

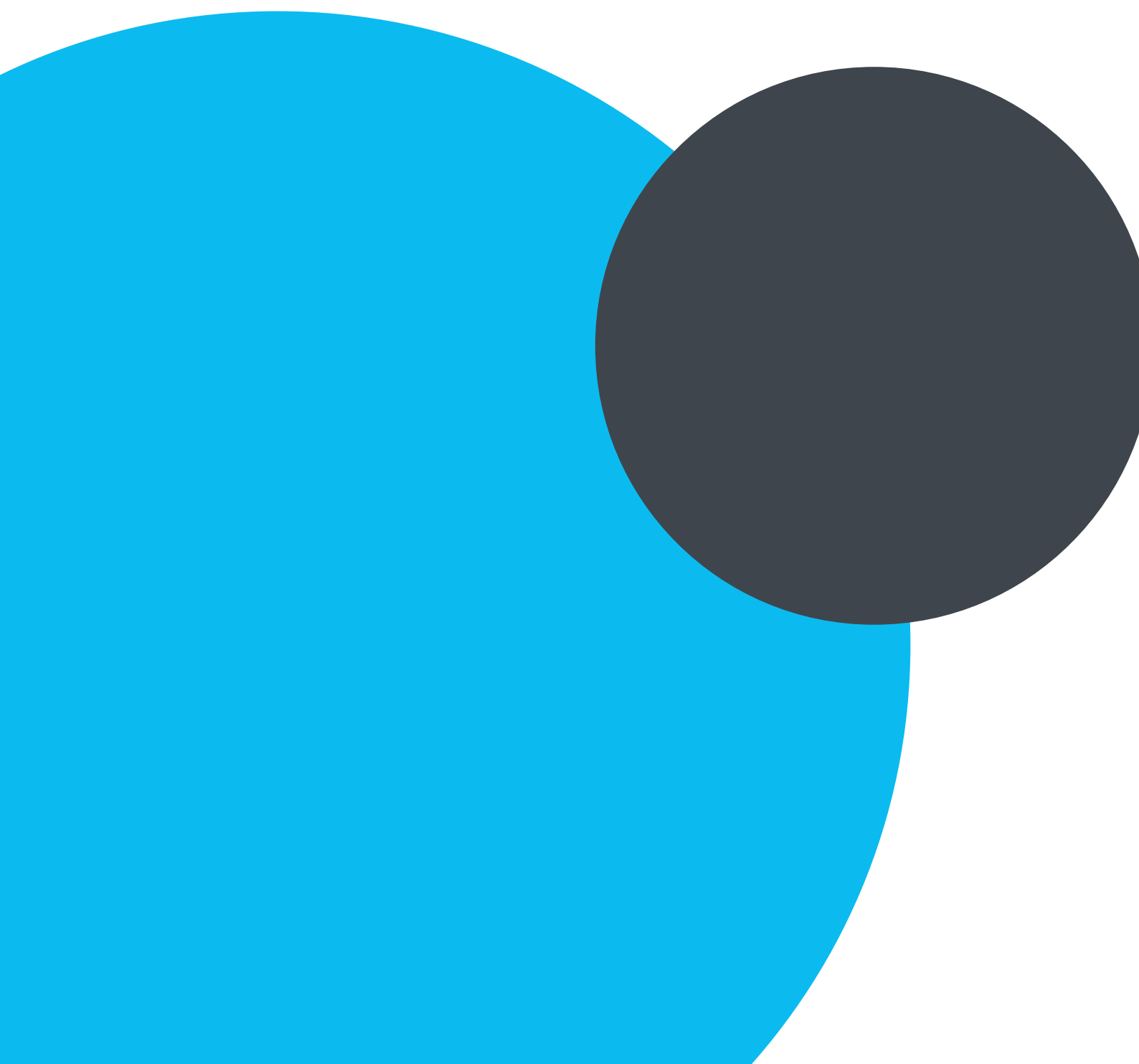
**University of Stuttgart**  
Germany

Faculty of Electrical Engineering and Computer Science  
Computational Cognitive Science

# **EEG Artifact Simulation: an open-source implementation for eye movements, power line noise, and drifts**

Maanik Marathe

M.Sc. thesis in Information Technology — December 2025



## **Supervisors**

<b>Main Supervisor:</b>	Benedikt Ehinger	Computational Cognitive Science Faculty of Electrical Engineering and Computer Science, Department of Computer Science
<b>Second Supervisor:</b>	Judith Schepers	Computational Cognitive Science Faculty of Electrical Engineering and Computer Science, Department of Computer Science





# Abstract

The data recorded in a study using electroencephalography (EEG) also contains unwanted signals, called 'artifacts', which must be removed in a preprocessing step before the data can be analyzed. The tools and methods developed for this purpose can be tested by means of applying the preprocessing to a dataset and comparing the result with the expected output. However, there is a lack of suitable datasets, and researchers are often missing knowledge of the ground truth or have specific ground truth requirements that the real study may not fulfil. Thus, using real recorded data for this purpose is not always ideal. Instead, researchers often decide to simulate data containing EEG and artifact components that correspond to their requirements.

Current tools and techniques for artifact simulation still depend on real datasets, are difficult to reproduce and are often based in proprietary software. This thesis describes the first steps towards modelling these artifacts based on their origin and properties and simulating them in an extensible and reproducible manner, keeping the simulation method open-source. In particular, the artifacts selected for simulation include the eye movement artifact, power line noise and slow drifts. The implementation of this simulation is designed to be compatible with and eventually integrated into an existing open-source EEG simulation library, UnfoldSim.jl.



# Acknowledgements

I would like to thank my supervisors, Dr. Benedikt Ehinger and Judith Schepers, for their guidance and support during my work on this thesis.

I also thank Nils Harmening for the correspondence on the HArtMuT model and for the simulation of the new forward models based on our suggested modifications.



# Contents

<b>Abstract</b>	<b>iii</b>
<b>Acknowledgements</b>	<b>v</b>
<b>Contents</b>	<b>vii</b>
<b>List of Abbreviations</b>	<b>ix</b>
<b>1 Introduction</b>	<b>1</b>
1.1 EEG Artifacts . . . . .	2
1.2 Motivation for simulation in EEG data processing . . . . .	3
1.3 Previous work on EEG and artifact simulation . . . . .	5
1.4 Aim and scope of this thesis . . . . .	6
<b>2 EEG Artifacts</b>	<b>9</b>
2.1 Eye Movement artifact . . . . .	9
2.2 Power Line Noise . . . . .	12
2.3 Drift artifact . . . . .	12
2.4 Characteristics of selected artifacts in EEG . . . . .	13
2.4.1 Eye Movements . . . . .	13
2.4.2 Power line noise . . . . .	13
2.4.3 Drifts . . . . .	14
<b>3 Simulation implementation</b>	<b>15</b>
3.1 Overview of simulation interfaces . . . . .	16
3.1.1 AbstractContinuousSignal . . . . .	16
3.1.2 Control Signal . . . . .	16
3.1.3 Generating the control signal . . . . .	18

