Towards Brain Swarm Interfaces

Aaron van Diepen
MSc Computer Science
Delft University of Technology
Delft, the Netherlands
S4695720

Kilian van Berlo
MSc Embedded Systems
Delft University of Technology
Delft, the Netherlands
S5436737

***Abstract*—The non-invasive brain-computer interface (BCI) records activity in the brain through electrodes and translates this activity to understand the intention of the user. As the demand for robotic swarm control systems increases it is more than logical that people start looking at BCI-based swarm control given its intuitive nature. After all, the best interface is no interface. This brain signal based system could prove especially helpful in industries like e.g. the military or Search and Rescue. In this paper, a prototype of a brain-swarm interface system is presented based on information collected from one electroencephalogram (EEG) signal only. We designed a system that can control the movement of multiple robots through instructions inferred from brain signals. Four subjects participated in our final experiment and were able to control multiple robots through the BCI system. Through these experiments, the potential of a brain swarm interface still proved to be a promising concept for controlling a drone swarm on a high level.**

***Index Terms*—brain-computer interface, brain swarm interface, robotic swarm control, electroencephalogram, electromyogram**

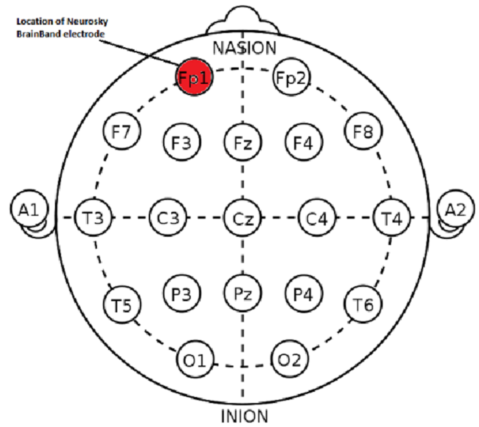

Fig. 1. Location of the electrode Fp1 (Frontal polar - left hemisphere), following the 10-20 Electrode Placement System, is highlighted in red [10]

I. Introduction

From the X-Men movie to episodes in Black mirror, for years people have speculated about the popular idea of using one's brain to communicate with the outside world – a brain-computer Interface (BCI). A BCI is a system that gives the user both communication and control capabilities based on signals recorded through electrodes placed either on the scalp, on the cortical surface, or even within the brain [1]. So far, most of the conducted research known is focused on non-invasive methods where the electrodes are placed on the scalp of the subject instead of invasive methods where surgery is required to place the electrodes on, or in the brain directly. Non-invasive methods make it therefore easier to conduct research, however at the cost of the quality of brain signals obtained. One of the most common (non-invasive) BCI methods used is electroencephalography (EEG). This EEG-based BCI has several paradigms for signal acquisition which can be divided into two groups, endogenous and exogenous paradigms. Exogenous paradigms require external stimuli such as an LED and are hence easier to set up [2]. However, endogenous paradigms, such as motor, visual, and speech imagery on its turn are more intuitive since it is independent of any external stimuli [3].

For acquiring the brain signals, the conventional EEG-based system makes use of a network of a large set of electrodes around the head [4]. When the BCI is to be used in industry, however, the reasonable desire is that it is convenient to use and varies little in performance between distinct users. This would mean that the BCI should be as simple as possible and the instructions as robust as possible. In this study, the *first challenge* is therefore to see how one can optimize the signal acquisition through extracting robust instructions with as few sensors as possible. Therefore, for this experiment, the Neurosky Mindwave Mobile V1 is used with only one EEG electrode. By only using one electrode, the full capabilities of this one sensor are ought to be exploited to get the desired information from the EEG signal.

Numerous BCI applications with only one EEG electrode have already been developed such as for games [5], mobile application control [6] and wheelchair control [7]. One other popular topic for which no one seems to have yet explored the one sensor EEG-based system possibility is the topic of Human-Swarm Control (HSC). The difference here is that instead of one agent, multiple agents are being controlled through neural oscillations within the brain. There have been studies conducted in this field, however only with a big number of electrodes and with the use of complicated paradigms such as motor [8] and visual imagery [9]. None of them, however, seems to look at the more practical BCI systems and how these could be used to achieve robust performance for the control of robotic swarms. In this study, the *second challenge* is hence to design a system with which a swarm of robots can be controlled through the instructions acquired from the one-electrode BCI which formed the first challenge of this paper.

In this study, EEG and Electromyography (EMG) signals are measured through the Neurosky headset from the Fp1 electrode (Fig. 1) and are used to extract instructions for controlling a small swarm of robots. EMG detects the responsiveness of signals originating from muscular movements such as e.g. blinking [7]. The instructions derived show the possibilities of a BCI with minimal sensors while the pipeline shows an effective communication channel for controlling the swarm by the user. As an example, we demonstrate the entire setup with two robots being controlled through the developed Brain-Swarm Interface. For all used code, experiments and results one can go to the GitHub repository of this study [11].

