European Commission

This project has received funding from the European Unions Horizon 2020 research and innovation programme under grant agreement №820742

HR-Recycler

HR-Recycler: Hybrid Human-Robot RECYcling plant for electriCal and eLEctRonic equipment

D6.3 - Grasping Strategy Evaluation

WP number and title	WP6 - Robotics actions planning and control
Lead Beneficiary	TUM
Contributor(s)	IBEC, SDK
Deliverable type	Report
Planned delivery date	November 28, 2019
Last Update	March 4, 2020
Dissemination level	PU

Disclaimer

The HR-Recycler Consortium consists of the following partners:

No	Participant organisation name	Short	Type	Country
1	CENTRE FOR RESEARCH AND TECHNOLOGY HELLAS	CERTH	RTO	GR
2	FUNDACIO INSTITUT DE BIOENGINYERIA DE CATALUNYA	IBEC	RTO	ES
3	TECHNISCHE UNIVERSITAET MUENCHEN	TUM	RTO	DE
4	COMAU SPA	COMAU	IND	IT
5	FUNDACION TECNALIA RESEARCH & INNOVATION	TEC	RTO	ES
6	ROBOTNIK AUTOMATION SLL	ROB	SME	ES
7	FUNDACION GAIKER	GAIKER	RTO	ES
8	SADAKO TECHNOLOGIES SL	SDK	SME	ES
9	DIGINEXT	DXT	IND	BE
10	VRIJE UNIVERSITEIT BRUSSEL	VUB	RTO	BE
11	INDUMETAL RECYCLING, S.A.	IND	SME	ES
12	INTERECYCLING - SOCIEDADE DE RECICLAGEM	SA	INT	PT
13	BIANATT ANAKYKLOSI AIIE ANONIMI BIOMICHANIKI EMPORIKI ETAIRIA	BNTT	IND	GR

Document History

Version	Date	Status	Author / Reviewer	Description
0.1	16.11.2019	Draft	TUM	Initial Draft
0.2	21.11.2019	Rev 1	Dr. D. Giakoumis (CERTH)	Internal Review
0.3	26.11.2019	Rev1	SDK	Added Section 4.1 and 5.1.1
0.4	28.11.2019	Rev1	IBEC	Added Section 5.6
0.5	29.11.2019	Final	TUM	Final Version
1.0	26.02.2020	Final	TUM	Revised after Review

Acronyms

CAD Computer-aided design.

D Deliverable.

FEL Feedback Error Learning.

HR-Recycler Hybrid Human-Robot RECYcling plant for electriCal and eLEctRonic equipment.

HSPC Hierarchical Sensory Predictive Control.

RGB Red-Green-Blue.

ROS Robot Operating Software.

T Task.

TCP/IP Transmission Control Protocol/Internet Protocol.

WEEE Waste of Electrical and Electronic Equipment.

WP Work Package.

List of Figures

Table of Contents

Executive Summary

The Hybrid Human-Robot RECYcling plant for electriCal and eLEctRonic equipment (HR-Recycler) project aims for a deployment of robotic systems in industrial disassembly sites. A depollution step is a mandatory part of the disassembly process for several types of waste and includes the removal of fragile components wherein their breakage presents environmental hazard and adversely affects the health of workers in the vicinity. Together with the significant economic costs of human labor in the time-consuming removal of such components, the safety-critical aspects make an automated robotic solution especially attractive. Thus, the HR-Recycler proposes a new robotic grasping strategy, allowing precisely controlled grasps for objects whose dimensions and locations are uncertain a priori. In order to fulfill the HR-Recycler use-case requirements defined under Deliverable (D)3.1: *User requirements and use cases*, this D6.3 describes a force-adaptive grasping strategy based on vision and tactile feedback, whose conception is associated with Task (T)6.3: *Versatile force-adaptive object grasping.* To achieve this, the main body of this deliverable is split into three stages. First, the requirements for the grasping strategy are worked out from the use cases. Second, a literature and state of the art survey is conducted, to inform how these requirements might be met with existing methods. Third, hardware choices and the proposed grasping strategy are developed based on the requirements and the state of the art methods with new extensions.
Hardware choices to this purpose include the gripper, a tactile sensor matrix for force feedback that can be attached to the fingers of the gripper and the camera setup used for vision. The proposed grasping strategy consists of several steps, which are

1. object localization and pose estimation from vision data, providing initial but uncertain information for the grasping configuration,
2. initial grasping point selection and gripper movement to the initial grasping point,
3. readjustment of the gripper position and orientation based on tactile feedback, aligning the gripper with the object in the presence of uncertainties regarding object position, pose and dimensions,
4. closing the gripper with controlled force.

As a preparatory work towards implementation, this deliverable focuses on theoretic concepts and introduces several methods separately. An integrated approach will be implemented and experimentally evaluated later in the project, contributing to D6.4: *Flexible probing* (due in Month 24 of the project).

State of the Art. The state of the art for robotic grasping in the context of the relevant use case (see chapter 2) is summarized below. For a more general and detailed overview, see chapter 3.

- In Industry, grasping typically with predetermined position and orientation without continuous control.
- Force feedback based on force estimation from motor current.
- Limited research with grippers that have force sensor arrays.
- Complex grasping point selection and grasp synthesis algorithms that rely on accurate object knowledge.
- Trial and error grasp synthesis with Machine Learning for unknown objects, but:
  - Many trials needed.
  - Data availability as bottleneck.
  - No incorporation of safety critical constraints.

HR-Recycler Contribution. The contribution of this deliverable beyond the state of the art is a novel grasping strategy (outlined in chapter 5) that does not rely on accurate object knowledge:

- Based on force sensor arrays.
- Estimation of relative alignment from force feedback.
- No prior grasp synthesis needed $\Rightarrow$ no accurate object model necessary.
- Grasp adapted online, alignment controlled to zero based on force feedback $\Rightarrow$ no accurate positional information necessary.

1 Introduction

This deliverable is an intermediate report of T6.3: Versatile force-adaptive object grasping. In the recycling of Waste of Electrical and Electronic Equipment (WEEE) materials, some parts of objects need to be extracted in a depollution step. In the use cases for Hybrid Human-Robot RECYcling plant for electriCal and eLEctRonic equipment (HR-Recycler), some of those parts are safety-critical and sometimes fragile so that they need to be handled with care. Typically, the extraction is manually performed by humans, which is an expensive solution, so that a robotic automated solution is desirable at least in part of the process. To this end, a robot should entice robust yet precise grasping capability and *careful* interaction strategies. This deliverable describes the initial decisions and concepts on grasping strategy for object dissasembly.

1.1 Overview

Every robot manipulation task fundamentally requires a capability to handle physical contact and the resulting forces between the robot and the environment, specifically between the grasping fingers and the object in the case of object grasping. The direction and magnitude of forces applied by the fingers are critical for success of the manipulation task, and the grasping force must stay within the friction cone of the object in order to stables the object in hand; insufficient force, or a misdirected force vector angle would results in object droppage. On the other hand, excessive force may lead to breakage of the robot or the object. Furthermore, the HR-Recycler project envisions object manipulation in the vicinity of and possibly in collaboration with humans. This mandates dependable and safe operation of the robot. Furthermore, the robot is expected to manipulate many different objects in possibly changing, unstructured or partially unknown settings. Thus, the grasping of the robot must be robust and versatile. To this purpose, pure motion control is typically inadequate, because the unavoidable modelling errors and uncertainties may cause a rise or drop of the contact force, ultimately leading to an unstable behaviour during the interaction [22]. Thus, force feedback and force control become necessary components for HR-Recycler. In order to establish force feedback and force control, an estimate or measurement of the current force acting on the gripping fingers (and therefore on the object) is needed (Figure 1.1). The first goal of this deliverable is to investigate different applicable force estimation or measurement techniques, and nominate a suitable force sensing solution. Based on the error between desired and measured forces, a robot controller will be able to 1) guide and align the gripper with the object and 2) loosen or tighten the grip, so that forces are applied in the desired direction and magnitude. The second goal of this deliverable is to present and discuss the proposed control strategy envi-

sioned for force-adaptive grasping, including issues such as derivation of grasping positions and forces, alignment of the robot finger along the object surface, and a control method of the grasping force modulation. The third goal of this deliverable, is to present hardware choices and discuss the envisioned technical implementation.

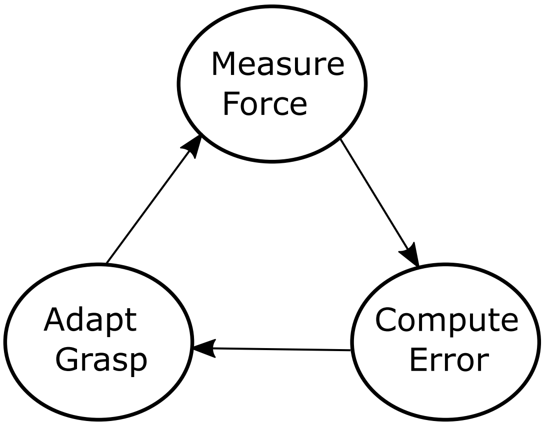


Figure 1.1: Force-Adaptive Grasping Strategy.

Connection to other Tasks, Work Packages and KPIs

This deliverable and T6.3 in general are connected to other work packages and Tasks within HR-Recycler. First, computer vision associated with Work Package (WP)5 will be used to estimate initial position and pose of the object to be grasped, and an initial grasping point (position of the finger grip on the object) will be selected based on the 3D object surface model provided by the vision pipeline. Second, force guided manipulation in T6.1 will be part of the grasping strategy. Both here and for T6.1, the robot motion and stiffness behaviours are to be controlled in a scenario of physical interaction with the environment. To control the forces in this physical interaction, control strategies related to variable impedance control will be used.
Related KPIs are

- KPI 1.4: Ability to cope with perception uncertainty in real-world environments.
- KPI 4.1: Robots successfully grasp known or unknown objects: success rate exceeds 70%.
- KPI 4.4: Increase of manipulation success rate through force feedback and adaptive control by 50%.

1.2 Structure of the deliverable

This D6.3 is structured as reported below:

Chapter 1 - Introduction - Provides an overview of D6.3.

Chapter 2 - Problem Statement - States the problem and use case requirements this deliverable aims towards solving, which is robot grasping for object disassembly within HR-Recycler.

Chapter 3 - State of the Art - Describes the state of the art in industry and research for solving robotic grasping.

Chapter 4 - Hardware Choices - Describes the hardware choices and setup for force-adaptive grasping in HR-Recycler disassembly.

Chapter 5 - Proposed Grasping Strategy - Introduces the proposed methods and concepts for solving robotic grasping in HR-Recycler disassembly.

Chapter 6 - Conclusion and Outlook - Gives a conclusion describing how the proposed grasping hardware and methods fit the requirements of the use cases and provides an outlook toward future work within T6.3.

2 Problem Statement

The work package description of WP6 states that one of its goals is to introduce a force-adaptive grasping strategy with tactile sensing gripper fingers for safe grasping of different components. This refers to T6.3, versatile force-adaptive object grasping. In the task description, it is specified that some components such as mercury lamps in LCD flat screens have to be handled with care. This requires a force-adaptive grasping strategy with tactile sensing gripper fingers which would allow precise grasping. It must be avoided that significant forces and torques are applied in directions other than the ones desired for disassembly. In HR-Recycler, there are several use cases (see D3.1) associated with precise force-adaptive grasping for components, where a component needs to be removed without breakage, rendering cutting or blind forceful grasping unviable:

- Hg fluorescent tubes in emergency lamps contain mercury and must be removed compulsory without breaking.
- Backlighting lamps in Flat Panel Displays contain mercury and must be removed compulsory without breaking.
- Capacitors in microwaves, must be removed compulsory without breaking.

In the following, we focus on Hg fluorescent tubes as an exemplary use case and derive requirements for the HR-Recycler's grasping strategy for disassembly.

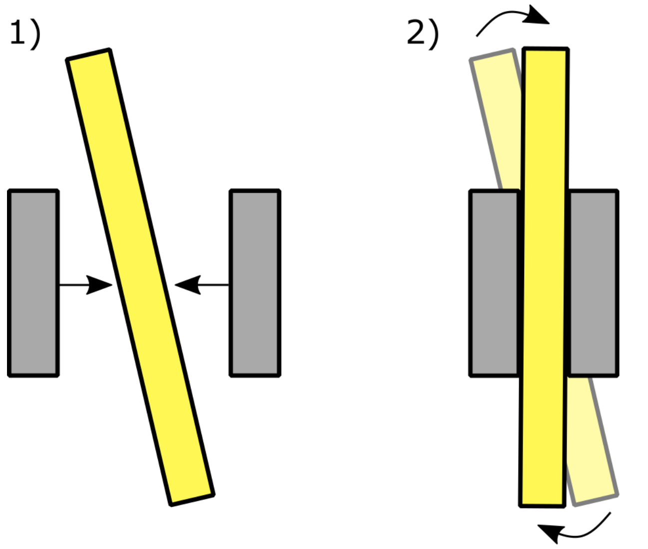
Use Case: Emergency Lamps

In the recycling of emergency lamps, the Hg lamp tube must be removed from the casing in order to depollute the object. The depollution of emergency lamps is very intensive in human labour, making a semiautomatic recycling process that increases productivity especially attractive for this use case. Due to safety reasons, it is critical that the Hg lamp does not break in the process. If a lamp were broken, the mercury contained inside would spread into the environment, which introduces a health hazard and a time- (and therefore cost-) expensive clean up of the area is needed. Therefore, a grasping point and angle should be optimized so that the pressure exerted on the object is minimal. For grasping a cylindrical tube, in particular, the gripper surface should be exactly parallel with the (long) axis of the glass tube.

Alignment

The Hg lamp is typically held in place on both sides by a fixture and needs to be rotated for extraction. The fixture constrains the object against force in one direction, but collision

(a) The emergency lamp. Fixtures on both ends hold the Hg tube in place.



(b) Top view on the gripper (grey) and the emergency lamp (yellow). The lamp would turn under a misaligned grasp if it was not fixed in place.