3.2	Artifact simulation . . . . .	18
3.2.1	Eye Movement Simulation . . . . .	18
3.2.2	Power Line Noise . . . . .	19
3.2.3	Drift . . . . .	20
3.2.4	User-defined artifact . . . . .	20
3.3	Simulation Example . . . . .	21
<b>4</b>	<b>Discussion</b>	<b>23</b>
4.1	Limitations . . . . .	23
4.1.1	Eye movements . . . . .	24
4.1.2	Power line noise . . . . .	24
4.1.3	Drifts . . . . .	25
4.2	Outlook and future Scope . . . . .	25
4.2.1	General improvements . . . . .	25
4.2.2	Evaluation of simulated artifacts . . . . .	26
4.2.3	Eye movements . . . . .	27
4.2.4	Power line noise . . . . .	27
4.2.5	Drift . . . . .	27
<b>5</b>	<b>Summary</b>	<b>29</b>
	<b>Bibliography</b>	<b>31</b>
<b>A</b>	<b>Appendices</b>	<b>37</b>
A.1	Use of AI tools . . . . .	37
<b>B</b>	<b>Declaration of authorship</b>	<b>39</b>

# List of Abbreviations

**CRD** corneo-retinal dipole

**EEG** electroencephalography

**EOG** electrooculogram

**HArtMuT** Head Artifact Model using Tripoles





# Introduction

In bioelectrical signal research, electroencephalography (EEG) is a method of recording the electrical activity of the brain via a set of electrodes (or sensors), usually placed on the scalp and/or the facial skin of the participant. During the EEG recording, each of the electrodes measures the electrical potential at its respective location, with one electrode being designated as the reference electrode. The recorded value of the EEG at a given electrode is a measure of the potential difference between that electrode and the reference electrode (Hari, 2017).

The time-series data thus gathered is later analyzed in order to understand more about the inner working of the brain. Of interest to researchers are the electrical potentials arising from brain activity; however, the raw recorded data also contains unwanted components, called “artifacts”. These artifact potentials partially or completely obscure the brain-related signals, since they often have a similar or much larger magnitude compared to the latter.

Thus, the data must be cleaned of artifacts before it can be analyzed. For this preprocessing, a variety of techniques and tools have been developed. These are then used in order to process real data for the purpose of answering specific research questions. In order to test these techniques, researchers often simulate

datasets containing the artifacts of interest for the particular technique in question. This thesis focuses on the modelling and simulation of eye movements, power line noise and drift artifacts in EEG in order to provide these researchers with a more flexible and reproducible method of EEG artifact simulation.

The following subsections discuss EEG artifacts and their simulation in more detail, along with laying out the aim and scope of this thesis.

## 1.1 EEG Artifacts

The artifact potentials in EEG data can be broadly divided into two categories: those originating from the subject's physiological processes, and those originating in the environment. The individual artifacts can vary in magnitude at different electrodes or over time as the recording progresses, and each artifact has certain characteristics that can help better distinguish it from the EEG and other artifacts.

For example, the subject's eye movements and blinks cause large artifacts in the recording. The muscles of the head, face, and neck are electrically activated when in use, causing their own artifacts in the measured scalp potentials; the electrical activity during the heartbeat also generates a measurable change in the EEG; and when the subject sweats it causes a corresponding artifact as well.

Non-physiological artifacts include power line noise (electrical noise caused by the interference from alternating-current electrical power supply), noise due to a bad connection with an electrode, sudden large changes in electrode potentials ('pop'), and artifacts due to swinging of the cable.

To a certain extent, the experimental setup and procedure at the time of EEG recording can help avoid the artifacts being recorded at all. For example, subjects can be asked to stay as still as possible during the EEG recording to avoid artifacts due to their movements; the setup can be well isolated from AC power lines; and the experimental room can be kept at a controlled temperature and humidity level so that artifacts due to sweating are minimized (Hari, 2017).

However, some artifacts (like eye movement, blink artifacts, or drifts) cannot be easily avoided at the source. These must be then removed from the raw data via preprocessing, before the cleaned data is then ready for analysis. This is

often done by means of special algorithms and processing tools implemented using computer code.

## 1.2 Motivation for simulation in EEG data processing

Modern EEG research often involves the use of software to process recorded data, for example to detect artifacts or to perform artifact correction on the detected artifacts. Popular software packages for EEG data processing include EEGLab (Delorme & Makeig, 2004), autoreject (Jas et al., 2016), and MNE (Larson et al., 2024). Individual researchers usually also write their own code to help perform further analysis, either independently of such packages or by building on the functions provided therein.

Whether in the form of a standalone tool or a custom script, this code must be tested in order to make sure it satisfies the requirements. For this, a sample dataset can be processed using the code and then the researcher can inspect or evaluate the dataset before and after processing to determine whether the required effect was achieved. Specifically, in the case of preprocessing for artifact detection, the artifacts in the sample dataset should all be correctly detected, and for artifact rejection the dataset after applying the preprocessing must be free of the particular artifact.

Care must be taken in choosing the test dataset - it should contain instances of the target phenomenon (in this case, the artifact), these instances should be well labeled, and there should be enough data (both, with as well as without the artifact) in order to tell with high certainty that the analysis code has performed the required task. In addition, since no two instances of the artifact are the same, the selected test dataset should also capture this variation. However, existing recorded EEG datasets do not necessarily fulfil these requirements.

Firstly, the dataset might not be large enough to support the analysis, for one or more of several reasons. Recording a larger dataset implies collecting longer individual recordings, more participants, or both; these may be restricted by lack of time and financial constraints on the researchers. Overall dataset size is limited due to constraints like the available storage space, and participant comfort considerations play a significant role during longer recordings. Low manpower or lack of access to suitable subjects can reduce the number of individual recordings that can be made. The modalities of data that can be

recorded are limited by factors like the available equipment and the complexity of the experimental setup.

Thus, for example, experimenters carrying out a particular study may have access to EEG equipment but no eye tracker, and thus the resulting dataset does not contain associated eye tracking data of the participants. This results in the dataset being less suitable for testing preprocessing methods that target eye artifacts. A different study may record data of fewer participants due to stricter exclusion criteria, or the experimental setup may be uncomfortable for the participant and limit how long the recordings are for each subject, leading to less data to work with overall.

This can mean that a single recording does not have well-suited or sufficient data to train a model or validate the analysis code. Therefore, either multiple test datasets must be used, or time must be spent to find or record a large enough usable dataset.

Next, there may not be sufficient instances of the artifact in the selected dataset, or the labeling may not be of good quality. The variety in the instances may also be low, such that it is difficult to get different test cases for the analysis code.

Finally, the ground truth, i.e. the sequence and details of the actual events that occurred during the recording, is not known to anyone except perhaps the original researchers. Later researchers often have to make assumptions and educated guesses when analyzing the data, and this can result in undiscovered errors.

One solution to this problem could be to record a new dataset specifically for the purpose of testing a particular piece of analysis code. It can be useful to have a degree of control over the ground truth in order to have a variety of different test cases for the current analysis, and in a custom study, the researcher has more freedom to design the experiment and instructions according to what they require. For example, to test code that identifies and removes blink artifacts, a recording could be carried out in which the participant is instructed to blink at specific times while looking in different specified directions.

However, designing and carrying out such a study can be expensive and time-consuming, and not all researchers have the resources for this. Further, even in a purpose-built study, the participant may deviate from the instructions, or the

ground-truth requirements may be too complex to easily communicate to the participant.

Simulating EEG data and artifacts for this purpose then becomes a useful option. Along with reducing the effort that would come along with conducting a full experiment, it also provides the researcher with more control over the artifacts present in the test dataset, in terms of characteristics like variety, frequency of occurrence, distribution over time and space, and so on.

