neuron consists of a cell body, a long axon, and dendrites. Other neurons can be triggered by neurotransmitters released at the synapses due to an action potential traveling to the axon terminal. Based on [22] and created with [23].

Band	Range	Associated with
δ	≤ 4 Hz	deep sleep
θ	4-8 Hz	meditative concentration, drowsiness, mental calculation
α	8-12 Hz	mental effort, inattention, daydreaming, closed eyes
β	12-30 Hz	attention processing [25], active attention and thinking, motor activities
γ	≥ 30 Hz	perception, motor functions

Table 1.1: The EEG is traditionally described as consisting of different frequency bands, each associated with a specific frequency range and different functions. A non-exhaustive list of associated functions is given (based on [26]).

sensors are non-invasively placed on the scalp, measuring action potentials from individually firing neurons is impossible. The EEG is instead the summation of thousands of spatially aligned neurons that are synchronously firing. Moreover, primarily the activity of pyramidal neurons near the scalp that are oriented perpendicularly to the brain's surface is picked up [24].

EEG frequency bands EEG signals are traditionally described as consisting of five distinct frequency bands, in increasing order of frequency: the δ-, θ-, α-, β-, and γ-band. Table 1.1 gives an overview of these different frequency bands, including their ranges and characteristics.

Measurement system EEG electrodes are usually placed on the scalp according to the internationally standardized 10-20 system. This standardization refers to the distances between the electrodes across the scalp between the nasion (nose bridge) and inion (bump on the back of the head). Each electrode is denoted with a letter, corresponding to the different cortical lobes (F = frontal, C = central, T = temporal, P = parietal, O = occipital), and a number, indicating the hemisphere on which the electrode lies (odd = left, even = right). Electrodes positioned on the midline of the scalp are indicated with a 'z' [24].

Many different EEG systems exist, from high-density layouts with more than 64 up to 256 (wired) electrodes positioned at proportional distances within the 10-20 system to more wearable and miniaturized EEG setups with a lower number of (wireless) electrodes. Furthermore, there exist both wet-electrode systems, which require electrode gel and thus hamper applicability, and dry-electrode systems, which are easier to apply and thus more suited for chronic neurorecording [27]. The gel used in wet systems enhances conductivity between

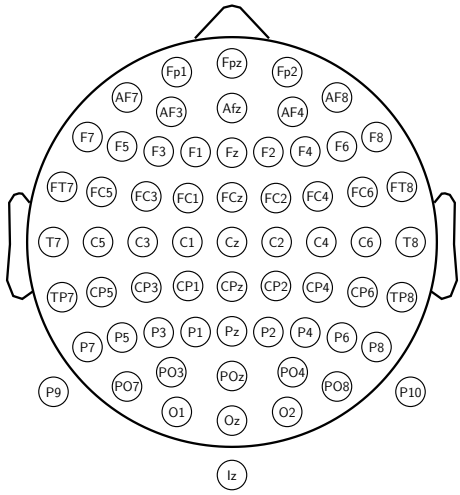


Figure 1.3: In this thesis, we mostly use the 64-channel BioSemi ActiveTwo EEG recording system, with its layout displayed here.

the skin and electrodes, reducing the impedance (which can be seen as a signal quality indicator) [24, 26]. In this thesis, we mostly use a state-of-the-art 64-channel BioSemi ActiveTwo EEG recording system[1] (Figure 1.3). Other, more wearable systems are, for example, the 24-channel SMARTING system of mBrainTrain[2], and miniaturized EEG devices, such as in-ear [28, 29], around-the-ear [30], or flex-printed forehead [31] EEG systems.

There exist many different EEG *montages*, which define to which electrode pairs EEG channels correspond (remember that EEG channels represent voltage differences between electrodes). In a bipolar montage, neighboring electrodes are used to define the different channels, while in a referential montage, a fixed reference electrode is used with respect to all other electrodes [24]. Common choices of the reference electrode are the Cz electrode, the mastoid electrodes, or a common average reference, where the reference signal equals the average across all electrodes.

Disadvantages The two main disadvantages of EEG as a neurorecording technique are the following:

[1] www.biosemi.com
[2] www.mbraintrain.com

1. **Low spatial resolution:** As the EEG sensors only measure the summed activity of thousands of neurons, which is, moreover, smeared out due to its propagation through the cerebrospinal fluid and scalp, EEG has a low spatial resolution. Furthermore, only activity close to the scalp of well-oriented neurons is measured, while deeper sources are hardly picked up. The spatial resolution is generally estimated in the order of 10 mm [26].

2. **Low signal-to-noise ratio:** EEG is known for its notoriously low signal-to-noise ratio (SNR). The neural signals of interest are often buried under irrelevant brain activity or other interferences. This can happen because the neural signals of interest are non-favorably located or oriented (see the previous paragraph) but also because of artifacts, which can be induced physiologically or non-physiologically. Examples of physiological artifacts are eye movements (e.g., blinks, lateral movements), muscle activity (e.g., jaw clenching, chewing) and movements, and sweat. Non-physiological artifacts result, e.g., from cable movements, electrode pop, or nearby electrical devices (causing, for example, powerline interference). Several signal processing algorithms exist to remove these artifacts after recording [32], e.g., based on a multi-channel Wiener filter (MWF) [33], independent component analysis (ICA) [34], or canonical correlation analysis (CCA) [35].

Advantages Despite these disadvantages, EEG is the most popular neurorecording technique [26]. It has been used in many brain-computer interface (BCI) applications, such as neuroprosthetic control and text-input systems for rehabilitation of paralyzed people, gaming, etc. [36–38]. This popularity is mainly due to the following four advantages:

1. **High temporal resolution:** Compared to other neurorecording techniques, EEG has a very high temporal resolution, generally estimated in the order of 0.01 s [26, 39]. This is a crucial feature in several BCIs, where real-time processing is essential.

2. **Non-invasiveness:** EEG sensors are attached to the scalp, with a relatively fast and easy setup. This is a crucial ingredient for the widespread usage of EEG, for example, in combination with hearing devices.

3. **Wearability:** Several wearable, concealable EEG systems have been developed in the past decades, showing its (potential) portability. This is again a crucial ingredient for the widespread usage of EEG in chronic neurorecording applications, for example, during daily-life activities.

4. **Relatively low cost:** Although Hans Berger, the German psychiatrist that invented the EEG in 1924, in the last footnote of his first paper on EEG complained about the high cost of the required instrumentation [40], the technological advancements of the 20$^{\text{th}}$ and 21$^{\text{th}}$ century have made EEG a relatively cheap technology, especially compared to other neurorecording technologies [26].

Other neurorecording techniques

Many other neurorecording techniques exist, which all serve their purposes. An overview of the different properties of the discussed neurorecording techniques can be found in Table 1.2. Techniques such as near-infrared spectroscopy (NIRS) (i.e., optically measuring fluctuations in cerebral oxygenation) or functional magnetic resonance imaging (fMRI) (i.e., measuring changes in cerebral blood flow) provide an indirect measure of neural activity and have a low temporal resolution [26], making them less suited for (real-time) AAD. Magnetoencephalography (MEG) and electrocorticography (ECoG) have, however, already been used in AAD studies (with MEG: e.g., [41–44], with ECoG: e.g., [45–50]). MEG non-invasively measures the magnetic activity generated by firing neurons. As opposed to EEG, MEG provides a better spatial resolution in the order of 5 mm and a higher SNR, which is mainly because magnetic fields are less distorted by the scalp [26]. However, its biggest disadvantage is that it is very costly and not wearable, as it requires a bulky setup in a magnetically shielded room[3]. ECoG is an invasive technique that measures the electrical activity of firing neurons on the surface of the cortex, below the scalp. As such, it can be seen as an invasive version of EEG. As a result, the temporal and spatial resolution, and SNR are better than in EEG. Potentially, this can also be combined with stereotactically inserted depth electrodes to record brain activity at deeper sites in the cortex. However, the invasive nature of this technique makes it less suited for widespread employment in daily-life usage, for example, in combination with hearing devices.

The advantages of EEG and disadvantages of other neurorecording techniques motivate why we choose EEG as a neurorecording technique for AAD.

[3] Some initial steps towards wearable MEG have recently been taken, however, still with very limited mobile applicability [51].

Method	Activity measured	(In)direct measurement	Temporal resolution	Spatial resolution	Invasive?	Wearable?	Cost
EEG	electrical	direct	$\sim 0.01\,\mathrm{s}$	$\sim 10\,\mathrm{mm}$	no	yes	low
MEG	magnetic	direct	$\sim 0.01\,\mathrm{s}$	$\sim 5\,\mathrm{mm}$	no	no	high
ECoG	electrical	direct	$\sim 0.001\,\mathrm{s}$	$\sim 1\,\mathrm{mm}$	yes	yes	high
NIRS	metabolic	indirect	$\sim 1\,\mathrm{s}$	$\sim 5\,\mathrm{mm}$	no	yes	low
fMRI	metabolic	indirect	$\sim 1\,\mathrm{s}$	$\sim 1\,\mathrm{mm}$	no	no	high

Table 1.2: The properties of the different neurorecording techniques, based on [26, 39].

1.4 The auditory system

The human auditory system, which transforms vibrations of air molecules (sound) to electrical signals, is divided into four hierarchical parts, each providing a more complex processing of the original sound[4] (Figure 1.4). The outer peripheral part comprises the outer ear, middle ear, and inner ear. The fourth part is the auditory pathway in the brain. The outer ear consists of the pinna, which is important in localizing sounds [53], and the ear canal. As a sound wave travels through the ear canal, it causes the tympanic membrane, which lies on the border of the outer and middle ear, to vibrate. A chain of three small bones in the middle ear, called the ossicles, amplify these vibrations and transmit them to the inner ear via the oval window. As the most important component of the inner ear, the spiral-shaped cochlea, contains fluids, the vibrations are transformed into fluctuations in the cochlear fluids. This causes the basilar membrane in the cochlea to move in a frequency-dependent manner. One could compare it, therefore, to a filterbank. As a result, the (inner) hair cells within the organ of Corti are displaced and release neurotransmitters as a response to these mechanical movements. These neurotransmitters then excite the neurons in the nerve fibers of the auditory nerve (similarly to Section 1.3.1). From thereon, the sound travels via neurons firing through the auditory pathway in the brain, from the auditory nerve, via the brainstem, to the primary auditory cortex, where higher-order processing of the sound takes place.

1.4.1 Listening in a cocktail party scenario

In a cocktail party scenario, such as in Figure 1.1, normal-hearing people can focus on one specific speaker and ignore all other sound sources [7, 17]. The human auditory system is essentially capable of performing auditory scene analysis in complex acoustic scenarios [55]. To this end, both monaural (i.e., which could be exploited with one ear only, such as pitch and intensity

[4] The following description of the auditory system is largely based on [52].

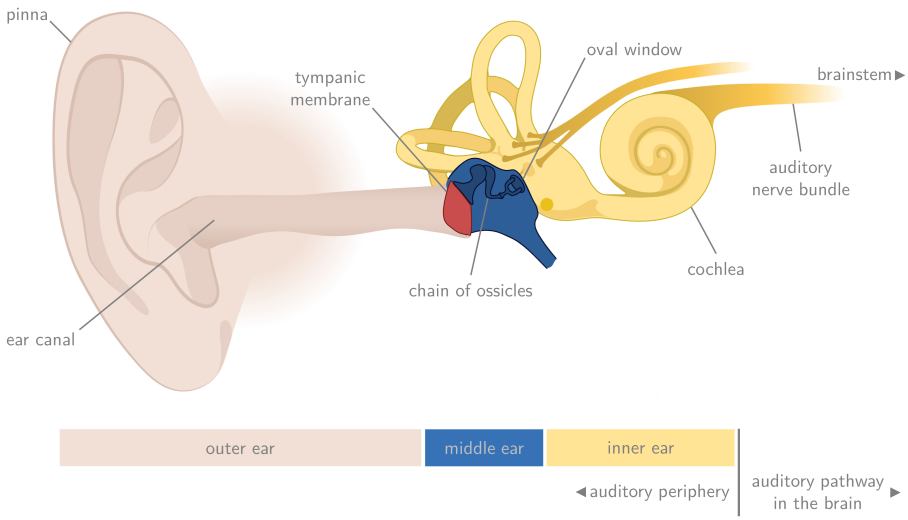


Figure 1.4: The anatomy of the human auditory system, which consists of the outer, middle, and inner ear (peripheral part), and the auditory pathway in the brain. Created with [54].

fluctuations) and binaural (i.e., which can only be exploited using both ears, such as interaural level and time differences - 'spatial' hearing) cues of the to-be-attended sound source can be used [56, 57]. The exploitation of binaural cues and auditory spatial information happens up the auditory pathway in the cortex, where the information coming from both ears is integrated to perform sound localization [53]. These binaural cues become more important to focus on a specific speaker in complex acoustic scenarios. This is called 'spatial release from masking' (you can try this out yourself by plugging one ear in a cocktail party scenario). On the other hand, this spatial release from masking becomes less important when speakers can be easily separated, for example, based on monaural cues [53, 56–62].

1.4.2 Hearing loss

In its essence, hearing loss means that there is damage somewhere along the auditory pathway. There are several types of hearing loss, e.g., sensorineural (damage in the inner ear, for example, in the hair cells), conductive (damage in the outer or middle ear), central (in the auditory pathway of the brain), or a mix. There is a high correlation between sensorineural hearing loss and

age [6,63]. While the damage mainly occurs in the outer peripheral part, changes can also occur higher up the auditory pathway in the brain. As explained in Section 1.3.1, our brain has the remarkable ability to adapt itself. Because of this neuroplasticity, the brain can even, to some extent, adapt to the hearing loss and a potential hearing device [64]. Furthermore, it explains why it is crucial for hearing device users to use their devices at all times to maximally leverage this neuroplasticity (Herman, however, does not always use his HA, regardless of how many times we explain him why it is important).

One of the most important rehabilitation strategies for hearing loss is hearing technology such as HAs and CIs. The most frequently used technology is a HA, which amplifies the recorded sounds and delivers them through the ear canal to the tympanic membrane. It is a non-invasive, safe, and very (cost-)effective rehabilitation strategy [6]. A CI can be used when the conventional HA does not offer a solution. CIs are surgically implanted and bypass the peripheral part of the auditory system to directly stimulate the auditory nerve electrically [6]. While these hearing devices significantly improve the speech intelligibility in simpler acoustic scenarios, they still underperform in cocktail party scenarios [65].

1.5 Neural tracking of the speech signal

While event-related potential studies based on auditory evoked potentials have given insight into selective auditory attention and auditory scene analysis [66], they are limited by their usage of simple, repeated stimuli that do not reflect the complex auditory stimuli that occur in the wild. Therefore, studies using single-trial, natural, continuous stimuli are preferred. These studies have extensively shown that several features of a speech stimulus are encoded in the human auditory cortex. For example, Aiken and Picton [67] have shown for one of the first times that the auditory cortex tracks the envelope of the presented speech stimulus (Figure 1.5), especially in the δ- and θ-band [47,68–70] (although, more recently, this is also shown for the γ-band [70,71]). Moreover, Mesgarani and Chang [46] have shown that also in a cocktail party scenario with two competing speakers, a speech spectrogram can be reconstructed from the cortical responses that reflects the spectro-temporal features of the attended speech signal.

Most studies, however, focus on the speech *envelope*, which represents the slowly-varying temporal modulations of the speech signal, as a crucial feature that is encoded in the neural signals. The speech envelope is one of the most important cues for speech understanding. Vocoded (synthesized) speech using only the speech envelope, for example, is still highly intelligible (without

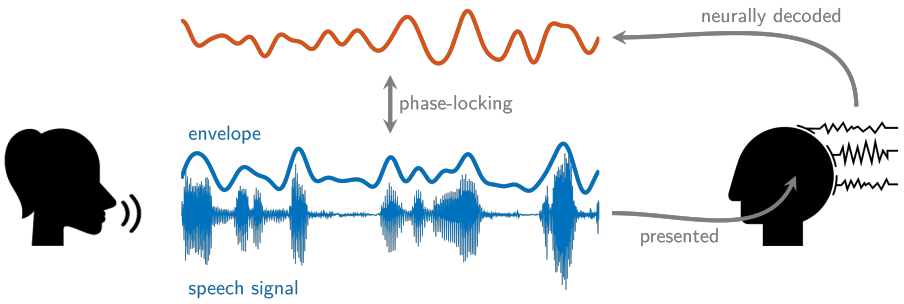


Figure 1.5: The auditory cortex tracks the envelope of a presented (and attended) speech signal, which manifests itself, for example, in a phase-locking of the (decoded) neural signals (for example, recorded EEG signals) with the speech envelope.

background noise) [72]. Ding and Simon [41] showed that in a competing speaker setup with two concurrent speech streams, the neural signals phase-lock (i.e., synchronize) both with the envelope of the attended and unattended speech signal. The former can, however, be easier decoded than the latter. Moreover, they demonstrate that when changing the intensity of the competing speakers, the neural signals only adapt to the intensity changes of the attended speaker and not of the unattended speaker. Furthermore, Ding and Simon [73], and Power et al. [73] show that the selective auditory attention mainly modulates the longer-latency responses around 100-250 ms in the auditory cortex, rather than the short-latency responses around 50 ms. Consistent with these findings, Zion Golumbic et al. [47] claim that the neural tracking of both the attended and unattended speech envelopes is present in the lower-level auditory cortices, but only of the attended speech envelope in the higher-order regions. As concluded by Simon [42], concurrent speech signals are individually encoded in the auditory cortex, even when they are spectrally overlapping and not resolvable in the auditory periphery. This stronger neural tracking of the attended versus unattended speech envelope has been exploited for the first time in EEG-based AAD by O'Sullivan et al. [74], who used the so-called 'stimulus reconstruction' (SR) paradigm, and by Horton et al. [75], who generated features by, for example, directly cross-correlating the speech envelopes with the EEG channels, and classified them to make a decision about the auditory attention. The SR paradigm is still a commonly used AAD paradigm (see Chapter 3). Lastly, the neural tracking of the speech envelope can also be used to, for example, objectively predict speech intelligibility, which could lead to automatic fitting of HAs and CIs [76–80].

More recently, neural tracking of other features of the speech signal than the

spectrogram or envelope has been proven, for example, based on linguistic features (e.g., phonetic or semantic features) [81–86] or other acoustic features, such as the fundamental frequency of the voice [87, 88].

1.6 Neural tracking of the attended spatial location

It is known that neural signals are modulated by visual and somatosensory spatial attention. For example, the EEG/MEG-power in the α-band in the (parieto-)occipital areas changes depending on the direction of attentional focus, in visual [89–91] and somatosensory [91, 92] attention. Therefore, it is hypothesized that similar changes of the neural signals exist depending on the direction of spatial attention to an auditory stimulus, resulting from selectively firing neurons.

Such neural changes and tracking of the attended spatial location is confirmed in several studies. Wolbers et al. [93] showed, based on fMRI recordings, that auditory motion (in this case, non-speech) is encoded in the occipito-temporal regions in people with a visual impairment. Also using fMRI, McLaughlin et al. [94] showed that there is mainly a contralateral (i.e., at the opposite side of the stimulus location) activation in the auditory cortex. In EEG/MEG recordings, it has been repeatedly shown that a similar α-band power lateralization is present to the direction of auditory spatial attention to a single sound source [95–98]. More specifically, this α-power lateralization corresponds to an ipsilateral (i.e., at the same side of the stimulus location) enhancement and contralateral suppression of α-power [95, 96, 98–100]. Furthermore, this neuronal selectivity to the spatial auditory direction is mainly found in the auditory cortex [94, 96, 97], but also in other parieto-occipital regions [98]. Deng et al. [98] hypothesized that this α-power lateralization in the parieto-occipital regions is the effect of the same cognitive processing as for other sensory modalities. Lastly, it has been shown that these α-power lateralization patterns to the spatial location of the attended sound source are also present in a competing-stimuli scenario with more than one sound source [99–101].

Using this information, Bednar et al. [102] could decode the position of a non-moving, static sound source from EEG. This was then extended to the decoding of the trajectory of a continuously moving sound source in [103], mainly based on the phase of the δ-band EEG signals in the auditory cortex and the α-band power in the parieto-occipital areas. Finally, this was extended to continuously moving *competing* (speech) stimuli in [104], showing that the attended sound source trajectory could be decoded from the EEG.

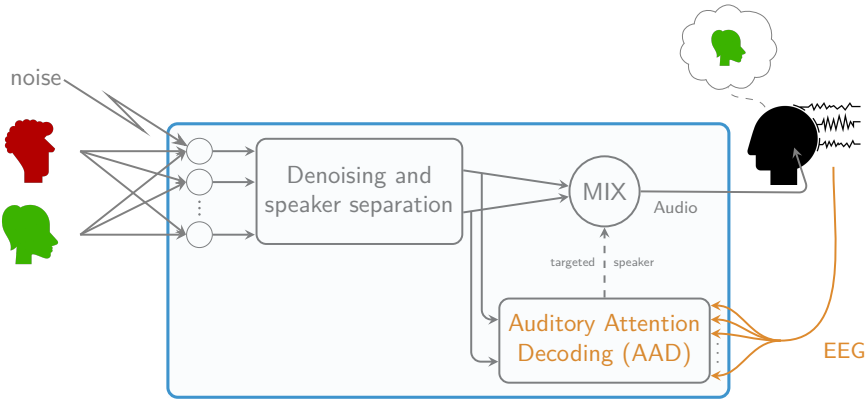


Figure 1.6: A conceptual overview of a neuro-steered hearing device with two competing speakers, consisting of a speaker separation block, AAD block, and the EEG modality [5]. The main focus of this thesis is on the AAD block. This overview assumes that the auditory attention decoding is performed separately from the speech enhancement, which is not necessarily the case (see Section 1.7.2).

1.7 The concept of neuro-steered hearing devices

Incorporating AAD in an assistive hearing device such as a HA or CI could lead to a so-called 'neuro-steered hearing device' that can assist the user in a cocktail party scenario. However, EEG-based AAD could not only be useful in the context of assistive hearing devices but also in, for example, consumer earphones and other hearables such as noise-canceling headphones and BCIs [105].

While Figure 1.1 already conceptually conveys the idea of a neuro-steered hearing device, Figure 1.6 shows a more detailed conceptual overview. The main ingredients are a speaker separation and enhancement block, an AAD block determining the targeted speaker based on the EEG, and a mix block that mixes the outputs of the speaker separation using the information about the auditory attention to enhance the attended speaker and suppress the other speakers. In this mixing procedure, other auditory objects and speakers should not be fully suppressed, as, for example, it needs to be still possible for the user to switch to another speaker. The main focus of this thesis is on the development of algorithms for the AAD block.

The following sections describe the different building blocks (AAD algorithms (Section 1.7.1), speech separation and enhancement (Section 1.7.2), EEG

Section 1.7 is largely based on Section IV of [5].

miniaturization and wearability (Section 1.7.3)) of a neuro-steered hearing device in the context of AAD, as well as the validation of AAD algorithms in realistic listening scenarios and on hearing-impaired listeners (Section 1.7.4), which is crucial for the success of neuro-steered hearing devices.

1.7.1 AAD algorithms

The most important building block of a neuro-steered hearing device is the AAD algorithm, which extracts information about the attended speaker from the EEG to inform the hearing device about which speaker to enhance and which other speaker(s) to suppress[5]. There exist many different AAD algorithms, on which we give an extensive review in Chapter 3. A common paradigm for AAD is the SR algorithm, which capitalizes on the stronger neural tracking of the attended speaker (Section 1.5) by reconstructing the attended speech envelope from the EEG using a neural decoder and identifying the attended speaker through correlating the decoded envelope with the speech envelopes of the individual speakers.

1.7.2 Speaker separation and enhancement

Several AAD algorithms, for example, those based on the SR paradigm, a priori require the individual speech signals of the competing speakers. Even for those AAD algorithms that do not require the individual speech signals for AAD itself, a speaker separation and enhancement block is eventually required to extract the attended speech signal for mixing and presentation to the hearing device user. While several advanced and well-performing signal processing algorithms exist for speech separation and enhancement [107–109], we here focus on the combination with AAD.

In the context of the SR algorithm, AAD has been performed using the unprocessed microphone signals as reference signals. Van Eyndhoven et al. [110] showed that there is a significantly lower performance without a speech or envelope separation step. This is confirmed by Aroudi et al. [111], who, however, also indicated that removing reverberation and other (non-interfering speech) background noise is less critical than separating the interfering speakers. Both studies show that an a priori speech or envelope demixing step is crucial. Therefore, many different speaker separation and enhancement algorithms have been proposed in combination with the SR algorithm, both based on traditional

[5]Such a separate AAD and speech enhancement is not strictly necessary, as shown in [106] (see Section 1.7.2).

source separation and beamforming methods [110, 112–115], and nonlinear deep neural networks (DNNs) [49, 50, 106, 113, 116, 117].

Van Eyndhoven et al. [110] proposed a first approach based on the multiplicative non-negative ICA (M-NICA) and MWF algorithm. M-NICA is used to demix the speech envelopes (exploiting the non-negativity of an envelope), which are then used to perform AAD. Using voice activity detection (VAD) on the identified attended speech signal by the AAD algorithm, an MWF is trained to perform the final speech enhancement step. The authors showed that this works well with substantial noise levels, is applicable in real-time, and requires only a low amount of computational resources, which is a crucial feature for practical hearing devices. Furthermore, Aroudi and Doclo [112] proposed to use a linearly constrained minimum variance (LCMV) beamformer to generate the reference signals. The output of the AAD algorithm is then used to steer another binaural LCMV beamformer to filter out the attended speaker. This approach again resulted in a significant improvement in speech quality, even with substantial noise levels.

Following the trend of DNNs in speech separation, O'Sullivan et al. [49] proposed to use a single-microphone long short-term memory (LSTM) DNN-based speech separation method to extract the different individual speech signals from a mixture. While this approach showed a significant improvement in speech quality (however, without background noise), it is limited to a closed set of fixed speakers. Han et al. [50] alleviated this limitation using an online deep attractor network, allowing generalization to new target speakers. Das et al. [113] proposed a similar system as in [110] using deep clustering. They found the best performance in challenging acoustic scenarios using a deep clustering-based speech separation for voice activity detection to inform an MWF for the speech demixing for AAD and the final speech enhancement step. Finally, Borgström et al. [116] proposed a convolutional neural network (CNN) similar to [109].

All previous approaches tackle the speaker separation and AAD step as separate problems, where the former needs to inform the latter. Another original approach, however, is to use the reconstructed speech envelope from the EEG to directly inform the speaker separation and enhancement, performing an all-in-one AAD and speaker enhancement operation. Ceolini et al. [117] employed this strategy using a CNN-based speech separation algorithm. Hosseini et al. [106] went one step further by directly feeding the EEG jointly with the speech mixture in a DNN for end-to-end brain-informed speech enhancement. Lastly, Pu et al. [114, 115] optimized a beamformer such that the beamformer output signal is maximally correlated with the neural decoder output.

1.7.3 EEG miniaturization and wearability effects

Most AAD data are recorded using heavy, bulky, and wet (i.e., gel-based) EEG recording systems. A practical neuro-steered hearing device, however, would require wearable, concealable EEG recording systems with (potentially dry) miniaturized EEG sensors, such as in-ear [28, 29] or around-the-ear [30] systems (see Section 1.3.2). These EEG systems only provide a limited amount of EEG channels, recording brain activity within a small area. It is, therefore, crucial to investigate the impact of these wearable, miniaturized EEG systems on the AAD performance. This can be done in a recording system-based (i.e., starting from an existing recording system or setup) or data-driven (i.e., finding an optimal, reduced set of electrodes or channels for AAD from a high-density EEG system) manner. The following studies all used the SR paradigm.

The recording system-based approach was taken by Mirkovic et al. [118], who showed that, although a significant decrease in AAD performance is obtained using an around-the-ear cEEGrid system [30] w.r.t a full-cap 64-channel system, still acceptable performances using cEEGrid are achievable. However, similar AAD performances to a full-cap 64-channel system could be obtained using a wearable 24-channel SMARTING system [3] and wearable, dry 18-channel DSI-24 system [119].

Using a data-driven channel reduction approach, a reduction from 96 to 25 EEG channels could be realized by Mirkovic et al. [120] without a loss in AAD performance. They even achieved still acceptable performances with less than ten EEG channels. Mundanad Narayanan and Bertrand [121] found a similar result, reducing the number of channels from 64 to ten without a loss in performance. Moreover, Mundanad Narayanan et al. [121, 122] demonstrated that similar AAD performances to the standard long-distance montages could be obtained with EEG measured with multiple optimally positioned electrode pairs with inter-electrode distances down to 3 cm. This is an important result for EEG miniaturization, where per EEG sensor, only a few electrodes within a confined area on the scalp are available. Lastly, all three studies reported that the selected electrodes using data-driven selection methods for SR are positioned above the temporal lobe, where the auditory cortex is located. This is consistent with the observation that the neural tracking of the speech envelope(s) is mainly present in the auditory cortex (Section 1.5).

1.7.4 Validation in realistic listening scenarios and on hearing-impaired listeners

AAD algorithms are often validated in very controlled scenarios, e.g., only using two competing speakers, without background noise and reverberation, using spatially well-separated speakers, or without switches in attention. Validating the performance of AAD algorithms in more realistic listening scenarios is, of course, paramount for the practical application of neuro-steered hearing devices in the wild. A few studies have performed AAD in more challenging scenarios. For example, Schäfer et al. [123] showed that the SR algorithm still worked with only a limited performance loss when using four competing speakers. Furthermore, Das et al. [4] showed for the SR algorithm that the AAD performance even improves with moderate background babble noise compared to a scenario without noise. It is hypothesized that this results from enhanced neural tracking due to the more challenging situation, requiring a higher effort by the listener. As expected, the AAD performance starts to degrade for further increasing noise levels. The authors also showed that the AAD performance improves with an increased speaker separation, as expected from spatial release from masking, while acceptable performance is obtained even with closely positioned competing speakers. On top of that, Fuglsang et al. [2] and Aroudi et al. [111] tested different reverberation settings, and Aroudi et al. [111] established that the SR algorithm could be trained on data recorded in different acoustic scenarios than the test scenario. Straetmans et al. [124] also showed that AAD is still possible in a natural environment, during free leisure walking.

Furthermore, some initial analyses on the effects of switches in auditory attention on the performance of AAD algorithms (for example, on the detection delay) have been performed [44, 49, 125–127]. Lastly, it has been extensively shown that the neural tracking of the attended envelope is not only present in normal-hearing people but also in hearing-impaired listeners [80, 128, 129], and that the SR algorithm still works with CIs and HAs [130, 131]. This is, of course, paramount for the successful employment of (SR-based) AAD in practical neuro-steered hearing devices.

1.8 AAD datasets

In this section, we give an overview of the AAD datasets that are used throughout this thesis. All datasets were available before the start of this PhD project, and no extra data collection was performed in the context of this thesis. Table 1.3 gives a summary of the most important characteristics of these datasets. In all datasets, an AAD experiment using natural, continuous speech

stimuli is performed, where the subjects are asked to attend to one out of two simultaneously talking speakers. During the experiment, the EEG data of the participants are recorded.

1.8.1 Dataset A - Biesmans et al. (2017) [1]

Dataset A is recorded with the aim to compare different envelope extraction methods for the SR algorithm and to analyze the effect of head-related filtering and the ear-specific decoding bias on AAD, and is extensively described in [1,134]. This dataset is freely available online [132] and has been used in several AAD-related studies (e.g., [1, 110, 121, 134–140]). 16 normal-hearing and natively Dutch-speaking subjects (eight male, eight female gender) participated in an AAD experiment, listening to one out of two competing speakers. The speech stimuli consisted of four Dutch short stories narrated by male speakers. These short stories were divided into two simultaneous tracks, consisting of four parts of approximately 6 min. After each presented part, the participants needed to answer a few multiple-choice questions, and after all four parts, there was an extended break. After the break, the participants switched attention to the other track. This resulted already in 48 min of AAD data. In the last part of the experiment, three extra repetitions of the first two minutes of each part, with attention only to the first track, were presented. This resulted in another 24 min of data. This brings the total amount of EEG and audio data per subject to 72 min.

The EEG data of the participants were recorded using a 64-channel BioSemi ActiveTwo system (Figure 1.3). Half of the time, the speech stimuli were presented dichotically (i.e., 'dry', presenting each speaker to a different ear), the other half they were processed using head-related transfer functions (HRTFs) in an anechoic room (i.e., without reverberation), simulating a realistic listening scenario with both competing speakers located at $\pm 90°$ left and right of the listener. In the HRTF-filtered conditions, the subjects thus perceived both stimuli in each ear. The attention was, furthermore, balanced to each ear (i.e., half of the data are left attended, half right attended). No other background noise was added to the mix of speech stimuli.

1.8.2 Dataset B - Fuglsang et al. (2017) [2]

Dataset B is recorded with the aim to assess AAD performance in more complex acoustic scenarios, in this case, with different amounts of reverberation. It is extensively described in [2], has been used in, e.g., [2, 138–142], and is freely available online [133]. 18 normal-hearing subjects participated in an

Label	Original references	Nb. of subjects	Amount of data per subject	EEG system	Sex of speakers	Location speakers	Acoustic room conditions	Originally designed for	Chapters
A	Biesmans et al. (2017) [1], available online [132]	16	72 min	64-channel BioSemi	male-male	±90°	dichotic and HRTF-filtered in anechoic room	effect of envelope extraction method, HRTF-filtering, and ear-specific decoding bias on AAD	Chapters 2 to 7
B	Fuglsang et al. (2017) [2], available online [133]	18	50 min	64-channel BioSemi	male-female	±60°	HRTF-filtered in anechoic, mildly, and highly reverberant room	effect of reverberation on AAD	Chapters 3 to 5
C	Zink et al. (2017) [3]	2	192 min	24-channel mobile SMARTING	male-male	NA	HRTF-filtered in anechoic room	in-home, longitudinal, neurofeedback experiments	Chapter 5
D	Das et al. (2018) [4]	18	138 min	64-channel BioSemi	male-male	±90°, ±5°, −90/− 30°, +30/+ 90°	HRTF-filtered with different levels of babble noise from nine locations	effect of speaker locations and noise on AAD	Chapters 6 and 7

Table 1.3: An overview of the datasets used in this thesis and their most important characteristics.

AAD experiment. The speech stimuli consisted of fictional stories narrated by a competing male and female Danish speaker. The competing stories were presented in 50 s trials, with multiple-choice questions after each trial. After ten trials, there was an extended break. In total, there are 60 min of EEG and audio data per subject.

Similar to Dataset A, the EEG data of the participants were recorded using a 64-channel BioSemi ActiveTwo system. The competing speech stimuli were HRTF-filtered, simulating both speakers on $\pm 60°$ azimuth direction. Three different acoustic room settings were simulated: an anechoic room, a classroom with mild reverberation, and the Hagia Irene church with high reverberation. Both the attended speaker (male/female) and acoustic room setting (anechoic/mild/high reverberation) were balanced. During the AAD experiment, the participants were asked to fixate on a crosshair.

1.8.3 Dataset C - Zink et al. (2017) [3]

Dataset C is recorded with the aim to assess AAD performance in a longitudinal, multiple-day recording with neurofeedback. It is extensively described in [3]. The data of two normal-hearing subjects are recorded in a longitudinal AAD experiment that is performed across multiple recording days. Data is recorded in eight different sessions of 24 min on seven different days. On the first day, there are two sessions, with around seven days of rest afterwards. Then data are recorded in four sessions on four consecutive recording days, followed by three days of rest. After another recording day/session, seven days of rest followed before the last recording session. Each session consisted of an AAD experiment with two competing Dutch-speaking male speakers narrating a fairytale story, and was split into four blocks of six minutes. After each 6 min-trial, multiple-choice questions were asked.

As opposed to the other datasets, the EEG data are not recorded in a laboratory setting but comfortably seated at the subject's home. A 24-channel mobile SMARTING EEG system of mBrainTrain was used. The speech stimuli were again HRTF-filtered to simulate a realistic listening scenario. During the four sessions on the four consecutive recording days, one subject received real-time feedback about the AAD performance to probe neurofeedback effects.

1.8.4 Dataset D - Das et al. (2018) [4]

Dataset D is again recorded with the aim to assess AAD performance in more complex acoustic scenarios, in this case, with different amounts of background

babble noise and different competing speaker locations. It is extensively described in [4] and has already been used in, e.g., [4, 113]. 18 normal-hearing and natively Dutch-speaking subjects participated in an AAD experiment, listening to two competing short stories narrated by two male speakers. Each trial had a different duration between 2 and 5 min. Multiple-choice questions were asked after each trial. A 64-channel BioSemi ActiveTwo system was used to record the EEG data.

Four different angular competing speaker positions were used ($\pm 90°, \pm 5°, -90/-30°, +30/+90°$ along the azimuth direction), while also background babble noise at different SNRs (no noise, $-1.1, -4.1, -7.1$ dB) were added to the competing speech streams. The babble noise consisted of four unique speech streams (two male, two female), HRTF-filtered and mixed at nine equidistant locations around the listener. A screen was shown to the participant with the experiment layout, indicating the required direction of attention.

1.9 Research objectives and overview

1.9.1 Research objectives

The general aim of this thesis is to develop novel signal processing algorithms for EEG-based AAD. It focuses, therefore, on the AAD block of Figure 1.6, however, while taking the other aspects of a neuro-steered hearing device described in Section 1.7 into account.

Before developing new AAD algorithms, it is crucial to identify the gaps in the current AAD literature that need to be addressed. Therefore, the first aim (Part I) of this thesis is to provide a comparative study on different existing AAD algorithms (Chapter 3). To be able to easily compare different AAD algorithms, we first design a new performance metric for AAD algorithms, taking the context of neuro-steered gain control into account (Chapter 2). Moreover, this new metric should allow resolving the speed-accuracy trade-off that occurs in AAD algorithms. Such a speed-accuracy trade-off is found in many different BCI applications [36].

From the comparative review study, several signal processing-related challenges for AAD are identified (related to the building blocks of a neuro-steered hearing device (Section 1.7)). In the remaining of the thesis, we focus on two of these challenges in Parts II and III, respectively:

1. **Time-adaptive, unsupervised AAD algorithms (Part II):** Most AAD algorithms require a per-user controlled experiment for training (i.e., they are *supervised* and thus require labeled training data) and use neural decoders that are fixed in time. Therefore, in this thesis, we aim to develop a novel unsupervised, time-adaptive AAD algorithm based on the SR method (Chapters 4 and 5). Such an algorithm would allow to automatically (i.e., without external intervention) adapt to changing conditions and situations, and other non-stationarities in the data.

2. **Fast and accurate AAD algorithms (Part III):** Most AAD algorithms - especially the SR-based ones - suffer from a speed-accuracy trade-off, i.e., when operating at higher speeds (shorter decision times to make a decision about the auditory attention), their accuracy drastically decreases (see Part I). To cope with this trade-off, we aim to develop a set of novel AAD algorithms that exploit a new paradigm: decoding the spatial focus of auditory attention from the EEG. In order to do so, we use the common spatial pattern (CSP) algorithm (Chapter 6) and a Riemannian geometry-based classification (RGC) algorithm (Chapter 7).

1.9.2 Chapter overview

This thesis is split into three main parts, each containing two chapters, as can be seen in the overview in Figure 1.7. The first part comprises the evaluation and comparison of different AAD algorithms (Part I). The performance metric designed in Chapter 2 to evaluate AAD algorithms is used in the comparative study but also in the other parts. Two signal processing-related knowledge gaps identified in the comparative study in Chapter 3 are addressed in the subsequent parts. In Part II, we develop an unsupervised time-adaptive AAD algorithm based on the SR method, while in Part III, the new paradigm of decoding the spatial focus of auditory attention is exploited using a CSP- and RGC-based method. As such, Parts II and III each focus on a different AAD paradigm: SR-based AAD (Part II) and spatial focus-based AAD (Part III). The last part describes the conclusions and future research directions (Part IV).

Each chapter is largely based (with minor adaptations) on a published/accepted and peer-reviewed paper. All papers are first-authored by myself, in close collaboration with both supervisors. Chapter 3 is the result of an international collaboration with various leading authors in the field of AAD. A more detailed overview of the different chapters can be found below.

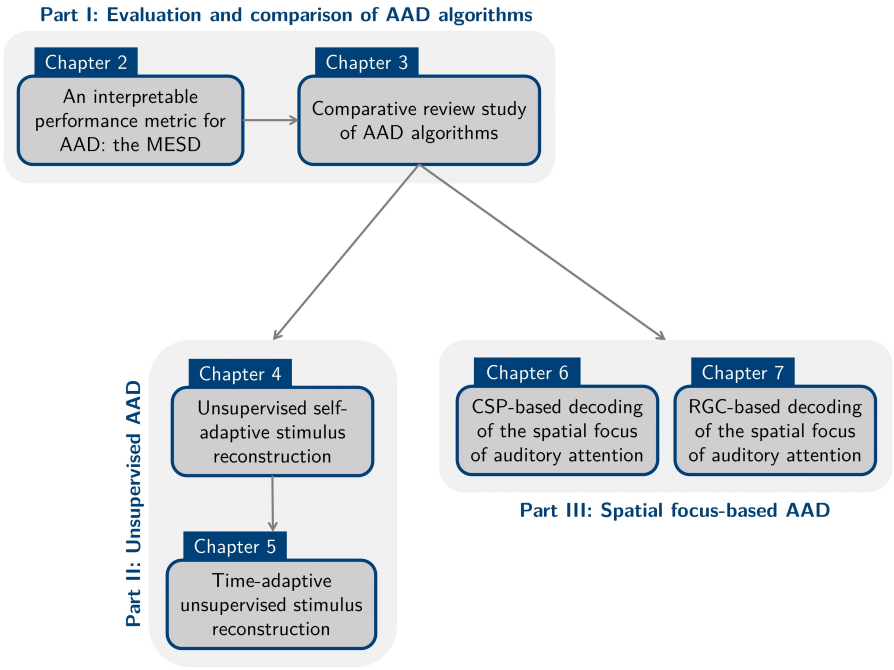


Figure 1.7: An overview of the different chapters in this thesis and their relations. The comparative review study of AAD algorithms allowed us to identify the most important signal processing-related challenges in AAD, two of which are addressed in the two subsequent parts. The performance metric designed in Chapter 2 is used throughout the thesis.

Part I: Evaluation and comparison of AAD algorithms

Chapter 2 describes the design of a novel, interpretable performance metric for AAD algorithms, which resolves the trade-off between accuracy and decision time (speed), allows for easy (statistical) comparison of different AAD algorithms, and returns as a by-product an optimized gain control system for a neuro-steered hearing device. This performance metric is an essential tool for the comparative study in Chapter 3 and is further used throughout the thesis.

Chapter 3 provides a comparative review study of different AAD algorithms. First, we provide a broad review of different AAD algorithms, which are then compared to each other using the performance metric of Chapter 2. The results show that linear SR methods are robust but too slow for practical purposes.

Decoding the spatial focus of auditory attention, which is further pursued in Part III, shows to be more promising. DNN-based algorithms turn out to be hard to replicate. From this review study, we also extract a few signal processing-related challenges for AAD, of which unsupervised, time-adaptive AAD is addressed in Part II.

Part II: Unsupervised AAD

Chapter 4 describes a novel unsupervised (i.e., without information on which speaker is attended or unattended) self-adapting training procedure for a subject-specific stimulus decoder, in which the decoder iteratively improves itself based on its own predicted labels. We show that the resulting decoder performs better than a supervised subject-independent decoder. Furthermore, we also provide a mathematical analysis that explains why this unsupervised training procedure works even when starting from a random initial decoder.

Chapter 5 extends the unsupervised stimulus decoder, which is trained in batch and fixed in time, to an online *time-adaptive* unsupervised SR method that continuously and automatically adapts over time to non-stationarities in the EEG. We propose both a sliding window and recursive implementation, and show that the proposed time-adaptive unsupervised decoder outperforms a time-invariant supervised decoder.

Part III: Decoding the spatial focus of auditory attention

Chapter 6 explores an alternative AAD paradigm to improve the performance on short decision windows: decoding the spatial focus of auditory attention using a filterbank CSP-based (FB-CSP) classification method. We show that the FB-CSP method outperforms the SR method, and performs at least on par with a similar CNN-based method. Furthermore, we show that the method still works using EEG channels only around the ear and that it can adapt to unlabeled data from an unseen subject.

Chapter 7 extends the CSP method to an RGC method. The covariance matrix of an EEG window is directly classified into a direction of auditory attention while taking its Riemannian structure into account. We show that this RGC method outperforms the CSP method both for a binary and multiclass classification problem.

Part IV: Conclusion

Chapter 8 summarizes the main findings of the thesis in relation to the research objectives in Section 1.9.1, and provides several suggestions for future research, building upon the work presented in this thesis.

Appendices

Appendix A contains an article for the general public about the thesis topic, originally written for the Flemish science magazine EOS (in Dutch), and the Leuven.AI-stories blog and BioVox newsletter (English).

Part I

Evaluation and comparison of AAD algorithms

2 | An interpretable performance metric for AAD algorithms in a context of neuro-steered gain control

This chapter is based on S. Geirnaert, T. Francart, and A. Bertrand, "An Interpretable Performance Metric for Auditory Attention Decoding Algorithms in a Context of Neuro-Steered Gain Control," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 28, no. 1, pp. 307-317, 2020. Two figures (Figures 2.4 and 2.5) and a table (Table 2.2) have been added.

ABSTRACT | While numerous AAD algorithms have appeared in the literature (see Chapter 3), their performance is generally evaluated in a non-uniform manner. Furthermore, AAD algorithms typically introduce a trade-off between the AAD accuracy and the time needed to make an AAD decision, which hampers objective benchmarking as it remains unclear which point in each algorithm's trade-off space is optimal in a context of neuro-steered gain control. To this end, we present in this chapter an interpretable performance metric to evaluate AAD algorithms based on an adaptive gain control system steered by AAD decisions. Such a system can be modeled as a Markov chain, from which the minimal expected switch duration (MESD) can be calculated and interpreted as the expected time required to switch the operation of the hearing device after an attention switch of the user, thereby resolving the trade-off between accuracy and decision time. Furthermore, we show that the MESD calculation provides an automatic and theoretically founded procedure to optimize the number of gain levels and decision time in an AAD-based adaptive gain control system.

2.1 Introduction

While several different AAD algorithms have been proposed in the literature (see Chapter 3 for an overview), an important question is how these AAD algorithms should be evaluated, especially in the context of neuro-steered hearing devices. Their accuracy, measured as the percentage of decision windows in which the attention was decoded correctly, depends on the length of the decision window, which defines how much EEG data are available to make a decision. Because of the low SNR of the neural response to the speech signals in the EEG, the accuracy of traditional (SR-based) AAD algorithms increases with the length of the decision window. However, a longer decision window implies that the algorithm also needs more time to, for example, react on a switch in attention, resulting in a trade-off. This trade-off between accuracy and decision window length leads to three fundamental issues regarding the evaluation of AAD algorithms:

1. The dependence of the accuracy on the decision window length hinders easy statistical comparison, as the different decision window lengths need to be taken into account as an extra factor. This hampers drawing adequate statistical conclusions.

2. Algorithm A might perform better than algorithm B for shorter decision window lengths, while algorithm B might perform better than algorithm A for long decision window lengths, leading to inconclusiveness when benchmarking both algorithms.

3. In several scientific reports, only one decision window length with the corresponding accuracy is reported. A different choice of the decision window lengths (for example, across two scientific reports) then obstructs a fair comparison.

The aforementioned problems motivate the need for a single-number metric to capture the overall AAD performance, which also takes the trade-off between accuracy and decision time into account by selecting the optimal point on the trade-off curve that is most relevant in the context of adaptive gain control for neuro-steered hearing devices.

de Taillez et al. [143] adopted the Wolpaw information transfer rate ($\mathrm{ITR_W}$) $[\frac{\mathrm{bit}}{\mathrm{s}}]$ from the BCI community to combine the accuracy and the decision window length in a single metric as follows [144]:

$$\mathrm{ITR_W} = \frac{1}{\tau}\left(\log_2 M + p\log_2 p + (1-p)\log_2\frac{1-p}{M-1}\right), \tag{2.1}$$

with p the accuracy (probability of a correct decision), τ the decision time (here: decision window length), and M the number of classes (here: speakers). Similarly, Wong et al. [142] used the Nykopp ITR (ITR_N) to evaluate AAD algorithms, which assumes an adaptive BCI setting in which not every time a decision has to be made [142]. The ITR_W was originally defined to quantify the performance of BCI systems that are used to re-establish or enhance communication and control for paralyzed individuals with severe motor impairments [144]. It quantifies the number of bits that can be transferred per time unit and matches, as such, the specific context of communicating through brain waves. However, the $\text{ITR}_\text{W/N}$ has no such clear interpretation in the context of AAD for neuro-steered hearing devices and is, therefore, not per se a relevant criterion to compare AAD algorithms. Instead, we are interested in *how fast* a hearing device can switch its operation from one speaker to another, following an intentional attention switch of the user, based on consecutive AAD decisions and taking into account that some decisions may be incorrect.

The lack of an *interpretable* metric in the context of neuro-steered hearing devices, which combines both decision time and accuracy in a single metric, and which facilitates making unambiguous conclusions on performance and easy comparisons between algorithms, motivates the design of a new metric, which we refer to as the *minimal expected switch duration* (MESD)[1]. The MESD metric is based on the performance of an adaptive gain control system that is optimized for the AAD algorithm under test. Therefore, the derivation of the MESD metric also leads to an automatic and theoretically founded procedure to optimize the step size and decision frequency in an AAD-based adaptive gain control system, thereby avoiding tedious manual tuning.

In Section 2.2, we develop this new metric step-by-step, leading to a closed-form expression based on which the metric can be computed. In Section 2.3, we give examples of the MESD metric on real EEG/audio data, as well as a comparison with the $\text{ITR}_\text{W/N}$ metric. Conclusions are drawn in Section 2.4.

2.2 Expected switch duration

2.2.1 An adaptive gain control system

Given that AAD algorithms decode the attention of a hearing device user, hearing devices could benefit from an adaptive gain/volume control system. Given a two-speaker situation, such a system would allow to adaptively over

[1] We provide an open-source implementation to compute the MESD metric, which can be found online on `https://github.com/exporl/mesd-toolbox` [145].

time change the gain of speaker one versus speaker two, tracking the attention of the hearing devices user (Figure 2.1). We, however, want to avoid the usage of only two volume settings or gain control 'states', i.e., all-or-nothing amplification of both speakers, as this would cause perceptually unpleasant spurious and sudden switching of speakers (of which many by mistake). Moreover, we want to enable the user to adequately react when the system starts switching towards the wrong speaker due to AAD errors, before the attended speaker becomes unintelligible. As a result, the system should have many states to gradually and adaptively change the relative gain between both speakers.

However, this results in two crucial design parameters which both affect the performance of the system, each leading to a fundamental trade-off, which is illustrated in Figure 2.1:

1. **How many gain levels should we use?** As Figure 2.1a illustrates, using fewer gain levels results in a faster gain switch after an attention switch but also results in a less stable gain process, negatively affecting the comfort of the user. Increasing the number of gains stabilizes the gain process and thus results in a more robust gain control but increases the gain switch time.

2. **How often should we take a step?** A short decision window length corresponds to a fast gain control system - as less EEG and audio data need to be buffered before a decision can be made - and thus a fast gain switch (Figure 2.1b). However, as is indicated in Section 2.1, a shorter decision window length also corresponds to a lower accuracy, resulting in a more unstable gain process - vice versa for a longer decision window length.

Note that optimizing a *discrete* gain level system does not imply that there needs to be a discrete implementation in a hearing device. One could also continuously interpolate between the discrete gain levels to provide a more pleasant user experience. In that case, optimizing the rate of change of the volume (for example, the slope) corresponds to optimizing the number of gain levels.

In the following sections, we translate this adaptive gain control system into mathematics using a Markov chain model. This mathematical formulation will allow us to rigorously address these fundamental issues and optimize these two design parameters, as well as provide a way to properly evaluate and rank different AAD algorithms through the novel MESD metric, which is derived from the optimal gain control design for the AAD algorithm under test. This MESD metric is formally defined in Section 2.2.5.

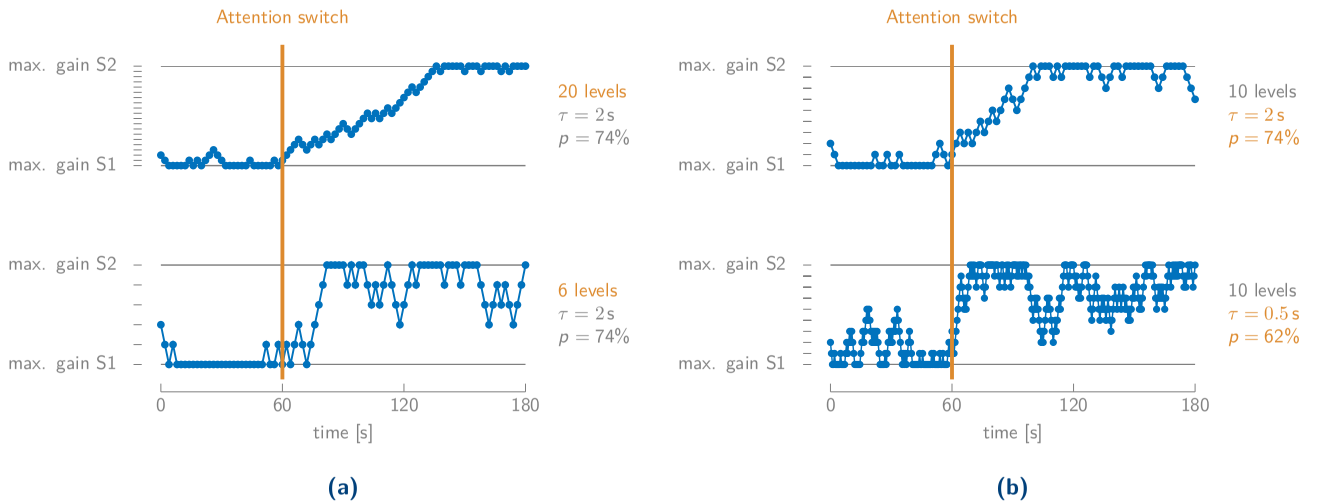


Figure 2.1: This example illustrates the two fundamental issues regarding an adaptive gain control system with decision window length τ and AAD accuracy p. In the first minute, speaker one (S1) is the attended speaker, while after 60 s, the attention switches to speaker two (S2). **(a)** When the number of gain levels decreases, the gain switch is performed faster, but the overall gain process is less stable. **(b)** Decreasing the decision window length - and correspondingly the accuracy - results in a faster gain switch but less stable gain process, and vice versa.

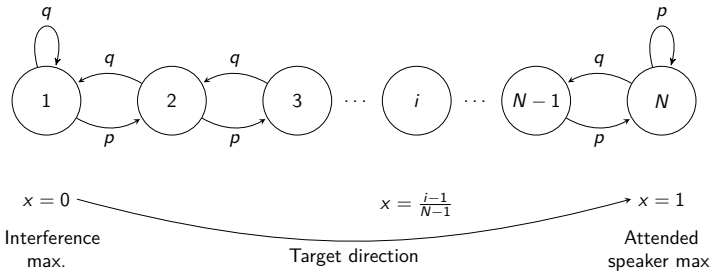


Figure 2.2: An adaptive gain control system can be modeled as a Markov chain with N states (gains) and a transition probability p in the target direction (attended speaker) equal to the accuracy of the AAD algorithm.

Adaptive gain control parameter	Markov chain parameter
gains	states $x \in [0, 1]$
number of (relative) gain levels	number of states N
AAD accuracy	transition probability p
decision window length	step time τ

Table 2.1: The different concepts of an adaptive gain control system have a straightforward translation to a Markov chain parameter.

2.2.2 Markov chain model

The adaptive gain control system of Section 2.2.1 can be straightforwardly translated into a mathematical model using a Markov chain (Figure 2.2). Table 2.1 shows how the parameters of the Markov chain embody several concepts of the adaptive gain control system. The different mathematical notations related to the Markov chain that are used and explained in the following sections are summarized in Table 2.2.

The Markov chain contains N states, each corresponding to a relative gain level $x \in [0, 1]$ of the attended speaker versus background noise, including the interfering speaker(s). Each state corresponds to one single gain level, such that a simple Markov model suffices and a hidden Markov model is not required. For illustrative purposes throughout the manuscript, but without loss of generality, we will consider the example of a noiseless two-speaker scenario. In this case, $x = 1$ would correspond to a target relative amplification of the attended speaker versus the unattended speaker, which is typically constrained to still enabling the listener to switch attention to the other speaker. $x = 0$ then corresponds to the maximal suppression of the attended speaker with a similar constraint, while

Mathematical notation	Meaning
$x \in [0, 1]$	gain level
$N \geq 2$	number of states
$i \in \{1, \ldots, N\}$	state index
$p \in [0, 1], q = 1 - p, r = \frac{p}{q}$	transition probability (and related quantities)
$\tau > 0$	step time
$\pi(i) \in [0, 1]$	steady-state distribution
$P_0 \in [0, 1]$	confidence level
$\bar{k} \in \{1, \ldots, N\}$	lower bound (state index) of the P_0-confidence interval
$\bar{x} \in [0, 1]$	lower bound (gain level) of the P_0-confidence interval
$c \in [0, 1]$	predefined comfort level
$k_c \in \{1, \ldots, N\}$	smallest state corresponding to gain $x \geq c$
$h_j(i) \geq 0$	mean hitting time from state i to j

Table 2.2: A summary of the different mathematical notations related to the Markov chain.

$x = 0.5$ implies equal gain for both speakers. These gain levels are assumed to be uniformly distributed over $[0, 1]$, resulting in a one-to-one relation between state i and gain level x:

$$x = \frac{i - 1}{N - 1}.$$

Given that $x = 1$ corresponds to the target gain level of the attended speaker, the transition probability $p \in [0, 1]$ in the target direction is equal to the probability of a correct AAD decision, i.e., the AAD accuracy. Similarly, $q = 1 - p$ corresponds to the probability of a wrong decision. In what follows, we assume that $p > 0.5$, i.e., the evaluated AAD algorithm performs at least better than chance level. A correct (step towards $x = 1$) or incorrect (step towards $x = 0$) decision always results in a transition to a neighboring state, except in state 1 or state N, where no state transition is made after an incorrect or correct decision, respectively (for example, in state N, the gain is maximal for the attended speaker, which is the best the system can obtain). The latter is indicated by the self-loops in Figure 2.2, which models the gain clipping in Figure 2.1.

Each step takes τ seconds - the decision window length -, as τ seconds of EEG and audio data need to be buffered before a new decision can be made. The application of an AAD algorithm on consecutive windows of τ seconds, which results in a gain process such as shown in Figure 2.1, thus corresponds to a random walk process through the Markov chain. Note that the AAD accuracy p directly depends on this decision window length τ, as denoted before. The $p(\tau)$-performance curve relates this AAD accuracy p with the decision window length τ for a particular AAD algorithm (see Figure 2.6 for an example).

The two fundamental issues regarding the gain control system as listed in Section 2.2.1 can now be translated into the optimization of the Markov chain parameters:

1. Optimizing the number of gain levels corresponds to the optimization of the number of states N (this will be derived in Section 2.2.3).
2. Determining the time resolution with which the gain should be adapted corresponds to determining the step time τ (this will be derived in Section 2.2.4). Note that equivalently, the transition probability p can be optimized. Addressing this second issue corresponds to jointly optimizing the AAD accuracy p and decision window length τ, as they are directly related through the $p(\tau)$-performance curve. The resulting pair $(\tau_{\text{opt}}, p_{\text{opt}})$ is called the *optimal working point* on the $p(\tau)$-performance curve.

We will answer both of these questions through a mathematical analysis on the corresponding Markov chain in Sections 2.2.3 and 2.2.4, respectively, which will lead to the MESD metric in Section 2.2.5. However, it should be emphasized that this Markov chain is a simplified model of a real gain control system, and, as always, this mathematical tractability comes at the cost of making some simplifying assumptions. Indeed, a Markov chain assumes independence of the consecutive decisions[2], which may be violated in a practical AAD algorithm, in particular when there is overlap in the data of consecutive windows.

2.2.3 Optimizing the number of states N

We first optimize the number of states N, where we mainly target a stable gain process, tackling one of the trade-offs in Figure 2.1a (a stable gain process versus fast switching).

Steady-state distribution

The steady-state distribution of the Markov chain in Figure 2.2 is needed in order to analyze the behavior of the modeled adaptive gain control system. This steady-state distribution $\pi(i) = P(x = \frac{i-1}{N-1}), i \in \{1, \ldots, N\}$ is defined as the probability to be in state i after an infinite number of random steps (starting from any position), for a fixed transition probability p. Defining $r = \frac{p}{q}$, the

[2]It is noted that the ITR metric uses a similar assumption, as it implicitly assumes independence between consecutive messages (i.e., AAD decisions).

steady-state distribution is shown in Appendix 2.A to be equal to:

$$\pi(i) = \frac{r-1}{r^N - 1} r^{i-1}, \forall\, i \in \{1, \ldots, N\}. \tag{2.2}$$

P_0-confidence interval

Based on the Markov chain model and the steady-state distribution, we determine a desirable operating region of the neuro-steered hearing device via the P_0-confidence interval $[\bar{x}, 1]$. This is the smallest interval in which the system must operate for at least P_0 percent of the time, despite the presence of AAD errors, while being in a steady-state regime. For example, if $P_0 = 0.8$, we expect the hearing device to operate in the operating region $x \in [\bar{x}, 1]$ for at least 80% of the time. This implies that we search for the largest $\bar{k}$ for which:

$$\sum_{j=\bar{k}}^{N} \pi(j) \geq P_0. \tag{2.3}$$

This leads to the following lower bound $\bar{k}$ of the P_0-confidence interval (the derivation is given in Appendix 2.B):

$$\bar{k} = \left\lfloor \frac{\log\left(r^N(1-P_0) + P_0\right)}{\log(r)} + 1 \right\rfloor, \tag{2.4}$$

with $\lfloor \cdot \rfloor$ the flooring operation yielding an integer output. The resulting P_0-confidence interval is thus defined as[3]:

$$[\bar{x}, 1] = \left[\frac{\bar{k} - 1}{N - 1}, 1 \right]. \tag{2.5}$$

The P_0-confidence interval is indicated in orange in Figure 2.3.

