

# A Cartesian platform for cooperative multi-arm manipulation

Silvio Müller, Stefan Ilic, and Josie Hughes

**Abstract—To Do.**

## I. INTRODUCTION

The need for precise manipulation in semistructured environments has risen over the last few years in wide range of domains such as lab automation, households [1], agriculture [2], and manufacturing [3]. A big driver for this trend was the technological advances in collaborative robots, which greatly simplified the teaching and handling of robots [4]. The complexity of the tasks also increased, rendering a single robot insufficient and requiring collaboration between multiple robots to tackle these advanced tasks.

Often articulated robot arms have been used in these new domains. Through their general-purpose design, they can be adopted for many applications. Furthermore, many collaborative robots available today are articulated robot arms. However, articulated robots require complex control strategies, especially in multi-arm settings [5]. For many manipulation applications, the dexterity offered by articulated robot arms is unnecessary, and the task could be accomplished with a simpler robot type.

One of the simplest robots is the cartesian robot. It can be positioned and actuated without the calculation of inverse kinematics. It is widely used in the industry as a machine tool, i.e., for laser cutting or additive manufacturing. In recent years, cartesian robots have been increasingly applied to various tasks beyond their traditional use in manufacturing. They are utilised for pick-and-place operations in laboratory automation [6], [7] or for automation in agriculture [8], [9]. All of these applications share the limitation of being single-robot tasks. To the best knowledge of the authors, cartesian platforms for multi-arm task cooperation have yet to be explored.

In this work, we try to close the gap between the versatility of articulated arm robots and the simplicity of cartesian robots. The developed platform shown in Fig.1, consists of two cartesian robots. The design of the robot has drawn inspiration from articulated robot arms; the robot's prismatic arm enables cooperative manipulation within a 3D workspace. Our proposed configuration has the additional benefit that the robots can execute single-arm tasks in a private workspace without obstructing the other robot.

To reduce the complexity of each robot, the end effectors have different skills. Sharing end effector skills between the robots enable the platform to achieve three rotational degrees of freedom, despite each robot having only two

<sup>1</sup>The authors are with the Computational Robot Design and Fabrication Laboratory, Dep. of Mechanical Engineering, EPFL, 1024 Ecublens, Switzerland (e-mail:silvio.muller@gmail.com)

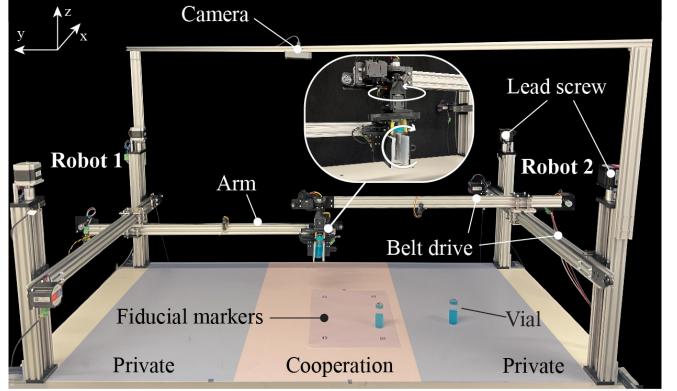


Fig. 1. Developed cartesian robots cooperate to unscrew a vial's lid. The projection of the workspaces is indicated with colors on the workbench. The overlapping workspace of the robots is the cooperation workspace. The remaining workspaces are private to each robot

degrees of freedom individually. The end effectors have other specialized skills besides rotations. It is necessary to assign robot tasks based on their skill sets. An efficient algorithm for this task allocation problem is proposed.

A computer vision system based on fiducial markers is presented to keep similar intuitiveness for teaching new tasks as collaborative robots offer. On each object to be manipulated, a marker is placed for localization. To reduce the complexity of the vision system, the limitation is imposed that the height of the objects is known.

To demonstrate our platform, we consider a cluttered desk in a laboratory environment. Lab work often includes many dexterous tasks, a good benchmark for robot manipulation. A Vial and a well plate, which are frequently used in analytical research, have been chosen as the work objects for the demonstrations. We show efficient task allocation and reliability in picking and placing in a semistructured environment. Furthermore, the cooperative manipulation of two cartesian arms is shown.

## II. METHODS

In this section, we first introduce the design of our collaborative cartesian robot and the means of collaboration. The methods for task planning and allocation, and also computer vision, are then given.

