A novel Graphical User Interface for Non-Invasive Brain Stimulations

Gregor Dederichs

May 2024

Content

- **Introduction**
  - Temporal Interference
  - Current Interface Limitations
- **Explored Approaches**
  - Troubleshooting in MATLAB
  - NI-MAX software
  - Alternative Python packages
- **Results – HummelGUI**
  - Setup
  - Functionalities and features
  - Code structure & Adding features, Limitations

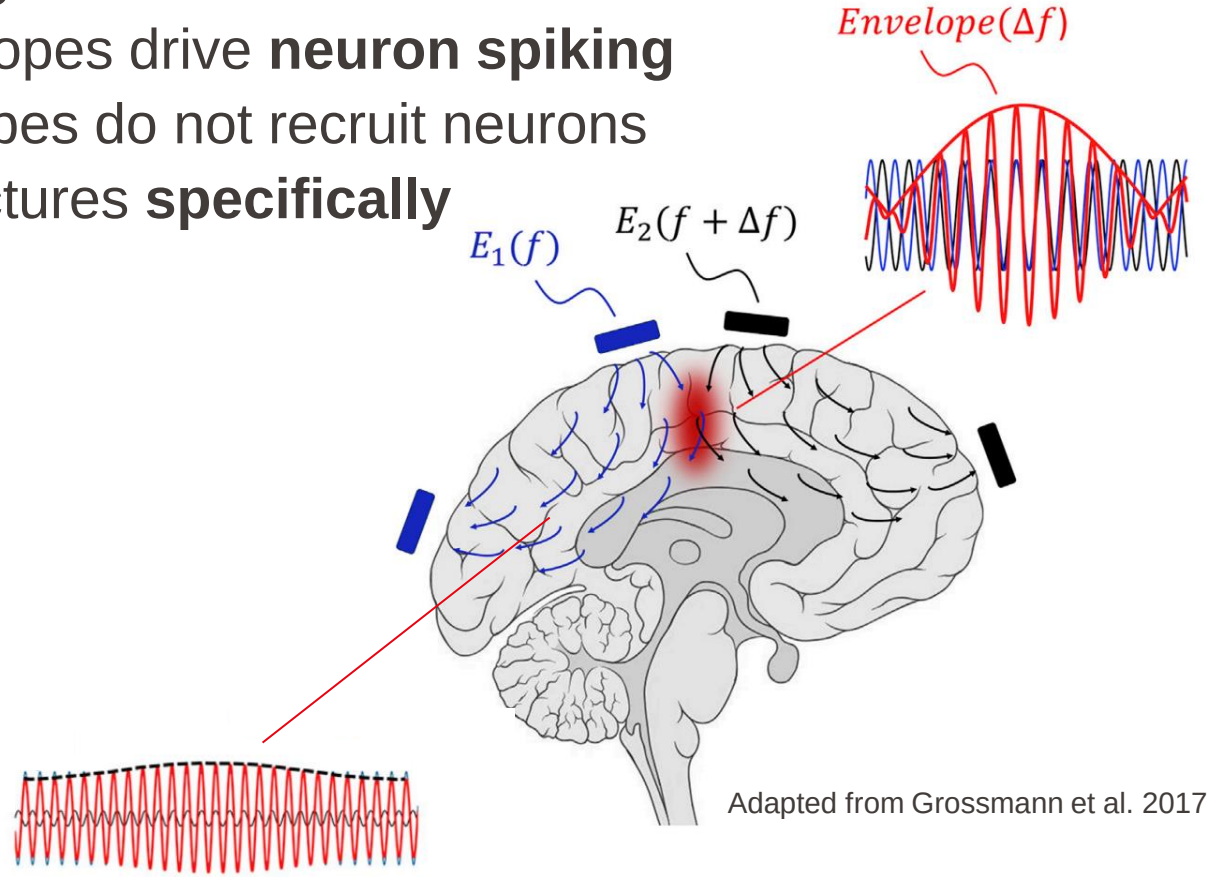
Introduction

Gregor Dederichs

Temporal Interference

- **Noninvasive Method for Deep-Brain Stimulation**

- **Temporally Interfering Electric Fields**
  - **Low frequency** envelopes drive **neuron spiking**
  - Low amplitude envelopes do not recruit neurons
  - Target areas and structures **specifically**

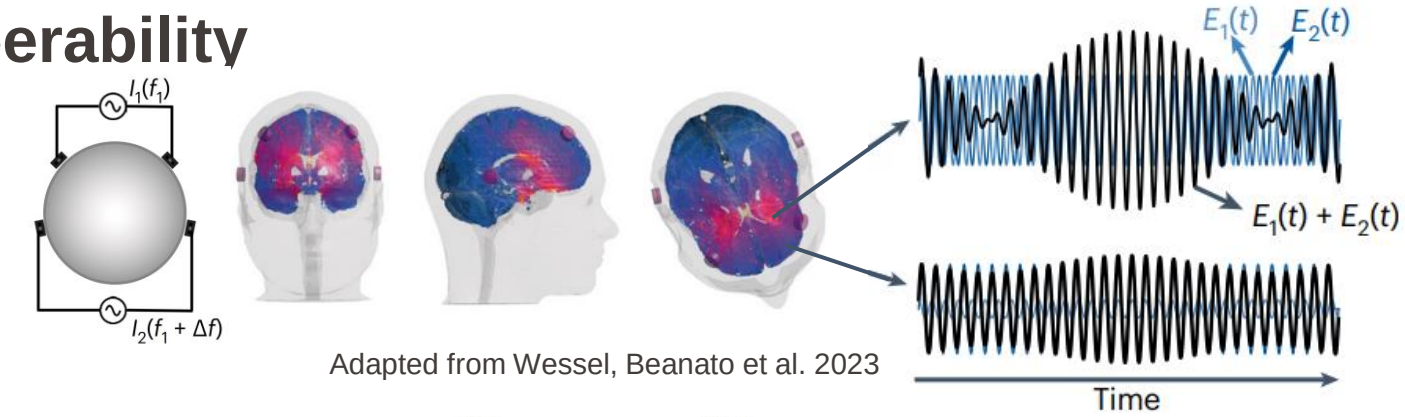


Adapted from Grossmann et al. 2017

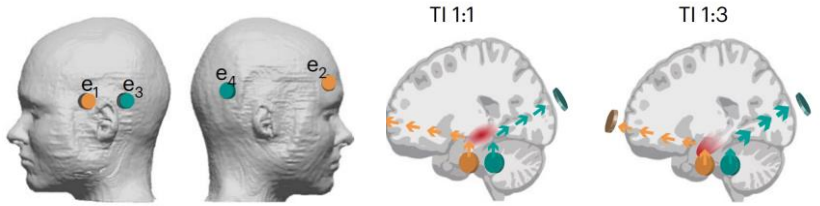
Gregor Dederichs

Temporal Interference

- **Parameters allow steerability**
  - **Striatum**
  - **Hippocampus**



Adapted from Wessel, Beanato et al. 2023



Adapted from Violante et al. 2023

- **Implications for Neurological Diseases**
  - **Alzheimer's** disease
  - **Parkinson's** disease
  - **Stroke**
  - …

