

An Open-Source Tele-Operated Mobile Manipulator: CHAP V1.

Guido Bugmann¹, Dominic Cassidy¹, Paul Doyle², Arunaganesan Swaminathan¹, and Khushdeep Singh Mann¹

¹ Plymouth University, Plymouth PL4 8AA, UK

² Hereward College, Coventry CV4 9SW, UK

gbugmann@plymouth.ac.uk; dominic.cassidy93@gmail.com;

Paul.Doyle@hereward.ac.uk; arunarunachala@gmail.com;

khushdeepvnit@gmail.com

Abstract. Teleoperated mobile manipulators are of use for disabled people and for the wider public interested in acting at distance. The high price of existing devices is a barrier to their diffusion. The paper reports on the first design produced in the Cheap Arm Project (CHAP). It costs less than £2000, uses easily available parts and can be assembled by anybody with basic technical skills. The manipulator can reach objects from floor-level up to shelves at a height of 170cm using a new low-cost arm design. Teleoperation is be done using a tablet, smartphone or browser. The cost could be further reduced by using different servo motors. The design and assembly instructions are made available on the open-source repository GitHub, with the hope that the community will build and improve the design. The first version has been tested in a college for disabled young people who provided initial recommendations for improvement.

Keywords: teleoperation, teleoperated, disability, assistive, robot

1 Introduction

Teleoperated mobile manipulators are of use for disabled people and for the wider public interested in acting at distance. We conducted this study with the disabled user in mind. The high price of existing devices significantly makes them inaccessible to most users. The most common designs consist of a mobile robot platform onto which a robot manipulator is attached. Manipulators usually have a limited reach and therefore need to be mounted on some sort of lifting device to cover a useful vertical operation range. An early example of such devices is the EL_E built at Georgia Tech. [1]. A nice study of a number of combinations of mobile platforms and manipulators can be found in [2], where a new low-cost combination is described, costing around \$4400 without a lifting mechanism. These design are mainly aimed at autonomy, where the user issues a high level command that is then executed using on-board planning and sensing. In this

paper we initially only consider teleoperation, giving the user full control of the device. The system supports this by providing a simple web interface with visual feedback. This approach is evaluated through initial user tests.

2 Design

2.1 Concepts

It was important to minimize the cost of parts. We noted that commercial robot manipulators all aim at precision and reproducibility. These devices are essentially designed to operate “blindly”. This is a significant cause for costs, due to the need for precisely engineered parts able to generate precise displacements for whatever load and position within their operational envelope. In our approach, we decided that such precision was not needed, as the user was there to correct the observed positions. Placing the user in the control loop (a form of “shared autonomy”) was a way to reduce costs.

Placing a conventional robot manipulator on top of a mobile base creates a highly redundant system, as both manipulator and base provide x-y positioning. To save the cost of redundant servomotors, we used an omnidirectional base also controlling the x-y position of the gripper. All that was needed was to provide a mechanism to control the height of the gripper.

2.2 Arm and lift

We aimed at a device able to lift common objects used in everyday life, of which the heaviest is probably a 2 liter bottle of water or fizzy drinks. So, we set the target to lift a maximum weight of 3 Kg, to have some margin. We aimed at a similar reach to the human arm, i.e. around 75cm (including gripper length). The arm needed to reach the ground level to pick up objects on the floor, as well as reaching the height of head-height shelves. In order to eliminate an expensive elbow joint, we thought of a simple lever with axis appropriately placed to offer the whole range of heights (figure 1A). In order to reduce the length of the lever protruding at the back of the robot, we introduced the concept of a “virtual rotation point”. In this approach, two points of the lever are actuated to cause the gripper to execute the desired trajectory. It turns out that, with an arm length of 50cm (without gripper) and a protruding length of 25cm, the back lever moves by 1/2 of the displacement of the lifting point. Hence both movements could be achieved with a double pulley activated by the same servomotor (figure 1B). Both pulleys have opposed actions: while the larger diameter one reels the cable in to lift the arm, the smaller one reels the cable out at 1/2 of the speed to let the back lever raise too. With this arrangement, the back lever actually reduces the torque required from the servomotor. The lifting force was further halved by using a pulley on both cables. Low stretch 2mm Dyneema ropes were used. For more details, see the instruction notes on the Github site: <https://mobile-chap.github.io/Web/>. A Dynamixel RX-64 servo was used that provided plenty of speed and torque.