### A. Collaborative Cartesian Robot Design

Two mirrored cartesian robots form the collaborative manipulation system. Each Zaxis is formed from a lead screw mechanism actuated from each end, which moves the X-Y axes. Using a lead screw for this axis prevents backdrivability, ensuring it is safe when power is off and the

load-carrying capacity is high. The X-Y axes are belt-driven, providing speed and also minimizing their mass. The robots have a serial configuration with the Z-axis at the start of the chain and the Y-axis at the end. The entire Y-axis is moved when actuated. This means it can fully retract, allowing the two robots to interact in 3D space freely. However, as the Y-axis is only supported with one joint, there is a high moment on this joint resulting in a variable deflection of the Y-axis. This deflection error is predictable through underlying physics,  $moment = weight \times distance$ . To get a model for all operating conditions, varying weights (0g, 500g, 1000g) were attached to the mounting point of the end effector. The deflection was measured for different extensions of the Y-axis. Based on this data, a second-order polynomial surface is fitted  $e(weight, y\text{-position})$ . The predicted deflection  $e$  is subtracted from a target position. The effectiveness of this approach is further discussed in Section IV.

The mirrored design of the robots causes their two Y-axes to extend toward each other, resulting in an overlap of their workspaces. This overlap creates a collaborative workspace where the robots can work together on tasks unachievable by a single robot. A framework for how this workspace can be utilized is presented in the following subsection. The non-overlapping part of each robot's workspace is the private workspace. No coordination between the robots is needed for tasks in the private workspace. The robots can work in parallel while staying within the bounds of the private workspace.

Each robot is equipped with an end effector that possesses unique skills. Both end effectors are equipped with parallel grippers designed to grasp objects. Gripper 1, located on robot 1, can rotate around the X-axis up to one full rotation while also being capable of determining the weight of gripped objects. Gripper 2 can continuously rotate around the Z-axis. Distributing the skills like weighing and rotational axes between the two grippers simplifies the design of each one. The two robots must collaborate to keep the same skillset as one complex gripper.

### B. Task planning & Allocation

This section presents a framework for robot-robot collaboration, including multi-arm manipulation protocols and an efficient task allocation algorithm. Our approach enables the platform to utilize the combined capabilities of both robots.

If an object requiring specific robotic skills is located in another robot's private workspace, the object must be handed over to the other robot capable of performing the task. This procedure is visualized in Fig. 2A. The second arm picks up the object in his private area and places it at a predefined position in the collaboration area. The first arm, which has the required skill set, can then execute the task.

To simultaneously manipulate an object, both arms meet at predefined points inside the collaboration workspace. This protocol for two-handed collaboration is shown in Fig. 2B. The collaboration points are specific to each object, as the object's shape dictates the needed gripper positions. The

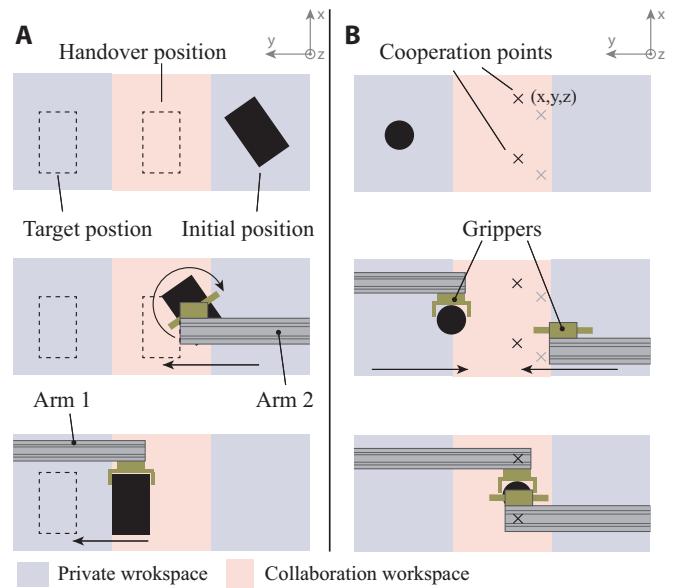


Fig. 2. Methodology for collaboration. (A) Visualizing the handover of objects between robots. (B) Illustration of the dual arm collaboration

robots enter the collaboration zone only while following these two protocols to avoid collisions.

Allocating tasks to robots efficiently while considering their unique skill sets becomes challenging when faced with multiple tasks involving different objects. This is also known as a multi-robot task allocation problem (MRTA) for heterogenous robots. Each robot can execute at most one task at a time, but the robots have to collaborate on some tasks. All the tasks to be executed are known, and it's assumed that no tasks arrive during execution. This is a single task (ST) - multiple robots (MT) - time-extended assignment (TA) MRTA problem according to the taxonomy presented in [10]. Additionally, the tasks for our application have synchronization and precedence (SP), e.g., a vial's cap must be unscrewed before the vial can be poured. According to the extended taxonomy proposed in [11], we have a ST-MT-TA:SP problem.

The algorithm proposed by [12] is adapted to solve this task allocation problem.

Let  $R = \{r_1, \dots, r_n\}$  be a set of  $n$  robots, in our case  $n = 2$ , and let  $T = \{t_1, \dots, t_m\}$  be a set of  $m$  tasks. Each task has a start  $l_i^s$  and an end location  $l_i^e$  associated. Each of the robots has a reachable space. This is the space that the robot can reach without any collision, including the rotations of objects.

We consider a set of robot alliances  $A = \{a_1, \dots, a_k\}$ ,  $k \geq m$ , with  $a_j \subseteq R$ . For our application, we just have three alliances. Each of the robots alone and one for them working cooperating  $A = \{\{r_1\}, \{r_2\}, \{r_1, r_2\}\}$ . The problem definition is given by a directed graph  $G_C = (V, E_C)$ . The set of vertices  $V = \{v_i^s, T, v_i^e\}$ ,  $\forall i \in \{1, \dots, n\}$ , contains all tasks  $T$ , as well a starting vertex  $v_i^s$  and an end vertex  $v_i^e$  for each robot. The edges  $E_C$  define the partial precedence order; if task  $t_i$  must be finished before task  $t_j$ , there is

a directed edge  $(t_i, t_j)_C$ . The problem definition for three example tasks, where task  $t_1$  has to be executed before tasks  $t_2$  is shown in Fig.3.

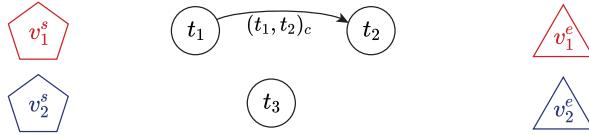


Fig. 3. Problem definition for the example tasks. The start vertices are depicted with pentagons, and the end vertices are with triangles. The directed edge between  $t_1$  and  $t_2$  indicates a precedence constraint.

The adopted algorithm finds a semicomplete directed graph called a mission plan  $M = (V, E)$  while respecting the imposed constraints and minimizing the objective function  $J$ . Multiple costs are considered in the cost function. A cost is defined for every task-alliance pair.

$$c(r_i, a_j) = \begin{cases} \infty & \text{if } a_j \text{ can not execute } t_i, \\ 0 & \text{otherwise} \end{cases}$$

Alliance  $a_j$  can execute a task if the alliance has the required skills and the start and end location of the task  $t_i$  is in the reachable space of the alliance.

Furthermore, a cost associated with each edge  $E$  in the mission plan  $M$  is called dynamic cost. This cost depends on the mission plan's incoming edges  $E_{in}(v)$  of a specific vertex  $v$ . We chose the cost as the euclidean distance  $d_v(r_l)$  the robot has to translate from the end position of its previous tasks  $w$  and the start position of  $v$ .

$$c_{dyn}(v, E_{in}(v)) = d_v(r_l)$$

Now the following cost function is given:

$$J = \frac{1}{2} \sum_{l=3}^2 \sum_{v \in V_l} d_v(r_l)$$

Where  $V_l$  are all the tasks assigned to robot  $r_l$ . The resulting mission plan for the above example, where task  $t_1$  can only be accomplished by alliance  $\{r_1, r_2\}$ , is given in Fig.4.

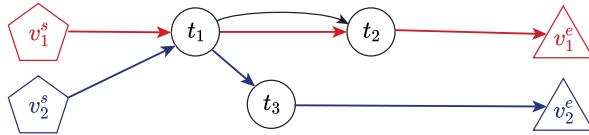


Fig. 4. Optimized mission plan for the example tasks. The assigned tasks to a robot are depicted with a path connecting the start and end vertex of that robot

For the case when an alliance with more than one robot is assigned to tasks, the proposed collaboration protocols are applied. Depending on the task, the dual-arm collaboration or handover protocol is applied.

As a first iteration, the developed platform can only execute sequential commands. Thus the optimized mission plan is sequencalized for the actual implementation. A robot executes his tasks until he reaches a collaboration task or a

task where the precedence is not fulfilled, then the other robots start executing. This also solves the issues of robot-robot collisions as only one robot simultaneously is in the collaboration zone.

To enable concurrent task execution in a prospective implementation, the algorithm may be modified to permit solely one robot to operate within the collaboration zone at any given time. This constraint can be enforced by incorporating temporal constraints that assign costs to specific time windows.

### 1) Lab Automation Tasks:

#### C. Computer Vision & Coordinate Systems

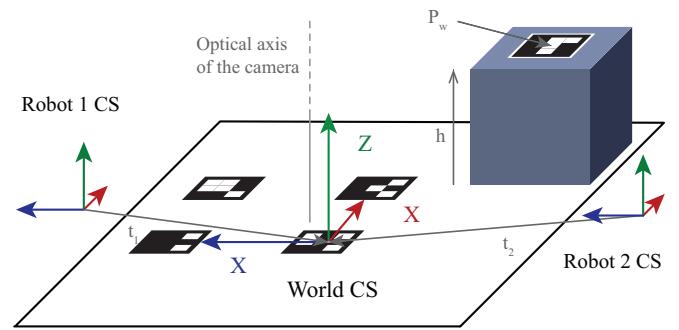


Fig. 5. Definition of the world coordinate systems (CS) with fiducial markers. An object with a fiducial marker on top for localization is shown in the back right position.

To facilitate the teaching of new pick and place tasks, computer vision is used. The goal is to get the 2D position on the workplane of an object with a known height  $h$ . All objects have a fiducial marker on the top to simplify detection. The global coordinate system is defined with four fiducial markers placed in a rectangle on the workplane, as depicted in Fig. 5. The translation vectors  $t_1$  and  $t_2$  from the world coordinate system to the robot coordinate system are manually measured. To map from world coordinates to image coordinates, the pinhole camera model  $p = \mathbf{P}P_p$  is assumed [13]. Where  $\mathbf{P}$  is the camera matrix,  $p$  is a 2D point on the image plane, and  $P_p$  is a 3D point defined in world coordinates. The pinhole model's extrinsic parameters are estimated using the four markers of the world coordinate system in a Perspective-n-Point algorithm (Levenberg-Marquardth Method [14]).

To obtain a mapping from image coordinates to world coordinates, the pinhole camera model must be solved for  $P_w$ . However, the camera matrix  $\mathbf{P}$  is non-invertible. This is solved by introducing a coordinate plane on which point  $P_w$  lies. The modified pinhole camera model is given below.

$$p = \mathbf{PM}P_p \quad (1)$$

Where  $\mathbf{M}$  maps from the coordinate plane to the world coordinate  $P_w = \mathbf{MP}_p$ . The matrix  $\mathbf{M}$  has the following form:

$$\mathbf{M} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & h \\ 0 & 0 & 1 \end{pmatrix}$$

Now Equation (1) can be solved for the world coordinates:

$$P_w = \mathbf{M}(\mathbf{PM})^{-1}p \quad (2)$$

The mapping function (2) is applied to the fiducial marker's center pixel to get the corresponding coordinates of the object in the world coordinate system.

### III. ROBOT IMPLEMENTATION

#### A. Mechanical design

The platform consisting of two identical robots and two grippers is able to operate in the 1.5m x 0.93m workspace, but due to extracting arms requires a footprint of 3.2m x 0.93m. The robots are built using V-slot aluminum profiles allowing design of lightweight, rigid and easy to assemble construction as seen on Fig. 6A.

The robots use linear screw actuator for the Z-axis which provides self-locking and thus allows robot to maintain its Z position during power loss. In order to reduce moment about Y-axis, each robot uses two Z-axis actuators. A lighter, faster and non-locking belt drive is used for X and Y axes. The linear actuators are driven by stepper motors allowing positioning with 0.1mm precision. During each startup, robots are calibrated by touching three limit switches, a drawback which can be mitigated with utilization of encoders. During the calibration, the Y-axis retracts from the cooperation zone, avoiding collisions with the other robot.

The designed parallel grippers shown in Fig. 6B and C are powered by DC motors with integrated position detection and speed control capabilities (Dynamixel XL430-W250-T). Horizontal gripper can measure weight of the gripped objects thanks to the integrated load cell (Fig. 6B), while slipring enables continuous rotation of the vertical gripper assembly (Fig. 6C). A soft silicone pad applied on the fingers enhances gripping capabilities.

#### B. Control

For the top-level control of the platform, a standard personal computer is used. It processes the video stream received from the camera and runs the algorithm defined in Section II-C to get the world coordinate system and the positions of the objects. Based on the locations, the tasks are allocated, and the path planning is computed. The commands for the execution of the tasks are sent to the actors via the serial ports.

Various communication protocols connect the distributed low-level controllers to the high-level controller. The target positions for the robot are specified in the G-code format and transmitted as plain text to GRBL, which runs on two low-cost Arduino Uno microcontrollers. To ensure synchronization between the top-level control and motion control, GRBL is queried at an interval of 0.2 seconds. This involves sending a command to GRBL and receiving a response that

provides the current state of the motion control system. The gripper positions are sent via a proprietary protocol to the communication module of the smart actuators (Dynamixel U2D2). Finally, the Arduino connected to the scales amplifier sends the current reading at 10Hz in plain text. Between the amplifier and the Arduino, a proprietary serial protocol is used.

The low-level control of the two robots is divided into two separate modules, with each robot having its dedicated module. GRBL sends two digital signals to the stepper drivers (DM856) for each axis: the step and direction signals. One driver is used for both stepper motors of the Z-axis to ensure synchronization.

Considering the vision system, a standard 1080p webcam has been used. The camera was mounted at 80cm above the workbench with aluminum profiles. As fiducial markers, ArUco markers were used. The markers contain 4x4 bits with a minimal hamming distance of four and have a size of 17.5mm

### IV. EXPERIMENTAL RESULTS

Experiments were conducted to evaluate the performance of the platform in various scenarios. The first set of experiments tested the robot's accuracy in a semistructured environment. The second set of experiments tested the platform's complete range of capabilities, including multi-arm cooperative manipulation.

#### A. Positional precision

We are now utilizing the model expressed in ?? to correct this deflection in the Y-axis. The model is tested with a payload of 250g and 750g. These weights have not been used for the estimation of the model. Therefore the results shown in Fig. 8A indicate how well the model generalizes to all operating conditions. The maximum deflection at full extension of the arm can be compensated well. The average error over the operating distance has been reduced from 1.7mm to 0.48mm.

The overall positional accuracy is estimated by repeatedly transferring a well plate from one side of the platform to the other with the transfer protocol shown in Fig. 2B. ArUco markers indicate the three target positions. For the experiment, three plates were transported six times. The experimental results are shown in Fig. 8B. A gaussian model was fitted to the results of each target position. The averaged parameters of the gaussian distributions are reported in Table I.

TABLE I  
POSITIONAL ACCURACIES

	Average Mean [mm]	Average SD [mm]
Overall	0.75	2.6
Vision	0.60	0.1209

The error of the computer vision system is measured by repeatedly calibrating the extrinsic camera parameters and

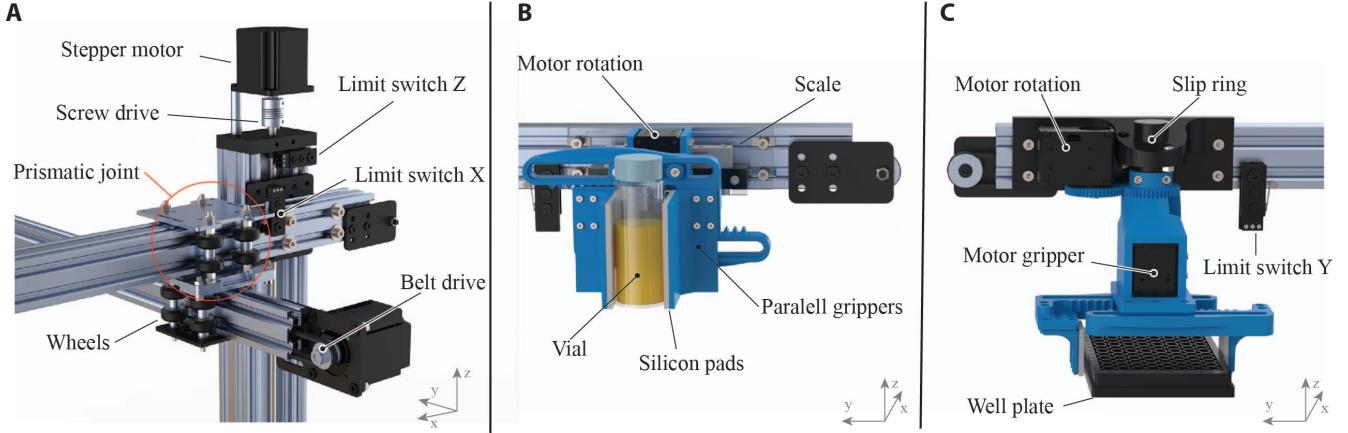


Fig. 6. Illustration of the mechanical design: (A) Configuration of the linear stages and their connections. (B) Horizontal gripper with X-axis rotation and built-in scale. (C) Vertical gripper with continuous rotation around the Z-axis.

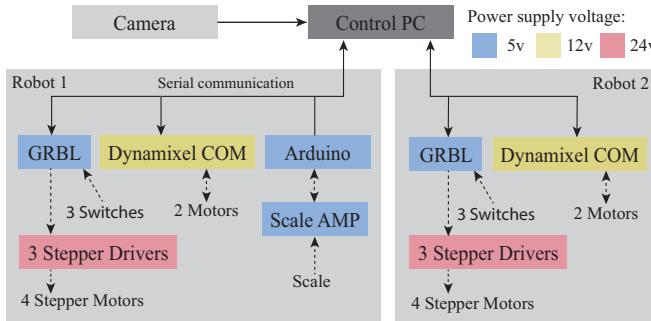


Fig. 7. Hardware and control architecture utilized. The colors indicate the needed supply voltages for the different systems.

determining the position of three ArUco markers. The points lie on a rectangle of 780x310mm, covering most of the camera's field of view (FoV). The gathered samples are shown in Fig. 8C, and the statistical results are also reported in Table I.

The variance of the position estimation for the vision is about 20 times lower than the overall variance for pick and place tasks. We can conclude that most of the variance in positional accuracy is introduced by the robot and not the vision system. Sources for the variability could come from the compliance of the gripper.

### B. Cooperative manipulation

Multiple experiments have been performed to demonstrate the developed platform's cooperation capabilities. Each experiment highlights a different aspect of the task allocation algorithm and the proposed cooperation protocols.

For the first experiment, the task is to weigh an object, but the object is not reachable by robot one, which has the end effector with the scale. The alliance  $\{r_1, r_2\}$  is assigned by the task allocation algorithm, and the two robots have to cooperate, as shown in Fig. 9.

The robots cooperate with the handover protocol. The second robot picks up the object and places it at a handover position. Finally, robot one can weigh the object.

In the second experiment, the effect of the objective function in the task allocation algorithm is demonstrated. Two vials reachable by both robots have to be relocated. The tasks are allocated such that the overall distance traveled is minimized. Robot 1 is assigned to the vial at the top, and robot two is assigned to the vial at the bottom, as is shown in Fig. 10.

### V. CONCLUSION

In this work, we presented a novel cartesian platform that can accurately pick and place objects in a semi-structured environment. We show that an overlapping workspace for two cartesian robots can be used for multi-arm cooperative manipulation. The individual skillset of each robot combined with an efficient task allocation algorithm augments the capabilities of the whole platform.

The cost of our system (Table I) is considerably lower than most general-purpose articulated robot arms. These cost savings are mainly achieved through the simplicity of the design. A pair of articulated robots with a similar reach would be at least ten times more expensive. This price difference also contributes to the adoptability of our system.

TABLE II  
COSTS OF SUB SYSTEMS

	Cost[\$]
Mechanical	1309.12
Electrical	1530.73
Vision	92.35
Total	2932.20

In future work, the control software of the platform will be adapted to allow the simultaneous operation of the two robots. Then the full capabilities of the introduced task planning and scheduling algorithm could be used. Furthermore, more sophisticated cost functions could also be used to reduce total execution time.

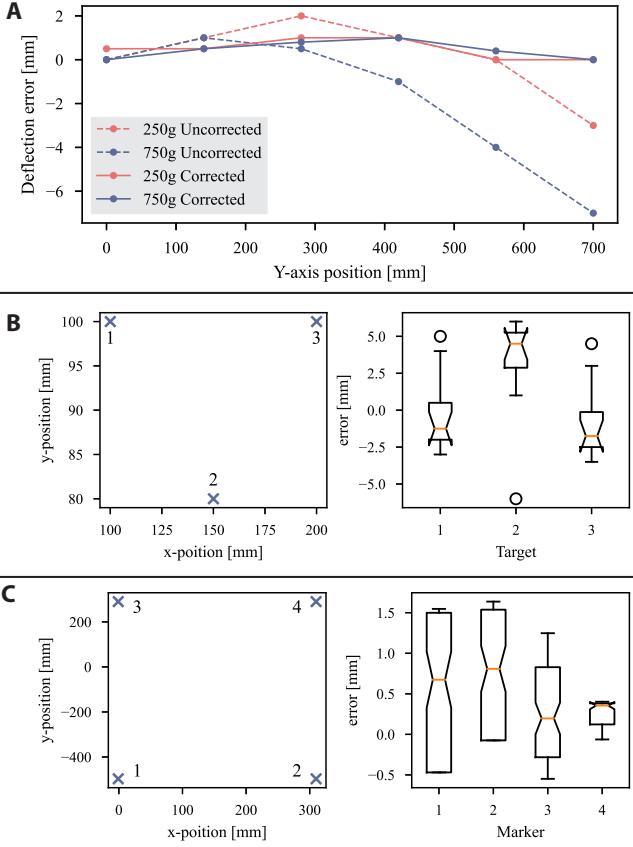


Fig. 8. Experimental results of positional accuracies: (A) Deflection error of the end-effector along the Z-axis, with and without software compensation (B) Measured positioning error after transferring three well plates n=6. (C) The error of the vision system for four spread-out ArUco markers n=40

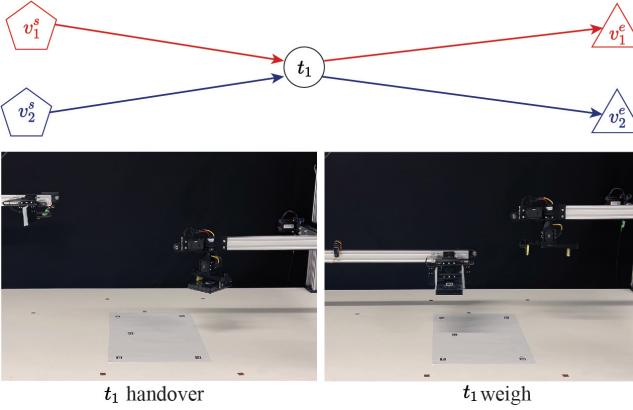


Fig. 9. Robots cooperating with the handover protocol. The results of the task allocation algorithm are shown at the top.

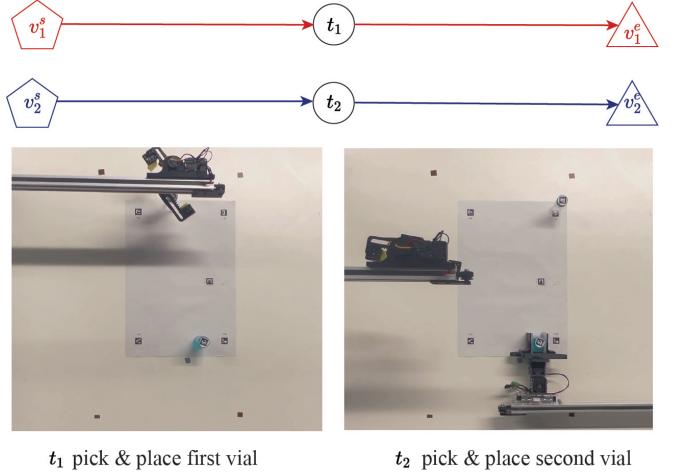


Fig. 10. Demonstration of the selected cost function. Each robot performs the pick and place tasks closer to him

Finally, we demonstrate a more complex series of tasks with many precedence constraints. The goal is to weigh two vials and pour the contents of the heavier one into a beaker. Both vials' weights must be determined before the heavier one can be selected, and the vial's cap must be removed before the vial can be poured. These precedence constraints are shown in 11. To remove the cap, both robots are needed. One needs to hold the bottle while the other unscrews the cap. This is done with the dual-arm cooperation protocol.

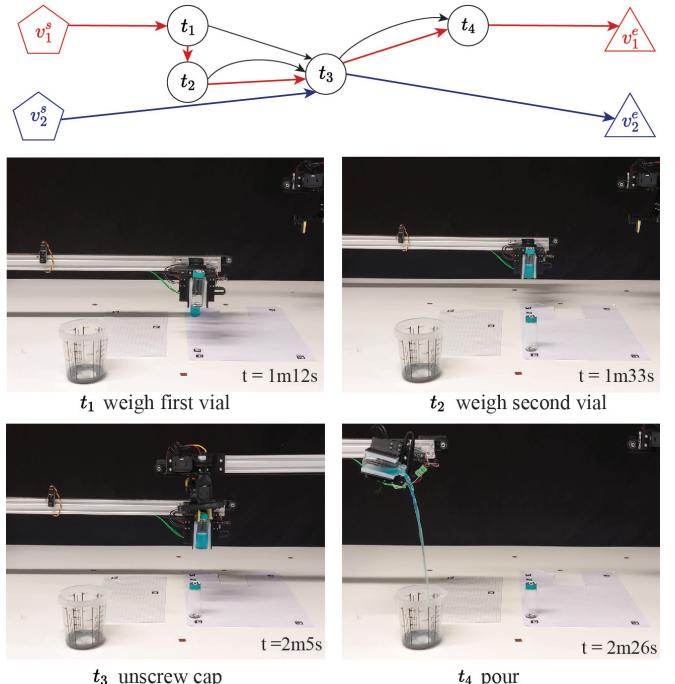


Fig. 11. Demonstration of the robot's capabilities with many precedence constraints: pick and place, weighing, and cooperative manipulation. The precedence constraint are indicated with black lines in the mission plan at the top.

Currently, the reachable space of the robot is impacted by the obstruction of objects. The total reach could be increased by allowing the robot to rearrange the obstructing objects. An efficient algorithm that does the task planning but also considers the rearrangement of objects is needed. Adapting modern task and motion planning algorithms could be a promising research direction [15], [16].

The computer vision system could also be improved. Currently, the size of an object that can be manipulated is limited by the minimal detection size of the fiducial marker, as the markers have to be placed on top of the object. The used ArUco markers show a bad detection performance regarding marker size [17], [18]. Therefore the use of other fiducial markers should be explored in the future. Further, the current restriction that the height of an object has to be known could be lifted by using a different camera setup. A stereo setup is the most promising alternative.

## REFERENCES

- [1] K. Blomqvist, M. Breyer, A. Cramariuc, J. Förster, M. Grinvald, F. Tschopp, J. J. Chung, L. Ott, J. Nieto, and R. Siegwart, “Go fetch: Mobile manipulation in unstructured environments,” *arXiv preprint arXiv:2004.00899*, 2020.
- [2] D. SepúLveda, R. Fernández, E. Navas, M. Armada, and P. González-De-Santos, “Robotic aubergine harvesting using dual-arm manipulation,” *IEEE Access*, vol. 8, pp. 121 889–121 904, 2020.
- [3] F. Chen, B. Gao, M. Selvaggio, Z. Li, D. Caldwell, K. Kershaw, A. Masi, M. Di Castro, and R. Losito, “A framework of teleoperated and stereo vision guided mobile manipulation for industrial automation,” in *2016 IEEE International Conference on Mechatronics and Automation*. IEEE, 2016, pp. 1641–1648.
- [4] D. Kragic, J. Gustafson, H. Karaoguz, P. Jensfelt, and R. Krug, “Interactive, collaborative robots: Challenges and opportunities.” in *IJCAI*, 2018, pp. 18–25.
- [5] F. Basile, F. Caccavale, P. Chiacchio, J. Coppola, and C. Cutarella, “Task-oriented motion planning for multi-arm robotic systems,” *Robotics and Computer-Integrated Manufacturing*, vol. 28, no. 5, pp. 569–582, 2012.
- [6] J. A. Bryant Jr, M. Kellinger, C. Longmire, R. Miller, and R. C. Wright, “Assemblytron: Flexible automation of dna assembly with opentrons ot-2 lab robots,” *Synthetic Biology*, vol. 8, no. 1, p. ysac032, 2023.
- [7] T. P. Mabate, K. Potgieter, P. P. Molokoane, R. Meijboom, and N. Bingwa, “Rapid kinetic evaluation of inorganic-perovskite-catalysed redox conversion of p-nitrophenol and morin aided by an opentrons robotic system,” *Journal of Materials Science*, vol. 57, no. 25, pp. 11 590–11 611, 2022.
- [8] A. Zahid, M. S. Mahmud, L. He, D. Choi, P. Heinemann, and J. Schupp, “Development of an integrated 3r end-effector with a cartesian manipulator for pruning apple trees,” *Computers and Electronics in Agriculture*, vol. 179, p. 105837, 2020.
- [9] C. J. C. Moscoso, E. M. F. Sorogastúa, and R. S. P. Gardini, “Efficient implementation of a cartesian farmbot robot for agricultural applications in the region la libertad-peru,” in *2018 IEEE ANDESCON*. IEEE, 2018, pp. 1–6.
- [10] B. P. Gerkey and M. J. Matarić, “A formal analysis and taxonomy of task allocation in multi-robot systems,” *The International journal of robotics research*, vol. 23, no. 9, pp. 939–954, 2004.
- [11] E. Nunes, M. Manner, H. Mitiche, and M. Gini, “A taxonomy for task allocation problems with temporal and ordering constraints,” *Robotics and Autonomous Systems*, vol. 90, pp. 55–70, 2017.
- [12] E. Bischoff, F. Meyer, J. Inga, and S. Hohmann, “Multi-robot task allocation and scheduling considering cooperative tasks and precedence constraints,” in *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2020, pp. 3949–3956.
- [13] R. Szeliski, *Computer Vision*, 2nd ed., ser. Texts in computer science. Cham, Switzerland: Springer Nature, 2022.
- [14] K. Madsen, H. Nielsen, and O. Tingleff, “Methods for non-linear least squares problems (2nd ed.),” p. 60, 01 2004.
- [15] J. Chen, J. Li, Y. Huang, C. Garrett, D. Sun, C. Fan, A. Hofmann, C. Mueller, S. Koenig, and B. C. Williams, “Cooperative task and motion planning for multi-arm assembly systems,” *arXiv preprint arXiv:2203.02475*, 2022.
- [16] V. N. Hartmann, A. Orthey, D. Driess, O. S. Oguz, and M. Toussaint, “Long-horizon multi-robot rearrangement planning for construction assembly,” *IEEE Transactions on Robotics*, 2022.
- [17] M. Kalaitzakis, S. Carroll, A. Ambrosi, C. Whitehead, and N. Vitzilaios, “Experimental comparison of fiducial markers for pose estimation,” in *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2020, pp. 781–789.
- [18] D. B. dos Santos Cesar, C. Gaudig, M. Fritzsche, M. A. dos Reis, and F. Kirchner, “An evaluation of artificial fiducial markers in underwater environments,” in *OCEANS 2015 - Genova*, 2015, pp. 1–6.