- **Controlled through a Graphical User Interface (GUI)**

User Interface Limitations

- **Voltage Discharge**
  - Occurs when using a trigger
  - Can be **pain**-causing for patients
  - **Impractical** for user – Prevents safe termination of stimulations
- **No flexibility**
  - Parameters are fixed
  - **No parameter update** possible during stimulations
- **Not open source**
  - **Decreased accessibility** and sharing
- **Impractical Blinding**
  - **Unblinding** risks and **decreased usability**

Explored Approaches

DAQ Theory and MATLAB Troubleshooting

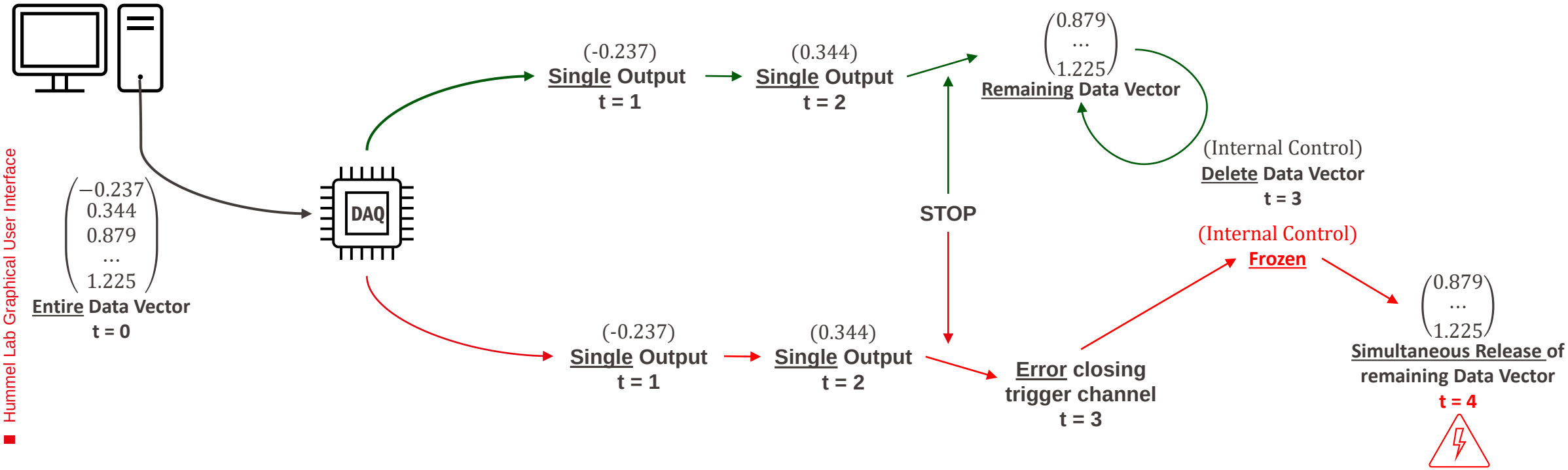
NI-MAX Software

Alternative packages for Python

Gregor Dederichs

MATLAB Troubleshooting

- **Voltage Discharge – Probable Source**
  - Entire signal is stored in DAQ
  - Stop → Simultaneous release
    - Without triggers: Internal control of DAQ → Spike Prevented
    - **With triggers**: Error when closing trigger channels
      DAQ freezes: Internal control lost → **Voltage Spike**



Gregor Dederichs

NI-MAX Software

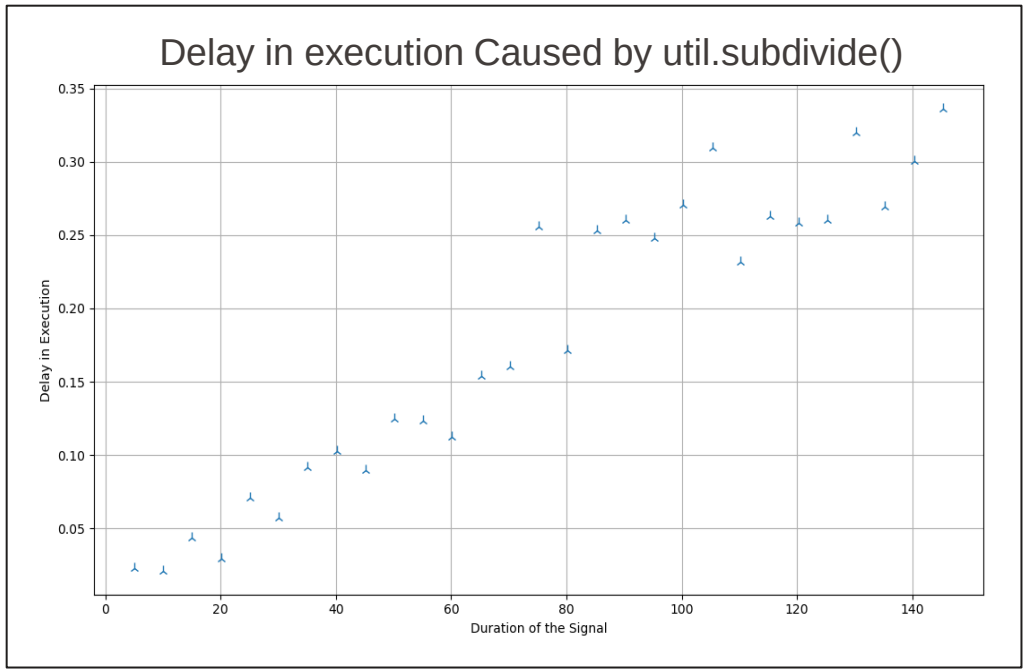
- **Free software by National Instruments**
- **Device Test Panel**
  - Allows testing of simple features
  - Improper trigger setup → **Voltage Spike**
- **Virtual Testing**
  - Through NI-MAX
  - Virtual devices – Rapidly test certain features without a complete setup
- **Device Setup**
  - Device **specifications**
  - Device **status** checks
  - Device **configuration** – Renaming devices

Alternative Python Package

- **Package: nidaqmx**
  - API for interacting with the NI-DAQmx driver
  - Open source → Increased **accessibility**

- **Limitations**
  - API **created generally** for all NI DAQs
  - **Not all DAQs are compatible** with all functionalities