Figure 2.1: The emergency lamp use case. If the gripper is not perfectly aligned with the emergency lamp, the grasp will exert a moment on the lamp, risking breakage due to the fixtures.

may lead to breakage. As this is a safety-critical task, it has considerable implications regarding possible robotic solutions in comparison to more typical pick and place scenarios in the literature (see Chapter 3). For example, the gripper needs to be very precisely aligned with the object to be grasped. For anything but very precise alignment, forces will build up and break the lamp (Figure 2.1b). In the research literature and typical pick and place scenarios where robots pick up objects from a table top or conveyor belt, this is most often not the case and objects can and will move once the gripper closes. With the requirement of precise grasp alignment comes a requirement of precise knowledge of the relative position of the object. Generally, we can not expect to gain this precise knowledge by the vision alone under uncontrolled environments due to lighting conditions and occlusions. To significantly reduce uncertainty of the relative position of the object, therefore, we propose to employ tactile feedback in the form of haptic sensing on the fingers (a sense of touch). Tactile feedback and force sensing is part of the literature overview in the next chapter and chosen tactile sensors and alignment concepts are part of Chapters 4 and 5 respectively.

Grasping Force

Another requirement of the identified robotic solution is connected to uncertainty in the dimension, particularly the width, of the Hg lamps. Since Hg lamps are fragile, excessive pressure applied to the hull of the lamp may break it, even for a precisely aligned grasp. This implies that forces need to be controlled and kept within reasonable bounds. As the use case requires rotating the lamp out of the fixture after grasping it, a unified

force control strategy is needed that allows manipulation of the object (in the sense of movement of the end effector) while keeping forces within a desired range. In addition to the force-adaptive task-optimal gripper alignment, the grasping force modulation is a crucial component in successful disassembly.

Generalization

As emergency lamps are treated in the same recycling stream with other small lighting equipment, the robotic solution should be generalizable to other similar objects (see D3.1). Therefore, it is important to keep the grasping strategy as general as possible, while still fulfilling the main specification of grasping Hg lamps. Feedforward modules and controllers should be able to adapt their strategies or gains based on sensory feedback and associated predictions. The idea is, that the proposed control strategy, partly based on the cerebellum, can result in successful adaptive grasping, without having to implement specific force controllers for individual objects.

3 State of the Art: Force Adaptive Grasping

Robotic grasping and manipulation strategy is a highly active research field. Yet, there currently exists no benchmarks and performance metrics suitable for comparison of different approaches, partly because robot grasping is highly dependent on the gripper and sensors in use. Thus, there is no fully developed and generally applicable grasping strategy that could be employed for HR-Recycler out of the box. In the following, we provide an overview of relevant current methods and developments. In Chapter 5, we pick from and build on these methods to tackle the problem (see Chapter 2) at hand. Much robotics research focuses on anthropomorphic robot hands for human-like manipulation. However, as our goal is to achieve reliable, precise and adaptive robot grasping in an industrial setting, engineering designs mimicking human hands are not ideal for HR-Recycler. Anthropomorphic robot hands are costly, fragile and not easy to repair, which would make them a liability in the envisioned HR-Recycler scenario. Besides, human hands offer much more manipulation capabilities than is needed for the given grasping task. We expect that comparably simple robot gripper configurations are sufficient, while requiring less engineering effort. Despite decades of research and technical development, modern robots commonly fail domestic tasks such as sort and package objects, chop vegetables or fold clothes in unstructured and dynamic environments [4]. In most production lines in industry, robotic systems are designed to manipulate objects with a known Computer-aided design (CAD)-model. The complete knowledge about the object and the location, execution of a task becomes much simpler as, for example, a set of grasping points can be pre-defined with a specifically built gripper. Nevertheless, a complete CAD model is not available in the scope of the HR-Recycler project, as the manipulating objects can vary not only in types but also in states due to fractures, damages and dirt.

3.1 Research Literature Overview

Robotic object grasping has been the topic of much research since the 1970s. Object grasping is seen as a fundamental and significant ability of robots, as it is expected to bring enormous productivity to the society [20]. Billard et. al [4] identify two main drivers for advancements made in the last ten years. One is breakthroughs in mechanics for sensors perceiving touch. The other is the leveraging of the immense progress in machine learning which can be used to identify uncertain models and parameters online, and support adaptive and robust control schemes, although learning to manipulate in real-world settings is costly in terms of both time and hardware.

Grasp Synthesis

When grasping an object, the specific locations on the object at which the fingers of the gripper (more precise: the fingers' center point) make contact, are called the grasping points. The location of grasping points on the object drastically influences the stability of the grasp, forces exerted on the object and forces and torques experienced by robot joints when holding or moving the object. Especially relevant, is whether a grasping point is close to the center of mass and how weight is distributed around it. To us humans, this is obvious when lifting any heavy object while grasping it far away from its center of mass: It is very hard to lift even a light chair by grabbing it on the end of a foot. It becomes very easy to do so if we can grab it at the backrest. Selecting a grasping point is part of what is called *grasp synthesis* in the robotics research literature. Given an object at a certain location within the robot's work space, grasp synthesis refers to the problem of finding a configuration of the robot that describes a suitable grasp. This generally includes finding

- a suitable grasping point on the object, with which the center point of the robot gripper should be aligned
- an approach vector or a similar description of the 3D angle that the gripper approaches the grasping point with
- the associated joint configuration of the robot

In the research literature (see [19, 5]), methodologies are often divided into *analytic* and *empirical*. Analytic grasp synthesis is usually formulated as a constrained optimization problem over criteria that measure stability or equilibria of forces, based on geometric, kinematic or dynamic formulations. Typically, these criteria are related to *form closure* and *force closure*. For form closure, kinematic constraints are applied to an object such that the object cannot perform any relative motion. The finger of the grippers restrict any movement. For force closure, a set of contact points is considered such that contact forces can balance an arbitrary external wrench. In general, deformable objects fail to be immobilized by force closure [20]. For rigid bodies and under assumptions regarding coulomb friction and simplified contact models, theoretic guarantees can be provided regarding dexterous, equilibrium, stable and certain dynamic behavior. However, theoretic guarantees of such nature are never iron-clad, and all analytic approaches rely on accurate models of the gripper, the object to be grasped and their relative alignment in space. In practice, systematic and random errors due to inaccurate models of the object, the robot's kinematics and dynamics and noisy sensors permeate every robotic system. In HR-Recycler, we can not expect to have precise geometric and physical models of objects to be grasped. Even for identified WEEE-components (e.g., brand and type of a microwave), we might have unknown weight, center of mass and weight distribution due to missing parts or damage to the object. In addition, we will not know surface properties and friction coefficients, which vary greatly with dust and dirt on the surface of the object.

Empirical approaches

Empirical or *data-driven* approaches rely on sampling candidates of possible grasp configurations and ranking them according to some metric. Metrics might be based on heuristics

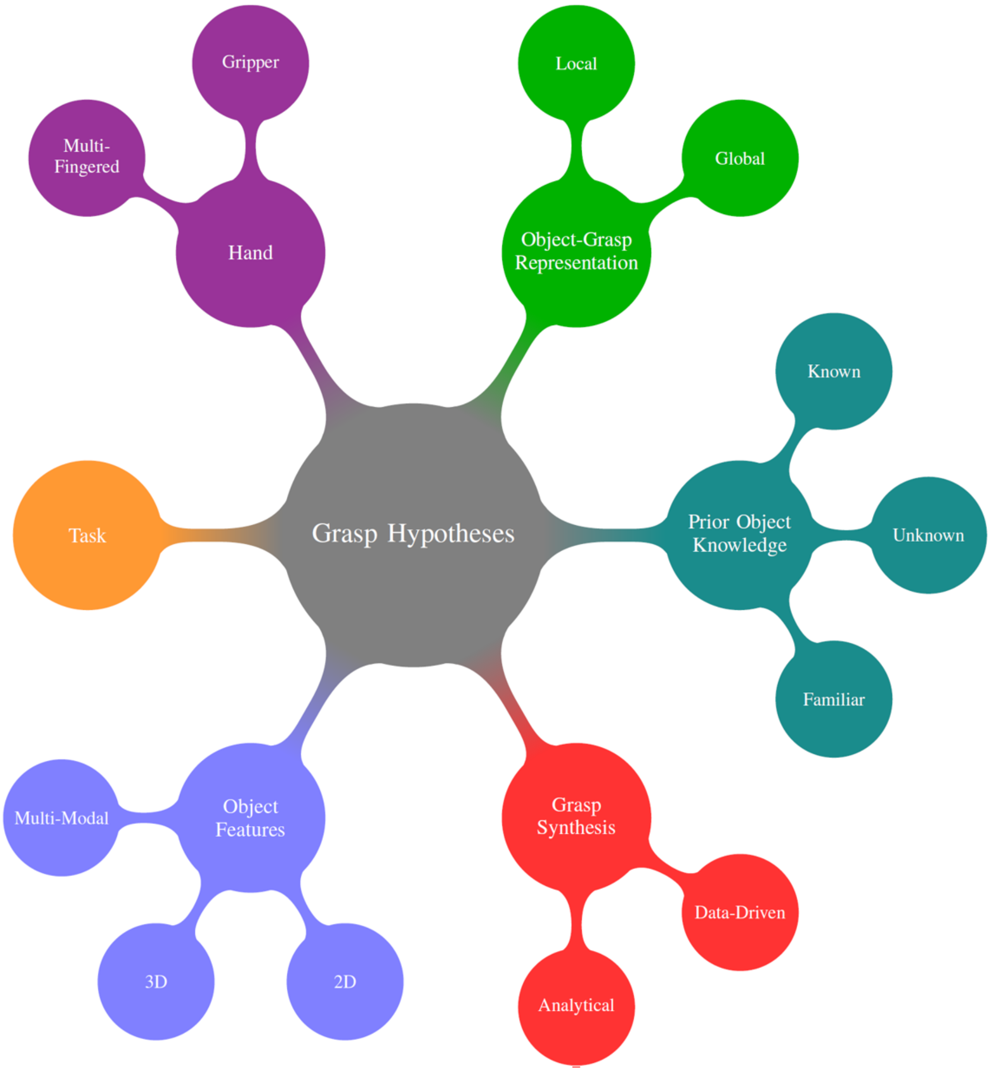


Figure 3.1: Identified aspects that influence the grasp hypotheses, taken from [5]. Prior object knowledge is identified as the most influential. Grasp synthesis can be data-driven or analytic, object-grasp representations can pertain parts of the object (local) or the whole object and rely on features extracted from 2D or 3D vision or tactile sensors. Furthermore, the grasping strategy is influenced by the task at hand and the gripper choice.

or generated in simulations. Data-driven grasp synthesis started to become popular with the availability of GraspIt! [16] in 2004 [5]. Many approaches have been developed in GraspIt! or other simulators since then, differing in terms of how candidates for grasp configuration are selected from the infinite space of possible configurations. For grasp ranking, metrics based on analytic formulations such as the widely used ϵ-metric [11] are typically used. The ϵ-metric (or *criterion of the largest ball* [18]) quantifies the quality of a grasp by the radius of the maximum sphere still fully contained in the convex hull over the wrenches at the contact points between the hand and the object. Roughly, this quantifies the quality of the force closure in terms of maximal wrenches (3D forces and 3D torques that can be externally applied to the object without breaking the grasp). Related and different grasp quality measures exist. A comprehensive survey is out of the scope of this deliverable, and can be found in [18].
Grasp candidates can be developed and evaluated in simulation, where a large number of experiments can be efficiently performed without much cost or time spent. However, it is never clear a priori if the simulated environment resembles the real world well enough to transfer solutions successfully. Bogh et. al [5] cite several studies that analyzed this question. The consensus conclusion is that classic metrics in simulation do not predict grasp success in the real world particularly well and resulting grasp configurations appear to be relatively fragile. One example is the study by Balasubramanian et. al [2], in which different grasps that were stable according to classical analytic metrics were tested in the real world. Compared to grasps planned by humans and transferred to a robot, classically synthesized grasps performed significantly worse. Another study [25] showed that the ϵ-metric poorly predicts grasp stability under errors of object pose estimation, especially for large objects. Since grasp closure is not equal to stability (which really concerns the behavior of the gripper after a perturbation from a grasping position in dynamic equilibrium, see Bicchi and Kumar [3]) the results of these studies are not surprising [5]. However, they show that currently available simulation models for grasping do not accurately reflect reality.

Improvement beyond classical metrics

One way to generally improve the success rate of transfer from simulation to the real world, is *domain randomization*. The idea is to generate a distribution of randomized plant models around the initial best guess for a plant model, with the hope that the real system is represented somewhere in that distribution. Control policies can then be evaluated over the whole distribution in simulation. If a control policy solves the given task with success for all possible plant configurations, then the real system can be successfully controlled by this policy. OpenAI et. al [9] used an automated version of this principle to learn a difficult real-world manipulation task (solving a rubik's cube with a robot hand) in simulation. The same concept can be applied to data-driven grasp synthesis. If grasping metrics do not accurately reflect real world grasp quality, but come close enough, a randomization of both metrics and models might lead to better success rates. Another way to improve grasping success is to let the robot learn how to grasp from data collected during execution. Although real-world data collection, i.e. letting the robot try grasps on a real object, takes a lot of time, the problem of finding very accurate grasping metrics is circumvented. Real-world machine learning approaches for data-driven grasp synthesis

have received a lot of attention in recent years (see for example [5, 9, 22] for an overview) and range from perfecting reasonably good grasp configurations obtained from simulation to tabula rasa learning a direct mapping from vision features to grasping points. From 2009, there were further developments in the area of 3D sensing, including the release of the Kinect, a highly accurate low cost depth sensing device. Although the importance of 3D data for grasping has been previously recognized, many new approaches were proposed that operate on real world 3D data. They are either heuristics that map structures in this data to grasp configurations directly or they try to detect and recognize objects and estimate their pose [5].
Contrary to analytic approaches, data-driven methods typically rely on perceptual pre-computations, such as feature extraction, similarity metrics, object recognition or classification and pose estimation [5] for retrieving grasps from a database or sample and rank them by comparison to previously performed grasps. Since the parameterization of the grasp hypotheses is less specific (e.g. an approach vector instead of fingertip positions), uncertainties in perception and execution do not influence grasping success as much. As further alluded to later, this provides a natural precursor to reactive grasping, which is the problem of robustly acquiring a given grasp hypotheses under uncertainty. However, data-driven methods cannot provide the same a priori theoretical guarantees as analytic methods, and can only be verified empirically [21]. Bogh et. al [5] group data-driven methodologies into three categories, based on a priori knowledge of the object to be grasped:

- *Known Objects*: These approaches assume that the object to be grasped has been encountered before. Successful grasps have already been synthesized and saved in a database. The grasping strategy is to recognize the object, estimate its pose and retrieve a suitable grasp.
- *Familiar Objects*: These approaches assume that the object to be grasped is not exactly identical, but similar to previously encountered ones. Object similarity needs to be quantified according to some metric based on properties such as shape, size, texture or object category. The underlying assumption is here, that similar objects can be grasped in a similar way. The challenge is to find an object representation and similarity metric that allows the transfer of grasp experience.
- *Unknown Objects*: These approaches assume no access to object models or any previous grasp experience. Grasping can be solved by identifying structural features from sensor data and generating and ranking grasp candidates.

It should be noted here, that methods for unknown objects are important in any case, since both methods for known and familiar objects rely on first setting up a database for the objects at hand, in general making use of some form of grasp synthesis without any previous grasp experience. In HR-Recycler, we may encounter all three categories of object familiarity. We could know the type of a WEEE device, have seen or never seen the brand or even exact model of WEEE device before and there might be breakage or parts missing. Generally, methodologies for familiar objects seem to present the most practical approach to the problem in the HR-Recycler setting, as they are capable of handling new objects (which will in practice always be somewhat related to previously seen ones) and making use of previous grasp experience.

Learning to grasp new objects

Let us consider the goal of having a robot helping with device disassembly in a WEEE recycling plant for the foreseeable future without restricting itself to certain types of WEEE and undamaged products only. In this scenario, we cannot expect that the programmer has foreseen all the different objects and situations that this robot will be confronted with. Additionally we will not be able to rely on having 3D models available for all objects encountered. Therefore, the ideal disassembly robot should be able to learn about new objects and how to grasp (and more generally manipulate) them *while* operating in the recycling plant. Several open research questions arise in this context [5]: How can the experience of having grasped an object be represented in memory, so that it is most suitable for learning (generalizing) from it? How can we quantify similarity between objects and guarantee successful transfer of grasping strategies? How can success and failure of a grasp be autonomously quantified? Can we bootstrap learning by employing human demonstration or training in simulation? The problem-setting of grasping familiar objects, as discussed above, is related to these questions and research is progressing on many fronts. However, no method exists that answers all of them in a satisfying way.

Autonomous Grasp planning

A lot of robotics research and applications in industry considered robot grasping as an isolated problem, such as in a pick and place scenario, where the single task of the robot is to pick an object from a table top in a research lab or from a conveyor belt in a factory (and release the object somewhere else). On the contrary, when considering grasping as part of a more complex robot action, higher-level tasks influence what the best grasp in a specific scenario might be. Examples in the context of HR-Recycler could be for the robot not to grasp certain devices at a part made of glass, not grasp a microwave oven by the door, or to not obstruct screws with the gripper when holding a device for human (or robot) disassembly. Task constraints have not yet been considered extensively in the research literature. Current approaches make use of Machine Learning [23] or semantic planning [6].

Robustness and Feedback

Inferring a grasp for a given object is necessary, but not sufficient for successful robotic grasping. Only if the execution is robust to disturbances and uncertainties in sensing and actuation, can a grasp succeed with high probability [5]. To achieve this, and to adapt to unforeseen situations, feedback is necessary. In the literature, there are a number of approaches that use either tactile or visual feedback during grasp execution to adapt to unforeseen situations. While tactile feedback relies on haptic measurements or force-torque sensors, visual feedback results from tracking the robot gripper and object simultaneously. Also in this area, open research questions remain [5]: How can tactile feedback be interpreted to choose an appropriate corrective action independent of the object, the task and environment? How can visual and tactile information be fused in the controller? Dang and Allen [7] use tactile sensing data to estimate grasp stability and make necessary hand adjustments after an initial grasp is established. This is done by using a learning approach to derive the relationship between tactile sensing data and grasp stability. Hsiao et. al [13]

propose a data-driven grasp synthesis with additional reactive adjustments based on tactile feedback from fingertip sensors. These reactive adjustments are shown to partially correct for uncertainty in the measured position and shape of the object. Felip and Morales [10] present a grasping controller that uses feedback from different haptic sensors to correct for uncertainty in the pose of the object. Sensors perceiving touch have been part of robotics for a long time. In contrast to vision, which is now widely used in industrial applications, tactile sensing always seems to be just out of reach from widespread applicability [22]. In nature, tactile sensing is essential. Receptors responding to mechanical pressure are abundant in the animal kingdom and in humans and the sense of touch is indispensable for effectively grasping objects of various shapes, sizes and materials. To better understand how feedback control can be introduced to adjust the grasp, that is to adjust forces applied to the object, we provide a general overview of force control for manipulation in the next section.

3.2 Force Control in Robotics

In practice, task or path planning errors (the robot imagines the object to be grasped at wrong location or misunderstands its shape) may give rise to an unforeseen contact force and moment (end effector colliding with object), causing a deviation of the end effector from the desired trajectory. A naive control system responsible for guiding the end effector to the correct position or along the desired path will react to reduce such deviations. This ultimately leads to a build-up of the contact force until either joint actuators saturate or parts in contact break. The higher the environment (object) stiffness and position control accuracy are, the more easily a situation like the one just described can occur. This drawback can be overcome if a compliant behaviour is ensured during the interaction. This compliant behaviour can be achieved either in a passive or in an active fashion.

Passive compliance

In passive interaction control the trajectory of the robot end effector can be changed by interaction forces due to an inherently compliant nature of the robot. An illustrative example is soft robot arms with elastic joints or links. In industrial applications, an end effector (called Remote Center of Compliance, or RCC) with passive compliance is widely adopted (see for example [24]). The passive approach to interaction control has several advantages. First, it is very simple and cheap (provided the robot is already designed compliant), because no (force/torque) sensors are required. Second, robot trajectories do not need to be re-planned during execution time, and third, the response time of a passive compliance mechanism is immediate and much faster than any active position or force control by an algorithm running on a computer. However, passively compliant systems lack flexibility. For every robotic task, a special-purpose compliant end effector has to be engineered. Furthermore, passively compliant systems can typically only deal with small position and orientation deviations and have a certain range limit beyond which they cannot be pushed any further. Finally, since no forces are measured, there is no guarantee that high contact forces will never occur [22].

Active compliance

In active interaction control, a purposefully designed control system ensures compliance of the robot. This approach usually requires the measurement of contact forces and moments, which are fed back to the controller and used to modify or even generate online the desired compliant trajectory of the robot end-effector. Active interaction control may overcome the aforementioned disadvantages of passive interaction control, but it is usually slower, more expensive, and more sophisticated [22]. To obtain a reasonable task execution speed and disturbance rejection capability, active interaction control has to be used in combination with some degree of passive compliance. For a general force-controlled task, three translational force components and three torques need to be measured or estimated to know the full state of the contact force. The majority of the applications of force control is concerned with force/torque sensors mounted at the robot wrist. Additionally, force (or pressure) sensors can be placed on the fingers of the robot gripper. The Springer Handbook of Robotics [22] groups active interaction control strategies into two categories: those performing indirect force control and those performing direct force control. Where the former achieve force control via motion control, without explicit closure of a force feedback loop; the latter instead offer the possibility of controlling the contact force and moment to a desired value, thanks to the closure of a force feedback loop. Impedance (or admittance) control belong to the first category. In the case of impedance control, the controller reacts to the position (or more general: motion) error of the end-effector by demanding the robot to generate a force. In the case of admittance control, the controller reacts to interaction forces by imposing a deviation from the desired motion. A robot manipulator under impedance or admittance control behaves like a mass-spring-damper system with adjustable parameters. Indirect force control schemes do not require, in principle, measurements of contact forces and moments; the resulting impedance or admittance is typically nonlinear and coupled. However, if a force/torque sensor is available, then force measurements can be used in the control scheme to achieve a linear an decoupled behavior [22].

4 Hardware Choices

In order to achieve the specifications set for the grasping strategy in Chapter 2, we propose the following hardware choices informed by the state of the art for robot grasping (see Chapter 3).

4.1 RGB and Depth Cameras for disassembly

The detection tasks include the detection of big parts (LCD screen, printed circuit board) as well as really small parts, such as screws. Because of the great variation in the size of the parts, the Red-Green-Blue (RGB)camera must have a great spatial resolution to be able to detect both big and small objects at a working distance of around 40-70 cm. Regarding the depth sensor, because of the small part size and accuracy required by the end actuators, the depth sensor's accuracy has to be inferior to the size of the smallest object to be detected (around 2 mm for screws in this case). The camera that best fits these criteria is the Ensenso N35 (Figure 4.1). It has the advantage of having a small size, and being easy to integrate on the robot arm. The Depth function is achieved via active stereoscopic technology (projection of an infrared light pattern). This camera works at up to 10 fps, allowing the robot to move at sufficient speed. The data is transmitted through Transmission Control Protocol/Internet Protocol (TCP/IP). Figure 4.2 shows the accuracy vs working distances curves of the N35 camera given by Ensenso, its manufacturer. The uncertainty increases with distance, but always stays under 1 mm. However, the Ensenso N35 does not integrate an RGB sensor; therefore an RGB camera must be added to the system to enable the object detection function. The JAI Go-5000 PGE has a resolution of 2560 x 2048 px (Figure 4.3). With an f = 8mm lens, it fits the requirements for the considered working distance (1 px = 0.5 mm at 80 cm distance).

Figure 4.1: The Ensenso N35 camera. Its enclosure is rated IP67.

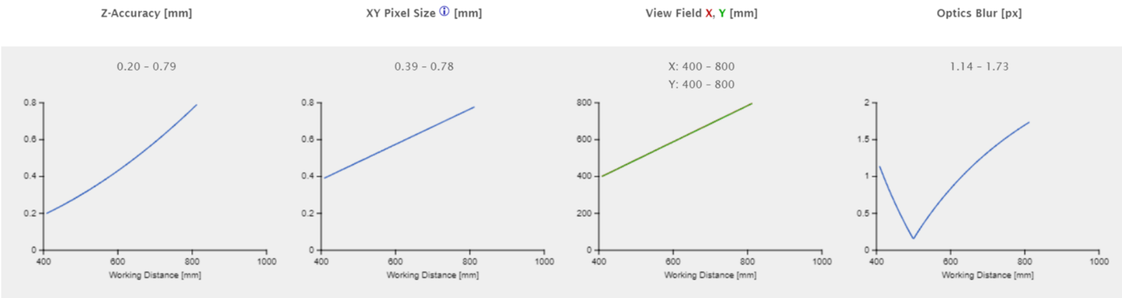


Figure 4.2: Accuracy vs Working distances curves of the N35 camera.

Figure 4.3: The JAI Go 5000PGE camera. Data is transmitted through TCP/IP.

4.2 Robot Gripper

In the scope of the HR-Recycler project, a variety of requirements need to be fulfilled for the gripper. First, with the grasping objects differ in size, so the gripper needs to provide a respective opening width range to increase the flexibility in grasping objects throughout a disassembly process. With the items further alternating in their fragility and weight, the respective force range needs to allow grasping forces below 10 N in order to handle fragile objects while allowing to grasp objects with 100 N force for holding objects steadily in the order of a kilogram mass. Regarding the requirement of force sensitive grasping, the ability to not only control the applied grasping force, but also measure the contact forces between the object and the gripper fingers forms another important requirement. Furthermore, the possibility to integrate the gripper in the overall software architecture by the means of Robot Operating Software (ROS), as outlined in WP 8, needs to be considered alike. After comparing the technical specifications of currently available industrial grippers with respect to the requirements stated above, the servo-electric Weiss WSG50 2-finger parallel gripper was chosen to be used within the device disassembly process of the HR-Recycler project. With a maximum opening range of up to 110 mm, the WSG50 provides a satisfactory range of object sizes to be grasped, while the closing speed limit of 400 mm/s further outperforms alternative industrial grippers. The grasping force range of the gripper is 5-80 N, however, in overdrive mode, up to 120 N can be achieved for a short time period. If no tactile sensing fingers are attached, the applied grip force is approximated via motor current. This method is applied in the majority of industrial grippers nowadays as it allows to detect distinct peaks in a grasping force profile without

the need of adding costly and sometimes also heavy force sensors within the grippers body. Another advantage of the WSG50 gripper lies in the flexibility, which comes within the variety of supported communication interfaces. Manual positioning and grasping control, settings and diagnostics are accessible over a web interface, which also include an interactive scripting environment (using the programming language Lua). Multiple scripts can be saved on the gripper's internal memory card. One of these scripts can be run automatically when the module starts. This allows to move grippers based on a service procedure (cf. Chapter 5.5) and read out sensors. The WSG50 gripper features integrated force, speed and position control where control parameters (proportional, integral and differential gain for the speed controller and proportional gain for the position controller) can be set in the auto start script of the gripper [14]. For the force controller, only the desired grasping force can be set via a service of the ROS driver.
In order to achieve an adaptive force control strategy for the WSG 50 gripper via ROS, the connection to the WSG50 gripper is established with the associated open source driver package, which is based on the original standalone driver for TCP/IP by Weiss Robotics. To identify the delay of the ROS gripper driver pipeline, we ran a custom script that immediately returns the commands sent to the gripper, without performing any gripper movement. With a compiled ROS executable of a SIMULINK model used to set a desired velocity profile, delay times were 0.030 s on average, with a standard deviation of 0.002 s. Nonetheless, the gripper comes with limitations. First, force measurements from the motor current approximation are noisy and naturally uncertain, as outlined in Chapter 3. Second, the inherent service-based grasping procedures can not directly be used for our desired control strategy. A workaround to achieve continuous and time-varying grasping forces is elaborated in Section 5.5.

4.3 Tactile Sensing Fingers

As stated above, the most common way of measuring grasping forces is found in the measurement of the dedicated motor current of the prismatic or revolute joint motors. While this method allows to detect distinct changes in the grasping force profiles, e.g. a contact with an object, the measurement suffers from a distinctly low signal-to-noise ratio and is thus barely suitable for force sensitive manipulation skills, such as grasping. For the specified problem setup (see Chapter 2), relying on motor current alone provides an insurmountable challenge: First, the force estimation is not precise enough to guarantee desired force profiles while gripping, and second, the one-dimensional measurement does not give any information about the alignment of the gripper. Additionally, a force approximation via motor current is not sufficient to deduce pressures applied to the object, as the pressure applied to the object is proportional to the inverse of the surface area in contact. To guarantee sensible pressure profiles in order to not break the object, the applied pressure, and therefore the surface area in contact, needs to be known. While shrinking the gripping fingers to a sharp point-like tip would lead to the same surface area for every grasping position (*all* of the finger in contact all the time), it would lead to excessively large pressures, even for small forces. Yet, precise alignment and sensible grasping forces (or pressures) are exactly what is required for the given use-cases. As a consequence of all the above, HR-Recycler strongly profits from attaching pressure sensors

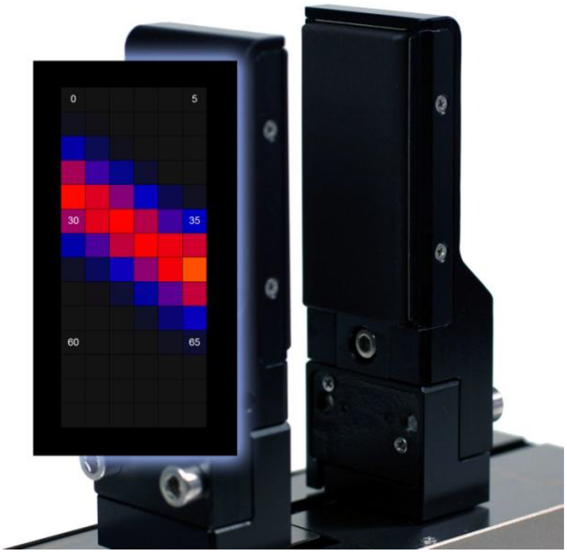


Figure 4.4: The WSG-DSA gripper fingers integrating a tactile sensing matrix.

(or force sensors, since the sensor area is known) to the finger tips that allow an estimation of the force distribution across the contact surface between gripper and object. The WSG-DSA, presented in Figure 4.4 is a gripper finger that integrates a tactile sensing matrix consisting of $6 \times 14 = 84$ sensor cells for a high-resolution pressure profile. The sensor cells use a WTS 0614-34 intelligent tactile transducer. The fingers are directly compatible with the selected parallel finger grippers WSG50 and fit onto the base jaws of the gripper. They directly interface to the gripper controller via the integrated sensor sort inside the base jaws. This enables the use of pressure profile as feedback for gripper control without needing further external components or cables. Thus, the advantage of the proposed robot (COMAU Racer5) used for HR-Recycler disassembly, which has no external cables either, is kept. The sensor matrix is covered by a silicone rubber protector. This provides the advantages of soft contacts discussed in Section 5.5 for precise and reliable force control in the presence of uncertainty. It consists of $6 \times 14 = 84$ sensor cells, each measuring the applied pressure. For prolonged sensor life, two aspects are important. First, shear forces should be avoided as they may directly destroy sensors. This means objects held tightly by the WSG-DSA should not be removed with force. Second, long time static load may deteriorate sensor performance. This means the WSG-DSA should not be used to hold objects, e.g. for another agent (human or robot) unscrewing, for an extended period of time. The sensor data provided by the finger arrays can be read out in a web interface, as shown in Figure 4.5. For integration into our external control strategy, ROS drivers need to be implemented for the finger sensor arrays.

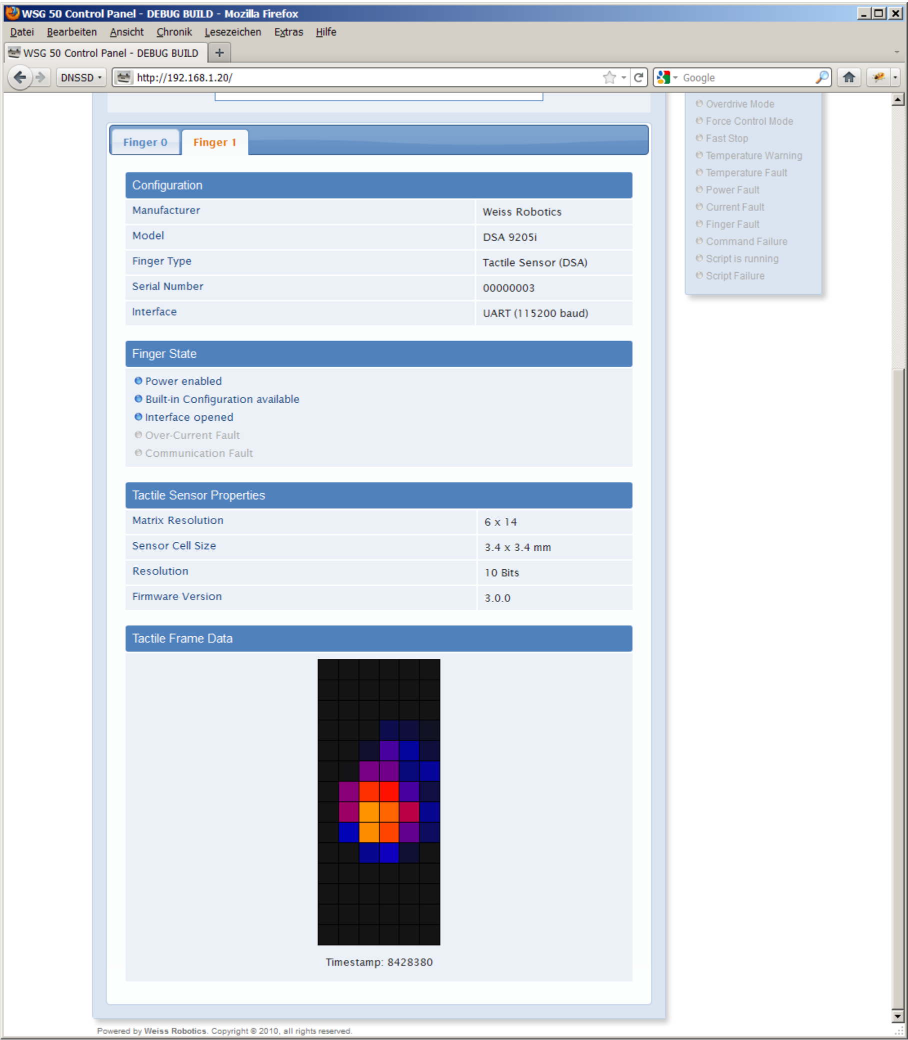


Figure 4.5: The web interface of the WSG-DSA tactile sensing fingers. The sensor matrix is displayed with color-coded pressure values under *Tactile Frame Data*.

5 The Force-Adaptive Grasping Strategy

This chapter presents and discusses proposed initial concepts for the force-adaptive grasping strategy proposed for device dissasembly in HR-Recycler. Much of the advantages and disadvantages of different methods pertaining to robot grasping have already been explained in the previous chapter (Chapter 3). Additionally, the problem setting in which grasping methods will be applied within HR-Recycler has already been defined in Chapter 2. Therefore, some choices can be presented without much further explanation. Figure 5.1 illustrates an overview of aspects influencing grasp hypotheses in HR-Recycler. Based on the two previous chapters, we can identify a grasping pipeline of subtasks required to successfully grasp an object with a robotic gripper. The chapter is split up into sections according to this pipeline, ranging from vision and tactile sensing to grasp synthesis and grasp control.

- First, the object needs to be localized and its shape, dimensions and pose estimated. This is the main task of the computer vision in T6.3 and can be summarized under a *Sensing* step. In addition to visual feedback, sensing will include tactile information later in the process.
- After the object and its features are perceived, a grasping point on the object is selected and a grasp configuration synthesized.
- The desired grasping configuration now needs to be approached by the robot. A motion is planned and tracked by a compliant controller to ensure safety and avoid a hard collision with the object.
- Once the object is reached, the orientation of the gripper may be controlled to better align with the object and reduce errors in the relative position. Using tactile feedback, the precision of this alignment can exceed limitations of uncertain visual information (especially in close range operation of the camera).
- After the gripper is properly aligned, the object will be grasped using a force-adaptive control.

5.1 Sensing

Sensing for object grasping will consist of two main data sources: Vision and Tactile Sensing. Vision-based perception will provide initial information for object location, shape and size, based on which we expect to be able to approach the object and align the gripper within some uncertainty range. Additionally, initial grasping point selection will use vision

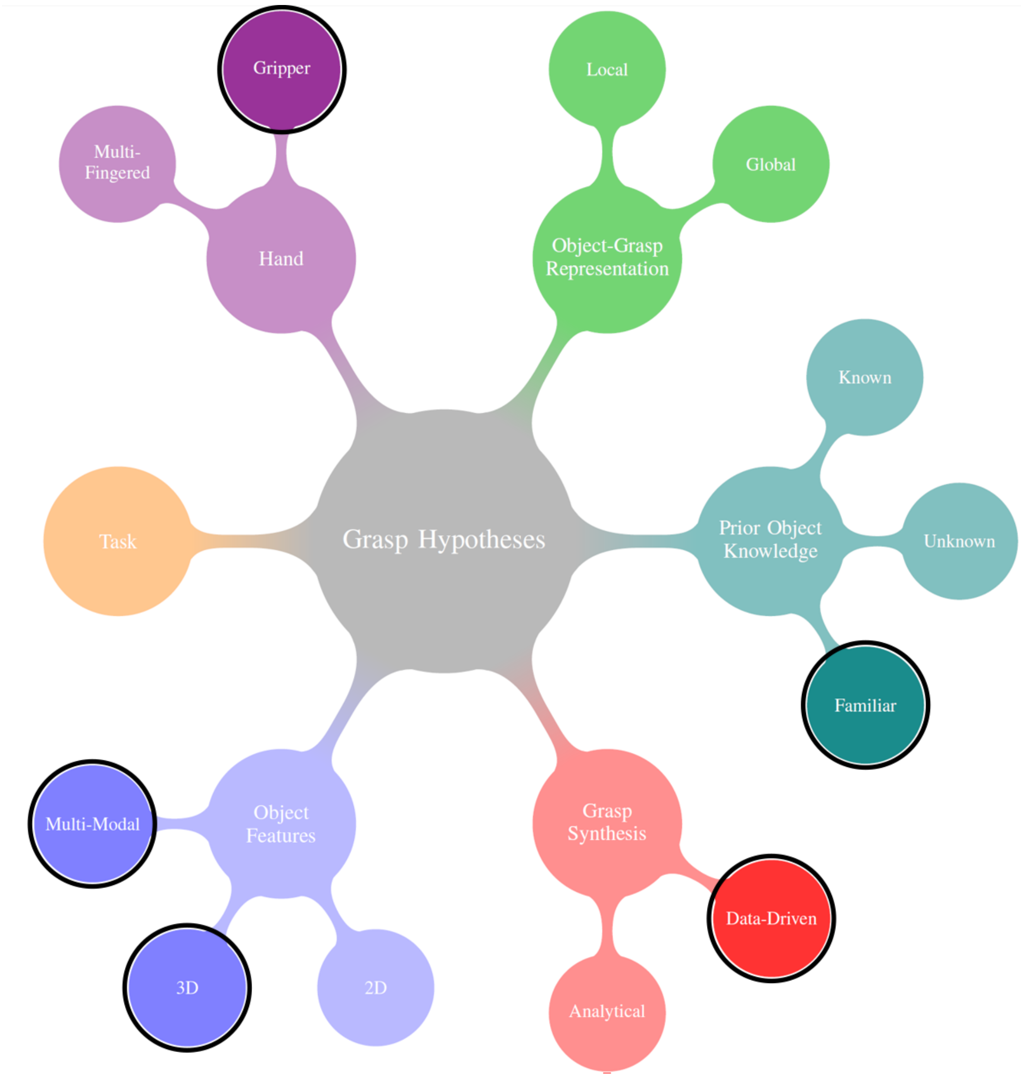


Figure 5.1: Identified aspects that influence the grasp hypotheses, taken from [5], and the choice for HR-Recycler. Objects encountered will be familiar but not identical. We propose a data-driven grasp synthesis due to uncertainty and our strategy will rely on 3D visual as well as tactile sensing.

data in the form of STL (stereolithography) files. Once the gripper is in touch with the object, tactile Sensing will provide additional and precise information about the objects relative position and the grippers alignment with the object surface.

5.1.1 Vision

The vision system will be composed of an RGB and Depth cameras. The cameras must be chosen so that the resolution given both by RGB and Depth sensors is sufficient to give the adequate inputs to successfully achieve the tasks that rely on the vision system. The data given by the depth sensor will be of great use to the sensing function as it will give an initial estimation of the shape and location of the objects in the surroundings of the gripper. The RGB data will be used as input of a neural network specifically trained to recognize the objects on which actions need to be taken (screws, printed circuit boards,...). In conclusion, data from the RGB and Depth sensors is used to obtain the type and location of the objects lying on the workbench, allowing further sensing functions.

5.1.2 Tactile Sensing

A measure of the gripping actuator effort can be obtained by directly measuring the motor current. Typically, this is done by measuring the voltage drop across a sensing resistor installed in series with the motor [22]. However, this actuator effort measure does not easily lend itself to precise gripping force estimates. The reason is, that gears connecting the motor with the gripping finger introduce nonlinear effects (such as blockage) that are hard to accurately model and account for. More promising is a direct measurement of the applied grasping force, using tactile sensing. Since we expect the robot to grasp objects of different shapes, where object shape and position is subject to uncertainty (see Chapter 2 and D6.2), tactile sensing is necessary to robustly perform precise grasps, as explained before in Chapter 3. If we have spatial information of forces applied to the object, we can estimate relative position and angle. To this end we will use two opposing tactile sensing fingers on the gripper. These fingers are each equipped with a matrix of pressure sensor cells, giving further information about the 3D angle (or approach vector) between gripper and object. The 3D angle can then be used as control variable to align the gripping fingers with the desired grasping point and approach vector to reach the desired grasping configuration. Both opposing fingers in parallel to flat surfaces of the object in between them corresponds to a symmetric pressure profile. By orienting the gripper such that the pressure profiles are symmetric, we can align the fingers with the object without relying on the object model.

5.1.3 Multi-modal approach

To arrive at a best possible estimate for relative position and angle of gripper and object, information provided by vision and tactile sensing can be fused. For the initial position and pose estimation of the object, as well as for the initial selected grasping point (see next section), information provided by vision will be used. To refine relative position and align with the pose of the object, tactile sensing will be used. Since visual feedback might be out of its optimal operation range (distance between camera and object too small)

or occluded by the fingers once the gripper is in contact with the object, tactile sensing fingers naturally extend the robots sensing capabilities provided by cameras.

5.2 Grasping point selection

As alluded to in Chapter 3, automatic grasp planning is difficult. There is a huge number of possible grasping points for any given object, and within HR-Recycler, we expect to find significant object variations (for example due to breakage) even within the same device category. To perform this task within HR-Recycler, we therefore propose a data-driven approach using sampled grasp hypotheses evaluated with GraspIt. To improve beyond initial models, learning may improve the grasping metric used (cf. Section 5.6) GraspIt is a tool for grasping research originally developed by the Comlubia University Robotics Group. It includes a grasp planning module, which we plan to use for grasping point selection. The tool supports Soft Finger Contacts, computes numerical grasp quality metrics and allows for visualization with a 3D user interface and virtual environment. GraspIt has shown promising results in finding grapsing points once the exact geometry and material decomposition of an object have been obtained. The tool allows to parse object specification such as the total mass, the center of mass and the intertia matrix to be read in in the form of a default `<xml>` file, while the geometry is read in using the Coin 3D engine, that either reads Virtual Reality Modeling Language (`.wrl`) or Open Inventor (`.iv`) file formats [16]. Using the current estimation of the geometry and dynamic propoerties of the object, graspit can be iteratively called to verify or reject current hypothesis of an object, thus allowing a robot agent to eventually, yet fuly autonomously, successfully grasp the unknown object.

5.3 Motion Planning and Trajectory tracking

The feasible grasp means a goal configuration for the robot. Then, given the initial and final arm configuration, a collision-free path for the arm is searched using some motion planning method. The problem of finding a trajectory from a start to a goal point while avoiding obstacles in the environment, has been tackled mainly using two approaches: sampling based and optimization based motion planning. Sampling based motion planning algorithms build a roadmap of collision-free configurations in the configuration space [8]. These configurations can be later queried for the desired path. This approach is particularly useful for high-dimensional configuration spaces. Optimization based motion planning methods formulate motion planning problem in terms of optimization problem that minimizes a suitable cost function [17]. The cost function is designed to lead to the avoidance of obstacles and smooth trajectories, which can be easily interpretable by a human in the vicinity. Kinematic and dynamic limitations are treated as constraints.

5.4 Alignment Control

As alluded to before, by orienting and positioning the gripper such that the pressure profiles on the tactile sensing fingers are symmetric, we can align the fingers with the object without relying on the object model. This approach is expected to work best for

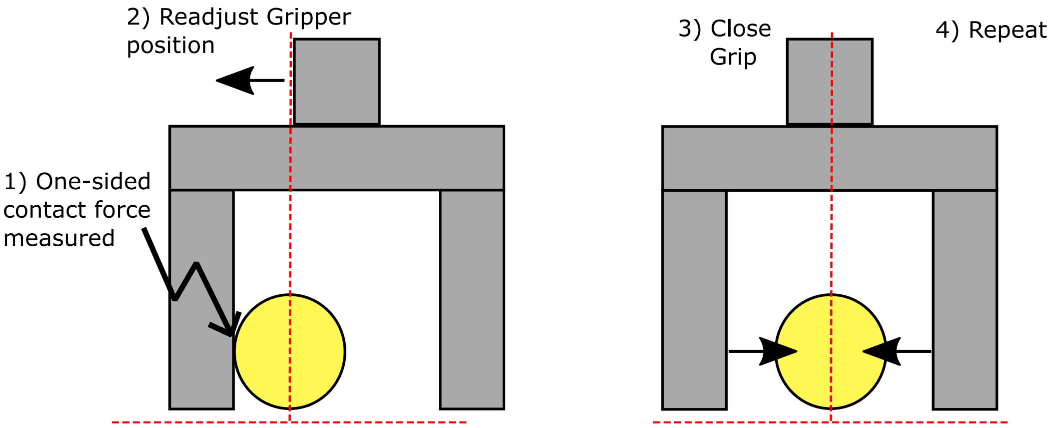


Figure 5.2: View from the side: Adjustment of relative position of the gripper (grey) to the object (yellow) based on haptic feedback. If the gripper were to close its fingers in the initial grasping position (on the left), the left finger would push the object to the right. For an emergency lamp that is fixed in place, this would lead to breakage.

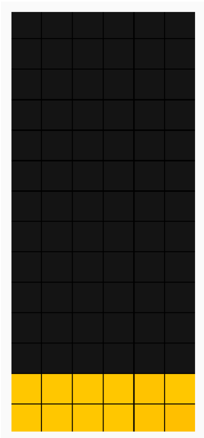
small deviations from symmetric pressure profiles, which makes it a well-fitting extension to a vision-based grasp approach.

Adjusting Gripper Position

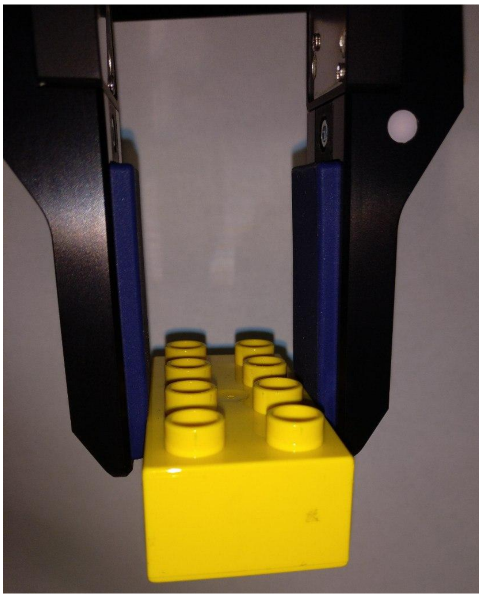
Haptic feedback allows us to reposition the gripper when contact is made with one finger before the other one. This is illustrated in Figure 5.2. For the use case of HR-Recycler, fine adjustments like these are crucial, especially for the emergency lamp, because it is held in place by a fixture. If the gripper closes assymetrically over the lamp, the finger that comes in contact first will exert a big force on the object, ever increasing as the gripper tries to close the fingers further, until the lamp is broken. With force measurements and the grasping force control concept introduced in the next section, we can stop the grasp once the finger that is in contact first exerts too much force. Based on an estimate of the object size, the gripper position will be readjusted and the gripper may close again. This process may repeat for cases where the object dimensions are very uncertain and thus allows for a very careful and precise grasp adjustment for uncertain objects, as is expected in the associated use case in HR-Recycler.

Adjusting Gripper Orientation

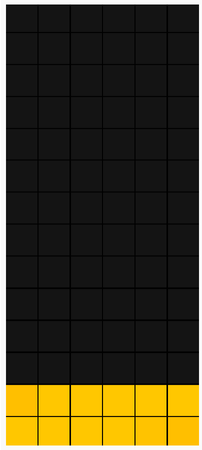
As can be seen in Figure 5.3, symmetric sensing arrays on the fingers correspond to an aligned grasping position (of fingers and object). In Figure 5.4, an exemplary scenario is shown to illustrate skewed sensing arrays corresponding to a misaligned grasp. While the gripper can hold the object in both scenarios, the forces applied to the object are uniformly spread across a large surface for the aligned case, but focused on specific points in the misaligned case, leading to increased pressure and increased probability of breakage. Additionally, for the misaligned case, the main pressure points of the two fingers on the object are not even directly opposite of each other, leading to shear forces and a moment applied to the object. This further increases the probability of breakage, especially



(a) Sensing array of the finger on the left.

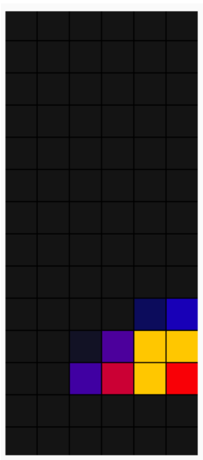


(b) Aligned Grasp of an object.

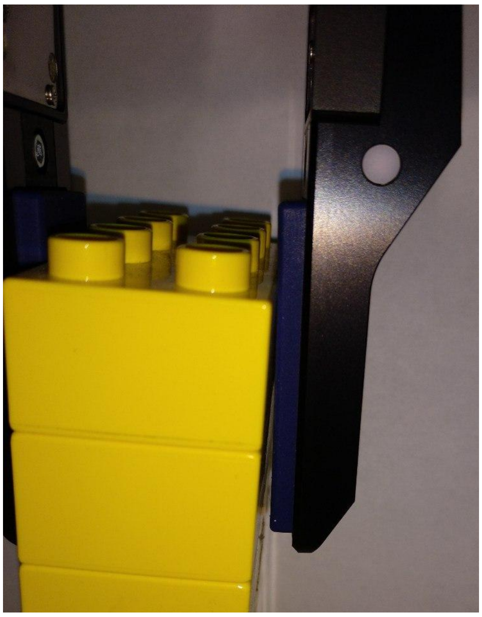


(c) Sensing array of the finger on the right.

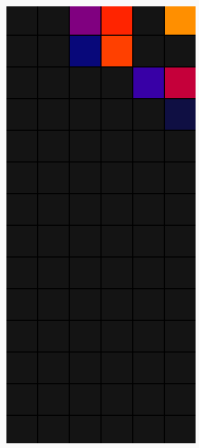
Figure 5.3: Grasping Scenario with tactile sensing fingers for an aligned grasp of an object.



(a) Sensing array of the finger on the left.



(b) Misaligned Grasp of an object. The lego brick only touches the right finger on the upper part and the left finger on the lower part.



(c) Sensing array of the finger on the right.

Figure 5.4: Grasping Scenario with tactile sensing fingers for a misaligned grasp of an object. The two pressure profiles of the fingers are not symmetric.

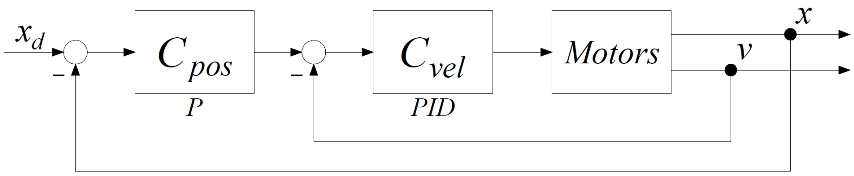
for the use case of fragile but fixed emergency lamps in HR-Recycler, where a wrongly applied moment will break the object. For proper gripper orientation in 3D, uniform and symmetric pressure profiles of the sensing arrays on the fingers are optimal. To that end, we develop a controller, orienting the gripper by controlling the wrist of the robot arm based on the provided pressure (force) feedback. The controller will be based on an impedance/admittance-related strategy and force-adaptivity.

5.5 Grasping Force Control

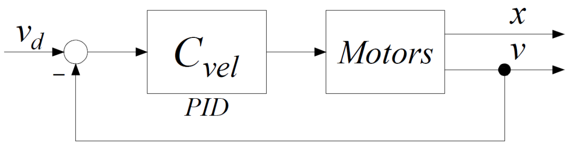
To apply the correct force while gripping an object, industrial grippers typically come with an internal controller that relies on force estimations based on the motor current. However, this estimate is not accurate. Externally, grasping force of a standard industrial gripper can in principle be regulated indirectly by using one of the two possible gripper commands other than grasping force: gripper target position and velocity. In both cases, the same control structure can be used, with the only difference being the controller output u. For the first option, position based force control, we have $u = x_{ctrl}$, where x_{ctrl} is the the desired gripper finger width sent to the gripper. The idea is to set x_{ctrl} smaller than the width of the object to be grasped. This causes the gripper to apply a force on the object. The magnitude of the applied force is proportional to the *clamping range*, the difference between desired and actual gripper finger width, and can therefore be set by the controller indirectly. However, this control strategy comes with a big drawback, namely, its blocking nature: The gripper only accepts a new position command if it has fulfilled the previous one. This leads to step movements and therefore step force profiles. Besides, position based force control requires information about object dimensions (the width of the object to be grasped). The accuracy of position based force control directly depends on the accuracy of this information. For HR-Recycler, we cannot expect to always a priori know object dimensions with great accuracy. Accurate identification of object dimensions might be costly and time-consuming. The second option is force control based on velocity, where we have $u = v_{ctrl}$ and $v_{ctrl} = \dot{x}_{ctrl}$. If a speed command is sent to the gripper when it cannot close any further, it will try anyway and apply a force to the object between its fingers. This force is proportional to the set closing speed and therefore does not rely on object dimensions. Additionally, for the investigated industrial gripper (Schunk WSG50) the closing speed can be set with higher resolution than the gripper target position. This allows for a smoother grip force control. To evaluate velocity based force control, we implemented the strategy on the common industrial SCHUNK WSG50 2-finger parallel gripper (see Chapter 4 for technical information and design decisions) mounted on a KUKA Light-Weight Robot (LWR) 4+. In the next section, this external force control approach will be presented in more detail. Results will serve as a comparison for future approaches that use direct force measurements from tactile sensing fingers, instead of estimates based on desired gripper velocities.

5.5.1 External indirect force control

The WSG50 gripper features integrated force, speed and position control. Position and velocity control presumably have the structure shown in Figure 5.5. Control parameters (proportional, integral and differential gain for the speed controller and proportional gain



(a) Position control



(b) Velocity control

Figure 5.5: Gripper internal position and velocity control structure

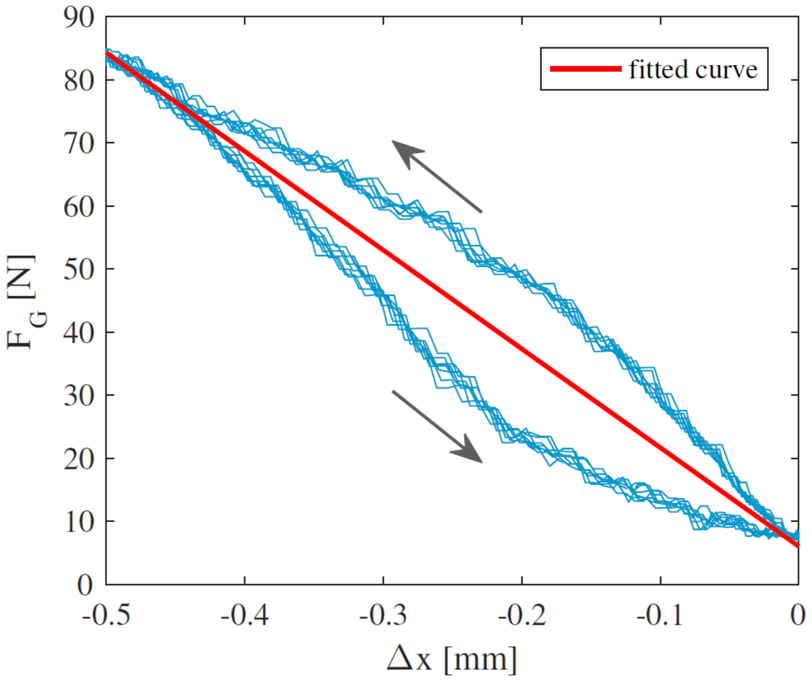
for the position controller) can be set manually. For all the following experiments, the desired grip force for the internal controller is set to 110 N, right after the start of the gripper driver. This allows utilization of the nominal force margin (5 to 80 N) of the WSG50 gripper as external force control objective, while taking eventual overshoots into account. Otherwise, if the gripper reaches the set grip force, the internal force controller would interfere and cause unpredictable behavior, hindering our external controller from carrying out its task. Furthermore, the integral gain of the gripper's internal speed controller is set to zero to allow repeatable external force control. We use MATLAB and SIMULINK to generate desired inputs and implement analysis tools and controllers. The velocity-based force controller relies on a gripper model that will be identified in the following section.

Relation Between Grip Force and Clamping Range

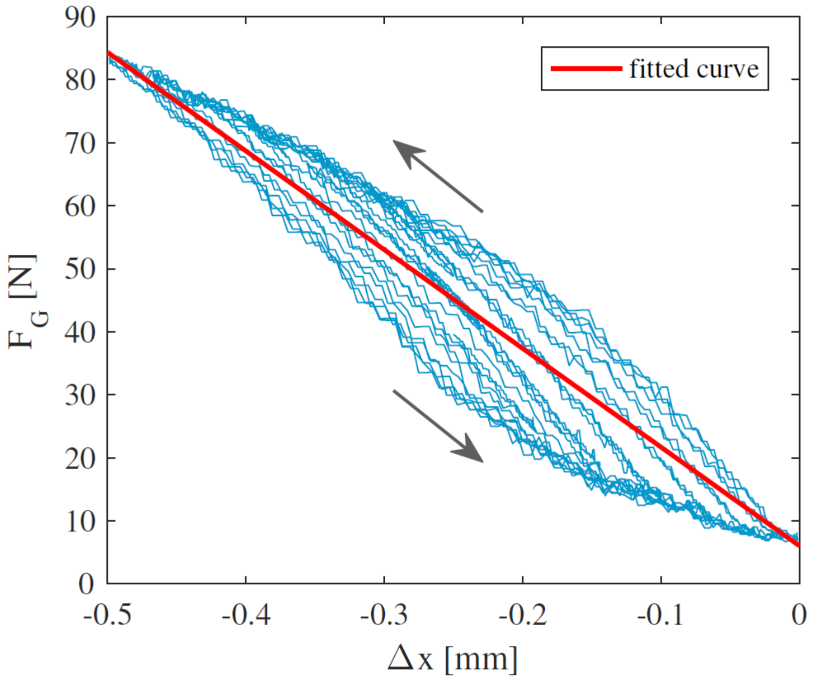
In this section, the dependence of grip force on clamping range is modeled. A sinusoidal signal is set as desired gripping finger width (position) and the resulting force exerted on the object is measured. Curve fitting results in a model that predicts the exerted force when giving the difference in desired and actual gripping position. A comparison of resulting measurements and linear model predictions is shown in Figure 5.6.

Controller Design

The resultant closed-loop system is shown in Figure 5.7. G_{id} is a fourth-order linear discrete-time model that represents the lag between actual gripper velocity and desired gripper velocity. We identified G_{id} using random step inputs for the desired gripper velocity. A comparison between predicted and measured velocities can be seen in Figure 5.8.
The obtained controller is implemented and evaluated on the real gripper. For that, a 30 Hz SIMULINK model with the designed proportional controller is created, which sends speed commands receives grip force measurements via the ROS driver (also running at 30 Hz). We test the grasping strategy for different objects. Results (see Figure 5.9) show that



(a) Simple sinusoidal position profile from Figure 4.4(b)



(b) Multiplexed sinusoidal position profile

Figure 5.6: Grip force as a function of clamping range

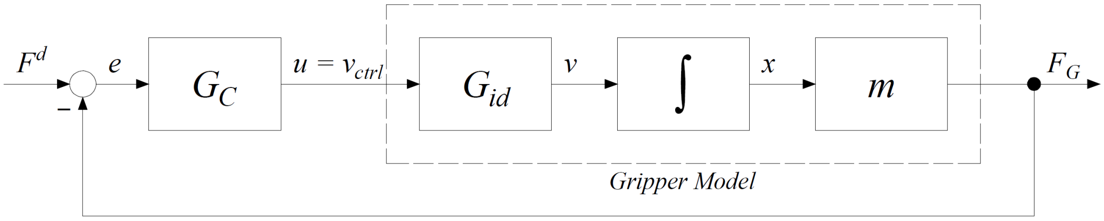


Figure 5.7: Control loop with gripper model for controller design

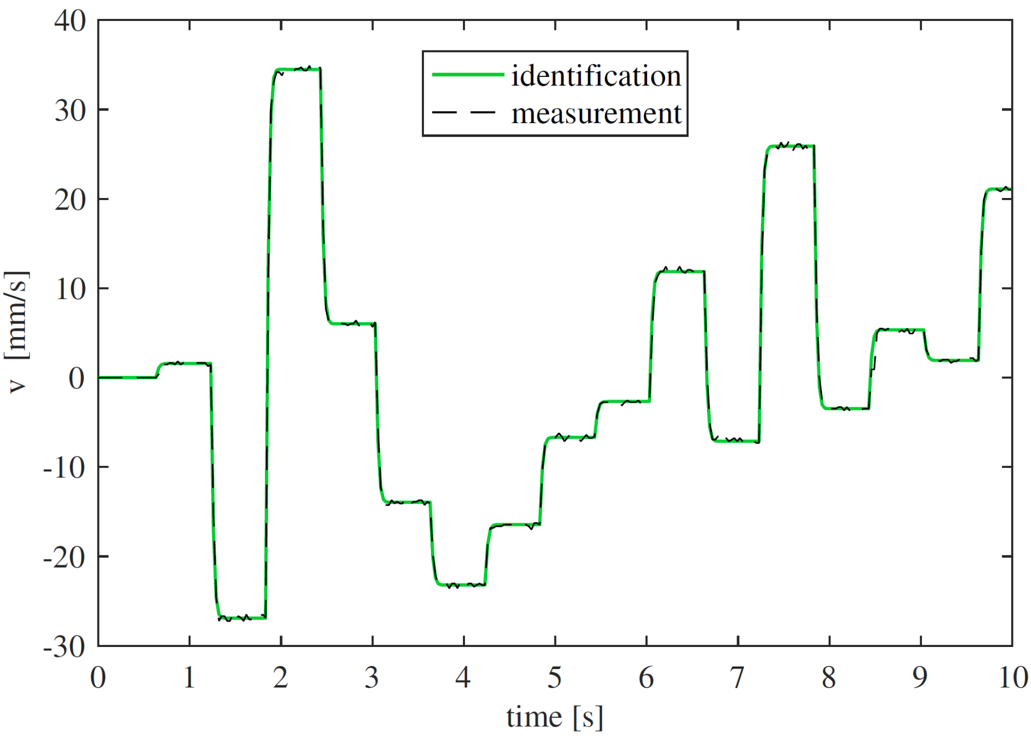
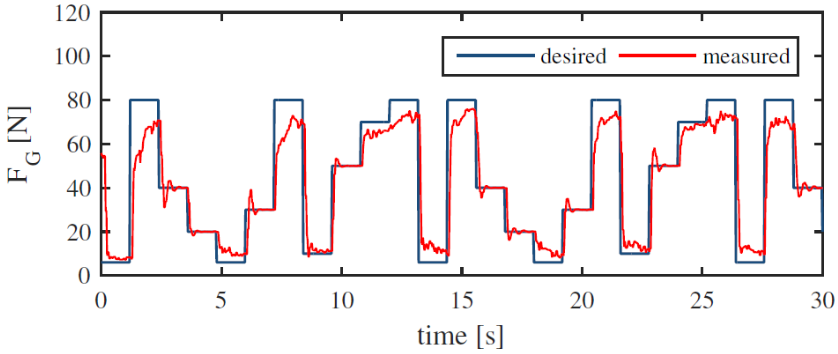
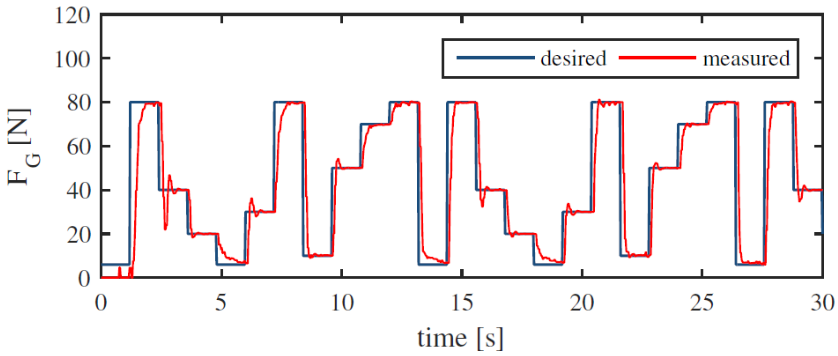


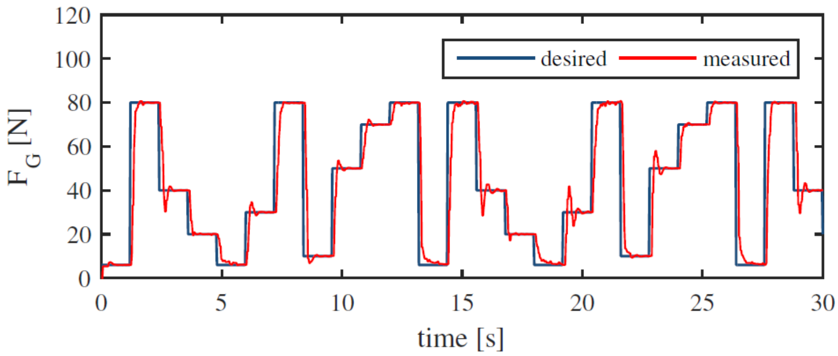
Figure 5.8: Identified gripper velocity model.



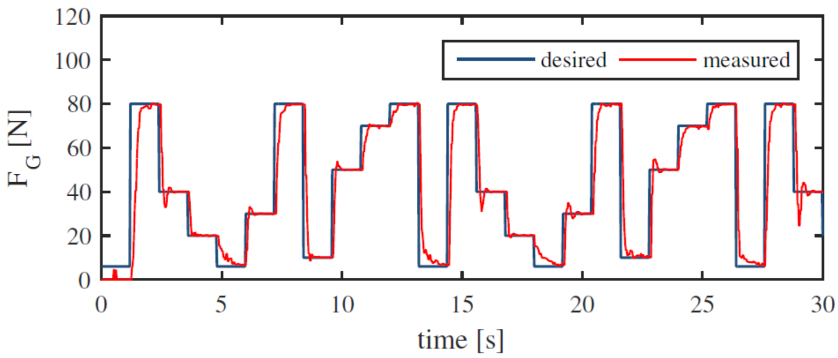
(a) Spray bottle



(b) LEGO brick



(c) Wood part of hammer



(d) Metal part of hammer

Figure 5.9: Grasping Force for different materials.

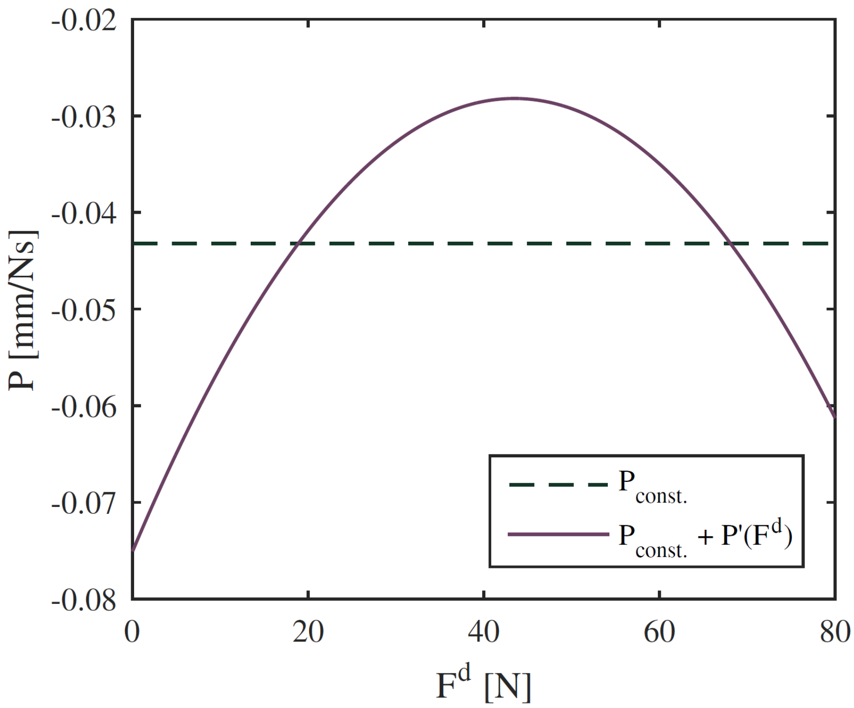


Figure 5.10: Constant (dashed) versus Force dependent (purple) gain function.

force control with the proposed strategy is feasible. However, since our gripper model is greatly simplified, performance is not always satisfactory. Depending on the applied grip force, the gripper shows differences in attenuation. For especially low and high grip forces, performance seems to be better than for medium grip forces. We identify this to be due to the higher modeling error occurring for the gripper model in the medium force range, as can be seen in Figure 5.6.

5.5.2 Force-Adaptivity

To improve performance for high and low grip forces and increase robustness in the medium force range, the proportional gain of the controller can be adapted depending on the magnitude of the desired grip force. Attenuation (corresponding to larger controller gains in absolute value) needs to be smaller for especially high and low forces, and larger for medium forces. To achieve this, we add to the constant gain a quadratic gain law that takes as input the desired force shifted by the mean of the force range. If the desired force is close to the mean (i.e. medium force), the resulting gain is small, if the desired force is far from the mean, the resulting gain is large. This adaptive gain law is illustrated in Figure 5.10. For comparison, the previous constant gain is shown. Figure 5.11 shows experimental results with the force-dependent gain. Objects to be grasped and desired force profiles remained the same as before (cf. Figure 5.9). Comparing the two experimental results, we can see that the adaptive force controller leads to a smaller overshoot for medium grip forces, while it leads to slightly larger overshoots for high grip forces. This indicates that

we can tune the adaptive gain law to adjust overshoots for different force ranges. Additionally, the adaptive force controller leads to shorter rise times (the desired force is reached faster). This is especially the case for harder objects (cases (b)-(d) in Figures 5.11 and 5.9). For the soft spray bottle (case (a) in Figures 5.11 and 5.9), the two controllers lead to similar grasps, as grip forces can only rise slowly while the object is being deformed. To examine the possibility of further grip force control quality improvements, a Smith Predictor is added to the control loop, as shown in Figure 5.12.

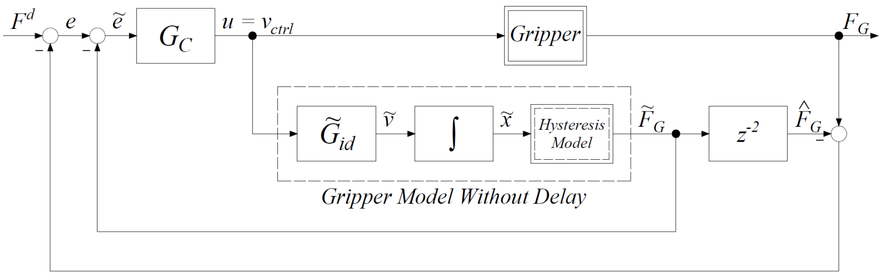
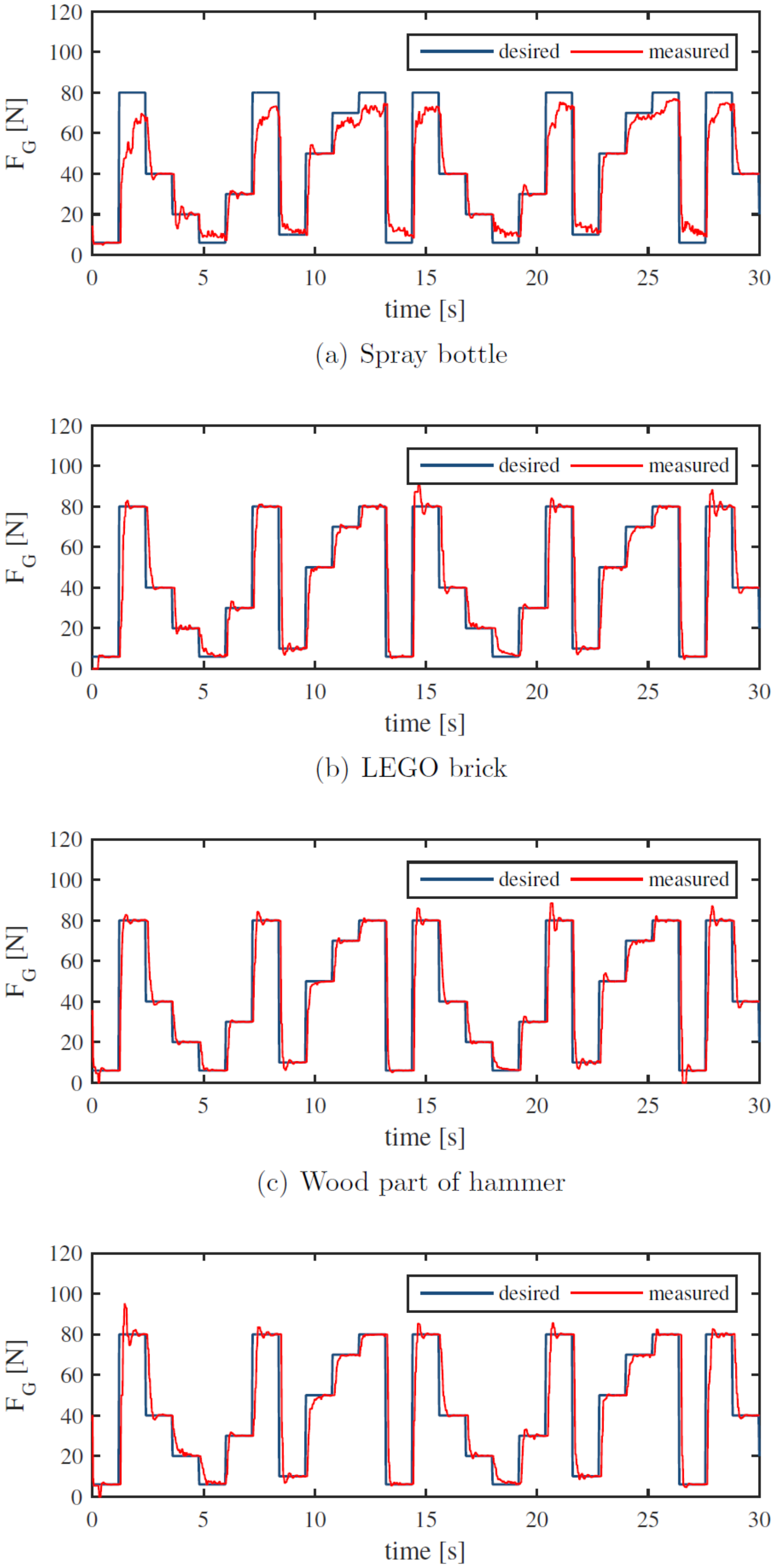


Figure 5.12: Enhanced control scheme with Smith Predictor.

A Smith Predictor is a (feed-forward) predictive controller designed for dead time handling in feedback control systems. It uses a different model for the gripper. Namely, the dead time of the gripper, the time it takes for the gripper to react to a received desired velocity by applying a force, is assumed to be zero. Using the delay-free prediction, the feedback controller can react to the impact of changes in the control signal before its effects are visible in the plant output. This allows for a more aggressive controller tuning based on the system without dead time. To prevent drifting and to reject external disturbances, the Smith Predictor also compares the actual plant output with a time-delayed version of its prediction. One drawback is, that the Smith Predictor introduces a certain dependence on the correctness of the system model, since it so heavily relies on direct predictions. Unlike feedback based on actual measurements (presumably very close the the real states of the plant), feedback based on predictions lacks robustness. In the following, the enhanced control scheme is implemented as SIMULINK model and its performance is compared against control without Smith Predictor for different gripper finger setups: with and without soft damping pads.

5.5.3 Soft fingers

Soft damping pads on the fingers bring several advantages: First, the pads can deform during the grip and adjust to non-flat object surfaces, leading to increased contact areas and a tighter and more stable grip. Second, the elasticity of the damping pads leads to higher friction between object and fingers, which prevents slip and allows for the grasping of heavier objects. Third, as could be seen in the grasping of the soft spray bottle, forces build up slower when soft damping pads are introduced between original gripper and object. This allows for fast closing speeds of the gripper without excessive force overshoot. Without damping pads, closing speeds would need to be greatly limited for hard objects, since large forces $F = ma$ are built up nearly instantly when the gripper



(a) Spray bottle

(b) LEGO brick

(c) Wood part of hammer

(d) Metal part of hammer

Figure 5.11: Grasping Force for different materials with force-adaptive controller.

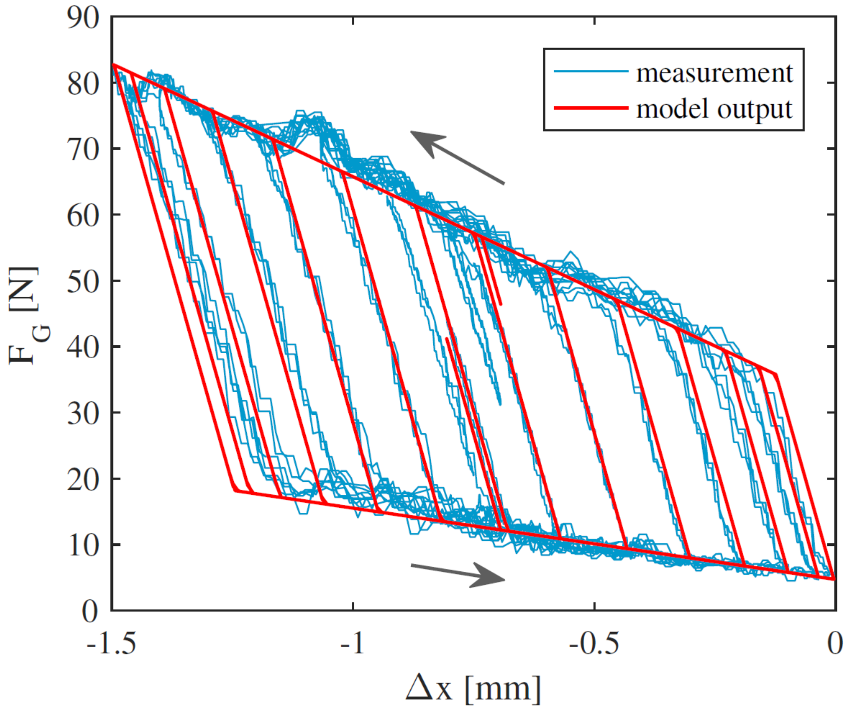


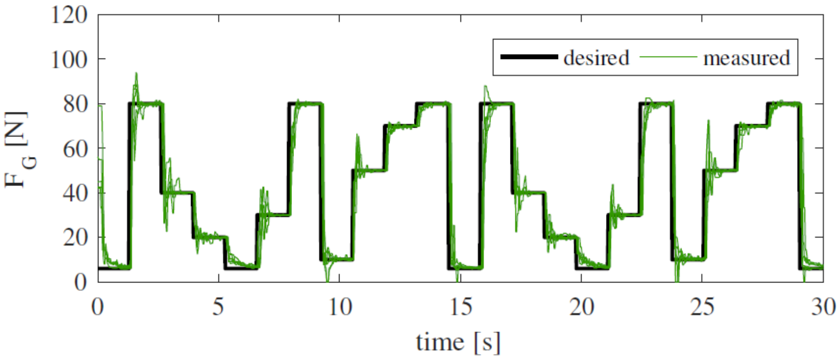
Figure 5.13: Grip force versus clamping range with soft fingers.

comes to an immediate stop (high deceleration a). To neither break the object or gripper, nor trigger gripper-internal safety control procedures, the force controller needs to have sufficient time to react to the build-up of forces. However, the introduced elasticity of the fingers changes - and significantly complicates - the relationship between grip force and clamping range, rendering our simple gripper model ineffective. Figure 5.13 show this relationship for a gripper with damping pads attached on both fingers (blue curves). Function slopes within the hysteresis deviate strongly from slopes in saturation. Thus, the linear model used above (see Figure 5.6) is no longer sufficient. We created a more complex model in SIMULINK, for which outputs are displayed in red in Figure 5.13.

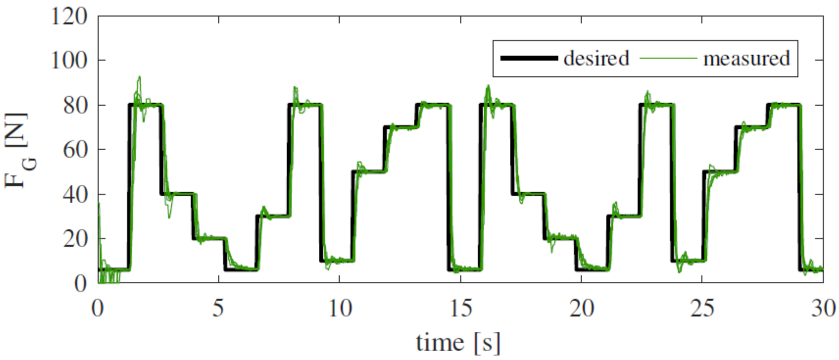
Hard vs. Soft Grippers, with vs. without Smith Predictor?

In Figure 5.14, the results of control with Smith Predictor are compared to the results without it. For each configuration, 5 measurements were conducted and plotted in the same figure. It is clearly visible that control quality with the Smith Predictor is worse than without it, independent of finger variant. With the Smith Predictor, overshoots are very high for medium force values and generally vary from measurement to measurement, resulting in unpredictable behavior of the gripper. This failure of the Smith Predictor can be attributed to the relatively inaccurate models for the inherent hysteresis. Feedback control with a Smith Predictor is in general not typically robust to modeling errors in the plant (here, the grip force to clamping range relationship). As can be seen in Figure 5.13, an even more precise model of the gripper would need to be very complex. As the gripper will be just one component of the robot control strategy for device disassembly in HR-

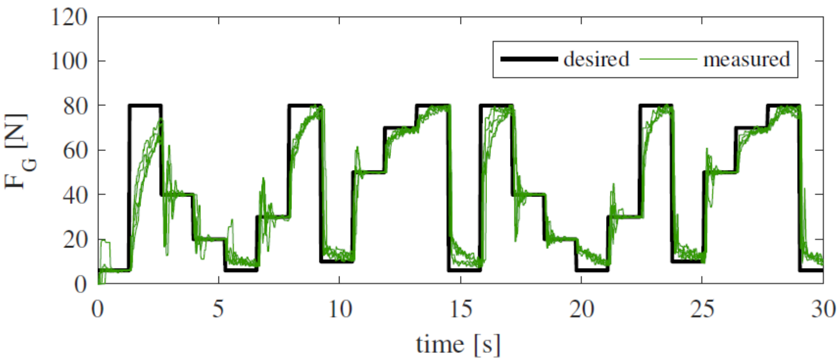
Recycler, we propose at this point to keep the control scheme as simple as possible and, for now, put the Smith Predictor on the shelf.



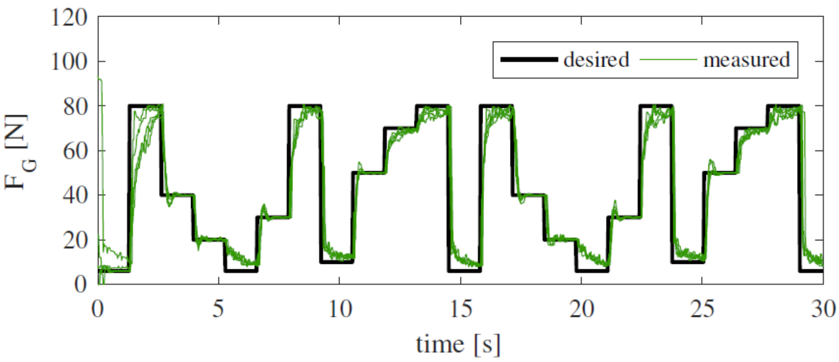
(a) Simple fingers, with Smith Predictor



(b) Simple fingers, without Smith Predictor



(c) Fingers with damper pads, with Smith Predictor



(d) Fingers with damper pads, without Smith Predictor

Figure 5.14: Comparison between control with and without Smith Predictor.

5.6 Cerebellum-based Module

The Problem of Robustness in Predictive Control

Increasing the tolerance to external disturbances makes a robot capable of coping with uncertainty in the world. Normally, this is addressed by means of high-frequency feedback controllers. An ideal feedback controller should be able to compensate for any disturbance magnitude in minimal time and with minimal effort, that is, it should be fast and efficient. However, due to the physical limitations of real systems and the intrinsic uncertainty of dynamic environments, trade-offs and limits in the desired performance should be considered. For example, short control time implies a higher actuation cost, whereas higher resistance to external disturbances leads to rigidity of the actuators and may often cause instability. Some of these trade-offs can be partly overcome by coupling a feedforward controller to the feedback loop. That is, while the closed-loop dynamics of the feedback allows a robot to react appropriately to an unpredictable disturbance and stabilize, a feedforward, or open-loop, controller can make use of signals that precede the disturbance and issue corrective responses in anticipation. Hence, by correctly anticipating the disturbances, the feedforward controller can minimize errors in advance and partly offload the feedback controller, resulting in increased robustness while maintaining a lower actuation cost. However, after acquiring prediction-based anticipatory responses, the control system faces the risk of over- or under-anticipating when events unfold in an unexpected way (i.e., exceptions). Therefore, reliable mechanisms for fast online corrections are also needed to re-stabilize the robot after the self-generated perturbations caused by wrong predictions. This type of robust predictive control is one of the features that best characterizes animals, including humans. The cerebellum is the key brain structure underlying anticipatory (i.e., prediction-based) motor learning. For that reason, computational models of the cerebellum have been successfully implemented to control robots, e.g., learning to compensate for sensory latencies and signal delays, effectively working as a Smith predictor. However, these models do not include any online corrective mechanism for compensating the behavioral effects of wrong predictions, rendering them incapable of handling exceptions correctly. Therefore, a novel solution was proposed as a modeling effort based on the latest advances in cerebellar neurophysiology and neuroanatomy. Specifically, it was hypothesized that flexible and robust anticipatory actions could be achieved by the brain through a cascade of purely sensory predictions that drive the motor system, reflecting the causal sequence of the perceptual events preceding an error. The formulation of this hypothesis was instantiated as a novel control architecture named Hierarchical Sensory Predictive Control (HSPC)and was successfully validated against the standard cerebellar model in simulation experiments [15].

HSPC Cerebellar Model - Technical Implementation

HSPC is a control architecture based on feedforward modules, which mimic the information processing that takes place in cerebellar microcircuits. A feedforward module expands and combines the input signals into a large set of alpha-like temporal bases functions. The bases of the new sparse representation, which implicitly performs sensor fusion, are in turn linearly combined to build an output signal. Finally, the gains (or weights) determining the shape of the output signal are continuously adapted based on real-time error signals, using

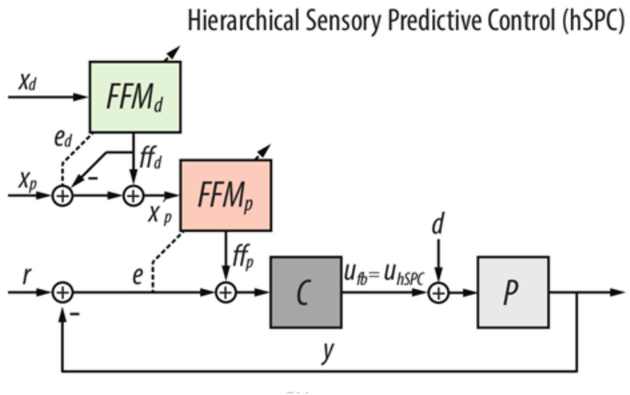


Figure 5.15: The Hierarchical Sensory Predictive Control (HSPC) architecture. FFM = feedforward module; Xi = input i; C = feedback controller; P = plant; d = disturbance; e = error; r = reference; ffi = feedforward output of i module; u = control signal.

a Least Mean Squares (LMS) rule extended with eligibility traces, to account for signal delays [12]. Therefore, a reservoir of complex spatio-temporal representations of the input sources is used to minimize incoming error signals, thus serving as an adaptive filter. Then, in essence, the HSPC architecture consists of a set of hierarchically connected feedforward cerebellar modules that precede the feedback loop (see Figure 5.15). Modules at higher layers predict sensory activity in lower layers, while the last layer sends "counterfactual" errors to the feedback controller so as to minimize predefined sensory performance errors. Hence, a forward model of the environment is subsequently used by an inverse model of the whole closed-loop system in order to minimize both prediction and performance errors, effectively driving robust anticipatory behaviors.
Our model has been shown to handle exceptions well [15, 1]. That is, after learning occurs, unexpected events are successfully "absorbed" by the system by means of online error corrections downstream through the hierarchy. Moreover, error landscapes determined by testing over large proportions of the state-space prove that HSPC generalizes better to unseen environments, unlike the standard cerebellar model, Feedback Error Learning (FEL), which overfits its gains to the training conditions (see Figure 5.16).

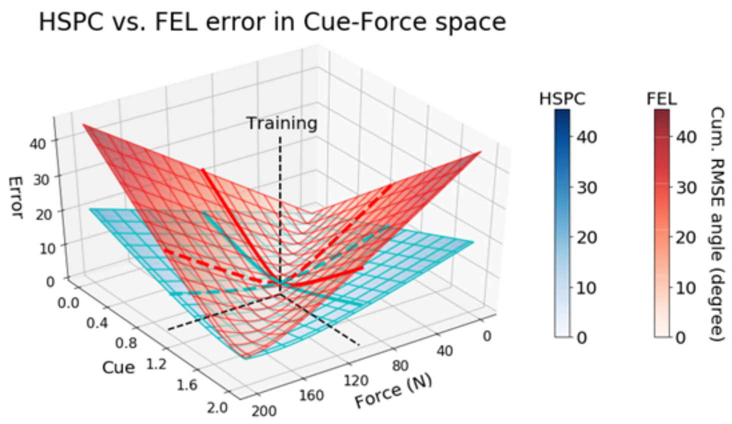


Figure 5.16: Error landscapes for HSPC and the standard cerebellar model FEL. The landscape is determined by the possible combinations of a contextual predictive cue and the force of the plant disturbance. It is shown than the generalization gradient for HSPC is better (i.e. flatter) than for FEL.

Hence, our model, HSPC, that is based on (1) a layered hierarchical structure, (2) error-based learning, (3) online error corrections, and (4) feedforward-feedback coupling, has been shown to advance towards the robust adaptive control capabilities that characterize human motor learning.

Cerebellar Module for Flexible Adaptive Grasping

Capitalizing on the robust adaptive control provided by HSPC, our cerebellar module will provide a predictive model that uses contextual sensory signals to predict forthcoming performance errors and cancels them out in advance using anticipatory control. Specifically, entire force profiles, current grasping points, haptic feedback and visual information will be used to predict and correct for predefined grasping-related errors (e.g., slip of objects) by sending "counterfactual" error signals to the feedback controllers via an interface system. Therefore, specific error profiles will be defined according to a preliminary study tailored at identifying the specific grasping needs and sources of errors for each of the use cases.

6 Conclusion and Outlook

Given the hardware choices presented in Chapter 4 and the proposed grasping strategy presented in Chapter 5, the following section will briefly reexamine how the use case requirements developed in chapter 2 are met. This chapter closes with a short overview of the future work for T6.3.

Conclusion

In Chapter 2, we identified several requirements for the force-adaptive grasping strategy in HR-Recycler. First, the gripper needs to be precisely aligned with the object before the grasp. Second, the grasping force needs to be bound to a desired range. Third, no forces or moments should be applied to the object besides the controlled grasping force in the aligned state. Fourth, the solution should be generalizable and not consist of very specific hardware and method choices that could only apply to a single object. To achieve these specifications, our proposed grasping strategy follows several steps. First, it relies on vision (and associated object localization and pose estimation) to place the gripper around the object and make initial contact at a desirable grasping point. Second, it relies on tactile sensing to readjust the gripper position and orientation without exerting significant forces or moments on the object. Third, it relies on force control to keep the applied grasping pressure in a desired range. Fourth, the proposed feedback control strategies and cerebellum-based feedforward modules, to make the robotic solution generalizable to different grasping tasks. In particular, the precise alignment of the gripper works for any object that fits in between the two gripping fingers and does not rely on precise object position and pose estimates from the vision system.

Outlook

Following this deliverable, work on T6.3 (complete on M26) will continue in several directions. First of all, the concepts described in Chapter 5 will be fully implemented and evaluated. Second, an integration of the cerebellum-based feed-forward module will be developed. Especially crucial here is the choice of error signals that enable learning of the feed-forward module. Furthermore, the vision module needs to be incorporated into the hardware approach, especially for testing related to grasping point selection based on visual information.

Bibliography

[1] A.F. Amil, G. Maffei, J.Y. Puigbo, X. Arsiwalla, and P. Verschure. *Robust Postural Stabilization with a Biomimetic Hierarchical Control Architecture.* Biomimetic and Biohybrid Systems, Springer, 2019.

[2] R. Balasubramanian, L. Xu, P.D. Brook, J.R. Smith, and Y. Matsuoka. *Physical human interactive guidance: Identifying grasping principles from human-planned grasps.* IEEE International Conference on Robotics and Automation (ICRA), 2012.

[3] A. Bicchi and V. Kumar. *Robotic grasping and contact.* IEEE International Conference on Robotics and Automation (ICRA), 2000.

[4] A. Billard and D. Kragic. *Trends and challenges in robot manipulation.* Science, 2019.

[5] J. Bohg, A. Morales, T. Asfour, and D. Kragic. *Data-Driven Grasp Synthesis - A Survey.* Transactions on Robotics, 2016.

[6] H. Dang and P.K. Allen. *Semantic grasping: Planning robotic grasps functionally suitable for an object manipulation task.* IEEE International Conference on Intelligent Robots and Systems (IROS), 2012.

[7] H. Dang and P.K. Allen. *Stable grasping under pose uncertainty using tactile feedback.* Autonomous Robots, Vol. 36, Springer, 2014.

[8] M. Elbanhawi and M. Simic. Sampling-based robot motion planning: A review. *IEEE Access*, 2:56–77, 2014.

[9] OpenAI et. al. *Solving Rubik's Cube with a Robot Hand.* arXiv preprint, 2019.

[10] J. Felip and A. Morales. *Robust sensor-based grasp primitive for a three-finger robot hand.* IEEE International Conference on Intelligent Robots and Systems (IROS), 2009.

[11] C. Ferrari and J. Canny. *Planning optimal grasps.* International Conference on Robotics and Automation, 1992.

[12] I. Herreros, X. Arsiwalla, and P. Verschure. *A forward model at Purkinje cell synapses facilitates cerebellar anticipatory control.* Proceedings in Advances in Neural information Processing Systems, 2016.

[13] K. Hsiao, S. Chittta, M. Ciocarlie, and E.G. Jones. *Contact-Reactive Grasping of Objects with Partial Shape Information.* IEEE International Conference on Intelligent Robots and Systems (IROS), 2010.

[14] Weiss Robotics GmbH & Co. KG. *WSG Series of Intelligent Servo-Electric Grippers - Scripting Reference Manual.* 2014.

[15] G. Maffei, I. Herreros, M. Sanchez-Fibla, K.J. Friston, and P. Verschure. *The perceptual shaping of anticipatory actions.* Proceedings of the Royal Society B: Biological Sciences, 2017.

[16] A.T. Miller and P.K. Allen. *GraspIt! a versatile simulator for robotic grasping.* IEEE Robotics & Automation Magazine, 2004.

[17] N. Ratliff, M. Zucker, J. A. Bagnell, and S. Srinivasa. Chomp: Gradient optimization techniques for efficient motion planning. In *2009 IEEE International Conference on Robotics and Automation*, pages 489–494, May 2009.

[18] M.A. Roa and R. Suarez. *Grasp quality measures: review and performance.* Autonomous Robots, Springer, 2015.

[19] A. Sahbani, S. El-Khoury, and P. Bidaud. *An overview of 3d object grasp synthesis algorithms.* Robot. Auton. Syst., 2012.

[20] J. Sanchez, J.A.C. Ramon, B.-C. Bouzgarrou, and Y. Mezouar. *Robotic Manipulation and Sensing of Deformable Objectst in Domestic and Industrial Applications: A Survey.* The International Journal of Robotics Research, SAGE Publications, 2018.

[21] K. Shimoga. *Robot Grasp Synthesis Algorithms: A Survey.* International Journal of Robotic Research, 1996.

[22] B. Siciliano and O. Khatib. *Handbook of Robotics (2nd Ed.).* Springer, 2016.

[23] D. Song, C.H. Ek, K. HÃ¼bner, and D. Kraigc. *Multivariate discretization for bayesian network scruture learning in robot grasping.* IEEE International Conference on Robotics and Automation (ICRA), 2011.

[24] W. Wang, Robert N. K. Loh, and Edward Y. Gu. *Passive compliance versus active compliance in robot-based automated assembly systems.* Industrial Robot, 1998.

[25] J. Weisz and P.K. Allen. *Pose Error Robust Grasping from Contact Wrench Space Metrics.* IEEE International Conference on Robotics and Automation (ICRA), 2012.