Design constraints

From Figure 2.1, it can be intuitively seen that to minimize the gain switch duration, we have to minimize the number of states N. However, we also know that this conflicts with the stability of the gain process (Figure 2.1). To guarantee a certain amount of stability or confidence of the system and comfort to the user, we propose the following design criteria for the Markov chain regarding N:

[3] Due to the discretization of x, the probability of being in $[\bar{x}, 1]$ is generally larger than P_0. However, (2.4) ensures that $[\bar{x}, 1]$ is the *smallest* possible interval such that $x \in [\bar{x}, 1]$ for *at least* P_0 percent of the time.

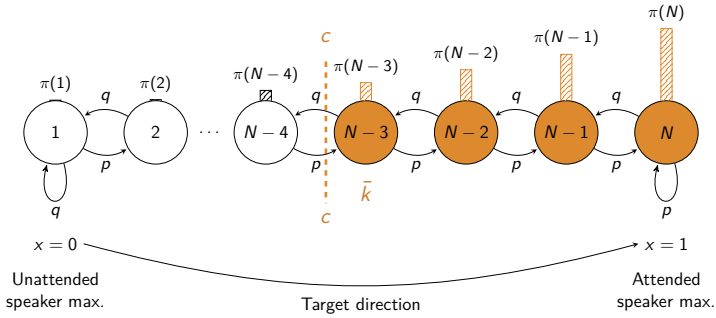


Figure 2.3: The P_0-confidence interval in orange is the smallest set of states for which the sum of the steady-state probabilities (bars) is larger than P_0. The second design constraint forces the lower bound $\bar{k}$ of this P_0-confidence interval to be above a predefined level c, assuring stability of the system.

- The lower bound of the P_0-confidence interval $\bar{x}$ should be larger than a pre-defined 'comfort level' c that defines the target operating region, i.e., $\bar{x} \geq c$. This comfort level c can be determined from hearing tests, for example, by interpreting it physiologically as the gain level below which it becomes uncomfortable to listen to the attended speaker (see Section 2.3.1, where we will motivate to choose $c = 0.65$). By controlling N, we can thus ensure that the hearing device is in P_0 percent of the time above this comfort level c, ergo, stabilizing the gain process. With (2.4) and (2.5), the above requirement results in the following inequality:

$$\bar{x} = \frac{\bar{k} - 1}{N - 1} \geq c, \tag{2.6}$$

  which should be viewed as a constraint when minimizing N (note that $\bar{k}$ also depends on N). A key message here is that a *lower* accuracy p requires *more* states N in order to guarantee (2.6), as illustrated in Figure 2.4.

- $N \geq N_{\min}$: a minimal number of states is desired to obtain a sufficiently smooth transition in the gain adaptation. In particular, we want to avoid the immediate crossing of the mid-level $x = 0.5$ (i.e., an immediate change of the loudest speaker) when leaving the P_0-confidence interval due to an incorrect AAD decision. In cases where (2.6) is satisfied for $N = 4$, the P_0-confidence interval also often[4] contains state 3, which would result in

[4] This holds unless $p > P_0$, in which case the P_0-confidence interval collapses to a single state $N = 4$.

an immediate crossing of $x = 0.5$ when leaving the P_0-confidence interval due to an AAD error. Therefore, we propose to fix $N_{\min} = 5$.

In practice, the minimal number of states N can be found by going over the candidate values $N = N_{\min} + i$, with $i = 0, 1, 2, \ldots$, in this specific order (as the gain switch duration increases with N), until a value N is found that satisfies (2.6). As shown in Appendix 2.C, such a value of N can always be found, for any value of c and P_0, assuming that $p > 0.5$.

2.2.4 Finding the optimal working point $(\tau_{\text{opt}}, p_{\text{opt}})$

In Section 2.2.3, we have constrained N such that the gain process has a minimum of stability. Therefore, we can now focus on minimizing the gain switch time. In this section, we rigorously define the expected switch duration (ESD), which quantifies this gain switch time, and use it as a criterion to determine the optimal working point $(\tau_{\text{opt}}, p_{\text{opt}})$.

Mean hitting time

A fundamental metric within the Markov chain is the *mean hitting time* (MHT), which quantifies the expected number of steps s needed to arrive in target state j when starting from a given initial state i. The MHT is defined as:

$$h_j(i) \triangleq \mathbb{E}\{s|i \to j\} \triangleq \sum_{s=0}^{+\infty} sP(s|i \to j), \tag{2.7}$$

with $i, j \in \{1, \ldots, N\}$, $\mathbb{E}\{\cdot\}$ denoting the expectation operator and where $P(s|i \to j)$ is the probability that target state j is reached for the first time after s random steps, when starting in state i. Note that we are only interested in the MHT for the case where $i \leq j$, i.e., when going from left to right in the Markov chain (Figure 2.2). This corresponds to the case where the hearing device switches from one speaker to the other. In Appendix 2.D, we show that the MHT can be computed as:

$$h_j(i) = \frac{j-i}{2p-1} + \frac{p(r^{-j} - r^{-i})}{(2p-1)^2}, \forall\, i \leq j. \tag{2.8}$$

Expected switch duration

We define a gain switch as the transition to the comfort level c, starting from *any* initial state i with a corresponding gain level outside the predefined working

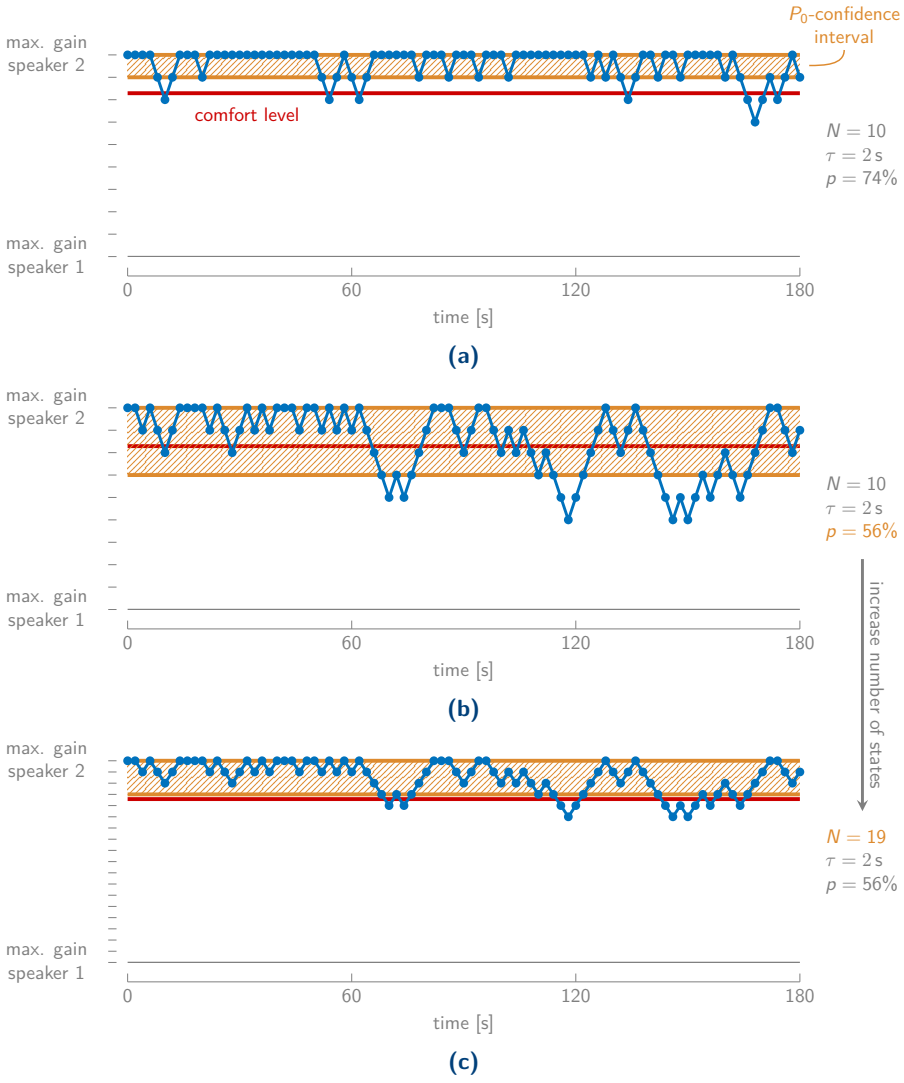


Figure 2.4: The number of states N needs to adapted such that the lower bound of the P_0-confidence interval is larger than the pre-defined comfort level. In **(a)**, $N = 10$ states is enough to ensure that the design constraint is obeyed, given an AAD accuracy of $p = 74\%$. In **(b)**, the accuracy decreases to $p = 56\%$, such that the lower bound of the P_0-confidence interval lies not anymore above the comfort level given the fixed number of states $N = 10$. Therefore, as shown in **(c)**, we need to increase the number of levels N to $N = 19$ to make sure that the design constraint is obeyed again.

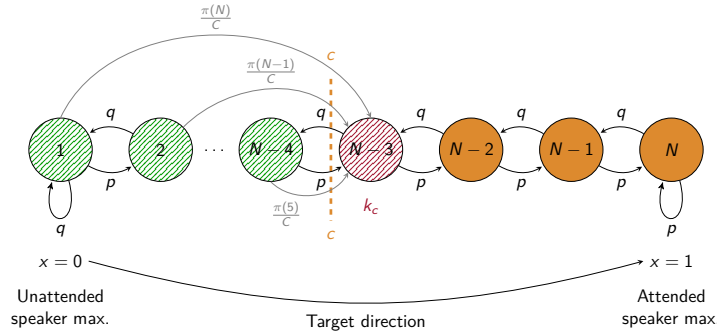


Figure 2.5: A gain switch is defined as the transition from any initial state outside the predefined working region $[c, 1]$ to k_c, the first state corresponding to a relative gain $x \geq c$. To compute the ESD, for each of the potential initial states, the MHT is weighted with the probability $\frac{\pi(N-i+1)}{C}$ to be in that state (given you are outside the working region) right *before* the switch (with $C = \sum_{l=1}^{k_c-1} \pi(N-l+1)$).

region $[c, 1]$, as visualized in Figure 2.5. This specific definition of a gain switch implies that we are aiming at quantifying the duration of a *stable* switch. The perceived gain switch towards the attended speaker by the hearing device user would typically occur earlier, for example, when $x = 0.5$ is reached. The corresponding gain switch time is called the *expected switch duration* (ESD) [s]. The ESD thus quantifies the time needed to change the operation of the system when the user shifts its attention *and* when the system is not yet in the desired operating region.

Assuming k_c is the first state corresponding to a relative gain $x \geq c$:

$$k_c = \lceil c(N-1) + 1 \rceil,$$

the ESD is formally defined as the expected time (step time τ times expected number of steps s) necessary to go from any state $i < k_c$ to target state k_c:

$$\text{ESD} \triangleq \tau \mathbb{E}\{s | i \to k_c, \forall\, i < k_c\} \triangleq \tau \sum_{s=0}^{+\infty} s P(s | i \to k_c, \forall\, i < k_c),$$

with $P(s|i \to k_c, \forall\, i < k_c)$ the probability that target state k_c is reached for the first time after s steps, when starting from any state $i < k_c$. Using marginalization in the initial state i, this can be written as:

$$\text{ESD} = \tau \sum_{s=0}^{+\infty} s \sum_{i=1}^{N} P(s | i \to k_c, i < k_c) P(i | i < k_c), \tag{2.9}$$

with $P(i|i < k_c)$ the probability to be in state i, given that $i < k_c$. Bayes' law can be applied to find $P(i|i < k_c)$:

$$P(i|i < k_c) = \frac{P(i < k_c|i)P(i)}{P(i < k_c)},$$

with:

- $P(i) = \pi(N - i + 1)$, where we reversed the order in the steady-state distribution (2.2) (see also Figure 2.5). Indeed, note that i is the initial state at the moment of the attention switch, i.e., when being in the steady-state regime from right before the switch, where state 1 was the target state (the reverse of Figure 2.2).
- $P(i < k_c) = \sum_{l=1}^{k_c-1} \pi(N - l + 1)$.
- $P(i < k_c|i) = 1$ when $i < k_c$ and $= 0$ otherwise.

Plugging this into (2.9) and using the definition of the MHT in (2.7) and the steady-state distribution in (2.2), we eventually find:

$$\mathrm{ESD}(p(\tau), \tau, N) = \tau \frac{r^{k_c+1} - r^{k_c}}{r^{k_c} - r} \sum_{i=1}^{k_c-1} r^{-i} h_{k_c}(i), \tag{2.10}$$

where $h_{k_c}(i)$ is given by (2.8). Note that the $\mathrm{ESD}(p(\tau), \tau, N)$ (2.10) implicitly depends on N as the state index $k_c = \lceil c(N-1) + 1 \rceil$ depends on N.

Given the $p(\tau)$-performance curve of an AAD algorithm, constructed by piecewise linear interpolation through the points $(\tau_i, p_i), i \in \{1, \ldots, I\}$ on the $p(\tau)$-performance curve for which the AAD performance is evaluated on real data[5], the optimal working point $(\tau_{\mathrm{opt}}, p_{\mathrm{opt}})$ is defined as the pair for which the $\mathrm{ESD}(p(\tau), \tau, N)$ is minimal, given that N obeys the constraints of Section 2.2.3.

2.2.5 The minimal expected switch duration

Optimizing N, τ, and p now results in an optimal Markov chain that satisfies the stability constraints and has minimal ESD. The minimal ESD over the $p(\tau)$-performance curve, which gave rise to the optimal working point $(\tau_{\mathrm{opt}}, p_{\mathrm{opt}})$, can now be used as a single-number metric, referred to as the *minimal expected*

[5] In this chapter, we assume that p is fixed and evaluated over all data windows (batch).

switch duration (MESD), allowing to compare different AAD algorithms or parameter settings of the latter. This metric is defined as follows:

Minimal expected switch duration (MESD)

The minimal expected switch duration (MESD) is the expected time required to reach a predefined stable working region defined via the comfort level c, after an attention switch of the hearing device user, in an optimized Markov chain as a model for an adaptive gain control system. Formally, it is the expected time to reach the comfort level c in the fastest Markov chain with at least $N_{\min}$ states for which $\bar{x} \geq c$, i.e., the lower bound $\bar{x}$ of the P_0-confidence interval is above c:

$$\begin{aligned} \text{MESD} = \min_{N,\tau} \quad & \text{ESD}(p(\tau), \tau, N) \\ \text{s.t.} \quad & \bar{x} \in [c, 1] \\ & N \geq N_{\min}, \end{aligned} \tag{2.11}$$

where $\text{ESD}(p(\tau), \tau, N)$ as in (2.10) and $\bar{x} = \frac{\bar{k}-1}{N-1}$, with $\bar{k}$ as in (2.4).

The solution of optimization problem (2.11) is straightforward, given that $\text{ESD}(p(\tau), \tau, N)$ is monotonically nondecreasing with N (see the proof in Appendix 2.E) for a fixed τ. Therefore, for each τ, choose the minimum $\hat{N}_\tau$ such that the two inequality constraints of (2.11) are obeyed (in Appendix 2.C, it is proven that such an $\hat{N}_\tau$ can always be found). As such, N is removed from the optimization problem, resulting in an unconstrained optimization problem:

$$\text{MESD} = \min_{\tau} \text{ESD}(p(\tau), \tau, \hat{N}_\tau).$$

The MESD is then defined as the minimal ESD over all window lengths τ, at optimal working point $(\tau_{\text{opt}}, p_{\text{opt}})$. Algorithm 1 summarizes the computation of the MESD metric.

It is important to minimize the ESD over an as large as possible interval of decision window lengths, especially towards short decision window lengths. As a rule of thumb, assuming a monotonically decreasing accuracy with decreasing decision window length, one should consider a larger evaluation interval when the optimal working point $(\tau_{\text{opt}}, p_{\text{opt}})$ obtained in Algorithm 1 is located at the boundary of the evaluated interval.

As an inherent by-product of the optimization problem in (2.11), the MESD metric also results in an optimal adaptive gain control system - optimal number of gains N and optimal working point $(\tau_{\text{opt}}, p_{\text{opt}})$ - for a neuro-steered hearing device.

Algorithm 1: Computation of the MESD metric
(code available in MESD toolbox [145])

Input: Evaluated points on the $p(\tau)$-performance curve $(\tau_i, p_i), i \in \{1, \ldots, I\}$, the required number of interpolated samples K of the performance curve $p(\tau)$ and the hyperparameters: confidence interval P_0, lower bound c, and minimum number of states $N_{\min}$. In order to standardize future AAD algorithm evaluations, the suggested default values are $K = 1000, P_0 = 0.8, c = 0.65$ and $N_{\min} = 5$ (Section 2.3.1).
Output: MESD, $(\tau_{\text{opt}}, p_{\text{opt}})$

1: Construct K samples of the performance curve $p(\tau)$ by piecewise linear interpolating through evaluated points $(\tau_i, p_i), i \in \{1, \ldots, I\}$
2: **for** each sampled τ **do**
3: Find $\hat{N}_\tau$ by going over the candidate values $N = N_{\min} + i$, with $i = 0, 1, 2, \ldots$, in this specific order, until the first value N is found that satisfies:

$$\frac{\bar{k} - 1}{N - 1} \geq c \text{ and } N \geq N_{\min},$$

with

$$\bar{k} = \left\lfloor \frac{\log\left(r^N(1 - P_0) + P_0\right)}{\log(r)} + 1 \right\rfloor \text{ and } r = \frac{p(\tau)}{1 - p(\tau)}.$$

4: Given $\hat{N}_\tau$, compute

$$\text{ESD}(p(\tau), \tau, \hat{N}_\tau) = \tau \frac{r^{k_c+1} - r^{k_c}}{r^{k_c} - r} \sum_{i=1}^{k_c-1} r^{-i} h_{k_c}(i),$$

with

$$h_{k_c}(i) = \frac{k_c - i}{2p - 1} + \frac{p(r^{-k_c} - r^{-i})}{(2p - 1)^2} \text{ and } k_c = \lceil c(N - 1) + 1 \rceil$$

.
5: **end for**
6: The MESD is equal to the minimum ESD over all sampled τ:

$$\text{MESD} = \min_\tau \text{ESD}(p(\tau), \tau, \hat{N}_\tau),$$

obtained at optimal working point $(\tau_{\text{opt}}, p_{\text{opt}}) = (\tau_{\text{opt}}, p(\tau_{\text{opt}}))$.

2.3 Experiments

We illustrate the MESD by applying the AAD algorithm in [74] on Dataset A, which consists of 72 minutes of recorded EEG and audio data per subject (16 normal-hearing subjects). The subjects were instructed to listen to a specific speech stimulus in a competing two-speaker situation, including 24 minutes of repetitions but without inter-trial attention switches. The 64-channel EEG data are bandpass filtered between 1–9 Hz and downsampled to 20 Hz. The speech envelopes are computed using a power-law operation with exponent 0.6 after subband filtering [1] and are afterwards similarly bandpass filtered and downsampled. We assume that the clean envelopes of the original speech signals are available.

A linear spatio-temporal decoder, where the temporal dimension of the filter covers from 0 to 250 ms post-stimulus, is trained to decode the attended speech envelope from the EEG data by minimizing the mean squared error (MMSE) between the actually attended and reconstructed speech envelope on a training set. Per-subject decoders are trained and tested in a leave-one-segment-out (LOSO) fashion, using segments of consistent attention with a length of 60 s. We apply the same adaptations to [74] as in [1], by training one decoder across all training segments and not averaging per-segment decoders. At test time, the trained filter decodes a speech envelope from a decision window of left-out EEG data of length τ (which is a subset of the left-out 60 s segment). The Pearson correlation coefficient is computed between the reconstructed speech envelope and the envelopes of both signals presented to the subject. The speech stream with the highest correlation is identified as the attended speaker.

To evaluate the algorithm on shorter decision window lengths, the left-out segment is segmented into shorter decision windows on which the corresponding decoder is applied. Reusing the decoders allows for a fair comparison of the algorithm across different decision window lengths. The percentage of correct decisions p, per subject and decision window length τ, is computed as the total number of correct decisions divided by the total number of decisions across all segments.

2.3.1 Hyperparameter choice

The MESD depends on three hyperparameters: the confidence level P_0, the lower bound of the desired operating region c, and the minimum number of states $N_{\min}$. When optimizing the design of a gain control system, the values of these hyperparameters can be set in a user-dependent fashion according to

individual users' needs and hearing capabilities (in particular for the desired comfort level c, which is very personal). However, in order to use the MESD as a standardized performance metric for comparing AAD algorithms, we also determined reasonable values for these hyperparameters and propose them as fixed inherent parameters of the MESD performance metric as a standard for future AAD algorithm comparison. We already motivated the choice for $N_{\text{min}} = 5$ in Section 2.2.3.

In order to find a value for the comfort level c, we need to determine the SNR (between attended and unattended speaker) corresponding to relative gain level $x = 1$ (SNR_{max}) and the SNR corresponding to relative gain level $x = c$ (SNR_c). Using that $x = 0.5$ corresponds to 0 dB, $x = c$ can be found from:

$$c = \frac{10^{\text{SNR}_c/20} - 1}{2(10^{\text{SNR}_{\text{max}}/20} - 1)} + 0.5. \tag{2.12}$$

We here define SNR_{max} objectively as the speech reception threshold (SRT), corresponding to the 50% speech intelligibility level of the suppressed speaker, which should enable the hearing device user to understand the suppressed speaker sufficiently, in order to assess whether they want to switch attention. Correspondingly, we define SNR_c as the SNR where there is full speech understanding *and* where the listening effort saturates, i.e., a higher SNR does not result in a better speech understanding nor less listening effort. Ohlenforst et al. [146] investigated the correct sentence recognition scores and peak pupil dilation, which quantifies the listening effort, when listening to standard Dutch sentences in the presence of a competing talker masker at SNRs corresponding to daily-life conditions. For normal-hearing subjects, in their test setup, the average SRT corresponded to −11.2 dB (see Table 1 in [146]), such that $\text{SNR}_{\text{max}} = 11.2\,\text{dB}$ (as SNR_{max} is defined from the perspective of the attended, dominantly amplified speaker), while the correct sentence recognition score and listening effort saturate around 5 dB (see Fig. 1 in [146]). Plugging both values into (2.12) results in $c = 0.65$. We performed an additional subjective listening test on a story stimulus, which confirms that this is also a representative value for connected discourse stimuli (details on this experiment can be found in Appendix 2.F).

Correspondingly, we choose $P_0 = 0.8$, i.e., we require the system to be in the 'comfortable' operating region for 80% of the time. This confidence level yields a good trade-off between a high confidence level and a short enough MESD. Larger confidence levels result in a steep increase in MESD, yielding very high switch durations that are impractical due to an overly strict confidence requirement.

A graphical analysis of the influence of the hyperparameters on the MESD metric is given in Appendix 2.G.

2.3.2 Illustrative example: MESD-based performance evaluation

To illustrate why and how the MESD is useful in the evaluation of AAD algorithms, we apply it to an illustrative example in which we compare two variants of the MMSE decoder for AAD as proposed in [74] and [1], respectively.

Description of the two variants

Given a training set of M data segments, in the first variant of O'Sullivan et al. [74] (also adopted in, for example, [147]), per-segment (corresponding to decision window length τ) decoders are computed, after which the M decoders are averaged to obtain one final decoder. The second variant of Biesmans et al. [1] (also adopted in, e.g., [4, 110, 112]) first averages the M per-segment autocorrelation matrices (or equivalently: the segments are all concatenated) to train a single decoder across all training segments simultaneously. We refer to Section 3.2.1 and Figure 3.3 for more information. Similarly to [1], ℓ_2-norm regularization is added to the former method to avoid overfitting effects due to the small amount of data per decoder. No regularization is needed in the latter method because more data are used to train the decoder [1]. The decoders are again cross-validated in a LOSO-CV manner, and the decoding accuracy is registered per regularization constant (between 10^{-5} and 10^{2}, relative to the mean eigenvalue of the EEG autocorrelation matrix), for every decision window length. Again, the LOSO-CV is performed based on 60 s-segments in order to keep the amount of training data constant for all decision window lengths. These segments are further segmented in shorter windows when the decision window length decreases. Finally, for every window length τ, the maximum decoding accuracy as a function of the regularization parameter is kept. Note that both variants thus use overall the same large amount of training data for each decision window length. When using a shorter decision window length, the decoders do not change for averaging autocorrelation matrices (as all data can be concatenated and the cross-validation (CV) is always done based on 60 s-segments), while for averaging decoders, more decoders are averaged out, each trained with a smaller amount of data.

Subject-averaged comparison

The accuracies are averaged over all 16 subjects, resulting in one performance curve per variant, shown in Figure 2.6 (with the standard deviation indicated by the shading). These performance curves can be interpreted in two ways,

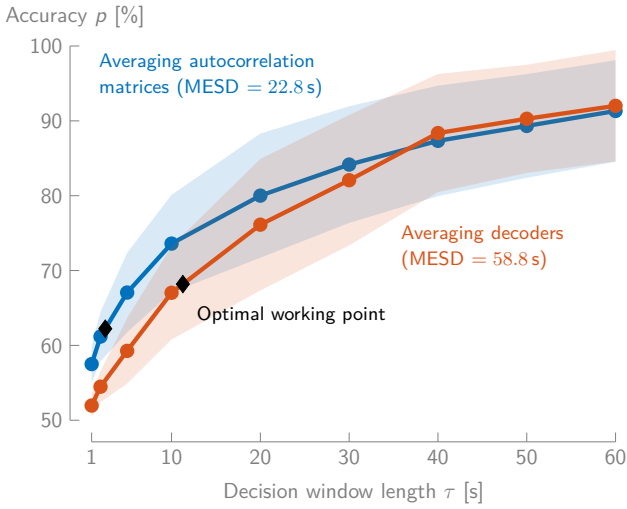


Figure 2.6: The MESD focuses on short decision window lengths as the relevant part of the performance curve, based on which it can be concluded that averaging autocorrelation matrices outperforms averaging of decoders.

leading to two different conclusions depending on where we look. When looking at the region where $\tau > 30\,\text{s}$, one could conclude that both methods perform equally. This is because enough data are still used in the estimation of the per-segment decoders in the method of [74]. However, in the region where $\tau < 30\,\text{s}$, one could conclude that averaging autocorrelation matrices is superior to averaging decoders, although, in total, an equal amount of training data has been used. Here, the loss of information when estimating decoders on short segments is not appropriately compensated by the averaging of a large number of decoders. Based on this analysis, it is not clear what the proper conclusion is, as it is a priori not clear which decision window lengths are more relevant in an AAD-based adaptive gain control system.

Here, the MESD and the corresponding optimal working point can resolve the dilemma mentioned above. Averaging autocorrelation matrices leads to an optimal Markov chain of seven states (optimized as in Sections 2.2.3 and 2.2.4), achieved at optimal working point $(\tau_{\text{opt}}, p_{\text{opt}}) = (2.54\,\text{s}, 0.62)$ where the ESD is minimal. Taking a lower accuracy and shorter decision window length would result in more states (Section 2.2.3), which is not compensated by the shorter decision window length, resulting in a longer ESD. The number of states could be further minimized to five by increasing the decision window length, but the limited decrease in target state k_c from five to four does not compensate enough

for the increase in the decision window length. More details can be found in Appendix 2.H. A different optimal working point is chosen by the MESD metric for the case of averaging decoders, namely $(\tau_{\text{opt}}, p_{\text{opt}}) = (11.28\,\text{s}, 0.68)$, meaning that it chooses for a slower, but more accurate decision process. Nevertheless, the MESD focuses in both cases on the shorter decision window lengths, based on a relevant and realistic criterion, and thus overcomes potential inconclusiveness. It points at averaging autocorrelation matrices as a better way of computing the MMSE decoder, as it allows users to switch almost three times as fast (22.8 s versus 58.8 s).

Statistical comparison

Instead of analyzing a single performance curve by averaging the performance curves per subject, which has the advantage of resulting in a *single*, generally optimal Markov chain and an easy-to-interpret overall picture of the performance, one could also first compute the MESD per subject and perform a comparison based on these MESD performances using proper statistical testing procedures. A key aspect is that the MESD is a single-number metric, thereby allowing to straightforwardly perform statistical tests while inherently taking the accuracy versus decision window length trade-off into account. A paired, one-sided non-parametric Wilcoxon signed-rank test shows that the averaging of decoders significantly performs worse than the averaging of autocorrelation matrices ($n = 16, p\text{-value} < 0.001$). This confirms the conclusion of Biesmans et al. [1], but more firmly, as we focused on the impact on a gain control system instead of arbitrarily choosing a decision window length to evaluate the related accuracy.

2.3.3 Comparison of $\text{ITR}_{\text{W/N}}$ and MESD

Similar to the $\text{ITR}_{\text{W/N}}$, the MESD quantifies the combination of the accuracy and decision time (window length) of an algorithm. As advocated before, the MESD uses, by design, a more relevant criterion to optimize the decision window length and accuracy in the context of AAD algorithms for gain control in hearing devices. By taking the maximum $\text{ITR}_{\text{W/N}}$ (max-ITR) over all decision window lengths, one can define an alternative single-number metric (albeit less interpretable than the MESD). There is, however, a clear quantitative nonlinear relation between both metrics (Figure 2.7a). Both the maximum of the Wolpaw ITR_{W} (2.1) (blue) and Nykopp ITR_{N}[6] (orange) are shown. Per subject, the performances are evaluated using MMSE decoders with averaging of autocorrelation matrices. Due to the nonlinearity, a significant difference

[6]The COCOHA MATLAB toolbox [148] has been used to compute ITR_{N}.

in the max-$\text{ITR}_{\text{W/N}}$ does not automatically imply a significant difference in MESD (and vice versa).

To highlight the differences between both metrics, we also compare the ESD, using the optimal working point based on maximizing the ITR_{W} ($\text{ESD}_{\text{ITR}_{\text{W}}}$), with the MESD (thus minimizing the ESD). Figure 2.7b shows the per-subject differences in switch duration between the original MESD and the $\text{ESD}_{\text{ITR}_{\text{W}}}$ (a similar experiment can be conducted for ITR_{N}). For the majority of the subjects, there is a clear increase in switch duration, which already indicates that the ITR_{W} criterion does not select a working point on the $p(\tau)$-performance curve that leads to an optimal working point for an adaptive gain control system, and, therefore, is not a representative metric to evaluate AAD algorithms in the context of neuro-steered hearing devices. Moreover, several relative differences between subjects have changed, indicating that both criteria fundamentally differ. A non-parametric Wilcoxon signed-rank test ($n = 16, p\text{-value} < 0.001$) confirms that there is significant difference between both switch durations. Optimality in the case of ITR_{W} thus has a fundamentally different and less clear interpretation than in the case of the MESD, which stems from the fact that $\text{ITR}_{\text{W/N}}$ focuses on optimizing ITR as such, which is different from optimizing and stabilizing a gain control system.

In conclusion, it is more relevant to perform (statistical) analysis on a metric that represents a major goal in the context of hearing devices: fast, accurate, and *stable* switching.

2.4 Conclusion

In this chapter, we have developed a new *interpretable* performance metric to evaluate AAD algorithms for AAD-based gain control: the minimal expected switch duration. This metric quantifies the expected time to perform a gain switch after an attention switch of the user in an AAD-based adaptive gain control system, towards a comfort level ($c = 0.65$) that can be maintained for at least 80% of the time. It is based on the concept of the MHT in a Markov chain model, which resulted in a closed-form expression because of the specific line-graph structure. The MESD can be computed from the performance curve of an AAD system by minimizing the ESD over this curve, after designing an optimal Markov chain such that it is for $P_0 = 80\%$ of the time in an optimal operating region. As a by-product, the derivation of the MESD also results in a design methodology for an optimal AAD-based volume control system (see also Section 8.2). The fact that the MESD provides a single-number AAD performance metric that combines accuracy and decision window length and that

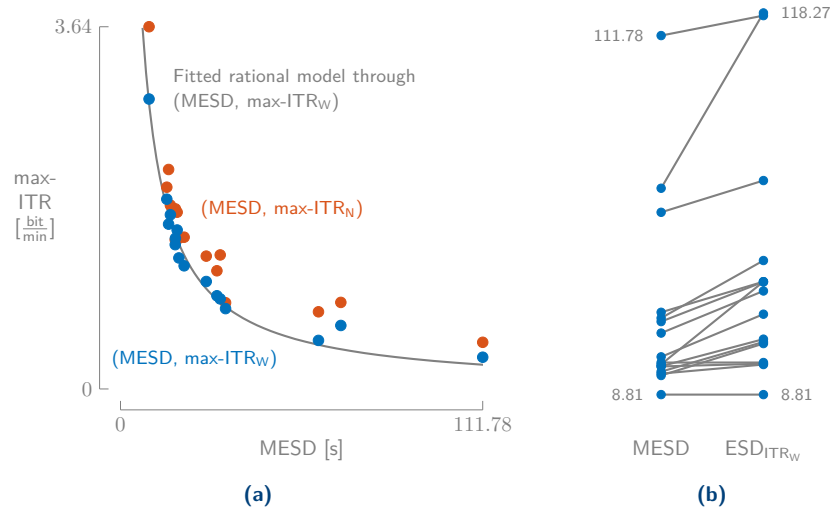


Figure 2.7: **(a)** A fitted rational model max-ITR$_W$(MESD) $= \frac{a}{MESD+b}$ shows that there is a nonlinear relationship between the max-ITR$_W$ and MESD. **(b)** Minizing the ESD (MESD) results in a significantly lower switch duration than optimizing the ESD based on the max-ITR$_W$ (ESD$_{ITR_W}$), indicating that the MESD and ITR$_W$ quantify performance in a fundamentally different way.

is also interpretable and relevant within the context of neuro-steered hearing devices is paramount in order to uniformize the evaluation of AAD algorithms in this context.

Experiments on real EEG and audio data showed that this metric can be used to globally compare AAD systems, both between subjects and between algorithms. Finally, we showed that the MESD is quantitatively related to the ITR$_{W/N}$ but that it uses a fundamentally different criterion that is more relevant in the context of hearing devices.

As a final remark, note that this metric can be easily extended to other BCI applications. In, for example, one-dimensional cursor control using EEG (see, for example, [149]), it could be used to quantify the expected time needed to move a cursor or object from one end to the other end in a stable fashion.

Appendices

2.A The steady-state distribution

The steady-state distribution can be found from the global balance equations and the normalization condition [150]:

$$\begin{cases} \pi(i) = \sum_{l=1}^{N} \pi(l) p_{li}, & \text{(balance equations)} \\ \sum_{l=1}^{N} \pi(l) = 1 & \text{(normalization condition)} \end{cases}$$

where p_{li} corresponds to the transition probability from state l to state i. We can solve the balance equations recursively, starting from $\pi(1)$:

$$\pi(1) = \pi(1)q + \pi(2)q \Leftrightarrow \pi(2) = \frac{1-q}{q}\pi(1) = \frac{p}{q}\pi(1),$$

$$\pi(2) = \pi(1)p + \pi(3)q \Leftrightarrow \pi(3) = (\frac{p}{q^2} - \frac{p}{q})\pi(1) = \frac{p^2}{q^2}\pi(1),$$

$$\vdots$$

By working out the recursion further on and by defining $\frac{p}{q} = r$, it can be seen that:

$$\pi(i) = \frac{p^{i-1}}{q^{i-1}}\pi(1) = r^{i-1}\pi(1), \forall\, i \in \{2, \ldots, N\}.$$

$\pi(1)$ can be found from the normalization condition:

$$\sum_{l=1}^{N} \pi(l) = \pi(1) \sum_{l=1}^{N} r^{l-1} = \frac{r^N - 1}{r - 1}\pi(1) = 1$$

$$\Leftrightarrow \pi(1) = \frac{r-1}{r^N - 1}.$$

2.B The lower bound of the P_0-confidence interval

Starting from (2.3) and using the steady-state distribution in (2.2), we obtain:

$$\frac{r-1}{r^N-1}\sum_{j=\bar{k}}^{N} r^{j-1} \geq P_0 \Leftrightarrow \quad \frac{r-1}{r^N-1}\frac{r^N-r^{\bar{k}-1}}{r-1} \geq P_0$$

$$\Leftrightarrow \quad \frac{r^N-r^{\bar{k}-1}}{r^N-1} \geq P_0$$

Since we assume that $p > 0.5$, it holds that $r > 1$. Hence, both the numerator and denominator are positive. Furthermore, the log-function is a monotonically increasing function, such that it can be applied to both sides without changing the inequality:

$$\frac{r^N-r^{\bar{k}-1}}{r^N-1} \geq P_0 \Leftrightarrow \quad r^N - r^N P_0 + P_0 \geq r^{\bar{k}-1}$$

$$\Leftrightarrow \quad \frac{\log\left(r^N(1-P_0)+P_0\right)}{\log(r)} + 1 \geq \bar{k}.$$

Flooring the last expression leads to (2.4).

2.C Proof of the existence of a solution for N

In this appendix, we prove that there always exists a solution for N such that (2.6) is satisfied. Using (2.4), it can be seen that:

$$\bar{k} - 1 = \left\lfloor \frac{\log\left(r^N(1-P_0)+P_0\right)}{\log(r)} \right\rfloor$$

$$> \frac{\log\left(r^N(1-P_0)+P_0\right)}{\log(r)} - 1 > \frac{\log\left(r^N(1-P_0)\right)}{\log(r)} - 1,$$

such that the constraint (2.6) is always satisfied when

$$\frac{\log\left(r^N(1-P_0)\right)}{\log(r)} - 1 \geq c(N-1). \tag{2.13}$$

Solving for N yields:

$$N \geq 1 - \frac{\log(1-P_0)}{\log(r)(1-c)}.$$

2.D The mean hitting time

The MHT can be found from the recursive definition in [150]:

$$\begin{cases} h_j(i) = 0, & i = j \\ h_j(i) = 1 + \sum_{l=1, l\neq j}^{N} p_{il} h_j(l), & i \neq j \end{cases} \tag{2.14}$$

When $i \leq j$, $h_j(i)$ can be found by starting the recursion in (2.14) with $h_j(1)$:

$$h_j(1) = 1 + h_j(2)p + h_j(1)q \Leftrightarrow h_j(1) = \frac{1}{p} + h_j(2),$$

$$h_j(2) = 1 + h_j(1)q + h_j(3)p \Leftrightarrow h_j(2) = \frac{1}{p} + \frac{q}{p^2} + h_j(3),$$

$$\vdots$$

Eventually, it can be found that:

$$h_j(i) = \frac{1}{p} + \frac{q}{p^2} + \frac{q^2}{p^3} + \cdots + \frac{q^{i-1}}{p^i} + h_j(i+1), \forall\, i \leq j.$$

For $i = j - 1$, this results in:

$$h_j(j-1) = \frac{1}{p} + \frac{q}{p^2} + \frac{q^2}{p^3} + \cdots + \frac{q^{j-2}}{p^{j-1}},$$

where $h_j(j) = 0$ because of (2.14). By propagating the solutions backwards, we find:

$$h_j(i) = (j-i)\sum_{l=1}^{i} \frac{q^{l-1}}{p^l} + \sum_{l=i+1}^{j-1} \frac{(j-l)q^{l-1}}{p^l}.$$

By computing the sums and simplifying the expressions, the expression in (2.8) is found.

2.E Proof $\mathrm{ESD}(p, \tau, N)$ is monotonically non-decreasing with N

We prove that the $\mathrm{ESD}(p, \tau, N)$ in (2.10) is monotonically non-decreasing with N. Starting from (2.9) and using Bayes' law as in Section 2.2.4, the ESD can

be written as:

$$\mathrm{ESD}(p(\tau),\tau,N) = \frac{\tau}{\sum_{l=1}^{k_c-1} r^{-l}} \sum_{i=1}^{k_c-1} r^{-i} h_{k_c}(i). \tag{2.15}$$

$\mathrm{ESD}(p(\tau),\tau,N)$ only implicitly depends on N via $k_c = \lceil c(N-1)+1 \rceil$. We use the notation $k_c(N)$ to explicitly show that k_c is a function of N. Note that $k_c(N+1) \leq k_c(N)+1$ as $k_c(N+1) = \lceil cN \rceil + 1$, while $k_c(N)+1 = \lceil cN+1-c \rceil + 1 \geq \lceil cN \rceil + 1$ as $c \leq 1$. Furthermore, $k_c(N)$ is monotonically increasing with N. This means that there are two possibilities: when $N \to N+1$, then either $k_c \to k_c$ or $k_c \to k_c+1$.

- *Case* $k_c \to k_c$: from (2.15) it can be easily seen that in this case $\mathrm{ESD}(p(\tau),\tau,N+1) = \mathrm{ESD}(p(\tau),\tau,N)$, as $h_{k_c}(i)$ (2.8) only depends on k_c and not explicitly on N.
- *Case* $k_c \to k_c+1$: the proof boils down to proving that:

$$\frac{\sum_{i=1}^{k_c} r^{-i} h_{k_c+1}(i)}{\sum_{l=1}^{k_c} r^{-l}} \geq \frac{\sum_{i=1}^{k_c-1} r^{-i} h_{k_c}(i)}{\sum_{l=1}^{k_c-1} r^{-l}}. \tag{2.16}$$

  If we can show that $\forall\, i \leq k_c - 1$:

$$\frac{r^{-i} h_{k_c+1}(i)}{\sum_{l=1}^{k_c} r^{-l}} \geq \frac{r^{-i} h_{k_c}(i)}{\sum_{l=1}^{k_c-1} r^{-l}}, \tag{2.17}$$

  then (2.16) is true (note that $\frac{r^{-k_c} h_{k_c+1}(k_c)}{\sum_{l=1}^{k_c} r^{-l}} \geq 0$). From (2.8) it can be found that:

$$h_{k_c+1}(i) = h_{k_c}(i) + \frac{1-r^{-k_c}}{2p-1}.$$

  By using the previous result and substituting $h_{k_c}(i)$ with (2.8) in (2.17), we eventually find, after some straightforward algebraic manipulations, that (2.17) boils down to:

$$(1-r^{-k_c})(r^{k_c}-r) \geq (r-1)\left(k_c - i + \frac{p\left(r^{-k_c}-r^{-i}\right)}{2p-1}\right).$$

After some further manipulation and using $r = \frac{p}{1-p}$, this becomes:

$$r^{k_c} - r - 1 \geq (r-1)(k_c - i) - r^{-i+1}. \tag{2.18}$$

We now show that the right-hand side of (2.18) is a decreasing function with $i \leq k_c - 1$. If $f(i) = (r-1)(k_c - i) - r^{-i+1}$, then $f(i+1)$ is equal to:

$$f(i+1) = f(i) + (r^{-i} - 1)(r-1) < f(i).$$

because $r > 1$ and $i \geq 1$. Given that the right-hand side of (2.18) is decreasing with i, we only have to proof (2.18) for $i = 1$:

$$r^{k_c} - r \geq (r-1)(k_c - 1), \tag{2.19}$$

which can be easily proven by induction. For $k_c = 2$ it holds that:

$$r^2 - r \geq r - 1 \Leftrightarrow (r-1)^2 \geq 0,$$

which is evidently true. Now we prove that if (2.19) is true for $k_c = j \geq 2$, then it is also true for $k_c = j + 1$. Setting $k_c = j$, (2.19) can be rewritten as

$$r^j - 1 \geq (r-1)j. \tag{2.20}$$

Furthermore, since $r > 1$, we have that $r^{j+1} - r \geq r^j - 1$ and therefore (2.19) holds for $k_c = j + 1$, using the induction hypothesis in (2.20). This concludes the proof.

2.F Validation of the comfort level c

To validate the chosen c-value ($c = 0.65$) of Section 2.3.1 in case of a (more relevant) connected discourse stimulus instead of standard sentences (as used in Section 2.3.1), we conducted a subjective listening experiment to determine SNR_c. Eight normal-hearing participants, aged between 24 and 29 and with Dutch as their mother tongue, were asked to listen to a mixture of two non-standardized, commercial recordings of stories, 6 min and 34 s long. The stimuli were biologically calibrated. The participants were allowed to adapt the SNR with a slider between 0 and 50 dB and were instructed to select the minimal SNR (between the dominantly amplified speaker and the competing speaker) that still allowed them to comfortably listen to the dominantly amplified speaker for a duration of, for example, 30 min. When they selected a value for SNR_c, they were instructed to listen to the dominantly amplified speaker for three more minutes at their selected SNR_c, where now the previously suppressed speaker is the dominantly amplified speaker. As a validation procedure, the participants self-reported their listening effort, probing the amount of effort required to

understand the loudest speaker. A review of the self-reported listening effort and other methods to assess listening effort can be found in [151]. The minimal reported, maximal reported, and median SNR_c is equal to 4.56 dB, 23.55 dB and 10.89 dB. All reported listening efforts were below 25%.

To obtain the SRT, we used the results from Decruy et al. [152], where they performed a similar experiment (using similar conditions) in an age-matched, normal-hearing group to determine the SRT of connected discourse using the self-assessed Békesy procedure. We use the median SRT $= -16.27$ dB as a value for $\text{SNR}_{\text{max}} = 16.27$ dB. Note that this SRT differs from the one reported in Section 2.3.1, as we are now dealing with a connected discourse instead of standard sentences, while also a different procedure for assessing speech intelligibility has been used.

The resulting c-value is equal to $c = 0.727$ (2.12). Given the large variability on the reported comfort level, we consider this value reasonably close to the proposed value $c = 0.65$, which was calculated based on data from the literature.

2.G The relation between the MESD and the hyperparameters

Figure 2.A shows how the MESD metric depends on the hyperparameters P_0 (the confidence level) and c (the comfort level). The MESDs are based on the results of an MMSE-based decoder with averaging of autocorrelation matrices, described in Section 2.3 and Figure 2.6. When varying one hyperparameter, the other hyperparameters are kept constant at their default values ($P_0 = 0.8, c = 0.65, N_{\text{min}} = 5$). The black diamonds indicate the chosen hyperparameter value. Figure 2.Aa shows that $P_0 = 0.8$ yields a good trade-off between a high confidence level and a short enough MESD. As the MESD has a positive second-order derivative as a function of P_0, an extra amount of confidence results in an even larger increase in MESD, which is why it is important to choose its value as low as possible, without giving too much in on the reliability of the gain control system.

The MESD is a discrete function of the comfort level c (Figure 2.Ab) because of the flooring operation in (2.4). As the lower bound of the P_0-confidence interval needs to be above comfort level c, a higher comfort level results in more states and thus in a higher MESD. Again, higher comfort levels result in a steeper increase in switch duration. The comfort level $c = 0.65$ that resulted from the analysis and experiments in Sections 2.F and 2.3.1 seems to avoid this high cost of extra comfort while assuring, by design, enough comfort for the user.

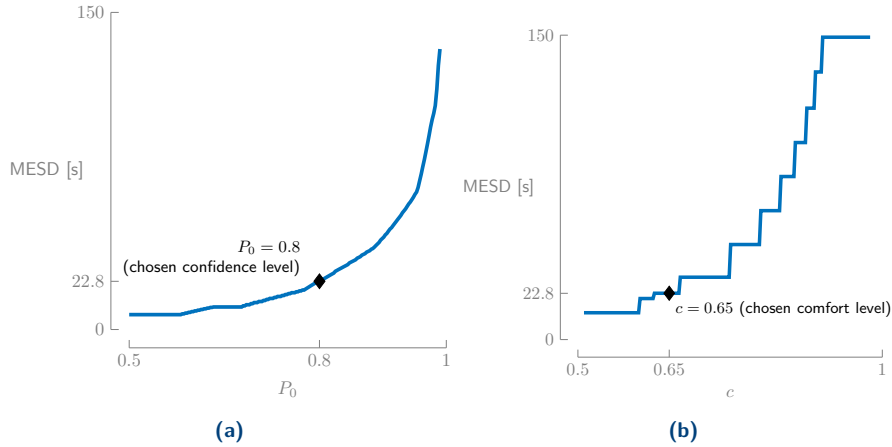


Figure 2.A: The MESD increases as a function of **(a)** the confidence level P_0, with a positive second-order derivative, and **(b)** the comfort level c, in a discrete way, also with an increasing slope. The MESDs are shown for the performance curve of the MMSE-based decoder with averaging of autocorrelation matrices. A diamond (♦) indicates the chosen confidence level and comfort level. When varying a hyperparameter, the other hyperparameter is kept constant at the default value ($c = 0.65, P_0 = 0.8$).

2.H The ESD and number of states as a function of the decision window length

In Section 2.3.2, the MESD has been applied to the performance curve of the MMSE-based decoder with averaging of autocorrelation matrices versus averaging decoders (Figure 2.6). We mentioned that the optimal MESD for averaging autocorrelation matrices is obtained at a Markov chain of seven states. Figure 2.B shows the optimal number of states $\hat{N}_\tau$, target state k_c, and the ESD (at the optimal number of states $\hat{N}_\tau$) per decision window length. It is over this curve that the ESD is minimized to obtain the MESD (Section 2.2.5 and Algorithm 1).

In Figure 2.B, it can be seen that when $\hat{N}_\tau$ remains constant, the ESD increases almost linearly with decision window length τ. In (2.10), when the number of states N and thus target state k_c remains constant, it appears that the step time τ is the dominant factor over the variation in transition probability p. This implies that the interesting decision window lengths coincide with changes in the number of states. Relative to $\hat{N}_\tau = 7$ at the MESD, an increase in decision window length results in a decrease of $\hat{N}_\tau$ to five. However, the target state k_c only decreases from five to four, such that the drop in ESD around $\approx 6\,\text{s}$ is not large enough to decrease below the minimal ESD for seven states.

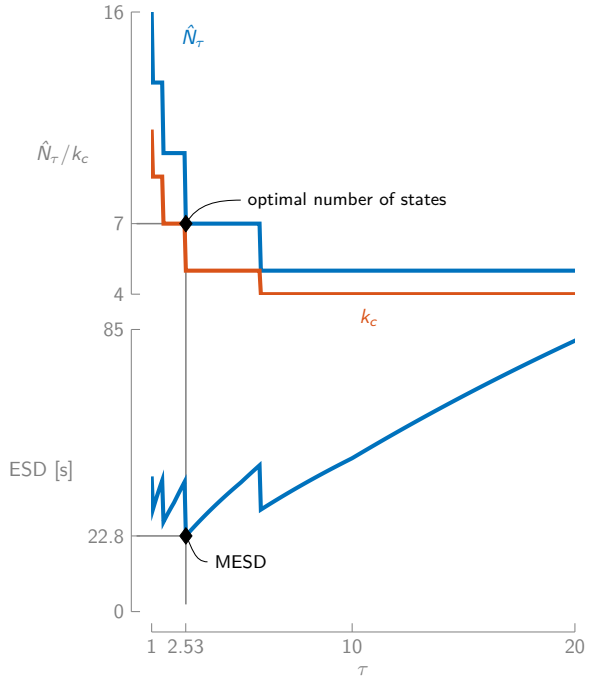


Figure 2.B: The optimal number of states $\hat{N}_\tau$ and corresponding target state k_c decrease as a function of the decision window length τ. The minimal ESD (MESD) depends both on the optimal number of states (via the AAD accuracy) and the decision window length.

When decreasing τ, $\hat{N}_\tau$ and k_c steeply increase because of the steep decrease in accuracy (Figure 2.6), which is not sufficiently compensated by the small decrease in step time τ. The AAD accuracy p (depending on decision window length τ) thus mainly plays a role in determining the optimal number of states $\hat{N}_\tau$ via the design constraints (Section 2.2.3), which is the first step in optimizing the ESD (Section 2.2.5 and Algorithm 1), while the transition points of $\hat{N}_\tau$ are most interesting for minimizing the ESD to obtain the MESD, as the ESD almost linearly increases with τ for a constant $\hat{N}_\tau$.

3 | A comparative review study of AAD algorithms

This chapter is based on S. Geirnaert, S. Vandecappelle, E. Alickovic, A. de Cheveigné, E. C. Lalor, B. T. Meyer, S. Miran, T. Francart, and A. Bertrand, "Electroencephalography-Based Auditory Attention Decoding: Toward Neurosteered Hearing Devices," *IEEE Signal Processing Magazine*, vol. 38, no. 4, pp. 89-102, 2021. Some parts of the introduction and the discussions in Section IV of the paper have been moved to Chapter 1. Two extra figures (Figures 3.3 and 3.4) and an extra discussion on newer AAD algorithms (Sections 3.2.1 and 3.2.2) have been added.

ABSTRACT | Recent neuroscientific advances have shown that it is possible to determine the focus of auditory attention from non-invasive neurorecording techniques, such as EEG. Based on these new insights, a multitude of AAD algorithms have been proposed, which could, combined with the appropriate speaker separation algorithms and miniaturized EEG sensor devices, lead to so-called neuro-steered hearing devices to assist people suffering from hearing impairment in 'cocktail party' scenarios (Section 1.7). In this chapter, we provide a broad review and a statistically grounded comparative study of EEG-based AAD algorithms on different datasets. Based on the MESD performance metric of Chapter 2, we conclude that even the best linear SR method is too slow for practical purposes. Decoding the spatial focus of auditory attention shows to be a promising alternative. Furthermore, it turns out to be hard to replicate the results of nonlinear (DNN-based) algorithms on multiple independent AAD datasets. Lastly, we give an outlook on the main signal processing-related challenges in the field, of which two are addressed in Parts II and III.

3.1 Introduction

As explained in Chapter 1, recent neuroscientific insights on how the brain synchronizes with the speech envelope have laid the groundwork for a new strategy to tackle the AAD problem: extracting attention-related information directly from the origin, i.e., the brain. Following these breakthroughs, several different (EEG-based) AAD algorithms have been proposed. O'Sullivan et al. proposed in [74] a first successful speech-based AAD algorithm using unaveraged single-trial EEG data. The main idea of [74] is to decode the attended speech envelope from a multi-channel EEG recording using a neural decoder and correlate the decoder output with the speech envelope of each speaker. Following this seminal work, many new AAD algorithms have been developed [1,44,119,136,138–141,143,147,153–156]. As explained in Section 1.7, these advances could, in combination with effective speaker separation algorithms and relying on rapidly evolving improvements in miniaturization and wearability of EEG sensors, lead to a new assistive solution for the hearing impaired: a *neuro-steered hearing device* (Figure 1.6).

Despite the large variety in AAD algorithms, an objective and transparent comparative study has not been performed to date, making it hard to identify which strategies are most successful. In this chapter, we will briefly review different types of AAD algorithms and their most common instances, and provide an objective and quantitative comparative study using two independent, publicly available datasets (Dataset A and Dataset B). This comparative study has been reviewed and endorsed by the author(s) of the original papers in which these algorithms were proposed to ensure fairness and correctness.

3.2 Review of AAD algorithms

In this section, we provide a brief overview of various AAD algorithms. This comparative study includes only papers published before the year 2020, when the paper on which this chapter is based was conceptualized. However, since this field is quickly progressing and several new papers have appeared since the conceptualization of this article, the reader is encouraged to look up new AAD algorithms (and extensions thereof) and compare them with the presented methods. Therefore, at the end of Sections 3.2.1 and 3.2.2, we have added a few of these newer papers, which are, however, not further used in this comparative study.

For the sake of an easy exposition, we assume that there are only two speakers (one attended and one unattended speaker), although all algorithms can be

generalized to more than two speakers. In the remainder of this chapter, we also make abstraction of the speaker separation and denoising block in a neuro-steered hearing device (Figure 1.6) and assume that the AAD block has direct access to the envelopes of the original unmixed speech sources, as often done in the AAD literature. An extended discussion on the combination of both blocks can be found in Section 1.7.2.

Most AAD algorithms adopt a *stimulus reconstruction* (SR) approach (also known as backward modeling or decoding). In this strategy, a multi-input single-output (MISO) neural decoder is applied to all EEG channels to reconstruct the attended speech envelope. This neural decoder is pre-trained to optimally reconstruct the attended speech envelope from the EEG data while blocking other (unrelated) neural activity. It is in this training procedure that most AAD algorithms differ. The reconstructed speech envelope is afterwards correlated with the speech envelopes of all speakers, after which the one with the highest Pearson correlation coefficient is identified as the attended speaker (Figure 3.2). This correlation coefficient is estimated over a window of τ seconds, which is referred to as the *decision window length*, corresponding to the amount of EEG data used in each decision on the attention. Typically, the AAD accuracy strongly depends on this decision window length because the Pearson correlation estimates are very noisy due to the low SNR of the output signal of the neural decoder.

Alternatively, the neural response in each EEG channel can be predicted from the speech envelopes via an encoder (also known as forward modeling or encoding) and can then be correlated with the measured EEG [125, 142, 147]. When the encoder is linear, this corresponds to estimating impulse responses (also called temporal response functions (TRF)) between the speech envelope(s) and the recorded EEG signals. For AAD, backward MISO decoding models have been demonstrated to outperform forward encoding models [142, 147], as the former can exploit the spatial coherence across the different EEG channels at its input. In this comparative study, we thus only focus on backward AAD models, except for the canonical correlation analysis (CCA) algorithm (Section 3.2.1), which combines both a forward and backward approach.

Due to the emergence of deep learning methods, a third approach has become popular: *direct classification* [119,136]. In this approach, the attention is directly decoded in an end-to-end fashion, without explicitly reconstructing the speech envelope.

The decoder models are typically trained in a supervised fashion, which means that the attended speaker must be known for each data point in the training set. This requires collecting 'ground-truth' EEG data during a dedicated experiment in which the subject is asked to pay attention to a predefined speaker in a speech

mixture. The models can be trained either in a *subject-specific* fashion (based on EEG data from the actual subject under test) or in a *subject-independent* fashion (based on EEG data from other subjects than the subject under test). The latter leads to a universal (subject-independent) decoder, which has the advantage that it can be applied to new subjects without the need to go through such a tedious ground-truth EEG data collection for every new subject. However, since each person's brain responses are different, the accuracy achieved by such universal decoders is typically lower [74]. In this chapter, we only consider subject-specific decoders, which allows achieving better accuracies, as they are tailored to the EEG of the specific end-user. Transfer learning techniques, which are becoming popular in the BCI field [38], may close the gap between subject-specific and subject-independent models, although this remains to be researched in the context of AAD (see also Part II).

Figure 3.1 depicts a complete overview and classification of all algorithms included in our comparative study, discriminated based on their fundamental properties. In the following sections, we distinguish between linear and nonlinear algorithms.

3.2.1 Linear methods

All linear methods included in this study, which differ in the features shown in the linear branch of Figure 3.1, adopt the so-called SR framework (Figure 3.2). This boils down to applying a linear time-invariant spatio-temporal filter $d_c(l)$ on the C-channel EEG $x_c(t)$ to reconstruct the attended speech envelope $s_\mathrm{a}(t)$:

$$\hat{s}_\mathrm{a}(t) = \sum_{c=1}^{C} \sum_{l=0}^{L-1} d_c(l) x_c(t+l), \tag{3.1}$$

where c is the channel index, ranging from 1 to C, and l is the time lag index, ranging from 0 to $L-1$, with L the per-channel filter length. The corresponding MISO filter is anti-causal, as the brain responds to the stimulus, such that only future EEG time samples can be used to reconstruct the current stimulus sample. Equation (3.1) can be rewritten as

$$\hat{s}_\mathrm{a}(t) = \mathbf{d}^\mathrm{T}\mathbf{x}(t),$$

using $\mathbf{d} \in \mathbb{R}^{LC\times 1}$, collecting all decoder coefficients for all time lags and channels, and

$$\mathbf{x}(t) = \begin{bmatrix}\underline{\mathbf{x}}_1(t)^\mathrm{T} & \underline{\mathbf{x}}_2(t)^\mathrm{T} & \cdots & \underline{\mathbf{x}}_C(t)^\mathrm{T}\end{bmatrix}^\mathrm{T} \in \mathbb{R}^{LC\times 1},$$

$$\text{with } \underline{\mathbf{x}}_c(t) = \begin{bmatrix}x_c(t) & x_c(t+1) & \cdots & x_c(t+L-1)\end{bmatrix}^\mathrm{T}.$$

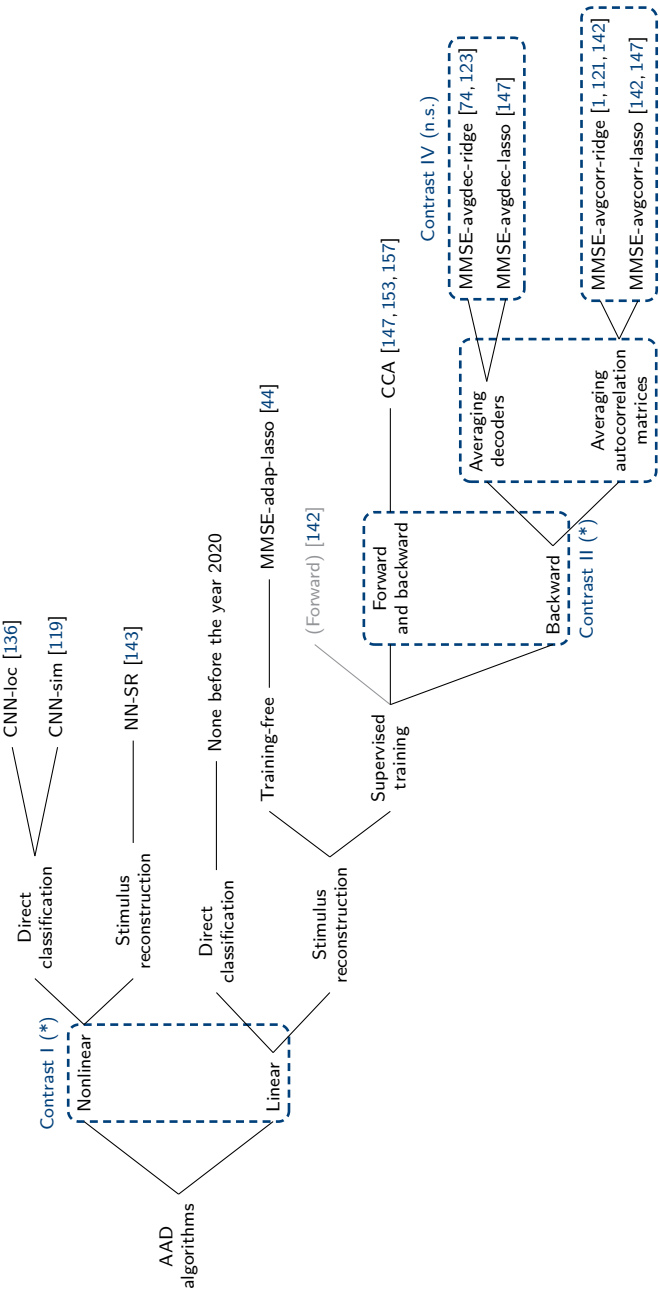


Figure 3.1: The AAD algorithms included in this comparative study (except for the forward models; see the introduction of Section 3.2) and the planned contrasts in the statistical analysis. (*) indicates a significant difference ($p < 0.05$), while (n.s.) indicates a non-significant difference (see Section 3.3.1 for more details).