### 1.3 Previous work on EEG and artifact simulation

Synthetic data creation has been a part of several previous studies. One possibility for creating a synthetic dataset is to create a physical simulator and to record the data with sensors as in usual EEG studies. Yu & Hairston (2021) created the “Open EEG Phantom”, an open-source project to allow researchers to create their own dummy head for EEG recording. This consists of a phantom head made of materials of varying conductivities to mimic the real human head. The phantom head contains several antennae placed at specific locations, and custom electrical signals are broadcast into space through the head via these antennae. This can then be used to simulate EEG and various artifacts, depending on the goal of the current study. For example, Downey & Ferris (2023) developed a novel algorithm to remove EEG artifacts. In order to validate their algorithm, they created test data by building and using a model head based on the Open EEG Phantom, and broadcasting EEG along with artifacts from eye movements, neck and facial muscles, and walking movements, into the model head. For some of these signals, they recorded data specifically for this study, and for others they extracted the EEG and artifact signals from an existing dataset, and broadcast those into the dummy head.

It is also possible to simulate the data entirely in software. For this purpose, EEG simulation packages provide general support to simulate EEG data from brain sources and to add different kinds of random noise to the data (Barzegaran et al., 2019; Krol et al., 2018; Larson et al., 2024; Schepers et al., 2025).

When creating an artifact-contaminated dataset for a particular study, some researchers choose to create a semi-simulated dataset by obtaining artifact-free recordings (usually by cleaning a regular recording) and then contaminating this data by taking samples of recorded artifacts and projecting them to the scalp electrodes (Anzolin et al., 2021; Couchman et al., 2024; Klados &

Bamidis, 2016; Toit et al., 2021). Romero et al. (2008) created a simulated EEG-EOG dataset from a real recorded dataset by calculating a weighted sum of the EEG potentials of certain randomly chosen subjects and the EOG potentials of different randomly chosen subjects.

Other methods are also possible: Mutanen et al. (2024) modelled muscle artifacts from an existing dataset and used their model to simulate further muscle artifacts, and similarly Kim et al. (2025) simulated eye blinks and two muscle artifacts that they identified from empirical data. Barbara et al. (2023) defined and extended their own battery model of the eye to simulate eye movement artifacts. Leske & Dalal (2019) simulated power line noise with fluctuating amplitude and varied on- and offsets for their combined MEG/EEG study.

However, most of the methods above still rely on real datasets to some extent. Furthermore, the simulation itself is not carried out in a standardized way, instead being done either manually by the researcher or via custom scripts. Most of the artifact simulation code currently available is primarily based in MATLAB, which is proprietary software and therefore less accessible for future researchers, and the custom scripts themselves may not be made publicly available by their authors. The individual researcher's judgement also plays a large role in the output of the simulation, and the reasons behind the choices they make are not always documented. Under these conditions, it is difficult to replicate the results of the original researchers. Finally, there is a large amount of effort required to create a custom model and simulation process based on the requirements of each individual study.

Thus, there is a need for a single method for artifact simulation in a standardized and accessible way that is also flexible enough to be customized if so required.

## **1.4 Aim and scope of this thesis**

As described above, there is a gap in the available software for EEG artifact simulation. The aim of this thesis is to take the first step towards filling this gap, in the form of an open-source, easy to use implementation that is compatible with an existing EEG simulation package.

For each artifact, there is first a discussion of its physical origin and how it manifests in EEG recordings. Then, a method to simulate it is presented, corre-

sponding to the code in the thesis repository. While designing and implementing the simulation, care is taken to take into account different possible use cases and ensure clean interfaces between sections of the code. Finally, limitations and possible future work are discussed.

For the scope of the thesis, a set of artifacts has been selected, namely eye movement artifacts, power line noise, and drifts. The simulation code starts from a simple, basic implementation and builds up to more complexity as required.

The following research questions were formulated in order to better guide the work done on this thesis:

1. Given a set of chosen EEG artifacts (eye movement, power line noise and electrode drift), what is known about their origin and characteristics?
2. How are these artifacts usually simulated?
3. How can we simulate them in a way that is standardized and compatible with existing open-source EEG simulation packages?



# /2

## EEG Artifacts

### 2.1 Eye Movement artifact

Eye-related EEG artefacts arise mainly from eyeball movements, eye muscle activation, and eyelid movements. This section primarily deals with the artefact due to eyeball movement.

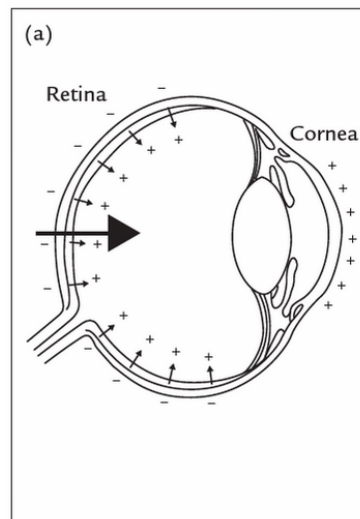
The eye is made up of different kinds of tissue, some of which generate a standing electrical charge distribution due to their intrinsic properties. The cornea (at the front of the eye) and the retina (at the back) together constitute the entire surface of the eyeball. Each of these tissue types has an electrical potential difference between its inner and outer surfaces. Specifically, the cornea is positively charged on the outside of the eyeball and the retina is negatively charged on the outside of the eyeball (Iwasaki et al., 2005).

As the eye rotates in its socket, the corresponding charge distribution rotates along with it and changes the resulting electric field, manifesting as eye movement artifacts measured in the EEG at the scalp. This can be more distinctly seen in the electrooculogram (EOG), a special signal recorded during an EEG study by placing additional electrodes near the eyes in order to more clearly record the electrical activity resulting from eye movements. One pair of electrodes is

placed above and below one eye, and one pair at the temples, outwards to the side from each eye. The vertical and horizontal EOG respectively are the values of the potential differences calculated between these electrodes in pairs, and the magnitude of this potential difference changes in accordance with the movement of the eye.

Previous studies like Mowrer et al. (1935) and Matsuo et al. (1975) found that an intact eyeball (i.e., having both cornea and retina) is required in order to observe the EOG effects due to eye movements, and concluded that the eye has an overall potential difference between the front and back of the eyeball. This potential difference was named the “corneo-retinal potential” (Mowrer et al., 1935) and has been used as a basis of the model of eye movements in existing studies (Berg & Scherg, 1991; Lins et al., 1993a; Plöchl et al., 2012).

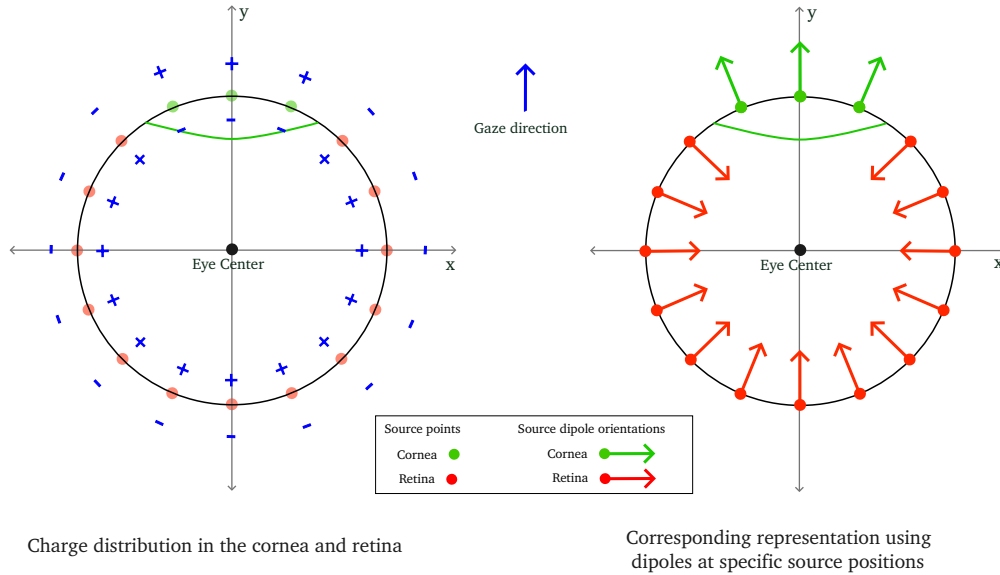
The resulting electrical potential distribution in the eye can thus be modelled in multiple ways. First, the charges of the cornea and retina can be combined and thought of as a single electrical current dipole with its positive end towards the cornea at the front, following the axis of gaze (Plöchl et al., 2012). This single dipole is commonly known as the “corneo-retinal dipole (CRD)” and the model can be called the “CRD model” (Marathe, 2025).



