

NEUROENGINEERING LABORATORY

Professor Pavan Ramdya



---

Semester Project:

# **Robotic Pantograph System to Automate Biological Experimentation**

---

Matteo RIGHI

Supervisor:

Matthias DURRIEU

LAUSANNE, 13 JANUARY 2023

**EPFL**

# Table of Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Design</b>	<b>2</b>
2.1	Constraints . . . . .	2
2.2	State of the Art . . . . .	4
2.3	Robotic Structure . . . . .	5
2.4	Optimization of the Pantograph Design . . . . .	8
2.5	Detailed Pantograph Design . . . . .	10
2.6	Dimensions and Arenas Grid . . . . .	11
<b>3</b>	<b>Fabrication and Programming</b>	<b>11</b>
3.1	Parts Selection . . . . .	11
3.2	Electronics . . . . .	12
3.3	Programming . . . . .	14
<b>4</b>	<b>Results</b>	<b>14</b>
4.1	Final Fabricated Structure . . . . .	14
4.2	Testing . . . . .	15
<b>5</b>	<b>Discussion</b>	<b>16</b>
	<b>References</b>	<b>19</b>

## Abstract

The goal of this project is to design, fabricate and program a robotic structure to automate experimentation on *Drosophila Melanogaster* flies. This project fits into experiments being carried out at the Neuroengineering Laboratory at EPFL on the interaction of *Drosophila* flies with 1.5 mm steel balls inside closed arenas. At the end of each experiment, a robotic system is needed to reset the balls to the start position. To maximize the efficiency of the experimentation, a robotic system must be developed which can perform these resets, as well as some other actions, on a grid of arenas in a high-throughput fashion.

Several different structures were considered, including Cartesian and rotational structures. To minimize the visual occlusion of the workspace and to limit the waste of space, it was chosen to use a pantograph structure mounted on a linear slider. A MATLAB simulation was written to better understand the workspace of a pantograph. Based on the simulation, optimal geometrical parameters were found.

The structure was fabricated from laser-cut Plexiglas segments and 3D-printed parts. Servo motors were chosen to actuate the shoulder joints and a linear stage driven by a stepper motor was used for the slider. The structure was controlled by an Arduino Uno microcontroller; code was written to generate a linear trajectory.

In testing, it was found that the structure was successfully capable of driving a ball inside a single arena. However, issues were encountered with the positional accuracy, which were attributed to the servos.

All related videos and files can be accessed at the following link:

<https://github.com/NeLy-EPFL/pantograph>.

## 1 Introduction

Object perception is a crucial part of the strategies used by humans and animals to navigate their environment. While low-level object interactions have been extensively studied, higher-level interactions are less well-understood. The work of James J. Gibson has been influential in quantifying such interactions notably with the concept of affordance. Affordance describes how a human or animal can abstract what objects in its environment can offer it. Humans learn to associate to certain objects corresponding actions; for example, a mug can be grasped to consume its contents. Previous biological studies have focused on non-human primates, however, there is evidence that insects are also capable of similar levels of understanding.

This project forms a part of ongoing studies at the Neuroengineering Laboratory of EPFL to test object affordance in *Drosophila Melanogaster* fruitflies. These studies focus on exploring the interactions of the flies with steel balls inside a closed arena. Arenas are made from sheets of transparent Plexiglas with paths cut out where the fly can walk, forming a simple maze (see Figures 1a and 1b). Arenas are closed on the top and bottom by two more sheets of Plexiglas, and feature a simple gate which allows flies to be placed into the arena. 1.5mm-diameter steel balls are placed inside the arena in front of the fly. The goal of the experiments is to explore how the fly learns to manipulate the ball to navigate the maze. Arenas are lit from above using a ring of LEDs and recorded from above; Figure 1b shows a sample frame. Video from the experiments

can then be analyzed using machine-learning software such as DeepLabCut and SLEAP, notably to estimate the pose of the animals.

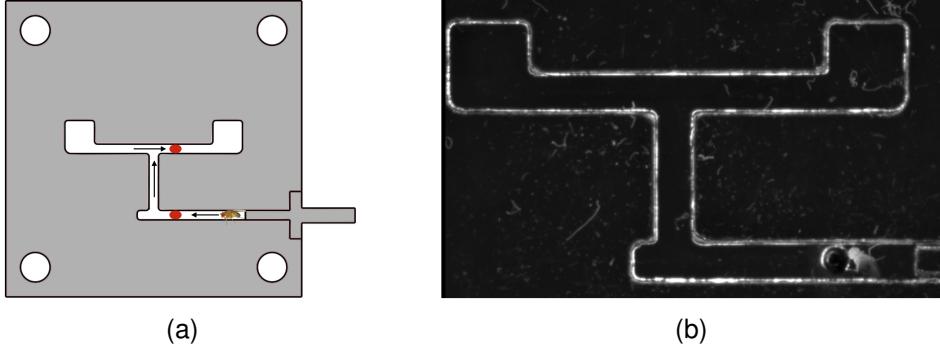


Figure 1: Example of the maze experiments; (a) graphic representation of the task, (b) sample recorded frame

The steel balls inside the arena can be manipulated from the outside using a magnet. A magnet is used to grab balls through the Plexiglas, and then to move them around as desired. This is necessary notably once the experiment is terminated, and the ball must be reset to the starting position. The task requires relatively high precision, and must be done quite frequently (an experiment typically lasts up to one hour, during which a fly will push the ball between 0 and 25 times, which leads to up to one magnet event every 2-3 min), which makes the task tedious for a human to carry out. For this reason, current experiments use the existing robotic system belonging to the lab, the Optobot gantry structure. However, the Optobot gantry is subject to the main limitation of having a small workspace (10cmx10cm), which does not allow it to process more than one arena at once.

Ideally, these object affordance experiments would be carried out with a very high throughput, with tens or even hundreds of experiments taking place in parallel. For all types of experiments with genetically modified animals, it is important to have a high throughput, to ensure that all the desired genetic modifications can be tested within a reasonable timeframe. The limited workspace of the Optobot gantry makes it unsuitable for this task. The goal of this project will be to design, build and program a robotic system allowing high-throughput automated experimentation on *Drosophila* flies. Concretely, this system should operate on a grid of arenas placed side-by-side, each with an experiment in progress. Once one experiment terminates, the system, using a magnet placed at the end-effector, should be capable of moving balls back to the start position.

## 2 Design

### 2.1 Constraints

As discussed in the previous section, the robotic system should navigate a magnetic end-effector around the array of arenas to reset the steel balls. Thus, a first constraint is for the system to have two translational degrees of freedom (DOF), i.e. free movement in the plane of the arenas grid. Note that theoretically, a third degree of freedom for vertical movement should be necessary: this

would allow the system to grab and release balls by moving the magnet up and down. For this project, it was chosen for the sake of simplicity to neglect this aspect; instead, the system releases balls by bringing them to a dead end, so that they cannot follow the end-effector as it continues to move forward.