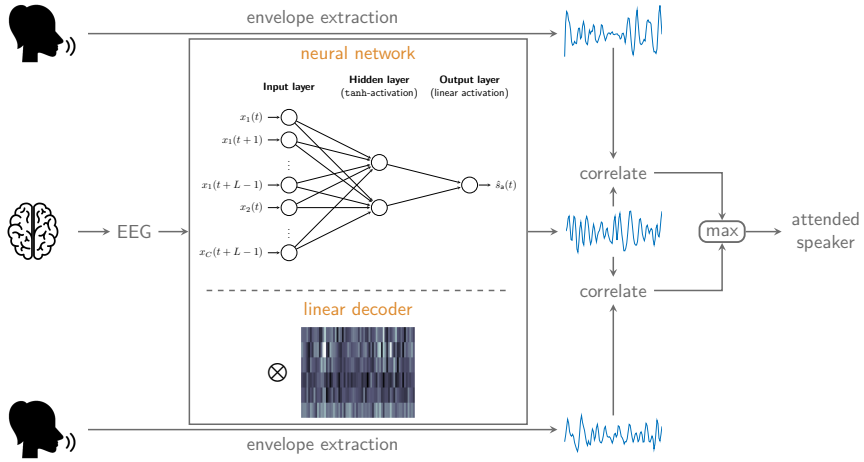


Figure 3.2: A conceptual overview of the linear SR algorithm and NN-SR.

The same indexing holds for the decoder $\mathbf{d}$.

In the following three sections, we introduce the different linear methods included in this study. These linear methods, which are all correlation-based, can be extended to more than two competing speakers by simply correlating the reconstructed speech envelope with all additional speech envelopes of the individual competing speakers and taking the maximum.

Supervised minimum mean-squared error backward modeling (MMSE)

The most basic way of training the decoder, first presented in the EEG-based AAD-context in [74], is by minimizing the mean squared error (MMSE) between the actual attended envelope and the reconstructed envelope. In [1], it is shown that minimizing the mean squared error is equivalent to maximizing the Pearson correlation coefficient between the reconstructed and attended speech envelope. Using sample estimates, assuming that there are T samples available, the MMSE-based formulation becomes equivalent to the least-squares (LS) formulation:

$$\hat{\mathbf{d}} = \underset{\mathbf{d}}{\operatorname{argmin}} \; ||\mathbf{s}_\mathrm{a} - \mathbf{X}\mathbf{d}||_2^2 \,, \tag{3.2}$$

with $\mathbf{X} = \begin{bmatrix}\mathbf{x}(0) & \cdots & \mathbf{x}(T-1)\end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^{T \times LC}$ and $\mathbf{s}_{\mathrm{a}} = \begin{bmatrix}s_{\mathrm{a}}(0) & \cdots & s_{\mathrm{a}}(T-1)\end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^{T \times 1}$. The normal equations lead to the solution

$$\hat{\mathbf{d}} = (\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{s}_{\mathrm{a}}.$$

The first factor corresponds to an estimation of the autocorrelation matrix

$$\hat{\mathbf{R}}_{xx} = \frac{1}{T}\sum_{t=0}^{T-1}\mathbf{x}(t)\mathbf{x}(t)^{\mathrm{T}} \in \mathbb{R}^{LC \times LC},$$

while the second factor corresponds to the crosscorrelation vector

$$\hat{\mathbf{r}}_{xs_{\mathrm{a}}} = \frac{1}{T}\sum_{t=0}^{T-1}\mathbf{x}(t)s_{\mathrm{a}}(t) \in \mathbb{R}^{LC \times 1}.$$

To avoid overfitting, two types of regularization are used in AAD literature: ridge regression/ℓ_2-norm regularization and ℓ_1-norm/sparse regularization, also known as the least absolute shrinkage and selection operator (lasso). The corresponding cost functions are shown in Table 3.1, where the regularization hyperparameter λ is defined relative to $z = \frac{\mathrm{Tr}(\mathbf{X}^{\mathrm{T}}\mathbf{X})}{LC}$ (for ridge regression)/$q = ||\mathbf{X}^{\mathrm{T}}\mathbf{s}_{\mathrm{a}}||_{\infty}$ (for lasso). Similar to [147], we here use the alternating direction method of multipliers (ADMM) to iteratively obtain the solution of the lasso problem. The optimal value λ can be found using a CV scheme. Other regularization methods, such as Tikhonov regularization, have been proposed as well [142].

Assume a given training set consisting of K data segments of a specific length T. These segments can either be constructed artificially by segmenting a continuous recording (usually for the sake of CV) or correspond to different experimental trials (potentially from different subjects, for example, when training a subject-independent decoder). There exist various flavors of combining these different segments in the process of training a decoder. As suggested in the seminal paper of O'Sullivan et al. [74], decoders $\mathbf{d}_k$ can be trained per segment k, after which all decoders are averaged to obtain a single, final decoder $\mathbf{d}$ (Figure 3.3). Biesmans et al. proposed in [1] an alternative scheme (also adopted in, e.g., [110–112,121,134,158]), where, instead of estimating a decoder per segment separately, the loss function (3.2) (with regularization) is minimized over all K segments at once. As can be seen from the solution in Table 3.1, this is equivalent to first estimating the autocorrelation matrix and crosscorrelation vector via averaging the sample estimates per segment, whereafter one decoder is computed (Figure 3.3). It is easy to see that this is mathematically equivalent to concatenating all the data in one big matrix $\mathbf{X} \in \mathbb{R}^{KT \times LC}$ and vector $\mathbf{s}_{\mathrm{a}} \in \mathbb{R}^{KT \times 1}$ and computing the decoder straightforwardly. As such, it is an example of the *early integration* paradigm, versus *late integration* in the former

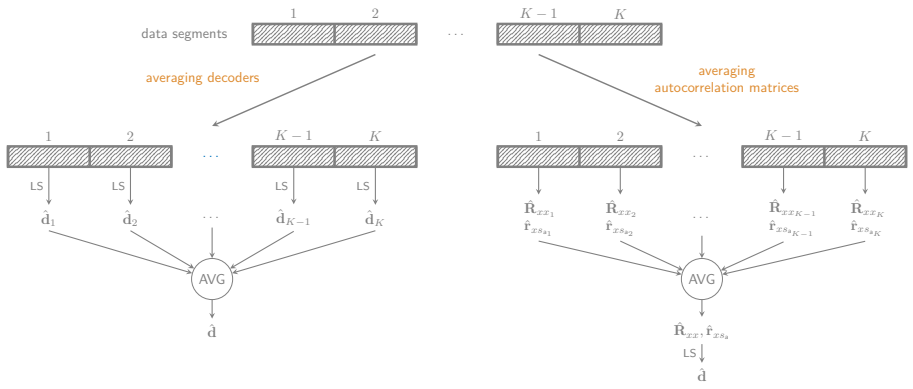


Figure 3.3: When averaging decoders, a decoder per segment is computed, whereafter all decoders are averaged to obtain a single final decoder. One final decoder is straightforwardly computed when averaging autocorrelation matrices, as it is equivalent to concatenating all the data. 'LS' = least-squares, 'AVG' = averaging.

case when averaging K separate decoders. Both versions are included in our comparative study.

Table 3.1 shows the four different flavors of the MMSE/LS-based decoder that were proposed as different AAD algorithms in [1, 74, 147], adopting different regularization techniques (ℓ_2/ℓ_1-regularization) or ways to train the decoder (averaging decoders or correlation matrices).

Canonical correlation analysis (CCA)

CCA to decode the auditory brain has been proposed by de Cheveigné et al. [153] and Dmochowski et al. [157]. It has been applied to the AAD problem for the first time by Alickovic et al. [147]. CCA combines a spatio-temporal backward (decoding) model $\mathbf{w}_x \in \mathbb{R}^{LC\times 1}$ on the EEG and a temporal forward (encoding) model $\mathbf{w}_{s_\mathrm{a}} \in \mathbb{R}^{L_\mathrm{a}\times 1}$ on the speech envelope (Figure 3.4), with L_a the number of filter taps of the encoding filter. In this sense, CCA differs from the previous approaches, which were all different flavors of the same MMSE/LS-based decoder. In CCA, both the forward and backward model are estimated *jointly* such that their outputs are maximally correlated:

$$\max_{\mathbf{w}_x,\mathbf{w}_{s_\mathrm{a}}} \frac{\mathbb{E}\{(\mathbf{w}_x^\mathrm{T}\mathbf{x}(t))\left(\mathbf{w}_{s_\mathrm{a}}^\mathrm{T}\mathbf{s}_\mathrm{a}(t)\right)\}}{\sqrt{\mathbb{E}\{(\mathbf{w}_x^\mathrm{T}\mathbf{x}(t))^2\}}\sqrt{\mathbb{E}\{\left(\mathbf{w}_{s_\mathrm{a}}^\mathrm{T}\mathbf{s}_\mathrm{a}(t)\right)^2\}}} = \max_{\mathbf{w}_x,\mathbf{w}_{s_\mathrm{a}}} \frac{\mathbf{w}_x^\mathrm{T}\mathbf{R}_{xs_\mathrm{a}}\mathbf{w}_{s_\mathrm{a}}}{\sqrt{\mathbf{w}_x^\mathrm{T}\mathbf{R}_{xx}\mathbf{w}_x}\sqrt{\mathbf{w}_{s_\mathrm{a}}^\mathrm{T}\mathbf{R}_{s_\mathrm{a}s_\mathrm{a}}\mathbf{w}_{s_\mathrm{a}}}}, \tag{3.3}$$

Method	Cost function	Solution
Ridge regression + averaging of decoders [74] (MMSE-avgdec-ridge)	$\hat{\mathbf{d}}_k = \underset{\mathbf{d}}{\text{argmin}} \ \|\|\mathbf{s}_{\mathrm{a}_k} - \mathbf{X}_k\mathbf{d}\|\|_2^2 + \lambda z_k \, \|\|\mathbf{d}\|\|_2^2$	$\hat{\mathbf{d}}_k = (\mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k + \lambda z_k \mathbf{I})^{-1}\mathbf{X}_k^{\mathrm{T}}\mathbf{s}_{\mathrm{a}_k}$ and $\hat{\mathbf{d}} = \frac{1}{K}\sum_{k=1}^{K}\hat{\mathbf{d}}_k$
Lasso + averaging of decoders [147] (MMSE-avgdec-lasso)	$\hat{\mathbf{d}}_k = \underset{\mathbf{d}}{\text{argmin}} \ \|\|\mathbf{s}_{\mathrm{a}_k} - \mathbf{X}_k\mathbf{d}\|\|_2^2 + \lambda q_k \, \|\|\mathbf{d}\|\|_1$	ADMM and $\hat{\mathbf{d}} = \frac{1}{K}\sum_{k=1}^{K}\hat{\mathbf{d}}_k$
Ridge regression + averaging of correlation matrices [1] (MMSE-avgcorr-ridge)	$\hat{\mathbf{d}} = \underset{\mathbf{d}}{\text{argmin}} \ \sum_{k=1}^{K} \|\|\mathbf{s}_{\mathrm{a}_k} - \mathbf{X}_k\mathbf{d}\|\|_2^2 + \lambda z \, \|\|\mathbf{d}\|\|_2^2$	$\hat{\mathbf{d}} = \left(\sum_{k=1}^{K}\mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k + \lambda z\mathbf{I}\right)^{-1}\sum_{k=1}^{K}\mathbf{X}_k^{\mathrm{T}}\mathbf{s}_{\mathrm{a}_k}$
Lasso + averaging of correlation matrices [147] (MMSE-avgcorr-lasso)	$\hat{\mathbf{d}} = \underset{\mathbf{d}}{\text{argmin}} \ \sum_{k=1}^{K} \|\|\mathbf{s}_{\mathrm{a}_k} - \mathbf{X}_k\mathbf{d}\|\|_2^2 + \lambda q \, \|\|\mathbf{d}\|\|_1$	ADMM

Table 3.1: A summary of the supervised backward MMSE-decoder and its different flavors.

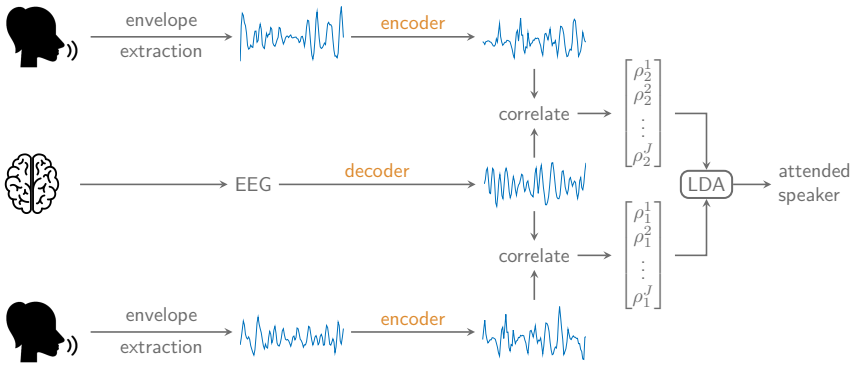


Figure 3.4: In the CCA algorithm, both backward, decoding filters and forward, encoding filters are applied to the EEG and audio envelopes, respectively, to maximize the correlations between the outputs of the filters. An LDA classifier is used to determine the attended speaker.

where $\mathbf{s}_\mathrm{a}(t) = \begin{bmatrix} s_\mathrm{a}(t) & s_\mathrm{a}(t-1) & \cdots & s_\mathrm{a}(t-L_\mathrm{a}+1) \end{bmatrix}^\mathrm{T} \in \mathbb{R}^{L_\mathrm{a} \times 1}$. As opposed to the EEG filter $\mathbf{w}_x$, the audio filter $\mathbf{w}_{s_\mathrm{a}}$ is a causal filter, as the stimulus precedes the brain response. The solution of the optimization problem in (3.3) can be easily retrieved by solving a generalized eigenvalue decomposition (details in [1, 147]).

In CCA, the backward model $\mathbf{w}_x$ and forward model $\mathbf{w}_{s_\mathrm{a}}$ are extended to a set of J filters $\mathbf{W}_x \in \mathbb{R}^{LC \times J}$ and $\mathbf{W}_{s_\mathrm{a}} \in \mathbb{R}^{L_\mathrm{a} \times J}$ for which the outputs are maximally correlated, but mutually uncorrelated (the J outputs of $\mathbf{W}_x^\mathrm{T}\mathbf{x}(t)$ are uncorrelated to each other and the J outputs of $\mathbf{W}_{s_\mathrm{a}}^\mathrm{T}\mathbf{s}_\mathrm{a}(t)$ are uncorrelated to each other). There are now thus J Pearson correlation coefficients between the outputs of the J backward and forward filters (also called canonical correlation coefficients), which are collected in the vector $\boldsymbol{\rho}_i \in \mathbb{R}^{J \times 1}$ for speaker i, whereas before, there was only one per speaker. Furthermore, because of the way CCA constructs the filters, it can be expected that the first components are more important than the later ones. To find the optimal way of combining the canonical correlation coefficients, a linear discriminant analysis (LDA) classifier can be trained, as proposed in [153] (Figure 3.4). To generalize the maximization of the correlation coefficients of the previous AAD algorithms (which is equivalent to taking the sign of the difference of the correlation coefficients of both speakers), we propose here to construct a feature vector $\mathbf{f} \in \mathbb{R}^{J \times 1}$ by subtracting the canonical correlation vectors: $\mathbf{f} = \boldsymbol{\rho}_1 - \boldsymbol{\rho}_2$, and classify $\mathbf{f}$ with an LDA classifier. As proposed in [153], we use principal component analysis (PCA) as a preprocessing

step on the EEG to reduce the number of parameters. In fact, this is a way of regularizing CCA and can, as such, be viewed as an alternative to the regularization techniques proposed in other methods.

Training-free MMSE-based with lasso (MMSE-adap-lasso)

Miran et al. proposed in [44] a fundamentally different AAD algorithm. In this comparative study, all other AAD algorithms are *supervised*, batch-trained algorithms, which have a separate training and testing stage. First, the decoders need to be trained in a supervised manner using a large amount of ground-truth data, after which they can be applied to new test data. In practice, this necessitates a (potentially cumbersome) a priori training stage, resulting in a fixed decoder, which does not adapt to the non-stationary EEG signal characteristics, e.g., due to changing conditions or brain processes. The AAD algorithm in [44] aims to overcome these issues by adaptively estimating a decoder for each speaker and simultaneously using the outputs to decode the auditory attention. Therefore, this training-free AAD algorithm has the advantage of adapting the decoders to non-stationary signal characteristics, however, without requiring the same large amount of ground-truth data as the supervised AAD algorithms.

In this comparative study, we have removed the state-space and dynamic decoder estimation modules to produce a single decision for each decision window, similar to the other AAD algorithms in this study (the full description of the algorithm can be found in [44]). This leads to the following formulation:

$$\hat{\mathbf{d}}_{i,l} = \underset{\mathbf{d}}{\operatorname{argmin}} \; ||\mathbf{s}_{i,l} - \mathbf{X}_l \mathbf{d}||_2^2 + \lambda q \, ||\mathbf{d}||_1 \,, \tag{3.4}$$

for the i^{th} speaker in the l^{th} decision window. In the context of AAD, for every new incoming window of τ seconds of EEG and audio data, two decoders are thus estimated (one for each speaker). As an attentional marker, these estimated decoders could be applied to the EEG data $\mathbf{X}_l$ of the l^{th} decision window to compute the correlation with their corresponding stimulus envelopes. In addition, Miran et al. [44] propose to identify the attended speaker by selecting the speaker with the largest ℓ_1-norm of its corresponding decoder $\hat{\mathbf{d}}_{i,l}$, as the attended decoder should exhibit more sparse, significant peaks, while the unattended decoder should have smaller, randomly distributed coefficients. The regularization parameter is again being cross-validated and defined in the same way as for MMSE-avgdec/corr-lasso. To prevent overfitting by decreasing the number of parameters to be estimated, Miran et al. [44] have proposed to a priori select a subset of EEG channels. In our comparative study, we also adopt this approach and select the same channels.

While we do not adopt the extra post-processing state-space modeling steps from [44, 159] in order to focus on the core AAD algorithm, it is noted that such an extra smoothing step, which also takes previous and/or future decisions into account, can effectively enhance the performance of most AAD algorithms, albeit at the cost of a potential algorithmic delay in the detection of attention switches [44].

Other linear methods

Since the writing of the paper on which this chapter is based, other linear AAD methods have been proposed which are not included in the comparative study. For example, in Chapter 6, we will propose a novel CSP-based method to decode the spatial focus of auditory attention [154]. Wang et al. proposed in [160] to train different MMSE-based backward decoders to reconstruct the attended speech envelope based on the root-mean-square level of the input segments, which resulted in a significant improvement for shorter decision windows. Kuruvila et al. proposed in [156] a sequential linear MMSE-based dynamic estimation of TRFs, followed by the extraction of the N1-P2 peaks of the estimated TRFs as features and a linear support vector machine (SVM) classification. While this inherent temporal smoothing of AAD decision led to higher performances, it also resulted in substantial delays in detecting an attention switch.

3.2.2 Nonlinear methods

Nonlinear methods based on (deep) neural networks (DNNs) can adopt a SR approach similar to the linear methods [143], but can also classify the attended speaker directly from the EEG and the audio (referred to as direct classification) [119, 136]. However, these nonlinear methods are more vulnerable to overfitting [136], particularly for the small-size datasets typically collected in AAD research. In order to appreciate the differences between current neural network-based AAD approaches, Figures 3.2, 3.5 and 3.6 show a conceptual overview of the different strategies and network topologies of the presented nonlinear methods. We give a concise description of each architecture below but refer to the respective papers for further details.

Fully connected SR neural network (NN-SR)

de Taillez et al. proposed in [143] a fully-connected neural network with a single hidden layer that reconstructs the envelope based on a window of EEG. As

shown in Figure 3.2, the input layer consists of LC neurons (similar to a linear decoder), with L the number of time lags and C the number of EEG channels. These neurons are connected to a hidden layer with two neurons and a tanh activation function. These two neurons are then finally combined into a single output neuron that uses a linear activation function and outputs one sample of the reconstructed envelope. As such, the network has $2 \times (LC + 1)$ (hidden layer) $+2 + 1$ (output layer) ≈ 3446 trainable parameters.

The network is trained to minimize $1 - \rho(\hat{\mathbf{s}}_\mathrm{a}, \mathbf{s}_\mathrm{a})$ over a segment of M training samples (within this segment the neural network coefficients are kept constant), with $\rho(\cdot)$ the Pearson correlation coefficient, and $\hat{\mathbf{s}}_\mathrm{a}, \mathbf{s}_\mathrm{a} \in \mathbb{R}^{M \times 1}$ the reconstructed and attended envelope, respectively. Minimizing this cost function is equivalent to maximizing the Pearson correlation coefficient between the reconstructed and attended speech envelope, similar to linear SR approaches. The trained network is then used as a decoder, where the speech envelope showing the highest correlation with the decoder output is selected as the attended speaker. This algorithm can be extended to more than two competing speakers similar to the other linear SR algorithms.

Convolutional neural network to compute similarity between EEG and stimulus (CNN-sim)

Ciccarelli et al. proposed in [119] a convolutional neural network (CNN) to directly compare a $C \times \tau$ EEG window with a $1 \times \tau$ speech envelope. This network is trained to output a similarity score $\in [0, 1]$ (similar to the correlation coefficient used in other approaches) between the EEG and the speech envelope using a binary cross-entropy cost function. The speech envelope that, according to the trained CNN, is most similar to the EEG is then identified as the attended speaker. This approach can be easily extended to more than two speakers by computing a similarity score for each speaker and taking the maximum over all scores to identify the attended speaker.

The network, depicted in Figure 3.5, consists of two convolutional layers, with max-pooling (stride two) after the first convolutional layer, and four fully connected (FC) layers. In total, this network has $64 \times (C + 1) \times L_1$ (first convolutional layer) $+2 \times 64 \times L_2$ (second convolutional layer) $+200 \times 3$ (first FC layer) $+200 \times 201$ (second FC layer) $+100 \times 201$ (third FC layer) $+101$ (fourth FC layer) ≈ 69070 trainable parameters. An exponential linear unit is used as a nonlinear activation function. Furthermore, drop-out is used as a regularization technique to prevent overfitting in the FC layers, while also batch normalization is used throughout the network. Details about the training can be found in [119].

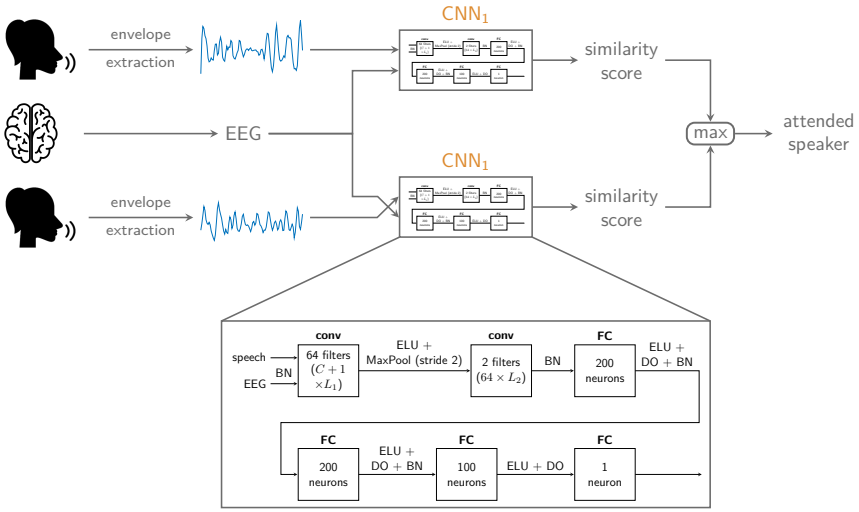


Figure 3.5: A conceptual overview and the network topology of the CNN-sim algorithm. 'conv' = convolutional layer, 'FC' = fully connected layer, 'BN' = batch normalization, 'ELU' = exponential linear unit, 'DO' = drop-out, 'MaxPool' = max-pooling.

Convolutional neural network to determine spatial focus of attention (CNN-loc)

Vandecappelle et al. proposed in [136] a CNN to determine the spatial/directional focus of attention (e.g., left or right), solely based on the EEG. This is a fundamentally different approach to tackle the AAD problem, which has the advantage of not requiring the individual speech envelopes. Furthermore, it avoids the requirement to estimate a correlation coefficient over a relatively long decision window length as in all aforementioned algorithms, thereby avoiding large algorithmic delays.

This CNN determines the spatial focus of attention, starting from a $C \times \tau$ EEG window. As shown in Figure 3.6, it consists of one convolutional layer and two FC layers. The convolutional layer consists of five spatio-temporal filters, with lags L similar to before, each outputting a one-dimensional time series of length τ, on which a rectified linear unit activation function is applied. Afterwards, an average pooling layer condenses each output series into a scalar, leading to a five-dimensional vector. This vector is then used as an input for two FC layers, the first one consisting of five neurons with a sigmoid activation function and the output layer consisting of two neurons and a softmax layer. In total, this

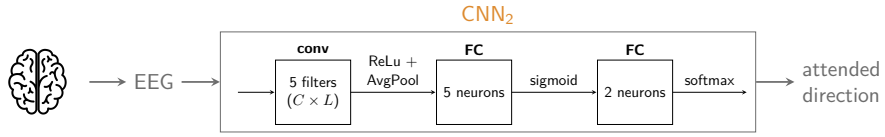


Figure 3.6: A conceptual overview and the network topology of the CNN-loc algorithm. 'conv' = convolutional layer, 'FC' = fully connected layer, 'ReLu' = rectified linear unit, 'AvgPool' = average pooling.

network has $5 \times C \times L$ (convolutional layer) $+5 \times 6$ (first FC layer) $+2 \times 6$ (second FC layer) ≈ 2708 trainable parameters. The CNN can be extended to more than two possible spatial locations (and thus competing speakers) by adding more output neurons to the network to generalize it to a multi-class problem, in which each class corresponds to a location or zone in which the attended speaker is believed to be positioned.

A cross-entropy cost function is minimized using mini-batch gradient descent. Weight decay regularization is applied, as well as a post-training selection of the optimal model based on the validation loss. Furthermore, during training, not only data from the subject under test (as in all other methods) but also data from other subjects are used, as it was found in [136] that this prevents the model from overfitting on the training data in case only a limited amount of data of the subject under test is available. Therefore, this inclusion of data from other subjects can be seen as a type of regularization.

Other nonlinear methods

Several other nonlinear AAD methods have been proposed since the writing of the paper on which this chapter is based but are not included in the comparative study. In Chapter 7, we propose a nonlinear RGC-based method to decode the spatial focus of auditory attention as an extension on the CSP-based method [155]. The steep uprise of DNN-based methods is also noticeable in AAD literature. For example, Kuruvila et al. proposed in [138] a new CNN/LSTM model on the EEG and speech spectrograms for AAD. Furthermore, various (spatio-temporal) attention mechanisms in the DNN have been combined with the DNN, such as in [139–141]. However, several of these methods have problems with cross-dataset generalization or concerning a proper validation (i.e., a CV procedure that avoids potential overfitting). This is consistent with the findings in the next section.

3.3 Comparative study of AAD algorithms

We compared the AAD algorithms discussed above on two publicly available datasets (Dataset A and Dataset B) in a subject-specific manner. Both datasets have been collected with the purpose of AAD, using a competing talker setup in which two stories are simultaneously narrated. Details on the datasets and the preprocessing of the EEG and audio data are described in Pop-out box 1. All algorithms, including the deep learning methods, are re-trained from scratch on each dataset separately.

Given a decision window length τ, the performance of each algorithm is evaluated via the accuracy $p \in [0, 100]\%$, defined as the percentage of correctly classified decision windows. Since EEG is the superimposed activity of many different (neural) processes, the correlation ρ between the reconstructed and attended envelope is typically quite low (in the order of 0.05-0.2). Therefore, it is important to use a sufficiently long decision window such that the decision process is less affected by estimation noise in ρ due to the finite sample size. As a result, the accuracy p generally increases for longer decision window lengths τ, leading to a so-called '$p(\tau)$-performance curve'. These accuracies are obtained using the CV procedure described in Pop-out box 2.

This $p(\tau)$-performance curve thus presents a trade-off between the accuracy and decision delay of the AAD system (a long decision length implies a slower reaction time to a switch in attention). In Chapter 2, the MESD metric has been proposed to resolve this trade-off in order to compare AAD algorithms more easily. The MESD metric determines the most optimal point on the $p(\tau)$-performance curve in the context of attention-steered gain control by minimizing the expected time it takes to switch the gain between two speakers in an optimized robust gain control system. As such, it outputs a single-number time metric (the MESD [s]) for a $p(\tau)$-performance curve and thus removes the loss of statistical power due to multiple-comparison corrections in statistical hypothesis testing (due to testing for multiple decision window lengths). Furthermore, the MESD ensures that the statistical comparison is automatically focused on the most practically relevant points on the $p(\tau)$-performance curve, which typically turn out to be the ones corresponding to short decision window lengths $\tau < 10\,\text{s}$ (Chapter 2). A higher MESD corresponds to a worse AAD performance and vice versa. This MESD metric is a theoretical metric that is not based on actual attention switches in the data, which are also not present in the datasets used. It is merely used here as a comparative metric, which does not necessarily reflect the true switching time as it relies on independence assumptions in the underlying Markov model, which can be violated in practice.

Pop-out box 1: Experiment details

Data: The characteristics of both datasets are summarized in the following table:

Attribute	Dataset A	Dataset B
Number of subjects	16	18
Amount of data per subject	72 min	50 min
EEG system	64-channel BioSemi	64-channel BioSemi
Speakers	male-male	male-female
Azimuth direction sources	±90°	±60°
Acoustic room condition	dichotic and HRTF-filtered in anechoic room	HRTF-filtered in anechoic, mildly, and highly reverberant room

Speech envelope extraction: The individual speech signals are passed through a gammatone filterbank, which roughly approximates the spectral decomposition as performed by the human auditory system. Per subband, the audio envelopes are extracted and their dynamic range is compressed using a power-law operation with exponent 0.6, after which the subband envelopes are summed into a single broadband envelope [134].

Frequency range: For computational efficiency, the speech envelopes as well as the EEG signals are both downsampled to $f_s = 64\,\text{Hz}$, and bandpass filtered between 1–32 Hz [119, 136, 143]. For the linear algorithms, this was further reduced to $f_s = 20\,\text{Hz}$ and 1–9 Hz in order to be able to reduce the number of parameters in the spatio-temporal decoders (linear SR methods have been demonstrated not to exploit information above 9 Hz [134]).

Hyperparameter settings: The decoder lengths and CNN kernel lengths are set as in the original papers. For all linear methods, this is $L = 250\,\text{ms}$, for NN-SR $L = 420\,\text{ms}$, for CNN-loc $L = 130\,\text{ms}$, and for CNN-sim $L_1 = 30\,\text{ms}$ (first layer) and $L_2 = 10\,\text{ms}$ (second layer). For CCA, 1.25 s is chosen as the encoder length. The full set of 64 channels are used in all algorithms, except for MMSE-adap-lasso, where the same 28 channels as in [44] are chosen to reduce the number of parameters (since the decoder is estimated on much less data). The regularization parameters are cross-validated using ten values in the range $[10^{-6}, 0]$. For CCA, it turned out that retaining all PCA components for both datasets is optimal.

Pop-out box 2: Details on cross-validation (CV) procedure

Two-stage CV: The different algorithms are evaluated via a two-stage CV procedure applied per subject and decision window length. The AAD accuracy is determined via an outer leave-one-segment-out CV (LOSO-CV) loop. Per outer fold, the optimal hyperparameter is determined via an inner ten-fold CV loop on the training set of the outer loop. The length of each left-out segment in the outer loop is chosen equal to 60 s, which is split into smaller disjoint decision windows. For example, for a decision window length of 30 s, each left-out segment results in two decisions. Additional details per algorithm are provided in the following table (standard CV corresponds to training on all but one segment, testing on the left-out segment):

Method	Outer LOSO-CV loop	Inner 10-CV loop
MMSE-avgcorr-ridge/lasso	standard	optimization of λ (independent of τ, tuned based on largest value of τ)
MMSE-avgdec-ridge/lasso	training data of each fold is split into segments of the same size as τ. A different decoder is estimated in each of these subsegments and the decoders are averaged across all training folds (similar to [74])	optimization of λ (re-optimized for τ due to the dependency of the training procedure on τ)
CCA	standard, additional LOSO-CV loop to train and test LDA classifier	optimization of the number of canonical correlation coefficients J as input for LDA (re-optimized for each τ)
MMSE-adap-lasso	optimization of λ per τ and fold by taking hyperparameter with highest accuracy on training fold	/
NN-SR	standard	/
CNN-loc	LOSpO-CV instead of LOSO-CV, training *and* testing redone for τ	/
CNN-sim	ten-fold CV instead of LOSO-CV (due to computation time), training *and* testing redone for τ	/

Leave-one-speaker-out CV (LOSpO-CV): When using the LOSO-CV method, the test set always contains a speaker that is also present in the training set. To avoid potential overfitting to speakers in the training set for the CNN-loc algorithm, we use the LOSpO-CV method for this algorithm, as proposed and explained in [136]. For the linear methods, there is no difference between the LOSO-CV and LOSpO-CV method. This is validated by performing 100 runs per subject, with in each run another random CV split (using the same amount of folds as for LOSpO-CV). We then tested whether the LOSpO-CV performance significantly differs from the median of this empirical distribution (i.e., the median over all random splits) across all subjects. For the CCA method, which has most degrees of freedom to overfit, the difference between the LOSpO-CV and median random-CV accuracy is less than 1% on 20s decision windows, and a paired Wilcoxon signed-rank test (over subjects) shows no significant difference ($n = 16, p = 0.38$).

3.3.1 Statistical analysis

To statistically compare the included AAD algorithms, we adopt a linear mixed-effects model (LMM)[1] on the MESD values with the AAD algorithm as a fixed effect and with subjects as a repeated-measure random effect. Five contrasts of interest were set a priori according to the binary tree structure in Figure 3.1. Algorithms that were not competitive or did not perform significantly better than chance are excluded from the statistical analysis, which is why some algorithms are not included in the contrasts (see Section 3.3.2). The planned contrasts reflect the most important different features between AAD algorithms, as shown in Figure 3.1, motivating how they are set. The significance level is set at $\alpha = 0.05$.

3.3.2 Results

Performance curves

Figure 3.7 shows the $p(\tau)$-performance curves of the different AAD algorithms on both datasets. For the MMSE-based decoders, it is observed that there is barely an effect of the type of regularization and that averaging correlation matrices (early integration) consistently outperforms averaging decoders (late integration). Furthermore, CCA outperforms all other linear algorithms. Lastly, on Dataset A, it is clear that decoding the spatial focus of attention using CNN-loc substantially outperforms the SR methods for short decision windows ($< 10\,\text{s}$), where CNN-loc appears to be less affected by the decision window length. However, the standard error on the mean is much higher for the CNN-loc algorithm than for the other methods, indicating a higher inter-subject variability.

The performances of MMSE-adap-lasso, CNN-sim, and NN-SR are not shown in Figure 3.7 as they did not exceed the significance level or were not competitive on either of the two datasets. For a decision window length of 10 s, the MMSE-adap-lasso algorithm achieves an average accuracy of 52.9% with a standard deviation of 4.3% on Dataset A and 49.8% with a standard deviation of 5.9% on Dataset B. The CNN-sim algorithm achieves 51.7% on average with a standard deviation of 2.3% on Dataset A (where there was no convergence for five subjects) and 58.1% with a standard deviation of 9.2% on Dataset B. Lastly, the NN-SR algorithm achieves on average only 52.1% (standard deviation 4.4%) on Dataset A and 52.3% (standard deviation 3.6%) on Dataset B. As these algorithms did not significantly outperform a random classifier or were not competitive, they

[1] See [161, 162] for some (introductory) material/tutorials on LMMs.

were also excluded from the statistical analysis. Furthermore, CNN-loc did not perform well on Dataset B (i.e., 56.3% with a standard deviation of 4.5% on 10 s decision windows). As such, planned contrast I was also excluded from the analysis for Dataset B.

Subject-specific MESD performance

A visual analysis of the per-subject MESD values (Figure 3.8) confirms the trends based on the performance curves. These trends are also confirmed by the statistical analysis[2] using the LMM. There indeed is a significant improvement when decoding the spatial focus of attention via a nonlinear method versus the linear SR methods ($p < 0.001$ (Dataset A)). Furthermore, CCA significantly outperforms all backward stimulus decoders ($p < 0.001$ (Dataset A), $p < 0.001$ (Dataset B)), while there is also a significant improvement when averaging correlation matrices compared to averaging decoders ($p = 0.0028$ (Dataset A), $p < 0.001$ (Dataset B)). There is no significant effect of the specific regularization technique ($p = 0.79$ (Dataset A), $p = 0.30$ (Dataset B) in averaging correlation matrices; $p = 0.57$ (Dataset A), $p = 0.91$ (Dataset B) in averaging decoders).

3.3.3 Discussion

From the results and statistical analysis, it is clear that CCA [153], which adopts a joint forward and backward model, outperforms the other SR methods. Furthermore, the CNN-loc method [136], which decodes the spatial focus of attention based on the EEG alone (i.e., without using the speech stimuli), substantially outperforms all SR methods on Dataset A at short decision window lengths, leading to substantially lower MESDs. This relatively high performance at short decision windows is attributed to the fact that this method avoids correlating the decoded EEG with the speech envelope, thereby not suffering from the noise-susceptible correlation estimation. However, the non-significant performance of CNN-loc on Dataset B implies that alternative algorithms for decoding the spatial focus of attention might be required to improve robustness and generalization to different conditions. We will pursue such alternative algorithms in Part III.

Remarkably, while the traditional linear SR methods are found to perform well across datasets, none of the tested nonlinear (DNN) methods achieve a competitive performance on *both* benchmark datasets, even though high performances were obtained on the respective datasets used in [119, 136, 143].

[2] The two outlying subjects of the CNN-loc algorithm were removed in all comparisons on Dataset A.

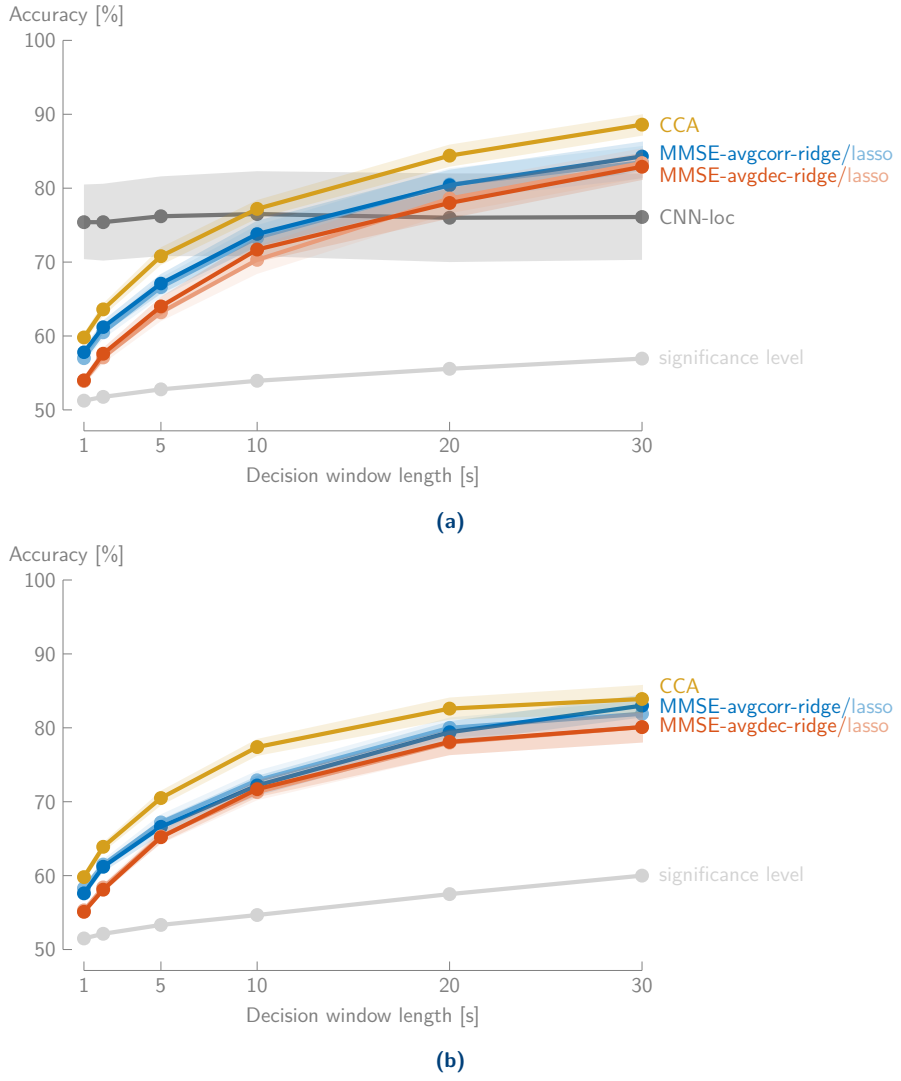


Figure 3.7: The accuracy p (mean $\pm$ standard error on the mean across subjects) as a function of the decision window length τ for **(a)** Dataset A and **(b)** Dataset B. MMSE-adap-lasso, CNN-sim, and NN-SR did not perform significantly better than a random classifier and are not depicted. CNN-loc achieved competitive results only on Dataset A.

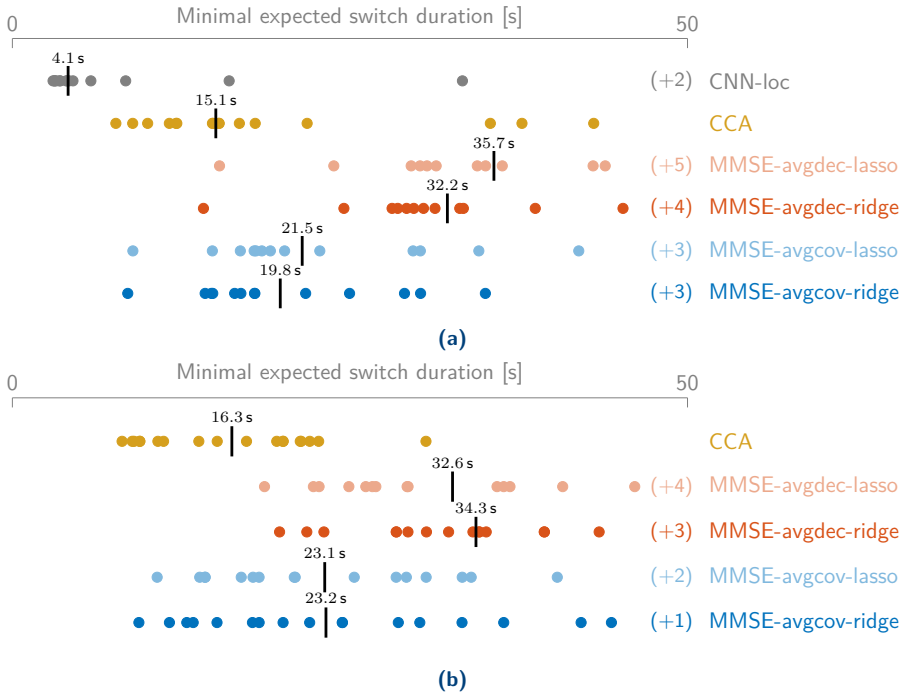


Figure 3.8: The per-subject MESD values, with the median indicated with a bar, for **(a)** Dataset A and **(b)** Dataset B. The number of data points with an MESD $> 50\,\mathrm{s}$ are indicated as (+x). These points were, however, included in the computation of the medians.

This shows that these architectures do not always generalize well, even after re-training them on a new dataset (the original authors validated the implementations in our benchmark study to rule out potential discrepancies in the implementation). Due to the black-box nature of these methods, it remains unclear what causes success on one dataset and failure on another. One possible explanation is that the design process that eventually led to the reported network architecture was too tailored to a particular dataset (and its size), despite proper CV. Furthermore, (D)NNs may potentially pick up subtle patterns that may change or become absent in different experimental set-ups due to differences in equipment, speech stimuli, or experiment protocols.

Although this lack of reproducibility across datasets seems to undermine the practical usage of the presented nonlinear AAD methods, the current benchmark datasets are possibly too small for these methods to draw firm conclusions. AAD based on (D)NNs may become more robust when larger

datasets become available, containing more subjects, more EEG data per subject, and more variation in experimental conditions. Nevertheless, the results of this comparative study point out the risks of overfitting and overdesigning these architectures, thereby emphasizing the importance of extensive validation with multiple independent datasets.

3.4 Outlook and conclusion

Several studies have demonstrated that it is possible to decode the auditory attention from a non-invasive neurorecording technique such as EEG. In our comparative study, we have shown that most of these results are reproducible on different datasets. However, even for the best linear (SR) method (CCA), the accuracy at short decision windows is still too low, potentially leading to too slow reactions of the system to shifts in auditory attention, as indicated by a median MESD of 15 s. The results of this study have demonstrated that an alternative strategy such as decoding the spatial focus of attention could significantly improve performance on these short decision window lengths. In Part III, we will develop new AAD algorithms that exploit this alternative paradigm of decoding the spatial focus of auditory attention. Although nonlinear (deep learning) methods are believed to be able to improve AAD performances substantially, our study has demonstrated that the reported results obtained by these methods are hard to replicate on multiple independent AAD datasets. A major future challenge for AAD research is the design of an algorithm or neural network architecture that reliably improves on short decision windows and is reproducible on different independent datasets.

Furthermore, most presented AAD algorithms require supervised training and are fixed during operation. To avoid cumbersome a priori training sessions for each individual user, as well as to adapt to the time-varying statistics of the EEG (for example, in different listening scenarios), training-free or unsupervised adaptive AAD algorithms should be developed. While several steps have been made in that direction [44], the results of this study show that we are still far away from a practical solution. Therefore, in Part II, we will discuss a novel SR-based unsupervised and time-adaptive AAD algorithm that aims to address this need of unsupervised adaptive AAD algorithms. Such online adaptive AAD algorithms are, moreover, paramount in the development of closed-loop systems for neuro-steered hearing devices, in which the end-user can react to and interact with the AAD algorithm and speech enhancement system. The interplay between the algorithmic processes in the hearing device and the user could enable neurofeedback effects that significantly improve the performance of the hearing device [3] (see also Section 8.2).

Lastly, these AAD algorithms need to be further evaluated in real-life situations, taking various realistic listening scenarios into account, as well as on potential hearing device users (Section 1.7.4). The individual building blocks of a neuro-steered hearing device (Figure 1.6) need to be integrated, in which an AAD algorithm is combined with a reliable and low-latency speaker separation algorithm, a miniaturized EEG sensor system, and a smart gain control system (see also Section 8.2).

Part II

Unsupervised AAD

4 | Unsupervised self-adaptive stimulus reconstruction

This chapter is largely based on S. Geirnaert, T. Francart, and A. Bertrand, "Unsupervised Self-Adaptive Auditory Attention Decoding," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 10, pp. 3955-3966, 2021.

ABSTRACT | As shown in Chapter 3, stimulus decoders, which are commonly used in AAD, are traditionally trained in a supervised manner, requiring a dedicated training stage during which the attended speaker is known. Pre-trained subject-independent decoders alleviate the need of having such a per-user training stage but perform substantially worse than supervised subject-specific decoders that are tailored to the user. This motivates the development of a new unsupervised self-adapting training/updating procedure for a subject-specific decoder, which iteratively improves itself on unlabeled EEG data using its own predicted labels. This iterative updating procedure enables a self-leveraging effect, of which we provide a mathematical analysis that reveals the underlying mechanics. Starting from a random decoder, the proposed unsupervised algorithm results in a decoder that outperforms a supervised subject-independent decoder. Starting from a subject-independent decoder, the unsupervised algorithm even closely approximates the performance of a supervised subject-specific decoder. The developed unsupervised AAD algorithm thus combines the two advantages of a supervised subject-specific and subject-independent decoder: it approximates the performance of the former while retaining the 'plug-and-play' character of the latter. As the proposed algorithm can be used to automatically adapt to new users, it contributes to more practical neuro-steered hearing devices.

4.1 Introduction

As shown in Chapter 3, the main class of current AAD algorithms exploits the neural envelope tracking phenomenon (Section 1.5) by reconstructing the attended speech envelope from the recorded EEG signals via a stimulus decoder. The reconstructed speech envelope can then be compared through the Pearson correlation coefficient with the speech envelopes of the active speakers to determine which speaker is the attended one.

AAD decoders/algorithms can be used in a subject-specific or subject-independent way [74], trading practical applicability with better performance:

- A **subject-specific** decoder is traditionally trained in a *supervised* manner, requiring a cumbersome a priori training stage in which data from the subject under test are collected to train an AAD decoder. This popular approach is thus less practical to implement on hearing devices. However, it is known that this approach results in the highest AAD performance for a given AAD algorithm [74].
- A **subject-independent** decoder also requires labeled data, but only of subjects other than the subject under test, which allows to pre-train it. At test time, this subject-independent decoder can be applied to the incoming data of the new, unseen subject without a priori requiring information about the attention processing of that particular subject. As such, it could be used in a 'plug-and-play' fashion, pre-installed on each neuro-steered hearing device and thus leading to a generic hearing device. However, this practical applicability comes at the cost of a lower AAD performance, as the decoder fails to capture the subject-specific differences in auditory processing [74].

Moreover, both decoders remain fixed during operation, when new data of the subject under test comes in. They do not adapt to changing conditions and situations and thus result in suboptimal decoding results.

Except for the algorithm in [44], other AAD algorithms are supervised and very often subject-specifically trained (Chapter 3). In [44], a dynamic AAD algorithm is proposed, in which a decoder is estimated for each speaker per new incoming segment of data. These decoders are then applied again to that same segment of data to determine the auditory attention. Although some labeled data is required to tune specific hyperparameters, this algorithm is by design unsupervised. However, this algorithm is substantially outperformed by all other traditional (supervised) AAD algorithms (Chapter 3).

We propose a fully unsupervised subject-specific AAD algorithm, in which a stimulus decoder is iteratively updated on the EEG data and speech envelopes. This iterative updating does *not* require ground-truth labels, i.e., knowledge about which is the attended or unattended speaker. Instead, the model updates itself based on its own predictions in the previous iteration. We hypothesize that this results in a self-leveraging effect. As such, it should automatically adapt to a new subject, integrating the two major advantages of a subject-specific and subject-independent decoder:

1. A higher performance than a subject-independent decoder.
2. Retaining the unsupervised 'plug-and-play' feature of a subject-independent decoder, thus without requiring knowledge about the labels during training.

Furthermore, such a self-adaptive algorithm could be applied adaptively in time. As EEG and audio data are continuously recorded, it adapts to changing conditions and situations.

First, we revisit the traditional supervised training of a stimulus decoder in Section 4.2. We then introduce the proposed method to update a stimulus decoder in an unsupervised manner in Section 4.3. In Section 4.4, the data, preprocessing, and performance evaluation are explained. In Section 4.5, we provide a recursive mathematical model to track the iterations of the unsupervised algorithm, with the aim to gain some insights into the mechanics of the self-leveraging effect. The proposed method is then tested on two separate datasets in Section 4.6. Applications, future work, and conclusions are discussed in Section 4.7.

4.2 Supervised training of a stimulus decoder

Before explaining the newly proposed unsupervised procedure in Section 4.3, we first revisit the traditional supervised training of a stimulus decoder for AAD.

In the classical approach[1] towards AAD (see, e.g., [1, 5, 74, 142, 163]), a linear spatio-temporal filter $d_c(l)$, referred to as a decoder, reconstructs the attended speech envelope $s_{\mathrm{a}}(t)$ from the C-channel EEG signal $x_c(t)$ by anti-causally integrating EEG samples over L time lags, for each EEG channel $c \in \{1, \ldots, C\}$:

$$\hat{s}_{\mathrm{a}}(t) = \sum_{c=1}^{C} \sum_{l=0}^{L-1} d_c(l) x_c(t+l), \tag{4.1}$$

[1] A MATLAB implementation of this AAD approach can be found in [163].

with l the time lag index and c the channel index.

Equation (4.1) can be rewritten in vector format as:

$$\hat{s}_{\mathrm{a}}(t) = \mathbf{d}^{\mathrm{T}}\mathbf{x}(t),$$

where

$$\mathbf{x}(t) = \begin{bmatrix} x_1(t) \\ \vdots \\ x_1(t+L-1) \\ x_2(t) \\ \vdots \\ x_C(t+L-1) \end{bmatrix} \in \mathbb{R}^{CL\times 1}$$

contains L lags, for each EEG channel. Similarly, the vector $\mathbf{d} \in \mathbb{R}^{CL\times 1}$ stacks all decoder coefficients $d_c(l)$, across all channels and time lags. The decoder $\mathbf{d}$ is then found by minimizing the squared error:

$$\hat{\mathbf{d}} = \underset{\mathbf{d}}{\operatorname{argmin}} \; ||\mathbf{s}_{\mathrm{a}} - \mathbf{X}\mathbf{d}||_2^2, \tag{4.2}$$

with $\mathbf{s}_{\mathrm{a}} = \begin{bmatrix} s_{\mathrm{a}}(0) & \cdots & s_{\mathrm{a}}(T-1) \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^{T\times 1}$ and $\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 & \cdots & \mathbf{X}_C \end{bmatrix} \in \mathbb{R}^{T\times CL}$ a block Hankel matrix, with

$$\mathbf{X}_c = \begin{bmatrix} x_c(0) & x_c(1) & x_c(2) & \cdots & x_c(L-1) \\ x_c(1) & x_c(2) & x_c(3) & \cdots & x_c(L) \\ x_c(2) & x_c(3) & x_c(4) & \cdots & x_c(L+1) \\ \vdots & \vdots & \vdots & & \vdots \\ x_c(T-1) & 0 & 0 & \cdots & 0 \end{bmatrix} \in \mathbb{R}^{T\times L}.$$

Defining the sample autocorrelation matrix $\hat{\mathbf{R}}_{xx} \in \mathbb{R}^{CL\times CL}$ and sample crosscorrelation vector $\hat{\mathbf{r}}_{xs_{\mathrm{a}}} \in \mathbb{R}^{CL\times 1}$ as:

$$\hat{\mathbf{R}}_{xx} = \frac{1}{T}\mathbf{X}^{\mathrm{T}}\mathbf{X} \text{ and } \hat{\mathbf{r}}_{xs_{\mathrm{a}}} = \frac{1}{T}\mathbf{X}^{\mathrm{T}}\mathbf{s}_{\mathrm{a}}, \tag{4.3}$$

the solution of (4.2) is equal to:

$$\begin{aligned} \hat{\mathbf{d}} &= (\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{s}_{\mathrm{a}} \\ &= \hat{\mathbf{R}}_{xx}^{-1}\hat{\mathbf{r}}_{xs_{\mathrm{a}}}. \end{aligned} \tag{4.4}$$

This classical supervised training approach is summarized in Figure 4.1.

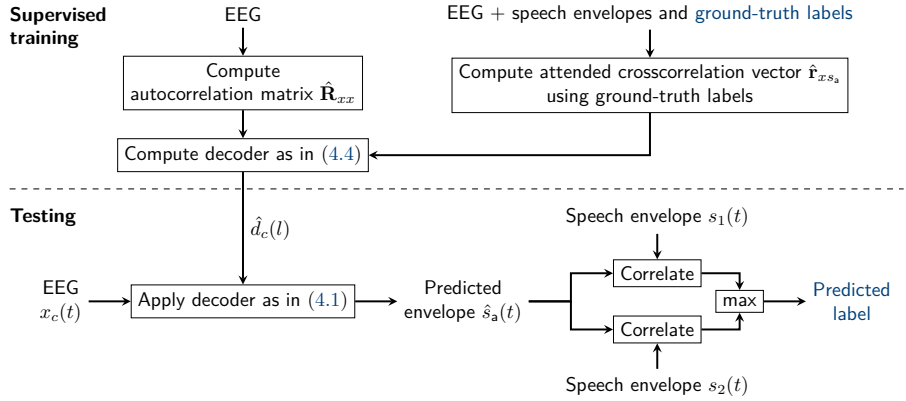


Figure 4.1: A conceptual overview of the traditional supervised training approach of a stimulus decoder and its application to new test data.

Often, ridge regression is used to avoid overfitting when only a limited amount of training data is available [1,5,142,163], such that the decoder is estimated as:

$$\hat{\mathbf{d}} = (\mathbf{X}^{\mathrm{T}}\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{s}_{\mathrm{a}}, \tag{4.5}$$

where the regularization parameter λ needs to be optimized, for example, through a CV step. When sufficient training data is available, the regularization can be omitted [1].

In practice, a labeled training set of K segments (for example, corresponding to different trials in an experiment) of EEG data and corresponding speech envelopes of the competing speakers, $\{\mathbf{X}_k, (\mathbf{s}_{1_k}, \mathbf{s}_{2_k}), y_k\}_{k=1}^{K}$, is available. Note that in a practical system, these speech envelopes need to be extracted from the recorded speech mixtures in a hearing device, for which various methods exist as explained in Section 1.7.2. The labels $y_k \in \{1, 2\}$ indicate whether $\mathbf{s}_{1_k}$ or $\mathbf{s}_{2_k}$ is the attended speech envelope. Per segment k, the attended speech envelope $\mathbf{s}_{\mathrm{a}_k}$ thus corresponds to the speech envelope of the set $(\mathbf{s}_{1_k}, \mathbf{s}_{2_k})$ that corresponds to label y_k. Then (4.5) becomes:

$$\hat{\mathbf{d}} = \underbrace{\left(\sum_{k=1}^{K}\mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k + \lambda\mathbf{I}\right)^{-1}}_{\hat{\mathbf{R}}_{xx}^{-1}} \underbrace{\sum_{k=1}^{K}\mathbf{X}_k^{\mathrm{T}}\mathbf{s}_{\mathrm{a}_k}}_{\hat{\mathbf{r}}_{xs_{\mathrm{a}}}} \tag{4.6}$$

It is crucial to realize that the estimation of the decoder in (4.6) is inherently a *supervised* problem, as the ground-truth label y_k needs to be known to select the attended speech envelope $\mathbf{s}_{\mathrm{a}_k}$ in each segment k.

At test time, the estimated decoder $\hat{\mathbf{d}}$ is used to reconstruct the attended speech envelope from a new EEG segment $\mathbf{X}^{(\text{test})}$. Given two speech envelopes $\mathbf{s}_1^{(\text{test})}$ and $\mathbf{s}_2^{(\text{test})}$, corresponding to two competing speakers, the first speaker is identified as the attended one if the sample Pearson correlation coefficient between the reconstructed speech envelope $\hat{\mathbf{s}}_\text{a} = \mathbf{X}^{(\text{test})}\hat{\mathbf{d}}$ and the first speaker is larger than with the second speaker, i.e.,

$$\rho\left(\hat{\mathbf{s}}_\text{a}, \mathbf{s}_1^{(\text{test})}\right) > \rho\left(\hat{\mathbf{s}}_\text{a}, \mathbf{s}_2^{(\text{test})}\right), \tag{4.7}$$

and vice versa. This is summarized in the 'Testing' part in Figure 4.1. Note that, for the sake of an easy exposition, we assume that there are two competing speakers, although all proposed algorithms can be generalized to more than two competing speakers.

4.3 Unsupervised training of a stimulus decoder

Assume the availability of a training set of K segments of EEG data and speech envelopes, $\{\mathbf{X}_k, (\mathbf{s}_{1_k}, \mathbf{s}_{2_k})\}_{k=1}^K$, but now *without* knowledge of the attended speaker, i.e., the labels y_k are *not* available. Only the presented competing speech envelopes $(\mathbf{s}_{1_k}, \mathbf{s}_{2_k})$ are known, of which one corresponds to the attended speaker, while the other corresponds to the unattended one. This means that training a decoder to reconstruct the attended speech envelope boils down to an *unsupervised* problem. We thus remove the requirement of subject-specific ground-truth labels. However, we implicitly assume that it is important for the training of the stimulus decoder to know which envelope corresponds to the attended speaker and which one to the unattended speaker. In other words, we assume that the attended and unattended speakers are encoded differently in the brain. If this would not be the case, one could simply train the decoder based on the sum of the envelopes of both speakers. Such a training procedure would also be unsupervised and would remove the necessity of determining which speaker is attended during the training process. While the assumption that both competing speakers are encoded distinctly in the brain is already verified in the literature (e.g., see [41, 42] and Section 1.5), we also confirm it here in Section 4.5.2.