**Figure 2.1** – Vertical section of the eyeball with corneo-retinal dipole represented by the large black arrow. Reproduced from Hari (2017).

It is also possible to model the potential differences on the retina and cornea tissues as a set of smaller electrical dipoles placed on the tissue surface and

oriented in different directions, as in the Head Artifact Model using Tripoles (HArtMuT) (Harmening et al., 2022). In this case, the individual dipoles (also known as “source dipoles”) have their positive ends towards the inside or the outside of the eyeball, for the retina and cornea tissue respectively.



**Figure 2.2** – Ensemble model: Representation of eyeball charges using source dipoles (viewed downwards from the top of the head, horizontal cross-section). Based on figures in Marathe (2025).

The CRD and ensemble models have been explored more in detail in a previous work (Marathe, 2025), and the implementation from that work has been used and built upon in this thesis for the simulation of eye movement artifacts.

Along with the movement of the eyeball itself, other factors also affect the EEG artifacts arising from eye movements. According to Iwasaki et al. (2005), the eyelids move along with eye movements, and the effect is visible in the frontal EEG observed during vertical eye movements. Matsuo et al. (1975) discuss a possible explanation: when the eyelid makes contact with the charged surface of the cornea, it conducts the positive charges away from the eyeball towards the frontal electrodes. This is also known as the “sliding electrode” effect and has also been used to explain blink artifacts, which also involve interaction between the eyelid and the charged eyeball (Lins et al., 1993b; Matsuo et al., 1975).

## 2.2 Power Line Noise

The modern world is powered primarily by alternating current (AC) electrical power, delivered to devices via power cables and connectors. “Alternating current” means that the electrical current passing through the cables and circuits reverses direction at defined intervals i.e. at a particular frequency. There are two standard frequencies around the world, namely 50 Hz and 60 Hz.

A current carrying conductor creates an electric field around it, and when the direction of the current changes, so does the electric field. This means that the electric potential in a space is also affected by the alternating current carried by the power cables running through it.

Since the EEG is a measure of electrical potential, it also measures the changes in electrical potential that occur as a result of the alternating direction of the current in the power supply. The alternating current may also induce effects in equipment around the subject, which in turn can affect the measured EEG. This means that in an EEG recording there is also an unwanted component that results from the power supply, and this component is known as power line noise.

The exact mechanisms by which the power line noise enters the EEG system are not yet definitely known (De Cheveigné, 2020). The magnitude of the artifact can be reduced by measures like electrically shielding the experimental room or avoiding the use of AC power during the recording. However, it is not entirely possible to avoid recording this artifact. Thus, the presence of this noise is widely known and accepted in EEG studies, and efforts are concentrated rather on reducing the magnitude of this noise where possible in the recording setup, along with using post-processing methods to remove the artifact after the recording is complete.

## 2.3 Drift artifact

In a given EEG recording, it is often observed that the reading measured on a particular electrode slowly tends to drift upwards or downwards over time, seemingly independent of the measured EEG itself. This phenomenon is known as the drift artifact and can likewise cause problems when interpreting recorded EEG data.

Huigen et al. (2002) investigated the cause of this electrode noise and concluded that it originates from the contact between the electrodes and the skin. They described the two factors affecting this electrode noise - namely, the type of gel used to make the electrode-skin contact and the condition of the skin. They found that for wet-gel electrodes, the noise magnitude also depended on the electrode impedance.

This impedance can be affected by good or bad electrical contact at the electrode-skin interface, or even by sweating during the experiment. Common methods to reduce the drift include maintaining a cool environment to discourage sweating, and gently abrading the scalp surface before making contact with the electrode in order to create a low-impedance connection to the skin (Hari, 2017; Kappenman & Luck, 2010).

## **2.4 Characteristics of selected artifacts in EEG**

### **2.4.1 Eye Movements**

Eye movements can be seen as large deflections in the EEG, prominently visible in the electrodes at the front of the head and near the eyes, and to a lesser extent at electrodes further away from the eyes. EOG signals show the clearest picture of the artifact, as they are closest to the eyes and relatively far away from the brain.

From a particular gaze direction, when the eye rotates horizontally to one side, the electrodes on that side read more positive after the rotation than they did before the rotation, and electrodes on the other side experience a negative deflection i.e. are more negative than they were before. Similarly, for upward eye movements, electrodes above the eyes are more positive after the movement and those lower on the face are more negative after the movement (Hari, 2017).

These observations correspond to the effect one would expect from the corneo-retinal potential described in the literature.

### **2.4.2 Power line noise**

The power line noise can be seen with the naked eye in the raw EEG as regular oscillations at much higher frequencies than is normally expected in an EEG

recording. When observing the power spectral density of the data, sharp peaks can be seen at the power line frequency and its multiples. The artifact may not be uniformly distributed across the scalp electrodes (De Cheveigné, 2020); this may be a result of different mechanisms of transmission of the power line noise into the EEG system and the relative positioning of different electrodes with respect to the power lines themselves.

### **2.4.3 Drifts**

Drifts manifest as a slow increase or decrease over time in the baseline level around which the measured EEG signal appears. When extracted from the data, it looks similar to the figure below.

The drift artifact differs across channels, although it may be somewhat related in electrodes that are closer to each other (De Cheveigné & Arzounian, 2018). It also changes over time: Huigen et al. (2002) found that the noise in wet-gel electrodes reduces over time and noted that applying the electrodes some time before the start of the recording would be an advantage where low noise levels are desired.

# /3

## Simulation implementation

As described in Chapter 1.3, flexible and open-source methods for EEG and artifact simulation are much sought after, and software packages that simulate EEG (brain) data have been actively developed in recent years. One of these packages is `UnfoldSim.jl` (Schepers et al., 2025), which allows the user to specify an experiment design and simulate continuous EEG signals along with additional random noise. The package is also designed to be extensible, allowing the user to define their own custom types and operate flexibly with existing code.

Therefore, for this thesis, `UnfoldSim.jl` was chosen as a base package on which to build the further artifact simulation methods. Since `UnfoldSim` is written in Julia, this thesis follows the same. The artifact simulation code is designed from the start of development to be compatible with the `UnfoldSim` package; for the sake of convenience, this ‘extended’ version of `UnfoldSim` shall be called ‘`UnfoldSimArtifacts`’. However, it is planned to eventually integrate the artifact simulation into the `UnfoldSim` package itself: a fork of the package has been created and the simulation code has been incorporated directly into the structure of `UnfoldSim`.