There is a constraint on the workspace. Arenas are square of side 5 cm, with mazes inside each arena taking up at most a square of side 3cm. Figure 2 shows an example of such an arena. Note that the arenas currently in use in the lab have a 3D-printed base with four projections (see Figure 2), to allow it to be mounted on the rails of the Optobot. These projections are not necessary for this project; furthermore, the additional space needed for these rails actually limits how tightly arenas can be packed together. Nonetheless, this design of arena was used for this project, as they are the standard in use in the lab, therefore it is impractical to design and print a new class of arenas for just this project. Based on the space available and on the feasibility of fabricating a larger structure, a tentative goal was for a workspace of 3 by 5 arenas.

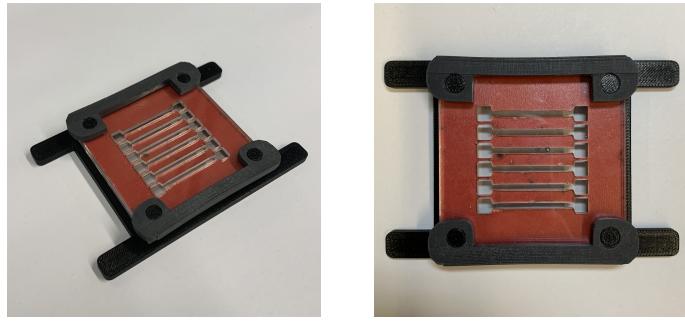


Figure 2: Example arena design

Another constraint is for the robot to be capable of generating an end-effector trajectory which can reliably carry the balls. It was found that using a 3 mm diameter ferrite magnet, it is possible to control the ball even when the magnet is up to 3 mm away. Note that some arenas, for example the arena shown in Figure 2, have several paths and several balls in close proximity, therefore, it is not possible to simply use a larger magnet as then it will grab balls from the nearby paths. Considering that in the arena in Figure 2 corridors are 5mm apart, a tentative target is to reach a positional precision of 2.5 mm.

Additionally, the trajectory should be smooth enough to prevent the ball from detaching due to inertial forces. In fact, it was found empirically that the ball is of such small mass for this to be a negligible concern.

Another crucial constraint is a limited occlusion of the workspace. Indeed, in the final version of this project, it is assumed that multiple experiments would run in parallel, and the robotic system should be able to reset an experiment once it finishes without disrupting the recording of the other experiments which are still taking place. Note that this constraint has some leeway possible: in the current software used to process video of the experiments, frames where the image is occluded are disregarded. Therefore, it is allowed to temporarily occlude certain scenes, however, occlusion should be minimized, to minimize the number of useful frames which are cut out. No quantitative values are given for this constraint.

A third, somewhat more arbitrary constraint is on the total space necessary for the robotic system. Indeed, the lab has limited space available. Concretely, for this setup, there is an available space

of  $60 \times 75$  cm. Note that for this first implementation, this constraint is less important, as the entire setup can be made to take up less space by simply reducing the number of arenas in the grid; however, for the sake of future versions which might have a minimum number of arenas necessary, it is important to find a design which minimizes the wasted space.

Table 1 summarizes the constraints:

Constraint	Metric
DOF	X and Y
Workspace	$5 \times 3$ arenas
Precision	2.5 mm
Limited workspace occlusion	N/A
Limited size	60 x 75 cm

Table 1: Summary of the design constraints

It should also be noted that for this project, the speed of the robotic structure is not a main constraint. It is assumed that the structure causes a negligible occlusion of the workspace, therefore it is not a problem if it is in the workspace for a longer period of time.

## 2.2 State of the Art

This section discusses robotic structures used in similar studies of experimentation on *Drosophila* flies.

A 2021 paper by Joshi et al. [1] presents a 3-DOF Cartesian structure for microinjection in *Drosophila*. The Cartesian structure has the advantages that it is very precise, and that it is easy to control. In fact, the existing Optobot gantry structure uses a similar structure. The downsides of implementing a Cartesian structure for this project are discussed in Section 2.3.

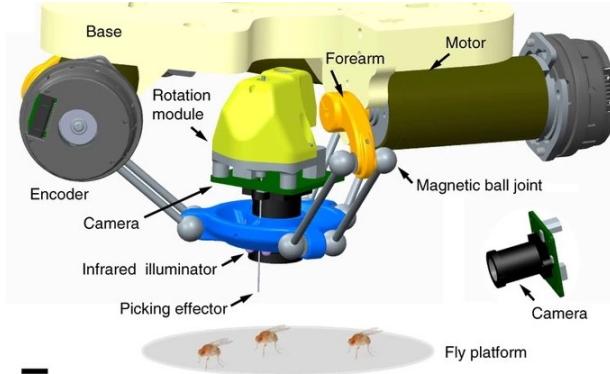


Figure 3: High-precision Delta structure for *Drosophila* handing. *Image source: Savall et al., 2015*

A 2015 paper by Savall et al. [2] proposes a 4-DOF robotic system for handing *Drosophila* flies (see Figure 3). DC motors and encoders are used to ensure accurate movement of the motor joints. The Delta structure is common for pick-and-place type tasks: it can allow for high-precision movements, and the parallel structure ensures fast movement of the end-effector. Additionally,

because the motors are mounted at the base, this structure causes a limited occlusion of the workspace. For this project, the Delta structure is too complex to build; additionally, the Delta structure has 4 degrees of freedom, while for this project only two are needed.

As regards the topic of minimizing the workspace occlusion, there is little literature available. A 2021 paper by Yoshioka et al. [3] proposes a system of mirrors which an auxiliary camera uses to detect objects. For this project, mirrors are not a good option, as they risk distorting the image or otherwise generating artifacts in the recordings.

### 2.3 Robotic Structure

To achieve the goals set out for this project, several classes of robotic structure were considered.

The simplest structure is the Cartesian serial structure, which is used in the Joshi paper [1] and is also used by the Optobot gantry robot. Figure 4a shows the implementation for the Optobot gantry, Figure 4b shows a render of what the same structure would look like for a 30x30 cm workspace. This structure uses two actuated linear stages placed one onto the other to control the two linear degrees of freedom in the plane. In particular, a Plexiglas end-effector is mounted on the second actuator, so that the end-effector will not occlude the recording of the workspace. To prevent the second linear actuator from occluding the workspace, the Plexiglas end-effector is of equal length to the workspace, such that when the end-effector is at the farthest edge of the workspace, the second actuator is at the other edge, and does not occlude the workspace. Thus, this structure takes up twice as much space as the desired workspace.

Using actuated linear stages, this structure can achieve very high positional precision with simple programming. Inexpensive (as low as 50 CHF) commercially available options use stepper motors with 200 discrete rotational positions, which combined with the screw to generate linear motion gives a positional precision of 0.025 mm; there exist Arduino libraries such as AccelStepper which automate the generation of the control signal from a desired target position. It was chosen not to use this structure because of the significant waste of space, as discussed above.

