Use of Biofeedback Systems to Uncover Brain-wide Activities During Learning

Jasmine Thuy Jayne Kofmel

Under the direction of Mr. Sunho Chung
Biological Engineering, Jasanoff Lab MIT
and Alan Jasanoff, PhD
Professor of Biological Engineering, Brain
Cognitive Sciences, Nuclear Science
Engineering

Research Science Institute
August 2, 2022

Abstract

This paper describes a novel method of receiving neuronal activity from the motor cortices of rats by stimulating their medial forebrain bundle (MFB) and ventral tegmental area VTA. Through neurobiofeedback we wished to shape the learning method of rats and specifically apply it to the MFB and VTA. The method presented is a combination of photometry and optogenetics. Using a feedback system, the rat can learn to exhibit the brain state that we specify.

Chrimson, an optogenetic sensor used to modify neurons to express light-sensitive ion channels, was delivered to the VTA. GCaMP8f and GCaMP8s, genetically encoded calcium indicators, were delivered to the motor cortices to record fluorescence activity signals via fiber photometry, enabling the investigation of how activity patterns in neuronal networks influence behavior. Through the analyzation of the photometry data the thresholds have been defined at which the MFB is stimulated by light from a laser shining through an optic fiber connected to an optic cannula implanted at the region of interest. Through data sleuthing of the obtained photometry data, the targeted brain activity characteristics were defined for the feedback system. By simultaneously recording and analyzing the motor cortices activity and stimulating the MFB at specific time points, it is possible to train a brain region to present the targeted brain state.

Summary

To be able to train an animal to reproduce a certain behavior, a combination of multiple actions and reactions are necessary. The trainer needs to recognize the behavior and reinforce it at the proper time point with the fitting treat. In accordance with this system, we trained rats through positive reinforcement to present a specified brain state.

The reward was given through stimulation of the ventral tegmental area, an area responsible for dopamine projection, with a laser. The moment of stimulation was determined by the detection and collection of photometry data from the motor cortices, and analysis and determination of the brain states we wished to reinforce. To collect the photometry data we used sensors that respond to light to detect the active neurons while fluorescing.

In the coming years, the test method of combining photometry and optogenetics needs to be refined, and test environment and test procedure can be optimized. In any case, this new method opens new fields of investigations to better understand how the brain functions.

1 Introduction

An underlying goal of neuroscience is to understand how the brain generates thought and action. Millions (in rats) to billions (in humans) of neurons are densely packed and interconnected. They interact with one another and with non-neuronal cell types to execute tasks such as controlling thoughts, memory, emotion, motor skills, and every process that regulates our body [1]. Yet, our understanding of the principles of complex brain functions and cognition remains incomplete because of its intricacy [2].

The brain regions we are interested in for the purpose of studying the learning processes and abilities are the motor cortices and ventral tegmental area (VTA). The motor cortices generate signals to direct the movement of the body. They also play a role in learning and cognition [3]. The VTA plays a significant role in reward, motivation, cognition, and aversion [4]. It sends projections through the medial forebrain bundle (MFB) to multiple regions of the brain, like the motor cortices.

Through performing and analyzing real-time rodent neurobiofeedback experiments, in which rats perform self-stimulation tasks, we observed the motor cortex activity with fiber photometry. Neurobiofeedback can be created by providing stimulation to the medial forebrain bundle (MFB) during photometry sessions [5]. In our experiments, the feedback signal is created through a fiber-coupled light source, where excitatory light is delivered to the region of interest via an optical fiber [6]. Previously, electrodes were used to measure and stimulate the electrical activity of the brain. However, with electrodes, we can stimulate general regions, which only lets us control at what time-point the neurons are activated but not what specific cell type.

This results in the stimulation of all the neurons that are close to the stimulation point [7]. To address this, we implemented the novel optogenetic stimulation technique, enabling more specificity. The inserted optical fiber, which is made of silica glass, creates less disturbance

around the implant-site.

1.1 Photometry

Fiber photometry, which we implemented at the motor cortices, provides the most sensitive and easiest way to record cell-type specific neuronal activities of the deep brain structures [7]. It can be used to explore the regulatory mechanisms behind animal behavior.

The fiber photometry system records changes in the fluorescence intensity of neurons in a specific brain area, which reflects neuronal population activity. When genetically encoded calcium indicators are stimulated with light, they produce fluorescence signals that can be measured using an implanted optic fibers [8].

In neurons, calcium ions regulate several important processes; for example, electrical activity is accompanied by an influx of Ca^{2+} ions [9]. GCaMP, a genetically encoded calcium indicator, can provide critical insight into calcium dynamics, allow the measurement of the activity of hundreds of individual neurons and genetically defined neuronal subclasses, as GCaMP fluoresces only when bound to calcium ions [10]. Based on similar principles, in fiber photometry, data are collected by analyzing the fluorescence change (F) relative to the initial baseline fluorescence (F) and observing the signal change corresponding to the calcium transient (F/F) [7].

The fluorescent signals of GCaMP return through the optical fiber and are detected. While recording the activity of the motor cortices we simultaneously stimulate the MFB using optogenetics.

1.2 Optogenetics

Optogenetics helps us understand the function of individual circuits and how they relate to different aspects of our behavior [11]. It is the combination of genetic and optical meth-

ods that allows a precise, targeted, and fast control of neuron activity. We can selectively stimulate which neurons we want to activate, specified either by location or by type.

Using optogenetics, we can investigate what roles different sets of neurons play within neural circuits by using light to activate some neurons and record the response of the others. It is possible to tune the neural activity by controlling the power and frequency of the light input. To be controlled by light, neurons are modified to express light-sensitive ion channels. Viral vectors were used to introduce the channelrhodopsin Chrimson into the VTA, allowing these neurons to make proteins called opsins, which in this case can be activated with red light.

When the principles of optogenetics are targeted to specific neurons in the brain, it enables us to drive or silence the neuronal activity by light [12]. This can help reveal how different sets of neurons contribute to behavioral functions of the brain [13].