The rest of this paper is organized as follows: Section II gives a description of the materials used and systems implemented. Section III describes the experiments conducted. Section IV presents the results of the experiments and discussion about them. Finally, in session V, a conclusion and a small roadmap to future work are given.

II. Materials and Implementation

A. Communication

For the entire system to work properly, first the global communication architecture is defined with all communication channels it requires to properly control multiple robots through BCI. This architecture is displayed in Fig. 2 and shows this project uses multiple communication protocols such as Bluetooth, TCP/IP and UART to share information across several devices.

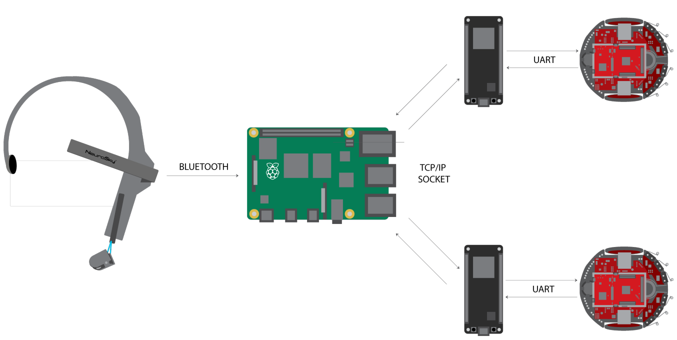

Fig. 2. The communication protocols used for the swarm as displayed in a connected network.

Some of the choices for which protocols to use and at what point did not need a lot of consideration since these options were rather trivial. The communication between the Neurosky and the Raspberry PI 4B (RPI) for example was simple since the only protocol supported by the Neurosky is Bluetooth. The choice of communication between the robots and their respective ESP32s was also trivial since UART communication is the easiest and one of the most effective approaches for the data to be transferred in this case.

Getting to the choice of the right communication protocol between the robots and the server, however, required more in-depth exploration. For this project, two options were readily available and therefore also most desired. This is because ordering other hardware would cost more time to gather everything necessary and given the project time frame this was not desired. The choices that were available for multi-robot communication, and the choices that were thus compared, were Wi-Fi and Bluetooth Low Energy (BLE) via the ESP32. From TI, also a BOOSTXL-CC2650MA providing Bluetooth was available. However, given the limited documentation compared to the ESP32, it was decided not to use this.

Wi-Fi and BLE have been compared on connection range, latency, power consumption, and the maximum number of connected devices. This comparison shows only the most important high-level aspects, characteristics such as e.g. data rate are not considered since a low amount of data is transmitted from the server to the robots anyways. Also, with the evaluation not only our current setup was taken into account but also thought has been given to the capabilities of both techniques in a scaled swarm control environment.

The optimal operation range of Wi-Fi is in general larger than BLE with a range within 50 meters compared to 25 meters approximately, hence a clear plus for Wi-Fi. On the latency, the difference is less clear given that both technologies reflect around the same latency for transferring data. Then, concerning the power consumption, BLE in its turn outperforms Wi-Fi given that BLE is designed specifically for low-energy usage [12]. Eventually, Wi-Fi seems to win the comparison in our case given the number of simultaneous connections it can deal with. The RPI's BLE is limited to a stable maximum of 7 devices [13] while Wi-Fi can easily connect to many more devices. For future use with big swarms being able to only use 7 devices does not seem to be ideal. All comparisons in this Design Space Exploration are summarized in Table I, where the results show a slight preference for Wi-Fi over BLE in our specific case.

TABLE I
Design Space Exploration for multi-robot communication

	Wi-Fi	**BLE**
Connection range	+	-
Latency	+	+
Power consumption	-	+
Number of connections	+	-

Hence, eventually, it was decided to do the communication between the server and the ESP32s over Wi-Fi via TCP/IP sockets, which is a two-way protocol for two devices to talk to each other. The RPI hosts a server application using Python's build-in socket package [14] and this server listens to incoming connections. Once an ESP32 is connected to the same Wi-Fi network it requests to create a socket connection to this server which then gets accepted by the RPI. From that point on the ESP32 forwards all data it receives over the socket from the server over UART to the TI robot. This communication can then be used to forward the output of the BCI to the robot. For debugging purposes, all data output by the robot over UART is also sent to the server over the socket. The RPI then displays this to the screen in a part of the terminal designated to that ESP32/robot pair, which has been achieved using the Python curses package [15].