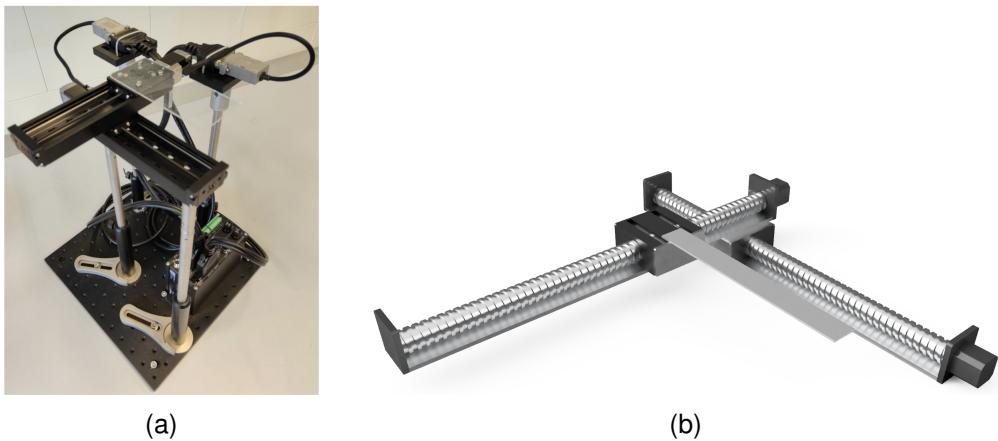


Figure 4: (a): Optobot gantry; (b): Serial Cartesian structure

To remedy the downsides of the serial Cartesian structure, a parallel Cartesian structure was considered. In this case, "parallel" denotes the fact that all actuators are placed at the base, and

actuate the end-effector in parallel. An example implementation of such a structure is shown in Figure 5: the end-effector is carried by two gantries, each of which is supported by a linear actuator on one side and a linear rail on the other. Similarly to the serial structure described above, the gantries might be made from transparent Plexiglas, to limit the visual occlusion of the workspace. It should be noted that compared to serial structures, parallel structures have the advantage of being faster, as the first actuator does not have to move the extra weight of the second actuator; however, for this project this is not a major consideration. Compared to the serial structure described in the passage above, the parallel structure presents the advantage that it uses space far more efficiently: its workspace covers the entire surface taken up by the structure. Similarly to the serial structure, the parallel structure can achieve extremely high accuracy and is relatively easy to program.

However, it was observed that there might be issues with the fabrication of this structure, namely, with the central joint which connects the two gantries. This joint should have very low friction, to prevent the two gantries from getting stuck when one actuator moves. Such a structure can be fabricated relatively easily using manufacturer parts in steel, however, then the gantries would cause a significant occlusion of the workspace. This structure was disregarded based on the fact that it would be too difficult to fabricate this structure using Plexiglas gantries.

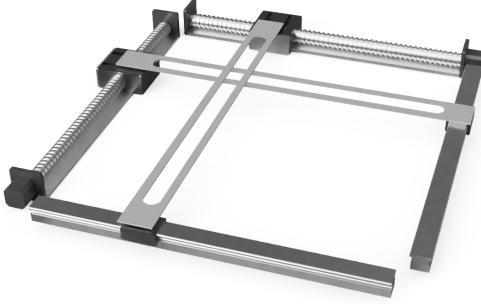


Figure 5: Example implementation of a Cartesian parallel structure

Some non-Cartesian structures were considered, notably using rotational actuators. One such structure might be a serial rotational-rotational (RR) structure. Figure 6 shows an example implementation: the structure forms a robotic arm, with rotational actuators driving the "shoulder" and "elbow" joints. In particular, to avoid occlusion of the workspace by the elbow actuator, both actuators are placed at the shoulder joint, and the elbow actuator uses a belt transmission to move the elbow joint. As before, all rigid parts are made from Plexiglas. Advantages of this structure are that, depending on how the arenas grid is arranged around the base, it is possible for the workspace to cover almost the entire surface taken up by the structure. This could be achieved by placing the base at the center of the arenas grid. A disadvantage is that this structure might be more difficult to control due to the more complicated kinematics. The main disadvantage of this structure is that it is more difficult to fabricate: notably, it was assumed that it would be difficult to fabricate a functioning belt transmission, which can be very sensitive to poor manufacturing tolerances.

Finally, a parallel RR structure was considered. An example implementation is shown in Figure 7a. The structure is made up of two "arms" joined at the end, with rotational actuators driving the two shoulder joints. It is more difficult to visualize, but the motion of the two actuators drives the end-effector to span a plane; for example, a straight line can be drawn by driving the two actuators

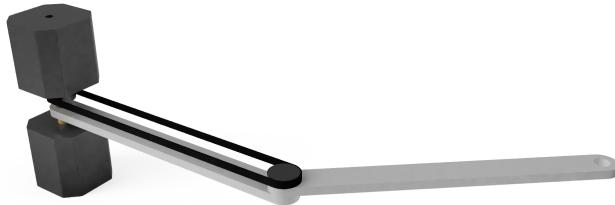


Figure 6: Example implementation of a serial RR structure

at the same speed in opposite directions.

This structure can be viewed as a parallel implementation of a RR-serial structure, or to some degree as a 2D generalization of a Delta structure (used in the Savall paper [2]). In fact, this is a well-studied robotic structure, which has also been referred to as a pantograph structure. This structure was first developed in 2005 by Campion et al. [4], and was originally designed for haptics implementations to provide force feedback.

Compared to the serial-RR structure, this structure has the advantage that it does not need a transmission to function. Additionally, the parallel construction makes it more robust to positional errors at the actuated joints. However, it still has the disadvantage that the control is more difficult to program. In particular, it is more difficult to generate a trajectory where the end-effector moves in a straight line. Additionally, this structure uses the space less efficiently than the serial-RR: many positions, notably positions close to the base, cannot be reached.

Despite its disadvantages, the pantograph structure remains a good option. To remedy the issue of the limited workspace, a solution might be to add a linear actuator. By mounting the pantograph onto an actuated linear stage, the structure can more efficiently span a rectangular workspace; this is discussed further in Section 2.4. Additionally, the fact of over-actuating the structure makes it possible to generate more complex trajectories, for example such that joints (which cannot be made transparent) move along the gaps between arenas, therefore do not occlude the recordings.