- Notably **limited on current DAQs**
  - Attempts to hard-code functionalities **failed**
    - Example: **No timing** configuration on **USB-6216**
    - Needs explicit specification in Python
    - Idea: Subdivide signal to be able to update it on the fly
    - Problem: **Accumulates delay**



Gregor Dederichs

HummelGUI

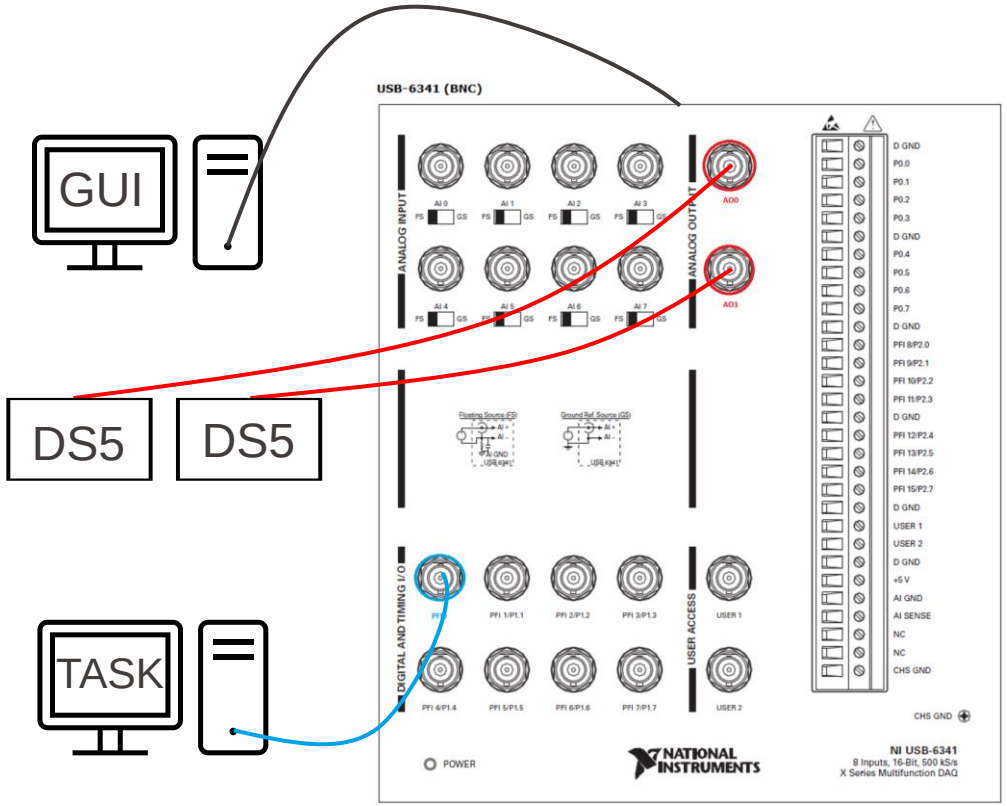
For NI USB-6341

Gregor Dederichs

System Setup

- **Connections**
  - **Identical** to current setup
    - BNC 1: "AO0" to stimulator 1
    - BNC 2: "AO1" to stimulator 2
    - BNC 3: "PFI0" to trigger port
    - USB: DAQ to computer running GUI

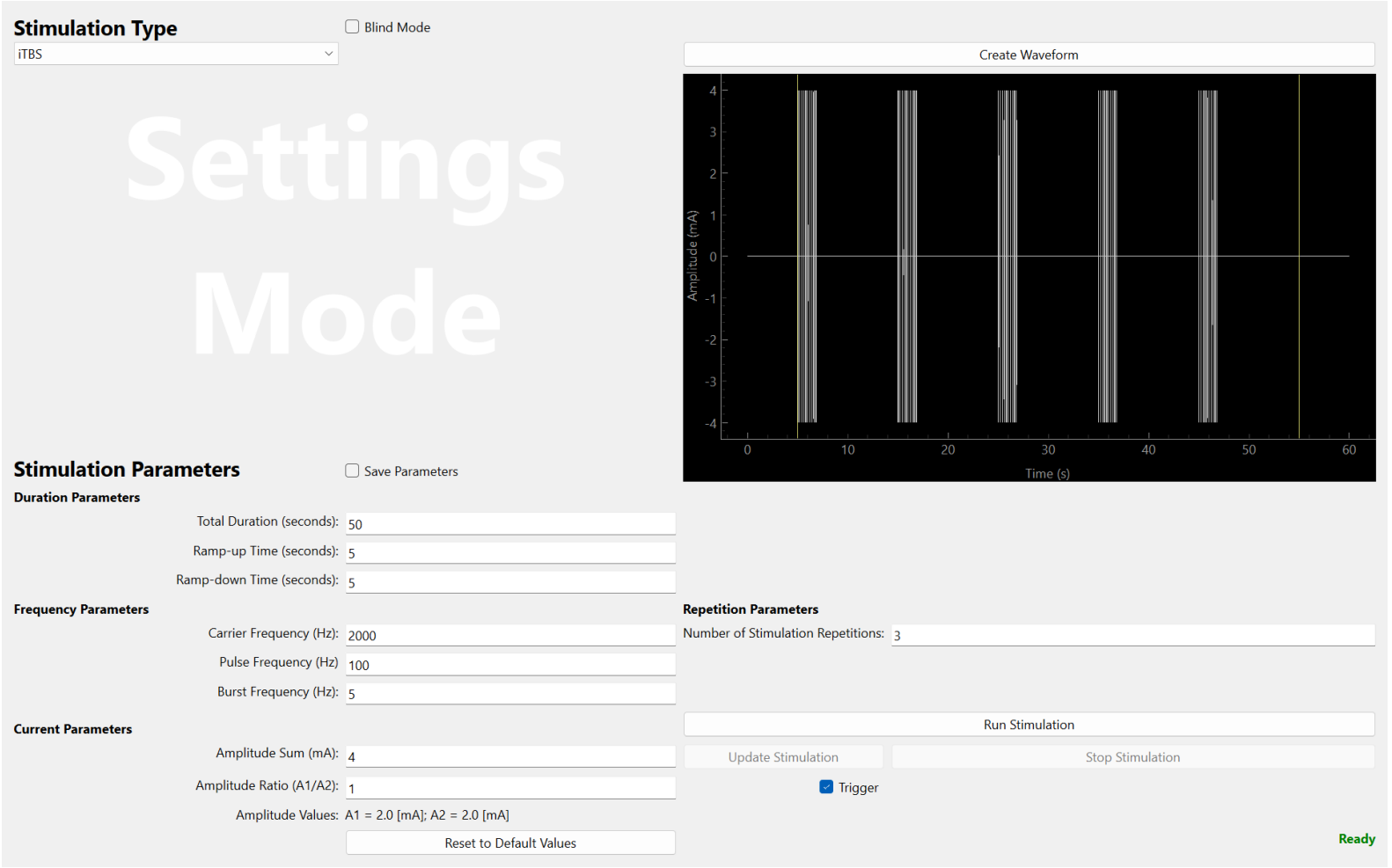
- **Dependencies**
  - **Different** from current setup
  - Python and related **packages**
    - Numpy
    - PyQt6
    - Pyqtgraph
    - Nidaqmx
    - Pandas
    - Matplotlib