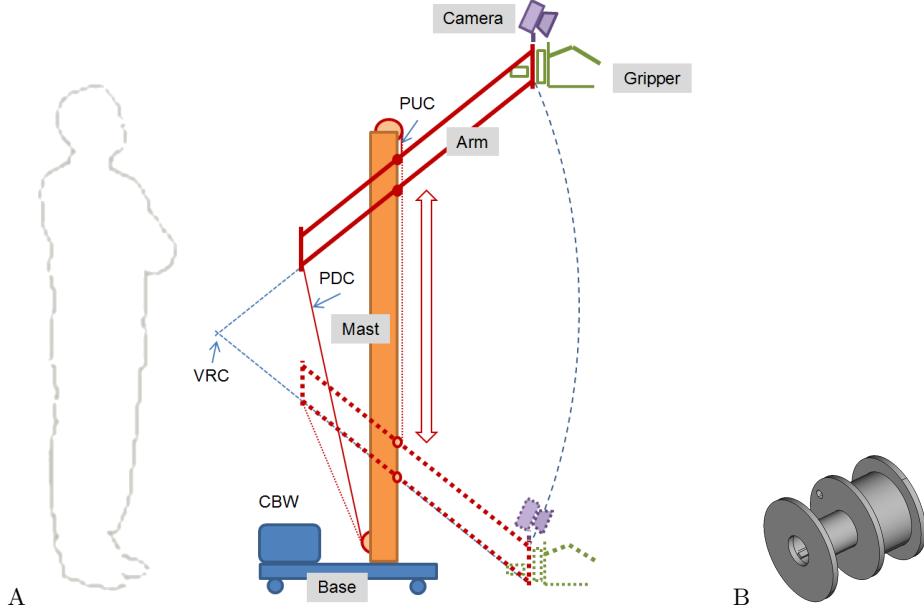


Fig. 1. A. Basic elements of the CHAP robot: VRC: Virtual Rotation Centre; PUC: Pull-Up Cable; PDC: Pull-Down Cable; CBW: Counter Balancing Weight. B. Lifting pulley at the base of the mast.

The arm is designed as a four-bar parallelogram gliding in rails on each side of the mast. The design was a parallelogram to ensure that the end plate on which the gripper is attached is always vertical. Therefore, for instance, a cup picked up off the floor would automatically stay level during the lifting process.

The mast was built out of wood and plywood with a U-profile for rigidity. It was attached using four cables linking it to the base, similar to shrouds on a sailing boat. This ensured maximal strength for a very small weight. Without shrouds, the base of the mast would have had to be attached to the mobile platform using a very strong and rigid fixation.

Overall, the robot weighs 11Kg and can easily be transported. If needed, the mast can be detached from the base by releasing the four shroud cables (using wing nuts).

2.3 Base

The base is an omnidirectional platform with 3 motor units (figure 2). Each motor unit is 3D-printed to hold an omniwheel between two ball-bearings and a Dynamixel MX-28 servomotor. These servos are addressed using a serial protocol and can be daisy-chained. They can be controlled in position mode or in speed control mode. They allow for a precise and smooth positioning of the base. The plate is made of 25mm thick plywood that can be laser-cut. The heart-shape

enables it to approach furniture parts more closely. The front wheels are placed 10cm ahead of the front of the mast. A counterweight is placed at the back. The counterweight is a flexible water container that can be filled up to 5L when in use, and emptied when the robot is transported. The base can move in any direction and the calculation of the speed commands for each wheel is very simple (see e.g. documentation on our github site).

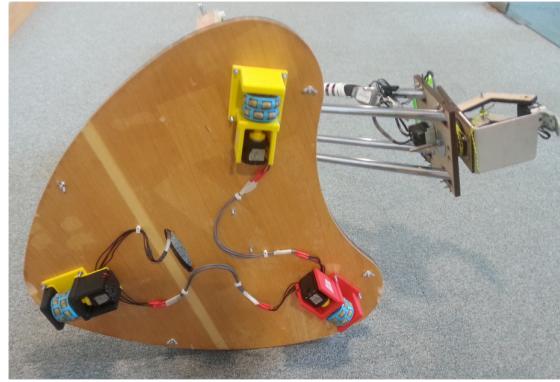


Fig. 2. Underside of the base showing three motor units. Parts of the arm can be seen in the background on the front side.

2.4 Gripper

The gripper had a design as simple as possible to support tasks of every day life, such as lifting a bottle, poring, lifting small objects from the floor, carrying small food trays, e.g. for microwave cooking. Although the initial impulse was to design a multi-fingered human-like hand, the aim here was to produce a gripper as simple as possible, using laser-cut parts, to enable initial testing and the collection of data for the real specification of the gripper. Nguyen et al (2008) [3] noted that indoor objects are usually found on flat surfaces, and our gripper design is probably adequate for the task of collecting such.