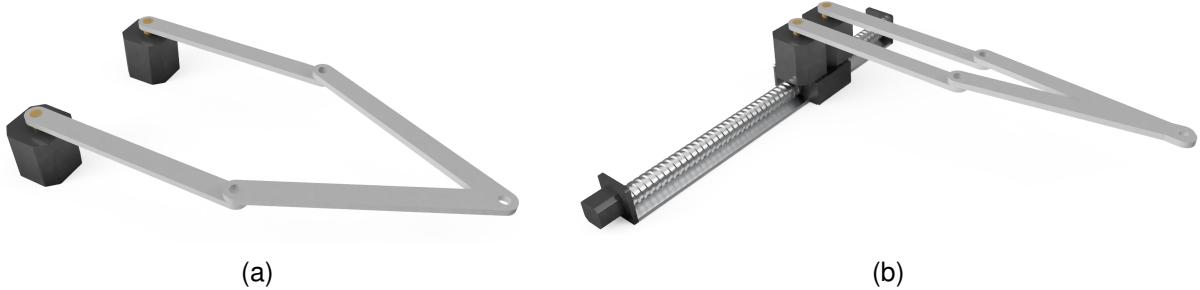


Figure 7: Example implementations of a pantograph structure: (a) normal structure, (b) over-actuated with a linear actuator

Table 2 summarizes the pros and cons of the structures discussed in this section. Based on this analysis, it was chosen to use the over-actuated pantograph structure, as it provides a good compromise between space efficiency and feasibility of fabricating it.

Structure	Space efficiency	Easy to fabricate	Easy to program
Cartesian serial	$\times$	✓	✓
Cartesian parallel	✓	$\times$	✓
RR-serial	✓	$\times$	$\times$
Overactuated pantograph	✓	✓	$\times$

Table 2: Summary of pros and cons of the robotic structures considered

## 2.4 Optimization of the Pantograph Design

Unlike with Cartesian structures, for the pantograph structure it is difficult to visualize the shape of the workspace. The pantograph structure has three free geometric parameters: the length of the upper-arms  $L_a$ , the length of the forearms  $L_{fa}$ , and the distance between the two shoulder joints  $d$ . To gain a more complete understanding of how the geometric parameters of the pantograph affect its workspace, a numerical simulation was created in MATLAB. The kinematics are taken from existing literature, all other aspects were written from scratch.

The first part of the simulation is the script `panto_dir` to compute the forward kinematics, i.e. the position of the end-effector as a function of the motor joint angles. Calculations are taken from the initial paper by Campion et al. proposing the pantograph design [4]. Next, in the script `wksp_fun` joint angles are swept from 0 to  $180^\circ$ , the corresponding position of the end-effector is found, and if this position is not a singularity then it is stored. The shape of the workspace for user-input geometrical parameters  $L_a$ ,  $L_{fa}$ ,  $d$  is then plotted.

Figure 8 gives the calculated workspaces for some sample geometrical parameters, for an angle sweep step of  $2^\circ$ .

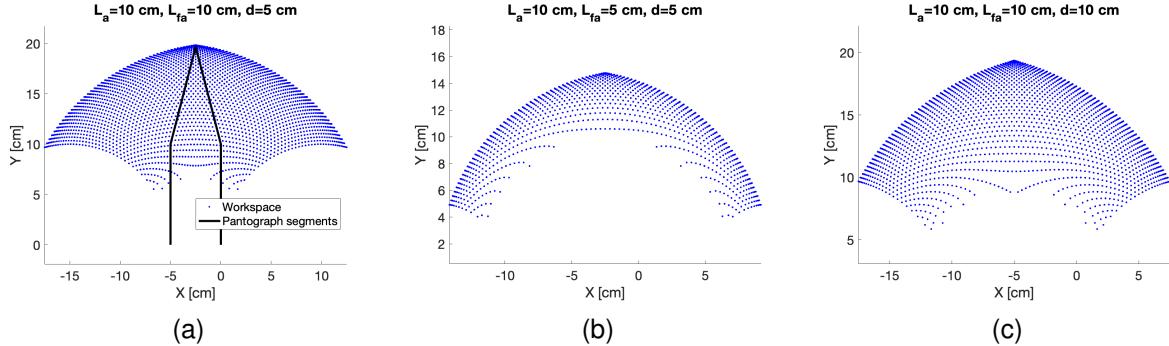


Figure 8: Example workspaces for the pantograph; (a) shows how the workspace is oriented relative to the pantograph segments; for each figure, each dot corresponds to a step of 2 degrees of a single shoulder joint

It can be observed that the workspace is wedge-shaped, with an oval "dead zone" centered at the base, which delimits the area where the end-effector enters a singularity position. A first important observation is that the use of the linear actuator will indeed increase the spatial efficiency: indeed, the workspace reaches lower (closer to the base) at its center; using the linear actuator eliminates using the edges of the workspace, where a lot of space is wasted. It can also be noted that this

structure has a variable resolution: farthest from the base, there is the highest positional resolution (for 10 cm arms and forearms, a  $2^\circ$  step moves the end-effector 0.1 mm), and closer to the base there is a lower positional resolution (the same  $2^\circ$  step moves the end-effector 0.2 mm).

Comparing Figures 8a and 8b, it can be seen that decreasing  $L_{fa}$  compared to  $L_a$  increases the radius of the dead zone at the bottom of the workspace. Comparing Figures 8a and 8c, it can be seen that increasing  $d$  widens the dead zone at the bottom of the workspace.

The next step is to evaluate the spatial efficiency of the structure, i.e. how much of the total space taken up makes up the workspace. As it can be observed in Figure 8, the efficiency is limited by the dead zone between the base and the bottom of the workspace.

It is considered that the pantograph structure will be placed on a linear stage, as discussed in Section 2.3. With this structure, it is assumed that the stage carries the pantograph up to the desired column of arenas, and the pantograph operates only on that strip. Recall that arenas are 5 cm wide; therefore, the pantograph structure should optimize the space usage only within that 5 cm strip. The useful space within a 5 cm strip is calculated in the script `wksp_fun_rec`. Figure 9 shows the useful space for the same geometrical parameters as Figure 8a.

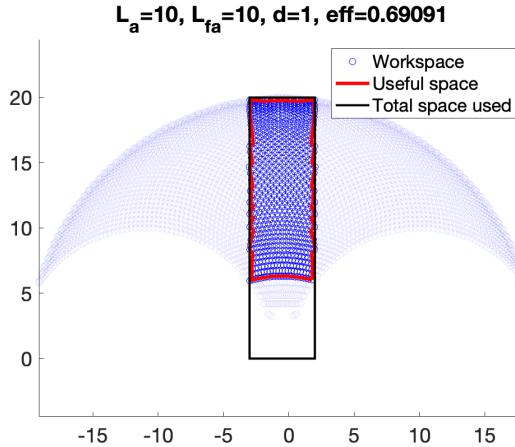


Figure 9: Example of the rectangle efficiency

Using the previous scripts, it is possible to compute the optimal geometrical parameters to maximize the size of the workspace within a 5 cm strip. This is done in the script `eff_rect` by exhaustively computing the spatial efficiency for a range of possible geometrical values. In particular, the absolute values of  $L_a$  and  $L_{fa}$  do not impact the efficiency (making both longer increases the size of the workspace, but not the efficiency), only their relative value  $r = \frac{L_a}{L_{fa}}$  has an effect; therefore, the simulation test parameters  $d$  and  $r$ . Figure 10 shows the computed spatial efficiency as a function of these two geometrical parameters. It was found that the efficiency has a peak value of approximately 80%. The optimal  $r$  and  $d$  values seem to be linearly related; a first-order polynomial approximation found that optimal parameters are linked by:

$$r_{opt} = 0.03 \cdot d_{opt} + 0.97 \quad (1)$$

All MATLAB scripts were submitted with the other files for this project.