We utilized the above-described input-output mechanisms in a closed-loop behavior system to probe questions about what kind of brain states are exhibited during the conditioned stimulus, unconditioned stimulus, and their sequential pairing processes. Our short-term goal is to establish small animal fMRI biofeedback paradigms.

This is important for understanding how certain behaviors are produced, and to understand what happens if the brain is damaged in a particular area for investigating diseases that disrupt communication between neurons in a particular area.

To achieve this aim, we performed fiber photometry recording experiments using a fiber photometry device to record the fluorescence signals in the primary motor cortices associated with GCaMP8f and GCaMP8s, while simultaneously stimulating the rat via a fiber-coupled light source that is under the control of a computer.

2 Methods

2.1 Animals

Male Long Evans rats were used to create the biofeedback models. They were housed in pairs until after the surgery to ensure the preservation of the implants. Following the surgical procedure, they were individually housed.

2.2 Surgical procedure

Under isofluorane anesthesia, the rats underwent a standard procedure for the injection of viral vectors which carry the genetic information for the production of a protein.

Viral vectors are tools commonly used by molecular biologists to deliver genetic material into cells. This process can be performed inside a living organism or in a cell culture. Viruses have evolved specialized molecular mechanisms to efficiently transport their genomes inside the cells they infect [14].

In this matter fluorescent calcium sensors (GCaMP8f and GCaMP8s) or optogenetic sensors (Chrimson) are delivered to two key regions in the brain, the primary motor cortices and the ventral tegmental area (VTA).

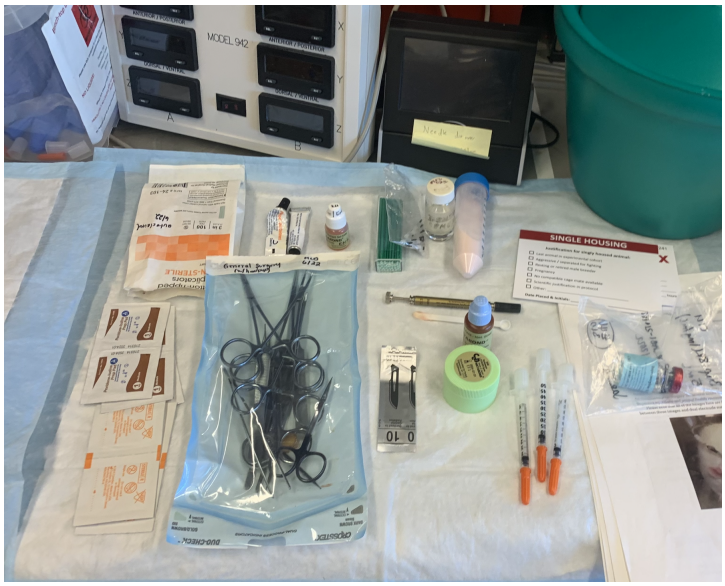

Figure 1: Surgical material

We assembled the following surgical supplies: cotton-tipped applicators, ethanol wipes, betadine wipes, eye ointment, adjustable applicator, pink cement solvent, pink cement powder, ceramic dish, metabond powder, metabond catalyst, metabond scoop, metabond solvent, drill, cauterizer, set screws, headpost, saline, viral vectors, optical cannula.

2.2.1 Viral injection

Firstly, the rat is anesthetized and its hair is shaved from the head for a cleaner incision area. Secondly, properly insert the ear bar so that the skull is placed securely. We placed eye ointment on the eyes to ensure that they do not dry out since the rat will not blink anymore. After cleaning the incision area with Betadine and ethanol we started the incision between the eyes and ended between the ears, subsequently removing the blood and fascia. To zero the small drillbit as accurately as possible we placed it over the Bregma. Next, we moved the drill to the desired injection position and ensured that the drill has penetrated the skull.

We injected AAV viral vectors that deliver fluorescent calcium sensors to the regions of

interest: GCaMP in the left and right motor cortices, and the optogenetic sensor Chrimson (used to provide electrical stimulus to the brain) to the VTA.

Chrimson is a highly light-sensitive and fast opsin, a protein group that was made light-sensitive and can be illuminated through a laser. Chrimson can evoke spikes with submillisecond precision and repeated firing up to 100 Hz [15]. They typically allow the fast depolarization of neurons upon exposure to light through direct stimulation of ion channels. [10] We implanted the fiber optic cannulae at the motor cortices and the medial forebrain bundle (MFB). The implantation site at the MFB was chosen for the reason of it being a rather large, accessible axon bundle which connects the VTA with the nucleus accumbens.

To increase the stability of the mounted head cap we scuffed the skull. The head cap is used to secure the rat's brain during photometry sessions. By pouring a metabond mixture (1 scoop powder, 4 drops solvent, 1 drop catalyst) and pink cement (no exact ratio) on the skull we secured it in place. It is important not to leave edges for the animal to scratch. Once we finished the surgical procedure, we turned off the isoflurane and removed the ear bars. Lastly, we filled out the surgery card.

2.3 Optogenetic system

To determine which rats produce the expected behavior upon laser-stimulation we had them conduct a lever activation task: The successful activation of a lever seen in Figure: 2 triggers the laser-stimulation of the VTA resulting in the release of dopaminergic projections to forebrain regions [16]. We used an available box and attached a lever press to the bottom middle of a wall, ensuring that it was reachable. During testing, the rats were connected to a fiber optic cable through a rotary optic joint to a class 1 laser, which was used to deliver red light once the lever is pressed. We constructed an alternative fiber optic cable that was longer and more flexible, yet less protected from being chewed or knotted up than the bought cable. We would use this cable only in case the bought one was restricting the rat's natural

behavior. Multiple pieces for the box and laser alignment were designed in Autodesk Fusion 360, optimized and 3D printed.

The intensity of the laser or the pulse width/frequency was varied depending on the behavior of the rat after stimulation.