B. Signal processing

To send user data from the RPI to the robot, first the EEG signals from the Neurosky have to be processed properly. Raw EEG signals can be analyzed but different signals can also be distinguished based on their amplitude, phase and frequency. The amplitude of EEG signals lies somewhere between 10 – 200 V while the frequency is in the range of 0.5 to 40Hz. This frequency range is classified into five different frequency bands [16]:

- Gamma γ (30Hz +)
  Characterizes various functions involved with active information processing
- Beta β (14 – 30Hz)
  Characterizes states of increased alertness and focused attention
- Alpha α (8 – 14Hz)
  Characterizes a relaxed but wakeful state primarily with closed eyes and attenuates with eyes opening or mental exertion
- Theta θ (4 – 8Hz)
  Characterizes the successful encoding of new information
- Delta δ (0.5 – 4Hz)
  Characterizes deep or slow-wave sleep (SWS)

With the help of frequency-domain analysis, these signal properties can be extracted from the brainwave recordings. Thereafter, based on the wave patterns, activities such as an eye blink or increased attention can be detected and used as commands for, in our case, robotic control.

For the extraction of increased attention values, the beta waves of the EEG data are used. When attention is high, the beta activity is also higher than when a person is in a neutral state [17]. Hence to detect attention levels in an individual one needs to track the beta activity and compare it to that person's neutral attention state. Neurosky already has good documentation for recognizing the attention value of individuals in their Software Development Kit [18] hence for this project it is decided to make use of this documentation.

For the extraction of blinking patterns, EMG pattern recognition is needed. Luckily, as mentioned before, this data can be measured using the same sensor as for the EEG signals. Hence, the Neurosky can detect eye blinking with the same sensor as with which it detects attention levels. Compared to other activities, eye blinking has the largest amplitude in EEG signals, making it easily detectable through its peaks [19]. This peak is produced through the closing (positive peak) and opening (negative peak) of the eyelids since at these points positive and negative deflections are produced as shown in Fig. 3.

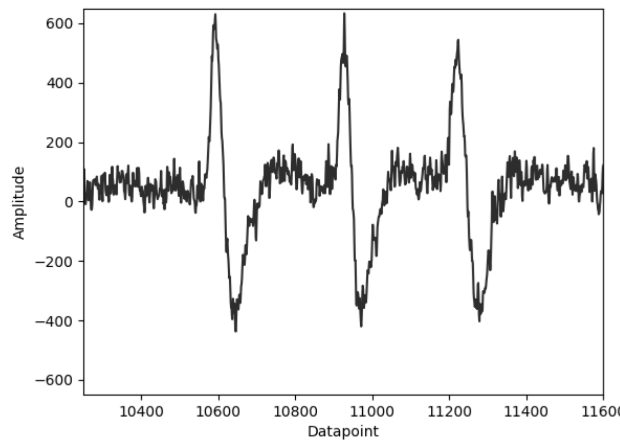

Fig. 3. Three blinks detected in a row

These peaks turned out to have the same threshold levels for different users, making it unnecessary to adjust this threshold to each person individually through specially dedicated algorithms. The way blink detection hence works in our study is that for both the positive and negative peaks a different threshold is set (300 and -120 respectively) based on the acquired EEG data from the experiment, which will be discussed later. With the help of these thresholds, the number of blinks within a specified time frame is kept track of. A blink is defined as a positive peak above 300 followed by a negative peak below -120. The number of times this order of events occurs within 500 received data points after the first blink determines how many blinks are detected. This time frame of 500 data points is chosen to control the robots without any impermissible delays whilst still being able to detect multiple blinking patterns with relative ease.

C. Robot control

The robots used for the experiments are two TI-RSLK MAX robots from Texas Instruments. The hardware of these robots is easy to assemble and the base model requires no soldering. The robot uses a differential drive, this means its movement is based on two separately driven wheels placed on either side of the robot body, removing the need for a steering motor. Therefore turning the robot while not changing its position is possible by running both motors at the same speed but in opposite directions.

Added onto the base system of the robots are three sensors, one facing forward and the other two facing 45 degrees to the left and right of the front of the robot. These sensors are eventually used for obstacle avoidance. Two different types were used for the robots to compare their results, one robot had three infrared sensors, while another had three ultrasound sensors. The way infrared sensors work is that they continuously output a beam of infrared light, this light is then reflected by nearby objects. Another part of the same sensor then measures the intensity of light reflected by these objects, after which this value is then converted into the relative distance to said objects. Ultrasound sensors work slightly different, with sounds. These sensors allow us to send out a sound wave that is inaudible for humans, which when reflected by nearby objects can then be detected by the sensors again. This gives us the difference in time between sending out the sound wave and measuring it coming back. Using the speed of sound this time difference can then be used to calculate how far an object is from the sensor. Both approaches are supposed to give us a simultaneously and continuously updated value for each of the sensors.

To eventually let the robots also partly function autonomously within the swarm, and so eventually create a bigger Degree of Freedom for swarm control with BCI, it was tried to give them the ability to go to a given goal position all by themselves while avoiding obstacles on the way. The way this is done is that the robot goes directly to the given goal until it can no longer freely move. This limited freedom of movement is defined to be the case when an obstacle is within a range of 50cm of one of its three sensors. For the robot to accurately behave this way, it uses the tachometers connected to both wheels of the robot which is used to update a robot's relative position (x, y) and relative angle to its starting point (θ), as shown in Fig. 4.