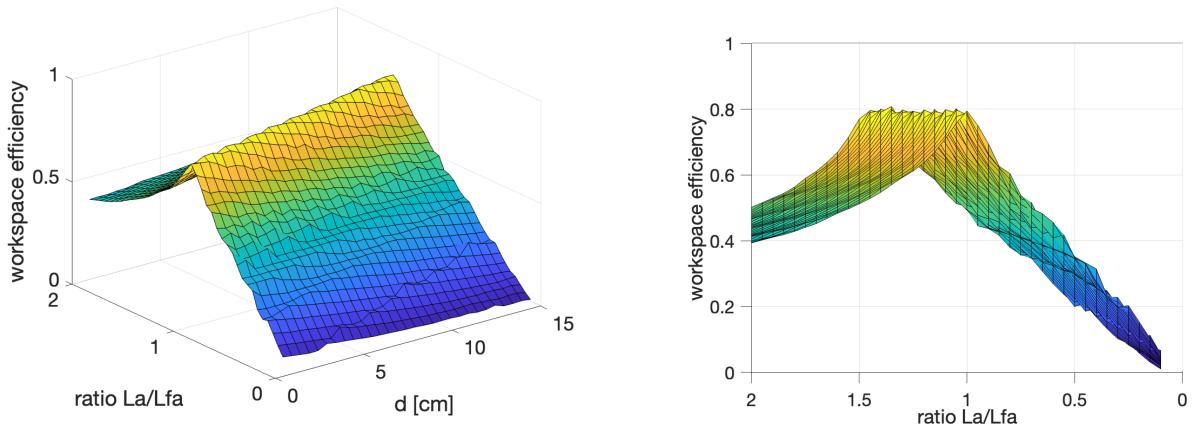


Figure 10: Rectangular spatial efficiency as a function of the two geometrical parameters  $r$  and  $d$

## 2.5 Detailed Pantograph Design

Once established that a pantograph structure should be used, the details of the mechanical assembly must be determined. Certain parts were inspired by a report from a student project at Northwestern University [5], which details the construction of a pantograph with steel parts.

Figure 11 shows the assembly which was chosen. Segments are made from laser-cut 4 mm thick Plexiglas; this thickness was chosen as a good compromise of weight and mechanical stiffness (resistance to vertical deflection). The upper arm section is made of two sheets, which among other reasons is to increase the stiffness. A screw with four bolts which runs through both sheets is used to ensure the correct spacing of the two sheets. The screw was chosen for easy assembly and disassembly; in a final version, spacing could be ensured for example by fixing several small sections of Plexiglas between the two sheets, using a transparent adhesive. Axial bearings are used over and under the screw connecting the lower sheet to the base. Indeed, the screw at this point needs to be tightened strongly, to prevent the structure from tipping forward. The second upper-arm sheet above the forearm sheet at the elbow joint ensures that the screw at the elbow joint stays straight, without having to tighten this screw; consequently, the elbow joint does not need bearings. The motor shaft is connected to the upper sheet of the arm, and the motor is fixed on a 3D-printed base.

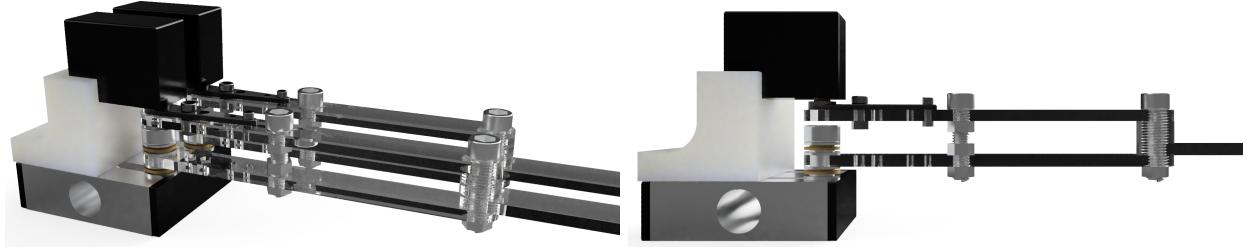


Figure 11: Render of the planned assembly

## 2.6 Dimensions and Arenas Grid

The linear stage which was purchased (see Section 3.1) has a spacing of 3 cm between the openings in which the screws for the shoulder joints are placed. For this distance, the optimal ratio of upper- to forearm length according to equation 1 is  $d_{opt} = 1.067$ . For the sake and simplicity, and to make it possible to use the same part for the upper arms and for the forearms, it was chosen to make the upper- and forearm segments the same length. According to the MATLAB simulation, this gives a space efficiency of 76.91%.

As discussed in Section 2.1, it was chosen to use the arenas made for the Optobot system. The design which was fabricated for this project is intended to be used with a grid of arenas three tall and four wide, with the projections from the arenas pointing horizontally. Allowing for a 5 mm spacing between arenas, this makes a grid with effective dimensions 17 cm by 31.1 cm. This dimension is coherent with the linear stage which was purchased, which has a working length of 37 cm.

For the grid structure mentioned above, and imposing that the upper- and forearm segments are the same length, it was chosen to use 12cm upper- and forearm segments, which give a useful workspace of length 18.42 cm. This is coherent with the workspace height which was chosen.

Ultimately, the arenas grid was not fabricated. It was assumed that in the context of any future work on this project, the grid would be re-built to suit the requirements determined at that time.

## 3 Fabrication and Programming

### 3.1 Parts Selection

Table 3 gives the complete list of parts used for this project.

It was chosen to use an Arduino Uno microcontroller to control the system. For the relatively simple computations necessary to drive the system, this is sufficient. Additionally, Arduino microcontrollers can be programmed quickly and easily, and are therefore excellent for rapid testing.

As mentioned in previous sections, the pantograph structures are mounted on an actuated linear stage. It was chosen to use a relatively inexpensive commercial linear stage actuated by a stepper motor, which is designed for use in 3D-printers or CNC machines. Compared to more sophisticated systems, such as the extremely-high-precision stage used in the Optobot gantry, these are less expensive and easier to program. The manufacturer for the chosen model lists the repeat accuracy as 30 micron, which is suitable for this application.

The shoulder joints of the pantograph need to be actuated by precise rotational motors. As these joints do not exceed a rotation of 180°, it is possible to use servos, in addition to stepper motors. Servo motors were chosen over steppers, because they do not need an external driver board as stepper motors do, and because they are easier to program (servos receive as input the desired final position, while steppers need to receive correctly-timed pulses up to the final position). This choice was made to maximize the likelihood of completing the project within the time limit.