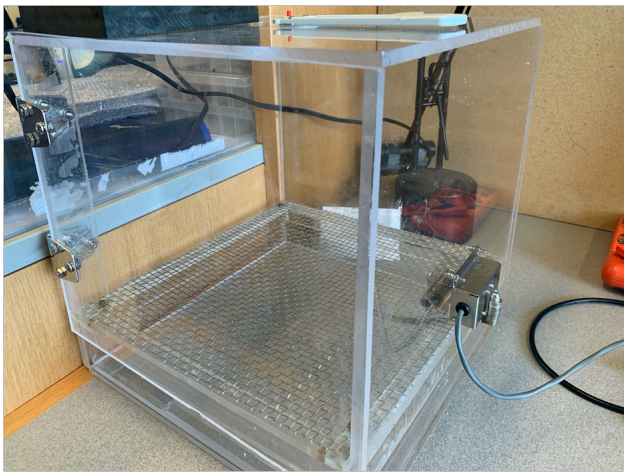

Figure 2: Optogenetics system test

We conducted multiple 15-minute rounds to test the optogenetic system and detect if the rat expresses the expected behavior. For the first round, the laser had the following settings: frequency 1: 80Hz; frequency 2: 0.1Hz for the duration of the stimulation continuing one second after the lever press. The laser intensity, pulse width, and pulse frequency are easily modulated for further behavioral tests with appropriate adjustments.

2.4 Photometry acclimation

After the identification of the rats that produce the expected lever-press behavior the screened animal was prepared for a no feedback pre-scan to get acclimatized to the environment of the enclosed testing facilities and fMRI. The rat underwent three acclimation phases. With the help of these acclimations, it is possible to test if the viral injection and optic cannulae placement in the motor cortices were done properly. If they were not properly inserted, the neural spikes and thus the activity would not be clearly detectable.

Some photometry data was collected but no stimulation tests were run. Once these are concluded, the feedback system experiments with laser stimulation can be executed.

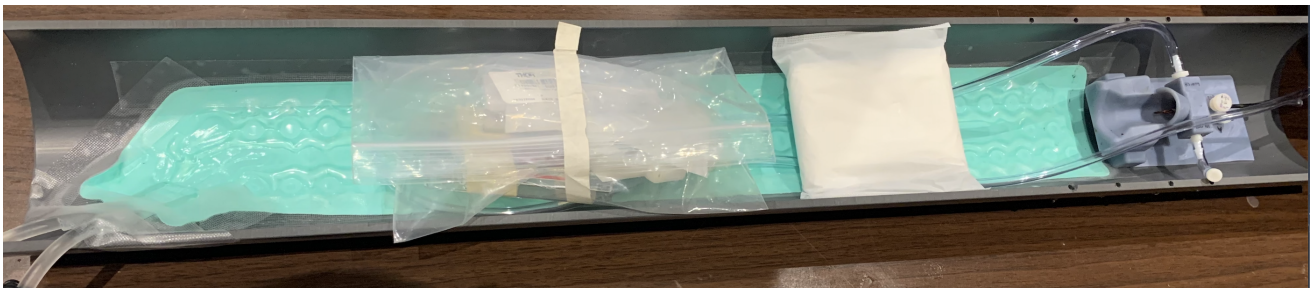

Figure 3: Rat cradle for photometry session

The rat was then anesthetized and placed on a heating pad in its cradle. The head mount was securely fastened to the cradle and two fibers are connected to the motor cortices' optical cannulae to receive photometry data.

In addition, the received datasets can be used to determine the future brain states with which we will create the feedback activation.

2.5 Data sleuthing

We sorted through previously recorded data sets to identify patterns, and decided what we will use for feedback schemes. The code we wrote in Python firstly standardizes the collected data from the acclimations, and convolves it with a sine function within the range of 2Hz to 20Hz in intervals of two. This step is needed to determine which brainwaves we wish to focus on. Depending on our choice, we define the threshold for 120-180 events per hour, which would trigger the stimulation of the VTA.

There are multiple variables that need to be fixed in order to have a uniform technique of collecting data, for example, the needed frequency (Hz) and thresholds which are defined after the data sleuthing. This can be adjusted to the specific rats we are imaging, depending on the first acclimation sessions and the amplitude of photometry data collected.

2.6 Feedback system

In closed-loop feedback scans, as shown in Figure: 4, the behavior-reinforcing current level is measured.

The "Conditioned Checker" is the already defined threshold that triggers the stimulation. This Condition Checker is applied to the data received from the behavioral output, collected using photometry. The incoming data is in real time being checked if it exceeds the threshold. If the altitude reaches/exceeds the threshold the laser is activated to stimulate the VTA. The data is collected every 0.3 seconds.

The data is recorded, so it can be additionally analyzed after the recording session on a computer, separately from the recording session. After the detection of a specified condition, the "Reward" is activated. This entails the laser stimulating the MFB for one second. This reward shall induce a greater abundance of the condition which we stimulated: the Behavioral Output. In addition to the specialized optical imaging, the rat was simultaneously placed in

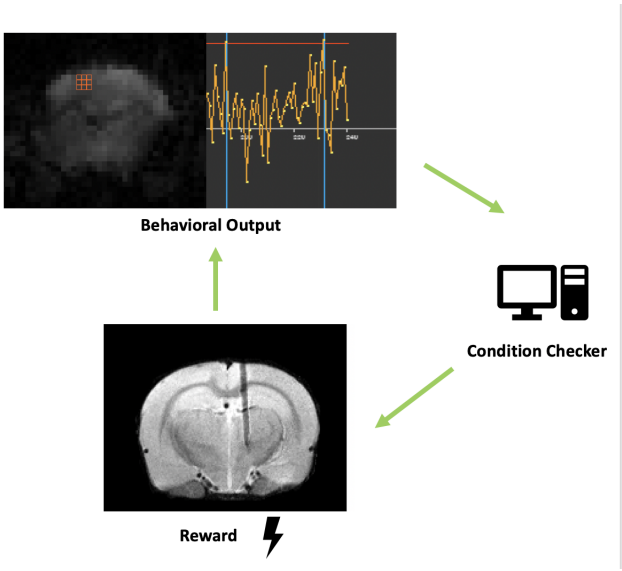

Figure 4: Experimental Setup

an fMRI scanner to be able to image the whole brain activity. Functional magnetic resonance imaging (fMRI) measures brain activity in a non-invasive way by detecting changes associated with blood flow. When an area of the brain is activated, blood flow to that region increases [17].