Gregor Dederichs

Features and Functionalities – Settings Mode

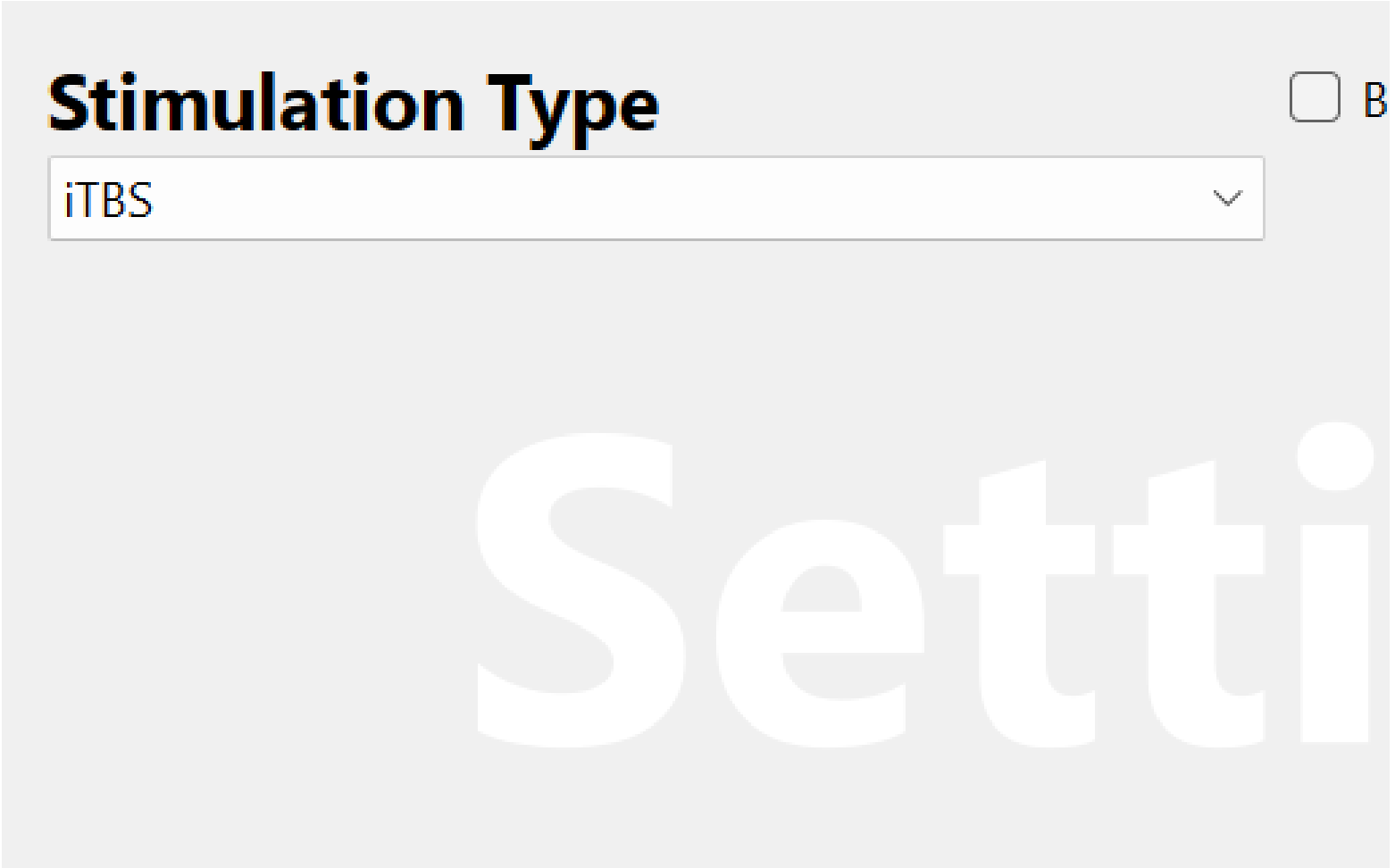
- **Intended Use**
  - Troubleshooting
  - Testing Parameters
  - Checking Functionality
  - Titration