As speed is not a constraint for this project, the torque of the servos is relatively arbitrary: lower-torque servos will simply move slower. To have the most flexibility, and due to the parts which were

Part name	Quantity	Supplier	Price [CHF]	Fabrication
<i>Purchased</i>				
Spektrum A6030 servo motor	2	Technik-Hobby	50	
SainSmart Motorized linear stage	1	TechStudio.ch	180	
Seafront stepper motor driver	1	TechStudio.ch	150	
Reely 6mm axial bearings	4	Conrad	8	
<i>Fabricated</i>				
Upper-arm segments	4			Laser-cut
Forearm segments	2			Laser-cut
Spacers	5			Laser-cut
Motors base	1			3D-printed
<i>Sourced in the lab</i>				
Arduino Uno microcontroller	1		30	
Mini breadboard	1		10	
5V power supply	1		25	
40V power supply	1		25	
Male-male jumper wires	20			
DC coaxial terminal adapter	2			
M6x20 Screws	3			
M6x25 Screws	4			
M6 Nuts	9			

Table 3: Complete list of parts

available within the time constraints, servos with  $20 \text{ kg} \cdot \text{cm}$  torque were used.

The segments for the arms were designed in Fusion 360 and cut using the Trotec Speedy 400 laser cutter at EPFL. A base on which to mount the motors was designed in Fusion 360 and 3D-printed in black ASA at the AFA at EPFL.

Final CAD files for the laser-cut and 3D-printed parts were submitted with the other files for this project.

The parts for this project cost a total 552 CHF, which remains relatively low-cost compared to, for example, the high-precision linear stages used for the Optobot. Note also that the linear stage and the stepper controller, which make up the majority of the price, were purchased (due to availability problems with other retailers) from a retailer who had them at a relatively high price; however, equivalent parts are available at a significantly lower price from main online retailers such as Amazon.

### 3.2 Electronics

Figure 12 shows the complete wiring diagram for the setup. In particular, the controller for the stepper motor is connected to an external 40 V power source. The two servos are connected to an external 5 V power source, as the Arduino pins cannot deliver a strong enough current, and are connected with a common ground with the Arduino.

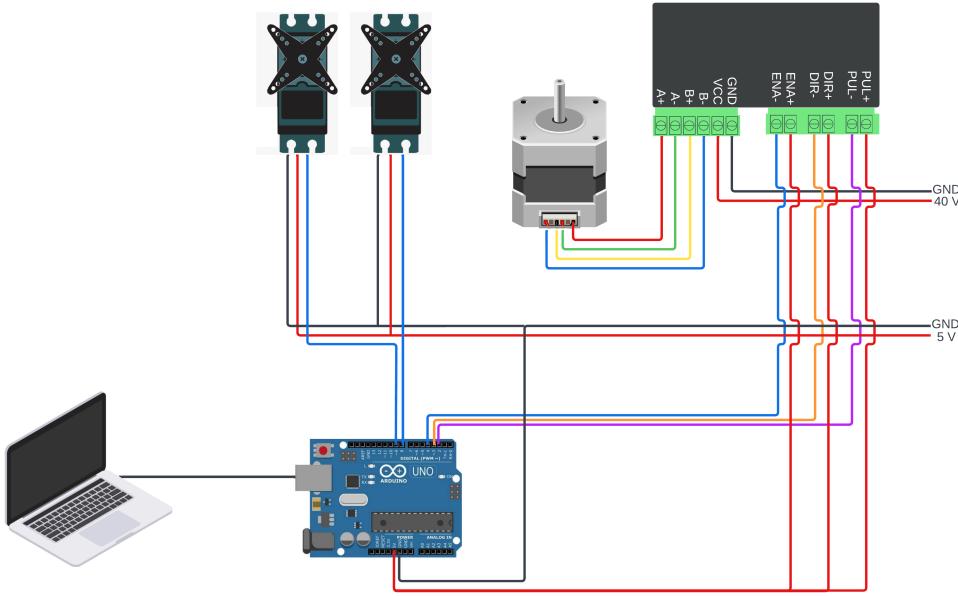


Figure 12: Wiring diagram for the setup

The servo motors were calibrated to ensure maximal accuracy. Servo motors are controlled by writing a high to the signal input for a duration linearly proportional to the desired angle. Usually  $500\text{ }\mu\text{s}$  maps to approximately  $0^\circ$  and  $2500\text{ }\mu\text{s}$  to  $180^\circ$ ; however, the exact values differ between individual servos. A calibration method was employed, inspired by an article on [Hackster.com](#). An Arduino program `servo_manual_pwm` was written which allows the user to input in the serial monitor the high-time of the signal, which is then applied to the servo. Using this program, the corresponding high-time for a range of angles was determined using a setup as shown in Figure 13a. A first-order linear approximation was then used to determine the exact mapping used by each servo (see Figures 13b and 13c). The final limit values found were  $644\text{ }\mu\text{s}$  and  $2316\text{ }\mu\text{s}$  for the left servo, and  $643\text{ }\mu\text{s}$  and  $2307\text{ }\mu\text{s}$  for the right servo; equivalently, the  $90^\circ$  positions were set, respectively, to  $1480\text{ }\mu\text{s}$  and  $1482\text{ }\mu\text{s}$ , with a gain of  $92.8$  and  $93.2\text{ }\mu\text{s-per-degree}$ .

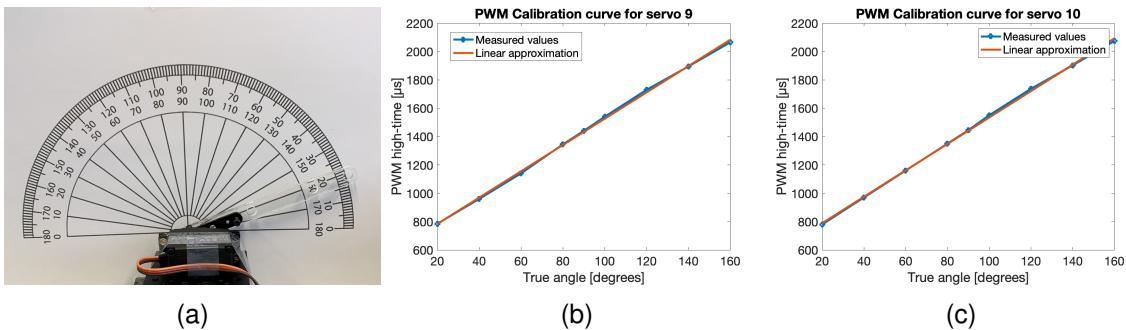


Figure 13: Servo calibration: (a) Setup employed (b) Measured values for the left servo (c) Measured values for the right servo

### 3.3 Programming

The structure is controlled by an Arduino Uno microcontroller programmed in C++ through the Arduino IDE. In particular, the Servo library was used to control the servos, and the AccelStepper library to control the stepper motor of the linear stage.

Structs `motor_angles` and `coordinate` were created to allow functions to pass and return the couplets. A function `inverse_kin` is used to compute the inverse kinematics for the system, i.e. to compute the corresponding motor angles for a given desired end-effector position. The calculations for this are taken from the Campion paper [4].