The addition of fMRI scanning to optical imaging helps recognize the influence of stimulation on the whole brain. Seeing the whole brain activity also helps determine the importance and function of specific brain regions by recognizing the connected pathways and the dependencies on the specific areas.

3 Results

3.1 Optogenetics behaviour box

We successfully constructed the behavior box by 3D printing and assembling components. The rat could reach the position of the lever and trigger it easily. The 3D printed laser mount helped the alignment of the focal point and the optic fiber, which led to clear and targeted laser stimulation.

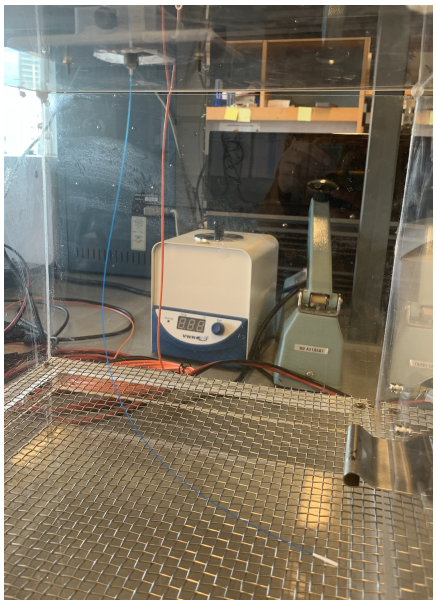

Figure 5: Optogenetics system test

In Figure: 5 the optogenetic testing box is depicted. The blue optic fiber and lever are recognizable on the right side of the box.

The behavior box was tested on two rats:: JuneRat1 and JuneRat2
JuneRat1 preferred one corner of the box resulting in the lever hardly being pressed and subsequently no stimulation to the MFB being initiated. JuneRat2 was more exploratory and pressed the lever on multiple occasions. The experiment would have to be conducted for

a longer period than 15 minutes to be conclusive about the stimulation reinforcing the lever press behavior.

3.2 Reward criterion

A promising reward criterion is defining a threshold that the photometry data waves are required to exceed in order for the stimulation of the MFB to be triggered. The thresholds that need to be exceeded have to be defined for each channel (right and left motor cortices). The amount of times the data exceeds the threshold is counted to ensure that the rat gets stimulated between 30-200 times per hour, ensuring that the rat is not over/under-stimulated. The threshold is defined for frequencies from 2 Hz until 20 Hz in intervals of two. By analyzing the signal convolved to multiple hertz, we can focus on different brain wavelengths and their corresponding functions. Multiple independent variables were tested to find the best prerequisites for the photometry feedback experiment. After comparing multiple frequencies we decided to focus on 8Hz amplitudes since in the collected photometry session data the greatest amount of 8Hz signal was identified. 8Hz brain wavelength corresponds to Theta brainwaves which occur during voluntary movement and may have some role in encoding spatial information [18].

In addition, the thresholds for Channel0 and Channel1 were determined for data sets one and two. Data set one; Channel0: 2870-3300, Channel1: 2500-2690; data set two; Channel0: 4240-4700, Channel1: 2900-3260.

If during a feedback experiment the signal of the region of interest exceeded the threshold, the MFB was stimulated. This results in coupled feedback.

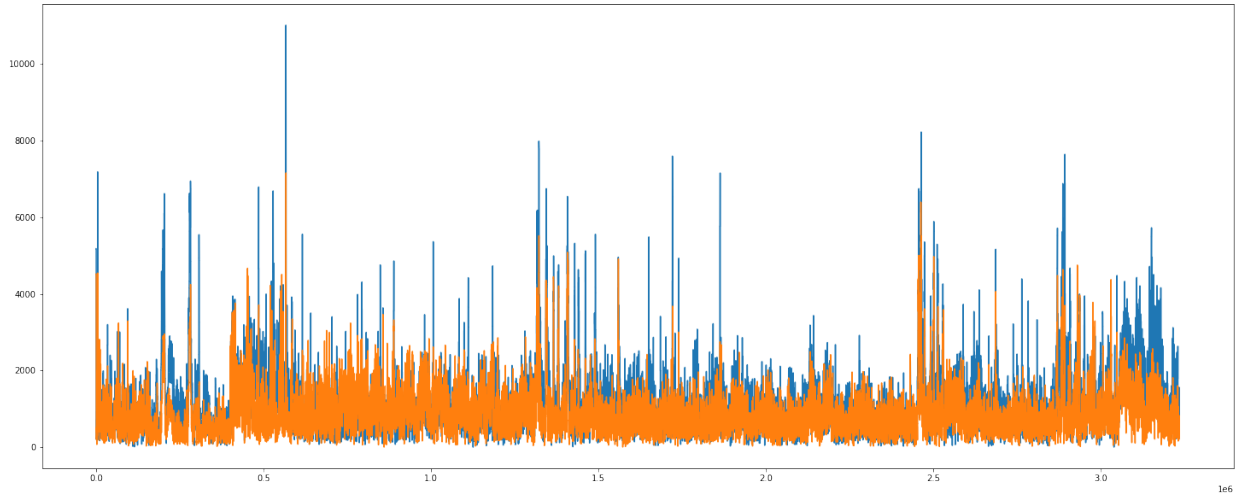

Figure 6: Data from 1h photometry Session convolved with 8Hz sine function; orange: photometry data from the left motor cortex, blue: photometry data from the right motor cortex