Features and Functionalities – Settings Mode

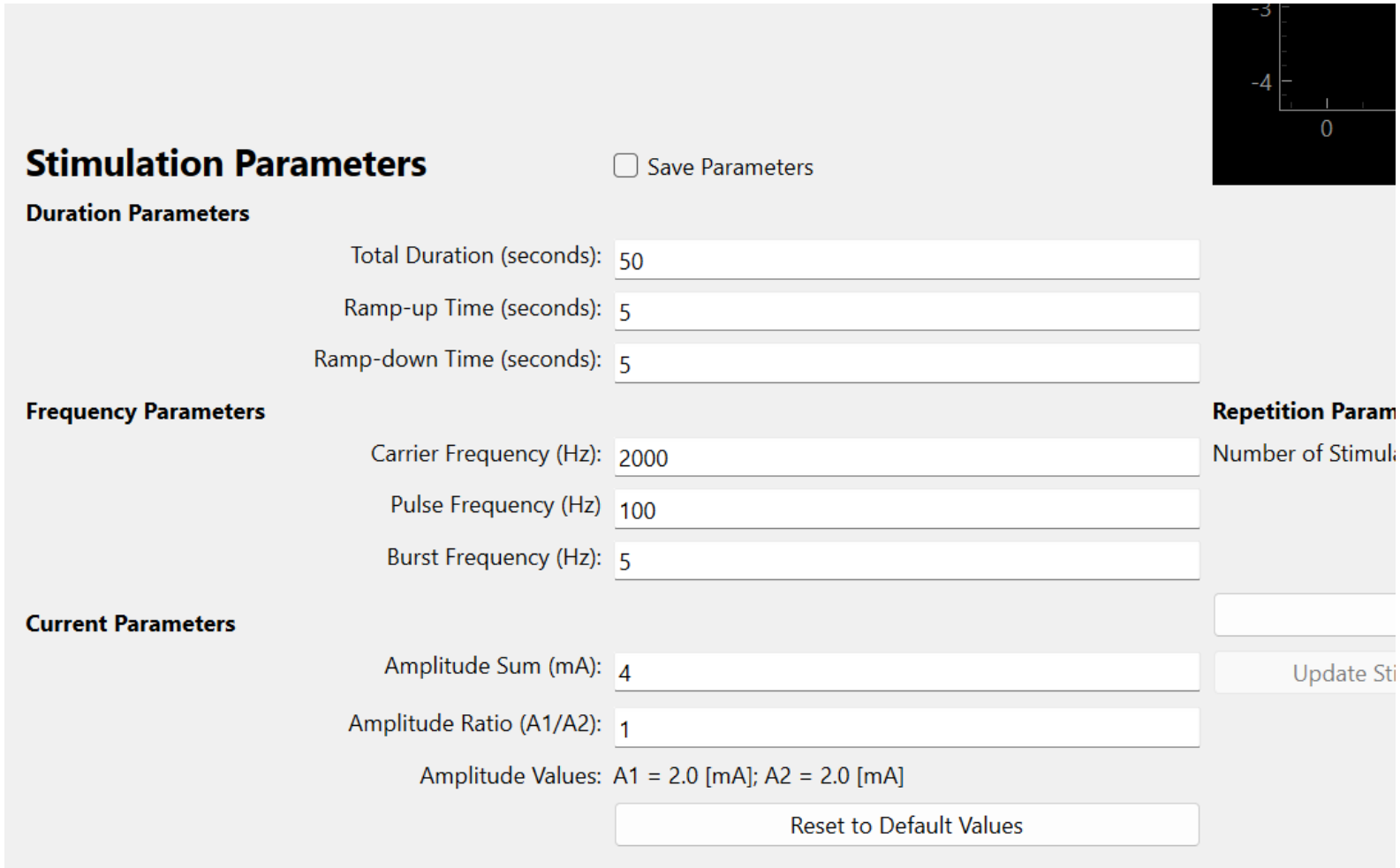
Gregor Dederichs

- **Selecting Stimulation**
  - Drop down menu
    - iTBS
    - cTBS
    - TBS control
    - TI



Features and Functionalities – Settings Mode

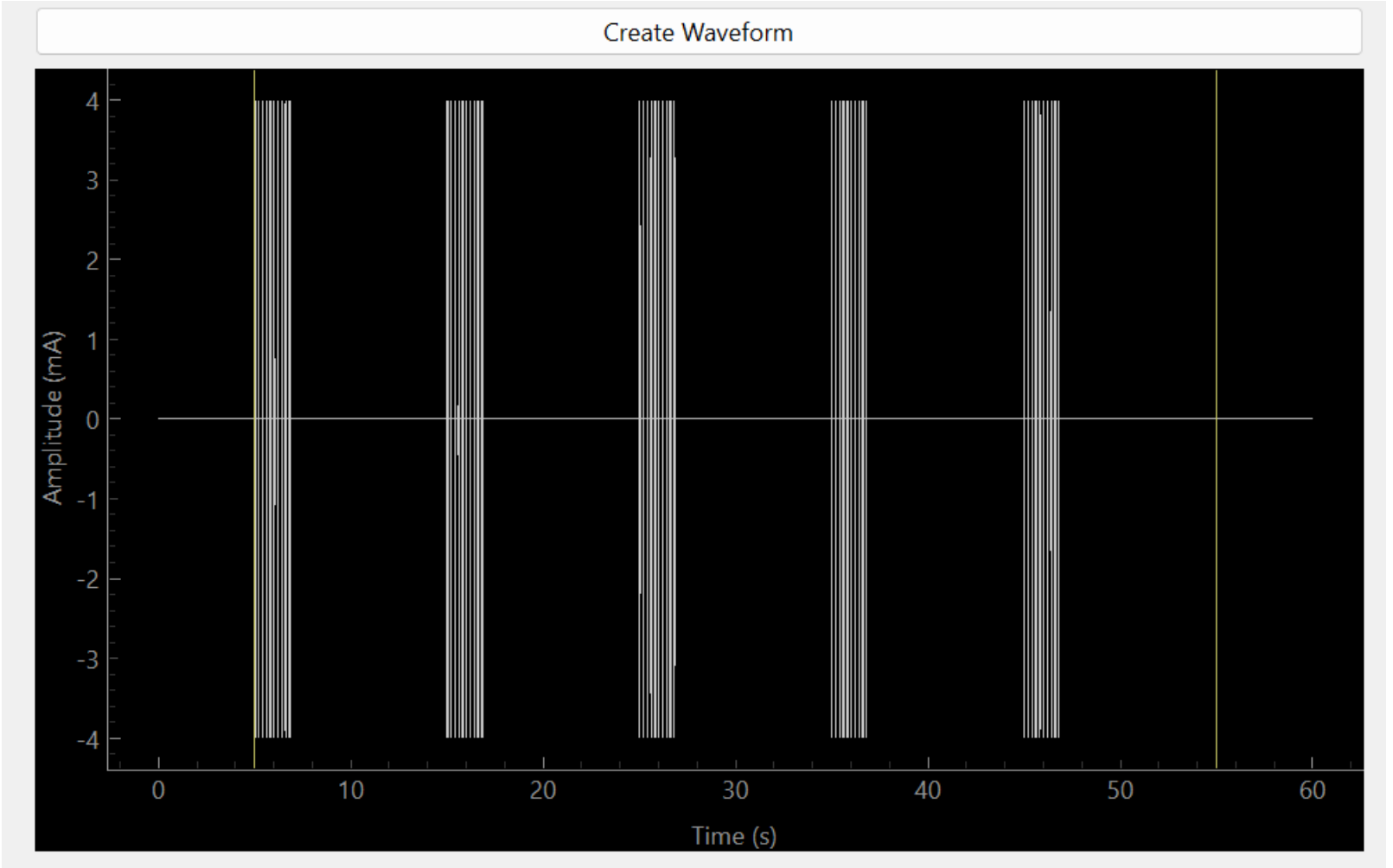
- **Adjusting Parameters**
  - Value Fields
    - Duration
    - Frequency
    - Currents
  - Save Option
    - Avoids unnecessary saves during troubleshooting
  - Reset Button