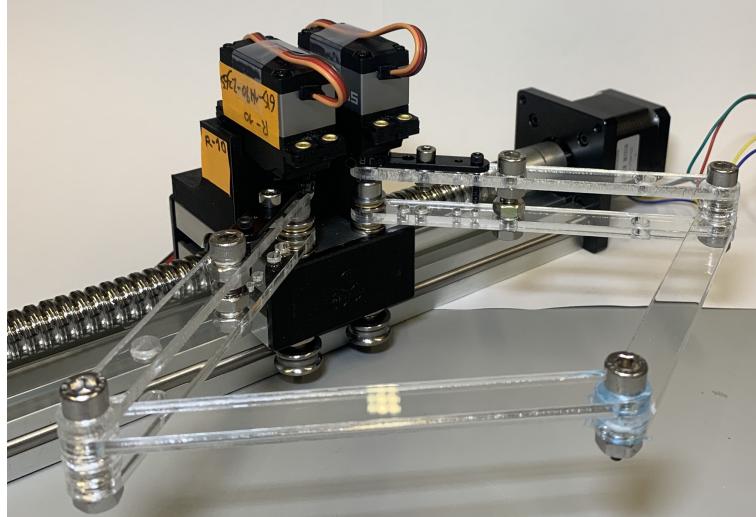
A function `generate_vline` is used to generate the motor trajectory for the end-effector to draw a straight vertical line from the current position to a given height. Note that for the pantograph structure, this is not as easy as with Cartesian structures: whereas with Cartesian structures, a straight line is drawn by simply driving the corresponding actuator, with the pantograph the two motors must sequentially be stepped one after the other to intermediate positions, to ensure that the line is not uneven due to one motor being ahead of the other. As onboard storage is limited on Arduino, it may not be possible to calculate all intermediate positions in advance; instead, intermediate positions are calculated recursively from the current position until the current position is equal to the target position.

The complete C++ code for the Arduino was submitted with the other files for this project.

## 4 Results

### 4.1 Final Fabricated Structure

Figure 14 shows the final fabricated structure.



(a)

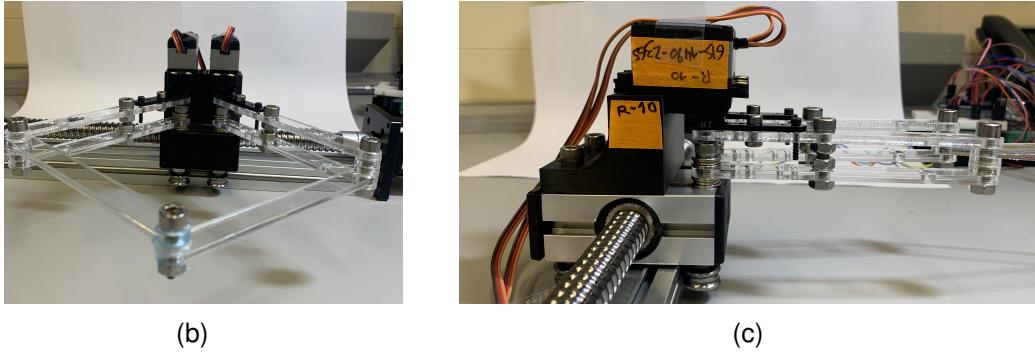


Figure 14: Final fabricated pantograph structure

## 4.2 Testing

It was observed that the structure was capable of reaching all points in the workspace which was computed from the MATLAB simulation. In particular, it was possible to reach the highest and lowest positions  $Y=6.5$  cm and  $Y=23.5$  cm from the centre of the workspace.

The execution time for the trajectory generation was not a problem: it was measured that generation of each point in the trajectory, excluding the time to step the motors, took 980 to 1000 microseconds.

It was found that the Plexiglas limbs did not cause a significant occlusion of the workspace; figure 15 shows an example occlusion.

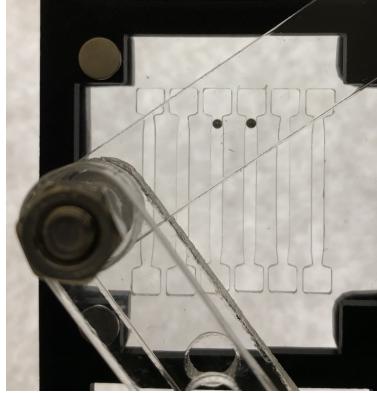


Figure 15: View of an arena occluded by a Plexiglas pantograph segment

In a first test, the pantograph was used to move balls in a single arena. It was found that the structure was able to move the ball as desired. Over this trajectory, the end-effector has a maximum sideways error of approximately 1.5 mm, measured from the video taken of the movement.

It was also found that the structure was capable of moving a ball in one arena without catching the ball in a nearby arena.

The absolute accuracy was tested by programming the pantograph to move to three different positions along the vertical line at the centre of the workspace: at 10, 15 and 20 cm away from the

base. In the referential of the pantograph (which is the same as in the Campion paper [4]), this corresponds to positions (-1.5, 10), (-1.5, 15), (-1.5, 20). It was found that for these three positions, there was an absolute positional error of the end-effector of up to 2.5 cm.

To correct for this absolute error, the pantograph was instead programmed to move to slightly different positions which would correspond to the desired true positions. Figure 16 shows the results when the end-effector was programmed to move to positions (-1.5, 8), (-1.75, 14), (-2, 19.5). With this correction, a precision of less than 1.5 cm was achieved. The biggest error occurred when a same position was visited in two different directions (going toward and away from the base); this can be seen comparing Figures 16b and 16d. This issue was attributed to the backlash of the gears inside the servos.

A more complete study, for example measuring more accurately and over a greater range of positions, would be required to better understand the absolute positional error.

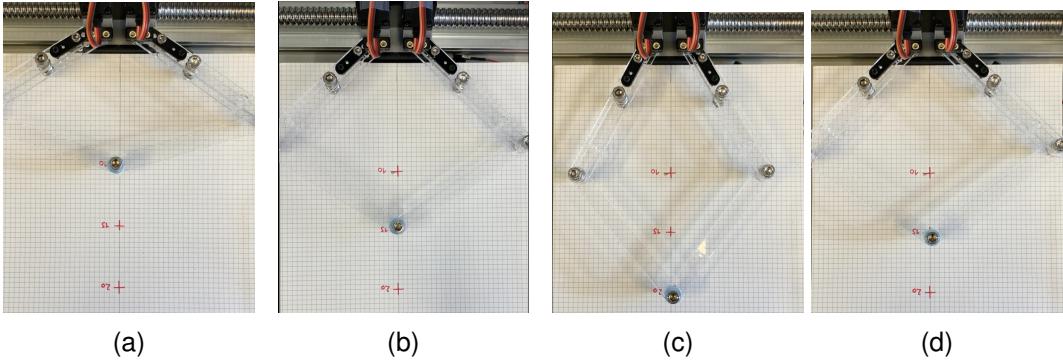


Figure 16: Movement over the full range of motion for the central X position; (a) to (c) are going away from the base and (d) is coming towards the base. Squares on the paper have sides 4mm long.