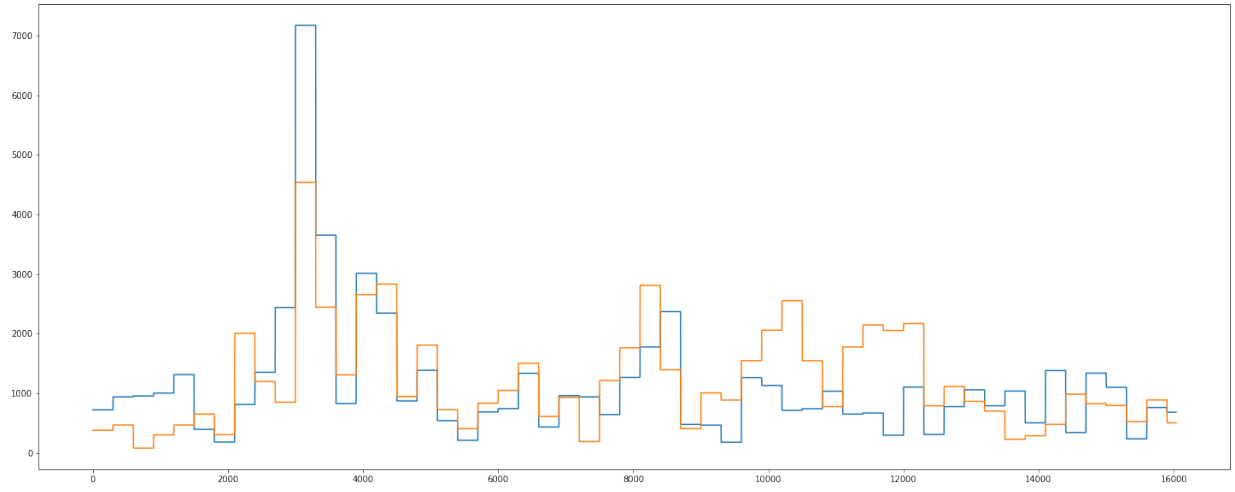

Figure 7: 15seconds from 1h photometry session; orange: photometry data from the left motor cortex, blue: photometry data from the right motor cortex

3.3 Photometry Acclimation

The first acclimation obtained reliable photometry data, meaning the injection and placement of GCaMP8f and GCaMP8s, and the optic cannulae were successful. These rats will need to undergo two more successful acclimations. Once these are concluded the feedback system experiments with laser stimulation can be executed.

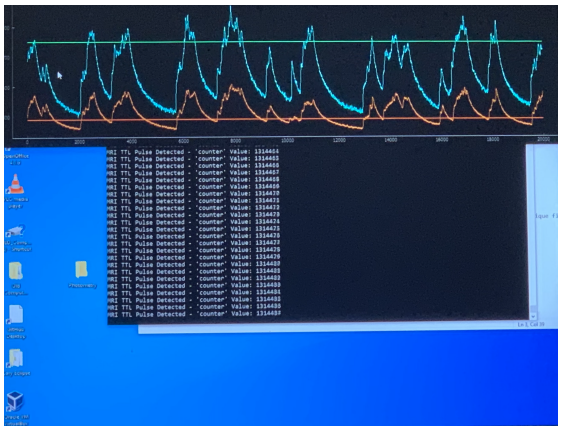

Figure 8: Acclimation1 photometry session

In Figure: 8 the collected photometry data as the rat is coming out of anesthesia is depicted.

4 Discussion

The optogenetic setup was a success, yet during the optogenetic testing we encountered two uncertainties: We tested if the rats can learn through the optogenetic positive stimulation, and if the surgery was successful. Since there were two uncertainties it is unclear which might be the reason if something doesn't go as planned: Through the test, we needed to ensure that the reward is stimulated strong enough by applying the optimal activation time point and frequency of the laser, and if the optical cannulae are placed correctly and reach

the region of interest.

The experiments with JuneRat1 and JuneRat2 displayed a rare amount of lever-presses, and hardly any correlation between the stimulation and further lever-presses. The expected amount of lever presses was 20 per minute. We only achieved around 1 lever press per minute. This could lie in the setup: Rats don't usually enjoy the metal net as a floor. In addition, we noticed flinching from a rat when the lever was pressed in response to the slight clicking it emits.

Possibilities to improve the setup include changing the lever-press to a different position on the wall, for example, closer to a corner, where the rats tend to interact more often, or setting up an optical-interruption-based nosepoke behavioral box instead, which avoids making clicking sounds that may impact the rats' behaviors. To receive the data from the first run we used an optic fiber bought from the store. In response to noticing that it was too stiff and restricted the rats' movement, we used the fiber cable we built, which gave the rat much more freedom to move and interact with its surroundings more naturally. Enlarging our sample size will help us rule out the possibility that these two rats' lack of obvious reinforcement of the lever press behavior simply happened by chance.

At the moment only one laser setting was tested. It would be worthwhile to adjust the laser settings to enhance the learning curve. The laser intensity, pulse width, and pulse frequency can easily be modulated for further behavioral tests, or we can implement a stronger laser.

The tests conducted only stimulated the rat when it fully completed the lever-press. This might be a difficult condition that we could modulate in the future.

For the photometry sessions to work smoother a more stable head structure should be designed to reduce the degree of freedom the rat has in producing motion, and to reduce the chances of implant's structural failure.

5 Future Work

In the near future, we could develop a possibility to give the rat a small amount of rewards when it gets close to the characteristic we wish to be expressed. This would speed up the learning process since it more often meets the necessary conditions and thus receives positive stimulation. The setup of the optogenetics testing and laser setting can be improved through the experience collected. In future studies, we wish to change the regions of interest and therefore eventually conduct a whole brain analysis.

We also wonder about the possibility to train brain regions to take over a function of a different region. After the possibility of human application, we could train the brain to be able to control, for example, prosthetic arms, and relearn activities after a stroke. It could be used to control prosthetics capable of correcting neural circuits that have gone awry in brain disorders. One could also work on the implementation of the feedback system in humans. This would open many possibilities to enhance our brain function. In distant research, one might be able to train the brain to control a prosthetic arm or regain lost brain function.

6 Conclusion

Through the preparation and testing of the optogenetic setup, we have constructed a working optogenetics testing system. Even if the results of the experiment depend on the rats' participation and further tests need to be conducted to ensure that we observe the expected behavior from the stimulation, the construction of the experiment and code works as expected. The feedback code displayed the expected results and is written in a way that allows the laser settings to easily be modulated.

The received photometry signal of the motor cortices was consistent. The fastening of the rats in the photometry cradle needs to be improved to allow the rat a more comfortable imaging session. Once everything comes together and the optogenetic reward stimulation

is implemented with the behavioral photometry output the results will have a significant impact on the way scientists view the ability to learn and train our brain regions to behave in a way we wish them to. This research has a great influence on how we conceive the ability to learn or teach a new task. It is a way to externally communicate with our brain and thus with our body and its functions. One substantial remaining question is if these techniques would also be applicable in humans and how many other implementations one could use it for.