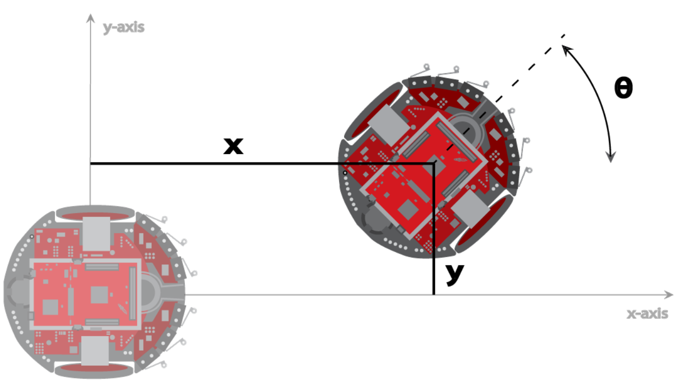

Fig. 4. Position and angle of the robot compared to its starting point

When no objects are detected, the robot uses its position and the position of its goal to calculate a vector pointing toward the goal. This vector is then used to calculate a turning speed. The turning speed can either be positive or negative depending on whether the robot should turn left or right. To calculate the speed of the left and right wheels we can subtract the turning speed from the default speed at which the robot moves. Passing these speeds as a PWM signal to the robot allows the robot to turn and move towards the given goal.

When an object is detected within 50cm of one of the sensors the robot switches to obstacle avoidance, here the measurements of all three sensors are converted to vectors relative to where they are located on the robot. These distance vectors are then added up together to determine the direction the robot has to move towards (Fig. 5). Because the sensor closest to the obstacle has the lowest distance measurement, simply adding these vectors gives us a single vector driving away from the obstacle. This directional vector, which roughly represents the robot's direction, is consecutively used to calculate the turning speed. This turning speed can then be used again to calculate a speed for the left and right wheels allowing the robot to turn and move towards the direction of that constructed vector.

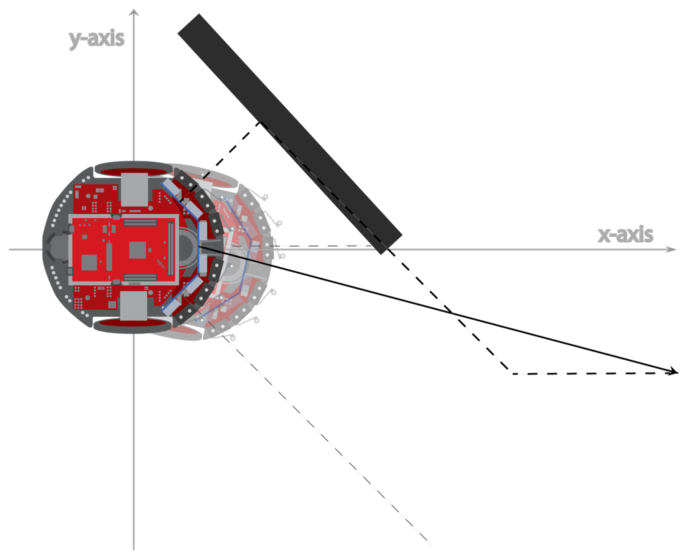

Fig. 5. Obstacle avoidance

III. Experimental Setup

To tackle the challenges in this study, several experiments have been conducted to test multiple aspects of the system. First, tests with regards to the EEG and EMG detection analysis are conducted after which the communication pipeline from the Neurosky to the Robot and back is assessed. Then the robots' autonomous capabilities are analysed and after this everything is put together for the final tests where the functioning of the entire system is evaluated.

A. Communication

To analyse the communication pipeline properly, the data of the Neurosky was sent via the server and the ESP32 to the robot, where, in its turn, an acknowledgement was sent back via the ESP32 to the server. On the server terminal, both the data sent (from the Neurosky) and the data received (from the robot) are outputted to check whether all data is being communicated properly. This experiment is done for both a one-to-one connection (one robot) and a one-to-many connection (two robots).

B. Signal processing

The evaluation of the EEG signals is done differently than the experiment for the communication since in this case test subjects are needed to complete the experiment. For tests of the proper functioning of the signal processing, the raw data from the electrode of the Neurosky is sampled at 512Hz and then sent to the server via a Bluetooth connection. Since the sensor is made of a dry electrode no skin preparation or conductive pastes are needed. For the experiment, the raw EEG signals of 7 subjects with ages ranging between 23 - 28 are collected. The experiment took place in a quiet environment with only a chair for the participant to sit on and a desk with a computer screen. For the attention value, the test consisted of measuring the attention value for an entire minute. The participant was instructed to keep the attention value as high as possible throughout the minute. By doing this it is evaluated how easy it is to obtain a certain value and to then stay high in attention level. For the blink detection, each participant had to partake in one session of 60 seconds where the participant had to blink the number of times as indicated on the screen (either 1, 2, or 3). To successfully conduct these sessions, python programs specifically for the experiment were used.