Gregor Dederichs

Features and Functionalities – Settings Mode

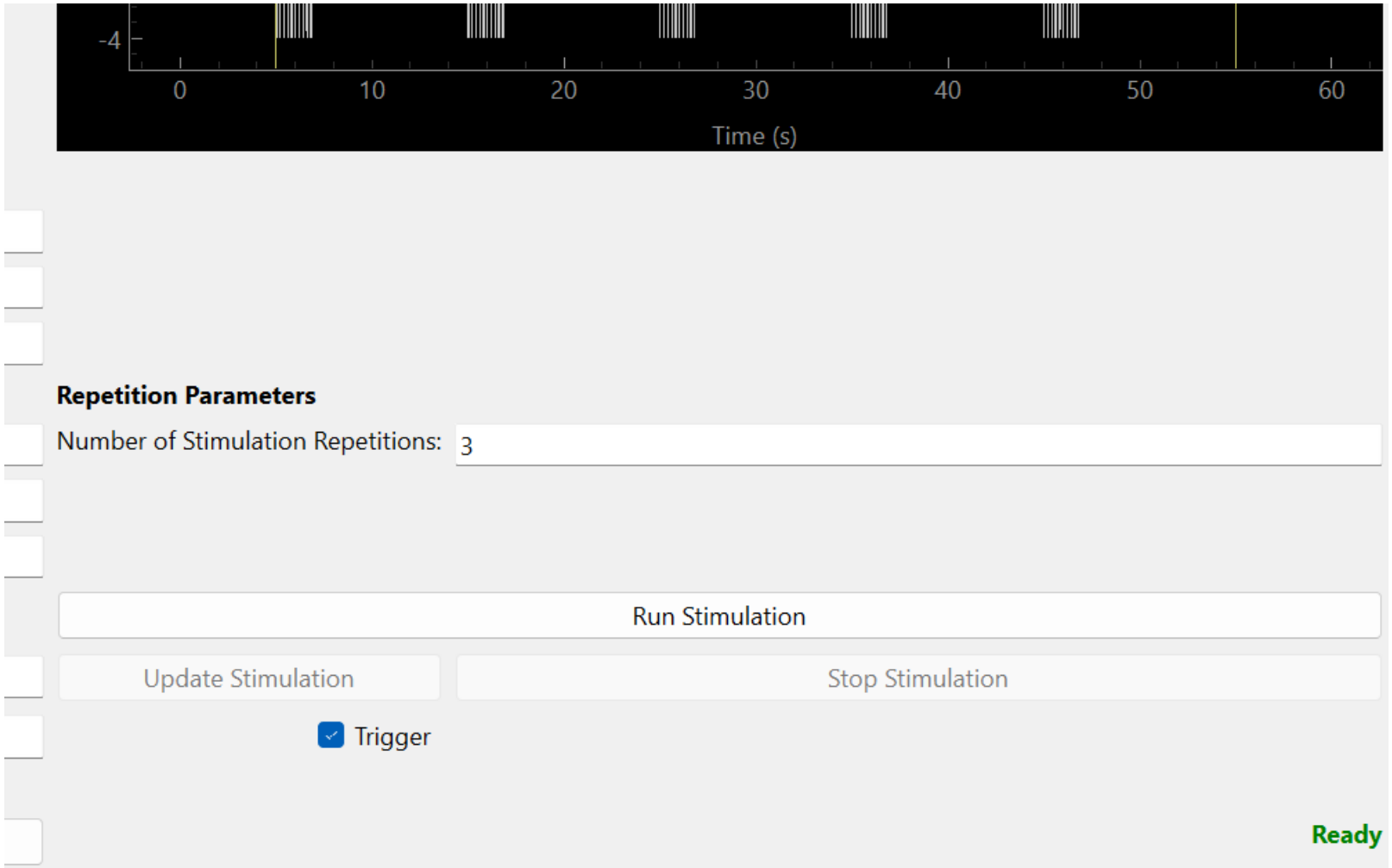
- **Checking Waveform**
  - Created Through a Button
  - **Necessary** before Running Stimulation **in Settings Mode**
  - Yellow lines mark signal between ramp-up and –down
  - Zoom on x-axis only for easy visualisation



Features and Functionalities – Settings Mode

Gregor Dederichs

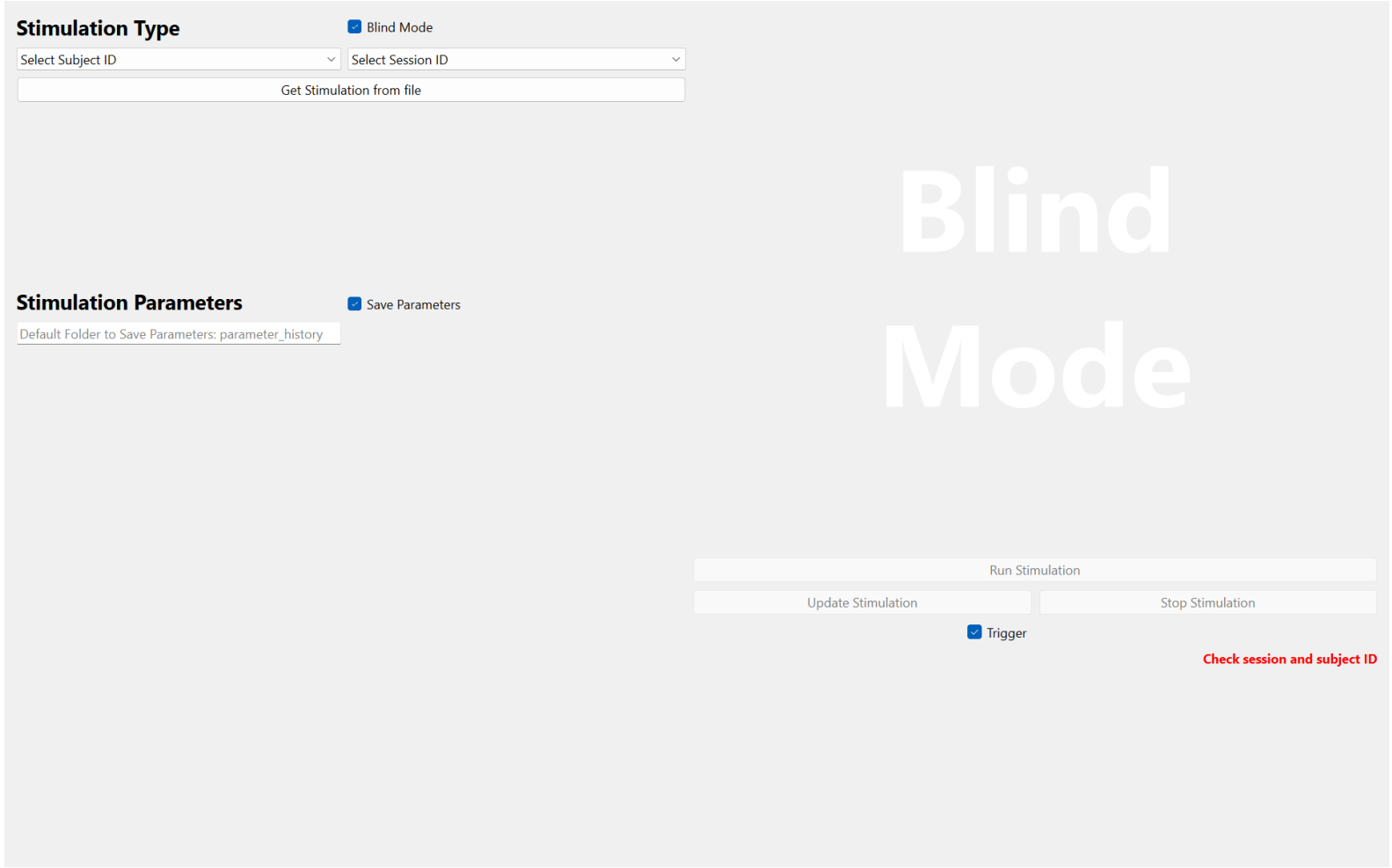
- **Running Stimulation**
  - Choose Number of Repetitions
  - Trigger Option
  - **While Running**
    - **Update** stimulation
      - Does not restart repetition count
    - **Stop** stimulation
      - Instantly updates to a ramp down signal
      - **No Discharge**