The code and other artifacts supporting this written thesis are hosted on the Internet in a repository on the developer platform ‘GitHub’ and can be located by visiting [https://github.com/s-ccs/2025\\_MSc\\_EEG\\_artifacts\\_simulation](https://github.com/s-ccs/2025_MSc_EEG_artifacts_simulation)<sup>o</sup>.

### 3.1 Overview of simulation interfaces

The interfaces for artifact simulation have been designed to match the conventions and structure of `UnfoldSim.jl`, such that minimal changes would be required to add artifact simulation to an existing `UnfoldSim` simulation code.

The flow of the artifact-inclusive simulation is as follows: The basic simulation ingredients (experimental design, component, onsets, and noise) are defined as usual for `UnfoldSim`. In addition, the desired artifacts are defined, using the interfaces provided by `UnfoldSimArtifacts`. Next, the `simulate` function is called. Within the `simulate` function, the EEG is first simulated without any noise. Then, each artifact in turn is simulated; here, noise is also considered to be an artifact. Finally, the results of all the simulations are added together to give the simulated artifact-contaminated EEG data.

Some details about the implementation follow in the next few subsections.

#### 3.1.1 AbstractContinuousSignal

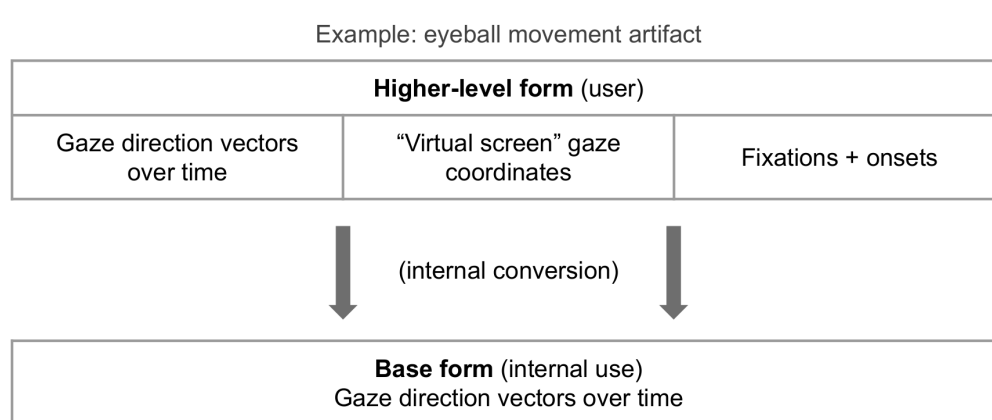
Keeping with the `UnfoldSim` convention of providing high-level abstract types with a few predefined concrete types, `UnfoldSimArtifacts` defines a new abstract type `AbstractContinuousSignal`, and for each of the three selected artifacts (eye movements, power line noise, drifts), a corresponding concrete subtype has been created. The user can define their requirements for the artifact by setting values for the fields in these concrete types. For example, the concrete type `PowerLineNoise` has a field `base_freq` to define the base frequency of the power supply. Similarly, each concrete type has its own set of fields that are required to simulate that artifact.

#### 3.1.2 Control Signal

Each artifact defined by a concrete subtype of `AbstractContinuousSignal` has a field `controlsignal`. This control signal is the means via which the user can control the simulated continuous signal i.e. artifact.

For each concrete artifact type, there is one basic kind of `controlsignal` that is internally used by `UnfoldSimArtifacts` when simulating the artifact. The user can choose to provide their control signal in the same form. However, since the internally used `controlsignal` is the control signal in its most detailed form, it may be inconvenient for the user to define their specifications at this level of detail. Thus, `UnfoldSimArtifacts` also provides other interfaces at a higher level i.e. closer to the format that the user can easily use. If a user chooses to provide their `controlsignal` in one of these higher-level forms, it will then be converted internally to the lowest-level control signal. This offers the user a greater degree of flexibility when defining their artifact.

For example, for eye movement artifact simulation, the internally used control signal is a set of vectors describing the direction of eye gaze in three-dimensional space at each time point in the simulation. However, the user may wish to define their control signal in terms of gaze angles at each time point, they may want to define eye gaze points on a screen instead of specifying gaze directions, or at an even higher level they may wish to simply provide a set of fixations and onsets and allow `UnfoldSimArtifacts` to generate the eye movements connecting these. Figure 3.1 shows this structure in a graphical manner.



**Figure 3.1** – Control signal at different levels of detail

The conversion from the high-level control signal to the low-level internally used form is made possible using the `generate_controlsignal` function, which is defined for various types of control signal.

### 3.1.3 Generating the control signal

```
function generate_controlsinal(
  rng::AbstractRNG,
  cs::GazeDirectionVectors,
  sim::Simulation
)
```

The function `generate_controlsinal` can have multiple methods, depending on the type of control signal passed in by the user. However, the output of `generate_controlsinal` will always be in the base form required for simulating the respective artifact. For example, for power line noise this could be a matrix of weights to be applied for each time point and channel, or for an eye movement artifact it could be a matrix of eye gaze coordinates.

## 3.2 Artifact simulation

### 3.2.1 Eye Movement Simulation

The eyeballs have been modelled using the HArtMuT model and the CRD and ensemble simulation models (Marathe, 2025). The concrete type for eye movements is constructed as below:

```
@with_kw struct EyeMovement{T} <: AbstractContinuousSignal
  controlsinal::T
  headmodel
  eye_model::String = "crd"
end
```

At the lowest level of simulation, the eye movement is controlled using the instantaneous **gaze direction vector**, a vector pointing from the center of the eye towards the current gaze target at that time point. This vector is described in three dimensions in the same coordinate frame as the head model.

Another possible interface would be to allow the user to specify the eye position in head-referenced angle coordinates, or place the simulated participant in front of a virtual screen and specify the screen coordinates where the participant looked instead of providing the gaze direction.

Based on the type of control signal given by the user, it is converted if required into gaze direction vectors. For each gaze direction, the lead field (i.e. the potential measured at the scalp electrodes when the eye source dipoles are turned on) is calculated, and this lead field calculated for all electrodes and at multiple time points gives us the EEG eye movement artifact matrix.

### 3.2.2 Power Line Noise

```
@with_kw struct PowerLineNoise <: AbstractContinuousSignal
    controlsignal = nothing
    base_freq::Float64 = 50
    harmonics::Array{Int64} = [1 3 5]
    weights_harmonics::Array{Float64} = ones(length(harmonics))
    sampling_rate::Float64 = 1000
end
```

The properties of the power line noise can be specified as shown in the code snippet: base frequency, harmonics required (i.e., additional sinusoidal waves having their frequency as multiples of the actual power line frequency), and the sampling rate of the power line noise. The user can also specify a set of weights for the harmonics to define the contribution of each harmonic component to the overall power line noise; for example, in order to give one harmonic twice the magnitude of the other, the weights for those harmonics can be defined as 2 and 1 respectively. By default, all harmonics are weighted equally in magnitude.

An optional `controlsignal` further specifies the weights to be applied to this noise type at each channel and time point before adding the noise to the rest of the simulated signal. If no `controlsignal` is specified by the user, it is generated to be a matrix of ones, i.e. all channels have the exact same power line noise added to them and the magnitude of this power line noise does not change over time.