C. Robot control

Before the robot can be controlled via BCI instructions it has to be made sure that the basic functions of the robot already work appropriately. Therefore, several components of the robot itself are tested before the complete system is put together. First, to check the functioning of the motors, directional instructions were given to the robot where it moves forward, backward, left and right several times. After the motor functioning was properly checked the sensors were tested by putting an obstacle at a certain distance from the sensors and noting down the difference of the sensor results from the actual distance (Fig. 6). Finally, when both motors and sensors work, both behaviours for obstacle avoidance and going to a given goal are supposed to be examined. This would be done by consecutively looking at how many obstacles are successfully avoided and how close to the goal the robot gets on average.

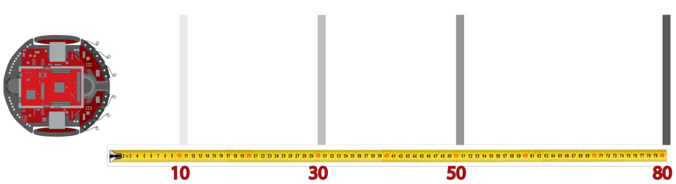

Fig. 6. Experimental setup for determining the accuracy of the IR and US distance sensors

D. Brain Swarm Interface

After the subsystems have been tested, eventually the entire system is put up for tests. This means that both robots are controlled simultaneously through the live instructions extracted from the BCI. The experiment was conducted with a total of four participants, in a quiet, square room (3m x 3m) with enough space for the robots to move around. Here, a participant gets a few minutes to get familiar with the BCI. After getting familiar, the participant instructs the robots what to do through attention levels and blinking.

With an attention value above the set threshold, the robots started driving, while at lower levels they stopped. By blinking twice and trice the robot went to the left and right respectively. Going backwards could have been implemented but it was chosen not to since this can be achieved by going left (or right) twice. The reason a high attention level was used for unlocking the robots was that when controlling anything, especially a swarm, everyone's safety must be guaranteed. The person controlling the swarm should therefore give enough attention to the task at hand to ensure safe execution.

IV. Results and discussion

Using the subsystems and the final prototype, the EMG and EEG detection analysis, communication and robotic autonomy were tested. Eventually, based on the results, the proposed system is evaluated.

A. Communication

The communication from the RPI via the ESP32 to the robot was tested for packet loss and delay in received messages. The information was gathered by sending messages via the terminal to both robots and back, as can be seen in Fig. 7 (together with a small chain of commands being sent out and being returned). The results were solely positive given that during the tests no messages were lost, achieving an accuracy of 100%, and the time it took for the acknowledgement to return from the robot was impossible to detect by the naked eye, which makes sense since the data being send is small (one character each time). Hence, it can safely be assumed that the communication between the components is working and performing as expected.

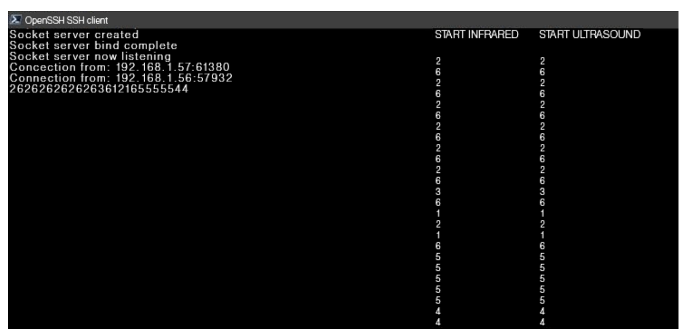

Fig. 7. Results of the communication pipeline, left is the established connections with the characters being sent by the server, right are both robots with the characters they received and sent back to the server

B. Signal processing

For evaluation of the attention levels, it was important to know just how easy it is for participants to maintain a certain level of attention over an extended period. This was important in light of the activation of the robot using a certain level of attention as the threshold. With 6 participants it was tested just how high their attention level was for an entire minute as shown in Fig. 8. Based on the results one can conclude that reaching a level of high attention is easier than actually maintaining it and fluctuations in this level are not rare. Based on these results it was eventually decided to set the attention level threshold for the complete system tests halfway the range, so at 50, since most participants were able to keep their attention consistently above this threshold.