Gregor Dederichs

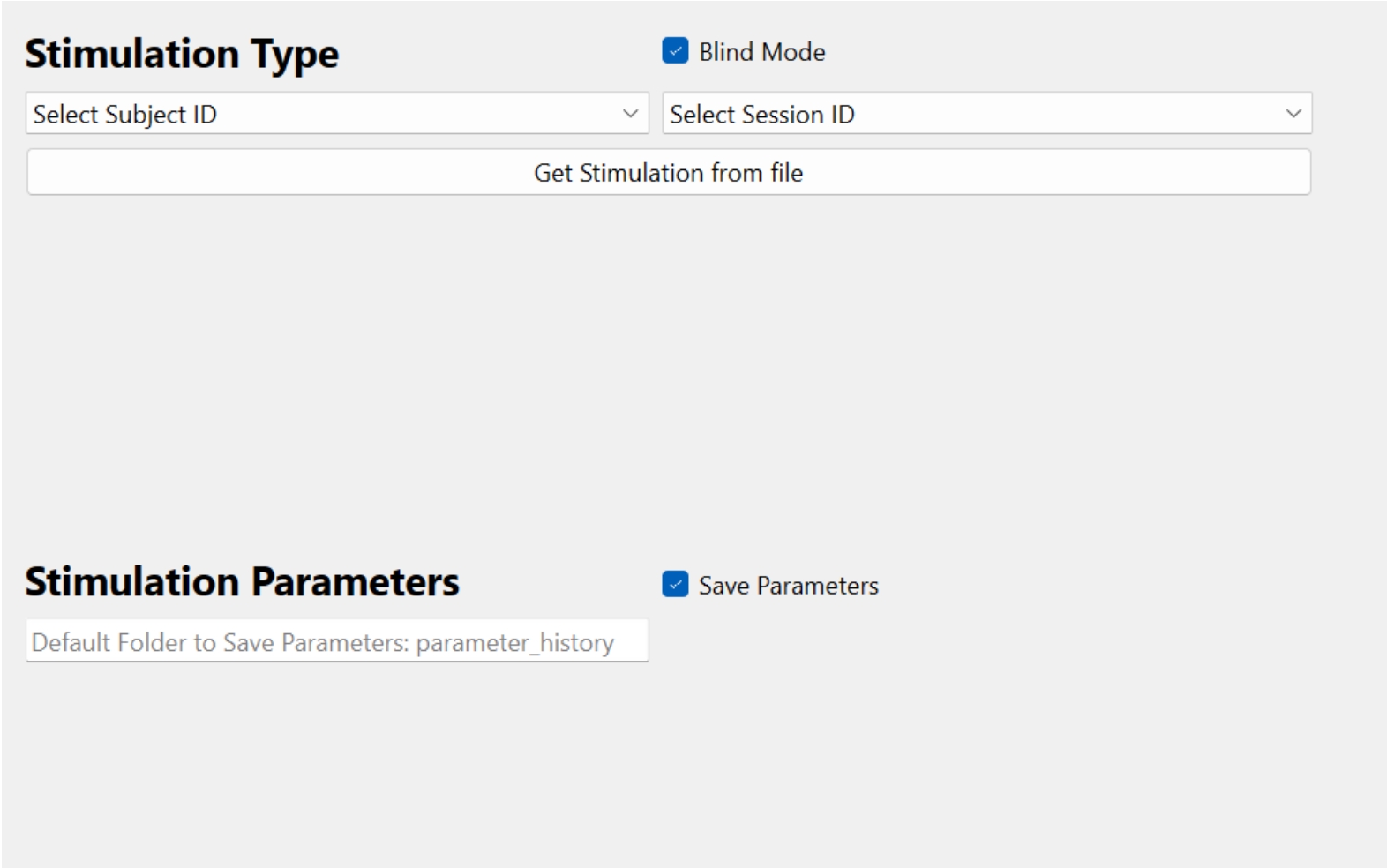
Features and Functionalities – Blind Mode

- **Intended Use**
  - Study Experiments
    - Removes access to parameters and waveforms
    - **No unblinding risk**