Figure 4.2 shows a conceptual overview of the proposed unsupervised training procedure, in which a decoder is trained in an unsupervised manner by iteratively (re)predicting the labels and updating the decoder. The key idea is thus to replace the ground-truth labels in the supervised training stage (top part of Figure 4.1), with the *predicted* labels from the testing stage (bottom part of Figure 4.1), and iterate a few times. Below, we will explain each step of the algorithm, while we refer to Algorithm 2 for a detailed summary.

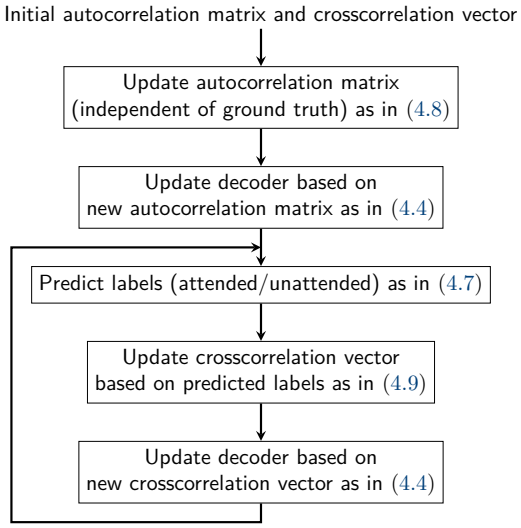


Figure 4.2: A conceptual overview of the iterative self-adaptive unsupervised training procedure of a stimulus decoder.

In the first step, the autocorrelation matrix in (4.6) is estimated using the subject-specific EEG data. This autocorrelation matrix is independent of the ground-truth labels, which are only required for the crosscorrelation vector. It is thus always possible to perform this update. If desired, the estimated and regularized autocorrelation matrix can be linearly combined with an initially provided autocorrelation matrix $\mathbf{R}_{xx}^{(\text{init})}$, controlled with the user-defined hyperparameter $0 \leq \alpha \leq 1$ (and $1 - \alpha$):

$$\hat{\mathbf{R}}_{xx} = (1 - \alpha) \left(\sum_{k=1}^{K} \mathbf{X}_k^{\mathrm{T}} \mathbf{X}_k + \lambda \mathbf{I} \right) + \alpha \mathbf{R}_{xx}^{(\text{init})}. \tag{4.8}$$

This hyperparameter can be interpreted as the amount of confidence in the a priori available autocorrelation matrix $\mathbf{R}_{xx}^{(\text{init})}$. This initial autocorrelation matrix can be estimated on, for example, subject-independent data and can be considered as an extra regularization term (for example, as used in Tikhonov regularization). If no such a priori autocorrelation matrix is available, α is simply set to 0. Using the updated autocorrelation matrix, the decoder is estimated in combination with an initially provided crosscorrelation vector $\mathbf{r}_{xs_{\text{a}}}^{(\text{init})}$. This crosscorrelation vector can again be estimated in a subject-independent manner but could also be generated fully randomly. It is recommended to normalize the initial autocorrelation matrix and crosscorrelation vector such that they

Algorithm 2: Unsupervised training or adaptation of a stimulus decoder

Input: A training set of K segments of EEG data and speech envelopes $\{\mathbf{X}_k, (\mathbf{s}_{1_k}, \mathbf{s}_{2_k})\}_{k=1}^{K}$; initial autocorrelation matrix $\mathbf{R}_{xx}^{(\text{init})}$ and crosscorrelation vector $\mathbf{r}_{xs_\text{a}}^{(\text{init})}$; regularization parameter λ and updating hyperparameters α and β; maximal number of iterations $i_{\max}$

Output: A stimulus decoder $\hat{\mathbf{d}}$

1: Compute/update the autocorrelation matrix and compute an initial decoder:

$$\begin{cases} \hat{\mathbf{R}}_{xx} = (1-\alpha)\left(\sum_{k=1}^{K} \mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k + \lambda\mathbf{I}\right) + \alpha\mathbf{R}_{xx}^{(\text{init})} \\ \hat{\mathbf{d}} = \hat{\mathbf{R}}_{xx}^{-1}\mathbf{r}_{xs_\text{a}}^{(\text{init})} \end{cases}$$

2: **while** $i \leq i_{\max}$ and $\hat{\mathbf{d}}$ changes **do**

3: Predict the labels on the training set:

$$\forall\, k \in \{1, \ldots, K\} : \begin{cases} \hat{\mathbf{s}}_k = \mathbf{X}_k\hat{\mathbf{d}} \\ \mathbf{s}_{\text{pred}_k} = \underset{\mathbf{s}_{1_k}, \mathbf{s}_{2_k}}{\text{argmax}} \ \left(\rho(\hat{\mathbf{s}}_k, \mathbf{s}_{1_k}), \rho(\hat{\mathbf{s}}_k, \mathbf{s}_{2_k})\right) \end{cases}$$

4: Update the crosscorrelation vector using the predicted labels and update the decoder:

$$\begin{cases} \hat{\mathbf{r}}_{xs_\text{pred}} = (1-\beta)\sum_{k=1}^{K} \mathbf{X}_k^{\mathrm{T}}\mathbf{s}_{\text{pred}_k} + \beta\mathbf{r}_{xs_\text{a}}^{(\text{init})} \\ \hat{\mathbf{d}} = \hat{\mathbf{R}}_{xx}^{-1}\hat{\mathbf{r}}_{xs_\text{pred}} \end{cases}$$

5: **end while**

have a Frobenius norm equal to the estimated autocorrelation matrix and crosscorrelation vector, improving the interpretability of the hyperparameters.

Using the updated autocorrelation matrix (4.8) and the initial crosscorrelation vector $\mathbf{r}_{xs_\text{a}}^{(\text{init})}$, we compute an initial decoder $\hat{\mathbf{d}}$ according to (4.4). This initial decoder acts as a bootstrap to initiate the iterative procedure to update the decoder weights. Starting from this initial decoder, the labels of the training segments are predicted based on the maximal sample Pearson correlation coefficient between the reconstructed envelope and the speech envelopes of the competing speakers. These predicted labels are then used to select the attended speech envelopes $\mathbf{s}_{\text{pred}_k}$ in each of the K segments, which are afterwards used to update the crosscorrelation vector. Note that it is crucial that the updating is performed not using the reconstructed envelope from the EEG, but with the speech envelope of one of the two competing speakers identified/predicted as the

attended one. Again, some prior knowledge can be introduced in the updating of the crosscorrelation vector using an initially provided crosscorrelation vector $\mathbf{r}_{xs_a}^{(\text{init})}$ and hyperparameter $0 \leq \beta \leq 1$:

$$\hat{\mathbf{r}}_{xs_{\text{pred}}} = (1-\beta)\sum_{k=1}^{K} \mathbf{X}_k^{\text{T}} \mathbf{s}_{\text{pred}_k} + \beta \mathbf{r}_{xs_a}^{(\text{init})}. \tag{4.9}$$

The updated crosscorrelation vector can then be used to re-estimate the decoder. Multiple iterations of predicting the labels and updating the decoder can be performed until the decoder has converged or a maximal number of iterations has been reached. It is expected that this iterative process initiates a *self-leveraging* effect, in which the decoder leverages its own predictions to improve. In Section 4.5, we provide a mathematical analysis that explains the underlying mechanism behind this self-leveraging effect and why it works.

Using the unsupervised updating scheme in Algorithm 2, a stimulus decoder can be trained. In Section 4.6, we evaluate this unsupervised algorithm using different hyperparameter settings and compare it to a supervised subject-independent and supervised subject-specific decoder.

4.4 Experiments and evaluation metrics

In this section, we provide all information on the data (Section 4.4.1), preprocessing and decoder settings (Section 4.4.2), and evaluation procedure and metrics (Section 4.4.3) required to replicate and reproduce all experiments and results. All experiments are performed in MATLAB.

4.4.1 AAD datasets

We validate the proposed unsupervised AAD algorithm on two separate datasets: Dataset A and Dataset B. The first one (Dataset A) consists of EEG recordings of 16 normal-hearing subjects, attending to one out of two competing speakers [1]. These competing speakers are located at $\pm 90°$ along the azimuth direction. Per subject, 72 min of EEG and audio data are available. This dataset is available online [132].

The second dataset (Dataset B) consists of EEG recordings of 18 normal-hearing subjects, attending to one out of two competing speakers, located at $\pm 60°$ along the azimuth direction [2]. Per subject, 50 min of EEG and audio data are available. Different acoustic room settings are used: anechoic, mildly

reverberant, and highly reverberant. This dataset is available online as well [133]. Both datasets are recorded using a 64-channel BioSemi ActiveTwo system.

4.4.2 Preprocessing and decoder settings

The preprocessing of the EEG and audio data is very similar to [1]. The audio signals are first filtered using a gammatone filterbank. From each subband signal, the envelope is extracted using a power-law operation with exponent 0.6, after which one final envelope is computed by summing the different subband envelopes. Both the EEG data and speech envelopes are filtered between 1–9 Hz [134] and downsampled to 20 Hz. Note that we here assume that the clean speech envelopes are readily available and need not be extracted from the microphone recordings. For Dataset B, the 50 s segments are normalized such that they have a Frobenius norm equal to one across all channels.

A maximum of $i_{\max} = 10$ iterations of predicting the labels and updating the decoder is used, which in practice showed to be sufficient (see also Section 4.6).

In the design of the stimulus decoder, $L = 250$ ms is chosen [74], such that the filter spans a range of 0–250 ms post-stimulus. Furthermore, the regularization parameter λ in (4.5), (4.6), and Algorithm 2 is analytically determined using [164], which is the recommended state-of-the-art method to estimate this regularization parameter [38]. Given data matrix $\mathbf{X} \in \mathbb{R}^{T \times CL}$ and sample autocorrelation matrix $\mathbf{S} = \frac{1}{T}\mathbf{X}^{\mathrm{T}}\mathbf{X} \in \mathbb{R}^{p \times p}$, the proposed shrinkage estimator $\hat{\mathbf{S}}$ in [164] of the autocorrelation matrix becomes [165]:

$$\hat{\mathbf{S}} = (1-\eta)\mathbf{S} + \eta \frac{\mathrm{Tr}\left(\mathbf{S}\right)}{p}\mathbf{I}, \tag{4.10}$$

with

$$\eta = \min\left(\frac{\sum_{t=1}^{T} ||\mathbf{x}_t \mathbf{x}_t^{\mathrm{T}} - \mathbf{S}||_{\mathrm{F}}^2}{T^2 \left(\mathrm{Tr}\left(\mathbf{S}^{\mathrm{T}}\mathbf{S}\right) - \frac{\mathrm{Tr}(\mathbf{S})^2}{p} \right)}, 1 \right). \tag{4.11}$$

Note that in our case, $p = CL$, and we ignore the normalization of the autocorrelation matrix (and crosscorrelation vector) by T. The shrinkage formula in (4.10) can easily be rewritten in the form of (4.5), (4.6) upon an irrelevant scaling, in which case λ is set as:

$$\lambda = \frac{\eta}{1-\eta} \frac{\mathrm{Tr}\left(\mathbf{X}^{\mathrm{T}}\mathbf{X}\right)}{CL}.$$

In [164], they show that (4.10) and (4.11) lead to a consistent estimator that is asymptotically optimal with respect to a quadratic loss function with the underlying unknown autocorrelation matrix.

4.4.3 Cross-validation and evaluation

For the *supervised subject-specific* decoder, a random ten-fold CV scheme is used to train and test the decoders. The supervised *subject-independent* decoders are evaluated using a leave-one-subject-out CV (LOSuO-CV) scheme where a decoder is trained on the data of all other subjects and tested on the left-out subject. The proposed *unsupervised subject-specific* decoder is tested in a random ten-fold CV manner as well, where the updating happens on the training set (without knowledge of the labels) and the testing on the left-out data. The partitioning of the data is performed on segments of 60 s for Dataset A and 50 s for Dataset B. Per subject, the continuous recordings are thus first split into these segments and then randomly distributed over a training and test set. At test time, the left-out 60/50 s segments are split into shorter sub-segments of length τ, from hereon referred to as 'decision windows'. The accuracy is then defined as the ratio of correctly decoded decision windows across all test folds. These shorter decision windows are only used in the test folds in order to evaluate the trade-off between the AAD accuracy and the decision window length (longer decision windows provide more accurate correlation coefficients, yielding higher AAD accuracies at the cost of slower decision-making; see Part I). However, the prediction and updating in Algorithm 2 are always performed on the longer 60/50 s segments, in order to maximize the accuracy of the unsupervised training labels.

To resolve the aforementioned trade-off between accuracy and decision window length, we use the MESD performance metric for AAD as proposed in Chapter 2. The MESD represents the theoretical expected time it takes to switch the gain in an optimal attention-steered gain control system, following a switch in auditory attention. Such a gain control system is modeled using a Markov chain model, where the time it takes to step from one state (i.e., gain level) to another is represented by the AAD decision window length and where the step size between gain levels is optimized to ensure stable operation within a pre-defined comfort region in the presence of AAD errors. The ESD can then be computed by quantifying the expected number of steps required to switch to the pre-defined comfort region associated with the other speaker. This gain control system/Markov chain model is optimized across decision window lengths to minimize the time it takes to switch the gain from one source to another while assuring a stable operation within the pre-defined comfort region when the attention is sustained. Note that this metric is computed based on

a stochastic model of a gain control system and is not evaluated using actual switches in attention. However, it allows to easily and statistically compare different decoders across different decision window lengths based on a single (practically relevant) metric. As such, it resolves the aforementioned accuracy-versus-decision-time trade-off. The underlying mathematical principles and definition of this metric can be found in Chapter 2. To compute the MESD, we used the publicly available MESD toolbox from [145].

4.5 Unsupervised updating explained: a mathematical model

Before extensively testing Algorithm 2 on the different datasets in Section 4.6, we attempt to demystify and explain the hypothesized self-leveraging mechanism through a mathematical analysis of the recursion induced by the algorithm.

4.5.1 Mathematical model

Assume that at iteration $i < i_{\max}$ of Algorithm 2, we obtain a decoder with an (unknown) AAD test accuracy of $p_i \in [0, 100]\%$. This means that there is a probability of p_i that the reconstructed envelope using this decoder will have a higher correlation with the attended envelope than with the unattended envelope. Correspondingly, there is a $100\% - p_i$ probability that the unattended envelope will show the highest correlation. Assume for simplicity that $\alpha = 0$ and $\beta = 0$. Due to the linearity of the computation of the crosscorrelation vector (see (4.3)), the updated crosscorrelation vector will then be, on average, equal to:

$$\hat{\mathbf{r}}_{xs_{\text{pred}},i+1} = p_i \hat{\mathbf{r}}_{xs_{\text{a}}} + (1 - p_i)\hat{\mathbf{r}}_{xs_{\text{u}}}, \tag{4.12}$$

with $\hat{\mathbf{r}}_{xs_{\text{a}}}$ the crosscorrelation vector using all attended envelopes and $\hat{\mathbf{r}}_{xs_{\text{u}}}$ the crosscorrelation vector using all unattended envelopes. Similarly, and again due to the linearity in the computations, the corresponding updated decoder becomes:

$$\hat{\mathbf{d}}_{i+1} = p_i \hat{\mathbf{d}}_{\text{a}} + (1 - p_i)\hat{\mathbf{d}}_{\text{u}}, \tag{4.13}$$

with $\hat{\mathbf{d}}_{\text{a}}$ the decoder trained with all attended speech envelopes (which would correspond to the supervised subject-specific decoder with accuracy p_{a}) and $\hat{\mathbf{d}}_{\text{u}}$ the unattended decoder that would be trained with all unattended speech envelopes. This unattended decoder has an accuracy equal to p_{u} on the unattended labels and thus $100\% - p_{\text{u}}$ on the attended labels. As a result, the reconstructed envelope using this updated decoder is a linear combination of

the reconstructed envelope obtained using the (supervised) attended decoder ($\hat{\mathbf{s}}_{\mathrm{a}}$) and the (supervised) unattended decoder ($\hat{\mathbf{s}}_{\mathrm{u}}$):

$$\hat{\mathbf{s}}_{\mathrm{pred},i+1} = p_i \hat{\mathbf{s}}_{\mathrm{a}} + (1 - p_i)\hat{\mathbf{s}}_{\mathrm{u}}. \tag{4.14}$$

The goal is now to find the AAD accuracy p_{i+1} of the updated decoder $\hat{\mathbf{d}}_{i+1}$ (4.13) in iteration $i+1$. We will propose a mathematical model for the function $p_{i+1} = \phi(p_i)$, which determines the accuracy p_{i+1} of the updated decoder as a function of the accuracy p_i of the previous decoder. If $p_{i+1} > p_i$, this implies a self-leveraging effect in which the accuracy improves from one iteration to the next. Given that the speech envelope that exhibits the highest Pearson correlation coefficient with the reconstructed envelope is identified as the attended speaker, this implies that:

$$p_{i+1} = \phi(p_i) = P(\rho(\hat{\mathbf{s}}_{\mathrm{pred},i+1}, \mathbf{s}_{\mathrm{a}}) > \rho(\hat{\mathbf{s}}_{\mathrm{pred},i+1}, \mathbf{s}_{\mathrm{u}}))\,, \tag{4.15}$$

with $\mathbf{s}_{\mathrm{a}}$ and $\mathbf{s}_{\mathrm{u}}$ the speech envelopes of the attended and unattended speaker. Using (4.14) and the definition of the Pearson correlation coefficient of two random variables X and Y:

$$\rho(X,Y) = \frac{\mathbb{E}\{(X - \mu_X)\,(Y - \mu_Y)\}}{\sigma_X \sigma_Y},$$

with the mean $\mu_{X/Y}$ and standard deviation $\sigma_{X/Y}$, (4.15) becomes:

$$\begin{aligned}
\phi(p_i) &= P(p_i \sigma_{\hat{\mathbf{s}}_{\mathrm{a}}} \rho(\hat{\mathbf{s}}_{\mathrm{a}}, \mathbf{s}_{\mathrm{a}}) + (1 - p_i)\,\sigma_{\hat{\mathbf{s}}_{\mathrm{u}}} \rho(\hat{\mathbf{s}}_{\mathrm{u}}, \mathbf{s}_{\mathrm{a}}) \\
&\qquad > p_i \sigma_{\hat{\mathbf{s}}_{\mathrm{a}}} \rho(\hat{\mathbf{s}}_{\mathrm{a}}, \mathbf{s}_{\mathrm{u}}) + (1 - p_i)\,\sigma_{\hat{\mathbf{s}}_{\mathrm{u}}} \rho(\hat{\mathbf{s}}_{\mathrm{u}}, \mathbf{s}_{\mathrm{u}})) \\
&= P(p_i \sigma_{\hat{\mathbf{s}}_{\mathrm{a}}} \left(\rho(\hat{\mathbf{s}}_{\mathrm{a}}, \mathbf{s}_{\mathrm{a}}) - \rho(\hat{\mathbf{s}}_{\mathrm{a}}, \mathbf{s}_{\mathrm{u}})\right) \\
&\qquad > (1 - p_i)\,\sigma_{\hat{\mathbf{s}}_{\mathrm{u}}} \left(\rho(\hat{\mathbf{s}}_{\mathrm{u}}, \mathbf{s}_{\mathrm{u}}) - \rho(\hat{\mathbf{s}}_{\mathrm{u}}, \mathbf{s}_{\mathrm{a}})\right)).
\end{aligned} \tag{4.16}$$

To simplify this expression, and without loss of generality[2], we assume that both speech envelopes have a similar energy content such that it is safe to assume that, on average, $\sigma_{\hat{\mathbf{s}}_{\mathrm{a}}} = \sigma_{\hat{\mathbf{s}}_{\mathrm{u}}}$. Furthermore, $\rho(\hat{\mathbf{s}}_{\mathrm{a}}, \mathbf{s}_{\mathrm{a}})\,, \rho(\hat{\mathbf{s}}_{\mathrm{a}}, \mathbf{s}_{\mathrm{u}})\,, \rho(\hat{\mathbf{s}}_{\mathrm{u}}, \mathbf{s}_{\mathrm{u}})\,,$ and $\rho(\hat{\mathbf{s}}_{\mathrm{u}}, \mathbf{s}_{\mathrm{a}})$ are independent of p_i and can be considered as random variables $\rho_{\mathrm{aa}}, \rho_{\mathrm{au}}, \rho_{\mathrm{uu}},$ and ρ_{ua}. These random variables represent the correlation coefficients between the reconstructed envelopes using the attended/unattended decoders and the speech envelopes of the attended/unattended speakers, computed over a predefined window length. As such, (4.16) becomes:

$$\phi(p_i) = P\left(\rho_{\mathrm{aa}} - \rho_{\mathrm{au}} > \frac{1 - p_i}{p_i}\left(\rho_{\mathrm{uu}} - \rho_{\mathrm{ua}}\right)\right). \tag{4.17}$$

[2] This can always be obtained by normalizing the (reconstructed) envelopes.

Define now the new random variables $R_1 = \rho_{aa} - \rho_{au} \sim \mathcal{N}(\mu_1, \sigma^2)$ and $R_2 = \rho_{uu} - \rho_{ua} \sim \mathcal{N}(\mu_2, \sigma^2)$. We assume that these random variables are normally distributed[3] with known mean and equal standard deviation. These means and standard deviation can be derived a priori from the supervised subject-specific decoders and experiments (note that these are not available in the unsupervised case, yet for analysis and validation purposes, we can use a supervised setting to estimate these). R_1 represents the difference between the correlation coefficients of both competing speakers when using the (supervised) attended decoder, while R_2 would be used when making AAD decisions based on the (supervised) unattended decoder. As the standard deviation of R_1 and R_2 is mostly determined by the noise, which is the same for the attended and unattended decoder, we can assume that they have the same standard deviation σ. This standard deviation can be estimated across the mean-centered $\tilde{R}_1 = R_1 - \mu_1$ and $\tilde{R}_2 = R_2 - \mu_2$ variables.

Finally, we can define $Z = R_1 - \frac{1-p_i}{p_i} R_2$, which is again normally distributed:

$$Z \sim \mathcal{N}\left(\mu_z(p_i), \sigma_z(p_i)^2\right),$$

with

$$\mu_z(p_i) = \mu_1 - \frac{1-p_i}{p_i}\mu_2 \text{ and } \sigma_z(p_i) = \sigma\sqrt{1 + \frac{(1-p_i)^2}{p_i^2}},$$

assuming that R_1 and R_2 are uncorrelated[4]. Equation (4.17) then becomes equal to $P(Z > 0)$, or equivalently:

$$\phi(p_i) = \frac{1}{\sigma_z(p_i)\sqrt{2\pi}} \int_0^{+\infty} e^{-\frac{1}{2}\left(\frac{x-\mu_z(p_i)}{\sigma_z(p_i)}\right)^2} dx. \tag{4.18}$$

By numerically evaluating (4.18) for $p_i \in [0, 100]\%$, we have modeled the AAD accuracy p_{i+1} in iteration $i+1$ as a function of the AAD accuracy p_i in iteration i. Note that p_i and $p_{i+1} = \phi(p_i)$ refer here to the *test* accuracy, as the model parameters will be computed from the correlation coefficients resulting from applying the subject-specific attended/unattended decoders to left-out test data.

Figure 4.3 shows the modeled curve $\phi(p_i)$ where μ_1, μ_2, and σ are estimated from Dataset A. The modeling is performed per subject based on the correlation coefficients of the attended and unattended decoders tested on 60 s decision

[3] For none of the 16 subjects in Dataset A, the Kolmogorov-Smirnov test indicates a deviation from a normal distribution, which provides empirical support for this assumption, in addition to the validation of the final model that we provide in Section 4.5.1.

[4] For none of the 16 subjects in Dataset A, there is a significant correlation between R_1 and R_2, which supports this assumption, in addition to the validation of the final model in Section 4.5.1.

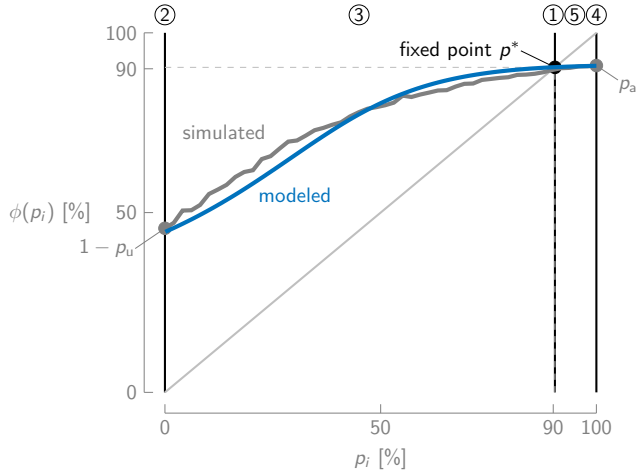


Figure 4.3: The modeled updating curve, averaged over all subjects of Dataset A, shows the accuracy $\phi(p_i)$ after updating, starting from a decoder with accuracy p_i, and closely corresponds to the simulated curve. As a reference, the identity line, where the updated accuracy is equal to the initial accuracy, is added.

windows with ten-fold CV. The modeled curves are then averaged across all subjects to obtain one 'universal' updating curve in Figure 4.3.

Verification of the $\phi(p_i)$ model

The updating curve in Figure 4.3 can be verified using simulations. Consider an oracle that can produce any mixture $(p_i, 100\% - p_i)$ of correct and incorrect labels. Using this oracle, we can perform a sweep of p_i values and compute a decoder based on this particular ratio of correct and incorrect labels. For each p_i, the corresponding decoder can be applied to the test set to evaluate p_{i+1}, which should be approximately equal to $\phi(p_i)$ if the model is correct. The simulated curve shown in Figure 4.3 is generated using random ten-fold CV, repeated five times per subject, and averaged over subjects, folds, and runs. As the simulated curve closely resembles the theoretical curve, we can confirm that the assumptions are sensible and that the theoretical updating curve (4.18) is valid and useful for interpretation and analysis.

4.5.2 Explaining the updating

Analysis of the updating curve

In Figure 4.3, five points/regions are indicated, which are discussed below:

- **Point ①** corresponds to $p_i = p^*$, i.e., the cross-over point. For initial accuracy p^*, the updated accuracy remains the same, i.e., $\phi(p^*) = p^*$. This cross-over point thus corresponds to the *fixed/invariant* point of $\phi(p_i)$.
- **Point ②** corresponds to $p_i = 0\%$, i.e., the decoder is trained using *only* the unattended ground-truth labels and is thus equal to $\hat{\mathbf{d}}_{\mathrm{u}}$. The updated accuracy then corresponds to $100\% - p_{\mathrm{u}}$, as the *unattended* decoder is used to predict *attended* labels. The unattended decoder generally performs worse than the attended decoder, obtaining accuracies below 100%, such that $\phi(0\%) > 0\%$, ergo, an increase in accuracy. This, furthermore, also confirms that unattended speech envelope is encoded differently in the brain than the attended speech envelope.
- **Region ③** corresponds to $0\% \leq p_i < p^*$. In this region, the accuracy increases after updating, i.e., $\phi(p_i) > p_i$. Even when using a majority of *unattended* speech envelopes to train the *attended* decoder, the accuracy increases. A possible explanation is that the resulting correlation vector still conveys information about which channels and which time lags are best suited to decode speech from the EEG, albeit unattended speech. It seems that there is still information to gain from unattended speech to compensate for the limited amount of attended speech. However, when p_i increases, the increase in accuracy in general decreases (i.e., the distance to the identity line decreases), possibly because there is less and less information to gain from the unattended speech. Furthermore, it is expected that the crosscorrelation of the EEG with the attended speech envelopes ($\hat{\mathbf{r}}_{xs_{\mathrm{a}}}$) is on average larger than of the EEG with the unattended speech ($\hat{\mathbf{r}}_{xs_{\mathrm{u}}}$). This reduces the relative weight of the unattended crosscorrelation vector (for example, see (4.12)) and could make the attended crosscorrelation vector more prominent in the estimated one, even when more unattended labels are used, enabling the self-leveraging effect.
- **Point ④** corresponds to $p_i = 100\%$, i.e., the decoder corresponds to the supervised subject-specific decoder from Figure 4.5a, with accuracy p_{a}. As even the attended decoder is not perfect, $\phi(100\%) < 100\%$, which results in a decrease in accuracy. This could be due to modeling errors (limited

capacity of a linear model), the low SNR of the stimulus-response in the EEG, and a small amount of incorrect ground-truth labels, for example, due to the subject's attention wandering off to the wrong speaker.

- **Region ⑤** corresponds to $p^* < p_i < 100\%$, where the accuracy decreases after updating, i.e., $\phi(p_i) < p_i$. The presence of unattended labels does not add information as in region ③, suffering from the same limitations as in point ④.

Lastly, because of the linearity of (4.3), the point $p_i = 50\%$ reflects the case where one would train the decoder based on the sum of both speech envelopes (i.e., across attended and unattended speaker). As discussed in Section 4.3, we implicitly assume that the attended and unattended speech envelopes are encoded differently in the brain. If not, the unsupervised training of a decoder based on the sum of the speech envelopes would result in a similar accuracy as the proposed unsupervised training method. The updating curve in Figure 4.3, however, shows that $\phi(50\%) < \phi(p^*)$. This indicates that such an unsupervised decoder trained on the sum of the speech envelopes performs worse than the proposed unsupervised method. As such, it confirms the assumption that both speech envelopes are encoded distinctly in the brain and that the inclusion of the unattended envelope misdirects the computation of the crosscorrelation vector in (4.3).

A fixed-point iteration algorithm

Using the theoretical model in Figure 4.3, we can interpret the unsupervised AAD algorithm in Algorithm 2 as a *fixed-point iteration* $p_{i+1} = \phi(p_i)$ on this curve. Before analyzing the uniqueness and convergence properties based on the model (4.18), we first provide an intuitive explanation of why there could only be *one* fixed point p^* on the updating curve. First of all, it is safe to assume that $\phi(0\%) > 0\%$, as the unattended decoder is never perfect. Furthermore, it is very unlikely that regions ③ and ⑤ in Figure 4.3 would alternate, as this would mean that, when using more attended labels to train the decoder, there is an increase-decrease-increase of AAD accuracy (or the other way around) with respect to the initial accuracy. This implies that there is a unique fixed point of the theoretical model. We show in Appendices 4.A and 4.B that, based on the model (4.18), the existence, uniqueness, and convergence of/to the fixed point are indeed mathematically guaranteed when three reasonable conditions on the accuracy p_{a} of the (supervised) attended decoder and the accuracy p_{u} of the (supervised) unattended decoder (on the unattended speech) are satisfied. Furthermore, we also demonstrate in Appendix 4.B that these conditions are satisfied for all subjects in both datasets.

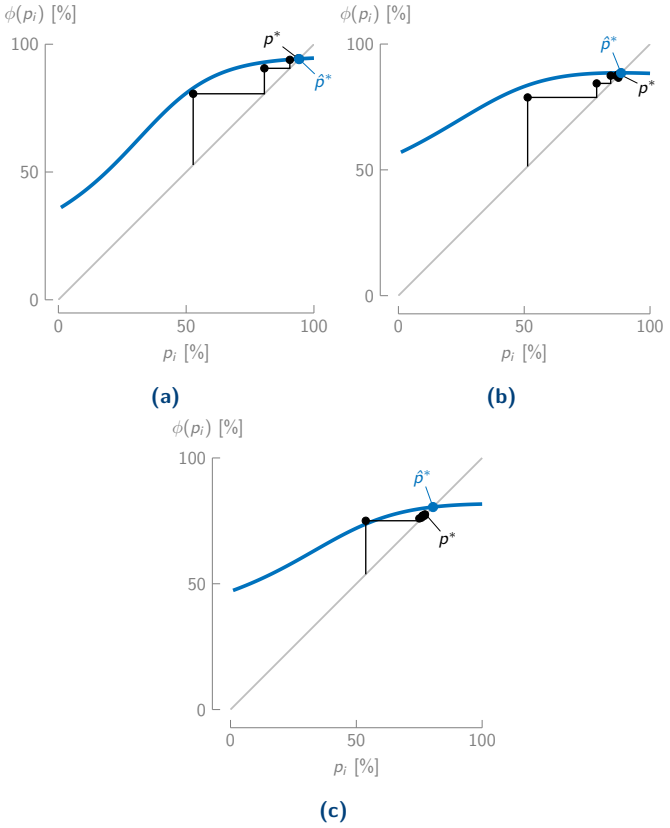


Figure 4.4: The realized fixed-point iteration paths closely follow the theoretical model (for three representative subjects **(a)**, **(b)**, **(c)** from Dataset A). The predicted fixed point $\hat{p}^*$ from the theoretical model accurately predicts the actual fixed point p^*.

These fixed-point iteration properties are also intuitively apparent from Figure 4.3 and hold in every example we have encountered in practice so far. This means that we could initialize the updating algorithm with *any* decoder, as we would always arrive at (or very close to) the fixed point p^*[5]. As a result, it explains why the updating procedure is possible starting from a random decoder. Figure 4.4 shows how the fixed-point paths (on average across all folds) follow the theoretical model for three representative subjects of Dataset A, starting from a random decoder.

[5]The fixed point to which there is convergence will, in practice, slightly differ across runs, initializations, etc. The model in Figure 4.3 should be interpreted in a probabilistic manner, as it represents an average across runs, initializations, etc.

The fixed point $\hat{p}^*$ based on the theoretical model (where the means and standard deviation in (4.18) are computed per subject individually) should thus give a good approximation of the unsupervised AAD accuracy p^*. Across all 16 subjects of Dataset A, on 60 s decision windows, the mean absolute error between the predicted and actual unsupervised AAD accuracy is 3.45%. We can thus accurately predict how well the unsupervised updating will perform by computing the fixed point of (4.18), where the parameters μ_1, μ_2, and σ in (4.18) can be easily computed from the corresponding *supervised* subject-specific decoders. Furthermore, as mentioned above, the model (4.18) also allows showing convergence to this fixed point when three reasonable conditions are satisfied (see Appendices 4.A and 4.B).

4.6 Results and discussion

In this section, we extensively validate the unsupervised algorithm on the two datasets and compare it with a supervised subject-independent and supervised subject-specific decoder.

4.6.1 Random initialization

We first evaluate the proposed unsupervised algorithm using a random initialization and without using any prior knowledge. As such, in Algorithm 2, we set $\alpha = 0$ and $\beta = 0$. The crosscorrelation vector $\mathbf{r}_{xs_a}^{(\text{init})}$ is initialized at random from a multivariate uniform distribution. Figure 4.5 shows for both datasets the AAD accuracy as a function of decision window length and Figure 4.6 the MESD values per subject for the supervised subject-specific decoder, the subject-independent decoder, and the proposed unsupervised subject-specific decoder (with random initialization). The significance level in Figure 4.5 is computed using the inverse binomial distribution as in [74].

As mentioned in Section 4.1, it is clear that a supervised subject-specific decoder outperforms a subject-independent decoder on both datasets (Figures 4.5 and 4.6). A Wilcoxon signed-rank test between the MESD values, with a Bonferroni-Holm correction for multiple comparisons, confirms this (Dataset A: $n = 16, p = 0.0022$, Dataset B: $n = 18, p = 0.0030$). On both datasets, the proposed unsupervised subject-specific decoder with random initialization outperforms the subject-independent decoder as well (although less clearly on Dataset B). Furthermore, it approximates the performance of the supervised subject-specific decoder, especially for the shorter decision window lengths. However, it does so without requiring ground-truth labels and thus retains the

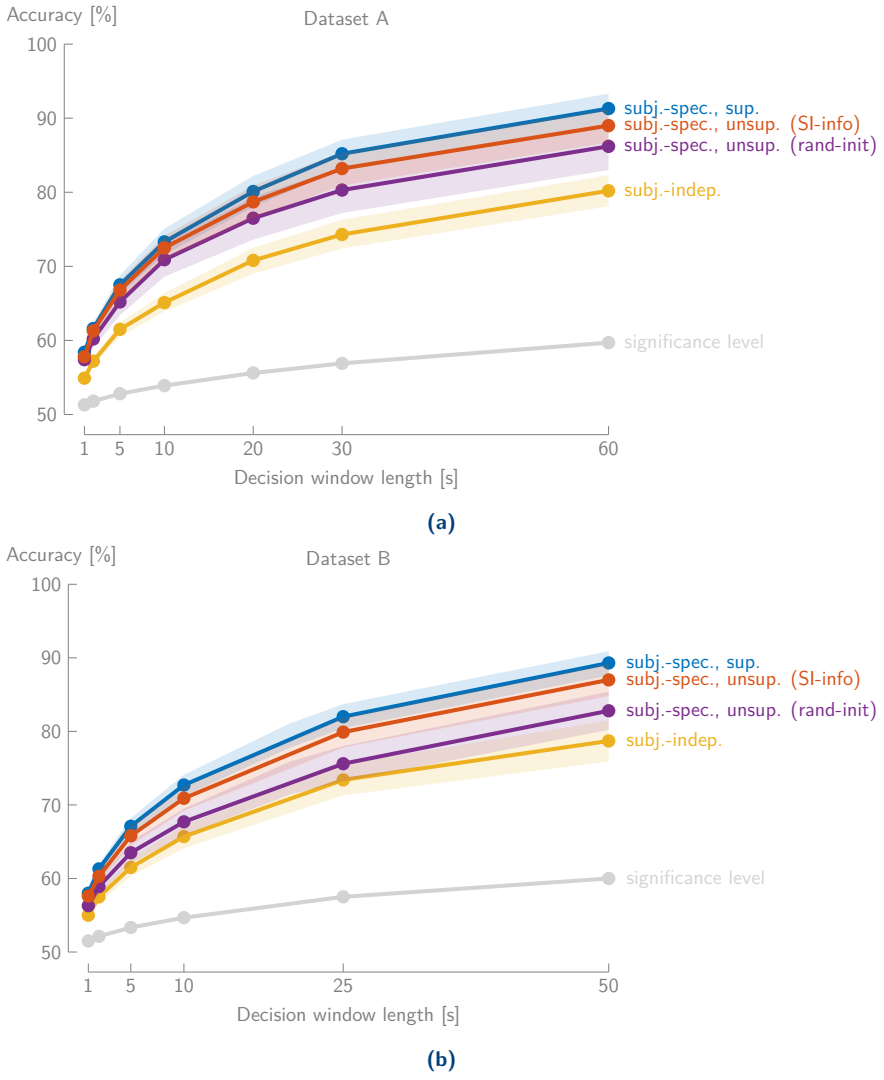


Figure 4.5: **(a)** The unsupervised subject-specific decoder, with both types of initialization (random: rand-init, subject-independent information: SI-info) clearly outperforms a subject-independent decoder on Dataset A, while approximating the performance of a supervised subject-specific decoder especially on short decision windows (mean ± standard error of the mean (shading) across subjects). **(b)** The same trend occurs for Dataset B, although the unsupervised subject-specific decoder with random initialization outperforms the subject-independent decoder less apparent.

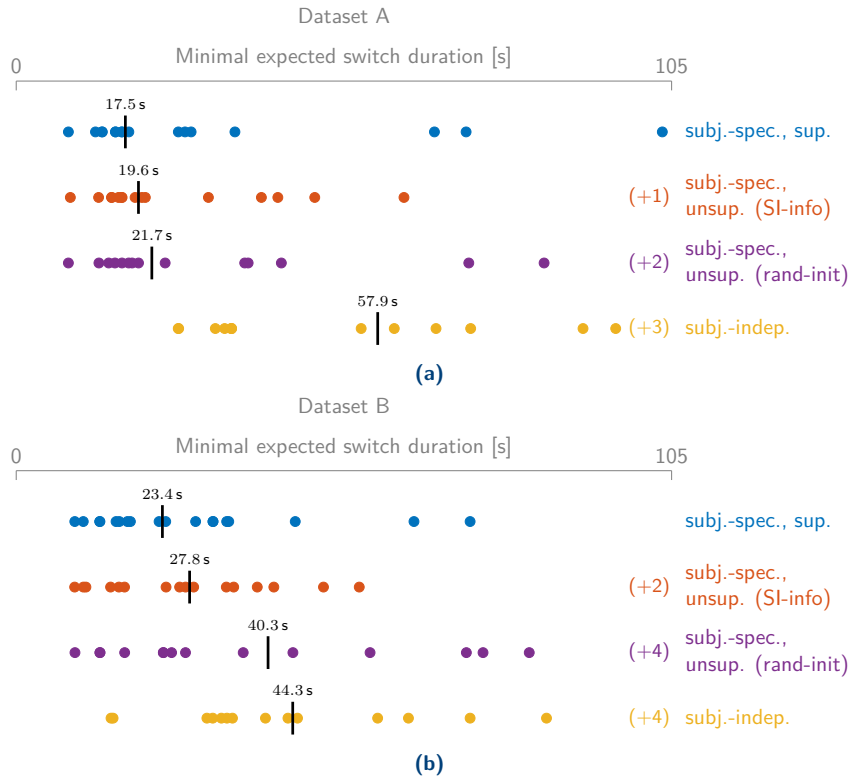


Figure 4.6: **(a)** The per-subject MESD values (each subject = one dot) of Dataset A, with the median indicated with the black bar, confirm that the unsupervised subject-specific decoder outperforms the subject-independent decoder. The number of outlying values that fell off the plot is indicated with (+x) (outliers are still included in the quantitative analysis). 'SI' = subject-independent **(b)** The same for Dataset B.

'plug-and-play' feature of the subject-independent decoder. A Wilcoxon signed-rank test between the MESD values, again with a Bonferroni-Holm correction, shows a significant difference between the unsupervised subject-specific decoder with random initialization and the supervised subject-independent decoder on Dataset A ($n = 16, p = 0.0458$), but not on Dataset B ($n = 18, p = 0.5862$). Lastly, there is a significant difference between the supervised and unsupervised subject-specific decoder with random initialization (Dataset A: $n = 16, p = 0.0034$, Dataset B: $n = 18, p = 0.0010$).

Note that this last result is not per se a negative result: it is not expected that an unsupervised subject-specific decoder, updated starting from a completely

random decoder, performs as well as the supervised version. The most important result is that the proposed unsupervised algorithm outperforms a subject-independent decoder, *even* when starting from a random decoder and while not requiring subject-specific ground-truth labels as well. Furthermore, such an unsupervised algorithm could be implemented on a generic hearing device, which trains and adapts itself from scratch to a new user.

Convergence plots

Figure 4.7 shows the AAD accuracy as a function of the iteration index for all subjects of Dataset A. Computing a decoder with the subject-specific autocorrelation matrix, but with a random crosscorrelation vector, seems not to perform better than chance (iteration 0). Surprisingly, even after one iteration of predicting the labels using the decoder after iteration 0, which performs on chance level, and updating the crosscorrelation vector, a decoder is obtained that on average performs with $\approx 75\%$ accuracy on 60 s decision windows (see also Figure 4.3). This implies that even using a random mix of attended and unattended labels results in a decoder that performs much better than chance. In the following iterations, the decoder keeps improving, settling after 4-5 iterations. This matches the fixed-point iteration interpretation of Section 4.5.2 and Figures 4.3 and 4.4, explaining the self-leveraging mechanism.

4.6.2 Subject-independent initialization/information

To use the information in the subject-independent decoder to our advantage, we can put $\alpha \neq 0$ and $\beta \neq 0$ in Algorithm 2. By adding subject-independent information to the estimation of both the autocorrelation matrix and the crosscorrelation vector, we can further improve the updating behavior when starting from a random initialization. Especially in the estimation of the crosscorrelation vector, the subject-independent crosscorrelation vector, which is estimated using ground-truth labels, can compensate for prediction errors.

The initial autocorrelation matrix $\mathbf{R}_{xx}^{(\text{init})}$ and crosscorrelation vector $\mathbf{r}_{xs_\text{a}}^{(\text{init})}$ are determined using the (supervised) information of all *other* subjects. The hyperparameters α and β are determined empirically. For Dataset A, $\alpha = 0$ is chosen, i.e., no subject-independent information is used in the autocorrelation estimation. Furthermore, $\beta = \frac{1}{3}$ is chosen, i.e., the subject-independent crosscorrelation is half as important as the computed subject-specific one.

The results on Dataset A of this unsupervised subject-specific decoder using subject-independent information are shown in Figures 4.5a and 4.6a.

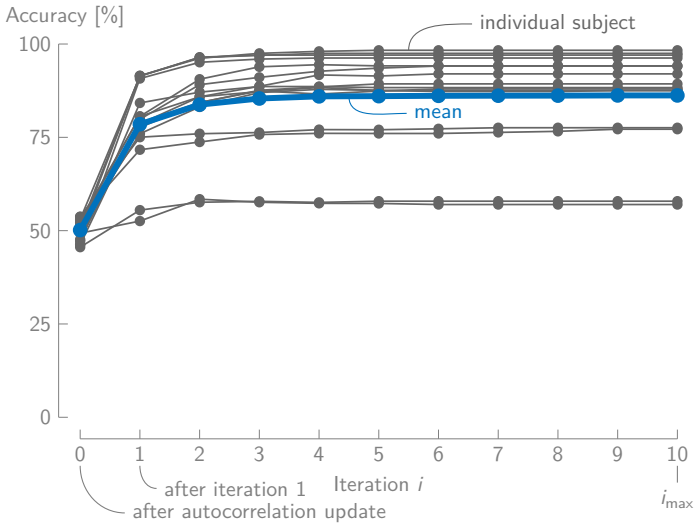


Figure 4.7: The convergence plots for all subjects of Dataset A using a random initialization, on 60 s decision windows, show that the AAD accuracy converges to the final unsupervised subject-specific accuracy after 4-5 iterations.

Remarkably, the unsupervised procedure here results in a decoder that very closely approximates the supervised subject-specific decoder, *without* requiring subject-specific ground-truth labels. Based on the MESD values, there is no significant difference to be found between the supervised and unsupervised subject-specific decoder with subject-independent information (Wilcoxon signed-rank test with Bonferroni-Holm correction: $n = 16, p = 0.3259$). For six subjects, the unsupervised decoder performs even better than the supervised subject-specific one (see also Figure 4.6a). Furthermore, note that using the subject-independent information with respect to a random initialization and no further information not only fixes poor updating results for some of the outlying subjects but also improves on most other subjects (12 out of 16).

For Dataset B, $\alpha = 0.5$ and $\beta = 0.5$, i.e., an equal weight to the subject-specific and subject-independent information, turn out to be good choices. Given that the unsupervised subject-specific decoder with random initialization performs worse than in Dataset A, it is not unexpected that a larger weight β of the subject-independent information is required to improve on the unsupervised procedure.

Figures 4.5b and 4.6b show the results on Dataset B of the unsupervised procedure with subject-independent information and with the aforementioned

choices of the hyperparameters. The usage of subject-independent information results here in an even larger improvement over the random initialization (e.g., both in MESD, for 15 out of 18 subjects, as spread around the median in Figure 4.6b) and again closely approximates the supervised subject-specific performance, without requiring subject-specific ground-truth labels. However, based on the MESD values in Figure 4.6b, there is still a significant difference to be found between the supervised and unsupervised subject-specific performance (Wilcoxon signed-rank test with Bonferroni-Holm correction: $n = 18, p = 0.0498$), albeit very close to the significance level of 0.05. This indicates again that the unsupervised procedure with subject-independent information closely approximates the supervised subject-specific performance *without* ground-truth labels. Furthermore, the unsupervised decoder has a higher performance for four subjects (out of 18) relative to the supervised subject-specific decoder. Lastly, there now is a clear significant difference between the MESD values of the unsupervised procedure and the subject-independent decoder (Wilcoxon signed-rank test with Bonferroni-Holm correction: $n = 18, p = 0.0030$).

Using some information about other subjects, we can thus adapt a stimulus decoder that performs almost as well as a supervised subject-specific decoder, but without requiring ground-truth information about the attended speaker during the training procedure.

Convergence plots

Figure 4.8 shows the AAD accuracy as a function of the different steps of Algorithm 2 for all subjects of Dataset A. It appears that fully replacing (i.e., $\alpha = 0$) the autocorrelation matrix in the subject-independent decoder with the subject-specific information, which is a fully unsupervised step, already results in a substantial increase in accuracy, despite the resulting mismatch between the auto- and crosscorrelation matrix/vector ('after autocorrelation update' versus 'subj.-indep.' in Figure 4.8). Further updating the crosscorrelation vector with the predicted labels while using subject-independent information with $\beta = \frac{1}{3}$ results in a self-leveraging effect, leading to a further increase in accuracy, which converges after a few iterations similarly to Figure 4.7.

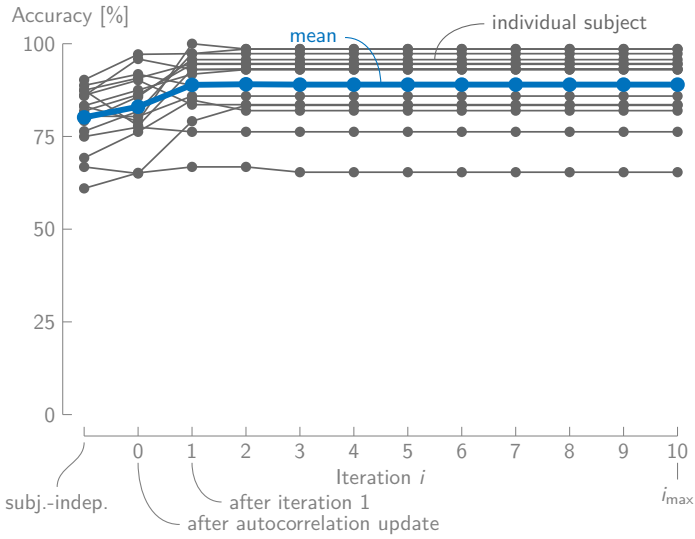


Figure 4.8: The convergence plots for all subjects of Dataset A using subject-independent information, on 60 s decision windows, show that mainly the autocorrelation update and the first iteration result in a substantial increase in accuracy.

4.7 Outlook and conclusion

4.7.1 Applications and future work

The proposed unsupervised self-adaptive algorithm paves the way for further extensions and applications. We presented a batch-version of the algorithm, i.e., the updating is performed on a large dataset of EEG and audio data. This enables the 'plug-and-play' capabilities of a stimulus decoder for a new hearing device user. However, Algorithm 2 could be extended to a time-adaptive version, tailored towards the application of neuro-steered hearing devices, where EEG and audio data are continuously recorded. As a result, the stimulus decoder could automatically update in an unsupervised manner when new data comes in and adapt to changing conditions and situations (e.g., non-stationarities in neural activity, changing electrode-skin contact impedances). Such an efficient, adaptive version of the unsupervised procedure is developed in Chapter 5.

The deployed SR approach performs worse on short decision window lengths (Figure 4.5), making this algorithm less suitable for real-time decoding of the

auditory attention (Part I). However, the proposed unsupervised updating of a stimulus decoder can still be used on a longer time scale to generate reliable labels to train another, potentially more accurate, algorithm on short decision windows (for example, the algorithms in Part III). Lastly, the adaptive implementation of the unsupervised procedure also potentially enables and improves the success of neurofeedback effects in a closed-loop implementation (see also Section 8.2).

4.7.2 Conclusion

We have shown that it is possible to train a subject-specific stimulus decoder for AAD using an unsupervised procedure, i.e., without requiring information about which speaker is the attended or unattended one. Training such a decoder on the data of a particular subject from scratch, even starting from a random decoder and without any prior knowledge, leads to a decoder that outperforms a subject-independent decoder. Unsupervised adaptation of a subject-independent decoder, trained on other subjects, to a new subject even leads to a decoder that closely approximates the performance of a supervised subject-specific decoder. The proposed updating algorithm thus combines the two main advantages of a supervised subject-specific and subject-independent decoder:

1. It substantially outperforms a subject-independent decoder, approximating the performance of a supervised subject-specific decoder.
2. It can be used in a 'plug-and-play' fashion, without requiring ground-truth labels and potentially automatically adapting to changing conditions without external intervention.

Using a mathematical model for the updating procedure, the unsupervised algorithm can be interpreted as a fixed-point algorithm. This interpretation explains why there is a self-leveraging effect, even when starting from a random decoder. Furthermore, using this mathematical model, we are able to accurately predict the accuracy of the unsupervised decoder starting from the results of the supervised subject-specific decoder.

The proposed unsupervised self-adaptive algorithm can be used in an online and adaptive manner in a practical neuro-steered hearing device, allowing the decoder to automatically adapt to the non-stationary brain and changing environments and conditions (see Chapter 5). Furthermore, it avoids having a cumbersome a priori training stage for each new hearing device user, as it automatically adapts to the new user. Lastly, the developed method potentially

enables stronger neurofeedback effects when using a closed-loop system, which is paramount for the successful application of AAD (see Section 8.2).

Appendices

In the appendices, we show convergence to a unique fixed point of the fixed-point iteration on the updating model (4.18). We hypothesize that under three reasonable conditions on the accuracies of the attended and unattended decoder, there exists a unique fixed point p^* to which the fixed-point iteration $p_{i+1} = \phi(p_i)$ converges, starting from any (possibly random) decoder. In Section 4.A, we first show that there always exists such a fixed point, while in Section 4.B, we check the uniqueness of and convergence to this fixed point under the hypothesized conditions.

4.A Existence

Consider the following fixed-point theorem, also known as Brouwer's fixed-point theorem [166]:

Theorem 1 (Brouwer's fixed point theorem [166])

Any continuous self map of a nonempty compact convex subset of a Euclidean space has a fixed point.

As the function $\phi(p_i)\colon [0,100]\% \to [0,100]\%$ in (4.18) is a continuous function that maps its domain onto itself and $[0,1]$ is a closed (thus, compact) convex subset of $\mathbb{R}$, Brouwer's fixed point theorem assures that there exists at least one fixed point.

4.B Uniqueness and convergence

We evaluate the model in (4.18) in a relevant range of the parameters μ_1, μ_2, and σ, obeying three reasonable conditions, to show the convergence to a unique fixed point.

Three conditions for convergence

Consider the supervised subject-specific attended decoder $\hat{\mathbf{d}}_\mathrm{a}$ with accuracy p_a (on the attended labels) and supervised subject-specific unattended decoder $\hat{\mathbf{d}}_\mathrm{u}$ with accuracy p_u (on the unattended labels). We then a priori postulate the

following three intuitive and reasonable conditions on the accuracies p_a and p_u (which will turn out to be satisfied for all subjects in both datasets):

- $p_a - p_u > 5\%$, i.e., the attended decoder needs to perform 5% better (on the attended labels) than the unattended decoder (on the unattended labels). Given that the attended speech envelope is typically better represented in the EEG, we indeed expect a difference in performance between both decoders. Moreover, this condition can be linked to the expectation that the crosscorrelation between the EEG and attended speech envelope is on average larger than with the unattended speech envelope, serving as a possible explanation for the self-leveraging effect (Section 4.5.2).
- $p_u < 85\%$, i.e., the *unattended* decoder may not perform better than 85% (on the unattended labels). If the unattended decoder performs too well, then, again, the self-leveraging effect may not be present for the same reason as mentioned in the previous condition.
- $p_a > 100\% - p_u$, i.e., the attended decoder is better at predicting attended labels than the unattended decoder. This assures that the starting point of the model curve $\phi(0\%) = 100\% - p_u$ (for example, see Figure 4.3) is below the end point $\phi(100\%) = p_a$.

In the following sections, we will use the model in (4.18) to show that there is convergence to a unique fixed point of the model when these three conditions are satisfied. However, it is noted that these postulated conditions are conservative in the mathematical sense, i.e., they are 'sufficient' but *not* 'necessary' conditions. When they are not satisfied, there can still be convergence to a unique fixed point.

Moreover, the three conditions are also intuitive and very reasonable from a practical point of view, as they are satisfied for all subjects in both datasets on 50/60 s decision windows (i.e., the length of the segments on which the updating is performed): the minimum across all subjects of $p_a - p_u = 8.3\% > 5\%$, the maximum across all subjects of $p_u = 76.7\% < 85\%$, and the minimum across all subjects of $p_a + p_u = 124\% > 100\%$.

Convergence to a unique fixed point

Consider the following fixed-point theorem that provides sufficient conditions for convergence to a unique fixed point of the fixed-point iteration $p_{i+1} = \phi(p_i)$ [167]:

Theorem 2 [167]

Let ϕ be a continuous function on $[a, b]$, such that $\phi(p_i) \in [a, b], \forall\, p_i \in [a, b]$, and suppose that ϕ' exists $\forall p_i \in [a, b]$ and that a constant $0 < \alpha < 1$ exists such that:

$$|\phi'(p_i)| \leq \alpha, \forall\, p_i \in [a, b],$$

then there is exactly one fixed point $p^* \in [a, b]$ and the fixed-point iteration $p_{i+1} = \phi(p_i)$ will converge to this unique fixed point in $[a, b]$.

We now evaluate the model $\phi(p_i)$ in (4.18) and its derivative $\phi'(p_i)$ to show convergence to a unique fixed point based on Theorem 2 for the case where the conditions in Section 4.B are satisfied.

The derivative $\phi'(p_i)$ of the model in (4.18) can be computed by hand or by using any symbolic math software and is equal to:

$$\phi'(p_i) = \frac{p_i \sigma_z(p_i)^2 \mu_2 + (1 - p_i)\sigma^2 \mu_z(p_i)}{\sqrt{2\pi} p_i^3 \sigma_z(p_i)^3} e^{-\frac{1}{2}\left(\frac{\mu_z(p_i)}{\sigma_z(p_i)}\right)^2}. \tag{4.19}$$

To evaluate (4.18) and its derivative (4.19), we take 300 equidistant samples of $\mu_1 \in [-2, 2]$, 300 equidistant samples of $\mu_2 \in [-2, 2]$, and 100 equidistant samples of $\sigma \in]0, 4]$. These intervals contain the complete range of parameters concerning the difference in correlation coefficients R_1 and R_2. From this parameter range, we select all combinations of (μ_1, μ_2, σ) for which the three conditions of Section 4.B are satisfied. The connection between p_a and p_u (as used in the three conditions) and the model parameters (μ_1, μ_2, σ) is given by:

$$p_\mathrm{a} = P\left(R_1 > 0\right) = \frac{1}{\sigma\sqrt{2\pi}} \int_0^{+\infty} e^{-\frac{1}{2}\left(\frac{x-\mu_1}{\sigma}\right)^2} dx \text{ and}$$

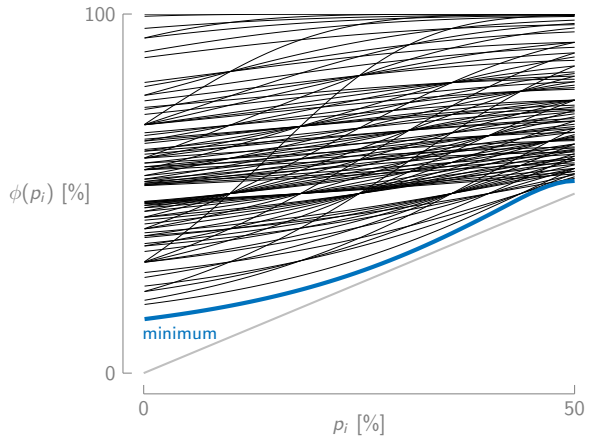
$$p_\mathrm{u} = P\left(R_2 > 0\right) = \frac{1}{\sigma\sqrt{2\pi}} \int_0^{+\infty} e^{-\frac{1}{2}\left(\frac{x-\mu_2}{\sigma}\right)^2} dx,$$

using the assumptions in Section 4.5.1. These connections can be derived from the updating model (4.18) by setting $p_i = 100\%$, resp. $p_i = 0\%$, resulting in the decoder accuracy of the supervised attended, resp. unattended decoder.

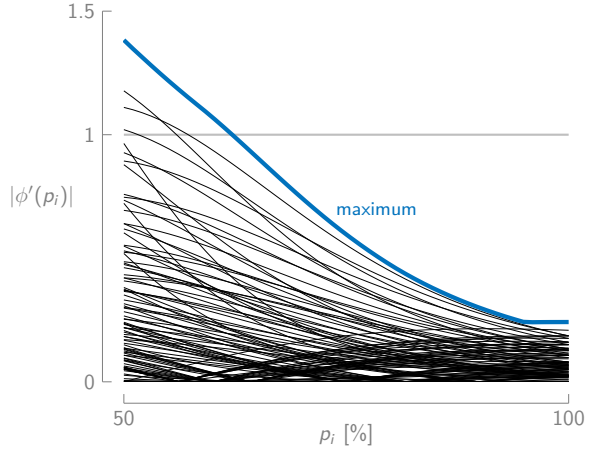
Figure 4.Aa now shows a subset of $\phi(p_i)$ for $p_i \in [0, 50]\%$, for all evaluated (μ_1, μ_2, σ) that obey the three conditions, together with the minimum over all these $\phi(p_i)$. Similarly, Figure 4.Ab shows a subset of $|\phi'(p_i)|$ for $p_i \in [50, 100]\%$, for all evaluated (μ_1, μ_2, σ) that obey the three conditions, together with the

maximum over all these $|\phi'(p_i)|$. Both results are required to show convergence to a unique fixed point using Theorem 2:

- **Result 1**: From Figure 4.Aa, it can be seen that $\phi(p_i) > p_i, \forall p_i \in [0, 50]\%$. This implies that there is no fixed point within this interval and that the fixed-point iteration will always diverge to the $p_i \in [50, 100]\%$ interval. This is because $\forall\, p_i \in [0, 50]\% : p_{i+1} = \phi(p_i) > p_i$, i.e., the new accuracy in the fixed-point iteration is always larger than the previous one, such that, inevitably, at a certain iteration, $p_{i+1} > 50\%$. It thus suffices to show that there is convergence to a unique fixed point for $p_i \in [50, 100]\%$, which is shown in the next result.
- **Result 2**: From Figure 4.Ab, there are two possible cases, which both individually can be shown to guarantee convergence to a unique fixed point:
  1. $|\phi'(p_i)| < 1, \forall p_i \in [50, 100]\%$. For all these cases, we then numerically confirmed that $\phi(p_i) \in [50, 100]\%, \forall\, p_i \in [50, 100]\%$ such that all conditions of Theorem 2 are fulfilled to show convergence to a unique point.
  2. $\exists\, x \in [50, 100]\% : \phi'(p_i) \geq 1, \forall\, p_i \in [50, x]\%$ and $|\phi'(p_i)| < 1, \forall\, p_i \in [x, 100]\%$. Since $\phi(50\%) > 50\%$ (see Result 1) and since the derivative is positive, it is guaranteed that $\phi(p_i) > p_i, \forall\, p_i \in [50, x]\%$, i.e., there is no fixed point and the fixed-point iteration diverges to the $p_i \in [x, 100]\%$ interval (using a similar reasoning as in Result 1). Furthermore, it can again be numerically checked that $\phi(p_i) \in [x, 100]\%, \forall\, p_i \in [x, 100]\%$ to show that there is a unique point to which there is convergence in this interval (see Theorem 2).