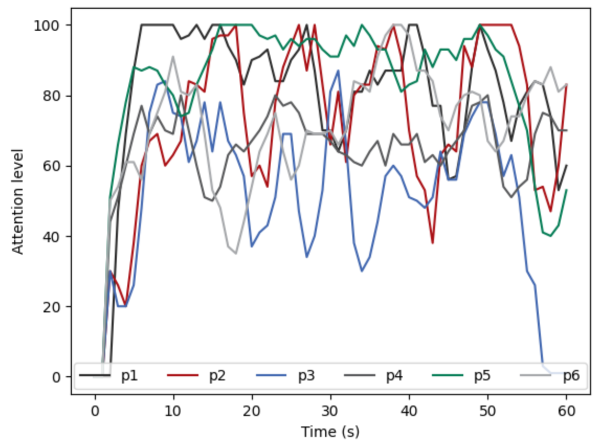

Fig. 8. Level of attention each participant was able to maintain throughout 60 seconds

The accuracy of blink pattern recognition by the program, performed by each participant, is represented in the bar chart of Fig. 9 (avg. accuracy of 92.46%). Here, the accuracy that each participant submitted during the blink detection experiment is shown. The maximum accuracy achieved was 100%, meaning the program classified all the blinking trials of the experiment correctly. The smallest number of trials someone underwent in one minute is 18, the maximum was 21.

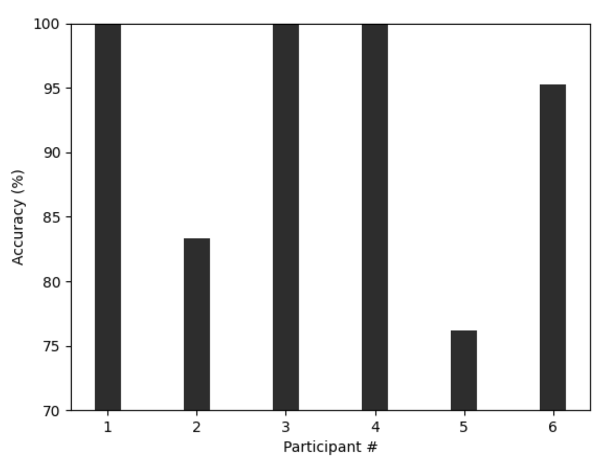

Fig. 9. Level of the accuracy of each participant for detecting blinks

C. Robot control

During testing, the actual action send to the robot and its resulting movement operated flawlessly. The communication proved to be working already before. Now, however, it was also shown that when the robot had to move in a certain direction, it did that perfectly. When the wheels had to turn, they exactly did as asked. Therefore, already the minimum desired test for the complete system can be executed. Controlling both robots through a BCI. The next step would be to have a more advanced interaction between the robot and its human operator by making the robot more autonomous.

For the autonomous functioning of the robot, however, unfortunately, multiple difficulties arose. First, when starting the test of the distance sensors the infrared sensors did not seem to be giving the consistent results that we observed in the weeks before we started the experiments. To make matters worse, the ultrasound did not seem to give any sensible data at all anymore. Eventually, we did not succeed in our efforts to get it working again within the time that we had left since we performed the tests relatively late in the knowledge that it had worked like a charm any time before. Because of this, there are no records of the accuracy of the sensors.

Logically, since the distance sensors did not work anymore it was impossible to get quantitative data about the obstacle detection (which also had worked before as can be seen in the video located in the GitHub at EXPERIMENTS/RobotFunction [11]). What we do know from the former tests with the obstacle detection is that the ultrasound had difficulty with the detection of particularly thin objects in front of the sensor, infrared on its turn was much more capable of dealing with these thin objects.

Then the last test for the robot to be performed is its capabilities to go to the indicated goal. Unfortunately also here we were not able to test its performance since we did not get the tachometer on the robot to execute fast enough to accurately measure the robot's position. Due to an error in the tachometer, it was frankly impossible to keep track of the robot's position which forced us again to not be able to provide any results on this test. The main issue here was that the tachometer did not function as was indicated by the documentation provided by TI [20], making it unnecessarily difficult to find out where the problem was arising. Following the tutorials on building the tachometer from scratch also proved to not be working, leaving us with no documentation to follow whatsoever. In the future, one thing that should be approached differently, looking back, is therefore that a different robot should be used since this robot seems to be out of date given the little amount of documentation that is available, which even proved inaccurate at times.

D. Brain Swarm Interface

Given the troubles experienced with the autonomous functioning of the robot, it was undesired to do the tests that partly depended on this autonomy. Therefore, with everything that did work, a simpler version of the final test was constructed. As discussed before, this experiment was about controlling both robots through attention levels and eye blinking. Luckily, the functioning of the entire system did show promising results in the trimmed down version. When controlling the robot, for each of the four different participants, the system showed considerable accuracy in following the commands given to them, which were:

- Drive: Attention above the threshold
- Turn left: Blink twice
- Turn right: Blink trice
- Stop driving: Attention below the threshold

One thing that turned out to be harder to maintain when operating the robots was the level of attention. As can be seen in Fig. 10 this level was on average lower and showed to be more difficult to keep above the threshold. The participants indicated that this was due to the sudden movement of the robots the moment the attention value got above the set threshold (50). The disruption of their attention also explains why the attention value does not get frequently as high as with the attention value experiment without robots. Hence, with events happening suddenly around someone it is more difficult for a person to maintain high attention levels and thus control the robot consistently.