The gripper is essentially a fixed 3mm aluminium plate that can be rotated into a vertical or horizontal position. Then, a V-shaped knuckle presses the object against the plate and hold it in place for lifting. The edge of the plate is filed down to a thickness of 1 mm to help collect small objects and plates. Both plate and knuckle are covered with rubber to increase grip. We measured how much force was needed to press a 3Kg bottle against the plate to stop it from slipping (15 Kg), and designed the gripping mechanism to provide that force. It uses a cable pulling the knuckle and a spring opening the hand. When the plate is horizontal, the small “finger” at the end of claw can be used to drag the object onto the plate. This finger is also useful for holding food containers, e.g.



Fig. 3. Gripper grabbing a bottle, a bowl and a LiPo battery.

a bowl (figure 3). The gripper uses three dynamixel servos, 2 x AX-12 for the single finger and the wrist rotation and one MX-28 for the closing mechanism.

2.5 Software and User Interface

The developed control system was centered around a Raspberry Pi and direct Dynamixel communication through two USB2Dynamixels. Three independent sub-systems ran on the Raspberry Pi in parallel allowing full teleoperation of the robot by a user on the same Wi-Fi connection using any touch or non-touch device. The sub-systems were as follows:

- Web Server: Hosted HTML/CSS/JavaScript client teleoperation web page.
- Control Listener: Listened using a web-socket for client commands and passed these to the correct Dynamixels. Multiple conditional arguments were used in order to check the name of the incoming command and correctly process it.
- Stream Server: Hosted a live image stream (captured by OpenCV) from the camera mounted on the robot gripper.

The sub-systems were all built on the Python (2.7) programming language, with JavaScript used by the client for detecting user input and communicating these to the control listener.

This simple system enabled high concurrence with limited bottle necks between systems. Each sub system was capable of running fully independently and simply waited for either passive usage (web and stream server), or direct inputs (control listener). Multiple Python modules were used to enable this behaviour cleanly. Specifically, Flask was used for hosting the web server, while Autobahn enabled the control listener to directly connect to the client using a web socket. Finally, Pypot enabled direct communication with the used Dynamixels individually and as groups. The live web stream was accessed by the client-loaded web-page and presented on the user interface background.

Figure 4 shows the placement of the controls that appear transparently on top of the video from the on-board camera when used (touching the screen will make the controls appear).