The noise is first simulated for a single channel using the given base frequency and harmonics: sinusoidal waves having the base frequency and harmonic frequencies are generated and weighted according to the between-harmonics weights `weights_harmonics`. The single-channel artifact thus created is then weighted according to `controlsignal` and returned in order to be added to the remaining simulation results at the appropriate stage in the process.

### 3.2.3 Drift

```
@with_kw struct DriftNoise <: AbstractContinuousSignal
    ar::ARDriftNoise = ARDriftNoise( $\sigma$ =1)
    linear::LinearDriftNoise = LinearDriftNoise()
    dc::DCDriftNoise = DCDriftNoise()
end
```

The type `DriftNoise` contains three fields, each sub-types of drift noise: `DCDriftNoise` (changing DC offset), `LinearDriftNoise` (options for simulating a linear drift), and `ARDriftNoise` (autoregressive drift noise). Each of these types have their own fields where the user can specify the parameters for that noise type, and a corresponding method in the simulation steps.

Here as well, a single channel is simulated at a time, like with power line noise; however, in contrast to the power line noise, the drift noise is simulated anew for each channel. This is because the drift in individual channels is not necessarily related to that of other channels, whereas power line noise is essentially the same underlying artifact that may manifest with differing magnitudes in different channels.

When the user simulates directly with the type `DriftNoise`, by default all these three types of noise will be simulated and added together. However, the user can also choose to simulate only one of these types of drift noise by using the appropriate concrete type. They may also choose to extend the `DriftNoise` artifact with their own implementation of additional types.

### 3.2.4 User-defined artifact

The simulation in `UnfoldSimArtifacts` is designed to be as flexible as possible while still providing a unified structure and flow to simulate different artifacts. To this end, a special type is defined, called `UserDefinedContinuousSignal`.

This type has a similar structure to the predefined artifacts provided, but is not bound to any one specific type of artifact. An advantage of this is that the user can choose to simulate their own artifacts separately or even reuse an artifact extracted from real data, similar to the workflows currently in use. However, the common simulation flow still remains and the addition of the user-defined artifact can be unified with the simulations provided in the package, resulting in a more standardized simulation overall.

### 3.3 Simulation Example

The simulation of these artifacts can be demonstrated in a single simulation code snippet, as follows:

```
simulate(  
  MersenneTwister(1), # random number generator  
  design, mc, onset, # experimental design  
  [noise; # AbstractNoise  
   EyeMovement(HREFCoordinates(href_trajectory), # control signal  
               eyemodel, # forward model for the eye movement simulation  
               "ensemble");  
  PowerLineNoise();  
  DriftNoise()  
);
```

First, as before with `UnfoldSim`, an EEG experimental design is defined, along with the desired component, onsets, and noise.

In addition, an eye movement artifact is defined: the movement is controlled via head-referenced eye gaze trajectories contained in `href_trajectory`; a forward model is provided using `eyemodel`, and it is specified that the ensemble model should be used to simulate the eye movement. Power line noise and drift noise are also added to the simulation, making use of the inbuilt default parameters rather than having those values be specified by the user.

The documentation in the forked version of `UnfoldSim`, referenced in the GitHub repository for this project, contains a guide explaining this simulation in more depth.



# /4

## Discussion

### 4.1 Limitations

The goal of the thesis was to develop a structure and some basic simulation implementations of the selected artifacts. In this light, these first implementations for the artifact simulation are based on simplified models of the artifacts, created with certain assumptions in mind. There is scope to build more complex models by lifting the assumptions made, or to build new models entirely. For example, drift noise has been implemented using an autoregressive model along with simple DC drift and a linear component. In the future, other models can be implemented and the corresponding noise simulation extended to include these new models.

There remains an open question about weightage of the different components of the simulated artifact-included EEG signal. When simulating a signal consisting of EEG along with multiple artifacts and noise, each of these components can be weighted differently to yield a simulated dataset contaminated with the artifacts to different degrees. Currently, however, there is no in-built way to describe the desired relative magnitudes of the different components at a high level (e.g., the user cannot simply describe that the power line noise should have half the magnitude of the EEG and the eye movements). The user

has control over the weights of the power line noise and drift noise via the `controlsignal` field, however they must pre-calculate the required weights for each channel and time point, per artifact, in order to do so.

The current implementation does not provide a way to define individual onsets for different artifacts, i.e. all simulated artifacts will begin from the first time point. The control signal provides a way to delay an artifact, by padding it with zeroes (or inserting zeroes at the places where the artifact is not desired). However, this is cumbersome for the user; a better interface would be for example to allow specifying an onset time point or sample for the individual artifacts.

Some of the further limitations related to specific artifacts are described in individual sections below.

#### 4.1.1 Eye movements

The CRD and ensemble eye models used for the eye movement simulation have their own limitations, described in more detail in Marathe (2025). The eyelid movements accompanying the eye movements are not accounted for in the model, and that the artifacts due to muscle contractions when moving the eye are not simulated. This model, in short, simulates only the movement of the eyeball itself. Additionally, the ensemble model in particular assumes that the magnitude of the retina and cornea dipoles is the same, whereas this is not definitively proven and there is some difference of opinion in the literature on how much the retina and cornea sources at all contribute to the corneo-retinal potential (Berg & Scherg, 1991; Hari, 2017; Plöchl et al., 2012).

The current interfaces for specifying details of the desired eye movements are also limited. The user can define the participant's eye gaze relative to the head; however, when working with eye tracking experiments, it may be more useful to define the gaze in terms of coordinates on the screen in front of them. This kind of interface is currently not available.

#### 4.1.2 Power line noise

In the current implementation, the power line noise is simulated just once and this same noise is added to all the channels. If the user does not provide their own weights via the `controlsignal` field, the noise will by default be uniform for all the channels. This is not necessarily realistic, as for example one side of

the head may be closer to the power source and may therefore experience a greater impact of the power line noise.

The ratio of magnitudes of the different harmonics is also constant over the entire simulation, which means that it is not possible to have certain harmonics fade in or out at different points in time, but rather adjust the power line noise magnitude as a whole.

Finally, real-world line noise artifacts rarely occur purely at the exact base frequency and its multiples. Instead, the frequency of the artifact signal may drift or deviate slightly from the “true” base frequency of power transmission. The current simulation, however, uses only pure sinusoids at exactly the base and harmonic frequencies specified.

### **4.1.3 Drifts**

The current simulation on drifts is completely based on mathematical models and by default weights the three different types of drifts equally. More study of real data is required to determine how the magnitudes of the different sub-parts are related and to simulate using models that are perhaps more complex in order to bring the simulation nearer to real-world observed drifts.

## **4.2 Outlook and future Scope**

### **4.2.1 General improvements**

There is scope to improve the interfaces for describing the desired relative strengths of the various artifact potentials in the simulated output. For example, the user may specify that the simulated power line noise must appear on only a selection of channels and that certain channels of these should have a stronger effect of the artifact; they may require the overall strength of the power line noise to change over the course of the simulated recording and the maximum magnitude of this power line noise to be only half that of the EEG.

At several stages in the simulation, default values have been chosen. For example, the strengths of the electrical dipoles in the eye, or the relative weights of the artifacts in the final signal. The current defaults are based on convenience or ease of simulation. In future, more realistic defaults can be determined in a

data-driven manner, by analyzing different datasets and setting these default values in the simulation package.