In terms of blink detection, the brain swarm interface (BSI) still worked as well for these four participants as with the former tests shown in Fig. 9 (average accuracy of $\sim 90\%$) while also the communication pipeline eventually did not show any notable delays nor information loss. In the beginning, the communication pipeline did show considerable delays. This, however, was solved by threading the communication with the Neurosky and the robots separately since this way they were not waiting for each other during execution.

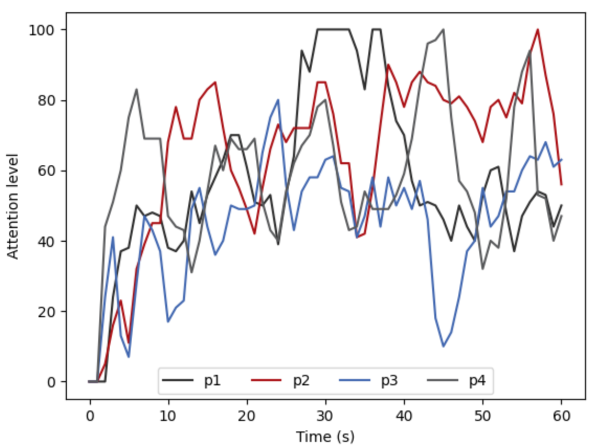

Fig. 10. Level of attention each participant was able to maintain throughout 60 seconds while controlling both robots

V. Conclusion and future work

In this work, several novel IoT solutions have been merged to form a brain swarm interface. EEG and EMG data from the Neurosky headset was measured after which this data was transmitted over Bluetooth and interpreted on a server. These signals were then published in the form of instructions over Wi-Fi from the server to each robot in the system (two in this case). These robots then executed the received instructions and showed promising future possibilities for controlling a big swarm of robots through neural oscillations. All code, experiments and results for this research can be found on the GitHub repository [11].

Unfortunately, not everything that was supposed to be developed was completed. With the development of the robot, numerous problems arose and eventually not all problems were completely solved. Despite this leading to a less complex brain swarm interface than aimed for initially, it was still possible to successfully come up with solutions for the challenges set in the beginning. The first challenge of finding robust instructions through data from as few sensors as possible was completed by detecting blinking bursts and attention levels of a person through just one sensor. Controlling attention levels proved to be fairly difficult to master in a dynamic environment. In the future, one might therefore improve this by finding another, better way to make the swarm of robots start and stop. The attention level could also be used to let the swarm perform a different action. Also, in the future one might even achieve to recognize more instructions from a single sensor as in literature promising results were shown e.g. meditation levels and jaw tension detection [7] [18].

The second challenge, to design a system with which a swarm of robots can be controlled through the instructions acquired from a one-electrode BCI, was also completed. Not entirely in the complex setup that we had hoped for but the great potential in the use of robot swarm interfaces was shown. One logical next step in the future would be to get the robots to be able to navigate their way to a goal through a dynamic environment with obstacles. This navigation should however be done while taking the position of all other robots in the swarm into account. This brings us to the question of how much autonomy is desired from the robots [21]. The lack of EEG-sensors means that fewer instructions can be acquired, which on its turn means that the swarms' behaviour cannot be controlled in as much detail. An interesting question to research in the future is therefore what the balance is that you want between robot autonomy and human interference, probably this also depends on the environment and the goal of the swarm. In search and rescue missions, for example, one might desire a greater level of control than for the use of swarms in agriculture, or exploration [22].

Regardless of the question about how much control and autonomy is desired, there are more technical aspects that could be researched as well. These are mainly about the communication in the case of a big swarm of robots and the autonomous behaviour of this swarm as a whole. In terms of improving communication in the future, it could be beneficial to explore mesh networks, which allow the user to create an integrated network of indirectly connected modules. This type of network operates on a flood network principle, every node that receives a message for the first time forwards it to all nodes connected, this way not all nodes have to be fully connected at the same time but still receive all messages sent by the host device. For the behaviour of the swarm, the research could be conducted on topics like aggregation, hovering, splitting, dispersing, etc. About these topics, just as with BSI, little is yet known and therefore they remain interesting challenges to this date.