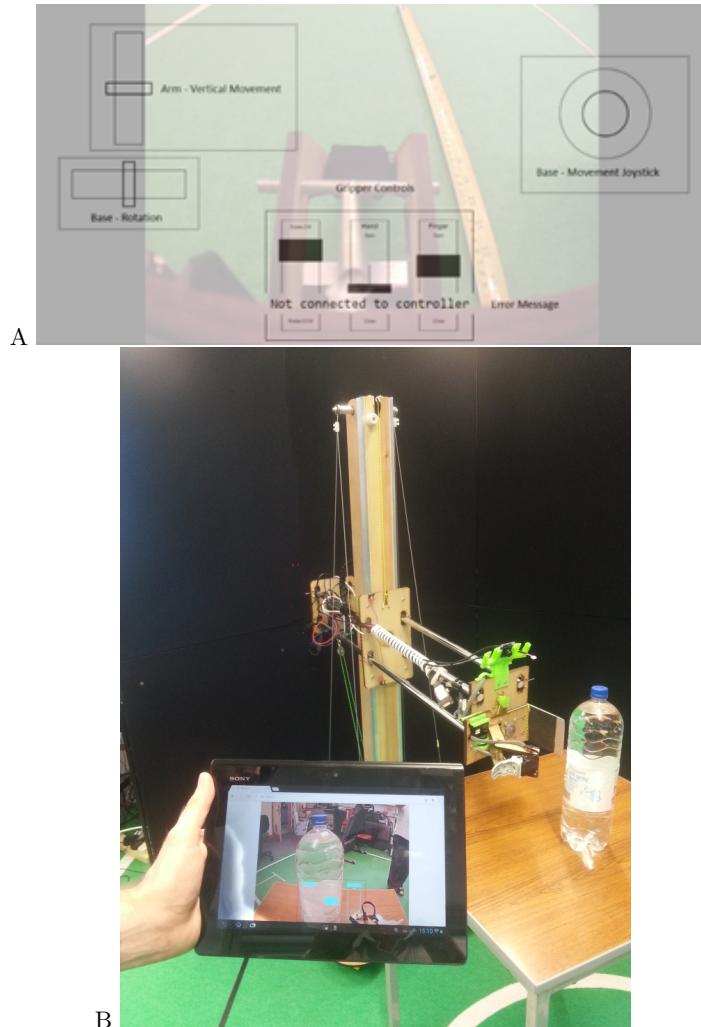


Fig. 4. A. User interface concept. The left controls act on the rotation of the base and the lift. The right control acts as a joystick to control the X-Y motion of the base. The three central controls act on the gripper. B. View of the interface in action.

3 User testing

3.1 Basic usability

Four participants were given a short (less than 2 minutes) visual introduction to the control system. The gripper and user interface were tested in an object handling task. Experimentation required three objects to be picked up and moved a short distance.

- Task 1: Roll of wire (200g)
- Task 2: 500ml bottle of water (600g)
- Task 3: Book (400g)

The three items used were of very different shapes. The complexity of the objects required effective use of the robot's 4 degrees of freedom and, as found during the experiments, all 4 degrees were used in response. The participants were placed in a seated position perpendicular to the work area 1 meter from a table and the robot. Objects were placed on the table prior to each experiment, with the user moving them from the right side of the table to the left (with the robot starting position being perpendicular to the object). The enforced seated position was to increase reliance on the image stream, and remove advantages of improved visibility that could not be replicated between users. During a prior pilot study, the gripper failed multiple times due to overload errors in the knuckle, and occasionally the finger. As a result, the main experiments required the researcher to be ready to power cycle the specific motors upon failure (a software-only procedure has been devised since).

Timed tasks showed mixed results. With $\frac{3}{4}$ of participants failing to pick-up the book; which was a deliberately slippery surface and reflected certain weaknesses in the gripper. One participant managed to pick up the bottle within 40 seconds compared to the other three participants who had an average time of 3 minutes 12 seconds. This result was due to superior technique (combined knuckle and finger use), and reflected the extent to which human participation may lower the effectiveness of the robot due to improper technique. During each of these experiments no issues were found with the base or tower systems. The control system was reliable and worked as expected. The robot was capable of carrying all weights attempted, ranging from 100g to 2kg (during the pilot), and lifted these over 1.5m from ground level. Controller usage showed that all degrees of freedom were used throughout the experiments. No issues were found with the control system UI, although some confusion was seen in differentiating the right and left hand side controls. Controls were otherwise exact and highly responsive. The extent to which all controls were used is important as it shows that each degree of freedom was needed for object manipulation and therefore required constant access within the interface.

Some participants commented on the placement of the camera (view sometime hidden by the gripper), too many degrees of freedom to control (auto-alignment with the object was suggested), or found the gripper “crude” (shape not matching some objects).

3.2 Feedback from disabled users

For half a day, 7 disabled students at Hereward College were given the robot to try in various conditions, first in a meeting room, and then in their own rooms. Note that the camera had been moved to a more off-centre position with a view not obstructed by the gripper. After experimenting with the robot, the students were asked to answer following questions:

8 Bugmann, Cassidy, Doyle, Swaminathan, Singh Mann.

1. Would you like to have such a robot to own /use?
2. When / where/ for what tasks would you use it?
3. What would you change to the prototype you have seen?
4. How would you source such a robot:
 - (a) Build you own from provided designs/ after modifying the design and source your own parts?
 - (b) Build from a kit of pre-cut parts/3d Printed parts + source all the rest (servo+ electronics), e.g. online shopping?
 - (c) Build from a full kit?
 - (d) Buy a fully assembled robot?
 - (e) Rent a robot?
 - (f) Use a robot owned by college?

Their answers can be summarized as follows. The number of replies in indicated in brackets.