(a)



(b)

Figure 4.A: **(a)** A subset of the evaluated $\phi(p_i)$ for $p_i \in [0, 50]\%$ and the minimum over all evaluated (μ_1, μ_2, σ) that obey the conditions are all above the identity line, where $\phi(p_i) = p_i$, which shows that $\phi(p_i) > p_i, \forall\, p_i \in [0, 50]\%$. **(b)** A subset of the evaluated $|\phi'(p_i)|$ for $p_i \in [50, 100]\%$, together with the maximum over all evaluated (μ_1, μ_2, σ) that obey the conditions.

5 | Time-adaptive unsupervised stimulus reconstruction

This chapter is largely based on S. Geirnaert, T. Francart, and A. Bertrand, "Time-adaptive Unsupervised Auditory Attention Decoding Using EEG-based Stimulus Reconstruction," Accepted for publication in *IEEE Journal of Biomedical and Health Informatics*, 2022.

ABSTRACT | In Chapter 4, we proposed an unsupervised AAD algorithm based on a stimulus decoder that does not require a dedicated 'ground-truth' EEG recording of the subject under test during which the attended speaker is known. However, this decoder is still trained on a batch of unlabeled data and remains fixed during operation, and can thus not adapt to changing conditions and situations. Therefore, in this chapter, we propose an online time-adaptive unsupervised SR method that continuously and automatically adapts over time when new EEG and audio data are streaming in. This adaptive decoder does not require ground-truth attention labels obtained from a training session with the end-user and instead can be initialized with a generic subject-independent decoder or even completely random values. We propose two different implementations: a sliding window and recursive implementation, which we extensively validate based on multiple performance metrics on three independent datasets. We show that the proposed time-adaptive unsupervised decoder outperforms a time-invariant supervised decoder, for example, when electrodes are disconnected, or when AAD is performed across multiple recording days. Therefore, the proposed time-adaptive unsupervised SR method represents an important step towards practically applicable AAD algorithms for neuro-steered hearing devices.

5.1 Introduction

In Chapter 4, we proposed an unsupervised algorithm to train a subject-specific stimulus decoder without the need for ground-truth labels. Consequently, one of the main issues with the traditional SR method, i.e., the need for acquiring labeled data during a dedicated training session, is resolved while approximating the performance of a fully supervised subject-specific decoder (Chapter 4). However, this unsupervised decoder is still trained in batch on a large amount of (unlabeled) training data and remains fixed during operation. Such fixed decoders do not adapt to long-term signal changes due to changing conditions and situations (e.g., non-stationarities in the neural activity, changing electrode-skin contact impedances, shifting or loosening electrodes). Therefore, in this chapter, we modify and extend the algorithm proposed in Chapter 4 such that the decoder adapts over time in an unsupervised manner. The resulting decoder does not require a dedicated training session and can automatically adapt to new incoming non-stationary EEG data from the end-user and thus serves as one of the first practical plug-and-play AAD algorithms for neuro-steered hearing devices.

In Section 5.2, we review the (unsupervised) SR algorithm for AAD. In Section 5.3, we then present and explain the proposed time-adaptive updating schemes. These implementations are investigated, and the hyperparameter choices are validated in Section 5.4. The proposed time-adaptive unsupervised decoder is then tested and compared to the fixed (time-invariant) supervised decoder in a time-adaptive context in Section 5.5, by simulating a scenario where electrodes are disconnected and applying the decoders on a dataset recorded across multiple recording days.

5.2 (Un)supervised SR for AAD

5.2.1 Review of SR

Consider a C-channel EEG signal of which the c^{th} channel is denoted by $x_c(t)$, with t the time sample index. In the linear SR paradigm, a spatio-temporal filter or decoder $d_c(l)$ is applied to this C-channel EEG signal to reconstruct the speech envelope of the attended speaker $s_{\text{a}}(t)$ [5, 74, 163] (see also Chapter 3):

$$\hat{s}_{\text{a}}(t) = \sum_{c=1}^{C} \sum_{l=0}^{L-1} d_c(l) x_c(t+l),$$

with the channel index c ranging from 1 to C (spatial combination of C channels) and the time lag index l ranging from 0 to $L-1$ (temporal integration over L time samples). This filter is an anti-causal filter, as L *post-stimulus* time lags are used to reconstruct the attended speech envelope from the EEG signal. In the AAD problem, to identify the attended speaker, the reconstructed envelope $\hat{s}_{\mathrm{a}}(t)$ from the EEG is compared with original speech envelopes $s_1(t)$ and $s_2(t)$ of the two simultaneously talking speakers through the Pearson correlation coefficient. For the sake of an easy exposition but without loss of generality, we here assume only two competing speakers, although all presented algorithms and procedures can be extended to more speakers.

In the remainder of the chapter, we will adopt a matrix-vector notation, in which the decoder is written as

$$\mathbf{d} = \begin{bmatrix} d_1(0) & d_1(1) & \cdots & d_1(L-1) & d_2(0) & \cdots & d_C(L-1) \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^{CL}.$$

Assume (for now) the availability of K training segments of T time samples, where the available training information is described as $\{\mathbf{X}_k, (\mathbf{s}_{1_k}, \mathbf{s}_{2_k}), y_k\}_{k=1}^{K}$, containing an EEG data $\mathbf{X}_k$ matrix (collecting all T time samples in training segment k; a rigorous definition is given in (5.3)), speech envelopes $\mathbf{s}_{1_k} \in \mathbb{R}^T$ and $\mathbf{s}_{2_k} \in \mathbb{R}^T$ (similarly), and attention labels $y_k \in \{1, 2\}$, indicating which speech envelope ($\mathbf{s}_{1_k}$ or $\mathbf{s}_{2_k}$) is the attended one ($\mathbf{s}_{\mathrm{a}_k}$) (assuming constant attention across the whole segment). For each training segment k, the attended speech envelope is determined as

$$\mathbf{s}_{\mathrm{a}_k} = \begin{cases} \mathbf{s}_{1_k} & \text{if } y_k = 1, \\ \mathbf{s}_{2_k} & \text{if } y_k = 2. \end{cases} \tag{5.1}$$

The decoder is then trained by minimizing the squared error between the actual attended and reconstructed speech envelope across all training segments:

$$\hat{\mathbf{d}} = \underset{\mathbf{d}}{\operatorname{argmin}} \sum_{k=1}^{K} ||\mathbf{s}_{\mathrm{a}_k} - \mathbf{X}_k \mathbf{d}||_2^2 = \underset{\mathbf{d}}{\operatorname{argmin}} \; ||\mathbf{s}_{\mathrm{a}} - \mathbf{X}\mathbf{d}||_2^2 \,, \tag{5.2}$$

with $\mathbf{s}_{\mathrm{a}} = \begin{bmatrix} \mathbf{s}_{\mathrm{a}_1}^{\mathrm{T}} & \cdots & \mathbf{s}_{\mathrm{a}_K}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^{KT}$ the concatenated actual attended speech envelope and where the block Hankel matrix $\mathbf{X} \in \mathbb{R}^{KT \times CL}$ represents the concatenated time-lagged C-channel EEG with L time lags:

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \vdots \\ \mathbf{X}_K \end{bmatrix}, \mathbf{X}_k = \begin{bmatrix} \mathbf{X}_{k,1} & \cdots & \mathbf{X}_{k,C} \end{bmatrix} \in \mathbb{R}^{T \times CL}, \tag{5.3}$$

with $\mathbf{X}_{k,c} \in \mathbb{R}^{T \times L}$ a Hankel matrix containing the time-lagged EEG data of the k^{th} training segment and c^{th} EEG channel:

$$\mathbf{X}_{k,c} = \begin{bmatrix} x_{k,c}(0) & x_{k,c}(1) & \cdots & x_{k,c}(L-1) \\ x_{k,c}(1) & x_{k,c}(2) & \cdots & x_{k,c}(L) \\ \vdots & \vdots & & \vdots \\ x_{k,c}(T-1) & 0 & \cdots & 0 \end{bmatrix}.$$

$\mathbf{X}\hat{\mathbf{d}} = \hat{\mathbf{s}}_{\text{a}} \in \mathbb{R}^{KT}$ then represents the reconstructed speech envelope over all training segments. The solution of (5.2) is found by solving the normal equations:

$$\hat{\mathbf{d}} = \mathbf{R}_{xx}^{-1}\mathbf{r}_{xs}, \tag{5.4}$$

with

$$\mathbf{R}_{xx} = \mathbf{X}^{\text{T}}\mathbf{X} = \sum_{k=1}^{K} \mathbf{X}_k^{\text{T}}\mathbf{X}_k \in \mathbb{R}^{CL \times CL}$$

the estimated EEG autocorrelation matrix and

$$\mathbf{r}_{xs} = \mathbf{X}^{\text{T}}\mathbf{s}_{\text{a}} = \sum_{k=1}^{K} \mathbf{X}_k^{\text{T}}\mathbf{s}_{\text{a}_k} \in \mathbb{R}^{CL} \tag{5.5}$$

the estimated crosscorrelation vector between the EEG and the attended speech envelope. It is important to notice that only in (5.5) we need the attention labels y_k to select the attended speech envelope $\mathbf{s}_{\text{a}_k}$ in segment k (see (5.1)). We use shrinkage to regularize the estimated autocorrelation matrix:

$$\mathbf{R}_{xx} = (1-\lambda)\mathbf{X}^{\text{T}}\mathbf{X} + \lambda\frac{\text{Tr}\left(\mathbf{X}^{\text{T}}\mathbf{X}\right)}{CL}\mathbf{I}, \tag{5.6}$$

with $\mathbf{I} \in \mathbb{R}^{CL \times CL}$ the identity matrix and where the shrinkage parameter $0 \leq \lambda \leq 1$ is analytically determined [164, 165]:

$$\lambda = \min\left(\frac{\sum_{k=1}^{K}\sum_{t=1}^{T}\left|\left|\mathbf{x}_{k,t}\mathbf{x}_{k,t}^{\text{T}} - \frac{1}{KT}\mathbf{X}^{\text{T}}\mathbf{X}\right|\right|_{\text{F}}^{2}}{\text{Tr}\left(\left(\mathbf{X}^{\text{T}}\mathbf{X}\right)^{2}\right) - \frac{\text{Tr}(\mathbf{X}^{\text{T}}\mathbf{X})^{2}}{CL}}, 1\right), \tag{5.7}$$

with $\mathbf{x}_{k,t}^{\text{T}} \in \mathbb{R}^{CL}$ the t^{th} row of the matrix $\mathbf{X}_k$.

Given the estimated decoder $\hat{\mathbf{d}}$ and τ_{test} time samples of a new EEG segment $\mathbf{X}^{(\text{test})} \in \mathbb{R}^{\tau_{\text{test}} \times CL}$ of a subject listening to one out of two competing speakers with speech envelopes $\mathbf{s}_1^{(\text{test})}$ and $\mathbf{s}_2^{(\text{test})}$, a decision about the auditory attention of the listener can be made by:

1. reconstructing the attended speech envelope from the EEG:

$$\hat{\mathbf{s}}^{(\text{test})} = \mathbf{X}^{(\text{test})}\hat{\mathbf{d}}$$

and

2. computing the Pearson correlation coefficients $\rho\left(\hat{\mathbf{s}}^{(\text{test})}, \mathbf{s}_1^{(\text{test})}\right)$ and $\rho\left(\hat{\mathbf{s}}^{(\text{test})}, \mathbf{s}_2^{(\text{test})}\right)$ between this reconstructed speech envelope and the original speech envelopes. The speaker corresponding to the highest correlation coefficient is identified as the attended speaker.

For SR, there is an important trade-off between the accuracy of the decision and the decision segment length τ_{test}, i.e., the number of time samples used to make a decision (Chapters 2 and 3). A longer decision segment leads to more accurate estimates of the Pearson correlation coefficients, thereby improving accuracy on the AAD decisions. However, this comes with the drawback of a poorer time resolution at which the AAD decisions are made, due to the longer decision segment length.

5.2.2 Unsupervised SR

Let us now assume that the attention labels $\{y_k\}_{k=1}^K$ are not known, even during training, meaning that we do not know to which of the speakers the subject is attending when designing our decoder (i.e., using (5.1) has become impossible). If we indeed can train the decoder without these labels, this would avoid the need for a dedicated training session during which the subject is instructed to attend to a specific speaker in order to collect ground-truth data. As a result, performing AAD becomes an *unsupervised* classification problem. The absence of labels is a roadblock in the computation of the crosscorrelation vector in (5.5), which requires the use of the correct speech envelope in its calculation (using the unattended envelope in (5.5) would result in a decoder that emphasizes the wrong speaker).

In Chapter 4, we proposed an unsupervised batch-training procedure on an unlabeled dataset $\{\mathbf{X}_k, (\mathbf{s}_{1_k}, \mathbf{s}_{2_k})\}_{k=1}^K$ of K segments (i.e., attention labels $\{y_k\}_{k=1}^K$ are unavailable/unknown). The main idea is to iteratively retrain a decoder by using labels that are predicted by the decoder from the previous iteration. We give a short description of this iterative procedure (see Chapter 4 and Algorithm 2 for a more extensive explanation). We start with an initial decoder $\hat{\mathbf{d}}^{(0)}$ at iteration 0, which can be, e.g., a pre-computed subject-independent decoder or even a decoder with random entries. First, the EEG autocorrelation matrix $\mathbf{R}_{xx}$ is estimated as in (5.6), which does not require

any attention labels. The iterative prediction of the labels and updating of the decoder then comprises the following steps:

1. given a decoder $\hat{\mathbf{d}}^{(i)}$ that is available after the i^{th} iteration, apply it on each EEG segment $\mathbf{X}_k$ to reconstruct the attended speech envelope:

$$\forall k : \hat{\mathbf{s}}_{\mathrm{a}_k}^{(i)} = \mathbf{X}_k \hat{\mathbf{d}}^{(i)}.$$

2. Per segment, correlate the reconstructed speech envelope $\hat{\mathbf{s}}_{\mathrm{a}_k}^{(i)}$ with both speech envelopes $(\mathbf{s}_{1_k}, \mathbf{s}_{2_k})$ to predict the attended speaker. As before, the first speaker is identified as the attended one if the sample Pearson correlation coefficient $\rho\left(\hat{\mathbf{s}}_{\mathrm{a}_k}^{(i)}, \mathbf{s}_{1_k}\right) > \rho\left(\hat{\mathbf{s}}_{\mathrm{a}_k}^{(i)}, \mathbf{s}_{2_k}\right)$ and vice versa. The predicted attended envelope is denoted as $\mathbf{s}_{\mathrm{pred}_k^{(i)}} \in \{\mathbf{s}_{1_k}, \mathbf{s}_{2_k}\}$.

3. Using the EEG segments and corresponding speech envelopes of the predicted attended speaker $\{\mathbf{s}_{\mathrm{pred}_k^{(i)}}\}_{k=1}^{K}$, the crosscorrelation vector can be computed/updated as in (5.5). It is crucial to use the *original* speech envelope ($\mathbf{s}_{1_k}$ or $\mathbf{s}_{2_k}$) and not the envelope $\hat{\mathbf{s}}_{\mathrm{a}_k}^{(i)}$ that was reconstructed from the EEG. Given the new crosscorrelation vector, the decoder $\hat{\mathbf{d}}^{(i+1)}$ can be updated as in (5.4). Then return to step (1) and iterate until convergence.

This iterative unsupervised predicting of labels could potentially inject incorrect labels and thus incorrect data in the estimation of the decoder, which could in principle lead to a downwards spiral of incorrect updating and thus to a badly-performing decoder. Remarkably, we showed in Chapter 4 that a *self-leveraging effect* occurs in this batch-mode iterative updating where the new decoder outperforms the previous decoder, despite the presence of labeling errors, resulting in an upwards instead of a downwards spiral. This happens even when the initial decoder is initialized with random values. This unsupervised subject-specific decoder outperformed a supervised subject-independent decoder (i.e., trained on data from other subjects than the one under test) and even closely approximated the performance of a supervised subject-specific decoder.

5.3 Time-adaptive unsupervised SR for AAD

The unsupervised training procedure of Section 5.2.2 assumes the availability of multiple data segments at once (i.e., batch-training). The batch computation is inherent to the procedure: once a new decoder is computed, *all* labels in the

recording are repredicted to improve the next decoder. After the unsupervised batch training, the final decoder is fixed and applied to unseen data from the subject under test. However, such a pre-trained time-invariant decoder does not adapt to non-stationarities due to changing conditions and situations and may thus perform suboptimally. Here, we propose a time-adaptive realization of such an unsupervised AAD decoder, i.e., a decoder that adapts itself over time when EEG and audio data (processed in envelopes) are continuously streaming in.

Assume some initial decoder. Data segments of T_{ud} samples of EEG and audio data start streaming in. At a certain point in time, assume the k^{th} segment of EEG data $\mathbf{X}_k \in \mathbb{R}^{T_{\text{ud}} \times CL}$ (see (5.3)) and corresponding segments of the speech envelopes of the two competing speakers $(\mathbf{s}_{1_k}, \mathbf{s}_{2_k}) \in \mathbb{R}^{T_{\text{ud}}}$ become available. There is no information available about which speaker is the attended or unattended one in this segment. The goal is now to update the decoder in an unsupervised manner based on the newly available information, to which end we will propose and compare two approaches (Sections 5.3.1 and 5.3.2).

In this time-adaptive procedure, it is important to distinguish the updating segment length T_{ud} from the decision segment length τ_{test}. The former, equivalent to the segment length T in the previous sections, corresponds to the length of the segments on which the prediction of the labels for the updating/training and the updating/training itself is performed. The latter corresponds to the length of the segments on which AAD decisions are made to in the end steer the enhancement algorithm in the hearing device. This decision segment length is, therefore, much more sensitive to speed (for example, because of switches in auditory attention (Chapter 2)) than the updating segment length, as there can be some delay allowed in updating the decoder. Therefore, the updating segment length is typically larger than the decision segment length $T_{\text{ud}} \geq \tau_{\text{test}}$, i.e., within each updating segment, multiple AAD decisions are made.

In the time-adaptive sliding window approach (Section 5.3.1), the aforementioned batch-mode procedure is mimicked (i.e., including repredictions of the labels of previous segments) but over a finite time horizon which is implemented as a sliding window. In Section 5.3.2, we propose an alternative time-adaptive approach that does *not* recompute previously predicted labels and, therefore, can be implemented in a recursive manner. This is much more attractive from a computational and memory usage point of view. Both approaches to update the decoder are explained in more detail in the following sections.

5.3.1 Sliding window implementation

In the sliding window implementation (Figure 5.1), a pool of K data segments of updating segment length T_{ud} is kept in memory and is updated using the first in, first out (FIFO) principle. When a new data segment $\{\mathbf{X}_k, (\mathbf{s}_{1_k}, \mathbf{s}_{2_k})\}$ becomes available, the oldest data segment is discarded and the pool is updated with the newest one, resulting in the new pool $\left\{\mathbf{X}_{k-m}, \left(\mathbf{s}_{1_{k-m}}, \mathbf{s}_{2_{k-m}}\right)\right\}_{m=0}^{K-1}$. The stimulus decoder is then updated similarly to the batch-mode implementation explained in Section 4.3, but on the finite pool of K segments.

First, the EEG autocorrelation matrix of the new segment is computed similarly to (5.6):

$$\mathbf{R}_{xx_k} = (1-\lambda_k)\mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k + \lambda_k \frac{\mathrm{Tr}\left(\mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k\right)}{CL}\mathbf{I}, \tag{5.8}$$

with the regularization parameter λ_k recomputed per new segment k using (5.7). The aggregated autocorrelation matrix across the whole pool of K segments can then be updated/recomputed:

$$\mathbf{R}_{xx} = \sum_{m=0}^{K-1} \mathbf{R}_{xx_{k-m}}. \tag{5.9}$$

The autocorrelation matrices of the previous segments can be recomputed or stored and retrieved from previous computations. We found empirically that better results are obtained when regularizing the new autocorrelation matrix (5.8) *before* being stored and combined in (5.9), instead of regularizing the combined autocorrelation matrix (5.9), i.e., *after* combining the autocorrelation matrices from the different segments.

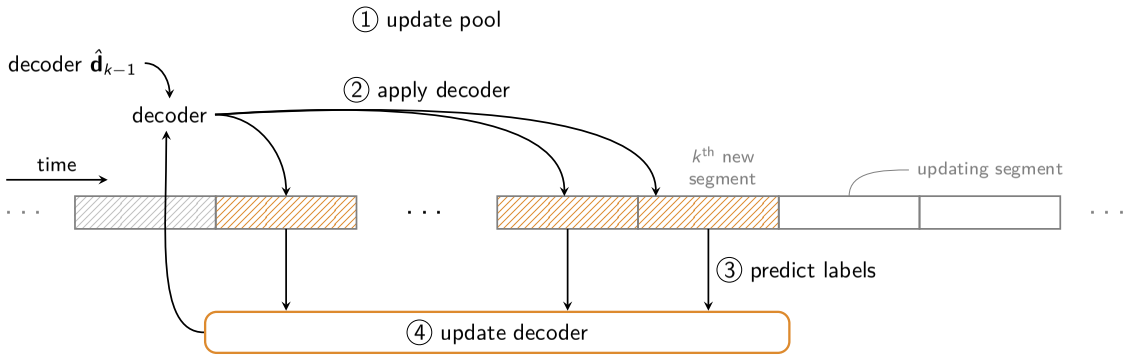
Using the decoder $\hat{\mathbf{d}}_{k-1}$ from the previous step, the iterative procedure of predicting labels, updating the crosscorrelation vector(s), and decoder on the pool of K segments $\left\{\mathbf{X}_{k-m}, \left(\mathbf{s}_{1_{k-m}}, \mathbf{s}_{2_{k-m}}\right)\right\}_{m=0}^{K-1}$ can be initiated. Given the per-segment predicted attended speaker $\{\mathbf{s}_{\text{pred}_{k-m}}\}_{m=0}^{K-1}$ (initially obtained using $\hat{\mathbf{d}}_{k-1}$), the crosscorrelation vectors can be updated as in (5.5):

$$\mathbf{r}_{xs_{k-m}} = \mathbf{X}_{k-m}^{\mathrm{T}}\mathbf{s}_{\text{pred}_{k-m}}, \forall m \in \{0, \ldots, K-1\}.$$

The aggregated crosscorrelation vector and corresponding decoder can then be computed as:

$$\mathbf{r}_{xs} = \sum_{m=0}^{K-1} \mathbf{r}_{xs_{k-m}} \Rightarrow \hat{\mathbf{d}}_k = \mathbf{R}_{xx}^{-1}\mathbf{r}_{xs}.$$

This predict-and-update procedure is then iterated I times on the same pool of K segments. Based on Chapter 4 (see, for example, Figure 4.7), we choose $I = 5$,



Sliding window algorithm

① update pool:

1. FIFO update pool of EEG/speech envelopes $\left\{\mathbf{X}_{k-m}, \left(\mathbf{s}_{1_{k-m}}, \mathbf{s}_{2_{k-m}}\right)\right\}_{m=0}^{K-1}$
2. Compute $\mathbf{R}_{xx_k} = (1-\lambda_k)\mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k + \lambda_k \frac{\mathrm{Tr}(\mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k)}{CL}\mathbf{I}$ and $\mathbf{R}_{xx} = \sum_{m=0}^{K-1} \mathbf{R}_{xx_{k-m}}$

while $i \leq I$ and $\hat{\mathbf{d}}_k$ changes

② apply decoder (on whole pool) ($\hat{\mathbf{d}}_{k-1}$ in first iteration)

③ predict labels (on whole pool)

④ update decoder, given updating segments $\left\{\mathbf{X}_{k-m}, \mathbf{s}_{\mathrm{pred}_{k-m}}\right\}_{m=0}^{K-1}$:

1. Compute $\mathbf{r}_{xs_{k-m}} = \mathbf{X}_{k-m}^{\mathrm{T}}\mathbf{s}_{\mathrm{pred}_{k-m}}, \forall m \in \{0, \ldots, K-1\}$ and $\mathbf{r}_{xs} = \sum_{m=0}^{K-1} \mathbf{r}_{xs_{k-m}}$
2. Recompute $\hat{\mathbf{d}}_k = \mathbf{R}_{xx}^{-1}\mathbf{r}_{xs}$

end while

Figure 5.1: The time-adaptive unsupervised sliding window scheme and algorithm to update a stimulus decoder, with repredictions of the labels on previous segments.

after which the iterative batch updating procedure is generally observed to have converged to a final decoder (see also Figure 5.6). Given that we iterate over this pool of K segments, this approach can only be implemented in a sliding window manner and not recursively. Lastly, the pool size parameter K represents an important trade-off between accuracy, adaptivity, and computational complexity and memory usage. A longer sliding window (i.e., larger K) means that more data is available to compute the decoder, resulting in a better approximation of the batch-mode decoder but also resulting in a lower adaptivity and higher memory requirements (see also Section 5.4).

5.3.2 Recursive implementation

As an alternative to the sliding window implementation, we propose a single-shot predict-and-update scheme (Figure 5.2). As opposed to the sliding window approach, the labels of previous updating segments are not repredicted, enabling a recursive implementation that is much more efficient from a computational and memory usage point of view. In this recursive implementation, the decoder resulting from the previous update $\hat{\mathbf{d}}_{k-1}$ is applied to the new segment $\{\mathbf{X}_k, (\mathbf{s}_{1_k}, \mathbf{s}_{2_k})\}$ of length T_{ud} to predict the label of this new segment, resulting in the predicted attended envelope $\mathbf{s}_{\text{pred}_k}$. To update the decoder, a regularized autocorrelation matrix is computed based on the new k^{th} segment as in (5.8), while the predicted attended envelope is used to compute a crosscorrelation vector as in (5.5):

$$\mathbf{r}_{xs_k} = \mathbf{X}_k^{\text{T}} \mathbf{s}_{\text{pred}_k}.$$

This new autocorrelation matrix $\mathbf{R}_{xx_k}$ and crosscorrelation vector $\mathbf{r}_{xs_k}$ can then be combined with the autocorrelation matrix $\mathbf{R}_{xx}$ and crosscorrelation vector $\mathbf{r}_{xs}$ integrating all previous information to update the decoder:

$$\begin{cases} \mathbf{R}_{xx} \leftarrow \alpha \mathbf{R}_{xx} + (1-\alpha)\mathbf{R}_{xx_k} \\ \mathbf{r}_{xs} \leftarrow \beta \mathbf{r}_{xs} + (1-\beta)\mathbf{r}_{xs_k} \end{cases} \Rightarrow \hat{\mathbf{d}}_k = \mathbf{R}_{xx}^{-1} \mathbf{r}_{xs}.$$

The influence of the weighting parameters α and β will be empirically evaluated in Section 5.4.

Unlike the sliding window implementation, which uniformly weighs the K past segments in the new decoder, this recursive algorithm implements an exponential weighting across all past segments. This exponential weighting could be advantageous, especially in an adaptive context, as the more relevant closest (past) segments have higher weights than those further in the past. One can choose the weighting parameters α and β such that the center of mass of the exponential weighting is the same as of the sliding window approach [168]:

$$\alpha = \frac{K-1}{K+1} \text{ and } K = \frac{1+\alpha}{1-\alpha}. \tag{5.10}$$

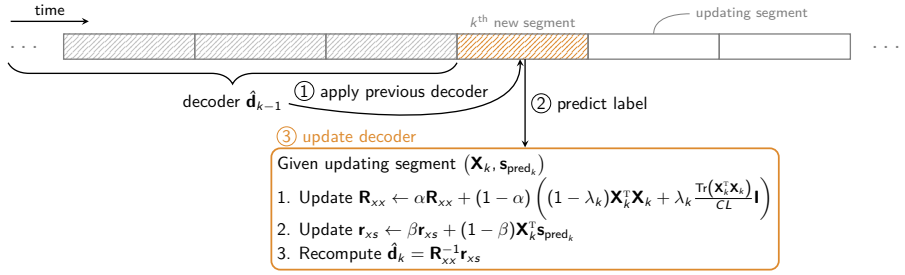


Figure 5.2: The time-adaptive unsupervised recursive predict-and-update scheme to update a stimulus decoder.

A possible drawback of this recursive implementation is that previous labels are not repredicted and that one can not apply multiple iterations over a pool of segments as in the sliding window version. This could lead to slower convergence or poorer accuracies. However, the upside is that the procedure is much easier to implement and much more efficient in terms of memory and computation resources (see Section 5.3.3). In Section 5.4, we will demonstrate that the impact on convergence speed and accuracy is negligible.

5.3.3 Memory usage

Sliding window implementation

For the sliding window implementation, at least the pool of K EEG segments ($K \times T_{\text{ud}} \times C$) and $2K$ speech envelopes ($2K \times T_{\text{ud}}$) need to be stored in memory. One does not need to store the L different time lags, as these can always be generated from the original EEG data. Furthermore, it is not possible to simply store the autocorrelation matrices and crosscorrelation vectors of the previous EEG segments - which would require less memory usage - as for the repredictions of the labels, the decoder has to be applied to the original EEG and speech envelope data. This leads to the following memory usage:

$$(C+2)KT_{\text{ud}} \sim \mathcal{O}(KT_{\text{ud}}C)\,,$$

where generally $T_{\text{ud}} >> K$ or C. As the storage of the sample dimension T_{ud} is required, this is generally a very high memory usage.

Recursive implementation

The recursive implementation minimally requires the storage of one autocorrelation matrix $\mathbf{R}_{xx}$ (built from $\frac{CL(CL+1)}{2}$ elements due to symmetry) and one crosscorrelation vector $\mathbf{r}_{xs}$ (built from CL elements). This leads to the following memory usage:

$$CL + \frac{CL(CL+1)}{2} \sim \mathcal{O}\big(C^2L^2\big)\,,$$

which is, as expected, much less than the sliding window approach.

To better appreciate the differences, consider the following practical realistic example with $C = 24$ EEG channels, $L = 6$ time lags, pool size $K = 19$, and an updating segment length T_{ud} equivalent to 1200 samples (corresponding to 60 s when the EEG and speech envelopes are downsampled to 20 Hz). The sliding window approach then requires the storage of 592 800 elements, which is more than 50 times more than for the recursive implementation, which requires to only story 10 584 elements.

5.4 Validation and comparison

We test both versions of the time-adaptive unsupervised SR algorithm of Section 5.3 for different hyperparameter settings (i.e., the updating segment length T_{ud}, pool size K in the sliding window implementation, and exponential forgetting factors α, β in the recursive implementation). In all experiments, we start from a different *fully random initial* decoder, generating the first prediction(s). In the recursive implementation, the initial autocorrelation matrix and crosscorrelation vector are initialized with all zeros. We compare the sliding window and recursive implementation and select a set of hyperparameters on one dataset (Dataset A), and validate the chosen algorithm on a second one (Dataset B). These datasets are concisely described in Section 5.4.1, while the performance metrics are described in Section 5.4.2. The experiments and results are discussed in Section 5.4.3, with a more detailed discussion on the effect of repredictions of the labels in the sliding window approach in Section 5.4.4. The final settings are validated on Dataset B in Section 5.4.5.

5.4.1 Data and preprocessing

AAD datasets

The first AAD dataset (Dataset A) is from [1] and contains the EEG (64-channel BioSemi ActiveTwo system) and audio data of 16 normal-hearing subjects participating in an AAD experiment, where the subjects were instructed to listen to one of two competing speakers located at $\pm 90°$ azimuth direction (in dichotic and HRTF-filtered listening conditions). Per subject, eight stories of 6 min and 12 repetitions of 2 min of those stories are presented, resulting in 72 min data per subject. This dataset is available online [132].

The second AAD dataset (Dataset B) is from [2] and will act as an independent validation dataset. It contains the EEG (64-channel BioSemi ActiveTwo system) and audio data of 18 normal-hearing subjects in a similar AAD experiment with two competing speakers, located at $\pm 60°$ azimuth direction (HRTF-filtered) and using different acoustic room properties. Per subject, 50 min of data (60×50 s trials) are available. This dataset is also available online [133].

Preprocessing

To preprocess the EEG and audio data, we applied the same preprocessing steps as in [1] and Chapter 4. The speech signals are first filtered using a gammatone filterbank. Using a power-law operation with exponent 0.6, an envelope is computed for each subband signal. All subband envelopes are afterwards summed to one envelope. Both EEG and speech envelopes are filtered between 1–9 Hz and downsampled to 20 Hz (Dataset A)/32 Hz (Dataset B). In neither of the datasets, additional re-referencing or artifact rejection has been applied.

5.4.2 Performance metrics

To evaluate a specific implementation with a specific set of hyperparameters, we use three performance metrics, quantifying the accuracy, adaptivity, and memory usage of each algorithm:

- **Final accuracy:** the final accuracy is defined as the average of the accuracies on the independent test set across the last 5 min of updating, i.e., after the adaptive decoder has had sufficient time to converge to a steady-state regime.

- **Settling time:** in a time-adaptive context, not only accuracy but also adaptivity or speed of adaptation is an important metric. Here, we quantify the adaptivity with the settling time, defined similarly as in control theory [169]. This settling time is defined as the point in time where the accuracy has reached a threshold for the first time *and* remains above the lower bound of a predefined error band for the remainder of the updating procedure. The threshold is defined as a convex combination of the final accuracy and the initial chance level performance (50%; before the updating procedure):

  $$\text{threshold} = 0.95 \times \text{final accuracy} + 0.05 \times 0.5.$$

  The error band, which allows taking the variability into account, is defined as:

  $$\text{error band} = \text{final accuracy} \pm 2M,$$

  with M the difference between the maximum and minimum across the last 5 min accuracies.

- **Memory usage:** the memory usage, i.e., the number of elements that need to be stored in memory, is computed as in Section 5.3.3.

Figure 5.3 illustrates the final accuracy and settling time performance metrics for a representative subject and specific implementation (this figure is only meant to illustrate the updating procedure and the definitions of the performance metrics, and should not be viewed as a validation result).

5.4.3 Hyperparameter selection

We test the different implementations of Section 5.3 for different hyperparameter settings on Dataset A. Per subject, we randomly permute the 6 min trials of the first 48 min and use those as the updating set, i.e., the data on which the time-adaptive unsupervised updating from a random initial decoder is performed. To track the accuracy of the updated decoder over time, after each update, we evaluate the decoder on the separate set of the last 24 min of repetition data, using $\tau_{\text{test}} = 30\,\text{s}$ decision segments to make a decision about the auditory attention (i.e., to compute the Pearson correlation coefficient with both speech envelopes). Per subject, we perform ten random permutations. For the decoder, we choose time lags up to 250 ms [1,74], which corresponds to $L = 6$ for Dataset A and $L = 9$ for Dataset B (as both are sampled at different rates). We choose updating segment length $T_{\text{ud}} = 60\,\text{s}$ (different from decision segment length $\tau_{\text{test}} = 30\,\text{s}$), i.e., we update the decoder every 60 s. As the performance of the stimulus decoder heavily depends on the amount of data available to make a

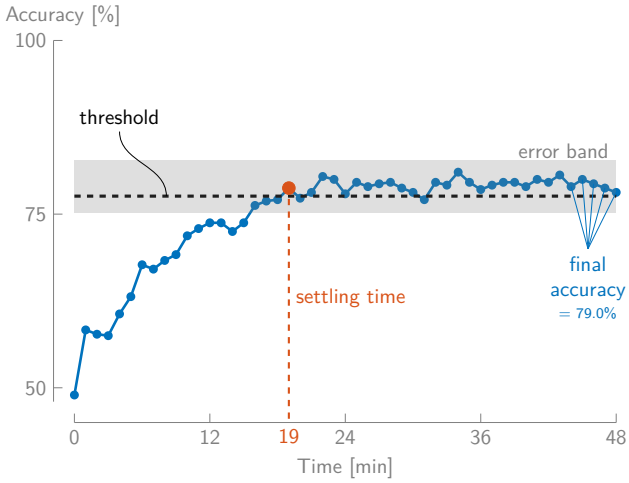


Figure 5.3: Illustration of the adaptation curve and performance metrics for a representative subject (Subject 4) of Dataset A, starting from a random initial decoder and updating every 60 s, for the recursive implementation with $\alpha = \beta = 0.9$ (average across ten runs).

decision (Chapters 2 and 3), similarly to Chapter 4, we choose T_{ud} as large as possible - without waiting too long - to produce as reliable labels as possible. As explained in the introduction of Section 5.3, we can afford such a longer delay in updating (as opposed to the decision segment length (Part I), explaining why we work on different time resolutions for the updating and testing). However, we do not take T_{ud} larger than 60 s, as the performance of the stimulus decoder starts to saturate above this segment length, and because this would require a too long sustained attention on the same speaker.

Figure 5.4 shows the average final accuracy, settling time, and memory usage for different settings of the sliding window and recursive implementations across the 16 subjects of Dataset A and ten random permutations. To compute the final accuracy per setting, 16 (number of subjects) $\times 10$ (random permutations) $\times 5$ (number of time points of updating: see Section 5.4.2) $\times 48$ (number of 30 s decision segments in the 24 min test set) evaluations are thus performed. The sliding window implementation is evaluated for different updating segment lengths ($T_{\text{ud}} \in \{60, 30, 10\,\text{s}\}$) and pool sizes ($K \in \{10, 20, \ldots, 60\}$, except for $T_{\text{ud}} = 60\,\text{s}$, where the maximum is $K = 40$). Only the results for $T_{\text{ud}} = 60\,\text{s}$ updating segments are shown, which are found to be superior to $T_{\text{ud}} = 30\,\text{s}$ and 10 s. Therefore, and for the clarity of Figure 5.4, the recursive implementation is only evaluated for $T_{\text{ud}} = 60\,\text{s}$ and α, β ranging independently from each other

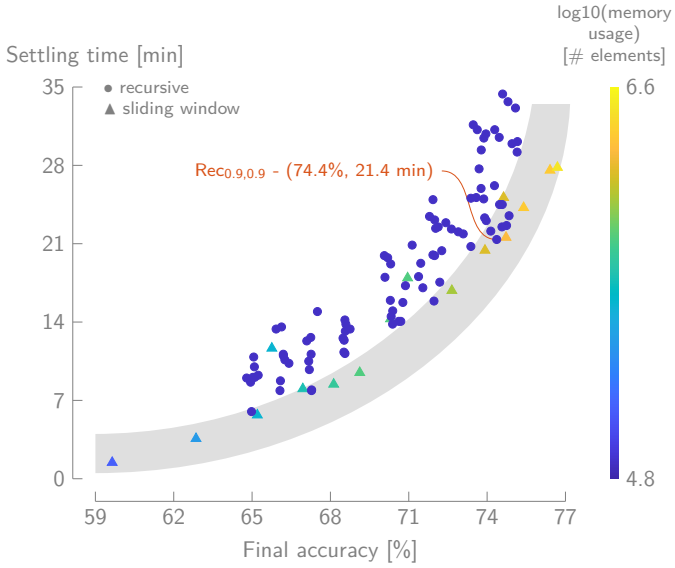


Figure 5.4: Settling time versus final accuracy for different parameter settings across the 16 subjects of Dataset A and ten random permutations per subject. The shaded area highlights the points that are close to or in the Pareto front. The indicated recursive algorithm with $\alpha = \beta = 0.9$ gives one of the best trade-offs between final accuracy, adaptivity, and memory usage across the evaluated settings.

from 0.6 to 0.95 in steps of 0.05 and $\alpha = \beta$ ranging in more fine-grained steps of 0.005 from 0.8 to 0.95. As can be seen in Figure 5.4, the memory usage of the recursive implementation is the same for every setting, while this is dependent on the pool size K and updating segment length T_{ud} for the sliding window implementation (Section 5.3.3). In general, there is a clear positive correlation between a higher final accuracy and a higher settling time, representing the trade-off between accuracy and adaptability of the decoder. The points in the shaded lower envelope area of the point cloud in Figure 5.4 represent the Pareto front, i.e., the settings that give the best trade-off between a high final accuracy and low settling time.

Surprisingly, the Pareto front of the recursive implementations (i.e., without repredictions of previous labels) seems to achieve very similar performances in terms of final accuracy and settling time as the sliding window implementations (i.e., with repredictions of previous labels). In Section 5.4.4, we will investigate more closely why these repredictions of previous labels seem to have such little effect. Moreover, the recursive implementation requires on average 16× less memory than the sliding window implementation (see Figure 5.4) and is

computationally more efficient, making it the preferred implementation.

As indicated in Figure 5.4, one of the best choices across the evaluated settings is the recursive implementation with $\alpha = \beta = 0.9$, resulting on average in 74.4% final accuracy (standard deviation 12.1%) after 21.4 min (st. dev. 11.7 min) (see Figure 5.5 for the per-subject performances). The latter implies that it takes about 20 min before the decoder has learned how to optimally decode the attended speaker starting from a random decoder. These settings are therefore used in the remainder of the experiments. Although there are a few settings of the sliding window implementation (i.e., $T_{\text{ud}} = 60\,\text{s}$ and $K = 40$) that give a slightly better final accuracy for similar settling times, they require more memory storage (42× more elements) and are also computationally heavier (due to the repredictions of the labels). Lastly, plugging in the hyperparameter values $\alpha = \beta = 0.9$ in (5.10), which allows converting the forgetting factors α and β of the recursive implementation to the equivalent pool size parameter K of the sliding window implementation, results in $K = 19\,\text{min}$. This is indeed consistent with the results of the sliding window implementation with $K = 19$, which has a very similar performance (73.6% (st. dev. 13.2%) in 19.4 min (st. dev. 11.5 min)) while requiring much more memory. There is no noticeable benefit from the exponential weighting over the uniform weighting, given that the performance is tested on an asynchronous, independent test set. In Section 5.5.2, we will concurrently test and update on the same data, potentially revealing the benefit of exponential weighting.

5.4.4 Effect of repredictions

The results in Section 5.4.3 show that the single-shot recursive implementation, without repredictions of the labels, performs on par with the sliding window implementation with repredictions of the labels of previous segments. This suggests that, in the considered time-adaptive context, the iterative repredictions of labels on the current pool of K segments have no additional benefit and that the labels hardly change between before and after the relabeling procedure. This is confirmed by computing the total number of labels that changed before and after the iterative relabeling procedure in all updates before the settling time, i.e., before reaching steady-state performance. For $T_{\text{ud}} = 60\,\text{s}$, this total number of labels that changed is (on average across subjects and random permutations in Section 5.4) only 0.16 for $K = 10$, 0.70 for $K = 20$, and 1.13 for $K = 40$. This shows that the number of self-corrected labels in the iterative relabeling procedure is minimal, even more so if the pool size K is small.

To more closely investigate this dependence of the relabeling on the pool size K, we compute the training accuracy (i.e., the percentage of correct labels in

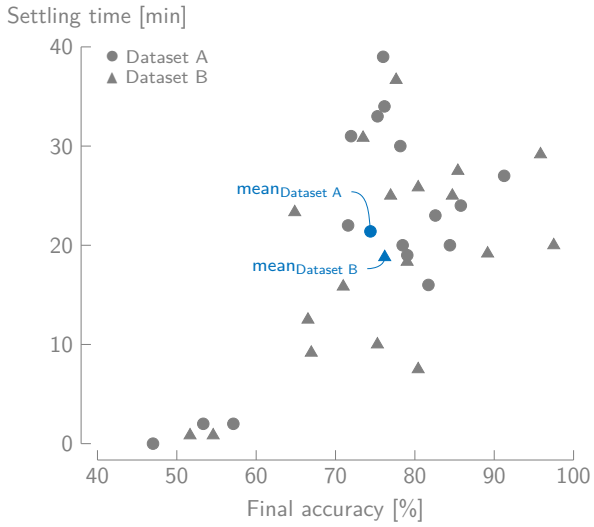


Figure 5.5: Individual settling time versus final accuracy per subject of Dataset A and Dataset B (average across runs) for the chosen recursive implementation with $\alpha = \beta = 0.9, T_{\text{ud}} = 60\,\text{s}$ (Dataset A) and $\alpha = \beta = 0.916, T_{\text{ud}} = 50\,\text{s}$ (Dataset B).

the updating set) in the different iterations as a function of the size of the updating set for the batch-mode unsupervised algorithm in Chapter 4 using $T_{\text{ud}} = 60\,\text{s}$ updating segments and $\tau_{\text{test}} = 30\,\text{s}$ decision segments. Per subject and size of the updating set, ten runs with different randomly selected updating sets of size K from the total dataset are performed. Figure 5.6 then shows the average training accuracy across subjects and runs on Dataset A. From Figure 5.6, it is clear that the self-correcting behavior on the predicted labels of the first iteration only starts to occur when the updating set contains more than 14–20 min of data. This can be explained from a mathematical point of view as an overfitting effect: when K is small, the decoder has enough degrees of freedom to span all initial predictions in the subsequent iterations, leading to an overfitted decoder.

When $K \geq 14$, there is a clear effect of the second and subsequent iterations (until convergence to the fixed point). This effect, however, seems not to be present in the time-adaptive context. This is explained by considering the initial decoder for each new decoder update when a new segment becomes available. In the batch-mode design, this initial decoder is always a random decoder, whereas in the time-adaptive context, this will only be the case for the first received data segment. In the later updates, the initial decoder is already improved based on past data. Consider the case of $T_{\text{ud}} = 60\,\text{s}, K = 20$

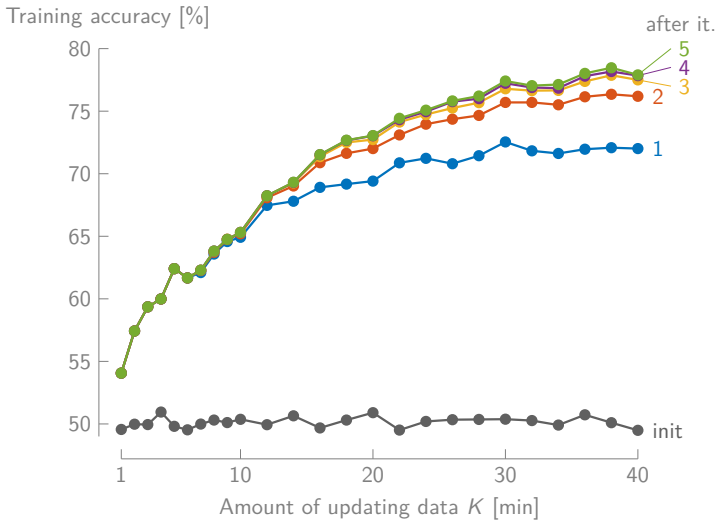


Figure 5.6: The training accuracy on 30 s decision segments for the batch-mode unsupervised iterative updating procedure as a function of the amount of updating data, for different numbers of relabeling iterations ('init' refers to the accuracy when no iterations are performed; average across all subjects of Dataset A and random permutations). Given that the updating segment length $T_{\text{ud}} = 60\,\text{s}$, the amount of updating data corresponds to pool size K (in minutes).

in Figure 5.6. In the time-adaptive case, already 19 updates will have been completed before a full pool of $K = 20$ segments becomes available. After 19 updates, however, the decoder has already substantially improved (see also Figure 5.3). Therefore, as suggested in Figure 5.6, the decoder will not exhibit random performance but performance close to the one of the converged decoder in Figure 5.6. Consequently, the effect of a reprediction on the labels on previous segments in the pool will be similar to the effect of the last iterations (3, 4, 5) in Figure 5.6, that is, very small. In other words, the initial decoder is then already close to the fixed point of the updating procedure (Chapter 4).

Given the important trade-off in memory usage and computational complexity, these insignificant improvements do not outweigh the additional required resources.

5.4.5 Validation on an independent dataset

To confirm that the recursive implementation with $T_{\text{ud}} = 60\,\text{s}$ and $\alpha = \beta = 0.9$ is a robust choice across subjects and datasets, and that no overfitting of the hyperparameters has occurred on Dataset A, we apply the recursive algorithm on the completely independent Dataset B, again starting from a fully random decoder. Given that Dataset B only contains 50 s trials, we choose $T_{\text{ud}} = 50\,\text{s}$. Furthermore, given that $\alpha = \beta = 0.9$ is equivalent with $K = 19$ according to (5.10), resulting in a 19 min history when using 60 s segments, this becomes $K = 22.8$ for 50 s segments. Using (5.10), the equivalent choice for Dataset B becomes $\alpha = \beta = 0.916$.

We test this recursive implementation with $T_{\text{ud}} = 50\,\text{s}$ and $\alpha = \beta = 0.916$ on each of the 18 subjects of Dataset B by ten times randomly selecting 40 min as the updating set and the remaining 10 min as test set. The average final accuracy and settling time (on 30 s decision segments) are 76.2% (st. dev. 12.4%) and 18.8 min (st. dev. 10.3 min) (see Figure 5.5 for the per-subject performances). As this is very similar to the performance obtained in Section 5.4.3 (and even slightly better), it confirms that the chosen specific recursive implementation is a robust choice.

5.5 Evaluation in time-adaptive context

In Section 5.4, we have tested the proposed time-adaptive unsupervised SR algorithm asynchronously, i.e., the test set is time-independent from the updating set. While these experiments allowed to investigate the behavior of the proposed method, they do not necessarily reflect a practical use case of the algorithm as in a neuro-steered hearing device application, i.e., while the time-adaptive unsupervised decoder needs to simultaneously update/adapt and provide AAD decisions. In Section 5.5.1, we simulate on Dataset A a situation where electrodes are disconnected, for example, due to movements. In Section 5.5.2, we then evaluate the time-adaptive unsupervised decoder on a third dataset (Dataset C), i.e., while needing to adapt across multiple recording days.

5.5.1 Suddenly disconnecting EEG electrodes

Experiment

We compare the fixed supervised decoder with the proposed time-adaptive unsupervised decoder using the selected recursive implementation with $\alpha =$

$\beta = 0.9$ and $T_{\text{ud}} = 60\,\text{s}$ from Section 5.4, when simulating a situation where electrodes are disconnected. Per subject of Dataset A, we randomly permute the 6 min-trials of the first 48 min, to which we add the last 24 min of repetition data, resulting in 72 min of data. The fixed supervised decoder is trained on the first 30 min (i.e., using the available attention labels). Furthermore, also during these first 30 min, the time-adaptive unsupervised decoder has time to update itself starting from a fully random initial decoder, with the autocorrelation matrix and crosscorrelation vector initialized with all zeros.

After these first 30 min of data, we simulate the case where a number of electrodes are disconnected, as could occur in practice, by setting some EEG channels to zero. On these last 42 min of data with disconnected electrodes, the original fixed supervised decoder (trained with all electrodes) is then applied on each $\tau_{\text{test}} = 30\,\text{s}$ decision segment, while the time-adaptive unsupervised decoder keeps on continuously updating per $T_{\text{ud}} = 60\,\text{s}$, and decoding the auditory attention per $\tau_{\text{test}} = 30\,\text{s}$ decision segment. We then compare both decoders after the electrodes are disconnected, i.e., by computing the accuracy across all binary decisions on the last 42 min (thus also taking the settling period of the adaptive decoder after the change into account).

This experiment is performed in two scenarios: when starting from the full high-density 64-channel EEG setup and from a reduced 22-channel subset, where the electrodes are selected corresponding to the mobile 24-channel SMARTING EEG system from mBrainTrain. The latter is added to compare the results with those in Section 5.5.2, where a third dataset (Dataset C) is introduced that is recorded using this 24-channel EEG system. The number of disconnected electrodes is varied from 0 to 32 (for the 64-channel case) and from 0 to 11 (for the 22-channel case). Per number of disconnected electrodes, ten random permutations (i.e., of randomly permuting the first eight 6 min-trials and set of disconnected electrodes) are performed.

Results

Figure 5.7a shows the average accuracy across all 16 subjects of Dataset A and the ten random permutations (per subject and number of disconnected electrodes) as a function of the number of disconnected electrodes, starting from the full high-density 64-channel setup and reduced 22-channel setup. When no electrodes are disconnected, the fixed supervised decoder outperforms the time-adaptive unsupervised one with around 4.4% in accuracy in both cases. This difference in accuracy is expected and in line with the batch results obtained in Chapter 4. However, in the 64-channel setup, already when disconnecting three electrodes, the adaptive unsupervised decoder performs better than the

fixed supervised one. This effect is already present after two electrodes for the more mobile setup of 22 channels. The difference between both decoders then increases up to 13.9% for the 64-channel setup and 8.3% for the 22-channel setup.

These performance differences seem to be mainly due to the decrease in accuracy of the fixed supervised decoder, which has been trained without taking the disconnected electrodes into account, while the time-adaptive unsupervised decoder remains relatively stable (especially in the 64-channel case). This shows that the latter decoder can effectively adapt to disconnected electrodes, quickly finding an almost equivalent alternative way to decode the attended speech envelope from the reduced set of electrodes. The fact that the adaptive decoder obtains similar performances with only 32 channels compared to 64 channels comes not as a surprise, given that in [121], it was shown that the number of EEG channels could be reduced to around ten without a loss in performance, however, given an optimal channel selection procedure (while here we simulate random electrode disconnections, as would occur in practice) (see also Section 1.7.3).

This experiment clearly shows the added value of the time-adaptive unsupervised approach, effectively and *automatically* adapting to changes in the EEG setup, here simulated by electrodes that are disconnected. Furthermore, we have only simulated one change in the (EEG) setup in an otherwise very controlled experiment, and already obtained a better performance with the time-adaptive approach when two or three electrodes are disconnected. In practice, such changes would occur in combination with other non-stationarities in the data, which is investigated in the next section.

5.5.2 Adaptation across multiple recording days

While in Section 5.5.1, the proposed time-adaptive unsupervised decoder is tested in a more time-adaptive context where electrodes are disconnected, the non-stationarities in the data are still limited to this single change. Furthermore, the EEG data per subject are recorded in one session, with only small breaks in between, and in a very controlled setup. Therefore, in this section, we evaluate the proposed method on a third dataset (Dataset C) where the decoder needs to adapt across multiple days of recordings, potentially combined with electrodes that are disconnected.

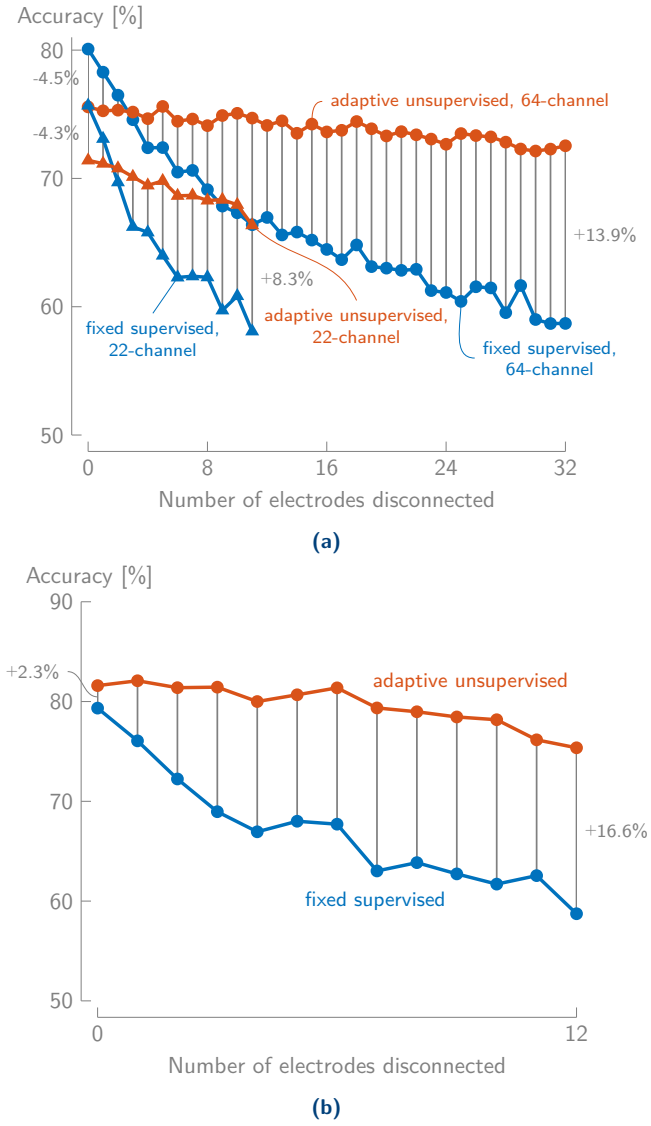


Figure 5.7: The accuracy on 30 s decision windows of the fixed supervised and time-adaptive unsupervised decoder as a function of the number of disconnected electrodes for **(a)** Dataset A (electrodes are disconnected after the first 30 min of (training) data) and **(b)** Dataset C (electrodes are disconnected after the first two (training) sessions). The accuracies are averaged over all 16/2 (Dataset A/Dataset C) subjects and ten runs (per subject and number of disconnected electrodes).

Data and preprocessing

AAD dataset We use a third dataset (Dataset C) containing EEG and audio data of two subjects from a longitudinal AAD experiment across multiple days, carried out at the participants' homes. A two-talker AAD experiment was conducted in eight different sessions that took place on seven different days. In each session, four blocks of 6 min stories are presented to each subject, resulting in a total of 192 min of AAD data per subject. The audio stimuli differed across all sessions. The first two sessions took place on the same day, while the other six sessions took place on different days (see also Figure 5.8). The EEG was measured using a 24-channel SMARTING mobile EEG system from mBrainTrain.

While this dataset was initially recorded for the purpose of neurofeedback experiments, it can be used to test the proposed time-adaptive unsupervised decoder as it reflects the practical use case of a neuro-steered hearing device. The algorithm will need to adapt over various days, meaning that there will be changes in, for example, EEG setup, electrode impedances, conditions, speaker and story characteristics, and state of mind of the user, as would all occur in practice. While two subjects are not enough to draw firm (statistical) conclusions, it allows showcasing how the proposed algorithm can be used in a practical, online context.

Preprocessing The EEG data and audio envelopes are preprocessed in the same way as in Section 5.4.1. The only difference is that $L = 400$ ms is chosen as the post-stimulus range of time lags for the decoder, reflecting the choice in [3].

Experiment

We compare the fixed supervised decoder with the proposed time-adaptive unsupervised decoder. Again, we only consider the recursive version of the time-adaptive decoder, as it performs similarly to the sliding window version while requiring much less resources. As shown in Figure 5.8, the supervised, fixed decoder is trained on the 48 min of data from the first two sessions on the first day, using the information about which speaker is the attended one. This fixed decoder is then applied per $\tau_{\text{test}} = 30$ s decision segment on all other sessions on the other days. As such, this decoder reflects the practical use case where first data of a new neuro-steered hearing device user need to be recorded in an a priori calibration session, whereafter the trained decoder is loaded onto the device.

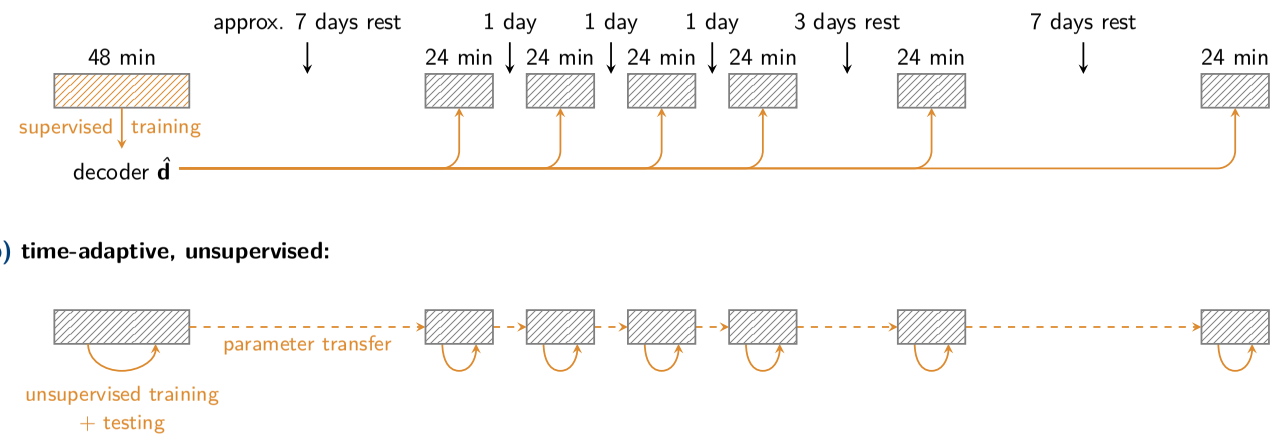


Figure 5.8: An overview of the setup of Dataset C. **(a)** The fixed decoder is trained in a supervised manner on the first two sessions on the first day and tested on all other sessions on the other days. **(b)** The proposed time-adaptive unsupervised decoder is initialized with a random decoder and is continuously updated over time and tested/applied on/to each next decision segment. The autocorrelation matrix and crosscorrelation vector are stored in between sessions ('parameter transfer').

The time-adaptive unsupervised decoder is implemented using the settings as determined in Section 5.4 and is initialized with a fully random decoder, while the autocorrelation matrix and crosscorrelation vector are initialized with all zeros. Per $T_{\text{ud}} = 60\,\text{s}$, the decoder is continuously updated using the recursive implementation with $\alpha = \beta = 0.9$. To fully leverage the time-adaptivity of this decoder, after each update, it is, similarly to Section 5.5.1, applied to the next two $\tau_{\text{test}} = 30\,\text{s}$ decision segments to make AAD decisions[1] (Figure 5.8). The first 48 min of the first two sessions on the first day are used to let the decoder initialize and converge as in Figure 5.3, starting from a random initial decoder. In between sessions, as one would do in a practical scenario as well, the current autocorrelation matrix $\mathbf{R}_{xx}$ and crosscorrelation vector $\mathbf{r}_{xs}$ are re-used (hence the 'parameter transfer' in Figure 5.8) and not re-initialized each time from scratch.