Features and Functionalities – Blind Mode

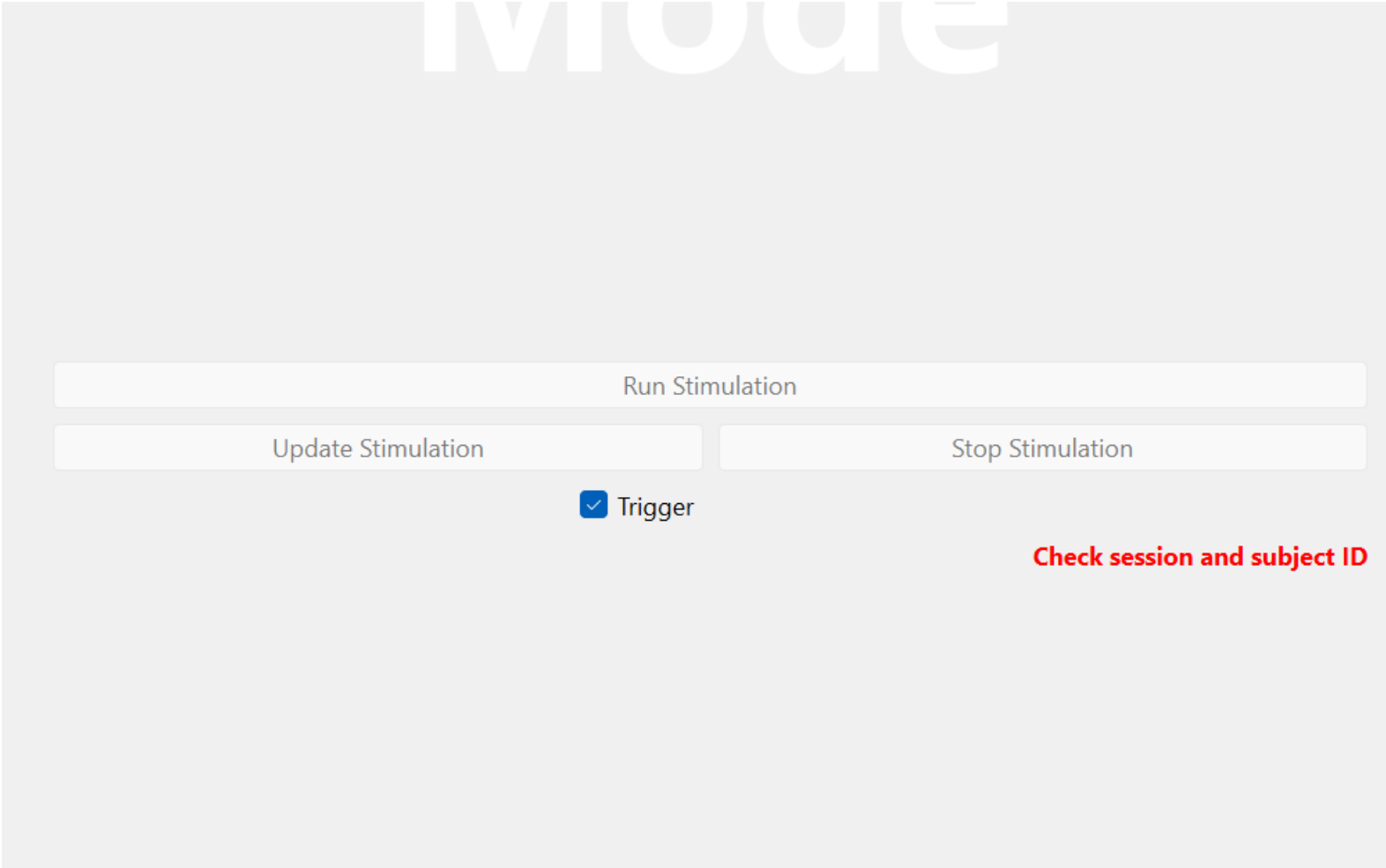
- **Selecting IDs**
  - **Only from Existing** Data in excel file
  - Load stimulation with button – Check Label
- **Saving Options**
  - Parameters are saved to *parameter_history* (directory) – Modifiable



Gregor Dederichs

Features and Functionalities – Blind Mode

- **Running Stimulation**
  - Identical to Settings Mode
  - Trigger option
  - Run, update and Stop options
- **Label**
  - In Blind and Settings Modes
  - Guides User with all important steps (errors, status, …)



Gregor Dederichs

Code Structure – Functional Files

- Implement **GUI function** and **DAQ communication** (stimulation)

File	Function	Modifications
main.py	**Launching GUI**	None
GUI.py	GUI Layout	Adding Widgets
	Waveform Creation	Adding Signals – following previous implementations as template – see also utility files
GUI_worker.py	DAQ Communication	Rare – Technical Functionalities (ex: changing trigger settings)
	Running Tasks	None

Gregor Dederichs

Code Structure – Utility Files

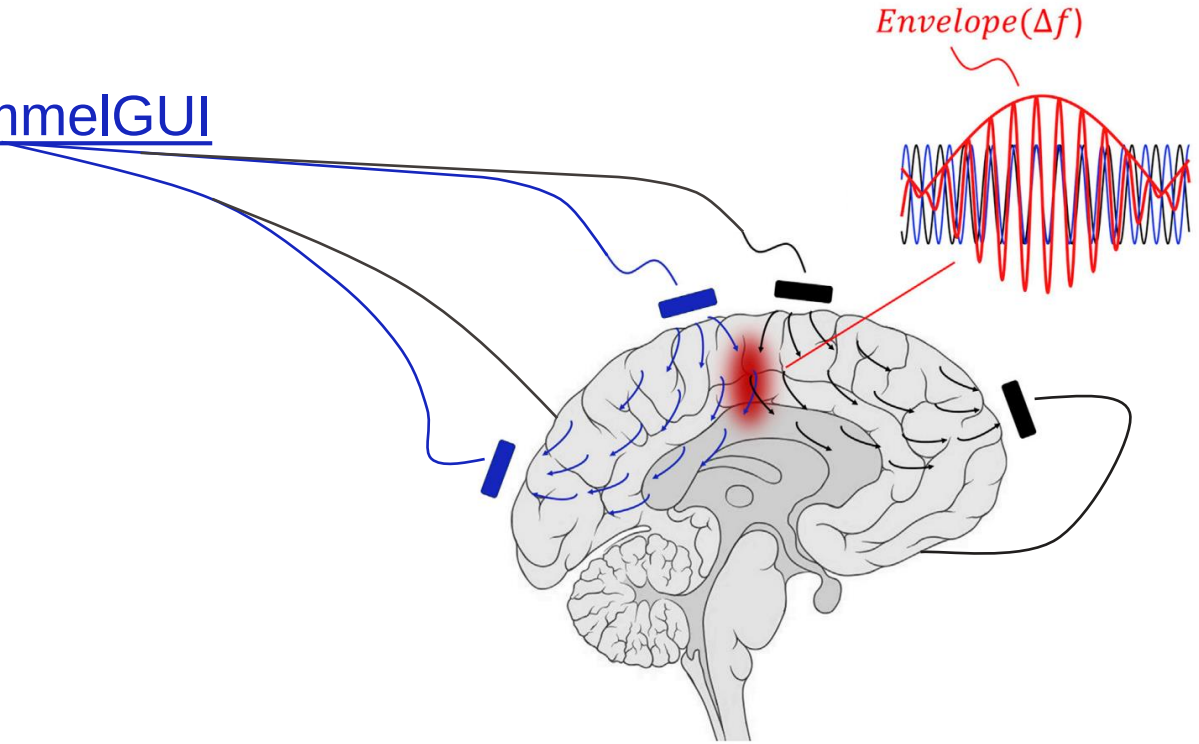
- Implement **Underlying Structure and Utility**

File	Function	Modifications
util.py	**Storing Defaults**	**Setting**/Adding Defaults
fbase.py	Storing Base Functions	Rare
iTBS.py	Creates Signals of Different Stimulation One File per Stimulation	Adding new Stimulations – using previous files as template – see also GUI.py
cTBS.py		
TBS_ctrl.py		
TI.py		

Code Availability

- **Available on GitHub**
  - https://github.com/gdederichs/HummelGUI

  - **Quick Guide**
  - **Installation** Procedure
  - **Setup** Instructions
  - Detailed GUI **Functionalities**
  - File **Descriptions**
  - Future **Modifications** Guide



Thank You