- They all wanted to have a robot.
- About sourcing the robot,
 - They were not prepared to build it themselves.
 - They would buy it assembled for themselves (111) or would be happy to use a robot provided by the school (1111).
- They would use it
 - in their rooms(111), in the kitchen (11), outdoors (1) and indoors (11), on college site (11).
 - To pick up things (111111), pick up rubbish (1), getting things that are high up (111), to cover the student up (1), turning light off (1), reaching things (111).
- What would they change?
 - Make it smaller (1111) (with the same height (111) and narrower (1), so that it is less of a hassle (1)).
 - Add a camera to help navigation (1).
 - Make the controls smoother (1). Left-right controls are confusing (1).
 - Make a coloured robot (1).
 - Have a different gripper (1) (We forgot to ask what should be different..), maybe a different gripper for different jobs (1).
 - Would be good to be able to use your own phone (1) (this student had a problem with the browser on his phone).
 - Make it faster (1) (the wheelchairs move really fast in corridors –In the common room, some commented that it was fast)

4 Discussion

Regarding the cost of the robot, a large contribution comes from the use of Dynamixel servomotors. One reason for using them was that we had a stock of them, but they are actually very convenient to control, almost directly from the USB port of a PC. Another important point is that they have a very good torque

vs speed curve that no cheaper digital servo can match. As the servos are a big contributor to the total cost, it is worth looking for alternative, but at present, the only route seems to be the design of new servos from scratch. We estimate that this would save at least £600 on the cost of parts.

Many design decisions ensured that the primary design and components of the robot were of materials and parts that can be accessed easily, and fabricated without expensive machinery. The reliance on 3D printing ensured effective and repeatable prototyping, however the limitations of this technique were also discovered, with specific difficulty caused by failures in accuracy and resolution. This was especially critical where parts were designed to fit into each other, such as those in the motor unit. While the continuous prototyping allowed by 3D printing solved or can solve these issues, this does cause concern in sharing computer-aided design documents as designs were adjusted with the specific peculiarities of the 3D printer used in mind.

Basic usability tests showed a good controllability of the x, y, z positioning and motion. Grasping some objects was slow and cumbersome.

Testing with disabled users showed a definite interest in using such a robot. The concept of an interface running on any smart phone or tablet was much appreciated. However, the controls were too responsive for some students leading to a frustrating jerky motion. Many students were using wheelchairs and would have liked the robot capable of the same displacement velocity as their wheelchairs. In short, CHAP is desirable with a smaller footprint, more speed, easier gripper controls and probably a redesign of the gripper. The latter is actually easy to replace on the prototype, without even using a screwdriver.

No wheelchair user suggested mounting the manipulator on the wheelchair itself (see e.g. the design in [4]), possibly due to the added advantage of remote action offered by a separate mobile device.

5 Conclusion

Overall the design proved to be highly effective considering the overall cost of under £2000 (over £1400 worth of which were motors). The Cheap Arm Project resulted in a mobile robot arm system capable of tasks usually reserved for robot systems 10-times the cost. This system is an important step towards developing a high-powered robot arm system that can be easily accessed by education institutions, developers, researchers, and those in need.

User interface design faced the usual issues of compatibility and latency, with design and functionality having to be shaped carefully around these issues. Nonetheless, a widely supported control system was implemented, allowing for simplified control of a high number of robot axes. For a robot arm lacking automation, some complexity existed in the degree of manual controls needed, and difficulty in adding these in an intuitive way while ensuring minimal clutter and complexity. This task was achieved, with all test participants capable of manipulating items with no prior experience.

Nevertheless, controlling the gripper was generally slow and some thoughts need to be given to a more intuitive approach, possibly relying on some in-built intelligence simplifying the user's task while preserving a sense of control. The mechanical design of the gripper is also open for improvements.

It appears that offering an open-source design is not sufficient to make the device available to potential users. Most users do not have the technical skills and tools (even simple ones) to build their own robot. Someone needs to build it for them.

We feel that a second design cycle is needed before manufacturing can be considered. This cycle does not need to take long and any help from the open source community will be welcome.

References

1. Advait Jain, Charles C. Kemp EL-E: an assistive mobile manipulator that autonomously fetches objects from flat surfaces Autonomous Robots ,Volume 28 Issue 1, January (2010), Pages 45 - 64
2. Eaton E, Mucchiani C, Mohan M. Isele D. Luna JM and Clingerman C. Design of a Low-Cost Platform for Autonomous Mobile Service Robots. IJCAI (2016) Workshop on Autonomous Mobile Service Robots.
3. Nguyen, H., Anderson, C., Trevor, A., Jain, A., Xu, Z. and Kemp, C. C. (2008), El-e: An assistive robot that fetches objects from flat surfaces, Proceedings of HRI Workshop on Robotic Helpers: User Interaction Interfaces and Companions in Assistive and Therapy Robots, (2008), 79-86.
4. Hillman, M., Hagan, K., Hagan, S., Jepson, J. and Orpwood, R. (2002), 'The weston wheelchair mounted assistive robot-the design story', Robotica 20(02), 125-132.
5. Project source files: <https://mobile-chap.github.io/Web/>