Per 30 s decision segment, a decision about the attended speaker is made, resulting in a binary correct/incorrect decision. To provide a comprehensible plot when plotting the accuracy over time, these binary decisions are smoothed using a 29-point moving average (i.e., per segment taking the past and following 7 min into account). To assure a fair comparison between the fixed supervised and time-adaptive unsupervised decoder, the total accuracy is computed as the average over all binary decisions across all but the first 48 min of the first two sessions on the first day.

Lastly, similarly to Section 5.5.1, we evaluate the performance in case one or more electrodes are disconnected by simulating 0 to 12 disconnected electrodes after the first two sessions (see also Section 5.5.1). Per number of disconnected electrodes, ten random selections of electrodes are performed.

Results

Figure 5.9 shows the smoothed accuracy as a function of time of both the fixed supervised and time-adaptive unsupervised algorithm for both test subjects. As explained before, there is no test accuracy present in the first 48 min for the fixed supervised decoder, as it is trained on those sessions. During those first two sessions, the time-adaptive unsupervised decoder converges after ±25 min, starting from a random initial decoder. This is more or less in line with the results of Sections 5.4.3 and 5.4.5.

The fixed supervised decoder reaches a total accuracy of 80.2% (subject 1) and 78.5% (subject 2), while the proposed time-adaptive unsupervised decoder reaches a total accuracy of 80.2% and 83.0%. The latter thus performs on par

[1] Two 30 s decision segments as the decoder is only being updated every 60 s.

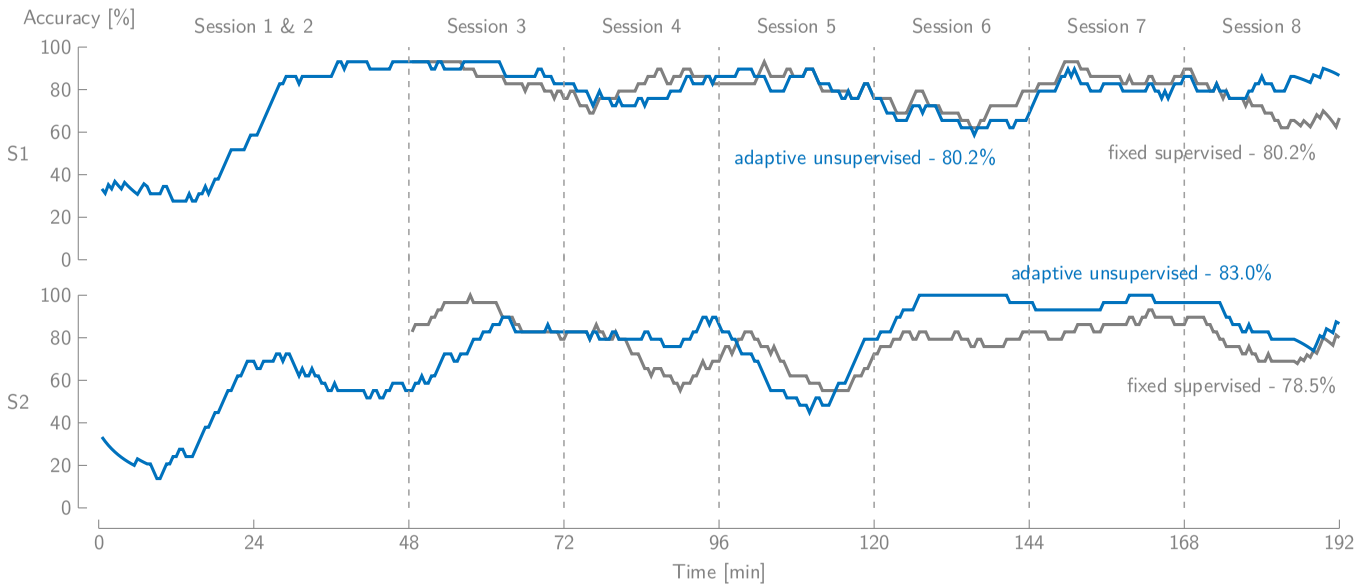


Figure 5.9: The smoothed accuracy of the fixed supervised and adaptive unsupervised decoder as a function of time for both subjects of Dataset C. The first 48 min are used as the training set for the fixed supervised decoder, explaining why there is no accuracy there. The final accuracies are computed from Session 3 onwards on 30 s decision windows.

with the former for the first subject while outperforming the former with 4.5% for the second subject. Furthermore, our approach does not require an a priori calibration session with the end-user but can be implemented in a plug-and-play fashion on a device. Lastly, more severe changes in the setup and conditions can occur. For example, Figure 5.7b shows the results of simulating disconnected electrodes. When no electrodes are disconnected, the accuracies are the same as in Figure 5.9. However, when one electrode is disconnected, the time-adaptive unsupervised decoder already outperforms the fixed supervised decoder with 6.0%, increasing to 16.6% when 12 electrodes are disconnected. This shows that the proposed method would be able to adapt to such changes, while the fixed supervised decoder only performs worse.

To evaluate whether, besides the favorable memory usage, the recursive implementation also benefits from the exponential weighting compared to the uniform weighting in a sliding window implementation in a time-adaptive context, we test the sliding window implementation with $K = 19$ (i.e., equivalent to $\alpha = \beta = 0.9$) but without repredictions of the labels (to be in line with the recursive implementation which also does not repredict previous labels). The resulting accuracy is 79.9% (subject 1) and 81.6% (subject 2). While this is for both subjects worse than the recursive implementation, we cannot draw firm conclusions about this based on two subjects alone. However, as expected, these results at least suggest that an exponential weighting is favorable compared to a uniform weighting in a time-adaptive context.

Although data from only two subjects are available in this experiment, hampering clear statistical conclusions, the results clearly show the potential of the proposed time-adaptive unsupervised decoder in a practical AAD use case.

5.6 Discussions and conclusion

We adapted the offline batch version of the unsupervised SR algorithm for AAD as proposed in Chapter 4 to a time-adaptive online version. This allows the decoder to automatically adapt to non-stationarities in the EEG and audio. We have developed both a sliding window implementation with repredictions of previous labels in a finite pool and a single-shot predict-and-update recursive implementation without repredictions. The latter has the advantage, as it results in similar performances (Section 5.4) for much less memory usage and computational requirements. We have selected the algorithm's hyperparameters via extensive experiments and validated these on an independent dataset. Furthermore, we explained why there are hardly any changes in labels when

using iterative repredictions in this time-adaptive context while this is the case in the batch-mode algorithm presented in Chapter 4.

We have also shown the additional benefit of the time-adaptive unsupervised decoder compared to the fixed supervised decoder in a time-adaptive scenario, for example, when simulating electrode disconnections (Section 5.5.1). When electrodes are disconnected, the former starts to clearly outperform the latter. Lastly, the proposed time-adaptive unsupervised decoder outperformed the fixed supervised decoder on a dataset that reflects a practical AAD use case (with testing across multiple sessions on different days; Section 5.5.2). Given that this dataset only contains two subjects, we are careful in drawing firm conclusions. The results, however, clearly show the potential of the proposed method.

As explained in Chapters 2 and 3, the SR method does not perform well enough on short decision segment lengths for the online AAD application. In Section 5.5.2, there were no switches in auditory attention present, such that a high accuracy could be obtained with 30 s decision segments. While this reduces the relevance of the proposed algorithm as the 'decision-maker' in AAD, it is an excellent candidate to provide reliable labels of the auditory attention to update a faster algorithm (such as the algorithms proposed in Part III) (see also Section 8.2).

In conclusion, the developed time-adaptive unsupervised SR method is an important step forward to the online application of AAD in neuro-steered hearing devices.

Part III

Decoding the spatial focus of auditory attention

6 | Common spatial pattern-based decoding of the spatial focus of auditory attention

This chapter is largely based on S. Geirnaert, T. Francart, and A. Bertrand, "Fast EEG-based Decoding of the Directional Focus of Auditory Attention Using Common Spatial Patterns," *IEEE Transactions on Biomedical Engineering*, vol. 68, no. 5, pp. 1557-1568, 2021.

ABSTRACT | As explained in Chapter 3, most state-of-the-art AAD algorithms employ an SR approach. This approach, however, performs poorly on short decision windows, while longer windows yield impractically long detection delays when the user switches attention. Therefore, we propose decoding the spatial focus of attention using filterbank common spatial pattern filters (FB-CSP) as an alternative AAD paradigm, which does not require access to the clean source envelopes. The proposed FB-CSP approach outperforms both the SR approach on short decision windows, as well as a CNN approach on the same task. We achieve a high accuracy (80% for 1 s windows and 70% for quasi-instantaneous decisions), which is sufficient to reach MESDs below 4 s. We also demonstrate that the decoder can adapt to unlabeled data from an unseen subject and works with only a subset of EEG channels located around the ear to emulate a wearable EEG setup. Given the high accuracy on very short data windows, the proposed algorithm is a major step forward towards practical neuro-steered hearing devices.

6.1 Introduction

The discovery that the cortical activity follows the envelope of the attended speech stream (Section 1.5) laid the foundation of a first class of AAD algorithms based on non-invasive neural recordings from EEG. These algorithms typically employ an SR approach in which a decoder reconstructs the attended speech envelope from the EEG (Section 3.2.1). The decoded envelope is then correlated with the speech envelopes of the individual speakers. The speaker corresponding to the highest correlation coefficient is identified as the attended speaker. As shown in Chapter 3, AAD algorithms using the SR approach, however, all suffer from the same limitations:

1. The SR approach takes too long to make a reliable decision. The AAD accuracy (the percentage of correct decisions) drastically decreases with shorter decision windows, especially below 10 s (Part I). A decision window corresponds to the signal length over which the correlation coefficients between the EEG-decoded envelope and the original speech envelopes are estimated, where short decision windows result in unreliable correlation estimates. This results in a speed-accuracy trade-off. In Chapter 2, it is shown that short decision window lengths are favorable in the context of robust AAD-based gain control during dynamic switching, even if they have a lower accuracy. Nevertheless, due to the low accuracy for these short decision window lengths, it theoretically takes more than 15 s to establish a reliable and controlled gain switch to the new attended speaker after the user switches attention (Chapter 3). This is impractically long for neuro-steered hearing device applications. The SR approach inherently suffers from this limited performance due to the decoding of a low-frequency envelope, which contains relatively little information per second, as well as due to the low SNR of the neural response to the stimulus in the EEG.

2. The SR approach requires the (clean) individual speech envelopes. Although several attempts have been made to combine speech separation algorithms with AAD (Section 1.7.2), the demixing of all speech envelopes adds a lot of overhead, and the demixing process often negatively affects AAD performance or may even completely fail in practical situations.

In this chapter, we employ a new paradigm that avoids these limitations, focusing on decoding the spatial focus of attention from the EEG rather than directly identifying the attended speaker. Inherently, this avoids the need to demix the speech mixtures into their individual contributions. Moreover, we hypothesize that this paradigm will improve AAD accuracy for short decision window lengths, as it is based on brain lateralization, which is an instantaneous

spatial feature rather than a correlation-based temporal feature. The approach proposed in this chapter is rather a classification-based than SR-based method and can therefore be classified into the 'linear direct classification' branch of the tree of AAD algorithms proposed in Figure 3.1.

This new AAD paradigm is justified by recent research that shows that the auditory attentional direction is spatio-temporally encoded in the neural activity (Section 1.6), ergo, that it could be possible to decode the spatial focus of attention from the EEG. Vandecappelle et al. proposed in [136] an AAD algorithm based on a CNN to decode the spatial focus of attention in a competing speaker scenario, which showed very good results on short decision windows (76.1% accuracy on 1 s decision windows) (see also Chapter 3 for a description). However, this CNN-based approach shows high inter-subject variability and requires large amounts of training data (for example, data of other subjects in combination with subject-specific data as in [136]) to train a subject-specific decoder. Therefore, in this chapter, we focus on data-driven *linear* filtering techniques, which typically require less training data, are more robust and stable, and are computationally cheaper, as well as easier to update. More specifically, we exploit the direction-dependent spatio-temporal signatures of the EEG using (filterbank) common spatial pattern (FB-CSP) filters, which are popular in various BCI applications [38, 170].

In Section 6.2, we concisely introduce the (FB-)CSP classification pipeline to determine the spatial focus of attention. In Section 6.3, we describe the data used to run experiments, the concrete choices for the FB-CSP filter design, and the performance metrics to transparently and statistically validate the experiments that are reported and analyzed in Section 6.4. Conclusions are drawn in Section 6.5.

6.2 Decoding spatial focus of attention using CSPs

In this section, we review the CSP procedure [170] to decode the spatial focus of attention. CSP filtering is one of the most popular techniques used for spatial feature extraction in BCI applications, for example, in motor imagery [38, 170, 171]. The goal is to project multi-channel EEG data into a lower-dimensional subspace that optimally discriminates between two conditions or classes based on variance (rather than separating the means as in LDA). This is established by optimizing a spatial filter in a data-driven fashion, which linearly combines the different EEG channels into a few signals in which this discriminative property is maximally present.

For the sake of an easy exposition, we first define CSP filtering for a binary AAD problem, i.e., decoding whether a subject attends to one of two speaker positions, in Sections 6.2.1 and 6.2.2. In Section 6.2.3, we explain how this can be generalized to more than two classes/directions. Finally, in Section 6.2.4, we explain how the method can be applied to EEG data from unseen subjects without the need for any ground-truth labels on their auditory attention.

6.2.1 CSP filtering

Consider a zero-mean C-channel EEG signal $\mathbf{x}(t) \in \mathbb{R}^{C\times 1}$, which can, on each time instance t, be classified into one of two classes $\mathcal{C}_1$ and $\mathcal{C}_2$ (for example, attending the left or right speaker). The goal is to design a set of K spatial filters $\mathbf{W} \in \mathbb{R}^{C\times K}$ that generate a K-channel output signal with uncorrelated channels $\mathbf{y}(t) = \mathbf{W}^{\mathrm{T}}\mathbf{x}(t) \in \mathbb{R}^{K\times 1}$, where the $\frac{K}{2}$ first filters maximize the output energy when $t \in \mathcal{C}_1$, while minimizing the output energy when $t \in \mathcal{C}_2$, and vice versa for the last $\frac{K}{2}$ filters.

For example, the first column $\mathbf{w}_1$ of $\mathbf{W}$ results in $y_1(t) = \mathbf{w}_1^{\mathrm{T}}\mathbf{x}(t)$, which should have a maximal output energy when $t \in \mathcal{C}_1$ and a minimal output energy when $t \in \mathcal{C}_2$:

$$\mathbf{w}_1 = \underset{\mathbf{w}}{\operatorname{argmax}} \frac{\frac{1}{|\mathcal{C}_1|}\sum_{t\in\mathcal{C}_1}\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}(t)\right)^2}{\frac{1}{|\mathcal{C}_2|}\sum_{t\in\mathcal{C}_2}\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}(t)\right)^2}$$

$$\Leftrightarrow \mathbf{w}_1 = \underset{\mathbf{w}}{\operatorname{argmax}} \frac{\mathbf{w}^{\mathrm{T}}\mathbf{R}_{\mathcal{C}_1}\mathbf{w}}{\mathbf{w}^{\mathrm{T}}\mathbf{R}_{\mathcal{C}_2}\mathbf{w}},$$

with $\left|\mathcal{C}_{1/2}\right|$ the number of time instances in $\mathcal{C}_{1/2}$ and

$$\mathbf{R}_{\mathcal{C}_{1/2}} = \frac{1}{\left|\mathcal{C}_{1/2}\right|}\sum_{t\in\mathcal{C}_{1/2}}\mathbf{x}(t)\mathbf{x}^{\mathrm{T}}(t) \tag{6.1}$$

the sample covariance matrices of class $\mathcal{C}_1$ and $\mathcal{C}_2$. Fixating the output energy when $t \in \mathcal{C}_2$, i.e., $\mathbf{w}^{\mathrm{T}}\mathbf{R}_{\mathcal{C}_2}\mathbf{w} = 1$, which is possible because $\mathbf{w}$ is defined up to a scaling, and solving the optimization problem using the method of Lagrange multipliers leads to the following necessary condition for optimality:

$$\mathbf{R}_{\mathcal{C}_1}\mathbf{w} = \lambda\mathbf{R}_{\mathcal{C}_2}\mathbf{w}, \tag{6.2}$$

which corresponds to a generalized eigenvalue decomposition (GEVD)/problem. It can easily be seen that the maximum is obtained for the generalized eigenvector (GEVc) corresponding to the largest generalized eigenvalue (GEVl). A similar

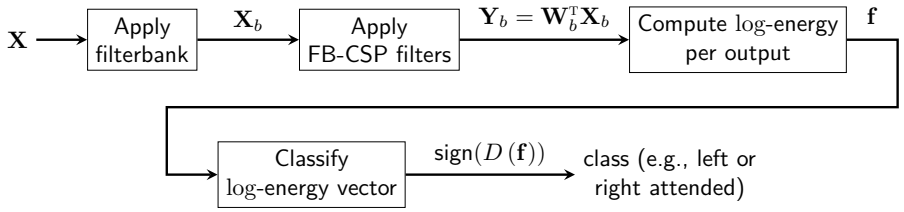


Figure 6.1: The FB-CSP filter outputs are used to generate the features that can be used to classify the EEG window $\mathbf{X}$.

reasoning can be followed for $\mathbf{w}_K$, which maximizes, respectively minimizes the output energy when $t \in \mathcal{C}_2$, respectively $\mathcal{C}_1$, and is equal to the GEVc corresponding to the smallest GEVl in (6.2). The other spatial filters can be found as the subsequent largest and smallest GEVcs. In its core essence, designing CSP filters thus corresponds to a joint diagonalization of the class-dependent covariance matrices [170].

6.2.2 Classification using CSP filters

The CSP filtering technique can now be employed in a classification pipeline, in which a newly recorded EEG signal $\mathbf{x}(t) \in \mathbb{R}^{C\times 1}$, containing C channels, is classified into one of two classes, representing different directions of auditory attention (Figure 6.1). The following sections describe the different components of this classification pipeline.

Filterbank CSP (FB-CSP)

Paramount for a well-performing CSP filtering is the selection of the appropriate frequency band related to the feature at hand (i.e., feature selection). For the case of AAD, one possibility is filtering in the α-band (Section 1.6). We here, however, do not want to make an a priori choice of the relevant frequency band(s). We thus adopt the so-called *filterbank* CSP (FB-CSP) technique, in which the EEG is first filtered into different frequency bands, after which the CSP filters are trained and applied per frequency band [38, 170, 171]. The filterbank thus results in B (number of frequency bands) filtered signals $\mathbf{x}_b(t) \in \mathbb{R}^{C\times 1}$, one per frequency band $b \in \{1, \ldots, B\}$, for all C EEG channels. The application of the pre-trained CSP filters per frequency band $\mathbf{W}_b \in \mathbb{R}^{C\times K}$ results in B K-dimensional output signals $\mathbf{y}_b(t) = \mathbf{W}_b^{\mathrm{T}}\mathbf{x}_b(t) \in \mathbb{R}^{K\times 1}$.

An alternative extension, which is not pursued here, is the so-called common spatio-*spectral* pattern filter, in which the relevant frequency bands are determined fully data-driven, as a spatio-temporal filter is optimized to be maximally discriminative [172]. This comes, however, at the cost of an increase in parameters and related problems with overfitting, in particular for high-density EEG data as used in this chapter. These problems can partly be overcome by using more advanced regularization or dimensionality reduction techniques on the extended spatio-temporal covariance matrices (for example, PCA [153] or the pre-selection of relevant time lags to introduce sparsity). Furthermore, a different filter basis than the Dirac basis could be chosen to reduce the number of parameters or to incorporate expert knowledge [153].

Feature extraction

The outputs of the FB-CSP filtering are now per decision window transformed into a feature vector $\mathbf{f} \in \mathbb{R}^{KB\times 1}$ that can be used for classification. This is typically done by computing the log-energy over these output signals per decision window [170], using a pre-defined decision window length τ:

$$\mathbf{f} = \begin{bmatrix} \log\left(\sigma_{1,1}^2\right) \\ \vdots \\ \log\left(\sigma_{K,1}^2\right) \\ \log\left(\sigma_{1,2}^2\right) \\ \vdots \\ \log\left(\sigma_{K,B}^2\right) \end{bmatrix},$$

with the output energy $\sigma_{k,b}^2$ of the k^{th} output $y_{k,b}(t)$, for the b^{th} frequency band:

$$\sigma_{k,b}^2 = \sum_{t=1}^{\tau} y_{k,b}(t)^2 ,$$

where τ is the number of time samples in the decision window. Note that the decision window length τ determines how much EEG data is used to make a single decision about the auditory attention of the subject. In a practical system, this will define the inherent delay to detect switches in attention.

Classification

The feature vector $\mathbf{f}$ is used as the input for a binary classifier to determine the spatial focus of attention. We here adopt Fisher's LDA, which is traditionally

used in combination with CSP filters [170]. In LDA, similarly to CSPs, a linear filter $\mathbf{v} \in \mathbb{R}^{KB \times 1}$ is optimized to provide the most informative projection. In this case, the most informative projection corresponds to maximizing the in between class scatter while minimizing the within-class scatter. This again leads to a GEVD, which can, in this case, be solved analytically, leading to the following solution [173]:

$$\mathbf{v} = \boldsymbol{\Sigma}_w^{-1} \left(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1 \right), \tag{6.3}$$

with $\boldsymbol{\Sigma}_w$ the covariance matrix of the features $\mathbf{f}$ computed across both classes, and $\boldsymbol{\mu}_{1/2}$ the class (feature) means. Choosing the threshold or bias as the mean of the LDA projected class means leads to the following decision function:

$$D(\mathbf{f}) = \mathbf{v}^{\mathrm{T}} \mathbf{f} + b,$$

with $\mathbf{v}$ defined in (6.3) and bias

$$b = -\frac{1}{2} \mathbf{v}^{\mathrm{T}} (\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2) \,. \tag{6.4}$$

Finally, $\mathbf{f}$ is classified into class one if $D(\mathbf{f}) > 0$ and into class two if $D(\mathbf{f}) < 0$.

6.2.3 Multiclass CSP classification

The classification scheme in Figure 6.1 can be easily extended to a multiclass scenario, in which multiple directions of auditory attention are combined. This can be achieved by applying the strategy of Sections 6.2.1 and 6.2.2 in combination with an appropriate coding scheme (e.g., one-vs-one, one-vs-all), both for the CSP and LDA step, or by approximating a joint diagonalization of all the class covariance matrices at once in the CSP block [170], and only applying a coding scheme to the LDA step. Note that the SR approach is also applicable for various directions/speakers [4, 113].

In this chapter, we adopt the popular one-vs-all approach in BCI research [170]. In this approach, an FB-CSP filter set and LDA classifier are trained for each direction to discriminate that particular direction from all the other directions. Given M directions (classes), this means that in (6.2), the $\mathbf{R}_{\mathcal{C}_2}$ is replaced by $\sum_{i=2}^{M} \mathbf{R}_{\mathcal{C}_i}$, i.e., the sum of the covariance matrices of all classes except class 1. Correspondingly, an LDA classifier is trained to discriminate direction 1 from all other directions. This is done for every other direction $m \in \{1, \ldots, M\}$. Given M directions (classes), this thus results in M different CSP/LDA pairs.

In the end, for a new window, the posterior probability of each classifier is computed using the multivariate normal distribution for the likelihood (which is assumed by LDA) and a uniform prior. To finally determine the correct class, the maximal posterior probability is taken over the M LDA classifiers.

6.2.4 CSP classification on an unseen subject

The FB-CSP filters and LDA classifiers can be trained *subject-specifically*, meaning that the training is based on EEG data from the actual subject under test. However, in a neuro-steered hearing device application, this would require a cumbersome per-user calibration session where the subject is asked to attend to specific speakers with the intention to collect ground-truth labels to inform the FB-CSP filter design. To eliminate this requirement, one could train an AAD model in a *subject-independent* manner, meaning that data from subjects other than the test subject are used in the training phase, as done in [74, 136] and Chapter 4 for the SR and CNN approaches. This pre-trained model could then be 'pre-installed' on every neuro-steered hearing device, using it in a 'plug-and-play' fashion.

However, it is known from the BCI literature that the FB-CSP method often fails in such subject-independent settings due to too large differences in the spatial/spectral EEG patterns across different subjects [174]. To improve performance, the data from the subject under test can be used to modify the pre-trained subject-independent FB-CSP filters/LDA classifier. We adopt here two popular approaches to perform such adaptations, without requiring ground-truth labels for the data of the unseen test subject:

1. A very effective way of unsupervised updating of an LDA classifier for BCIs has been proposed in [175]. They conclude that simply updating the bias of the LDA classifier in (6.4) results in a significant improvement. Here, we update the bias of the subject-independently trained LDA with the unlabeled subject-specific features (resulting from the subject-independent FB-CSP filters), as this only requires the global mean, which is label-independent:
$$D(\mathbf{f}) = \mathbf{v}^{(\mathrm{SI})^{\mathrm{T}}}\mathbf{f} + b^{(\mathrm{SS})},$$
with subject-independent coefficients $\mathbf{v}^{(\mathrm{SI})}$ as computed in (6.3) on the data from all other subjects, and the subject-specific bias computed as:
$$b^{(\mathrm{SS})} = -\mathbf{v}^{(\mathrm{SI})^{\mathrm{T}}}\boldsymbol{\mu}^{(\mathrm{SS})},$$
using the global mean $\boldsymbol{\mu}^{(\mathrm{SS})}$ over all features $\mathbf{f}^{(\mathrm{SS})}$ of the new subject. The only requirement is that the subject-specific data on which the bias is updated is approximately balanced over both classes.

2. Lotte et al. [174] found that a subject-independent FB-CSP method often fails potentially because of the too high spectral subject-to-subject variability when using many narrow frequency bands. To overcome this issue, we replace the filterbank setting with a single filter ($B = 1$) to

extract and pool a broader frequency range, of which the boundaries will be determined experimentally. This basically reduces the FB-CSP method to a CSP classification method with a prior bandpass filtering of the data. Note that only for the subject-independent experiments (Section 6.4.7), the FB-CSP method is reduced to a single frequency band. In all other subject-specific experiments, the FB-CSP approach is used.

6.3 Experiments and evaluation

6.3.1 AAD datasets

We apply the proposed FB-CSP classification method on two different datasets. The first dataset (Dataset A) has already been used extensively in previous work, mostly in the context of the SR approach, and consists of 72 minutes of EEG recordings for each of the 16 normal-hearing subjects, who were instructed to attend to one of two competing (male) speakers. The competing speakers were located at $-90°$ and $+90°$ along the azimuth direction, and there was no background noise. This dataset is used in all experiments, except those of Sections 6.4.3 and 6.4.4[1].

The second dataset (Dataset D) consists of 138 minutes of EEG recordings for each of the 18 normal-hearing subjects, again instructed to attend to one of two male speakers, however, now with background babble noise at different SNRs. Furthermore, per subject, different angular speaker positions are combined (i.e., different angular separation between the competing speakers): $-90°$ versus $+90°$, $+30°$ versus $+90°$, $-90°$ versus $-30°$, and $-5°$ versus $+5°$ (Figure 6.2). This second dataset allows us to validate the decoding of the spatial focus of attention for different angular separations and is used in Sections 6.4.3 and 6.4.4. Both datasets are recorded using a $C = 64$-channel BioSemi ActiveTwo system, using a sampling frequency of 8192 Hz.

6.3.2 Design choices

EEG bandpass filtering

Before CSP filtering, a filterbank is applied to the EEG, consisting of $B = 14$ 8^{th}-order Butterworth filters. The first filter corresponds to frequency band 1–4 Hz,

[1]The code for the subject-specific experiments on this dataset are available at https://github.com/exporl/spatial-focus-of-attention-csp.

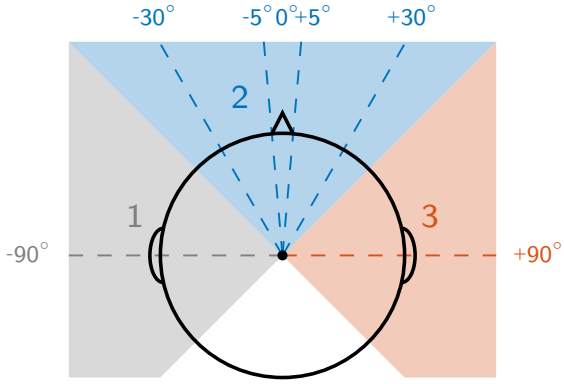


Figure 6.2: The competing speakers of Dataset D are located at different angular positions. The azimuth plane is divided into three angular domains, which are used in the multiclass problem of Section 6.4.4.

the second to 2–6 Hz, the third to 4–8 Hz. This continues, with bands of 4 Hz, overlapping with 2 Hz, until the last band 26–30 Hz. In this way, a similar range of frequencies is covered as in [136]. The group delay is compensated for per filter using the filtfilt-function in MATLAB, resulting in a zero-phase filtering. Afterwards, the EEG data is downsampled to 64 Hz. No further preprocessing or artifact rejection is applied, as the CSP filters already implicitly suppress EEG content that is irrelevant for discrimination between both classes through a spatial filtering per frequency band.

Covariance matrix estimation

To avoid overfitting in the estimation of the class covariance matrices in (6.2), the sample covariance matrices in (6.1) are regularized using ridge regression:

$$\mathbf{R}_{\mathcal{C}_{1/2}}^{(\mathrm{reg})} = \alpha \mathbf{R}_{\mathcal{C}_{1/2}} + \beta \mathbf{I},$$

with $\mathbf{R}_{\mathcal{C}_{1/2}}$ the sample covariance matrix from (6.1) and $\mathbf{I} \in \mathbb{R}^{C \times C}$ the identity matrix. The regularization parameters α and β are not estimated using CV but are analytically determined (details in [164, 165] or Sections 4.4.2 and 5.2.1). This method has proven to be superior in various BCI applications and is the recommended state-of-the-art covariance matrix estimator [38].

CSP filter design

As described in Section 6.2.1, traditionally, the GEVls are used to select an appropriate subset of filters, as they represent the relative output energies of each spatially filtered signal. However, these GEVls can be influenced by outlier segments with a very high variance, which consecutively can (negatively) affect the selection of the CSP filters. To avoid this issue, the filters are selected based on the ratio of *median* output energies (RMOE) between both classes [170], taken over all training segments with length equal to the maximal decision window length that is used in the analysis.

Furthermore, $K = 6$ CSP filters, corresponding to the 3 most discriminative filters for one and the other direction, are selected based on the cut-off point on the plot of sorted RMOEs.

6.3.3 Performance evaluation

The FB-CSP classification pipeline is first tested per subject separately using ten-fold CV. The data per subject are therefore split into segments of 60 s (30 s for Dataset D) and randomly shuffled into ten folds. This division into segments is performed in order to be able to do random shuffling over time, such that the impact of factors such as fatigue is minimized. Only in Section 6.4.2 and Appendix 6.A, a leave-one-(story+)speaker-out CV (LOSpO-CV) is performed, retaining the chronological order of the segments that originate from continuously recorded trials. For each 60/30 s segment, the mean is set to zero per channel. Furthermore, each segment is normalized over all channels at once (the Frobenius norm is set to one) to assign equal weight to each segment in the training phase. During the testing phase, the normalized 60/30 s segments are split into shorter sub-segments, referred to as 'decision windows' (of which the length will be varied). The significance level for the accuracy is determined via the inverse binomial distribution [74].

In Chapter 2, the importance of evaluating AAD algorithms on different decision window lengths, i.e., the amount of data used to make an AAD decision, has been stressed. In typical AAD algorithms, a trade-off exists between the decision window length and accuracy. In Chapter 2, the optimal trade-off is determined by means of a criterion based on the expected time it takes to perform a stable gain switch in an attention-steered gain control system. Based on a stochastic model for the latter, and for any given decision window length, the expected time to switch the gain between speakers is minimized under the constraint of guaranteeing a pre-defined level of 'stability' to avoid spurious gain switches due to errors in the AAD decisions. The latter is achieved by increasing the

number of gain levels, thereby increasing the gain switch duration. The optimal trade-off point between decision window length and accuracy is found as the one that leads to the shortest ESD under this model, which is referred to as the MESD. The MESD is a single-number metric, facilitating the use of statistical tests to compare different AAD algorithms, as it resolves the inherent trade-off between decision window length and AAD accuracy. In Chapter 2, we found that the optimal decision window length selected within the computation of the MESD consistently shows the importance of *short* decision window lengths ($< 10\,\mathrm{s}$), allowing faster and more robust switching between speakers despite the lower AAD accuracy. To determine the accuracies on shorter decision window lengths, the left-out segments are split into shorter decision windows, on which the testing routine in Figure 6.1 is applied. Note that the MESD is a theoretical metric and only provides a theoretical prediction on how an optimized AAD-based gain control algorithm would track attention switches. Here, we do not experiment with data containing actual attention switches.

The hyperparameters of the LDA classifier are optimized on the CSP output energies of the training set using five-fold CV.

6.4 Results and discussion

6.4.1 Comparison with SR approach

The FB-CSP classification method is compared to the current state-of-the-art AAD method, which adopts the SR approach, on Dataset A. Here, CCA is used, which is considered to be one of the best decoding methods to date, outperforming other backward and forward models (Chapter 3). In CCA, a jointly forward (i.e., mapping the stimulus envelope to the EEG) and backward (i.e., mapping the EEG to the stimulus envelope) model is trained and applied to new data [153]. The attended speaker is identified by classifying the difference between the canonical correlation coefficients of the competing speakers using an LDA classifier. A forward lag of 1.25 s is used on the speech envelopes, and a backward lag of 250 ms is used on the EEG as in Chapter 3. The CCA method is tested using the same ten-fold CV procedure as for the FB-CSP method. The number of correlation coefficients used in the LDA classification is determined by an inner ten-fold CV loop. No a priori PCA or change of filter basis as in [153] is used. The EEG and speech envelopes, which are extracted using a power-law operation with exponent 0.6 after subband filtering [1], are filtered between 1–9 Hz (thus mainly without α/β-activity, which was determined to be optimal for linear SR [1, 4, 74, 113, 134]) and downsampled to 20 Hz. Note that this method employs an inherently different strategy for AAD than FB-CSP, by

(in a way) reconstructing the attended speech envelope rather than decoding the spatial focus of attention.

In Figure 6.3a, it is observed that this SR approach is characterized by a degrading accuracy for shorter decision window lengths, while the accuracy of the FB-CSP method barely decreases. It thus clearly outperforms the SR approach for short decision window lengths. This is one of the most important properties of this new strategy for AAD to decode the spatial focus with FB-CSP rather than to reconstruct the stimulus. This effect was also seen in [136] and Chapter 3, where the spatial focus is decoded based on a CNN. While the SR approach tries to determine the attended speaker by reconstructing the attended speech envelope, the FB-CSP method only needs to *discriminate* between two angular directions, which is an inherently easier filter design strategy. Furthermore, in the former, correlation is used as a feature, of which the estimation is inaccurate when computed on short decision windows, in particular because the correlation coefficients observed in SR are very small, making their estimation susceptible to noise. Lastly, as mentioned before, the FB-CSP method is mainly based on an instantaneous spatial feature (brain lateralization) rather than a temporal feature.

Note that the accuracy of the FB-CSP method exhibits a higher inter-subject variability than the SR method. We do not consider this as a major disadvantage of the FB-CSP method, as, for example, on 1 s decision windows, performance is still better for *all* subjects compared to CCA. On average, there is a 20% gap in accuracy for 1 s decision windows.

For long decision window lengths, however, CCA outperforms the FB-CSP method. To resolve this trade-off and to (statistically) determine which method performs better in a context of neuro-steered gain control for hearing devices, we use the MESD metric, a relevant criterion for AAD that optimizes the speed-accuracy trade-off and thus resolves the inconclusiveness based on the performance curve (Chapter 2). Figure 6.3b shows the MESDs per subject, for both algorithms. It is clear that FB-CSP (median MESD 4.1 s) results in much faster switching than CCA (median MESD 17.0 s). A Wilcoxon signed-rank test confirms that there indeed is a significant difference between the MESD for the FB-CSP versus CCA method ($n = 16, p < 0.001$). The sustained performance for short decision window lengths thus results in a superior performance (for all subjects) of the FB-CSP method over CCA. We note that the MESD is, by definition, longer than the decision window length (Chapter 2). In particular, the theoretical lower limit for the MESD is 3 s when using a minimal decision window length of 1 s[2] (Chapter 2).

[2]The theoretical lower limit of the MESD is equal to 3× the shortest decision window length that is tested with, as for 100% accuracy, three steps must be taken in the Markov chain (Chapter 2).

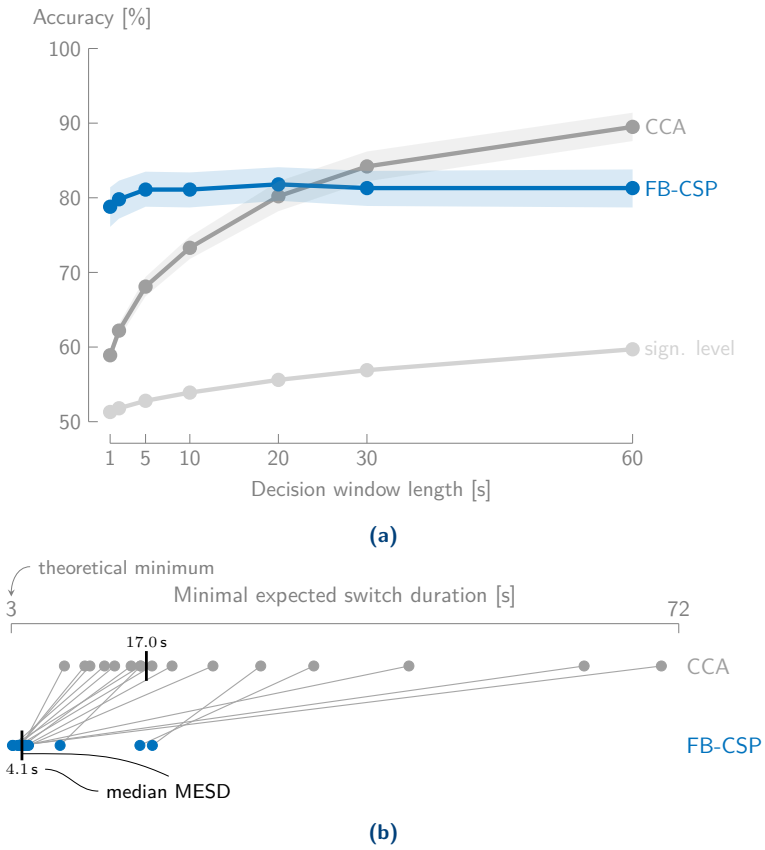


Figure 6.3: **(a)** The accuracy (mean ± standard error of the mean across subjects; Dataset A) of the FB-CSP classification method barely decreases for shorter decision window lengths and outperforms the SR approach (CCA) for short decision window lengths. Note that the significance level ('sign. level') decreases for shorter decision window lengths due to the higher number of test windows. **(b)** The median MESD (black vertical line) is significantly lower (better) for the FB-CSP method than for the SR approach (CCA). Each dot represents one subject, the lines connect the same subjects across methods.

6.4.2 Comparison with convolutional neural network approach

Vandecappelle et al. [136] used a CNN to perform the same task, i.e., decoding the spatial focus of attention. This CNN approach has been validated on the same dataset (Dataset A), but with a different testing procedure to avoid overfitting on speakers, i.e., LOSpO-CV instead of random CV. To provide an honest and transparent comparison of our FB-CSP method with this CNN method, we have cross-validated the performance of the FB-CSP method in the same way as in [136], at the cost of less training and testing data. While data of other subjects are included in the training of the CNN method as a regularization technique [136], this is not done for the FB-CSP method. The EEG data are filtered between 1–32 Hz, as proposed in [136] and equivalent to the FB-CSP method.

Given the performances in Figure 6.4, we, first of all, want to stress that the results of the FB-CSP method for a LOSpO-CV are very similar to using a random CV (Figure 6.3). This confirms that, as opposed to the CNN method, our FB-CSP method does not overfit on speakers or stories, which could occur when using random CV. For the CNN method, the results were significantly better when not leaving out the speaker and/or story in the training set, which could be a sign of overfitting [136]. Furthermore, our FB-CSP method does not perform worse than the CNN method, as a Wilcoxon signed-rank test ($n = 15, p = 0.85$, one outlier subject removed) shows no significant difference based on the MESD (Figure 6.4b).

To conclude, we have identified the following advantages of the FB-CSP method over the CNN method:

- The FB-CSP method does not perform worse than the CNN method, it even tends to outperform it.
- The FB-CSP method shows less inter-subject variability and is more stable (see the standard error of the mean in Figure 6.4a and the spread in Figure 6.4b).
- The FB-CSP method requires less training for a better performance. The CNN method uses training data of all (other) subjects, including the test subject, to avoid overfitting.
- The FB-CSP method has a lower computational complexity, which is paramount to be applicable in mobile and wireless hearing devices.

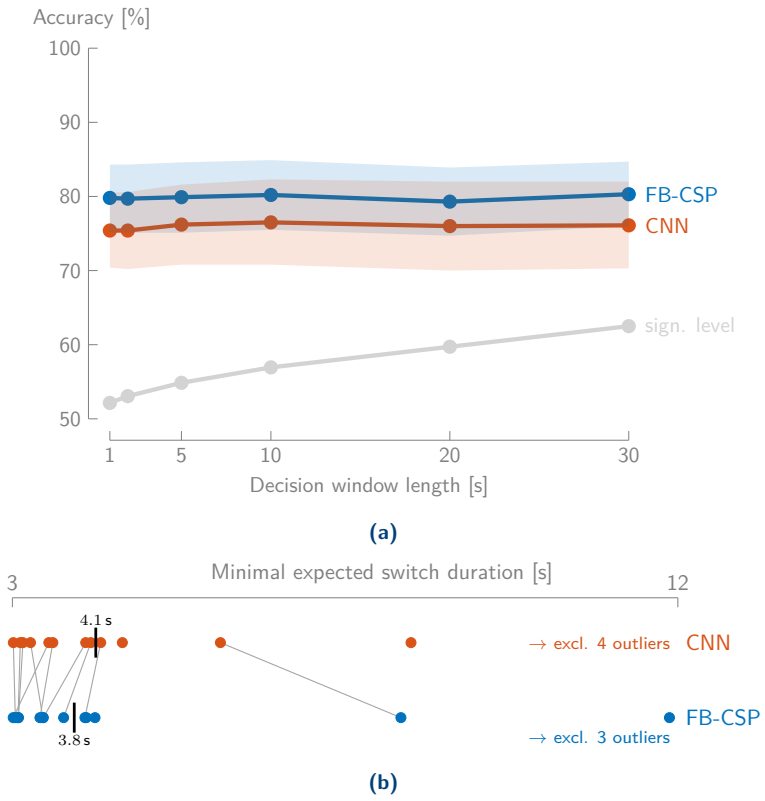


Figure 6.4: **(a)** The FB-CSP method outperforms the CNN method with on average $\approx 4\%$ in accuracy (mean $\pm$ standard error of the mean; Dataset A). **(b)** The median MESD is lower for the FB-CSP method than for the CNN method. Note that when the MESD of a subject is not connected to the corresponding MESD, it corresponds to an outlier value of the other method.

6.4.3 Binary FB-CSP classification at various speaker positions

Whereas in the previous experiments, the competing speakers are located at $-90°/+90°$, this section treats binary AAD classification at the various speaker positions that are present in Dataset D. Figure 6.5 shows the performance of the FB-CSP classification method. For each pair of competing speaker positions, all babble noise conditions are pooled, and the FB-CSP classification method is applied. Each pair of positions is thus treated separately, with independently trained CSP filters and LDA classifiers.

First of all, the results in Figure 6.5a confirm and reproduce the previous results from Figure 6.3a. The accuracy for the $-90°/+90°$ condition is on average even 10% higher, and there is a smaller inter-subject variability (standard deviation is on average $\approx 7\%$ over all decision window lengths, while this was $\approx 10\%$ in Dataset A). A possible explanation for this difference in performance is that due to the presence of background noise, the spatial cues become more important; or that the subject has to focus harder, thereby generating stronger neural responses. A similar advantageous effect of the presence of background noise for SR was observed in [4].

Furthermore, these results allow analyzing the effect of the angular speaker separation on the decoding performance. The main result from these performances is that decoding the spatial focus of attention from the EEG using the FB-CSP classification method still works for various angular scenarios, and even when the speakers are positioned very closely together ($-5°/+5°$) or are positioned at the same side of the head ($+30°/+90°$ and $-90°/-30°$). The MESDs in Figure 6.5b confirm these findings.

As can be expected, the decoding for the $-90°/+90°$ scenario is easier than in the other scenarios. Although the decoding will fail when speakers are co-located at the same spatial position, the FB-CSP method still succeeds in reliably discriminating between very closely positioned speakers at $-5°/+5°$. Furthermore, as the results for $-5°/+5°$ are still better than when the competing speakers are positioned at the same side of the head, it seems that when speakers are located at different sides of the head, this provides a substantial advantage in decoding the spatial focus of attention. However, even when speakers are located at the same side of the head, the method finds sufficient spatio-temporal discriminative patterns to differentiate between speaker locations.

As an important consequence of these results, the FB-CSP method can be used as a basic building block for a new AAD strategy in which, for example, the whole plane along the azimuth direction is split into angular domains. Depending on the multiclass coding strategy, several FB-CSP filters are then

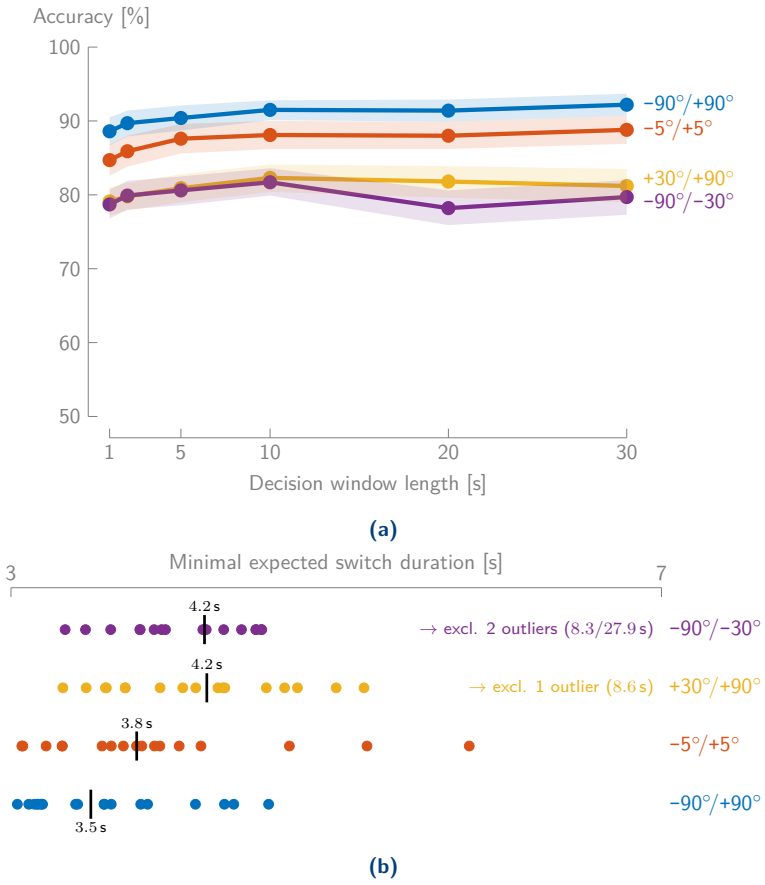


Figure 6.5: **(a)** The FB-CSP classification method performs well for all speaker separation angles (Dataset D). Again, the accuracy (mean ± standard error of the mean) of the FB-CSP classification method barely decreases for shorter decision window lengths. **(b)** The median MESD is lower (better) for the $-90°/+90°$ than for the other scenarios. Decoding the spatial focus of attention for speakers that are positioned at the same side of the head is harder than when they are symmetrically positioned on different sides of the head.

combined to locate the attended speaker in the plane and to steer a beamformer into the correct direction. This AAD strategy is tested in the following section.

6.4.4 Multi-condition and -class FB-CSP classification

Using Dataset D, we can verify whether a multi-condition or -class strategy is feasible. In the first experiment, all data are pooled, and the FB-CSP classifier tries to determine whether the *left- or right-most* speaker is attended. In the second experiment, all angles are divided into three angular domains (left/frontal/right) as depicted in Figure 6.2.

Classifying the left/right-most speaker as attended speaker

Instead of training the FB-CSP method for each angular condition separately, all conditions can be pooled, and the FB-CSP method can be trained to determine whether the user is listening to the left-most or right-most speaker (in a two-speaker scenario), *independent* of where these speakers are positioned in the plane. As a consequence, a speaker positioned at $-30°$ (which is located at the left side of the head) can be the right-most attended speaker, relative to $-90°$, while $+30°$ (which is located at the right of $-30°$) can be the left-most attended speaker, relative to $+90°$. This angular condition-independent FB-CSP classifier could then be used generically to steer a beamformer or to select the attended speaker, provided the angular positions of the competing speakers are known or can be detected from a hearing device's microphone array. To test this, all the data of Dataset D are pooled and randomly divided into ten folds. Note that a limitation of this experiment is that the different speaker positions only appear in fixed pairs and that not every position is combined with all other positions.

Figure 6.6 shows that the accuracy when classifying attention to the left/right-most speaker is still high (77.7% on average over all decision window lengths), although lower than when classifying each condition separately (Figure 6.5a). This confirms that this strategy is viable.

When investigating the MESDs per angular condition (still when classified all together), it is clear that there are two groups (Table 6.1): the first group contains the conditions where the competing speakers are located along different sides of the head and show only a small increase in MESD compared to when they are classified separately (compare with Figure 6.5b), while there is a larger increase in MESD when the competing speakers are positioned at the same side of the head. Furthermore, the first group shows a lower MESD than the latter one.

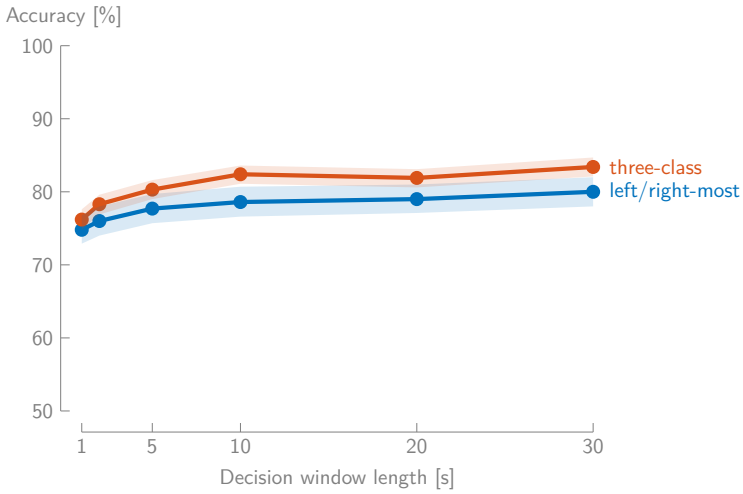


Figure 6.6: These performance curves show that the accuracy (mean ± standard error of the mean; Dataset D) is still high even when pooling all conditions and only classifying the attention to the left/right-most speaker, or when dividing the upper half-plane into three angular domains as in Figure 6.2.

Angular condition	MESD LR-most [s]	MESD sep. [s]
−5°/+5°	4.53	3.77
−90°/+90°	3.77	3.49
+30°/+90°	5.74	4.20
−90°/−30°	8.22	4.19

Table 6.1: The median MESD is generally lower when the speakers are located on different sides of the listener. Furthermore, the MESDs for left/right-most classification (LR-most) are higher compared to the case where each condition is classified separately (MESD sep.; see Figure 6.5b).

Classifying between left/frontal/right spatial focus

The intuitive multiclass extension of the binary classification of only two angular conditions is to classify multiple speaker positions simultaneously, i.e., determining the spatial focus of attention among several possibilities. A possible strategy could be to divide the azimuth plane into different angular domains, which are classified together. In this way, a beamformer could be steered towards the correct angular domain (without also having to estimate the direction of arrival (DOA) of each speaker separately from the microphone recordings). The higher the spatial resolution of the multiclass strategy, the lower the chance that multiple speakers are present in the same angular domain (in case of multiple competing speakers), but the higher the misclassification error. In case multiple speakers are detected within each angular domain, more angle-specific classifiers or the aforementioned strategy of classifying the left/right-most classifier (Section 6.4.4) could be used as a complementary approach.

To test the feasibility of this strategy, we divide the azimuth plane into three classes based on speaker position as in Figure 6.2. The segments in Dataset D are divided into these classes accordingly. Note that the same limitation as before (limited speaker pairs) holds here and that there are no other positions present than $\pm 90°$ in domains one and three. A one-vs-all coding scheme is used, which means that there are three binary classifiers trained, which each classify one angular domain versus the other two domains combined.

Figure 6.6 shows the performance curve for this three-class problem. The accuracies are very high and show low subject-to-subject variability (standard deviation $\approx 5.6\%$ over all decision window lengths). Note that the accuracy decreases faster for shorter decision window lengths than usual. This effect is, to a lesser extent, also present in the binary case and is amplified here because of the multiclass nature of this problem. However, the decrease is still very limited and results in short switch durations (median MESD of 4.32 s over all angular domains, 4.58 s for switching to domain 1, 4.01 s for switching to domain 2, and 5.13 s for switching to domain 3).

6.4.5 Channel selection

For the FB-CSP method to be applicable in the context of neuro-steered hearing devices, which is an inherently mobile application, we test the method with a reduced set of EEG channels. However, we do not adopt a traditional data-driven feature/channel selection method but take a recording system-based/application-based point of view. The five electrodes closest to each ear are selected from

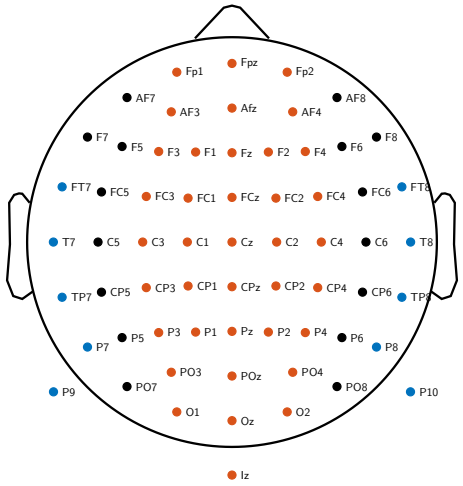


Figure 6.7: The five electrodes of the 64-channel BioSemi system closest to the ear are selected for the channel selection in Section 6.4.5 (blue). In Section 6.4.8, 38 central electrodes are chosen (orange).

the 64-channel BioSemi system (see the blue channels on Figure 6.7). This can be viewed as a representative selection that mimics current behind-the-ear EEG approaches such as the cEEGrid array [30], which has also been used for AAD [118]. However, it is noted that our analysis is not fully representative of an actual cEEGrid setup due to different recording equipment and different electrode positions. We mainly want to verify whether decoding the spatial focus of attention is possible while dominantly measuring from the electrodes on the temporal lobe.

To eliminate the dependence on an 'external' or joint reference electrode, the selected EEG channels are re-referenced using a common average reference for each ear separately. By averaging and re-referencing per ear separately, the two sets of ear channels are galvanically isolated, i.e., emulating two standalone EEG sensor devices that do not have to be connected with a wire. Furthermore, common average referencing is used to eliminate the need for selecting a particular reference electrode. Per ear, one random (as CSP filtering is invariant to the removed channel) re-referenced EEG channel is removed to avoid rank-deficiency in the EEG covariance matrices, effectively leading to 4 channels per ear ($C = 8$). After the removal of the other channels and the re-referencing, the complete FB-CSP pipeline (Figure 6.1) is retrained and evaluated using the reduced set of EEG channels.

Figure 6.8a shows that the decrease in accuracy on Dataset A (binary classification) when selecting the ear channels is limited to $\approx 5.6\%$ on average. Furthermore, the median MESD increases from 4.10 s (64 channels) to 4.74 s, which is statistically significant ($n = 16, p < 0.001$) but is still limited. Lastly, from Figure 6.8b, it can be seen that there is only a limited increase in variability over subjects.

Furthermore, also the performance of CCA is shown, using the same reduced set of channels and corresponding re-referencing method. The accuracy decreases on average with $\approx 10\%$ over all decision window lengths (Figure 6.3a) and does not outperform the FB-CSP method anymore on long decision window lengths. The median MESD drastically increases as well (Figure 6.8b). The SR approach thus suffers more from the channel reduction and is completely outperformed by the FB-CSP method, which is another advantage of the newly proposed method.

We conclude that decoding the spatial focus of attention with the FB-CSP method using a reduced set of channels close to the ear could be possible, but that there is more research required to further validate this approach.

6.4.6 Performance on very short decision window lengths (< 1 s)

Figure 6.9 shows the performance of the FB-CSP method on Dataset A (binary classification) for 64 channels and the channels close to the ear (Section 6.4.5) for decision window lengths below 1 s. Below 1 s, the accuracy further degrades, with a limited loss of $\approx 5.5\%$ accuracy on 31.25 ms decision windows and $\approx 8.5\%$ on 15.63 ms decision windows compared to 1 s decision windows. As a result, for both setups, there still is an acceptable performance when taking quasi-instantaneous decisions, resulting in a median MESD of 76.5 ms (64 channels) and 195.0 ms (ear channels) over all subjects. Note that caution is needed when interpreting these MESD values, as on such short decision window lengths, the independence assumption of the Markov model underlying the MESD metric is gravely violated due to the significant autocorrelation values of EEG signals below 1 s lags. The actual time to achieve a sufficiently stable switch may be higher than the one predicted by the model behind the MESD metric.

While it may seem surprising that the method can still decode the spatial focus of attention quasi-instantaneously (< 32 ms) with an accuracy that is better than chance, we note that CSP only exploits spatial information (differences between channels) rather than temporal information. Integrating over a longer time window only helps to achieve a better estimate of the log-energies that are

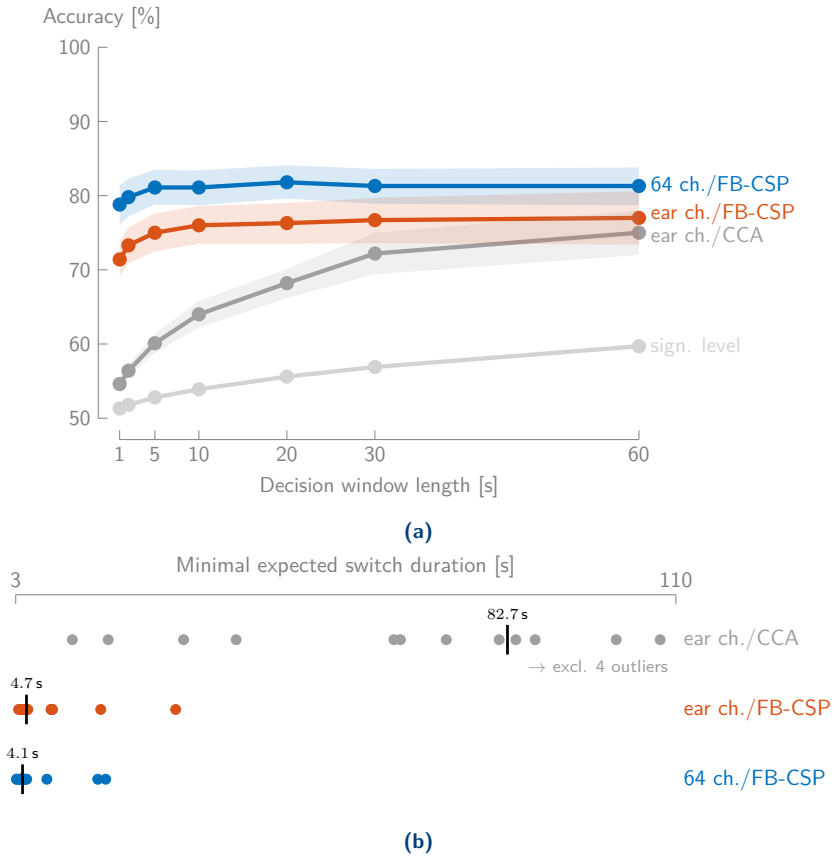


Figure 6.8: **(a)** The mean accuracy (± standard error of the mean) when using only ten electrodes close to the ear on Dataset A (binary classification) decreases relatively little compared to the full 64-channel setup. **(b)** There is a limited increase in median MESD when selecting ten electrodes for the FB-CSP method, while the CCA method greatly suffers from the channel reduction (compared to Figure 6.3b).