Another possibility is to simulate artifacts as originating at specific spatial locations relative to the head. For this, a head model can be created that provides scalp potential leadfields from sources located even outside the head, similar to the current forward model (e.g. HArtMuT, Harmening et al. (2022)) that provides leadfields from sources within the head and neck. With such a forward model, it becomes possible to virtually place the head and the source for an artifact (e.g. power line noise or the heartbeat) somewhere in space and simulate the strength of the artifact more realistically at different channels according to the spatial relationship between the source and the electrodes.

Additional details on possible future features of the simulation can be found in the file “Improvements” in the project repository on GitHub.

#### **4.2.2 Evaluation of simulated artifacts**

The artifact simulations implemented here can be tested and evaluated in different ways. For example, artifact characteristics can be extracted from a known contaminated dataset and simulated, before comparing the simulated output with the original dataset. The evaluation can be done by applying the same artifact detection or rejection preprocessing steps to both, the real as well as simulated dataset, and the results of this preprocessing compared.

For eye movements, a dataset containing EEG as well as eye tracking can be used as a source dataset. The eye tracking data provides details about the exact eye movements performed, and eye related EEG artifacts can be extracted via preprocessing methods like independent component analysis. The same eye movements as the real dataset can be simulated, before comparing the output with the extracted eye artifact.

For power line noise, a similar comparison can be carried out by extracting the power line noise from a known contaminated dataset, calculating the power spectral density and comparing it with the output of simulating power line noise. Different weights can be specified for different channels, assisted by analyzing the magnitude of the power line noise at different electrodes. Topoplots can be used to visualise these differences between channels and between real and simulated data. A similar comparison can be made for real and simulated drift artifacts.

### 4.2.3 Eye movements

Since the model used in this thesis work only simulates the eyeball movement, an avenue for improvement could be to simulate the effect of the half-closed eyelid and its movement that accompanies eye movements. This can be achieved for example by differently weighting the contributions of source points that are in contact with the eyelid, similar to the “sliding electrode” model of eyelid closure discussed previously.

There are also multiple other possible interfaces to define or control the desired eye movement to be simulated. For example, the head of the simulated participant can be placed in front of a virtual screen and the user can specify gaze locations on the screen instead of defining them relative to the head. Similarly, rather than the user needing to define the position of the eye gaze at each individual time point, an interface can be provided for them to specify the gaze position at only certain time points, along with these time points as onsets, and then the simulation may independently generate the trajectories of the eye gaze as the eye moves between these gaze positions. The trajectory can be calculated based on what is known about the properties of natural eye movement, e.g. saccade lengths, typical fixation duration and the saccade main sequence.

Finally, a method can be defined that simply simulates realistic eye movements given a head and screen setup, for the case where the user does not need exact control over the eye movement or gaze position but instead simply wants eye movement artifacts to be present in the simulated dataset.

### 4.2.4 Power line noise

As described above, the power line noise simulation can be improved by providing better interfaces to specify weights and potentially by allowing the line noise source to be placed in the same virtual space as the head. In addition, the implementation can be made more complex by simulating a base frequency that deviates from the specified base frequency, to better match the real-world properties of the artifact.

### 4.2.5 Drift

Similar to the other two artifacts, further work on drift simulation can be done by studying drifts in real data and developing different models based on the results of the study.



# /5

## Summary

As new tools and techniques are developed for preprocessing and processing EEG data, there will be a continued need for simulation tools that enable researchers to more easily and efficiently generate test data for their analyses. Some researchers choose use an off-the-shelf tool developed specifically for simulation, and tailor it to their needs. While this fulfils the purpose of the specific study, these tools are usually part of proprietary software, and thus less accessible than open-source alternatives. Other researchers choose to themselves simulate datasets, with artifacts placed at known times or with known properties. This, however, adds extra work and leads to a lack of standardization and replicability in synthetic data creation.

The need for an open-source, flexible EEG simulation package has been partially filled by packages such as `UnfoldSim.jl`. However, there is still a lack of such simulation packages for artifacts in EEG.

In this thesis, some basic simulations of eye movement artifacts, power line noise and drift artifacts in EEG were built as an extension to `UnfoldSim.jl` (an existing EEG data simulation package). The implementation of the artifact simulation has certain limitations and there is scope for further development, as discussed in previous sections.

Finally, simulations for EEG artifacts other than the three discussed in this thesis remain to be explored. In the future, more work can be done based on the starting point explored here, in the same way that this first artifact simulation built on the work in `UnfoldSim.jl`.

# Bibliography

- Anzolin, A., Toppi, J., Petti, M., Cincotti, F., & Astolfi, L. (2021). SEED-G: Simulated EEG Data Generator for Testing Connectivity Algorithms. *Sensors*, 21, 3632. DOI: 10.3390/s21113632°
- Barbara, N., Camilleri, T. A., & Camilleri, K. P. (2023). Monopolar and bipolar electrooculography signal characteristics due to target displacements—have we seen the whole picture?. *Physiological Measurement*, 44(3), 35011. DOI: 10.1088/1361-6579/acb03d°
- Barzegaran, E., Bosse, S., Kohler, P. J., & Norcia, A. M. (2019). EEGSourceSim: A framework for realistic simulation of EEG scalp data using MRI-based forward models and biologically plausible signals and noise. *Journal of Neuroscience Methods*, 328, 108377. DOI: DOI: 10.1016/j.jneumeth.2019.108377°
- Berg, P., & Scherg, M. (1991). Dipole models of eye movements and blinks. *Electroencephalography and Clinical Neurophysiology*, 79(1), 36–44. DOI: 10.1016/0013-4694(91)90154-V°
- Couchman, J., Kaparounakis, O., Samarakoon, C., & Stanley-Marbell, P. (2024, ). *Simulated Eyeblink Artifact Removal with ICA: Effect of Measurement Uncertainty*. <https://arxiv.org/abs/2410.03261>°

- De Cheveigné, A. (2020). ZapLine: A simple and effective method to remove power line artifacts. *Neuroimage*, 207, 116356. DOI: 10.1016/j.neuroimage.2019.116356<sup>°</sup>
- De Cheveigné, A., & Arzounian, D. (2018). Robust detrending, rereferencing, outlier detection, and inpainting for multichannel data. *Neuroimage*, 172, 903–912. DOI: 10.1016/j.neuroimage.2018.01.035<sup>°</sup>
- Delorme, A., & Makeig, S. (2004). EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. *Journal of Neuroscience Methods*, 134(1), 9–21. DOI: 10.1016/j.jneumeth.2003.10.009<sup>°</sup>
- Downey, R. J., & Ferris, D. P. (2023). iCanClean Removes Motion, Muscle, Eye, and Line-Noise Artifacts from Phantom EEG. *Sensors*, 23(19). DOI: 10.3390/s23198214<sup>°</sup>
- Hari, R. (2017). *MEG-EEG Primer*. Oxford University Press USA - OSO. <https://ebookcentral.proquest.com/lib/uni-stuttgart/detail.action?docID=5746005><sup>°</sup>
- Harmening, N., Klug, M., Gramann, K., & Miklody, D. (2022). HArtMuT—modeling eye and muscle contributors in neuroelectric imaging. *Journal of Neural Engineering*, 19(6), 66041. DOI: 10.1088/1741-2552/aca8ce<sup>°</sup>
- Huigen, E., Peper, A., & Grimbergen, C. A. (2002). Investigation into the origin of the noise of surface electrodes. *Medical and Biological Engineering and Computing*, 40(3), 332–338. DOI: 10.1007/BF02344216<sup>°</sup>
- Iwasaki, M., Kellinghaus, C., Alexopoulos, A. V., Burgess, R. C., Kumar, A. N., Han, Y. H., Lüders, H. O., & Leigh, R. J. (2005). Effects of eyelid closure, blinks, and eye movements on the

electroencephalogram. *Clinical Neurophysiology*, 116(4), 878–885.  
DOI: DOI: 10.1016/j.clinph.2004.11.001<sup>°</sup>