Videos of the performance were submitted with the other files for this project.

## 5 Discussion

The goal of this project was to design, fabricate and program a pantograph robot structure to automate experimentation tasks at the millimetre-scale. The main constraints were to minimize the workspace occlusion, and to maximize the efficient use of space. A MATLAB simulation was successfully developed from scratch to thoroughly characterize the workspace and parameters affecting it. The structure was successfully fabricated, and the mechanics were found not to have major issues. In testing, it was found that the pantograph was capable of carrying out ball-reset tasks with accuracy within the permitted values.

It was found that the pantograph has an error in absolute positional accuracy beyond the tolerated values. This is attributed to the servos: there was some error due to backlash in the gears; it is also likely that there was an error intrinsic to the servos. Indeed, the servo architecture chosen for this project uses a potentiometer to measure the position of the motor shaft, which may not reach the precision desired for this project. Additionally, it is possible that the function in the Arduino code to generate the PWM signal to communicate with the servo, which came from the Arduino Servo

library, may have a limited precision.

Although the absolute accuracy was outside of the constraints, it was found that the end-effector is capable of performing ball-reset tasks as desired, as long as it is first moved to the correct starting position. Consequently, for future work on this project, it would be possible to keep the servo motors as they are, but to implement an absolute position correction in software. For example, the software position corresponding to each cell could be measured in advance, then this position, instead of the true position, would be sent to the servos to reach a given arena in the grid.

However, to remain as versatile as possible, it is recommended that future versions of this project should change the motors. Good options might be more advanced high-accuracy servos which use an encoder instead of a potentiometer (see for example [Dynamixel](#), which however use the more complex I2C communication protocol), or stepper motors.

Another aspect to develop would be fabricating the arenas grid discussed in Section 2.6. This was not developed in this project, partly to focus on the robotic structure, and partly because the final structure of the grid will depend on how the grid is used: for example, the recording and lighting setup may impose additional constraints.

A more long-term goal for this project will be to implement recording, experiment-end detection, and triggering of the pantograph into a program running on a single microcontroller. This would allow complete automation of the recording process.

Timings for this project were difficult, notably due to delays in receiving parts and due to a new direction for the project being chosen after the midterm presentation; consequently, the original Gantt chart (see Figure 17) was not followed. Despite this, it was possible to finish the project within the time limit.

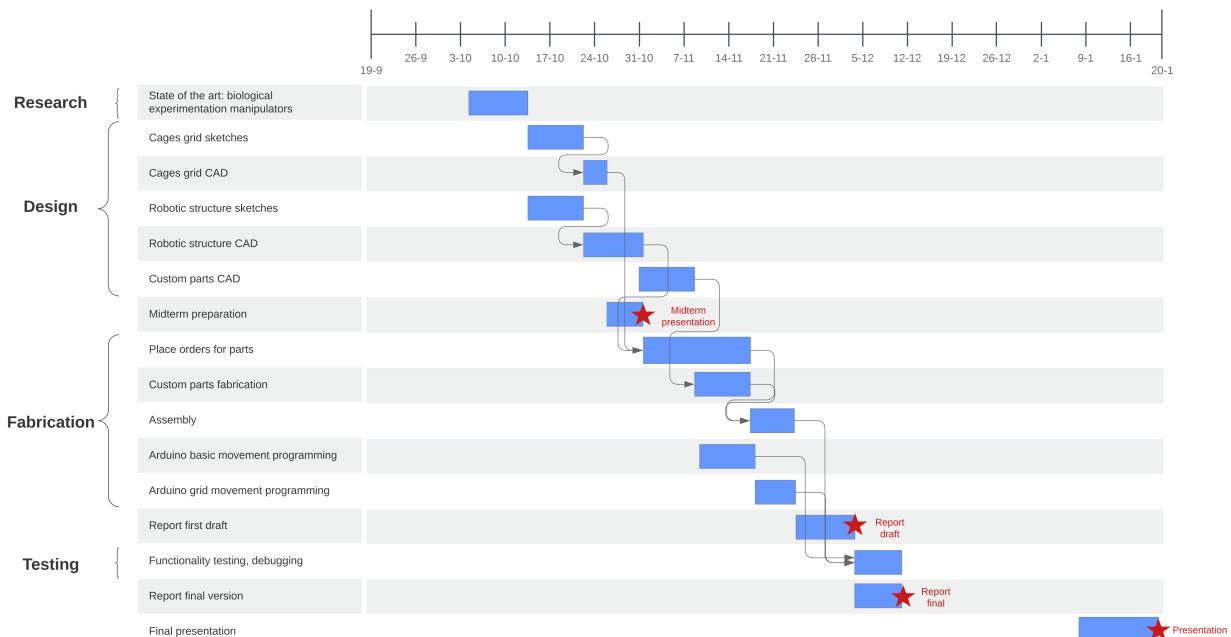


Figure 17: Original Gantt chart for the project

## **Acknowledgments**

I would like to thank my supervisor Matthias Durrieu for his help and guidance throughout the semester. Thank you also to Victor Rios for the help and technical advice, and to Gizem Ödzil for the help with logistics. I would like to thank all the staff of the Neuroengineering Laboratory for making me feel welcome at the lab and for the interest in my project.

I would like to thank Prof. Pavan Ramdy for giving me the chance to work at the Neuroengineering Laboratory and for giving me the opportunity to take on this project.

## References

- [1] Amey S. Joshi et al. "Multiscale, multi-perspective imaging assisted robotic microinjection of 3D biological structures". In: *Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual International Conference 2021* (Nov. 2021), pp. 4844–4850. ISSN: 2694-0604. DOI: 10.1109/EMBC46164.2021.9630858.
- [2] Joan Savall et al. "Dexterous robotic manipulation of alert adult *Drosophila* for high-content experimentation". In: *Nature Methods* 12.7 (July 2015), pp. 657–660. ISSN: 1548-7105. DOI: 10.1038/nmeth.3410.
- [3] Kentaro Yoshioka et al. *Through the Looking Glass: Diminishing Occlusions in Robot Vision Systems with Mirror Reflections*. Aug. 30, 2021. DOI: 10.48550/arXiv.2108.13599. arXiv: 2108.13599[cs]. URL: <http://arxiv.org/abs/2108.13599> (visited on 01/12/2023).
- [4] G. Campion, Qi Wang, and V. Hayward. "The Pantograph Mk-II: a haptic instrument". In: *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems. ISSN: 2153-0866. Aug. 2005, pp. 193–198. DOI: 10.1109/IROS.2005.1545066.
- [5] *Design and Control of a Pantograph Robot - Northwestern Mechatronics Wiki*. URL: [http://hades.mech.northwestern.edu/index.php/Design\\_and\\_Control\\_of\\_a\\_Pantograph\\_Robot](http://hades.mech.northwestern.edu/index.php/Design_and_Control_of_a_Pantograph_Robot) (visited on 01/10/2023).