7 Practical Takeaways

Since we don't know everything about the brain's functions, we are generating substantial steps towards uncovering the learning process of the motor cortices and its interaction with the VTA. This research project requires a combination of a variety of skills, which end up being interconnected. It demands a lot of patience and persistence. Even if the experiments do not always go as expected the possible outcome keeps the motivation high.

The feedback system may be modified and implemented in other mammals and humans. There could be future medical applications, like the possibility to retrain functions of a brain that were lost through damage. It would also revolutionize the possibilities of teaching and learning new skills.

8 Acknowledgments

I would like to thank my mentor Sunho Kevin Chung for his advice, support, and guidance throughout these past four weeks and for this amazing opportunity to work on this fascinating project. I am grateful to Alan Jasanoff, our primary investigator, for allowing me to join the team during this period.

A big thank you to Andrew Becker, my alternate mentor, for conducting and commenting on the multiple surgeries, and answering my questions. Also a huge thank you to Jennifer Wen for her assistance, patience, and wonderful explanations.

Thank you to Diane Ballestas for organizing the lab lunch I was allowed to join.

In addition, I would like to thank my tutor Catherine Xue, and the first week teachers assistants Siya Goel and Steven Liu. And Kenneth Choi for being the best counselor we could have asked for.

Of course, I thank the Massachusetts Institute of Technology (MIT) for providing the facilities and equipment, and The Center for Excellence in Education (CEE) for organizing the mentorships and offering this life-changing possibility to conduct research.

I would like to thank "Förderung Begabter Kinder Bern" and Luc Schnell, and the "Fritz-Gerber-Stiftung", and "René Susanne Braginsky Stiftung" for sponsoring my participation at RSI. I am very grateful for the trust in me to take on this challenge, and for giving me an experience of a lifetime.

Last but not least, I am eternally grateful for all the support and encouragement from my peers and family members!

References

[1] A. R. Nectow and E. J. Nestler. Viral tools for neuroscience. *Nature Reviews Neuroscience*, 21(12):669–681, 2020.

[2] D. S. Bassett and O. Sporns. Network neuroscience. *Nature neuroscience*, 20(3):353–364, 2017.

[3] C. Vitrac and M. Benoit-Marand. Monoaminergic modulation of motor cortex function. *Frontiers in neural circuits*, 11:72, 2017.

[4] R. Bourdy and M. Barrot. A new control center for dopaminergic systems: pulling the vta by the tail. *Trends in neurosciences*, 35(11):681–690, 2012.

[5] H. Marzbani, H. R. Marateb, and M. Mansourian. Neurofeedback: a comprehensive review on system design, methodology and clinical applications. *Basic and clinical neuroscience*, 7(2):143, 2016.

[6] A. Schlehahn, S. Fischbach, R. Schmidt, A. Kaganskiy, A. Strittmatter, S. Rodt, T. Heindel, and S. Reitzenstein. A stand-alone fiber-coupled single-photon source. *Scientific reports*, 8(1):1–7, 2018.

[7] Q. Guo, J. Zhou, Q. Feng, R. Lin, H. Gong, Q. Luo, S. Zeng, M. Luo, and L. Fu. Multichannel fiber photometry for population neuronal activity recording. *Biomedical optics express*, 6(10):3919–3931, 2015.

[8] C. A. Bruno, C. O'Brien, S. Bryant, J. I. Mejaes, D. J. Estrin, C. Pizzano, and D. J. Barker. pmat: An open-source software suite for the analysis of fiber photometry data. *Pharmacology Biochemistry and Behavior*, 201:173093, 2021.

[9] W. Gobel and F. Helmchen. In vivo calcium imaging of neural network function. *Physiology*, 22(6):358–365, 2007.

[10] J. E. Hartung and M. S. Gold. Gcamp as an indirect measure of electrical activity in rat trigeminal ganglion neurons. *Cell Calcium*, 89:102225, 2020.

[11] G. D. Luker and K. E. Luker. Optical imaging: current applications and future directions. *Journal of Nuclear Medicine*, 49(1):1–4, 2008.

[12] K. Deisseroth. Optogenetics. *Nature methods*, 8(1):26–29, 2011.

[13] E. S. Boyden. A history of optogenetics: the development of tools for controlling brain circuits with light. *F1000 biology reports*, 3, 2011.

[14] W. Walther and U. Stein. Viral vectors for gene transfer. *Drugs*, 60(2):249–271, 2000.

[15] E. Ronzitti, R. Conti, V. Zampini, D. Tanese, A. J. Foust, N. Klapoetke, E. S. Boyden, E. Papagiakoumou, and V. Emiliani. Submillisecond optogenetic control of neuronal firing with two-photon holographic photoactivation of chronos. *Journal of Neuroscience*, 37(44):10679–10689, 2017.

[16] C. Bouarab, B. Thompson, and A. M. Polter. Vta gaba neurons at the interface of stress and reward. *Frontiers in neural circuits*, 13:78, 2019.

[17] S. Ogawa, R. Menon, S.-G. Kim, and K. Ugurbil. On the characteristics of functional magnetic resonance imaging of the brain. *Annual review of biophysics and biomolecular structure*, 27(1):447–474, 1998.

[18] N. A. Russell, A. Horii, P. F. Smith, C. L. Darlington, and D. K. Bilkey. Lesions of the vestibular system disrupt hippocampal theta rhythm in the rat. *Journal of neurophysiology*, 96(1):4–14, 2006.