Jas, M., Engemann, D., Raimondo, F., Bekhti, Y., & Gramfort, A. (2016, June). Automated rejection and repair of bad trials in MEG/EEG. *6<sup>th</sup> International Workshop on Pattern Recognition in Neuroimaging (PRNI)*. <https://hal.science/hal-01313458><sup>°</sup>

Kappenman, E. S., & Luck, S. J. (2010). The effects of electrode impedance on data quality and statistical significance in ERP recordings. *Psychophysiology*, 47(5), 888–904. DOI: 10.1111/j.1469-8986.2010.01009.x<sup>°</sup>

Kim, H., Chang, C.-Y., Kothe, C., Iversen, J. R., & Miyakoshi, M. (2025). Juggler's ASR: Unpacking the principles of artifact subspace reconstruction for revision toward extreme MoBI. *Journal of Neuroscience Methods*, 420, 110465. DOI: DOI: 10.1016/j.jneumeth.2025.110465<sup>°</sup>

Klados, M. A., & Bamidis, P. D. (2016). A semi-simulated EEG/EOG dataset for the comparison of EOG artifact rejection techniques. *Data in Brief*, 8, 1004–1006. DOI: DOI: 10.1016/j.dib.2016.06.032<sup>°</sup>

Krol, L. R., Pawlitzki, J., Lotte, F., Gramann, K., & Zander, T. O. (2018). SEREEGA: Simulating event-related EEG activity. *Journal of Neuroscience Methods*, 309, 13–24. DOI: 10.1016/j.jneumeth.2018.08.001<sup>°</sup>

Larson, E., Gramfort, A., Engemann, D. A., Leppakangas, J., Brodbeck, C., Jas, M., Brooks, T. L., Sassenhagen, J., McCloy, D., Luessi, M., King, J.-R., Höchenberger, R., Goj, R., Brunner, C., Favelier, G., Vliet, M. van, Wronkiewicz, M., Rockhill, A., Holdgraf, C., ... luzpaz. (2024, December). *MNE-Python*. Zenodo. DOI: 10.5281/ZENODO.592483<sup>°</sup>

Leske, S., & Dalal, S. S. (2019). Reducing power line noise in EEG and MEG data via spectrum interpolation. *Neuroimage*, 189, 763–776. DOI: 10.1016/j.neuroimage.2019.01.026°

Lins, O. G., Picton, T. W., Berg, P., & Scherg, M. (1993b). Ocular artifacts in EEG and event-related potentials I: Scalp topography. *Brain Topography*, 6(1), 51–63. DOI: 10.1007/BF01234127°

Lins, O. G., Picton, T. W., Berg, P., & Scherg, M. (1993a). Ocular artifacts in recording EEGs and event-related potentials II: Source dipoles and source components. *Brain Topography*, 6(1), 65–78. DOI: 10.1007/BF01234128°

Marathe, M. (2025). *Simulation of Eye Movement Artefacts in EEG: introducing the Ensemble and Corneo-Retinal Dipole methods*. [https://github.com/s-ccs/2024-ocular-artifacts-simulation/blob/d6d6294222603304a7b52a7c1b21b149dcc4fc75/report/thesis/report\\_2025-03-31.pdf](https://github.com/s-ccs/2024-ocular-artifacts-simulation/blob/d6d6294222603304a7b52a7c1b21b149dcc4fc75/report/thesis/report_2025-03-31.pdf)°

Matsuo, F., Peters, J. F., & Reilly, E. L. (1975). Electrical phenomena associated with movements of the eyelid. *Electroencephalography and Clinical Neurophysiology*, 38(5), 507–511. DOI: 10.1016/0013-4694(75)90191-1°

Mowrer, O. H., Ruch, T. C., & Miller, N. E. (1935). THE CORNEO-RETINAL POTENTIAL DIFFERENCE AS THE BASIS OF THE GALVANOMETRIC METHOD OF RECORDING EYE MOVEMENTS. *American Journal of Physiology-Legacy Content*, 114(2), 423–428. DOI: 10.1152/ajplegacy.1935.114.2.423°

Mutanen, T. P., Ilmoniemi, I., Atti, I., Metsomaa, J., & Ilmoniemi, R. J. (2024). A simulation study: comparing independent component analysis and signal-space projection – source-informed reconstruction for rejecting muscle artifacts evoked by transcranial magnetic stimulation. *Frontiers in Human Neuroscience*. DOI: 10.3389/fnhum.2024.1324958°

- Plöchl, M., Ossandón, J. P., & König, P. (2012). Combining EEG and eye tracking: identification, characterization, and correction of eye movement artifacts in electroencephalographic data. *Frontiers in Human Neuroscience*, 6. DOI: 10.3389/fnhum.2012.00278<sup>°</sup>
- Romero, S., Mañanas, M. A., & Barbanoj, M. J. (2008). A comparative study of automatic techniques for ocular artifact reduction in spontaneous EEG signals based on clinical target variables: A simulation case. *Computers in Biology and Medicine*, 38(3), 348–360. DOI: 10.1016/j.combiomed.2007.12.001<sup>°</sup>
- Schepers, J., Lips, L., Marathe, M., & Ehinger, B. V. (2025). UnfoldSim.jl: Simulating continuous event-based time series data for EEG and beyond. *Journal of Open Source Software*, 10(107), 6641. DOI: 10.21105/joss.06641<sup>°</sup>
- Toit, W. B. du, Venter, M., & Heever, D. van den. (2021). Semi-Synthetic EEG Data for the Evaluation of EEG Cleaning Methods. *SSRN Electronic Journal*. DOI: 10.2139/ssrn.3995406<sup>°</sup>
- Yu, A. B., & Hairston, W. D. (2021, January). *Open EEG Phantom*. OSF. DOI: 10.17605/OSF.IO/QRKA2<sup>°</sup>





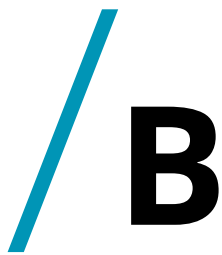
# Appendices

## A.1 Use of AI tools

Parts of the code associated with this Master thesis were produced with the assistance of AI tools. Specifically, in the “Visual Studio Code” code editor, the extension “GitHub Copilot” was used, for the purpose of code completion in sections of the file `AbstractContinuousSignal.jl`.

No AI tools were employed for the production of the written thesis or that of the materials used for the associated presentation.





## Declaration of authorship

I hereby declare that the work presented in this thesis is entirely my own. I did not use any sources or aids other than those specified (including AI-based applications or tools). All literal or paraphrased quotations and citations are identified and referenced. I assure that I have not used any aids whose use has been explicitly excluded by the examiner.

In the appendix “Use of AI Tools”, I have documented the AI tools I have used. By submitting this work, I assume responsibility for the entire product submitted. I am therefore also responsible for any AI-generated content that I have included in my work. I have checked the accuracy of the (AI-generated) statements and content to the best of my knowledge.

Neither this work nor any substantial parts of it have been the subject of any other examination procedure to date. I have not published this work either in part or in full to date. The electronic copy corresponds to all hard copies submitted.

Date and Signature:

