Title:	Self-reconfiguration of a robotic workcell for the recycling of electronic waste
Acronym:	**ReconCycle**
Type of Action:	Research and Innovation Action
Grant Agreement No.:	871352
Starting Date:	01-01-2020
Ending Date:	31-03-2024

Deliverable Number:	D5.4
Deliverable Title:	Use case 1: solution with adaptation and self-reconfiguration
Type:	Demonstrator
Dissemination Level:	Public
Authors:	Mihael Simonič, Boris Kuster, Matevž Majcen Horvat, Bojan Nemec, Aleš Ude, Sebastian Ruiz, Kübra Karacan, Hamid Sadeghian, Riccardo Persichini, Umberto Fontana, Manuel Catalano, Vinicio Tincani, and Jürgen Schulz
Contributing Partners:	JSI, UGOE, TUM, QB, IIT, ECYC

Estimated Date of Delivery to the EC:	31-03-2022
Actual Date of Delivery to the EC:	03-11-2022

Contents

Executive summary

Deliverable **D5.4** is a demonstration deliverable that contains videos showing the implementation of the disassembly process for different heat cost allocators (HCAs).

The implemented disassembly workflow has been evaluated with two different HCAs, the Kalo 1.5 and Qundis HCA. We previously presented the archetypical solution for Kalo 1.5 HCA in deliverable **D5.2** based on a workflow proposed in deliverable **D5.1**. In this deliverable, we extend this workflow to support additional devices by employing adaptation and self-reconfiguration of the workcell. The approach is demonstrated in two separate videos that can be viewed via the links provided below.

- Video 1: The ReconCycle modular robotic workcell performing the disassembly of the Kalo 1.5 and Qundis heat cost allocators using adaptation and self-reconfiguration:

  Video link: http://reconcycle.eu/videos/d5.4/hca_disassembly

- Video 2: Adaptation of the levering operation for extraction of PCBs from different heat cost allocators in the ReconCycle modular robotic workcell:

  Video link: http://reconcycle.eu/videos/d5.4/levering

Additionally, we show some aspects of adaptation and reconfiguration that exhibit the newly developed methods and technologies that can be used to expand the functionalities of an adaptive and reconfigurable recycling workcell:

- Video 3: Self-reconfiguration of a variable stiffness gripper:

  Video link: http://reconcycle.eu/videos/d5.4/gripper

- Video 4: Precise grasping of small objects with qb SoftHand2:

  Video link: http://reconcycle.eu/videos/d5.4/softhand

- Video 5: Teaching contact skills and adaptation to unknown object geometry with force controller:

  Video link: http://reconcycle.eu/videos/d5.4/contact_skills

- Video 6: Learning of disassembly operations in constrained environments:

  Video link: http://reconcycle.eu/videos/d5.4/constrained_disassembly

These implementations demonstrate how the current ReconCycle system can be applied to disassemble different heat cost allocators for the purpose of electronic waste recycling. The proposed solutions for adaptation and self-reconfiguration in a robotic workcell enable handling of different devices of the same general type, e.g. heat cost allocators, for the purpose of automated recycling of electronic waste. The proposed technologies provide a solid basis for further developments of the ReconCycle system to support disassembly protocols for a further spectrum of electronic devices.

1 Introduction

The goal of the ReconCycle project is to introduce the concept of adaptation and reconfiguration to the field of electronic waste recycling, which is still largely dominated by manual labor due to the lack of flexibility of existing solutions. In this deliverable, we first present the implementation of the disassembly protocol for two different heat cost allocators to demonstrate that our framework can be applied for disassembly tasks in the domain of electronic waste recycling, where adaptative, reconfigurable software and hardware are necessary to automate the recycling procedures.

The complete disassembly solution was implemented in the ReconCycle reconfigurable robotic workcell located at the JSI facilities. The complete disassembly process for two different HCAs is shown in Video 1, which is described in more detail in Section 2. To handle different object of the same type, both hardware and software need to support a high level of flexibility. For this purpose, UGOE and JSI developed a vision-based system that can suggest the next action to proceed with the disassembly. In addition, to grasp different objects at different locations, we need to be able to not only identify them, but also to estimate their poses with sufficient accuracy.

Adaptation of the levering operation for the extraction of PCBs from different heat cost allocators is shown in detail in Video 2, which is described in Section 3.

The qb Variable Stiffness gripper (VSG), developed by the project partners QB and IIT, is used to perform pin pushing, levering and grasping various parts in the shown disassembly procedures. In Video 3 and Section 4 we show how the VSG can be reconfigured to support handling even wider range of objects. Video 4 shows how the upgraded qb SoftHand2 Research can precisely grasp small objects such as batteries.

Videos 5 and 6 show some further aspects of adaptation, which were not yet integrated in the ReconCycle disassembly pipeline.

2 Adaptable disassembly process for heat cost allocators

We have implemented two types of adaptation in the recycling cell:

- vision-based action prediction for situations where several actions are possible to continue the disassembly process,
- adaptation of disassembly skills to different objects within the same device type.

The first demonstrator (Video 1) shows how the semantic scene analysis can be used to determine relations between the parts present in the scene in order to determine which of the possible follow-up actions should be performed. The second demonstrator (Video 2) shows how the parameters of a predefined levering actions can be adapted to learn a levering behavior for a specific heat cost allocator.

2.1 Workflow for the disassembly of two different HCAs

We used two different heat cost allocators to demonstrate the adaptation processes in the recycling cell (see Fig. 1). The workflows for their disassembly are shown in Fig. 2 and 3, respectively. The main difference between the two workflows is in the additional step necessary

(a) Kalo 1.5 (b) Qundis

Figure 1: Two different heat cost allocators: Kalo 1.5 and Qundis. In some Qundis HCAs, a plastic pin prevents performing the levering operation directly, as it blocks access to the gap. An additional pin pushing operation is necessary.

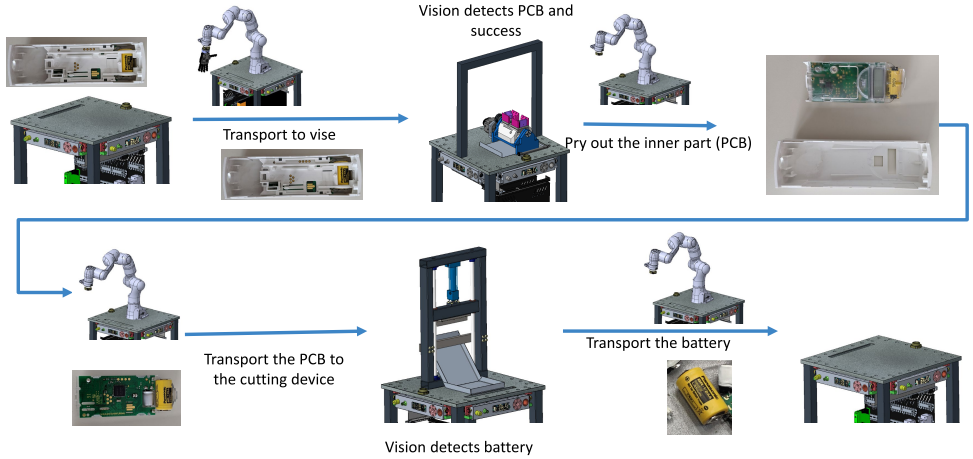


Figure 2: Disassembly workflow for Kalo 1.5 heat cost allocator

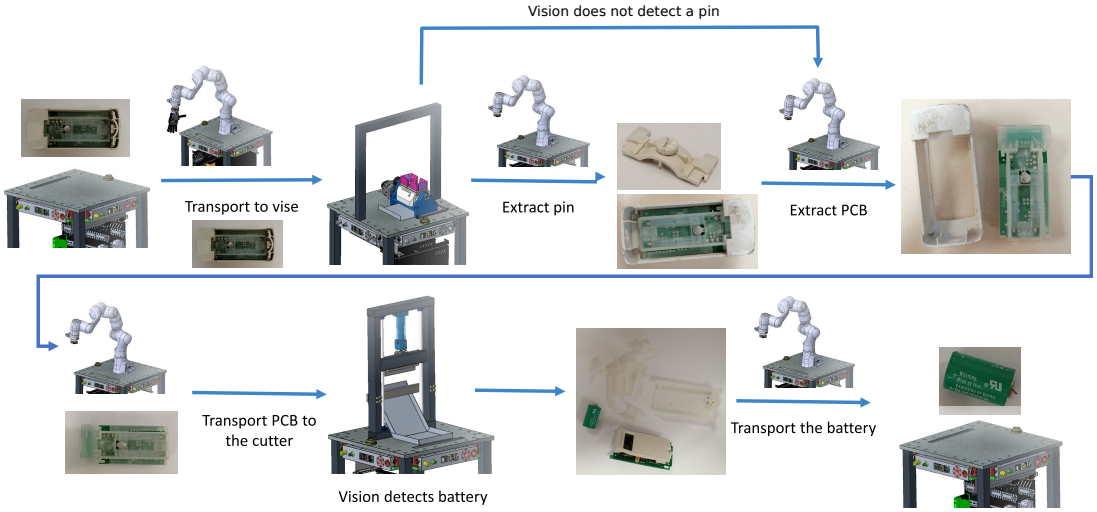


Figure 3: Disassembly workflow for Qundis heat cost allocator

to extract the PCB of Qundis HCA. This is because the Qundis HCA contains a pin that must be removed before the PCB can be levered out. However, the pin is not always present, in which case the workflow for Qundis HCA is the same as for Kalo 1.5 HCA. We use vision to determine which action should be executed after the "transport to the vise" action.

Another important difference between the two HCAs is the size of the gap between the edge of the HCA housing and the PCB. Note that in the case of Kalo 1.5, the gap is very wide while in the case of Qundis, the gap is narrow. Thus the levering procedure suitable for Kalo 1.5 is not applicable to Qundis and vice versa. We implemented a skill adaptation procedure to automatically learn the levering actions for both cases.

2.2 Modular workcell for adaptive recycling

The robotic workcell required for the required disassembly processes was implemented based on a modular workcell that was first utilized in deliverable **D5.2**. For the programming of disassembly procedures, we use the FlexBE behavior engine, as presented in **D5.2** and **D1.1**. The poses and trajectories needed to parametrize the FlexBE state machines are acquired by kinesthetic teaching and/or using the vision system.

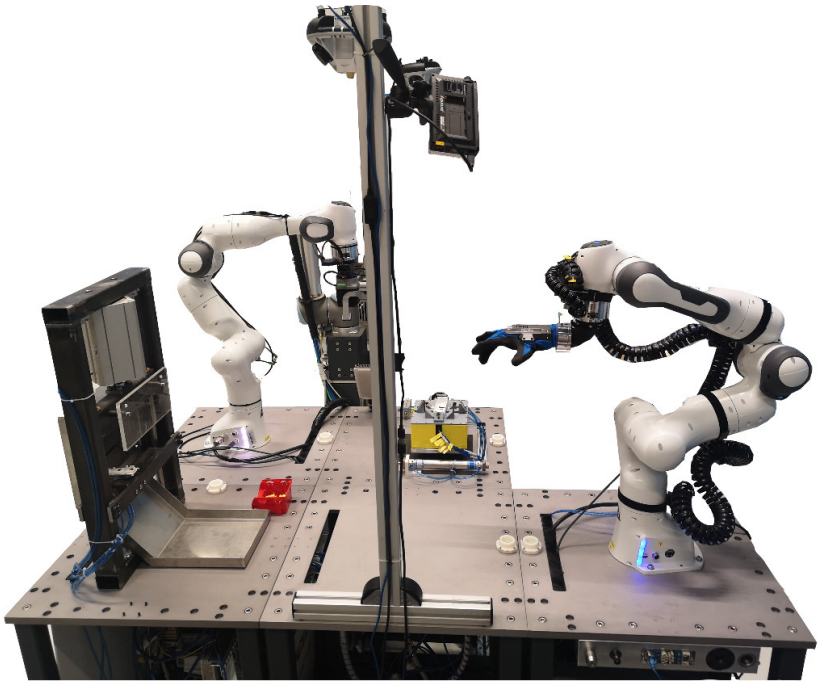


Figure 4: Workcell performing the disassembly

In the current workcell, five ReconCycle modules – all developed based on the ReconCycle archetypical module – are used to implement the disassembly process for different heat cost allocators:

- *Material Input Module*, where HCAs to be disassembled are stocked,
- *Robot Module* with the Franka Emika Robot and the qb SoftHand Research gripper,
- *Robot Module* with the Franka Emika Robot and the qb Variable Stiffness Gripper (VSG),

- *Vise Module*, with the pneumatic vise, and
- *Cutter Module*, with the pneumatic cutter.

See deliverable **D1.2** and reference [7] for more details about these modules. The modules were integrated in the ReconCycle ROS network as described in deliverable **D1.1** and reference [9].

2.3 Implementation of action prediction framework

Contrary to the archetypical solution presented in **D5.2**, where the disassembly procedure was fixed and the parameters of the disassembly workflow were recorded by kinesthetic teaching, we employed a vision-based system for action prediction in this demonstrator. It is based on the ReconCycle vision system for scene analysis described in deliverable **D2.1**.

The generic state machine, implemented as a FlexBe behaviour, consists of several blocks as shown in Fig. 5. The block *Get next action* obtains the information about the previous action execution and suggests the next action(s) to proceed with the disassembly. The blocks *Cut*, *Lever*, *Move*, *Push*. etc. trigger the respective operations. The block *Concurrency* indicates the possibility of concurrent execution of several actions following the FlexBe mechanism for the concurrent execution of several actions.

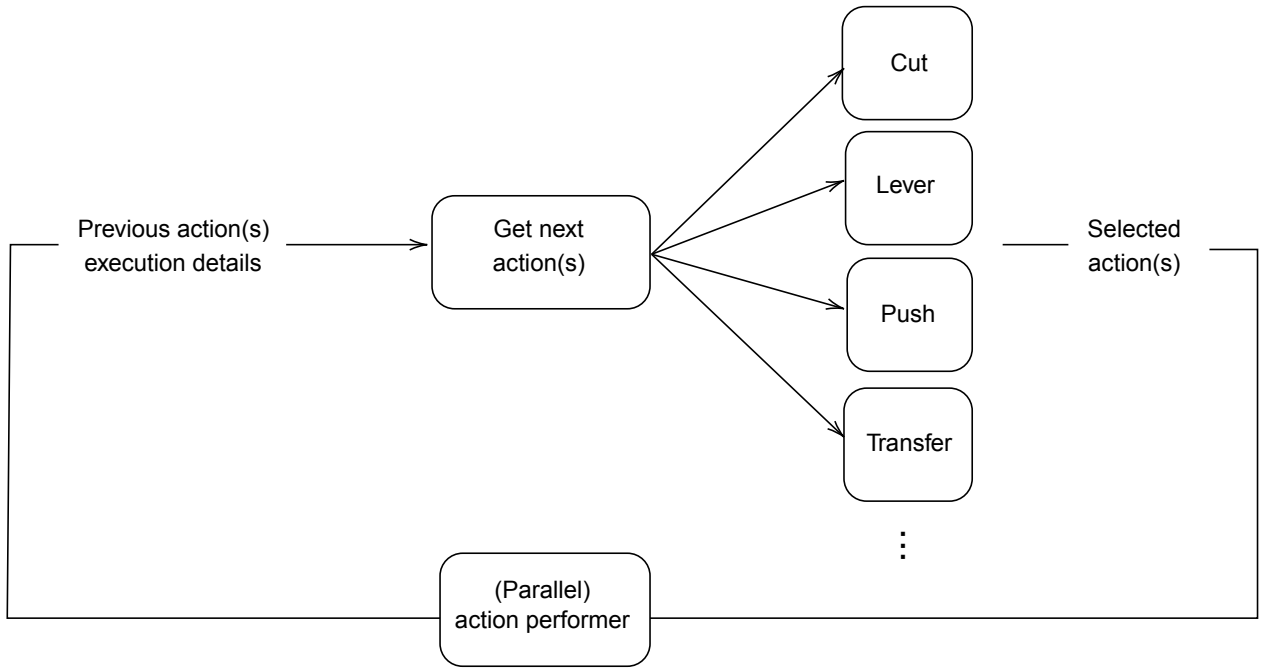


Figure 5: A simplified scheme of the action prediction framework

Each block is implemented as a FlexBE state, which passes the action parameters via the corresponding ROS interfaces to perform the necessary robot movement, query the vision system to extract additional information, control the peripheral devices, etc. The *Parallel action performer* block allows for concurrent execution of actions (e.g., when one robot is moving the plastic housing into the dedicated bin for plastic waste, while the other is transferring the PCB into the cutter module for further manipulation).

Currently, the action prediction framework follows a predefined disassembly workflow. Based on the success of the previously executed action, it continues with the next possible action from the action queue. Nevertheless, this kind of architecture allows us to develop models for fully adaptive disassembly processes, where the order of operations may depend on the conditions of the parts. The action prediction framework will be described in more detail in the upcoming deliverable **D3.2**.

2.3.1 Demonstration of vision-based action prediction

The application of the action prediction system is demonstrated in **Video 1**.

The vision system detects an HCA in the scene and determines the model as Kalo 1.5 (see Fig. 6). The first robot starts by picking up the HCA using qb SoftHand Research gripper and places it in a pneumatic vise, which closes in order to firmly clamp the housing. The vision system detects that the gap for levering exists and suggests to proceed with the levering action.

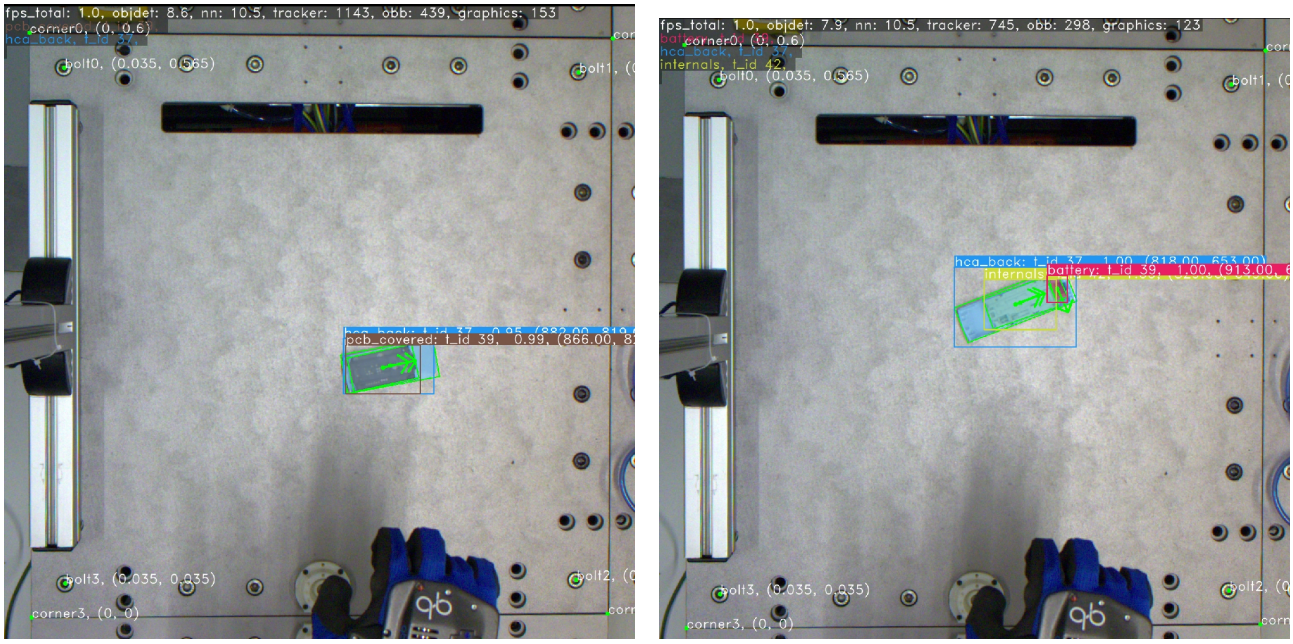


Figure 6: Segmented image of Qundis (left) and Kalo 1.5 HCA (right). The segmented images are used to determine the type of HCA and the picking location.

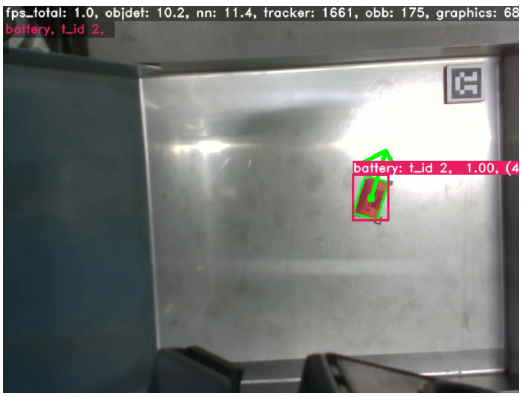


Figure 7: Battery location on the tray after cutting operation as detected by the vision system

The second robot creates the plan for the levering action and breaks the PCB out of the housing, using the lever tool incorporated in the VSG. The robot grasps the PCB using the VSG and places it into the cutter. When the battery is cut from the PCB, the robot uses the same tool to pick up the battery and places it into a container designated for the removed batteries. The battery location is determined with the vision system as shown in Fig. 7. Concurrently, the jaws of the vise are opened, allowing the first robot to pick up the empty housing using the qb SoftHand and transfer it to the designated container.

In the next task, a different model of HCA is detected, namely the Qundis HCA. The disassembly steps are similar as in the first cycle. Parameters for grasping, levering, and inserting the PCB into the cutter are obtained from the vision system. The difference occurs before the levering action as no suitable gap to perform the levering out of the plastic housing is detected. Thus the action predictor suggests removing the pin before levering. To ensure that the pin does not remain stuck in the levering gap, the vise is rotated upside down so that the pin is removed and the vise is rotated back up. The remaining steps follow in the same order as for the Kalo 1.5 device.

3 Adaptation of the levering operation for extraction of PCBs from different heat cost allocators

One of the main steps in the disassembly pipeline for the considered devices is the levering operation. When extracting the PCB, levering allows the robot to apply greater forces using a fulcrum (levering support) at the edge of the HCA. If the HCA is one of the known HCA types, this task can be accomplished using pre-recorded robot trajectories. A more general approach to levering is needed for new HCA types or if the HCA to be disassembled is damaged so that the preprogrammed trajectory does not result in a successful levering operation. To address this, we implemented an adaptive levering procedure based on periodic dynamic movement primitives (DMPs) and force-torque feedback control to determine contact points.

Fig. 8 shows a typical levering setup. As the lever, we use the fixed finger of the VSG. The part to be levered lies within the object. To increase mechanical advantage, the lever is positioned against the fulcrum.

The levering action consists of four steps. It assumes that the HCA has a flat bottom and lies on a horizontal surface. In the first step, the vision system detects the gap between the edge of the HCA and the PCB where the lever should be placed. The vision system also indicates the direction in which the lever should move to establish a contact with the PCB (x direction

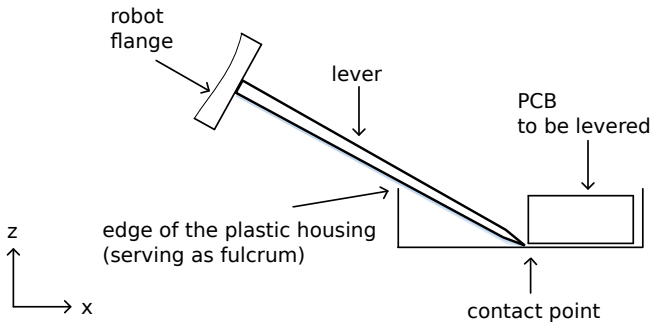


Figure 8: A simplified scheme of elements in the levering process.

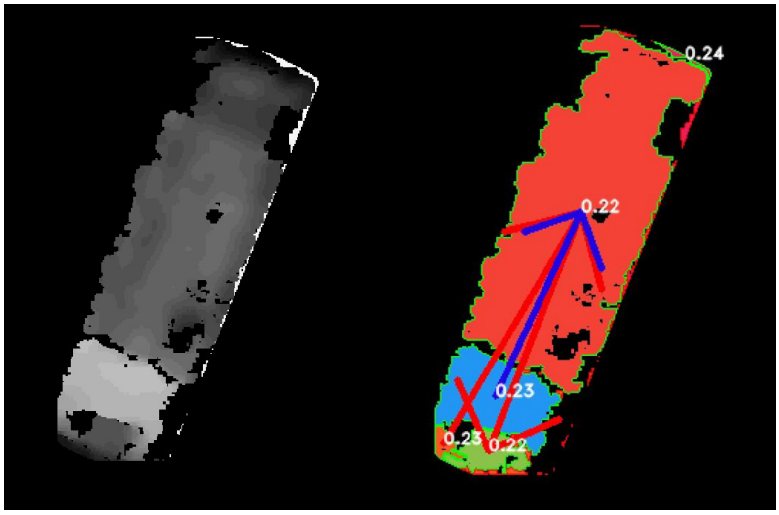


Figure 9: Gap detection results for Kalo 1.5 HCA (right) based on depth image (left). The blue area shows the gap where levering is possible. Blue arrow shows a possible levering direction, while red arrows show invalid directions.

in Fig. 8). The desired lever position is in the middle of the gap, which is detected by vision as shown in Fig. 9. Based on the length of the gap, we determine also the initial inclination of the lever. Next, the robot starts moving in the negative z direction to establish contact with the bottom of the HCA housing. The motion is stopped once the force F_z^{TCP} acting on the tool center point in vertical direction exceeds a predefined threshold $F_{\max}$. The force-torque estimation is performed internally by the Franka Emika robot control system using internal joint torque sensors [4].

In the second step, the robot moves along the positive x axis until the horizontal force F_x^{TCP} exceeds the force threshold $F_{\max}$. At this point the lever is in contact with the PCB to be levered out of the HCA.

The aim of the third step is to determine the fulcrum. A rotational motion around the y axis positioned at the tip of the lever is performed. The motion is stopped once the torque M_y exceeds the threshold $M_{\max}$. The contact with the fulcrum has been established at this point.

Finally, we can proceed with levering out the PCB. We have observed that humans often use periodic movements when levering, especially when they do not know the force required to dislodge an object with the lever. In doing so, they slightly increase the force applied to the lever in each repetition. To mimic this behavior, we generate a single degree-of-freedom (DOF) sinusoidal movement with an amplitude A and duration t_c, which were respectively set to 10° and $5s$ in our experiments. The movement was encoded as a periodic DMP [1]. Initially, the lever is in a position where it touches both the PBC and the fulcrum. While the periodic DMP is executed, the arising forces and torques are estimated. The levering is considered successful when shortly after a peak in the estimated force, a sudden drop is detected, meaning that the locking mechanism has been broken. If one cycle of the periodic DMP is completed without encountering the success signal, the periodic DMP's amplitude is increased. This effectively increases the force applied to the levered part during each iteration. This process is repeated until the success signal is detected.

The key points of the movements to establish contact with the bottom of the HCA housing, PCB mounted in the housing, and the fulcrum at the edge of the housing as well as the

parameters of a periodic DMP that lead to the successful levering action are recorded. Thus next time the adaptation procedure is not necessary if the same heat cost allocator needs to be disassembled. A paper presenting an early version of this approach was published in [5].

This process is presented in Video 2 for Kalo 1.5 and Qundis HCA. We first show the adaptation of the levering operation, followed by the faster execution of the newly acquired levering action with the learned parameters.

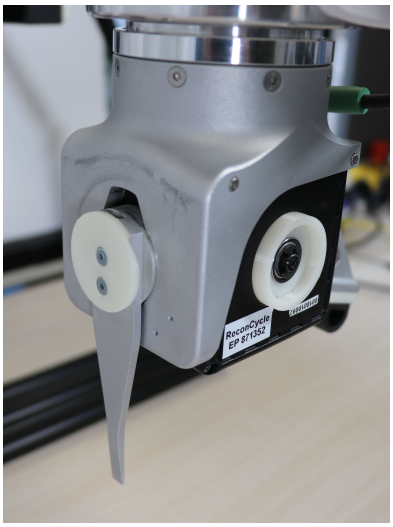
4 Variable stiffness gripper reconfiguration

The gripper finger exchange system complements the existing reconfiguration possibilities of the ReconCycle system (e.g. Plug and Produce connectors built into archtypical modules as described in deliverable **D1.2**).

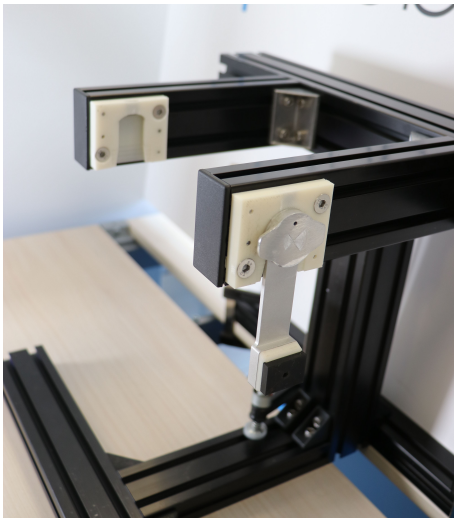
The design of the finger exchange system is based on the division of the fixed finger into two parts: the housing connected to the base of the gripper and the removable finger tool. The interface between these two parts is designed to allow both automated and manual insertion of the finger tool into the housing. The mechanism is shown in detail in deliverable **D4.2**.

Complemented with a finger exchange station shown in Fig. 10, it becomes possible to perform self-reconfiguration of the variable stiffness gripper. Video 3 shows a sequence performing automatic fingers exchange in the premises of QB.

This gives us the possibility to develop well suited tools for different disassembly operations where variable stiffness actuator is beneficial.



(a) VSG with a custom finger



(b) Finger exchange station with an alternative finger

Figure 10: Finger exchange system

5 Precise grasping of small objects with soft hand

Robotic workcells for the recycling of electronic devices must deal with devices and parts that differ significantly in the dimensions and shape. Large electronic devices are often composed of many small parts that need to be disassembled. Although the original SoftHand features good

adaptability to objects of different shapes, this capacity is limited when small parts require a precision grasp, as pointed out in the first periodic project report. The first prototype of the SoftHand2 is an evolution of the SoftHand that implements the first and the second synergy of the human hand to allow for precise pinch grasp while preserving the more general adaptive power grasp. Based on the design of the first prototype, QBR developed the qb SoftHand2 Research as described in more detail in deliverable **D4.2**. Its market launch was this year at ICRA 2022. The qb SoftHand2 Research is able to successfully grasp not only the whole heat cost allocators such as Kalo 1.5 but also its internal components such as the battery. The video showing these functionalities was prepared in collaboration between IIT and QBR.

6 Teaching contact skills and adaptation to unknown object geometry with force controller

Tactile robots are useful for dynamic task execution in recycling environments where a large variety of devices need to be disassembled. Therefore, these robots should have the ability to respond to changing conditions and be easy to (re-)program. Operating under such uncertain environments for recycling requires unifying subsystems such as robot motion and force policy into one framework, referred to as tactile skills.

To define and enhance the tactile skills used in ReconCycle, we propose a framework submitted to IROS2022 [2]. To achieve the increased level of adaptability, we represent all tactile skills by three basic primitives: contact initiation, manipulation, and contact termination. To ensure passivity and stability, we develop an energy-based approach for unified force-impedance control that allows humans to teach the robot motion through physical interaction during the execution of a tactile skill. While the polishing of a car-door task is presented as our use case in the accompanying video, various tactile skills, such as levering in the scope of ReconCycle, can also be carried out.

To handle unexpected changes of contact surfaces for the heat-cost-allocators and adjust the tool-contact alignment, we extend unified force-impedance control. In the work submitted to ICRA2023 [3], we propose a framework incorporating an adaptation metric into force-impedance control such that tactile skill policies can be updated online and progressively via impedance shaping w.r.t the physical constraints of the environment. Even though, we use the polishing skill as our use-case, any tactile skill, which requires the robot applying motion with desired force, can be expressed by our framework. Impedance shaping and force shaping introduces improved adaptability and robustness to ReconCycle skills onto an unknown and changing surface.

7 Learning of disassembly graphs in constrained environments

The final video shows how a robot can learn a disassembly graph for a given device by autonomous exploration. The aim is to extract a new disassembly skill sequence for devices where the optimal sequence is unknown. For this purpose, we developed a new hierarchical learning scheme where the disassembly task is represented as a directed graph. The graph is built of nodes and branches, which respectively correspond to the states and actions in the disassembly process. The upper hierarchical level is essentially a decision-making algorithm. The optimal decision-making is acquired using reinforcement learning (RL) techniques. The

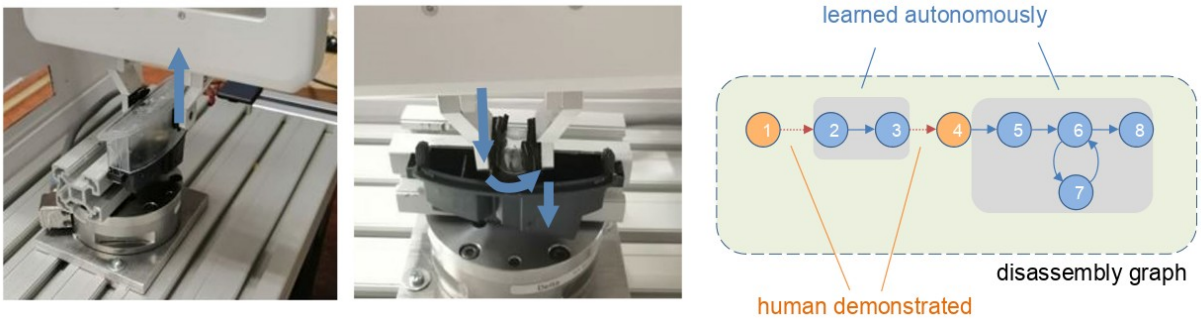


Figure 11: Disassembly of a licence car plate light. Left: removal of the transparent cover, center: disassembly of the bayonet bulb, right: disassembly graph for the entire disassembly procedure involving both operations.

explorative actions are generated by a constraint-space following (CSF) controller, which looks for directions where motion is possible using force control. The controller generates explorative robot motion by adjusting its stiffness in the direction defined by the Frenet-Serret frame attached to the robot path. The executed robot motion trajectories are stored as Cartesian space dynamic movement primitives and added to the disassembly skill library.

The proposed framework was experimentally verified on a task of learning the disassembly of a bayonet mount bulb from the casing. To start the learning process, the robot was first manually guided to a position where it can grasp the bulb. From this initial configuration, the above-described learning process was started. Due to the constraints imposed by the environment, the robot needed only two learning cycles to find the optimal disassembly policy.

The second part of the video shows a more complex example of the car licence plate light, where first the plastic housing needs to be removed as shown in Fig. 11. Afterwards, the controller proceeds with the disassembly of the bulb in a similar manner as for the 3D printed mock-up case.

The paper presenting the overall framework is currently under review [8]. In addition, we developed an invariant representation of a disassembly policy [6]. This formulation can be used for force-based adaptation of disassembly policies for damaged devices, where the originally learnt disassembly policy should be adapted.

8 Conclusion

With videos included in this deliverable, we present a solution for the disassembly of two different heat cost allocator devices using the same workcell. For this purpose we developed suitable adaptation and reconfiguration methods. Video 1 demonstrates that the ReconCycle workcell can perform the disassembly of different heat cost allocators. The required steps to perform the disassembly are obtained on-the-fly by the vision-based action prediction framework. Video 2 shows the details of adaptation of the levering operation for the extraction of PCBs from different heat cost allocators.

The operation of a system for automatic gripper reconfiguration is shown in Video 3. This is important to support the disassembly of a larger variety of devices. Video 4 shows an upgraded version of qb SoftHand2 Research, which allows to precisely grasp small objects such as batteries.

In Videos 5 and 6 we demonstrate some advanced methods for skill adaptation, which can contribute to the overall flexibility of the recycling cell.

References

[1] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal. "Dynamical movement primitives: learning attractor models for motor behaviors". In: *Neural Computation* 25.2 (2013), pp. 328–73.

[2] K. Karacan, H. Sadeghian, R. J. Kirschner, and S. Haddadin. "Passivity-Based Skill Motion Learning in Stiffness-Adaptive Unified Force-Impedance Control". In: *IEEE-RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2022.

[3] K. Karacan, H. Sadeghian, F. Wu, and S. Haddadin. Progressive Adaptation of Tactile Skill Policy to Physical Constraints. Submitted to 2023 *IEEE International Conference on Robotics and Automation (ICRA)*.

[4] R. J. Kirschner, A. Kurdas, K. Karacan, P. Junge, S. Birjandi, N. Mansfeld, S. Abdolshah, and S. Haddadin. "Towards a Reference Framework for Tactile Robot Performance and Safety Benchmarking". In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2021, pp. 4290–4297.

[5] B. Kuster and M. Majcen Horvat. "Human-Inspired Robotic Levering Using Periodic Dynamic Movement Primitives". In: *International Electrotechnical and Computer Science Conference (ERK)*. 2022, pp. 212–215.

[6] B. Nemec, M. Majcen Hrovat, M. Simonič, S. Shetty, S. Calinon, and A. Ude. *Robust Execution of Assembly Policies Using Invariant Task Representation*. In preparation.

[7] P. Radanovič, J. Jereb, I. Kovač, and A. Ude. "Design of a modular robotic workcell platform enabled by plug & produce connectors". In: *20th International Conference on Advanced Robotics (ICAR)*. 2021, pp. 304–309.

[8] M. Simonič, A. Ude, and B. Nemec. *Hierarchical learning of robotic contact policies*. 2022, submitted to a journal, preprint available. DOI: 10.2139/ssrn.4237312.

[9] M. Simonič, R. Pahič, T. Gašpar, S. Abdolshah, S. Haddadin, M. G. Catalano, F. Wörgötter, and A. Ude. "Modular ROS-based software architecture for reconfigurable, Industry 4.0 compatible robotic workcells". In: *20th International Conference on Advanced Robotics (ICAR)*. 2021, pp. 44–51.