A Code

```
1
2 ##### OPTOGENETIC STIMULATION CODE
3 ### WRITTEN BY: Jasmine Kofmel, Jennifer Wen, and Kevin Chung (2022).
4 ## Changelog: Kevin - July 26, 2022 - added comments, changed some minor
     formatting details, and added logging features.
5
6 ### IMPORT FIELD
7 import RPi.GPIO as GPIO, time, math, numpy as np, os.path
8 from datetime import datetime
9
10 ### SPECIFY SUBJECT, EXPERIMENT DETAILS, & SAVE DIRECTORY HERE.
11 subj = "202206_Rat1"# Subject name. e.g.: "2022Feb08Rat2"
12 note = "First_StimulationTest" # Brief experimental description & quirks (
     e.g. 'NoDelayNoCooldown').
13 fileSaveDirectory = "/home/pi/Desktop/OptogeneticFeedbackBox/Data/"
14
15 ### GPIO SETUP
16 GPIO.setwarnings(False) # Ignore all GPIO-related warnings.
17 GPIO.setmode(GPIO.BOARD)
18 light_pin = 32 # Optical stimulation output pin number.
19 lever_pin = 38 # Lever press input pin number.
20 GPIO.setup(light_pin, GPIO.OUT)
21 GPIO.output(light_pin, False) # Set default mode of operation before
     feedback to the off state.
22 GPIO.setup(lever_pin, GPIO.IN)
23
24 ### STIMULATION SETUP
25 # Intensity modulation frequency (PWM frequency)
26 frequency1 = 80 # Units in hertz.
27 period1=1.0/frequency1 # PWM period.
28 # Stimulation train frequency (equivalent to pulse frequency in electrical
      stimulations). Units in seconds.
29 frequency2 = 0.1 # Units in hertz.
30 period2=1.0/frequency2 # Duration of 1 block of the stimulation train + 1
     inter-block-interval. Units in seconds.
31 # Duration of pulse train
32 stim_duration = 1 # Duration of the entire stimulation train. Units in
     seconds.
33 # Duration of the entire experiment.
34 experiment_duration = 15*60 # Units in seconds.
35 # Frequency of logging data acquisition. Do not worry about aliasing with
     PWM frequency or train frequency (unless you want to use the data about
      PWM state directly), as the update of log is NOT blocked with respect
     to the entire duration of the 1/(log-update-frequency).
36 SamplesPerSecond = 1000 # Units in samples per second (= hertz). Each loop
      takes about 30usec, thus max sample rate is about 33k samples per
     second.
37 SampleInterval = 1/SamplesPerSecond # Units in seconds.
```

```
# User console update frequency.
UpdateEveryXSeconds = 10 # Update the user every 10 seconds on remaining time.
LastUpdateMsgTime = 0 - UpdateEveryXSeconds # Tracker for the last user interface update.

### SAVE FILE NAME SEARCH (AVOIDS NAME CLASHING AUTOMATICALLY).
sessionCount = 1
fileSpecifier = fileSaveDirectory + subj + "_" + note + "_" + "Session" + "_" + (str)(sessionCount)
logFileName = fileSpecifier + "_" + "OptoFeedbackBoxStimulus" + ".csv"
# Check if file with the desired name already exists. If so, add 1 to the numeric part of the filename and check again.
while(os.path.exists(logFileName)): # Iterate over different session number until a unique one is found.
    sessionCount += 1
    fileSpecifier = fileSaveDirectory + subj + "_" + note + "_" + "Session"   + "_" + (str)(sessionCount)
    logFileName = fileSpecifier + "_" + "OptoFeedbackBoxStimulus" + ".csv"
# Above loop should end upon finding a unique non-existant name. Use the same prefix for the summary log file.
summaryFileName = fileSpecifier + "_" + "OptoFeedbackBoxSummary" + ".txt"

### TRACKER VARIABLES
StartTimeLever = -1 # Unix time of the start of the most recent lever press. Initially set to -1 as nonsense value.
EndTimeLever = -1 # Unix time of the end of the most recent lever press. Initially set to -1 as nonsense value.
PrevLeverState = False # Tracker for detecting changes in lever press states.
LeverRecentlyPressed = False # Tracker indicating whether or not lever had been pressed in recent past.
MostRecentLogUpdate = 0 - SampleInterval # Tracker for the time of the last logging event. Set as 0-SampleInterval so that a new sample will be taken immediately at the start, and after every 'SampleInterval' second(s) thereafter.
CumulativePresses = 0 # Tracker for number of distinct lever presses.

### PWM TIMING CALCULATION & RELATED VARIABLES
TimeSinceStart = 0 # Numeric value that starts from 0 approx. at the start of the code execution, and rises with the passage of time. Units in seconds.
NumP1inP2 = period2 // period1 # Total Number of P1 blocks (1 PWM period) in P2 interval (1 stimulation train duration), rounded down.
loop_counter = 0 # Loop index counter
log_counter = 0 # Log index counter
FLICKER_PREVENTION_INTERVAL = 0.1 # Short interval of time where we ignore oscillation artifacts of lever states.

### POPULATE THE SUMMARY LOG FILE WITH INFORMATION THAT IS ALREADY KNOWN.
```

```
# We open and close this log file, only to later re-open this file, as raspberry pi seems to only be able to maintain 1 stream at a time in active memory during code execution interruptions.
summary_log = open(summaryFileName, 'a')
summary_log.write("Start time (unix): " + str(time.time()) + '\n' +
                          "Subject: " + subj + '\n' +
                          "Note: " + note + '\n' +
                          "Experiment duration (secs): " + str(experiment_duration) + "\n" +
                          "PWM frequency (Hz): " + str(frequency1) + '\n' +
                          "Pulse train frequency (Hz): " + str(frequency2) + '\n' +
                          "Samples per second (Hz): " + str(SamplesPerSecond) + '\n' +
                          "Lever flicker filter width (secs): " + str(FLICKER_PREVENTION_INTERVAL) + '\n' +
                          "IF NO CONTENT EXISTS BELOW, PROGRAM WAS INTERRUPTED EARLY." + '\n' +
                          '--------------------' + '\n')
summary_log.close()

### DECLARE START OF DATA ACQUSITION
print("Start of data acquisition for " + subj + ', session #' + str(sessionCount) + '.')
print("Experiment duration: " + str(experiment_duration) + " seconds.")
print('--------------------')

### SAFELY OPEN LOG FILES
with open(logFileName, 'a') as record_log:
      ### WRITE THE COLUMN HEADERS FOR RECORD LOG FILE
      record_log.write("LogIndex,LoopIndex,UnixTime,TimeSinceStart(s),LeverPressed,LeverPressedOrRecent,PWMOn\n")

      ### MAIN DATA ACQUISITION & FEEDBACK LOOP