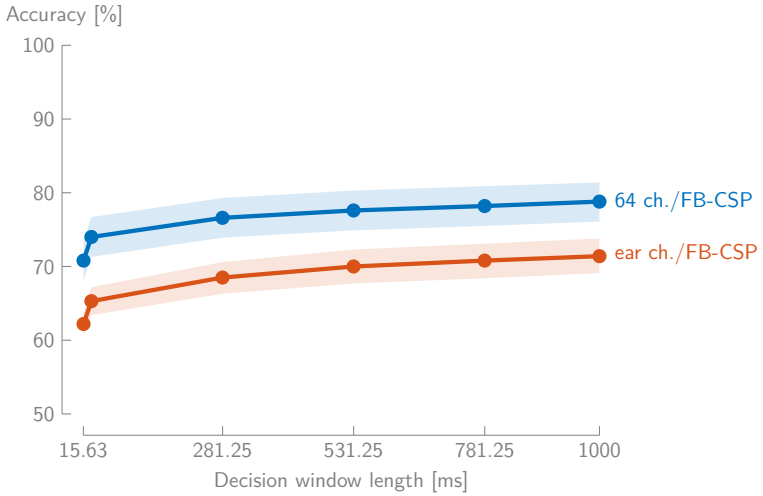


Figure 6.9: The performance curves (mean ± standard error of the mean) of the FB-CSP method degrade below 1 s decision window lengths while demonstrating acceptable performance even for quasi-instantaneous decisions (Dataset A, binary classification).

fed to LDA, which is the reason behind the slight increase in performance for longer decision windows (compared to instantaneous log-energy estimates). In the case of CSP, the length of the decision window is less critical than in SR approaches, where temporal modulations in the speech envelopes are exploited and where the decision window length directly determines how much of this information is available for discrimination between both speakers. Furthermore, in the case of FB-CSP, the estimation errors on the log-energies (due to quasi-instantaneous estimation) can be further compensated by the LDA classifier by exploiting redundancy in the different filterbands and CSP components to make a reliable decision. Lastly, although the FB-CSP method makes a decision based on a few samples, because of the filterbank on the EEG, these samples are also the result of a weighted integration of previous samples. This means that effectively more samples than the number of samples in the decision window are used.

6.4.7 CSP classification on an unseen subject

In the preceding experiments, the FB-CSP filters and LDA classifiers are trained subject-specifically. Here, we test the viability of the subject-independent

approach of Section 6.2.4, to improve the practical applicability of this method in neuro-steered hearing devices. The same (FB-)CSP classification pipeline (Figure 6.1) and design choices (Section 6.3.2) as before are used, but now tested on Dataset A (binary classification) in a *LOSuO-CV* manner. Per test subject, the (FB-)CSP filters and LDA classifier are trained on the 15 other subjects. Without using any of the adaptations from Section 6.2.4, the subject-independent FB-CSP method (SI-FB-CSP) exhibits a large drop in performance in comparison with the subject-specific FB-CSP method (SS-FB-CSP) (Figure 6.10a).

Updating the bias as in Section 6.2.4 results in a substantial increase of performance of $\approx 4\%$ (SI-FB-CSP-bias-update). The second adaptation reduces the FB-CSP method to a CSP method by using a single frequency band (the β-band: 12-30 Hz, $B = 1$), which was experimentally determined (see Section 6.4.8). Using this CSP method in combination with a bias update of the LDA classifier results in another increase of accuracy (Figure 6.10a; SI-CSP-bias-update).

The best subject-independent CSP classifier, with a bias update and only one frequency band (SI-CSP-bias-update), is compared with the subject-specific FB-CSP classifier (SS-FB-CSP) in Figures 6.10a and 6.10b. Note that using a single frequency band for the subject-specific method (SS-CSP) results here in a $\approx 2\%$ decrease in accuracy over all decision window lengths. From the MESD, we can see that the subject-independent method quite nicely approximates the performance of the subject-specific method. For two subjects, the subject-independent method even performs better than the subject-specific FB-CSP method. However, there still is a significant difference (Wilcoxon signed-rank test: $n = 16, p = 0.0023$). Furthermore, from Figure 6.10b, it can be seen that the subject-independent method has a larger spread, with more negative outlier values.

We conclude that the subject-independent CSP classification on average approximates the performance of a subject-specific FB-CSP classifier in terms of MESD but that there is no guarantee that it will work on every subject. However, this slightly worse performance is traded for practical applicability, as no a priori calibration session per user is required.

6.4.8 Decoding mechanisms

Given that it is possible to decode the spatial focus of attention with CSPs, it is relevant to get a handle on what drives the decoding. To investigate which frequency bands are most important, the subject-independent FB-CSP pipeline is trained on all subjects with $B = 4$ filterbands, corresponding to the main EEG frequency bands (1–4 Hz (δ), 4–8 Hz (θ), 8–12 Hz (α), and 12–30 Hz (β)).

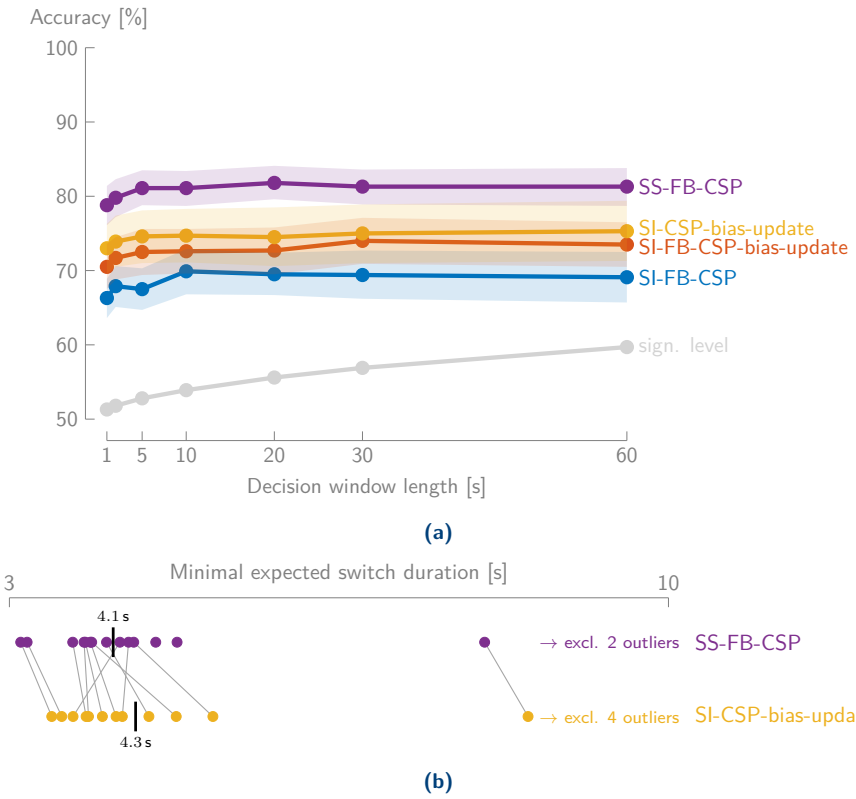


Figure 6.10: **(a)** Using a bias update (SI-FB-CSP-bias-update) and only one frequency band (SI-CSP-bias-update) in the subject-independent CSP classification method on Dataset A (binary classification) results in a substantial increase of performance over the baseline (SI-FB-CSP) (mean± standard error of the mean). **(b)** The median MESD of the subject-independent CSP classifier (SI-CSP-bias-update) is very close to the one of the subject-specific FB-CSP classifier (SS-FB-CSP). There is, however, a larger spread, with more negatively (higher) outlying MESD values. Note that when the MESD of a subject is not connected to the corresponding MESD, it corresponds to an outlier value of the other method.

The mean LOSuO accuracy over all subjects using a 60 s decision window length is 79.7%[3]. To assess the importance of each band, the $K = 6$ energies related to each band are left out (while keeping all others), leading to a decrease in accuracy to 79.0% for the δ-band, 79.3% for θ-band, 79.0% for the α-band, and 73.2% for the β-band. This indicates that the β-band is the most important band, motivating the choice of this band in Section 6.4.7. Similar conclusions have been drawn in [136, 176]. Furthermore, the performance does not degrade over time when the attention is sustained (see Appendix 6.A), which has been reported in the context of α-power lateralization [100].

Figure 6.11 shows the spatial activations of the β-band CSP filters. These topographic maps show activations mainly above the fronto-temporal cortex, consistent with the β-band activity found in [136, 176]. However, caution is needed when interpreting these spatial maps: the CSP filters implement a so-called 'backward' decoding model, which could implicitly also perform suppression of non-related EEG activity and artifacts, and can thus result in misleading interpretations [177]. To make the spatial maps as interpretable as possible, eye (blink) artifacts have been removed with ICA, and muscle artifacts have been removed with CCA [35], making it impossible for the CSP filters to reconstruct and exploit them. Note that because of the artifact removal, the spatial filters shown in Figure 6.11 do not correspond to the ones applied in the experiments. We merely try to highlight the neural underpinnings of the spatial filters by removing artifact-related activity before computing the CSP filters. As such, the topographic plots are not affected by the artifact removal mechanism that would normally be implicitly present in the CSP filters themselves (i.e., if such an implicit artifact removal would help in maximizing the discrimination between the classes). We reiterate that (linear) artifact removal is unnecessary in the experiments, as the CSP filters can deal with artifacts. Furthermore, the subject-specific performance on 60 s windows, using only the β-band, with and without mentioned artifact removal is very similar (77.2% versus 79.0% respectively), which indicates that the artifacts are not harming nor driving the CSP-based decoder.

Whether the CSPs exploit neural information or some correlated artifact signal (e.g., eye artifacts, muscle activity) is impossible to determine. Intuitively, the CSPs could potentially exploit two specific types of artifact signals: eye artifacts (for example, lateral movements) and muscle activity (especially subtle directive ear movements [178]). However, there are several indications that the CSP filters do not exploit these effects and indeed focus mainly on neural activity.

It is unlikely that the CSPs exploit eye artifacts, as they are primarily contained

[3] As the data of the subject under test is used in the CSP training (but not in the LDA training), this accuracy is slightly higher than in Figure 6.10.

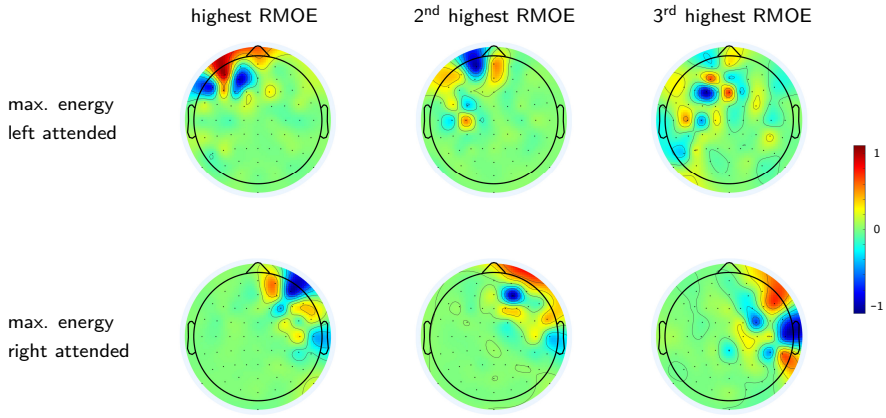


Figure 6.11: The topographic plots of the six spatial β-band CSP filters, computed on all data of all subjects of Dataset A, show mainly fronto-temporal activity. The filters of the first row maximize the output energy when left is attended, while those in the second row maximize the output energy when right is attended. The columns correspond to different RMOEs.

in the δ- and θ-band, whereas the CSP filters focus on β-band activity. Secondly, the explicit removal of the eye blinks using ICA does not affect the performance (80.0% on 60 s decision windows). Furthermore, the decoding also works well when the competing speakers are located at the same side of the head (Section 6.4.3) and even when the subjects are asked to fixate on a cross (tested on a second dataset of [3], results not shown).

As ear movements spectrally (β- and γ-band) overlap with the information used by the CSPs, it is more difficult to exclude the exploitation of subtle ear movements [178]. There are, however, two counterindications. Firstly, the approach also works for speakers located at the same side of the head (Section 6.4.3). Secondly, when using only the 38 most central electrodes (out of the 64 channels) furthest away from the ears (see the orange electrodes in Figure 6.7) and the β-band activity, we still obtain a subject-specific accuracy (on Dataset A) of 74.1% on 60 s windows. This at least shows that the decoding still works when only using the central channels, and thus most probably while not being able to pick up ear muscle activity. Furthermore, the topographic plots in Figure 6.11 show that the CSP filters also exploit channels that are rather far away from the ears, even when the channels close to the ears are included in the data-driven design.

6.5 Conclusion

We have shown that a (FB-)CSP classification method is capable of decoding the spatial focus of attention solely based on the EEG. An inherent limitation of this approach is that it requires the competing speakers to be spatially separated. Furthermore, this spatial separation needs to be perceived by the user, which is more difficult for certain hearing-impaired populations [101].

The proposed method has shown to not only outperform the classical SR approach for AAD in a two-speaker situation but does also not perform worse than a computationally more complex CNN approach that performs the same task [136]. It achieves practically viable MESDs below 4 s, which has not been achieved by any other AAD method so far (Chapter 3). Furthermore, the proposed method has several important advantages, which are essential for practical usage in neuro-steered hearing device applications:

1. the FB-CSP method does not require clean speech envelopes (in contrast to the traditional SR approach), such that the extra (error-prone) speech separation step for AAD can be avoided,
2. the performance barely decreases for short decision window lengths and still achieves acceptable performance for quasi-instantaneous decisions, potentially resulting in very fast and robust switching between speakers,
3. the method still works using a limited set of EEG channels above the ears,
4. the method is capable of discriminating between different angular speaker positions,
5. the method can be employed within a multi-condition or multiclass strategy to handle multiple speaker positions at the same time,
6. the method can, provided minor updates, be used in a subject-independent way, trading a minimum of performance for practical applicability.

We believe that these assets make the FB-CSP method an excellent candidate and a major step forward towards practical neuro-steered hearing devices.

Appendices

6.A Decoding the spatial focus during sustained attention

To investigate the AAD accuracy as a function of time during *sustained attention*, we use the LOSpO-CV of Section 6.4.2 on Dataset A, allowing to leave out full continuous recordings. It is important to verify whether decoding the spatial focus of attention is possible during the *full* duration of a continuous recording, while the subject sustains its attention towards a particular speaker/direction. If the AAD accuracy degrades over time, this means the FB-CSP method only exploits brain lateralization patterns when the subject initially focuses its attention, which has been reported in the context of α-power lateralization [100].

Figure 6.A shows the averaged performance over continuous trials and subjects as a function of time. As Dataset A contains 6-minute continuous recordings (here referred to as trials) of EEG with sustained attention, the AAD accuracy is shown per 1 s sliding decision window (no overlap) over these trials. The mean accuracy, over all decisions, 6-minute trials, and subjects, is equal to 80.0% and is the same as the accuracy on the 1 s-point in Figure 6.4a. Furthermore, there is no apparent decrease in performance over time, on the contrary, the accuracy seems to slightly increase in the first minute, whereafter the accuracy remains constant. This confirms that the FB-CSP method is capable of decoding the spatial focus of attention when the attention is sustained, furthermore, with a similar accuracy as when using random CV (Figure 6.3a).

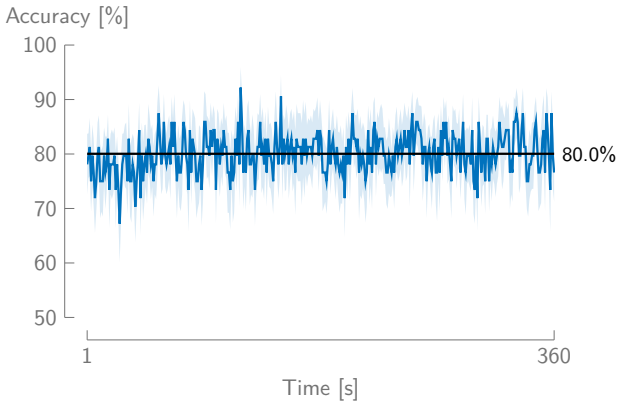


Figure 6.A: The performance (mean accuracy ± standard error of the mean over subjects and different 6-minute trials; Dataset A) does not degrade as a function of time when the attention is sustained. A sliding window of 1 s is used.

7 | Riemannian geometry-based decoding of the spatial focus of auditory attention

This chapter is based on S. Geirnaert, T. Francart, and A. Bertrand, "Riemannian Geometry-Based Decoding of the Directional Focus of Auditory Attention Using EEG," in Proceedings of the *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1115-1119, 2021. Three extra figures (Figures 7.1, 7.2b and 7.3) and an extra analysis on a multiclass problem have been added (Sections 7.2.3 and 7.3.5).

ABSTRACT | In Chapter 6, an alternative paradigm to the SR approach was proposed, in which the spatial focus of auditory attention is determined using CSP filtering, solely based on the EEG. In this chapter, we propose Riemannian geometry-based classification (RGC) as an alternative for this CSP approach, in which the covariance matrix of a new EEG window is directly classified while taking its Riemannian structure into account. While the proposed RGC method performs similarly to the CSP method for short decision lengths (i.e., the amount of EEG samples used to make a decision), we show that it significantly outperforms it for longer decision window lengths. Furthermore, we show how the RGC method is inherently more suited for multiclass problems, when multiple directions of auditory attention are combined. Therefore, the presented results imply that the RGC-based decoding of the spatial focus of attention is one of the best AAD methods to date.

7.1 Introduction

AAD algorithms traditionally use an SR approach (Chapter 3). This approach, however, suffers from low decoding accuracy at high speed, i.e., when using few data to decode the auditory attention (Part I). As these short decision windows (i.e., the amount of data used to decode the attention) $< 10\,\text{s}$ are paramount for the practical applicability of AAD algorithms, for example, when the attention is switched between two speakers (Chapter 2), the SR approach might be too slow for practical neuro-steered hearing devices or for conducting research experiments that require tracking of attention. Furthermore, this approach requires an error-prone speech separation step to retrieve the individual speech envelopes from the recorded mixture of speech sources (Section 1.7.2).

As an alternative paradigm, we proposed decoding the spatial focus of auditory attention in Chapter 6, solely based on the EEG (see also [136]). In this approach, the CSP filtering method is used to discriminate between different angular positions of the attended and unattended speaker(s). This CSP approach significantly outperforms the SR approach on short decision windows. Furthermore, this paradigm does not require a preceding speech separation step. As such, this alternative paradigm improves the practical applicability of neuro-steered hearing devices.

In this chapter, we propose a new AAD algorithm, capitalizing this new paradigm of decoding the spatial focus of auditory attention but replacing the traditional CSP filter method with a so-called *Riemannian geometry-based classifier* (RGC). Therefore, it can be classified into the 'nonlinear direct classification' branch of the tree of AAD algorithms in Figure 3.1. This technique has become very popular in the BCI community [38] and outperforms the classical CSP approach in various BCI applications, particularly in motor imagery paradigms [38, 179, 180]. In Section 7.2, we explain how this RGC can be used to classify the spatial focus of auditory attention. In Section 7.3, we compare the proposed RGC classifier with the state-of-the-art CSP method and SR approach. Conclusions are drawn in Section 7.4.

7.2 Riemannian geometry-based classification

In recent years, a new class of RGCs has gained a lot of attention in the BCI community [38]. Instead of prefiltering the EEG using data-driven filters based on the EEG covariance structure (as is the case in CSP filtering [170]), the EEG covariance matrices are classified directly, as it is assumed that all spatial (and potentially temporal) information concerning different conditions is encoded in

these covariance matrices [179,180]. However, covariance matrices are symmetric positive definite (SPD), such that they live on a differentiable Riemannian manifold rather than in a Euclidean space. RGCs take this specific structure into account to improve classification performance. Figure 7.1 summarizes the proposed classification pipeline, which is explained in the following sections. More details about RGCs and their use in BCIs can be found in [38,179–181].

7.2.1 The tangent space mapping

As covariance matrices live on a *differentiable* Riemannian manifold, a tangent space at each point (i.e., covariance matrix) can be computed. Such a tangent space, containing symmetric matrices, is Euclidean, where Euclidean distances between tangent vectors approximate Riemannian distances (i.e., distances between covariance matrices on the Riemannian manifold) of the (projected) covariance matrices. As traditional classifiers rely on Euclidean metrics, which conflict with the Riemannian structure of the manifold on which covariance matrices live, it is preferred to first project all covariance matrices onto the tangent space of a reference matrix (Figure 7.1). This is the crucial difference with a straightforward direct classification of covariance matrices, which assumes a Euclidean structure of the covariance matrices. In the RGC, the intermediate *tangent space mapping* (TSM) assures that Euclidean metrics are applicable (Figure 7.1). For the tangent space to be a good local approximation of the Riemannian manifold, where Euclidean distances between tangent vectors closely approximate Riemannian distances between the covariance matrices, a good choice of the reference point of the TSM is the geometric or Riemannian mean.

Let $\{\mathbf{X}_k, y_k\}_{k=1}^{K}$ be a training set containing K segments of bandpass filtered zero-mean EEG data $\mathbf{X}_k \in \mathbb{R}^{C\times\tau}$, with C channels and τ time samples, and with known labels $y_k \in \{-1, 1\}$ (for example, attending to the left or right speaker). The corresponding (sample) covariance matrices are defined as

$$\mathbf{R}_k = \frac{1}{\tau - 1}\mathbf{X}_k\mathbf{X}_k^{\mathrm{T}} \in \mathbb{R}^{C\times C}. \tag{7.1}$$

As in Chapter 6 and Part II, we estimate the covariance matrices using ridge regression, where the regularization hyperparameter is determined automatically using the method proposed in [164, 165]. This hyperparameter estimation method is considered to be the state of the art in BCI research [38].

The geometric or Riemannian mean of these K covariance matrices is then given by the SPD matrix $\mathbf{R}_{\mathfrak{G}}$ that minimizes the mean squared Riemannian

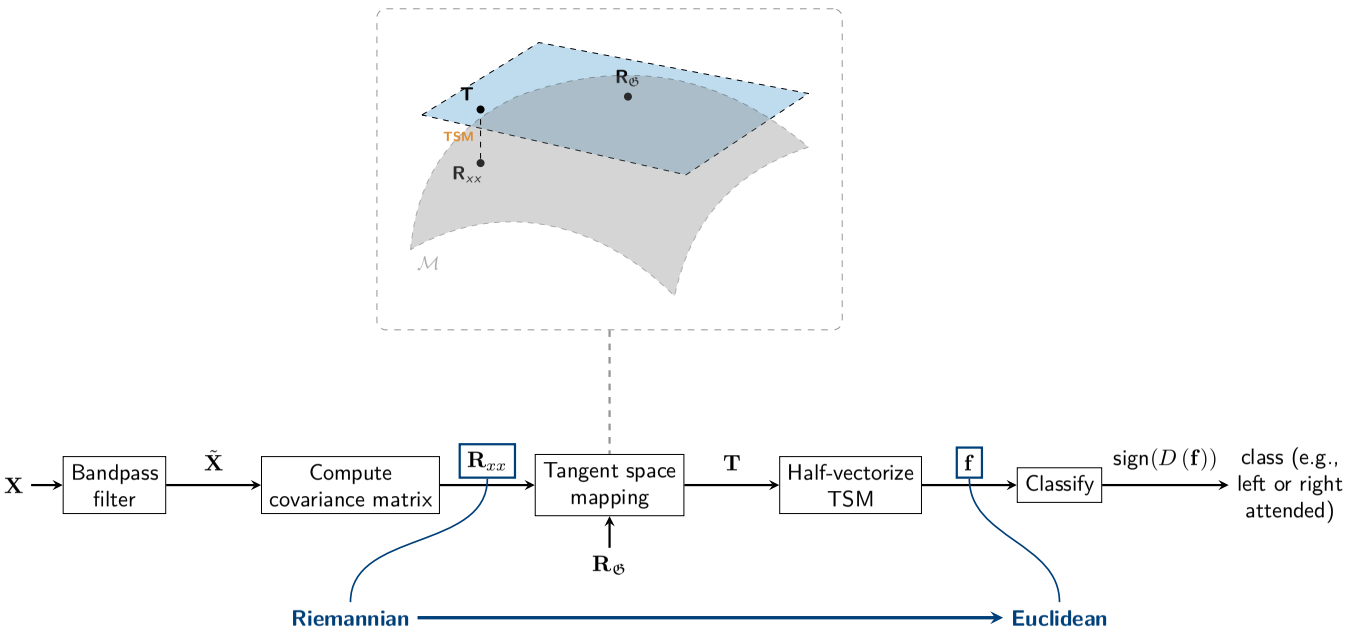


Figure 7.1: The sample covariance matrix is mapped onto the tangent space of Riemannian mean of the training set, after which it is classified. The TSM 'converts' the Riemannian structure of the covariance matrix to a Euclidean one by projecting the covariance matrix, which lies on the differentiable Riemannian manifold $\mathcal{M}$, onto the tangent space at reference point $\mathbf{R}_{\mathfrak{G}}$ (subfigure based on [179]). This assures that traditional classifiers are applicable.

distance from each $\mathbf{R}_k$ to $\mathbf{R}_{\mathfrak{G}}$ [179]:

$$\mathbf{R}_{\mathfrak{G}} = \mathfrak{G}(\mathbf{R}_1, \ldots, \mathbf{R}_K) = \underset{\mathbf{R} \text{ is SPD}}{\operatorname{argmin}} \sum_{k=1}^{K} \delta_R^2(\mathbf{R}_k, \mathbf{R}), \tag{7.2}$$

where $\delta_R(\mathbf{R}, \mathbf{S})$ denotes the Riemannian distance between two SPD matrices $\mathbf{R}$ and $\mathbf{S}$, which can be computed as [181]:

$$\delta_R(\mathbf{R}, \mathbf{S}) = \left|\left|\log\left(\mathbf{R}^{-\frac{1}{2}}\mathbf{S}\mathbf{R}^{-\frac{1}{2}}\right)\right|\right|_F, \tag{7.3}$$

with $\log(\cdot)$ the matrix-logarithm. Given a diagonalizable matrix $\mathbf{A} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^{-1}$, the matrix-logarithm of $\mathbf{A}$ is defined as:

$$\log(\mathbf{A}) = \mathbf{V}\log(\mathbf{\Lambda})\mathbf{V}^{-1}, \tag{7.4}$$

with $\log(\mathbf{\Lambda})$ a diagonal matrix with diagonal elements $\log(\lambda_i)$. The Riemannian mean in (7.2) can only be computed in an iterative way, by iteratively computing the Euclidean mean in the TSM, or can be approximated using log-euclidean metrics [182]:

$$\mathbf{R}_{\mathfrak{G}} \approx \exp\left(\frac{1}{K}\sum_{k=1}^{K}\log(\mathbf{R}_k)\right), \tag{7.5}$$

where the matrix-exponential $\exp(\cdot)$ is defined similarly as the matrix-logarithm in (7.4). We here use the latter estimation method in (7.5) to efficiently compute the Riemannian mean covariance matrix.

The normalized TSM of the covariance matrix $\mathbf{R}_k$ onto the tangent space at reference point $\mathbf{R}_{\mathfrak{G}}$ (7.2) is then equal to [179]:

$$\mathbf{T}_k = \log\left(\mathbf{R}_{\mathfrak{G}}^{-\frac{1}{2}}\mathbf{R}_k\mathbf{R}_{\mathfrak{G}}^{-\frac{1}{2}}\right). \tag{7.6}$$

7.2.2 Riemannian geometry-based classification

The TSM $\mathbf{T}_k$ can now be half-vectorized (i.e., a vectorization over the lower-triangular part only, as it is a symmetric matrix), which leads to the feature vector $\mathbf{f}_k \in \mathbb{R}^{\frac{C(C+1)}{2}\times 1}$, representing EEG segment $\mathbf{X}_k$ of the training set. Similarly, for a new test window $\mathbf{X}^{(\text{test})}$, the test feature vector can be found by computing the TSM of its covariance matrix using the Riemannian mean $\mathbf{R}_{\mathfrak{G}}$ over the training set.

The generated feature vectors with the aforementioned method can then be classified using any (Euclidean) classifier, trained with the training set

$\{\mathbf{f}_k, y_k\}_{k=1}^{K}$. We here choose an SVM classifier with a linear kernel. Such a classifier works well in high-dimensional feature spaces, which we are dealing with here. Note that combining the TSM with a linear SVM can be interpreted as applying an SVM with a *Riemannian kernel* on the half-vectorized original covariance matrix [180]. The classification algorithm is summarized in Algorithm 3.

Algorithm 3: Riemannian geometry-based classification

Input: Test EEG window $\mathbf{X}^{(\text{test})} \in \mathbb{R}^{C\times\tau}$ and given Riemannian mean $\mathbf{R}_{\mathfrak{G}}$ over a training set and (linear) SVM classifier $D(\cdot)$
Output: Class label $y^{(\text{test})}$ (e.g., left or right attended)

1: Bandpass filter $\mathbf{X}^{(\text{test})}$ between 12–30 Hz
2: Compute a regularized covariance matrix:

$$\mathbf{R}^{(\text{test})} = \frac{1}{\tau - 1}\mathbf{X}^{(\text{test})}\mathbf{X}^{(\text{test})^{\mathrm{T}}} + \delta\mathbf{I},$$

with regularization constant δ computed as in [164, 165]
3: Compute the TSM onto Riemannian mean $\mathbf{R}_{\mathfrak{G}}$:

$$\mathbf{T}^{(\text{test})} = \log\left(\mathbf{R}_{\mathfrak{G}}^{-\frac{1}{2}}\mathbf{R}^{(\text{test})}\mathbf{R}_{\mathfrak{G}}^{-\frac{1}{2}}\right)$$

4: Compute the feature vector as the half-vectorization $\mathbf{f}^{(\text{test})} = \text{vech}\left(\mathbf{T}^{(\text{test})}\right)$ of the TSM
5: Classify: $y^{(\text{test})} = \text{sign}(D(\mathbf{f}))$

7.2.3 Multiclass RGC

In Section 6.2.3, we explained how the CSP method can be extended to a multiclass scenario, i.e., when more than two directions of auditory attention are possible (see, for example, Figure 6.2). In this method, a coding scheme (i.e., one-vs-all) is applied *both* to the FB-CSP filter and LDA classifier design. More specifically, given M different directions/classes, per direction m, the GEVD of the matrix pencil $\left(\mathbf{R}_{\mathcal{C}_m}, \sum_{\substack{i=1\\ i\neq m}}^{M} \mathbf{R}_{\mathcal{C}_i}\right)$ needs to be solved, with $\mathbf{R}_{\mathcal{C}_m}$ the sample covariance matrix across all $\mathbf{X}_k$'s belonging to class $\mathcal{C}_m$. Afterwards, an LDA classifier is trained similarly, resulting in M different CSP+LDA pairs.

The proposed RGC method is, however, inherently more suited to extend to a multiclass scenario. While the feature extraction step in the CSP method (i.e., the log-energies of the outputs of the CSP filtering (Figure 6.1)) is also subject to the multiclass one-vs-all coding scheme, this is *not* the case for the RGC method. The feature extraction step in the latter (i.e., the TSM and half-vectorization in Algorithm 3) is decoupled from the multiclass nature of the problem (with the reference matrix $\mathbf{R}_{\mathfrak{G}}$ the Riemannian mean across all covariance matrices $\mathbf{R}_k$ from all classes). In other words, the multiclass problem only affects the last classification step in Algorithm 3 (Step 4), where a coding scheme is applied to the SVM classifier similarly to the multiclass CSP method. Here, we adopt a one-vs-all coding scheme for the SVM classifier.

7.3 Experiments and results

We compare the proposed RGC method with the CSP method (Chapter 6), which is the state-of-the-art method for decoding the spatial focus of auditory attention. In this CSP method, features are generated by applying six spatial filters that maximize discriminability [170] and are classified with an LDA classifier. The state-of-the-art SR method (CCA + LDA), as shown in Chapter 3, is also added as a reference in Section 7.3.4. For the CCA method, the same preprocessing steps and design choices as in Chapter 6 are used.

7.3.1 AAD datasets

The proposed RGC method is compared with the CSP (and CCA) methods on two different datasets, similarly to Chapter 6. The first, publicly available Dataset A contains the EEG of 16 subjects, attending to one of two simultaneously active competing speakers, located at $\pm 90°$ along the azimuth direction. Per subject, 72 minutes of data is available. This dataset is used in Section 7.3.4.

The second dataset (Dataset D) contains the EEG of 18 subjects attending to one of two competing speakers located at different angular positions ($\pm 90°, +30/+90°, -90/-30°$, and $\pm 5°$), with babble background noise at different SNRs. Similarly to Chapter 6, we use this dataset to compare the RGC and CSP methods in a multiclass scenario (Section 7.3.5). The EEG in both datasets is recorded using a $C = 64$-channel BioSemi ActiveTwo system.

7.3.2 Design choices

According to the analysis of the filterband importance in the state-of-the-art CSP approach, the β-band (12–30 Hz) is the most useful EEG frequency band to decode the spatial focus of attention (Section 6.4.8). As such, both for the baseline CSP algorithm, as for the proposed RGC method, the EEG is prefiltered in the β-band using an 8^{th}-order Butterworth filter and downsampled to 64 Hz (unless mentioned otherwise).

For the CSP method, the other design choices are the same as in Section 6.3.2.

7.3.3 Performance evaluation

The proposed RGC method is tested in a subject-specific way using ten-fold CV. Therefore, the 72 (Dataset A)/138 (Dataset D) minutes of EEG data per subject are split into 60 s (Dataset A)/30 s (Dataset D) segments, which are randomly distributed across ten folds. Note that these 60/30 s segments are normalized by setting the mean per channel to zero, as well as setting the Frobenius norm across all channels to one. The decision window length is defined as the length of the EEG window over which a single AAD decision is made (this usually results in a trade-off between AAD accuracy and decision latency (Chapter 2)). In the case of our RGC framework, the decision window length is defined by the number of samples τ over which the covariance matrices are estimated. To evaluate the AAD accuracy for various decision window lengths, all 60/30 s segments are split into shorter decision windows. The Riemannian mean in (7.2) and linear SVM are retrained for every decision window length. The significance level for above-chance AAD accuracy is computed based on the inverse binomial distribution [74]. Note that shorter decision window lengths result in more decisions over the test fold, resulting in a lower significance level. A similar ten-fold CV procedure is used for the CSP and CCA method.

Evaluating the AAD accuracy across different decision window lengths is important, for example, in the context of detecting switches in auditory attention. To resolve the traditional trade-off between accuracy and decision window length, the MESD metric [s] is used, as proposed in Chapter 2. This single-number AAD performance metric quantifies the minimal expected time it takes to switch the gain from one speaker to another, following a switch in attention, based on an optimized stochastic model of a robust (i.e., assuring stable operation above a pre-defined comfort level) attention-steered gain control system.

7.3.4 Binary RGC

Figure 7.2a shows the AAD accuracies as a function of decision window length for the RGC, CSP, and CCA method on the binary left/right classification problem on Dataset A. Below 1 s decision window lengths (i.e., using $\tau = 64$ samples at $f_s = 64$ Hz), the RGC and CSP methods have very similar accuracies. Between 1 s and 5 s, there is a much faster increase in performance for the RGC method than for the CSP method. This is mostly due to the quickly improving covariance matrix estimation (required for the RGC method) at these shorter decision window lengths. Indeed, as more data become available for increasing decision window lengths to estimate the covariance matrix, less regularization is required, introducing a smaller bias on the estimated covariance matrix. There is no similar effect for the CSP method, as there is no direct covariance matrix estimation involved. The RGC method could potentially be improved on these very short decision windows by applying an intelligent dimensionality reduction or feature selection method (see Section 8.2). Beyond 5 s, the performance levels off in both cases, and the RGC method outperforms the CSP method with $\approx 6\%$.

As is also shown in Chapter 6, the traditional SR method (CCA) outperforms the CSP method for the - less practical - long decision windows > 20 s. As the RGC method outperforms the CSP method on almost all decision window lengths, the region in which the CCA method is the best has decreased to the range > 40 s. If one would construct an AAD algorithm combining both approaches (RGC + CCA), the envelope would largely, and in the most important regions, be dominated by the RGC method.

The per-subject MESD values (Figure 7.2b) are all < 5 s (except for two outliers due to poorer performing subjects with MESDs > 5 s, but < 24 s), with median MESD $= 2.26$ s and $[25, 75]$%-quantiles $= [2.13, 2.62]$s. Note that the MESD values of the CCA method are all above 5 s (due to poor performance at short decision windows, median MESD $= 16.07$ s). The median MESD of the CSP method is $= 2.34$ s, with $[25, 75]$%-quantiles $= [2.12, 2.61]$s. For the CSP and RGC method, as there are still relatively high accuracies on the very short decision windows, the optimal trade-off point between AAD accuracy and decision window length is very often located at the shortest decision window lengths. As both methods have very similar accuracies there (Figure 7.2a), the MESD values are also very similar across both methods, with similar median values. Furthermore, a paired Wilcoxon signed-rank test ($n = 16, p = 0.0627$) shows no significant difference between both methods.

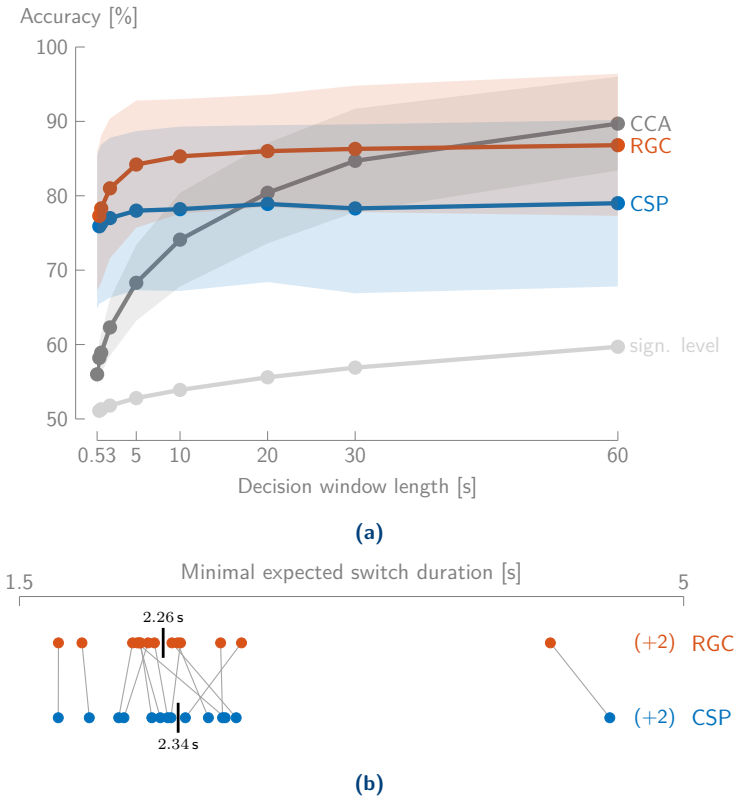


Figure 7.2: **(a)** The mean AAD accuracy across subjects (± standard deviation across subjects; Dataset A) shows that the RGC method outperforms the CSP approach on almost all decision window lengths but exhibits a faster decrease in performance on very short decision window lengths, resulting in very similar performances on 531 ms decision windows. **(b)** The per-subject MESD values (each dot = one subject) are very similar across the CSP and RGC methods, as they are mostly taken on the shortest decision window lengths. CCA is omitted, as all MESD values for this method are above 5 s. The median MESD across subjects is indicated with a bar, and the gray lines connect the same subjects across methods. Outlying subjects are indicated by (+x).

7.3.5 Multiclass RGC: classifying between left/frontal/right spatial focus

We now compare the multiclass RGC method discussed in Section 7.2.3 with the (FB-)CSP method on Dataset D in a similar experiment as in Section 6.4.4. The azimuth plane is divided into three zones that correspond to the three to-be-classified classes: left, frontal, or right attended (Figure 6.2). In all methods, a one-vs-all coding scheme is used.

Figure 7.3 shows the performances curves of the multiclass RGC method, the multiclass CSP method only using the β-band (as for the RGC method), and the multiclass FB-CSP method as proposed in Section 6.4.4. The FB-CSP method is also shown here (with the same performance as in Figure 6.6), as it gives a significant improvement over the CSP method using only the β-band. The trend of the RGC method is similar to the one in Section 7.3.4: it clearly outperforms the (FB-)CSP method on the longer decision window lengths, achieving up to 92% accuracy, but has a decreasing performance on very short decision windows. Beyond 5 s, the RGC method outperforms the FB-CSP method with $\approx 10\%$ and the CSP method using only the β-band with $\approx 16\%$, and has a smaller standard deviation. As expected, this difference is even larger than in Section 7.3.4, given that the RGC method is inherently more suited for a multiclass scenario than the CSP method(s) (Section 7.2.3). The median MESD, over all angular domains, is 4.42 s (RGC), 4.58 s (CSP with β-band), and 4.32 s (FB-CSP). While a paired Wilcoxon signed-rank test ($n = 18, p = 0.27$) shows no significant difference based on the MESD between the RGC and FB-CSP method, there is a significant difference ($n = 18, p = 0.0016$) with the CSP method only using the β-band (i.e., the same frequency information as the RGC method).

7.4 Discussions and conclusion

We have shown that the proposed RGC is capable of outperforming the state-of-the-art CSP method to decode the spatial focus of auditory attention by 6% on most decision window lengths on a binary classification task and even by 10% to 16% on a multiclass problem. However, two limitations are to be noted. Firstly, the RGC method performs similarly to the CSP method on very short decision windows (Figures 7.2 and 7.3), due to the worse covariance matrix estimation on small sample sizes. As the MESD values indicate that these very short decision windows are most relevant in the context of attention switching, the RGC method achieves a similar overall MESD as the CSP method. Furthermore, this RGC method has, due to the TSM in (7.6), a higher computational load than

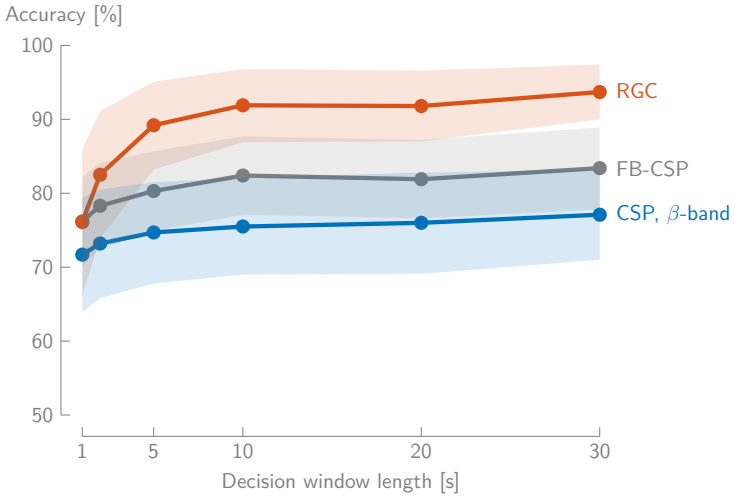


Figure 7.3: The RGC method outperforms the CSP (with β-band only) and FB-CSP methods even more on the multiclass problem, i.e., when classifying attention to one out of three angular domains (mean AAD accuracy $\pm$ standard deviation across subjects; Dataset D).

applying a simple spatial filter. Both limitations need to be considered for the real-time AAD application in neuro-steered hearing devices.

To conclude, the large increase in AAD accuracy compared to the state-of-the-art CSP method makes the proposed method a good candidate to decode the auditory attention, given that it also outperforms the SR (i.e., CCA) approach for decision window lengths below 40 s. This makes the RGC-based decoding of the spatial focus of auditory attention one of the best AAD methods to date.

Part IV

Conclusion

8 | Conclusion

8.1 Main findings and implications

In this section, we summarize the main findings and implications of the presented work per part of the thesis.

8.1.1 Part I: Evaluation and comparison of AAD algorithms

In Chapter 2, we have developed a novel performance metric for AAD based on the notion of a stable switch in an attention-steered adaptive gain control system, following a switch in auditory attention of the user. We have modeled this gain control system as a Markov chain, where a stable switch corresponds to a gain switch to a predefined comfort level. This Markov chain is then optimized to provide a comfortable listening experience and a minimal gain switch duration. While this mathematical model implicitly assumes, for example, independence of AAD decisions (which is not the case in practice, especially not for short decision windows) and is not based on actual switches in attention, it results in an elegant and easy-to-compute AAD performance metric that quantifies the average duration of a stable gain switch - the MESD. As such, it provides a tool that enables easy and statistical evaluation and comparison of AAD algorithms across different decision window lengths based on a relevant and interpretable criterion for AAD. The MESD performance metric has shown to be a crucial tool in Chapter 3 and has been used throughout the thesis to compare AAD algorithms. As such, the MESD realizes one of the first main research objectives outlined in Section 1.9.1.

Apart from providing a performance metric, the (development of the) MESD metric gives a few additional insights and by-products. First, it underlines the importance of evaluating a new AAD algorithm at different decision window lengths. While SR-based algorithms typically exhibit a speed-accuracy trade-off, this is much less pronounced in algorithms that decode the spatial focus of attention, as shown in Part III. The analysis of an algorithm at different decision window lengths is, therefore, crucial to identify the interesting operating regions of an algorithm. For example, while the traditional SR algorithm is less interesting to provide online, fast AAD decisions (as evidenced by the MESD), it could be interesting to provide decisions on a slower time scale (see Section 8.2). Related to this analysis, the MESD stresses the importance of short decision windows (i.e., making fast decisions) to detect switches in attention. While the MESD metric is not based on actual attention switches, the fast detection of such a switch always needs to be kept in mind when designing a new AAD algorithm. It is, for example, relatively easy to design an algorithm that has a very high accuracy on short decision windows by implementing some post-processing smoothing procedure across AAD decisions[1] to correct for errors. However, such an algorithm would result in substantial detection delays of AAD switches and would thus seriously hamper the practical applicability of such an algorithm. For every AAD algorithm, it is thus important to disentangle the decision window length as used in the AAD algorithm from the *effective* decision window length that determines the temporal resolution of the AAD decisions, as well as to determine the expected decoding delay on a switch in attention. Lastly, as a by-product, the MESD results in an optimal gain control system (i.e., an optimal number of states and working point of the algorithm) with only a few tunable parameters such as the comfort and confidence level. Such a system could be used as a basis for a practical gain control system (see Section 8.2).

Based on the MESD performance metric, we have been able to establish in Chapter 3 that CCA was the best AAD algorithm (at least up to the year 2020), not only because it outperforms other linear SR-based algorithms but, moreover, because it proved to be stable across different independent datasets. However, this algorithm is supervised and fixed in time, necessitating the development of time-adaptive unsupervised decoders, which we have pursued in Part II. Furthermore, the MESD also indicates that even the CCA algorithm is too slow to detect attention switches (i.e., a delay around 15 s) in a practical neuro-steered hearing device. This implies that faster and more accurate AAD algorithms are required. Decoding the spatial focus of auditory attention (using a CNN) showed to be very promising towards these most important short decision window lengths. A third conclusion from this study is that DNN-based

[1] This would, furthermore, gravely undermine the independence assumption in the MESD metric.

methods are hard to replicate on different AAD datasets. A possible explanation is the limited size of the benchmark datasets in relation to the complexity of these systems but also the possible overtuning and overfitting to a particular dataset. These DNN methods are even more capable of (unwanted) latching onto experimental conditions, speech stimuli, equipment, etc. However, these (novel) DNN methods for AAD should not be discarded, given the nonlinear processing and additional complexity they can provide[2]. Therefore, we propose a few guidelines to take into consideration when evaluating AAD algorithms (in general):

- Use a large enough (training) dataset. While the required size of a training set depends on the complexity of the trained model, current AAD datasets might be too small for the design of, for example, DNN models. Furthermore, 'size' should not only be interpreted quantitatively but also qualitatively: enough variation (for example, concerning artifacts) should be present.
- A strict and proper CV scheme needs to be used. A random CV needs to be avoided for subject-specific AAD algorithms, especially for DNN models. This is crucial to avoid a too large similarity between the training and test set and thus to assure generalization to new data. The training and test sets need to be time-independent, potentially with extended breaks in between. Validation in a subject-independent manner is an alternative way to verify that (cross-subject) generalization is possible. The latter would assure that within-subject generalization across different recording sessions is also possible [184].
- The AAD algorithm needs to be evaluated on independent datasets, preferably with differences in setup, protocol, etc. One dataset could be used to design the AAD algorithm (for example, the DNN architecture), a second one to afterwards test the algorithm with minimal changes in the parameter settings. This is important to exclude overfitting (of DNN architectures) on specific datasets and, therefore, to assure generalizability to new data from different users in potentially different setups.

A more general overview of different methodological considerations when (linearly) modeling neural responses to speech can be found in [185]. The proper evaluation of AAD algorithms for more complex DNN methods should, of course, be viewed in the more general trend towards the proper design and evaluation of DNNs.

[2]Some argue, however, that 'complexity' of models is a more relevant way of thinking about neural en-/decoding models than linear versus nonlinear models, also, vis-à-vis the (research) goals [183].

8.1.2 Part II: Unsupervised AAD

In Chapter 4, we have developed a procedure for unsupervised training of a stimulus decoder for AAD based on a batch of unlabeled training data by iteratively updating the decoder based on its own predictions. This iterative relabeling resulted in a self-leveraging effect, which we have explained by developing a mathematical model for the updating procedure. This allows interpreting the developed algorithm as a fixed-point iteration, converging after approximately five iterations. Furthermore, the mathematical model allowed to accurately predict the accuracy of the resulting unsupervised decoder from the supervised decoder.

The resulting stimulus decoder from the unsupervised training procedure, starting from a random initial decoder, outperforms a subject-independent decoder trained on EEG data from other subjects (which thus also does not require attention labels on the test subject). Furthermore, we have shown how the unsupervised subject-specific updating could be combined with subject-independent information in a transfer learning approach, resulting in a decoder that even approximates the performance of a supervised subject-specific decoder. This batch training procedure then laid the foundation of the time-adaptive extension that we have developed in Chapter 5.

In this time-adaptive approach, we assume that EEG and audio data segments are continuously streaming in as in a practical neuro-steered hearing device. We have compared a sliding window implementation, which mimics the batch approach in Chapter 4, with a recursive one and have shown that this simple and efficient single-shot recursive updating scheme is most optimal. From the results, it was clear that this approach has the potential to outperform a static supervised decoder when conditions and situations (such as electrodes that are disconnected) change. This unsupervised time-adaptive AAD algorithm could in principle be employed in a plug-and-play fashion, pre-implemented on an off-the-shelf hearing device. This AAD algorithm would then on-the-fly train itself from scratch during operation and would be tuned to the end-user after 20 min, after which it would continuously update. While the low accuracy of a stimulus decoder on short decision windows reduces the potentially high impact of this unsupervised time-adaptive algorithm in making AAD decisions, it can still be used to provide labels for another algorithm and to enable neurofeedback effects (see Section 8.2). As such, we consider this algorithm a crucial step forward in the online application of AAD in neuro-steered hearing devices, providing a solution for one of the main signal processing-related challenges outlined in the research objectives in Section 1.9.1.

8.1.3 Part III: Decoding the spatial focus of auditory attention

As established in [136] and Chapter 3, decoding the spatial focus of auditory attention is very promising due to its ability to achieve high accuracies at very short decision window lengths. In Part III, we have built upon this alternative AAD paradigm with a computationally efficient linear CSP algorithm (Chapter 6) and its nonlinear RGC extension (Chapter 7), which both greatly outperformed the CCA SR algorithm at (very) short decision times. Moreover, the RGC extension provided an additional significant improvement for the longer decision window lengths, especially for a three-class scenario with multiple possible angular directions. The CSP method performs at least as well as the CNN method in [136] but requires less training data and computational resources, making it the preferred choice according to Occam's razor. We have, furthermore, evaluated the CSP method in various scenarios, i.e., with a reduced amount of EEG channels around the ear, in a subject-independent context using a subject-specific but unsupervised bias update, with different speaker positions and babble noise levels, and in a three-class scenario, classifying auditory attention to a left, frontal, or right angular domain, or only detecting whether the left- or right-most speaker is attended. All these experiments underline the practical applicability of the CSP algorithm. Lastly, these methods do not require the original speech signals of the competing speakers for AAD itself, such that they are not prone to errors in the speaker separation step.

We have also tried to get a grasp on the decoding mechanisms of CSP-based decoding of the spatial focus of attention. Our experiments have shown that mainly the fronto-temporal β-band activity is the driving force. Furthermore, we have tried to rule out the possibility that eye- or ear-activity confounds are driving the decoding. While we have several arguments based on the experimental design, artifact removal, and channel selection, it is, however, impossible to completely rule out these confounds (see Section 8.2).

The very low MESDs resulting from these methods show that the CSP and RGC approaches are excellent candidates as fast and accurate decision-making AAD algorithms that allow fast switching between speakers, potentially resolving the second main signal processing-related challenge discussed in Section 1.9.1. Therefore, they hold great potential to be implemented in practical neuro-steered hearing devices. To be able to realize this potential, several next steps need to be taken, as outlined in Section 8.2.

8.2 Future directions

In this thesis, we have made several contributions to the different signal processing-related challenges for AAD in neuro-steered hearing devices, especially towards time-adaptive, unsupervised, fast, and accurate AAD algorithms. Nevertheless, several additional steps need to be taken to realize practical neuro-steered hearing devices. In this section, we discuss possible future research directions starting from the work in this PhD thesis.

8.2.1 Ecological validity

As discussed in Section 1.7.4, AAD algorithms need to be validated in realistic listening scenarios that are encountered in the real world (ecological validity) and on the targeted end-user: the hearing-impaired listener. Therefore, we propose the following next steps.

MESD validation on actual switches in attention

As explained in Chapter 2, the MESD performance metric is based on a theoretical model of an adaptive gain control system, in which the switch duration can be quantified. Therefore, the MESD is a theoretical metric that is not based on actual switches in auditory attention. A next step is thus to validate the theoretical MESD based on data that contain actual (spontaneous) switches in auditory attention, especially when the working point of the AAD algorithm is situated at very short decision window lengths (as with the CSP and RGC methods) and the independence assumption in the Markov chain could be violated.

Validation of CSP method in various listening scenarios

Recent experiments show that the CSP method does not work on every dataset or subject. For example, on Dataset B, similarly to [136] in Chapter 3, the CSP method does not perform better than chance level. A similar effect is present on 50% of the 44 subjects of the AAD dataset in [129] (containing normal-hearing and hearing-impaired listeners, however, seemingly being unrelated to it). A possible explanation is that the datasets used in Chapter 6 contain male-male speakers, while the datasets on which the CSP method fails contain male-female speakers (see also Table 1.3). In the latter datasets, binaural, spatial cues could be less important to focus attention on one of the two competing

speakers, while in the former, spatial hearing becomes much more important to provide spatial release from masking. This could explain why the spatial focus of auditory attention is not decodable in the mentioned datasets. This hypothesis is supported by the work in [57–60, 62], suggesting that the effect of spatial release from masking becomes less important when the competing sound sources can be easily separated, for example, based on monaural cues such as differences in fundamental frequency. Furthermore, recent pilot experiments have shown there might be a significant time-dependency in the CSP decoding, i.e., good performances are only obtained when the algorithm is trained on data close to the test data (in time). This effect seems to be especially present when the amount of training data is low, pointing towards an overfitting effect. Therefore, several new experiments are required to further investigate the effects on the CSP decoding of potential eye- and ear-artifacts, different fundamental frequencies of the competing speech signals, the amount of training data, and the time-dependency of the test on training data. For the latter, several transfer learning and domain adaptation solutions have already been proposed and could be explored in this context [38, 175, 186–189].

Lastly, as similarly done for the SR algorithm (Section 1.7.4), decoding the spatial focus of auditory attention using the CSP or RGC method should be further evaluated with multiple competing speakers, with (spontaneous) switches in attention, different noise types, and in a hearing-impaired population. The outcomes of these experiments will provide essential information on how and when decoding of the spatial focus of attention can be used.

8.2.2 Integration in a neuro-steered hearing device

Adaptive gain control system based on the MESD

As explained in Chapter 2, the MESD performance metric also results in an optimally designed Markov chain as a model for an adaptive gain control system. Such a gain control system is paramount in the online application of an AAD algorithm in a neuro-steered hearing device. Therefore, the optimally designed Markov chain from the MESD could be used to initialize a user-specific adaptive gain control system. This would then require some gain interpolation between the different states and finding the optimal settings (comfort level, confidence level, and minimal number of states) for the end-user. Therefore, it could be interesting to investigate the desired parameter settings (such as the comfort level) and final gain control experience in a cohort of (potentially hearing-impaired) subjects. Furthermore, the rate of gain change can be controlled via the number of states in the Markov chain. Given the desired parameters, this

number of states could be adapted over time using some confidence estimation in the decision of the AAD algorithm.

Speaker separation and enhancement for CSP decoding

Similar to the SR algorithm in Section 1.7.2, the CSP (or RGC) algorithm should be evaluated in combination with the speaker separation and enhancement step before the adaptive gain change. Simultaneously, the limits in terms of spatial resolution with the CSP algorithm should be investigated: how many angular domains can the algorithm handle, and what is the most optimal decoding strategy? The decoded attended direction could then be used to steer a beamformer to the decoded location. Note that preserving binaural cues in the beamforming is crucial, as the CSP algorithm inherently exploits spatial hearing [190, 191].

Neurofeedback

Similar to other BCI applications, neurofeedback could further enhance the performance of AAD, as already indicated by the experiments in [3]. This essentially means that we also consider human learning as an essential cornerstone of neuro-steered hearing devices (Figure 1.6). To enable such neurofeedback effects, in which the user learns to control and regulate its own neural activity [192, 193], a real-time AAD system is required to close the loop. An excellent candidate to feed back the AAD decisions to the user would be the adaptive gain control system (based on the Markov chain).

The work in Part II, however, allows going beyond a 'one-sided' neurofeedback scheme, where only the user learns to control a (static) AAD decision system through its neural activity. By also allowing the AAD algorithm to automatically adapt to the non-stationary neural signals of the user (for example, as the result of neurofeedback learning), a potentially stronger effect could be achieved. However, to obtain the desired enhanced performance from the interplay between a learning human and adapting algorithm, the rate of learning in the time-adaptive algorithm should be tuned to the human learning [38, 194]. Lastly, closing the loop could not only enable neurofeedback effects from human learning but could also allow decoding error-related potentials, i.e., decoding whether an error in gain change has been perceived by the user (different from its intention) [38].

8.2.3 EEG miniaturization and wearability effects

Data-driven channel selection for CSP decoding

In Chapter 6, we took a recording system-based point of view to test the CSP method on a reduced set of EEG channels by selecting the electrodes closest to the ears, emulating a cEEGrid array. While further tests with mobile setups, for example, using the actual cEEGrid array and in natural environments [3, 124] are important to probe the practical applicability, also a data-driven channel selection approach can be adopted. In [195], we[3] have developed a group-sparse channel selection method specifically for generalized Rayleigh quotient/GEVD problems as used in the design of the CSP-based decoder in Chapter 6. Such a channel selection could not only be used to optimally position miniaturized EEG sensors but also to provide insight into the neural decoding process. Furthermore, the reduced number of channels could result in a necessary dimensionality reduction to improve RGC-based decoding (see below).

8.2.4 Fast and accurate AAD algorithms

Improved and extended RGC-based decoding of the spatial focus of attention

While the RGC-based decoding of the spatial focus of attention in Chapter 7 shows great promise as an AAD algorithm, it has not yet been as extensively validated as the CSP algorithm in Chapter 6. Therefore, this algorithm should also be evaluated in a more mobile context, when generalizing to unseen subjects, and with the extensions proposed in Section 8.2.1. We, however, do not expect surprising results, given its similarity to CSP decoding. Moreover, to fully exploit the potential of the RGC-based decoding, the performance on very short decision window lengths should be improved. Potential solutions are to apply advanced regularization techniques in the covariance matrix estimation, for example, through a smart channel selection that could be re-used from the channel selection method in [195], or feature selection methods to improve the SVM classification.

[3] S. Geirnaert is joint first author of this journal article [195]. It is not included in this thesis as it is considered to be mostly peripheral to the scope of the thesis.

8.2.5 Time-adaptive AAD algorithms

Unsupervised CSP/RGC decoding

As extensively shown throughout this thesis, the SR algorithm does not perform well enough on very short decision window lengths for attention switch detection in neuro-steered hearing devices. This reduces the relevance of the time-adaptive, unsupervised stimulus decoder from Part II as the 'decision-maker' in AAD. Therefore, it is crucial to investigate a similar time-adaptive, unsupervised CSP/RGC decoding. The most elegant solution is to perform a similar iterative self-adapting approach on the CSP and LDA decoders. It is, however, unsure that the same self-leveraging effect will be present, as indicated in [38]. Therefore, an alternative and potentially easier approach is to use the time-adaptive, unsupervised stimulus decoder from Part II to provide reliable labels to update the CSP algorithm. This updating could be performed on longer updating segments, as the decoder updating rate can be much slower than the AAD decision rate (also in light of, for example, neurofeedback effects). Given that the SR algorithm is still one of the best AAD algorithms on very long decision windows (> 40 s, see Chapter 7), this could result in the ultimate combination of the time-adaptive, unsupervised, slow but accurate stimulus decoder to provide labels in order to update the supervised, fast and accurate CSP/RGC decoding algorithm, which then acts as the 'decision-maker'.

8.3 Final thoughts

While Section 8.2 shows that several additional steps need to be taken towards the realization of practical neuro-steered hearing devices, in this thesis, we have contributed different important pieces of the puzzle of AAD. Building upon the advances made in this thesis, a neuro-steered hearing device as in Figure 1.6 could potentially be built, encompassing a plug-and-play, time-adaptive, unsupervised, fast, and accurate AAD algorithm, properly integrated with a low-latency speaker separation and enhancement algorithm, a (Markov chain-based) adaptive gain control system, and a wearable, miniaturized EEG system. Therefore, I am hopeful that we can create a smart HA for Herman that allows him to enjoy the Christmas family dinner once again and break the pattern of loneliness.

Appendices

A | Listening to chaos: could you control a hearing aid with your brain?

This chapter contains an article for the general public about the thesis topic of AAD for neuro-steered hearing devices, initially written in Dutch for the Flemish science magazine EOS. It has been adapted and translated to English for the Leuven.AI-stories blog and the BioVox newsletter.

As the number of people struggling with hearing impairment is on the rise, KU Leuven researchers are attempting to develop smart hearing aids that use brainwaves to help users home in on specific conversations, cutting through the noise in chaotic situations like a busy family dinner.

Christmas Eve. The 76-year-old Johan is having a family dinner with his children and grandchildren. Although he is very happy that everyone is together, he feels lonely.