REFERENCES

[1] Wolpaw, J. R. (2013). Brain–computer interfaces. *Neurological Rehabilitation, 110*, 67–74. https://doi.org/10.1016/b978-0-444-52901-5.00006-x

[2] Kwak, N.-S., Müller, K.-R., & Lee, S.-W. (2017). A convolutional neural network for steady state visual evoked potential classification under ambulatory environment. *PLOS ONE, 12*(2). https://doi.org/10.1371/journal.pone.0172578

[3] Lee, D.-H., Jeong, J.-H., Ahn, H.-J., & Lee, S.-W. (2021). Design of an EEG-based drone swarm control system using endogenous BCI PARADIGMS. *2021 9th International Winter Conference on Brain-Computer Interface (BCI), 4*. https://doi.org/10.1109/bci51272.2021.9385356

[4] Dias, N. S., Carmo, J. P., Mendes, P. M., & Correia, J. H. (2011). Wireless instrumentation system based on dry electrodes for acquiring EEG signals. *Medical Engineering & Physics, 34*(7), 972–981. https://doi.org/10.1016/j.medengphy.2011.11.002

[5] Marshall, D., Coyle, D., Wilson, S., & Callaghan, M. (2013). Games, gameplay, and BCI: The state of the art. *IEEE Transactions on Computational Intelligence and AI in Games, 5*(2), 82–99. https://doi.org/10.1109/tciaig.2013.2263555

[6] Dobosz, K., & Wittchen, P. (2015). Brain-computer interface for mobile devices. *Journal of Medical Informatics & Technologies, 24*, 215–222.

[7] Narayana, S., Prasad, R. R. V., & Warmerdam, K. (2019). Mind your thoughts: BCI using single EEG Electrode. *IET Cyber-Physical Systems: Theory & Applications, 4*(2), 164–172. https://doi.org/10.1049/iet-cps.2018.5059

[8] Karavas, G. K., Larsson, D. T., & Artemiadis, P. (2017). A hybrid BMI for control of robotic swarms: Preliminary results. *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 5065–5070. https://doi.org/10.1109/iros.2017.8206390

[9] Jeong, J.-H., Lee, D.-H., Ahn, H.-J., & Lee, S.-W. (2020). Towards brain-computer interfaces for drone swarm control. *2020 8th International Winter Conference on Brain-Computer Interface (BCI)*. https://doi.org/10.1109/bci48061.2020.9061646

[10] Shirazi, A. S., Hassib, M., Henze, N., Schmidt, A., & Kunze, K. (2014). What's on your mind? *Proceedings of the 5th Augmented Human International Conference*, 1–4. https://doi.org/10.1145/2582051.2582096

[11] Majid, A., van Diepen, A. & van Berlo, K. (2021) Brain-Swarm Interface [Source Code]. https://github.com/amjadmajid/O_Robot/tree/ti-rslk-max.

[12] Inpixon. (n.d.). *Ble (bluetooth) Location Tracking & Posit-ioning*. Inpixon. Retrieved November 15, 2021, from https://www.inpixon.com/technology/standards/bluetooth-low-energy.

[13] Miyajima, Y. (2019, August 31). *Raspberry pi 3B+ maximum ble connections #RaspberryPi: Developersio*. DevelopersIO. Retrieved November 15, 2021, from https://dev.classmethod.jp/articles/raspberry-pi-3b-plus-max-ble-connection-en/.

[14] *Socket - low-level networking interface*. socket - Low-level networking interface - Python 3.10.0 documentation. (n.d.). Retrieved November 1, 2021, from https://docs.python.org/3/library/socket.html.

[15] Kuchling, A. M., & Raymond, E. S. (n.d.). *Curses programming with python*. Curses Programming with Python - Python 3.10.0 documentation. Retrieved
November 1, 2021, from https://docs.python.org/3/howto/curses.html.

[16] Campisi, P., & La Rocca, D. (2014). Brain waves for automatic biometric-based user recognition. *IEEE Transactions on Information Forensics and Security, 9*(5), 782–800. https://doi.org/10.1109/.2014.2308640

[17] Kučikienė, D., & Praninskienė, R. (2018). The impact of music on the bioelectrical oscillations of the brain. *Acta Medica Lituanica, 25*(2), 101–106. https://doi.org/10.6001/actamedica.v25i2.3763

[18] Neurosky. (2017, June 14). *Brainwave starter kit product support*. mindwavemobile [NeuroSky Developer - Docs]. Retrieved October 12, 2021, from http://developer.neurosky.com/docs/doku.php?id=mindwavemobile.

[19] Abo-Zahhad, M., Ahmed, S. M., & Abbas, S. N. (2015). A new EEG acquisition protocol for biometric identification using eye blinking signals. *International Journal of Intelligent Systems and Applications, 7*(6), 48–54. https://doi.org/10.5815/ijisa.2015.06.05

[20] *RSLK Max Edition curriculum.* TI. (n.d.). Retrieved September 24, 2021, from https://university.ti.com/en/faculty/ti-robotics-system-learning-kit/ti-rslk-max-edition-curriculum.

[21] Hussein, A., & Abbass, H. (2018). Mixed Initiative Systems for human-swarm interaction: Opportunities and challenges. *2018 2nd Annual Systems Modelling Conference (SMC).* https://doi.org/10.1109/sysmc.2018.8509744

[22] Olaronke, I., Rhoda, I., Gambo, I., Oluwaseun, O., & Janet, O. (2020). A systematic review of Swarm Robots. *Current Journal of Applied Science and Technology, 39*(15), 79–97. https://doi.org/10.9734/cjast/2020/v39i1530719