      # Update the start time record right before entering the main loop.
      StartTime = time.time()
      # Enter the main loop.
      while TimeSinceStart < experiment_duration: # Run loop only for the duration of the experiment.
            # Update the time tracker.
            UnixTimeNow = time.time() # Unix time (seconds since Jan 1, 1970).
            TimeSinceStart = UnixTimeNow - StartTime
            # Get the lever state (True if pressed, False otherwise).
            LeverState = GPIO.input(lever_pin)
            # If lever was not pressed before, but is now pressed, update the "StartTimeLever" variable.
            if LeverState:
                  if not PrevLeverState and TimePassedSinceLeverStart > FLICKER_PREVENTION_INTERVAL: # Prevent recognizing very fast (~0.1ms)
```

```
    flickering of lever states with the regularization variable '
    FLICKER_PREVENTION_INTERVAL'
107                     StartTimeLever = time.time()
108                     CumulativePresses += 1 # Update the number of
    cumulative presses as well.
109                     print("New lever press detected - lever press #" +
     str(CumulativePresses))
110             # Conversely, if lever WAS pressed before, but just got
    unpressed, update the "EndTimeLever" variable.
111             else:
112                   if PrevLeverState and TimeSinceLastPressed >
    FLICKER_PREVENTION_INTERVAL: # Prevent recognizing very fast (~0.1ms)
    flickering of lever states with the regularization variable '
    FLICKER_PREVENTION_INTERVAL'
113                     EndTimeLever = time.time()
114             # Update the time tracker variables.
115             TimePassedSinceLeverStart = time.time() - StartTimeLever #
    Units in seconds.
116             TimeSinceLastPressed = time.time() - EndTimeLever # Units in
    seconds.
117             # Bin time intervals by 1 stimulation pulse+interpulse period
118             TimeElapsedPeriod2 = TimePassedSinceLeverStart % period2 #
    Units in seconds.
119             # Index each subintervals within one stimulation pulse+
    interpulse period by the PWM period.
120             WhichP1inP2 = TimeElapsedPeriod2 // period1 # Index value
    between 0 and NumP1inP2-1
121             # Normalize the above index to range from 0 (start of current
    PWM period) to 1 (end of current PWM period).
122             FractionP1inP2 = WhichP1inP2 / NumP1inP2 # Value between 0 and
     1.
123             # Calculate time elapsed since the start of the PWM period.
124             TimeElapsedPeriod1 = TimeElapsedPeriod2 % period1 # Units in
    seconds
125             # Calculate the duty cycle & PWM timing based on which point
    within the PWM period and the pulse period the current time is at.
126             DutyCycle = .2 # pulse on for 2.5 msec and off for the rest
127             TimeOn=period1*DutyCycle   # Duration of the time when PWM
    pulse should be on within one PWM period.
128             TimeOff=period1*(1-DutyCycle) # Duration of time when PWM
    pulse should be off within one PWM period.
129             # Check if lever was recently pressed.
130             LeverRecentlyPressed = LeverState or TimeSinceLastPressed <
    stim_duration
131             # Define (or reset for 2nd loop onwards) the tracker for light
     being on or off (default - off).
132             shouldLightBeOn = False
133             # Check if it is within the phase of the PWM pulse to even
    shine the light (true when TimeElapsedPeriod1 < TimeOn).
134             if TimeElapsedPeriod1 < TimeOn:
135                   # If PWM condition is met, only turn light on when lever
```

```
    is currently pressed or if it has been pressed recently (last press
    within 'stim_duration' second(s) from now).
136                 if LeverState or LeverRecentlyPressed:
137                       shouldLightBeOn = True
138           # Toggle the GPIO pin to match the desired state of the light
    output.
139           GPIO.output(light_pin, shouldLightBeOn)
140           # Check if the log was updated recently (within "
    SampleInterval" of now). If not, update the log.
141           if MostRecentLogUpdate + SampleInterval < TimeSinceStart:
142                 # Log the most recently updated set of states into the
    log file.
143                 record_log.write(str(log_counter) + "," + str(
    loop_counter) + "," + str(UnixTimeNow) + "," + str(TimeSinceStart) + ",
    " + str(int(LeverState)) + "," + str(int(LeverRecentlyPressed)) + "," +
     str(int(shouldLightBeOn)) + "\n")
144                 log_counter += 1 # Update the log counter index.
145                 # Update the most recent log update as an addition of
    the sample interval rather than the actual current time to prevent loss
     of logging points due to lag-based bunching.
146                 MostRecentLogUpdate += SampleInterval
147           # Save the previous lever state with the most recently
    acquired lever state before looping back for getting a newer lever
    state.
148           PrevLeverState = LeverState
149           # Update the user of the remaining time if the time is right.
150           if LastUpdateMsgTime + UpdateEveryXSeconds < TimeSinceStart:
151                 LastUpdateMsgTime += UpdateEveryXSeconds
152                 remainingTime = experiment_duration - TimeSinceStart
153                 print("Remaining time: " + str(int(remainingTime//60)) +
     ":" + str(remainingTime%60)[:4] + "; " + str(round(TimeSinceStart*100/
    experiment_duration, 2)) + '% complete.' )
154           # Update the loop counter
155           loop_counter += 1
156
157 # Wrap up the summary log.
158 summary_log = open(summaryFileName, 'a')
159 summary_log.write("Cumulative unique lever press count: " + str(
    CumulativePresses) + '\n' +
160                         '--------------------' + '\n' +
161                         "PROGRAM SUCCESSFULLY RAN TO COMPLETION." + '\n
    ')
162 summary_log.close()
163
164 ### FINISHING MESSAGE
165 print('--------------------')
166 print("Experiment completed. Total unique lever press count: " + str(
    CumulativePresses))
167
168 ### FINISHING CLEANUP
169 # Reset pin states to default idle mode.
```

```
170 GPIO.cleanup()
```