Johan suffers from hearing loss. He wears a hearing aid that amplifies sounds. Unfortunately, his hearing aid doesn't know which person he wants to listen to and the ensuing auditory chaos causes Johan to switch off his hearing aid.

Since he's unable to follow any distinct conversations, his experience, though in a room full of family, is one of isolation. This is what author Helen Keller meant when she said that "blindness isolates people from things, while deafness isolates people from people."

'Cocktail party' chaos

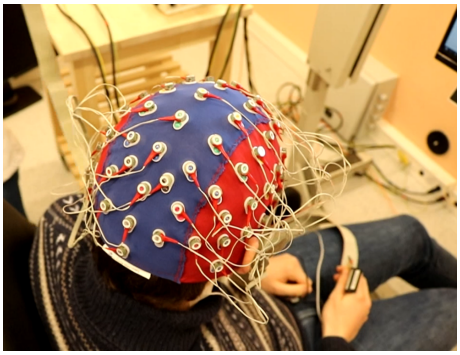
Like Johan, one in ten Belgians suffer from hearing loss. According to the World Health Organization, this number will increase in the coming decades partly due to the aging population, reaching one in four by 2050. Smart hearing aids that are able to listen in a targeted manner are therefore of paramount importance for the well-being of millions.

Unfortunately, current hearing aids do not work well in cocktail party situations, when several people are speaking at the same time. At receptions, parties, and dinner tables, the hearing aid does not know to which speaker the user wants to listen and simply amplifies all speakers equally, preventing the user from following any conversation. Perhaps you too – even without hearing loss – experience these difficulties in such situations?

Finding out who Johan wants to listen to

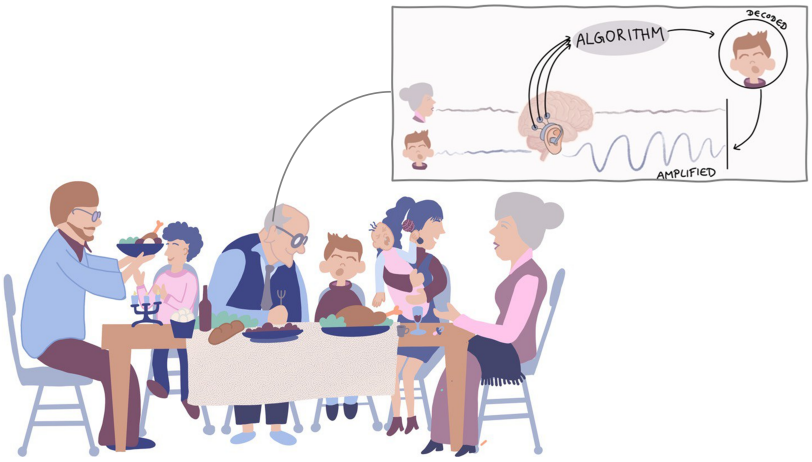
A possible solution to Johan's problem is to simply amplify the speaker closest to him or the person he's looking at. But if Johan wants to eavesdrop on what his partner at the other end of the table is saying about him (something he hasn't been able to do for years), this approach would fail.

What if you could read directly into Johan's brain who he wants to listen to? Only there can we find the correct information. This seemingly impossible feat is precisely what our interdisciplinary team at KU Leuven is working on. Measuring the electrical activity of the brain via an electroencephalogram (EEG), using sensors placed on the skull, we can use these data to find out who Johan wants to listen to.



We measure the electrical activity of the brain via the electroencephalogram.

Our team is designing algorithms to read the EEG and determine what a person wants to focus on. For example, you can use artificial intelligence to reconstruct features of the speech signal that a user pays attention to. By matching those reconstructed features from the EEG with all the speech signals that the hearing aid picks up, you can identify the correct speaker. The hearing aid can then suppress all other conversations and amplify the right conversation.



At KU Leuven, we are developing algorithms to find out who Johan wants to listen to from his brain waves. This way, we can amplify the right speaker.

Where's that sound coming from?

Problem solved? Not quite. The EEG can be compared to a blurry video (see also Can you read thoughts from EEG?). The relevant brain activity is buried under all kinds of other activity, with countless processes going on at any given moment. To find out what is happening on that blurry video, we need enough footage – at least 30 seconds. Unfortunately, that's too slow for Johan, who'll miss what his partner is saying right this minute.

Can you read thoughts from EEG? We reconstruct features of the speech signal to which someone is paying attention from the brain waves. Can we also find out what that person is saying? Isn't it even possible to decipher what you're thinking?

Luckily not. EEG is a non-invasive technique. This means that we measure the signals on the skull and not under the skull. As a result, there is a significant distance between the EEG sensors and the neurons – the basic cells of our brains –, which cause electrical activity by firing. In addition, the cerebrospinal fluid and the skull attenuate this activity. These processes are precisely what makes the EEG so 'blurred'. Therefore, I am convinced that it is fundamentally very difficult to reconstruct speech or language from EEG.

To overcome this delay, we have developed a new technique that based on the brain patterns determines the spatial direction of Johan's attention. Colleagues at Columbia University in New York have, for example, determined that different brain processes are active when you listen to the left or right. Using artificial intelligence to tease out these specific brain patterns, we can now determine very quickly - in less than two seconds – which direction someone is listening to. The hearing aid can then amplify the speaker at that specific location.

This new technology is very promising due to its high speed, but also raises many questions. For example, it is currently unclear exactly which brain processes are active when we listen to a specific direction. The answers to these questions will have a significant impact on the practical applicability of this innovative technique.

There's work to be done

There is still a lot of work to be done before this promising technology becomes a reality. Among other things, measuring the EEG with wearable sensors remains

a challenge. We are currently testing this technology in different scenarios and environments, and experiment with self-learning algorithms that automatically adapt to new situations.

In overcoming these hurdles, we hope to design a brain-controlled hearing aid to help improve the lives of people like Johan – to help people with hearing aids rediscover the joys of Christmas parties, and tune in on the conversations of their loved ones with ease.

Bibliography

[1] W. Biesmans, N. Das, T. Francart, and A. Bertrand, "Auditory-Inspired Speech Envelope Extraction Methods for Improved EEG-Based Auditory Attention Detection in a Cocktail Party Scenario," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 5, pp. 402–412, 2017.

[2] S. A. Fuglsang, T. Dau, and J. Hjortkjær, "Noise-robust cortical tracking of attended speech in real-world acoustic scenes," *NeuroImage*, vol. 156, pp. 435–444, 2017.

[3] R. Zink, S. Proesmans, A. Bertrand, S. Van Huffel, and M. De Vos, "Online detection of auditory attention with mobile EEG: closing the loop with neurofeedback," *bioRxiv*, 2017.

[4] N. Das, A. Bertrand, and T. Francart, "EEG-based auditory attention detection: boundary conditions for background noise and speaker positions," *Journal of Neural Engineering*, vol. 15, p. 066017, 2018.

[5] S. Geirnaert, S. Vandecappelle, E. Alickovic, A. de Cheveigné, E. C. Lalor, B. T. Meyer, S. Miran, T. Francart, and A. Bertrand, "Electroencephalography-Based Auditory Attention Decoding: Toward Neurosteered Hearing Devices," *IEEE Signal Processing Magazine*, vol. 38, no. 4, pp. 89–102, 2021.

[6] World Health Organization, "World Report on Hearing," Tech. Rep., 2021.

[7] E. C. Cherry, "Some Experiments on the Recognition of Speech, with One and with Two Ears," *The Journal of the Acoustical Society of America*, vol. 25, no. 5, pp. 975–979, 1953.

[8] S. Haykin and Z. Chen, "The Cocktail Party Problem," *Neural Computation*, vol. 17, no. 9, pp. 1875–1902, 2005.

[9] I. Sturm, S. Dähne, B. Blankertz, and G. Curio, "Multi-Variate EEG Analysis as a Novel Tool to Examine Brain Responses to Naturalistic Music Stimuli," *PLOS ONE*, vol. 10, no. 10, p. e0141281, 2015.

[10] G. Cantisani, G. Trégoat, S. Essid, and G. Richard, "MAD-EEG: an EEG dataset for decoding auditory attention to a target instrument in polyphonic music," in Proceedings of the *Speech, Music and Mind (SMM), Satellite Workshop of Interspeech 2019*, pp. 51–55, 2019.

[11] G. Cantisani, S. Essid, and G. Richard, "EEG-Based Decoding of Auditory Attention to a Target Instrument in Polyphonic Music," in Proceedings of the *2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pp. 80–84, 2019.

[12] G. M. Di Liberto, C. Pelofi, R. Bianco, P. Patel, A. D. Mehta, J. Herrero, A. de Cheveigné, S. A. Shamma, and N. Mesgarani, "Cortical encoding of melodic expectations in human temporal cortex," *eLife*, vol. 9, p. e51784, 2020.

[13] G. M. Di Liberto, G. Marion, and S. A. Shamma, "Accurate Decoding of Imagined and Heard Melodies," *Frontiers in Neuroscience*, vol. 15, no. 673401, 2021.

[14] N. J. Zuk, E. S. Teoh, and E. C. Lalor, "EEG-based classification of natural sounds reveals specialized responses to speech and music," *NeuroImage*, vol. 210, p. 116558, 2020.

[15] N. J. Zuk, J. W. Murphy, R. B. Reilly, and E. C. Lalor, "Envelope reconstruction of speech and music highlights stronger tracking of speech at low frequencies," *PLOS Computational Biology*, vol. 17, no. 9, pp. 1–32, 2021.

[16] O. Etard, R. B. Messaoud, G. Gaugain, and T. Reichenbach, "No Evidence of Attentional Modulation of the Neural Response to the Temporal Fine Structure of Continuous Musical Pieces," *Journal of Cognitive Neuroscience*, 2021.

[17] E. M. Kaya and M. Elhilali, "Modelling auditory attention," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 372, no. 1714, p. 20160101, 2017.

[18] F. A. Azevedo, L. R. Carvalho, L. T. Grinberg, J. M. Farfel, R. E. Ferretti, R. E. Leite, W. J. Filho, R. Lent, and S. Herculano-Houzel, "Equal Numbers of Neuronal and Nonneuronal Cells Make the Human Brain an Isometrically Scaled-Up Primate Brain," *Journal of Comparative Neurology*, vol. 513, no. 5, pp. 532–541, 2009.

[19] B. Pakkenberg, D. Pelvig, L. Marner, M. J. Bundgaard, H. J. G. Gundersen, J. R. Nyengaard, and L. Regeur, "Aging and the human neocortex," *Experimental Gerontology*, vol. 38, no. 1-2, pp. 95–99, 2003.

[20] Foglets.com, "Computation Power: Human Brain vs Supercomputer," 2019. [Online]. Available: https://foglets.com/supercomputer-vs-human-brain/

[21] J. Carlsmith, "How Much Computational Power Does It Take to Match the Human Brain?" 2020. [Online]. Available: https://www.openphilanthropy.org/brain-computation-report

[22] R. Zink, "Background: Recording neural activity of the human brain," in *Mobile EEG and tensor approaches for auditory attention analysis in real-life (PhD thesis)*, 2017, ch. 2, p. 10.

[23] "Created with BioRender.com." [Online]. Available: www.biorender.com

[24] E. K. S. Louis and L. C. Frey, *Electroencephalography (EEG): An Introductory Text and Atlas of Normal and Abnormal Findings in Adults, Children, and Infants*. Chicago: IL: American Epilepsy Society, 2016.

[25] T. Egner and J. H. Gruzelier, "EEG Biofeedback of low beta band components: frequency-specific effects on variables of attention and event-related brain potentials," *Clinical Neurophysiology*, vol. 115, no. 1, pp. 131–139, 2004.

[26] L. F. Nicolas-Alonso and J. Gomez-Gil, "Brain Computer Interfaces, a Review," *Sensors*, vol. 12, no. 2, pp. 1211–1279, 2012.

[27] J. W. Kam, S. Griffin, A. Shen, S. Patel, H. Hinrichs, H.-J. Heinze, L. Y. Deouell, and R. T. Knight, "Systematic comparison between a wireless EEG system with dry electrodes and a wired EEG system with wet electrodes," *NeuroImage*, vol. 184, pp. 119–129, 2019.

[28] D. Looney, P. Kidmose, C. Park, M. Ungstrup, M. L. Rank, and K. Rosenkranz, "The In-the-Ear Recording Concept," *IEEE Pulse*, vol. 3, no. 6, pp. 32–42, 2012.

[29] S. L. Kappel, M. L. Rank, H. O. Toft, M. Andersen, and P. Kidmose, "Dry-Contact Electrode Ear-EEG," *IEEE Transactions on Biomedical Engineering*, vol. 66, no. 1, pp. 150–158, 2019.

[30] S. Debener, R. Emkes, M. De Vos, and M. G. Bleichner, "Unobtrusive ambulatory EEG using a smartphone and flexible printed electrodes around the ear," *Scientific Reports*, vol. 5, no. 16743, 2015.

[31] S. Blum, R. Emkes, F. Minow, J. Anlauff, A. Finke, and S. Debener, "Flex-printed forehead EEG sensors (fEEGrid) for long-term EEG acquisition," *Journal of Neural Engineering*, vol. 17, no. 034003, 2020.

[32] J. A. Urigüen and B. Garcia-Zapirain, "EEG artifact removal - state-of-the-art and guidelines," *Journal of Neural Engineering*, vol. 12, no. 3, p. 031001, 2015.

[33] B. Somers, T. Francart, and A. Bertrand, "A generic EEG artifact removal algorithm based on the multi-channel Wiener filter," *Journal of Neural Engineering*, vol. 15, p. 036007, 2018.

[34] T.-P. Jung, S. Makeig, C. Humphries, T.-W. Lee, M. J. McKeown, V. Iragui, and T. J. Sejnowski, "Removing electroencephalographic artifacts by blind source separation," *Psychophysiology*, vol. 37, no. 2, pp. 163–178, 2000.

[35] W. De Clercq, A. Vergult, B. Vanrumste, W. Van Paesschen, and S. Van Huffel, "Canonical correlation analysis applied to remove muscle artifacts from the electroencephalogram," *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 12, pp. 2583–2587, 2006.

[36] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan, "Brain-computer interfaces for communication and control," *Clinical Neurophysiology*, vol. 113, no. 6, pp. 767–791, 2002.

[37] G. Pfurtscheller, G. R. Müller-Putz, R. Scherer, and C. Neuper, "Rehabilitation with Brain-Computer Interface Systems," *Computer*, vol. 41, no. 10, pp. 58–65, 2008.

[38] F. Lotte, L. Bougrain, A. Cichocki, M. Clerc, M. Congedo, A. Rakotomamonjy, and F. Yger, "A review of classification algorithms for EEG-based brain-computer interfaces: a 10 year update," *Journal of Neural Engineering*, vol. 15, no. 3, p. 031005, 2018.

[39] N. J. Hill, T. N. Lal, M. Schröder, T. Hinterberger, G. Widman, C. E. Elger, B. Schölkopf, and N. Birbaumer, "Classifying Event-Related Desynchronization in EEG, ECoG and MEG Signals," in *Lecture Notes in Computer Science*, 2006, vol. 4174, pp. 404–413.

[40] H. Berger, "Über das Elektrenkephalogramm des Menschen," *Archiv für Psychiatrie und Nervenkrankheiten*, vol. 87, pp. 527–570, 1929.

[41] N. Ding and J. Z. Simon, "Emergence of neural encoding of auditory objects while listening to competing speakers," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 109, no. 29, pp. 11 854–11 859, 2012.

[42] J. Z. Simon, "The encoding of auditory objects in auditory cortex: Insights from magnetoencephalography," *International Journal of Psychophysiology*, vol. 95, no. 2, pp. 184–190, 2015.

[43] S. Akram, A. Presacco, J. Z. Simon, S. A. Shamma, and B. Babadi, "Robust decoding of selective auditory attention from MEG in a competing-speaker environment via state-space modeling," *NeuroImage*, vol. 124, pp. 906–917, 2016.

[44] S. Miran, S. Akram, A. Sheikhattar, J. Z. Simon, T. Zhang, and B. Babadi, "Real-time tracking of selective auditory attention from M/EEG: A Bayesian filtering approach," *Frontiers in Neuroscience*, vol. 12, p. 262, 2018.

[45] J. Besle, C. A. Schevon, A. D. Mehta, P. Lakatos, R. R. Goodman, G. M. McKhann, R. G. Emerson, and C. E. Schroeder, "Tuning of the Human Neocortex to the Temporal Dynamics of Attended Events," *Journal of Neuroscience*, vol. 31, no. 9, pp. 3176–3185, 2011.

[46] N. Mesgarani and E. F. Chang, "Selective cortical representation of attended speaker in multi-talker speech perception," *Nature*, vol. 485, pp. 233–236, 2012.

[47] E. M. Zion Golumbic, N. Ding, S. Bickel, P. Lakatos, C. A. Schevon, G. M. McKhann, R. R. Goodman, R. G. Emerson, A. D. Mehta, J. Z. Simon, D. Poeppel, and C. E. Schroeder, "Mechanisms Underlying Selective Neuronal Tracking of Attended Speech at a "Cocktail Party"," *Neuron*, vol. 77, no. 5, pp. 980–991, 2013.

[48] K. Dijkstra, P. Brunner, A. Gunduz, W. Coon, A. Ritaccio, J. Farquhar, and G. Schalk, "Identifying the Attended Speaker Using Electrocorticographic (ECoG) Signals," *Brain-Computer Interfaces*, vol. 2, no. 4, pp. 161–173, 2015.

[49] J. A. O'Sullivan, Z. Chen, J. Herrero, G. M. McKhann, S. A. Sheth, A. D. Mehta, and N. Mesgarani, "Neural decoding of attentional selection in multi-speaker environments without access to clean sources," *Journal of Neural Engineering*, vol. 14, no. 056001, 2017.

[50] C. Han, J. A. O'Sullivan, Y. Luo, J. Herrero, A. D. Mehta, and N. Mesgarani, "Speaker-independent auditory attention decoding without access to clean speech sources," *Science Advances*, vol. 5, no. eaav6134, 2019.

[51] E. Boto, N. Holmes, J. Leggett, G. Roberts, V. Shah, S. S. Meyer, L. D. Muñoz, K. J. Mullinger, T. M. Tierney, S. Bestmann, G. R. Barnes,

R. Bowtell, and M. J. Brookes, "Moving magnetoencephalography towards real-world applications with a wearable system," *Nature*, vol. 555, no. 7698, pp. 657–661, mar 2018.

[52] B. Moore, *An Introduction to the Psychology of Hearing*, 6th ed. Leiden, The Netherlands: Brill, 2013.

[53] J. Blauert, *Spatial Hearing: The Psychophysics of Human Sound Localization*. MIT Press, 1996.

[54] BioRender.com, "Adapted from "Middle and Inner Ear Anatomy"," 2022. [Online]. Available: https://app.biorender.com/biorender-templates

[55] A. S. Bregman, *Auditory Scene Analysis: The Perceptual Organization of Sound*. Cambridge: MIT press, 1994.

[56] J. H. McDermott, "The cocktail party problem," *Current Biology*, vol. 19, no. 22, pp. 1024–1027, 2009.

[57] R. Y. Litovsky, "Spatial Release from Masking," *Acoustics Today*, vol. 8, no. 2, p. 18, 2012.

[58] W. Noble and S. Perrett, "Hearing speech against spatially separate competing speech versus competing noise," *Perception & Psychophysics*, vol. 64, no. 8, pp. 1325–1336, 2002.

[59] V. Best, E. Ozmeral, F. J. Gallun, K. Sen, and B. G. Shinn-Cunningham, "Spatial unmasking of birdsong in human listeners: Energetic and informational factors," *The Journal of the Acoustical Society of America*, vol. 118, no. 6, pp. 3766–3773, 2005.

[60] B. G. Shinn-Cunningham, "Influences of spatial cues on grouping and understanding sound," in Proceedings of the *Forum Acusticum Budapest 2005: 4th European Congress on Acoustics*, pp. 1539–1544, 2005.

[61] M. A. Bee and C. Micheyl, "The "Cocktail Party Problem": What Is It? How Can It Be Solved? And Why Should Animal Behaviorists Study It?" *Journal of Comparative Psychology*, vol. 122, no. 3, pp. 235–251, 2008.

[62] K. Allen, S. Carlile, and D. Alais, "Contributions of talker characteristics and spatial location to auditory streaming," *The Journal of the Acoustical Society of America*, vol. 123, no. 3, pp. 1562–1570, 2008.

[63] A. Davis, C. M. McMahon, K. M. Pichora-Fuller, S. Russ, F. Lin, B. O. Olusanya, S. Chadha, and K. L. Tremblay, "Aging and Hearing Health: The Life-Course Approach," *Gerontologist*, vol. 56, no. S2, pp. S256–S267, 2016.

[64] G. Cardon, J. Campbell, and A. Sharma, "Plasticity in the Developing Auditory Cortex: Evidence from Children with Sensorineural Hearing Loss and Auditory Neuropathy Spectrum Disorder," *Journal of the American Academy of Audiology*, vol. 23, no. 6, pp. 396–411, 2012.

[65] B. G. Shinn-Cunningham, ""I want to party, but my hearing aids won't let me!"," *Hearing Journal*, vol. 62, no. 2, pp. 10–13, 2009.

[66] J. S. Snyder, C. Alain, and T. W. Picton, "Effects of Attention on Neuroelectric Correlates of Auditory Stream Segregation," *Journal of Cognitive Neuroscience*, vol. 18, no. 1, pp. 1–13, 2006.

[67] S. J. Aiken and T. W. Picton, "Human Cortical Responses to the Speech Envelope," *Ear and Hearing*, vol. 29, no. 2, pp. 139–157, 2008.

[68] B. N. Pasley, S. V. David, N. Mesgarani, A. Flinker, S. A. Shamma, N. E. Crone, R. T. Knight, and E. F. Chang, "Reconstructing Speech from Human Auditory Cortex," *PLoS Biology*, vol. 10, no. 1, p. e1001251, 2012.

[69] N. Ding and J. Z. Simon, "Neural coding of continuous speech in auditory cortex during monaural and dichotic listening," *Journal of Neurophysiology*, vol. 107, no. 1, pp. 78–89, 2012.

[70] V. Viswanathan, H. M. Bharadwaj, and B. G. Shinn-Cunningham, "Electroencephalographic Signatures of the Neural Representation of Speech during Selective Attention," *eNeuro*, vol. 6, no. 5, pp. 1–14, 2019.

[71] S. R. Synigal, E. S. Teoh, and E. C. Lalor, "Including Measures of High Gamma Power Can Improve the Decoding of Natural Speech From EEG," *Frontiers in Human Neuroscience*, vol. 14, no. 130, 2020.

[72] R. V. Shannon, F.-G. Zeng, V. Kamath, J. Wygonski, and M. Ekelid, "Speech Recognition with Primarily Temporal Cues," *Science*, vol. 270, no. 5234, pp. 303–304, 1995.

[73] A. J. Power, J. J. Foxe, E.-J. Forde, R. B. Reilly, and E. C. Lalor, "At what time is the cocktail party? A late locus of selective attention to natural speech," *European Journal of Neuroscience*, vol. 35, pp. 1497–1503, 2012.

[74] J. A. O'Sullivan, A. J. Power, N. Mesgarani, S. Rajaram, J. J. Foxe, B. G. Shinn-Cunningham, M. Slaney, S. A. Shamma, and E. C. Lalor, "Attentional Selection in a Cocktail Party Environment Can Be Decoded from Single-Trial EEG," *Cerebral Cortex*, vol. 25, no. 7, pp. 1697–1706, 2014.

[75] C. Horton, R. Srinivasan, and M. D'Zmura, "Envelope responses in single-trial EEG indicate attended speaker in a "cocktail party"," *Journal of Neural Engineering*, vol. 11, no. 046015, 2014.

[76] J. Vanthornhout, L. Decruy, J. Wouters, J. Z. Simon, and T. Francart, "Speech Intelligibility Predicted from Neural Entrainment of the Speech Envelope," *Journal of the Association for Research in Otolaryngology*, vol. 19, no. 2, pp. 181–191, 2018.

[77] D. Lesenfants, J. Vanthornhout, E. Verschueren, L. Decruy, and T. Francart, "Predicting individual speech intelligibility from the cortical tracking of acoustic- and phonetic-level speech representations," *Hearing Research*, vol. 380, pp. 1–9, 2019.

[78] L. Decruy, J. Vanthornhout, and T. Francart, "Evidence for enhanced neural tracking of the speech envelope underlying age-related speech-in-noise difficulties," *Journal of Neurophysiology*, vol. 122, no. 2, pp. 601–615, 2019.

[79] E. Verschueren, B. Somers, and T. Francart, "Neural envelope tracking as a measure of speech understanding in cochlear implant users," *Hearing Research*, vol. 373, pp. 23–31, 2019.

[80] L. Decruy, J. Vanthornhout, and T. Francart, "Hearing impairment is associated with enhanced neural tracking of the speech envelope," *Hearing Research*, vol. 393, no. 107961, 2020.

[81] G. M. Di Liberto, J. A. O'Sullivan, and E. C. Lalor, "Low-Frequency Cortical Entrainment to Speech Reflects Phoneme-Level Processing," *Current Biology*, vol. 25, no. 19, pp. 2457–2465, 2015.

[82] M. P. Broderick, A. J. Anderson, G. M. Di Liberto, M. J. Crosse, and E. C. Lalor, "Electrophysiological Correlates of Semantic Dissimilarity Reflect the Comprehension of Natural, Narrative Speech," *Current Biology*, vol. 28, no. 5, pp. 803–809.e3, 2018.

[83] C. Brodbeck, L. E. Hong, and J. Z. Simon, "Rapid Transformation from Auditory to Linguistic Representations of Continuous Speech," *Current Biology*, vol. 28, no. 24, pp. 3976–3983.e5, 2018.

[84] H. Weissbart, K. D. Kandylaki, and T. Reichenbach, "Cortical Tracking of Surprisal during Continuous Speech Comprehension," *Journal of Cognitive Neuroscience*, vol. 32, no. 1, pp. 155–166, 2020.

[85] M. Koskinen, M. Kurimo, J. Gross, A. Hyvärinen, and R. Hari, "Brain activity reflects the predictability of word sequences in listened continuous speech," *NeuroImage*, vol. 219, no. 116936, 2020.

[86] M. Gillis, J. Vanthornhout, J. Z. Simon, T. Francart, and C. Brodbeck, "Neural Markers of Speech Comprehension: Measuring EEG Tracking of Linguistic Speech Representations, Controlling the Speech Acoustics," *Journal of Neuroscience*, vol. 41, no. 50, pp. 10 316–10 329, 2021.

[87] J. Van Canneyt, J. Wouters, and T. Francart, "Neural tracking of the fundamental frequency of the voice: The effect of voice characteristics," *European Journal of Neuroscience*, vol. 53, no. 11, pp. 3640–3653, 2021.

[88] J. Van Canneyt, J. Wouters, and T. Francart, "Enhanced Neural Tracking of the Fundamental Frequency of the Voice," *IEEE Transactions on Biomedical Engineering*, vol. 68, no. 12, pp. 3612–3619, 2021.

[89] J. J. Foxe, G. V. Simpson, and S. P. Ahlfors, "Parieto-occipital ˜10Hz activity reflects anticipatory state of visual attention mechanisms," *NeuroReport*, vol. 9, no. 17, pp. 3929–3933, 1998.

[90] M. S. Worden, J. J. Foxe, N. Wang, and G. V. Simpson, "Anticipatory Biasing of Visuospatial Attention Indexed by Retinotopically Specific alpha-Band Electroencephalography Increases over Occipital Cortex," *Journal of Neuroscience*, vol. 20, no. 6, 2000.

[91] M. Bauer, S. Kennett, and J. Driver, "Attentional selection of location and modality in vision and touch modulates low-frequency activity in associated sensory cortices," *Journal of Neurophysiology*, vol. 107, no. 9, pp. 2342–2351, 2012.

[92] S. Haegens, B. F. Händel, and O. Jensen, "Top-down Controlled Alpha Band Activity in Somatosensory Areas Determines Behavioral Performance in a Discrimination Task," *Journal of Neuroscience*, vol. 31, no. 14, pp. 5197–5204, 2011.

[93] T. Wolbers, P. Zahorik, and N. A. Giudice, "Decoding the direction of auditory motion in blind humans," *NeuroImage*, vol. 56, no. 2, pp. 681–687, 2011.

[94] S. A. McLaughlin, N. C. Higgins, and G. C. Stecker, "Tuning to Binaural Cues in Human Auditory Cortex," *Journal of the Association for Research in Otolaryngology*, vol. 17, pp. 37–53, 2016.

[95] J. Ahveninen, S. Huang, J. W. Belliveau, W.-T. Chang, and M. Hämäläinen, "Dynamic oscillatory processes governing cued orienting and allocation of auditory attention," *Journal of Cognitive Neuroscience*, vol. 25, no. 11, pp. 1926–1943, 2013.

[96] J. N. Frey, N. Mainy, J. P. Lachaux, N. Muller, O. Bertrand, and N. Weisz, "Selective Modulation of Auditory Cortical Alpha Activity in an Audiovisual Spatial Attention Task," *Journal of Neuroscience*, vol. 34, no. 19, pp. 6634–6639, 2014.

[97] P. Patel, L. K. Long, J. Herrero, A. D. Mehta, and N. Mesgarani, "Joint Representation of Spatial and Phonetic Features in the Human Core Auditory Cortex," *Cell Reports*, vol. 24, no. 8, pp. 2051–2062.e2, 2018.

[98] Y. Deng, I. Choi, and B. G. Shinn-Cunningham, "Topographic specificity of alpha power during auditory spatial attention," *NeuroImage*, vol. 207, no. 116360, 2020.

[99] J. R. Kerlin, A. J. Shahin, and L. M. Miller, "Attentional Gain Control of Ongoing Cortical Speech Representations in a "Cocktail Party"," *Journal of Neuroscience*, vol. 30, no. 2, pp. 620–628, 2010.

[100] M. Wöstmann, B. Herrmann, B. Maess, and J. Obleser, "Spatiotemporal dynamics of auditory attention synchronize with speech," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 113, no. 14, pp. 3873–3878, 2016.

[101] B. T. Paul, M. Uzelac, E. Chan, and A. Dimitrijevic, "Poor early cortical differentiation of speech predicts perceptual difficulties of severely hearing-impaired listeners in multi-talker environments," *Scientific Reports*, vol. 10, no. 6141, 2020.

[102] A. Bednar, F. M. Boland, and E. C. Lalor, "Different spatio-temporal electroencephalography features drive the successful decoding of binaural and monaural cues for sound localization," *European Journal of Neuroscience*, vol. 45, no. 5, pp. 679–689, 2017.

[103] A. Bednar and E. C. Lalor, "Neural tracking of auditory motion is reflected by delta phase and alpha power of EEG," *NeuroImage*, vol. 181, pp. 683–691, 2018.

[104] A. Bednar and E. C. Lalor, "Where is the cocktail party? Decoding locations of attended and unattended moving sound sources using EEG," *NeuroImage*, vol. 205, no. 116283, 2020.

[105] J. Belo, M. Clerc, and D. Schön, "EEG-Based Auditory Attention Detection and Its Possible Future Applications for Passive BCI," *Frontiers in Computer Science*, vol. 3, no. 661178, 2021.

[106] M. Hosseini, L. Celotti, and E. Plourde, "Speaker-Independent Brain Enhanced Speech Denoising," in Proceedings of the *2021 IEEE*

International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1310–1314, 2021.

[107] T. Gerkmann, M. Krawczyk-Becker, and J. Le Roux, "Phase Processing for Single-Channel Speech Enhancement: History and recent advances," *IEEE Signal Processing Magazine*, vol. 32, no. 2, pp. 55–66, 2015.

[108] S. Gannot, E. Vincent, S. Markovich-Golan, and A. Ozerov, "A Consolidated Perspective on Multimicrophone Speech Enhancement and Source Separation," *IEEE/ACM Transactions on Audio Speech and Language Processing*, vol. 25, no. 4, pp. 692–730, 2017.

[109] Y. Luo and N. Mesgarani, "Conv-TasNet: Surpassing Ideal Time-Frequency Magnitude Masking for Speech Separation," *IEEE/ACM Transactions on Audio Speech and Language Processing*, vol. 27, no. 8, pp. 1256–1266, 2019.

[110] S. Van Eyndhoven, T. Francart, and A. Bertrand, "EEG-Informed Attended Speaker Extraction from Recorded Speech Mixtures with Application in Neuro-Steered Hearing Prostheses," *IEEE Transactions on Biomedical Engineering*, vol. 64, no. 5, pp. 1045–1056, 2017.

[111] A. Aroudi, B. Mirkovic, M. De Vos, and S. Doclo, "Impact of Different Acoustic Components on EEG-Based Auditory Attention Decoding in Noisy and Reverberant Conditions," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 4, pp. 652–663, 2019.

[112] A. Aroudi and S. Doclo, "Cognitive-Driven Binaural Beamforming Using EEG-Based Auditory Attention Decoding," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 862–875, 2020.

[113] N. Das, J. Zegers, H. Van hamme, T. Francart, and A. Bertrand, "Linear versus deep learning methods for noisy speech separation for EEG-informed attention decoding," *Journal of Neural Engineering*, vol. 17, no. 4, p. 046039, 2020.

[114] W. Pu, J. Xiao, T. Zhang, and Z.-Q. Luo, "A Joint Auditory Attention Decoding and Adaptive Binaural Beamforming Algorithm for Hearing Devices," in Proceedings of the *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 311–315, 2019.

[115] W. Pu, P. Zan, J. Xiao, T. Zhang, and Z.-Q. Luo, "Evaluation of Joint Auditory Attention Decoding and Adaptive Binaural Beamforming Approach for Hearing Devices with Attention Switching," in Proceedings of the *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 8728–8732, 2020.

[116] B. J. Borgström, M. S. Brandstein, G. A. Ciccarelli, T. F. Quatieri, and C. J. Smalt, "Speaker separation in realistic noise environments with applications to a cognitively-controlled hearing aid," *Neural Networks*, vol. 140, pp. 136–147, 2021.

[117] E. Ceolini, J. Hjortkjær, D. D. Wong, J. A. O'Sullivan, V. S. Raghavan, J. Herrero, A. D. Mehta, S. C. Liu, and N. Mesgarani, "Brain-informed speech separation (BISS) for enhancement of target speaker in multitalker speech perception," *NeuroImage*, vol. 223, no. 117282, 2020.

[118] B. Mirkovic, M. G. Bleichner, M. De Vos, and S. Debener, "Target Speaker Detection with Concealed EEG Around the Ear," *Frontiers in Neuroscience*, vol. 10, no. 349, 2016.

[119] G. A. Ciccarelli, M. Nolan, J. Perricone, P. T. Calamia, S. Haro, J. A. O'Sullivan, N. Mesgarani, T. F. Quatieri, and C. J. Smalt, "Comparison of Two-Talker Attention Decoding from EEG with Nonlinear Neural Networks and Linear Methods," *Scientific Reports*, vol. 9, no. 1, p. 11538, 2019.

[120] B. Mirkovic, S. Debener, M. Jaeger, and M. De Vos, "Decoding the attended speech stream with multi-channel EEG: implications for online, daily-life applications," *Journal of Neural Engineering*, vol. 12, no. 4, p. 46007, 2015.

[121] A. Mundanad Narayanan and A. Bertrand, "Analysis of Miniaturization Effects and Channel Selection Strategies for EEG Sensor Networks with Application to Auditory Attention Detection," *IEEE Transactions on Biomedical Engineering*, vol. 67, no. 1, pp. 234–244, 2020.

[122] A. Mundanad Narayanan, R. Zink, and A. Bertrand, "EEG miniaturization limits for stimulus decoding with EEG sensor networks," *Journal of Neural Engineering*, vol. 18, no. 5, p. 056042, 2021.

[123] P. J. Schäfer, F. I. Corona-Strauss, R. Hannemann, S. A. Hillyard, and D. J. Strauss, "Testing the Limits of the Stimulus Reconstruction Approach: Auditory Attention Decoding in a Four-Speaker Free Field Environment," *Trends in Hearing*, vol. 22, 2018.

[124] L. Straetmans, B. Holtze, S. Debener, M. Jaeger, and B. Mirkovic, "Neural tracking to go: auditory attention decoding and saliency detection with mobile EEG," *Journal of Neural Engineering*, vol. 18, no. 6, p. 066054, dec 2021.

[125] S. Miran, A. Presacco, J. Z. Simon, M. C. Fu, S. I. Marcus, and B. Babadi, "Dynamic estimation of auditory temporal response functions via state-space models with Gaussian mixture process noise," *PLoS Computational Biology*, vol. 16, no. 8, p. e1008172, 2020.

[126] A. Presacco, S. Miran, B. Babadi, and J. Z. Simon, "Real-Time Tracking of Magnetoencephalographic Neuromarkers during a Dynamic Attention-Switching Task," in Proceedings of the *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 4148–4151, 2019.

[127] S. Haro, H. M. Rao, T. F. Quatieri, and C. J. Smalt, "EEG alpha and pupil diameter reflect endogenous auditory attention switching and listening effort," *European Journal of Neuroscience*, pp. 1–16, 2022.

[128] E. B. Petersen, M. Wöstmann, J. Obleser, and T. Lunner, "Neural tracking of attended versus ignored speech is differentially affected by hearing loss," *Journal of Neurophysiology*, vol. 117, no. 1, pp. 18–27, 2017.

[129] S. A. Fuglsang, J. Märcher-Rørsted, T. Dau, and J. Hjortkjær, "Effects of Sensorineural Hearing Loss on Cortical Synchronization to Competing Speech during Selective Attention," *Journal of Neuroscience*, vol. 40, no. 12, pp. 2562–2572, 2020.

[130] W. Nogueira, G. Cosatti, I. Schierholz, M. Egger, B. Mirkovic, and A. Büchner, "Towards Decoding Selective Attention from Single-Trial EEG Data in Cochlear Implant Users," *IEEE Transactions on Biomedical Engineering*, vol. 67, no. 1, pp. 38–49, 2019.

[131] F. J. Vanheusden, M. Kegler, K. Ireland, C. Georga, D. M. Simpson, T. Reichenbach, and S. L. Bell, "Hearing Aids Do Not Alter Cortical Entrainment to Speech at Audible Levels in Mild-to-Moderately Hearing-Impaired Subjects," *Frontiers in Human Neuroscience*, vol. 14, no. 109, 2020.

[132] N. Das, T. Francart, and A. Bertrand, "Auditory Attention Detection Dataset KULeuven," 2019. [Online]. Available: https://doi.org/10.5281/zenodo.3997352

[133] S. A. Fuglsang, D. D. Wong, and J. Hjortkjær, "EEG and audio dataset for auditory attention decoding," *Zenodo*, 2018. [Online]. Available: https://doi.org/10.5281/zenodo.1199011

[134] N. Das, W. Biesmans, A. Bertrand, and T. Francart, "The effect of head-related filtering and ear-specific decoding bias on auditory attention detection," *Journal of Neural Engineering*, vol. 13, no. 056014, 2016.

[135] N. Das, J. Vanthornhout, T. Francart, and A. Bertrand, "Stimulus-aware spatial filtering for single-trial neural response and temporal response function estimation in high-density EEG with applications in auditory research," *NeuroImage*, vol. 204, no. 116211, 2020.

[136] S. Vandecappelle, L. Deckers, N. Das, A. H. Ansari, A. Bertrand, and T. Francart, "EEG-based detection of the locus of auditory attention with convolutional neural networks," *eLife*, vol. 10, no. e56481, 2021.

[137] A. Mundanad Narayanan, P. Patrinos, and A. Bertrand, "Optimal Versus Approximate Channel Selection Methods for EEG Decoding with Application to Topology-Constrained Neuro-Sensor Networks," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 29, pp. 92–102, 2021.

[138] I. Kuruvila, J. Muncke, E. Fischer, and U. Hoppe, "Extracting the Auditory Attention in a Dual-Speaker Scenario From EEG Using a Joint CNN-LSTM Model," *Frontiers in Physiology*, vol. 12, no. 700655, 2021.

[139] P. Li, S. Cai, E. Su, and L. Xie, "A Biologically Inspired Attention Network for EEG-Based Auditory Attention Detection," *IEEE Signal Processing Letters*, 2021.

[140] E. Su, S. Cai, L. Xie, H. Li, and T. Schultz, "STAnet: A Spatiotemporal Attention Network for Decoding Auditory Spatial Attention from EEG," *IEEE Transactions on Biomedical Engineering*, 2022.

[141] S. Cai, P. Li, E. Su, and L. Xie, "Auditory Attention Detection via Cross-Modal Attention," *Frontiers in Neuroscience*, vol. 15, no. 652058, 2021.

[142] D. D. Wong, S. A. Fuglsang, J. Hjortkjær, E. Ceolini, M. Slaney, and A. de Cheveigné, "A Comparison of Regularization Methods in Forward and Backward Models for Auditory Attention Decoding," *Frontiers in Neuroscience*, vol. 12, no. 531, 2018.

[143] T. de Taillez, B. Kollmeier, and B. T. Meyer, "Machine learning for decoding listeners' attention from electroencephalography evoked by continuous speech," *European Journal of Neuroscience*, vol. 51, no. 5, pp. 1234–1241, 2017.

[144] J. R. Wolpaw, H. Ramoser, D. J. McFarland, and G. Pfurtscheller, "EEG-Based Communication: Improved Accuracy by Response Verification," *IEEE Transactions on Rehabilitation Engineering*, vol. 6, no. 3, pp. 326–333, 1998.

[145] S. Geirnaert, T. Francart, and A. Bertrand, "MESD Toolbox," *GitHub*, 2019. [Online]. Available: https://github.com/exporl/mesd-toolbox

[146] B. Ohlenforst, A. A. Zekveld, T. Lunner, D. Wendt, G. Naylor, Y. Wang, N. J. Versfeld, and S. E. Kramer, "Impact of stimulus-related factors and hearing impairment on listening effort as indicated by pupil dilation," *Hearing Research*, vol. 351, pp. 68–79, 2017.

[147] E. Alickovic, T. Lunner, F. Gustafsson, and L. Ljung, "A Tutorial on Auditory Attention Identification Methods," *Frontiers in Neuroscience*, vol. 13, p. 153, 2019.

[148] D. D. Wong, J. Hjortkjær, E. Ceolini, and A. de Cheveigné, "COCOHA Matlab Toolbox," 2018. [Online]. Available: https://cocoha.org/the-cocoha-matlab-toolbox/

[149] G. Schalk, D. J. McFarland, T. Hinterberger, N. Birbaumer, and J. R. Wolpaw, "BCI2000: A General-Purpose Brain-Computer Interface (BCI) System," *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 6, pp. 1034–1043, 2004.

[150] P. Brémaud, *Markov Chains: Gibbs Fields, Monte Carlo Simulation and Queues*, 2nd ed. Springer, Cham, 2020.

[151] R. McGarrigle, K. J. Munro, P. Dawes, A. J. Stewart, D. R. Moore, J. G. Barry, and S. Amitay, "Listening effort and fatigue: What exactly are we measuring? A British Society of Audiology Cognition in Hearing Special Interest Group 'white paper'," *International Journal of Audiology*, vol. 53, no. 7, pp. 433–445, 2014.

[152] L. Decruy, N. Das, E. Verschueren, and T. Francart, "The Self-Assessed Békesy Procedure: Validation of a Method to Measure Intelligibility of Connected Discourse," *Trends in Hearing*, vol. 22, pp. 1–13, 2018.

[153] A. de Cheveigné, D. D. Wong, G. M. Di Liberto, J. Hjortkjær, M. Slaney, and E. C. Lalor, "Decoding the auditory brain with canonical component analysis," *NeuroImage*, vol. 172, pp. 206–216, 2018.

[154] S. Geirnaert, T. Francart, and A. Bertrand, "Fast EEG-based decoding of the directional focus of auditory attention using common spatial patterns," *IEEE Transactions on Biomedical Engineering*, vol. 68, no. 5, pp. 1557–1568, 2021.

[155] S. Geirnaert, T. Francart, and A. Bertrand, "Riemannian Geometry-Based Decoding of the Directional Focus of Auditory Attention Using EEG," in Proceedings of the *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1115–1119, 2021.

[156] I. Kuruvila, K. Can Demir, E. Fischer, and U. Hoppe, "Inference of the Selective Auditory Attention Using Sequential LMMSE Estimation," *IEEE Transactions on Biomedical Engineering*, vol. 68, no. 12, pp. 3501–3512, 2021.

[157] J. P. Dmochowski, J. J. Ki, P. DeGuzman, P. Sajda, and L. C. Parra, "Extracting multidimensional stimulus-response correlations using hybrid encoding-decoding of neural activity," *NeuroImage*, vol. 180, pp. 134–146, 2018.

[158] S. Geirnaert, T. Francart, and A. Bertrand, "An Interpretable Performance Metric for Auditory Attention Decoding Algorithms in a Context of Neuro-Steered Gain Control," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 28, no. 1, pp. 307–317, 2020.

[159] A. Aroudi, T. de Taillez, and S. Doclo, "Improving Auditory Attention Decoding Performance of Linear and Non-Linear Methods Using State-Space Model," in Proceedings of the *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 8703–8707, 2020.

[160] L. Wang, E. X. Wu, and F. Chen, "EEG-based auditory attention decoding using speech-level-based segmented computational models," *Journal of Neural Engineering*, vol. 18, no. 4, p. 046066, 2021.

[161] A. Gałecki and T. Burzykowski, *Linear Mixed-Effects Models Using R: A Step-by-Step Approach*, ser. Springer Texts in Statistics, Springer, New York, NY, 2013.

[162] V. A. Brown, "An Introduction to Linear Mixed-Effects Modeling in R," *Advances in Methods and Practices in Psychological Science*, vol. 4, no. 1, pp. 1–19, 2021.

[163] M. J. Crosse, G. M. Di Liberto, A. Bednar, and E. C. Lalor, "The Multivariate Temporal Response Function (mTRF) Toolbox: A MATLAB Toolbox for Relating Neural Signals to Continuous Stimuli," *Frontiers in Human Neuroscience*, vol. 10, no. 604, 2016.

[164] O. Ledoit and M. Wolf, "A well-conditioned estimator for large-dimensional covariance matrices," *Journal of Multivariate Analysis*, vol. 88, no. 2, pp. 365–411, 2004.

[165] Y. Chen, A. Wiesel, Y. C. Eldar, and A. O. Hero, "Shrinkage algorithms for MMSE covariance estimation," *IEEE Transactions on Signal Processing*, vol. 58, no. 10, pp. 5016–5029, 2010.

[166] J. M. Borwein and A. S. Lewis, *Convex Analysis and Nonlinear Optimization: Theory and Examples*, 2nd ed. Springer-Verlag New York, 2006.

[167] W. Gautschi, *Numerical Analysis*, 2nd ed. Boston: Birkhäuser Boston, 2012.

[168] S. J. Orfanidis, "Exponential Smoothing," in *Applied Optimum Signal Processing*, 2018, ch. 6, pp. 221–315.

[169] K. Ogata, "Transient and Steady-State Response Analyses," in *Modern Control Engineering*, 5th ed. Pearson Education, 2010, ch. 6, pp. 159–268.

[170] B. Blankertz, R. Tomioka, S. Lemm, M. Kawanabe, and K.-R. Müller, "Optimizing Spatial Filters for Robust EEG Single-Trial Analysis," *IEEE Signal Processing Magazine*, vol. 25, no. 1, pp. 41–56, 2007.

[171] K. K. Ang, Z. Y. Chin, C. Wang, C. Guan, and H. Zhang, "Filter bank common spatial pattern algorithm on BCI competition IV Datasets 2a and 2b," *Frontiers in Neuroscience*, vol. 6, no. 39, 2012.

[172] S. Lemm, B. Blankertz, G. Curio, and K.-R. Müller, "Spatio-Spectral Filters for Improving the Classification of Single Trial EEG," *IEEE Transactions on Biomedical Engineering*, vol. 52, no. 9, pp. 1541–1548, sep 2005.

[173] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY, USA: Springer, 2006.

[174] F. Lotte, Cuntai Guan, and K. K. Ang, "Comparison of Designs Towards a Subject-Independent Brain-Computer Interface based on Motor Imagery," in Proceedings of the *2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 4543–4546, 2009.

[175] C. Vidaurre, M. Kawanabe, P. von Bünau, B. Blankertz, and K.-R. Müller, "Toward Unsupervised Adaptation of LDA for Brain-Computer Interfaces," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 3, pp. 587–597, 2011.

[176] Y. Gao, Q. Wang, Y. Ding, C. Wang, H. Li, X. Wu, T. Qu, and L. Li, "Selective Attention Enhances Beta-Band Cortical Oscillation to Speech under "Cocktail-Party" Listening Conditions," *Frontiers in Human Neuroscience*, vol. 11, no. 34, 2017.

[177] S. Haufe, F. C. Meinecke, K. Görgen, S. Dähne, J. D. Haynes, B. Blankertz, and F. Bießmann, "On the interpretation of weight vectors of linear models in multivariate neuroimaging," *NeuroImage*, vol. 87, pp. 96–110, 2014.

[178] D. J. Strauss, F. I. Corona-Strauss, A. Schroeer, P. Flotho, R. Hannemann, and S. A. Hackley, "Vestigial auriculomotor activity indicates the direction of auditory attention in humans," *eLife*, vol. 9, no. e54536, 2020.

[179] A. Barachant, S. Bonnet, M. Congedo, and C. Jutten, "Multiclass Brain-Computer Interface Classification By Riemannian Geometry," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 4, pp. 920–928, 2012.

[180] A. Barachant, S. Bonnet, M. Congedo, and C. Jutten, "Classification of covariance matrices using a Riemannian-based kernel for BCI applications," *Neurocomputing*, vol. 112, pp. 172–178, 2013.

[181] S. Chevallier, E. K. Kalunga, Q. Barthélemy, and E. Monacelli, "Review of Riemannian Distances and Divergences, Applied to SSVEP-based BCI," *Neuroinformatics*, vol. 19, pp. 93–106, 2021.

[182] M. Congedo, B. Afsari, A. Barachant, and M. Moakher, "Approximate Joint Diagonalization and Geometric Mean of Symmetric Positive Definite Matrices," *PLoS ONE*, vol. 10, no. 4, p. 121423, 2015.

[183] A. A. Ivanova, M. Schrimpf, S. Anzellotti, N. Zaslavsky, and L. Isik, "Beyond linear regression: mapping models in cognitive neuroscience should align with research goals," *bioRxiv*, pp. 1–22, 2021.

[184] K. B. Mikkelsen, Y. R. Tabar, C. B. Christensen, and P. Kidmose, "EEGs Vary Less Between Lab and Home Locations Than They Do Between People," *Frontiers in Computational Neuroscience*, vol. 15, no. 565244, 2021.

[185] M. J. Crosse, N. J. Zuk, G. M. Di Liberto, A. R. Nidiffer, S. Molholm, and E. C. Lalor, "Linear Modeling of Neurophysiological Responses to Speech and Other Continuous Stimuli: Methodological Considerations for Applied Research," *Frontiers in Neuroscience*, vol. 15, no. 705621, 2021.

[186] F. Lotte, "Signal Processing Approaches to Minimize or Suppress Calibration Time in Oscillatory Activity-Based Brain–Computer Interfaces," *Proceedings of the IEEE*, vol. 103, no. 6, pp. 871–890, 2015.

[187] B. Blankertz, M. Kawanabe, R. Tomioka, F. U. Hohlefeld, V. Nikulin, and K.-R. Müller, "Alleviating nonstationarities in brain-computer interfacing," in Proceedings of the *20th International Conference on Neural Information Processing Systems (NIPS)*, pp. 113–120, 2007.

[188] F. Lotte and C. Guan, "Regularizing Common Spatial Patterns to Improve BCI Designs: Unified Theory and New Algorithms," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 2, pp. 355–362, 2011.

[189] W. Samek, C. Vidaurre, K.-R. Müller, and M. Kawanabe, "Stationary common spatial patterns for brain-computer interfacing," *Journal of Neural Engineering*, vol. 9, no. 026013, 2012.

[190] S. Doclo, W. Kellermann, S. Makino, and S. Nordholm, "Exploiting spatial diversity using multiple microphones," *IEEE Signal Processing Magazine*, vol. 32, no. 2, pp. 18–30, 2015.

[191] L. Wang, V. Best, and B. G. Shinn-Cunningham, "Benefits of Beamforming With Local Spatial-Cue Preservation for Speech Localization and Segregation," *Trends in Hearing*, vol. 24, pp. 1–11, 2020.

[192] R. Sitaram, T. Ros, L. Stoeckel, S. Haller, F. Scharnowski, J. Lewis-Peacock, N. Weiskopf, M. L. Blefari, M. Rana, E. Oblak, N. Birbaumer, and J. Sulzer, "Closed-loop brain training: the science of neurofeedback," *Nature Reviews Neuroscience*, vol. 18, no. 2, pp. 86–100, 2017.

[193] S. Enriquez-Geppert, R. J. Huster, and C. S. Herrmann, "EEG-Neurofeedback as a Tool to Modulate Cognition and Behavior: A Review Tutorial," *Frontiers in Human Neuroscience*, vol. 11, no. 51, 2017.

[194] J. S. Müller, C. Vidaurre, M. Schreuder, F. C. Meinecke, P. von Bünau, and K.-R. Müller, "A mathematical model for the two-learners problem," *Journal of Neural Engineering*, vol. 14, no. 3, p. 036005, 2017.

[195] J. Dan, S. Geirnaert, and A. Bertrand, "Grouped Variable Selection for Generalized Eigenvalue Problems," *Signal Processing*, vol. 195, no. 108476, 2022.

Acknowledgments

This work was carried out at the ESAT-STADIUS Center for Dynamical Systems, Signal Processing, and Data Analytics (Department of Electrical Engineering), and the Research Group Experimental Oto-rhino-laryngology (ExpORL) (Department of Neurosciences), both at KU Leuven.

Simon Geirnaert was supported by an Aspirant Grant from the Research Foundation - Flanders (FWO) (no. 1136219N). Furthermore, the work in this thesis was supported by FWO project no. G0A4918N, the European Research Council (ERC) under the European Union's Horizon 2020 Research and Innovation Programme (grant agreement no. 802895 and no. 637424), and the Flemish Government (AI Research Program).

Curriculum vitae

Simon Geirnaert was born in Waregem, Belgium on December 18, 1995.

In 2018, he obtained an MSc degree in Mathematical Engineering (summa cum laude) at KU Leuven. In September 2018, he joined the STADIUS Center for Dynamical Systems, Signal Processing, and Data Analytics (Department of Electrical Engineering, KU Leuven) and the research group Experimental Oto-rhino-laryngology (ExpORL, Department of Neurosciences, KU Leuven) as a PhD student under the supervision of Prof. dr. ir. Alexander Bertrand and Prof. dr. ir. Tom Francart. In his research, he focused on developing novel signal processing algorithms for EEG-based auditory attention decoding in neuro-steered hearing devices. His research was supported by an Aspirant Grant of the Research Foundation Flanders (FWO).

He was a finalist of the Agoriaprijs 2018 for his master thesis on tensor-based atrial fibrillation detection. In 2019, he received the best presentation and paper award at the 40^{th} IEEE Symposium on Information Theory and Signal Processing in the Benelux, awarded by the Stichting Gauss. During his PhD, he was active in numerous science outreach activities, and his research has been covered by various (inter)national media.

List of Publications

Articles in internationally reviewed journals

1. **S. Geirnaert**, T. Francart, and A. Bertrand, "An Interpretable Performance Metric for Auditory Attention Decoding Algorithms in a Context of Neuro-Steered Gain Control," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 28, no. 1, pp. 307-317, 2020.

2. **S. Geirnaert**, S. Vandecappelle, E. Alickovic, A. de Cheveigné, E. C. Lalor, B. T. Meyer, S. Miran, T. Francart, and A. Bertrand, "Electroencephalography-Based Auditory Attention Decoding: Toward Neurosteered Hearing Devices," *IEEE Signal Processing Magazine*, vol. 38, no. 4, pp. 89-102, 2021.

3. **S. Geirnaert**, T. Francart, and A. Bertrand, "Fast EEG-based Decoding of the Directional Focus of Auditory Attention Using Common Spatial Patterns," *IEEE Transactions on Biomedical Engineering*, vol. 68, no. 5, pp. 1557-1568, 2021.

4. **S. Geirnaert**, T. Francart, and A. Bertrand, "Unsupervised Self-Adaptive Auditory Attention Decoding," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 10, pp. 3955-3966, 2021.

5. J. Dan[*], **S. Geirnaert**[*], and A. Bertrand, "Grouped Variable Selection for Generalized Eigenvalue Problems," *Signal Processing*, vol. 195, no. 108476, 2022.
([*]: joint first author)

6. **S. Geirnaert**, T. Francart, and A. Bertrand, "Time-adaptive Unsupervised Auditory Attention Decoding Using EEG-based Stimulus Reconstruction," Accepted for publication in *IEEE Journal of Biomedical and Health Informatics*, 2022.

Papers in proceedings of international conferences

1. **S. Geirnaert**, G. Goovaerts, S. Padhy, M. Boussé, L. De Lathauwer, and S. Van Huffel, "Tensor-based ECG Signal Processing Applied to Atrial Fibrillation Detection," in Proceedings of the *2018 52nd Asilomar Conference on Signals, Systems, and Computers (ACSSC)*, Pacific Grove, CA, USA, pp. 799-805, Oct. 2018.

2. **S. Geirnaert**, T. Francart, and A. Bertrand, "A New Metric to Evaluate Auditory Attention Detection Performance Based on a Markov Chain," in Proceedings of the *2019 27th European Signal Processing Conference (EUSIPCO)*, A Coruña, Spain, pp. 1-5, Sept. 2019.

3. **S. Geirnaert**, T. Francart, and A. Bertrand, "Riemannian Geometry-Based Decoding of the Directional Focus of Auditory Attention Using EEG," in Proceedings of the *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Toronto, ON, Canada, pp. 1115-1119, June 2021.

Abstracts in proceedings of (inter)national conferences

1. **S. Geirnaert**, G. Goovaerts, S. Padhy, M. Boussé, L. De Lathauwer, and S. Van Huffel, "Tensor-based ECG Signal Processing Applied to Atrial Fibrillation Detection", *17th National Day on Biomedical Engineering*, Brussels, Belgium, 30 Nov. 2019. Poster presentation.

2. **S. Geirnaert**, S. Vandecappelle, N. Das, T. Francart, and A. Bertrand, "Towards Neuro-Steered Hearing Prostheses", *f-TALES: Neuro Sense & Sense-ability*, Leuven, Belgium, 1-2 April 2019. Poster presentation.

3. **S. Geirnaert**, T. Francart, and A. Bertrand, "Expected Switching Time: a Markov Chain Based Performance Metric to Evaluate Auditory Attention Decoding Algorithms," *SITB2019: 40th WIC Symposium on Information Theory in the Benelux/9th joint WIC IEEE SP Symposium on Information Theory and Signal Processing in the Benelux*, Ghent, Belgium, 28-29 May 2019. Oral presentation.

4. **S. Geirnaert**, S. Vandecappelle, A. Bertrand, and T. Francart, "Assessing Auditory Attention Decoding Algorithms for Gain Control in Neuro-Steered Hearing Aids," *International Hearing Instruments Developer Forum 2019*, Oldenburg, Germany, 13-14 June 2019. Oral presentation.

5. **S. Geirnaert**, S. Vandecappelle, T. Francart, and A. Bertrand, "A Comparative Study of Auditory Attention Decoding Algorithms," *Auditory EEG Signal Processing (AESoP) Symposium*, Leuven, 16-18 Sept. 2019. Oral presentation.
6. **S. Geirnaert**, T. Francart, and A. Bertrand, "An Interpretable Performance Metric for Evaluating Neural Decoders in the Context of Auditory Attention-Based Gain Control," *Auditory EEG Signal Processing (AESoP) Symposium*, Leuven, 16-18 Sept. 2019. Poster presentation.

Science communication and outreach

1. **S. Geirnaert**, T. Francart, and A. Bertrand, "Hoorapparaat van de toekomst luistert naar hersengolven", *KU Leuven News*, Nov. 2020. News article. This article around the work published in [154] has been covered in various national (newspapers, radio, tv) and international media (newspapers).
2. **S. Geirnaert**, T. Francart, and A. Bertrand, "Hersengestuurde hoorapparaten," *Science Figured Out*, Sept. 2021. Video pitch.
3. **S. Geirnaert**, T. Francart, and A. Bertrand, "Luisteren in de chaos: stuur een hoorapparaat aan met je hersenen", *EOS blogs*, Nov. 2021. Science blog. Also published on the Leuven.AI-stories blog and in the BioVox newsletter.
4. **S. Geirnaert**, D. Fieberg, T. Francart, and A. Bertrand, "Riemannian geometry-based decoding of the directional focus of auditory attention using EEG", *Youtube*, Oct. 2021. Video presentation.
5. **S. Geirnaert**, T. Francart, and A. Bertrand, "Demo EEG-based auditory attention decoding using the stimulus reconstruction algorithm", *Youtube*, Oct. 2021. Demo.
6. **S. Geirnaert**, T. Francart, and A. Bertrand, "Demo EEG-based decoding of the directional focus of auditory attention using common spatial patterns", *Youtube*, Oct. 2021. Demo.
7. **S. Geirnaert**, D. Fieberg, T. Francart, and A. Bertrand, "Unsupervised self-adaptive auditory attention decoding", *Youtube*, Oct. 2021. Video presentation.
8. **S. Geirnaert**, T. Francart, and A. Bertrand, "Demo time-adaptive unsupervised auditory attention decoding using EEG-based stimulus reconstruction", *Youtube*, Mar. 2022. Demo.

FACULTY OF ENGINEERING SCIENCE
DEPARTMENT OF ELECTRICAL ENGINEERING
STADIUS CENTER FOR DYNAMICAL SYSTEMS, SIGNAL PROCESSING AND DATA ANALYTICS
Kasteelpark Arenberg 10 box 2446
B-3001 Leuven
simon.geirnaert@esat.kuleuven.be
http://esat.kuleuven.be/stadius