Robot Competitions Kick Innovation
in Cognitive Systems and Robotics
FP7-ICT-601012

RoCKIn@Work

– Innovation in Industrial Mobile Manipulation –
Competition Design, Rule Book, and Scenario Construction

Deliverable: D-2.1.4
Due Date: September 30, 2013
Delivery Date: December 6, 2013
Revision: 0.7

Editors: Jakob Berghofer, Rhama Dwiputra, Gerhard Kraetzschmar
Contributors: Aamir Ahmad, Francesco Amigoni, Iman Awaad, Jakob Berghofer, Rainer Bischoff, Andrea Bonarini, Rhama Dwiputra, Frederik Hegger, Nico Hochgeschwender, Luca Iocchi, Gerhard Kraetzschmar, Pedro Lima, Matteo Matteucci, Daniele Nardi, Sven Schneider

TÉCNICO LISBOA
SAPIENZA UNIVERSITÀ DI ROMA
Hochschule Bonn-Rhein-Sieg
KUKA
POLITECNICO DI MILANO
INNOCENTIVE

Abstract

RoCKIn is a EU-funded project aiming to foster scientific progress and innovation in cognitive systems and robotics through the design and implementation of competitions. An additional objective of RoCKIn is to increase public awareness of the current state-of-the-art in robotics in Europe and to demonstrate the innovation potential of robotics applications for solving societal challenges and improving the competitiveness of Europe in the global markets.

In order to achieve these objectives, RoCKIn develops two competitions, one for domestic service robots (RoCKIn@Home) and one for industrial robots in factories (RoCKIn-@Work). These competitions are designed around challenges that are based on easy-to-communicate and convincing user stories, which catch the interest of both the general public and the scientifc community. The latter is in particular interested in solving open scientific challenges and to thoroughly assess, compare, and evaluate the developed approaches with competing ones. To allow this to happen, the competitions are designed to meet the requirements of benchmarking procedures and good experimental methods. The integration of benchmarking technology with the competition concept is one of the main objectives of RoCKIn.

This document describes the first version of the RoCKIn@Work competition, which will be held for the first time in 2014. The first chapter of the document gives a brief overview, outlining the purpose and objective of the competition, the methodological approach taken by the RoCKIn project, the user story upon which the competition is based, the structure and organization of the competition, and the commonalities and differences with the RoboCup@Work competition, which served as inspiration for RoCKIn@Work. The second chapter provides details on the user story and analyzes the scientific and technical challenges it poses. Consecutive chapters detail the competition scenario, the competition design, and the organization of the competition. The appendices contain information on a library of functionalities, which we believe are needed, or at least useful, for building competition entries, details on the scenario construction, and a detailed account of the benchmarking infrastructure needed — and provided by RoCKIn.

Contents

Chapter 1

Brief Overview on RoCKIn@Work

This chapter gives a brief overview on RoCKIn@Work. It is a quick guide to anyone who wants to learn what RoCKIn@Work is about, and it should raise the interest of anyone working on innovative applications of mobile manipulators, i.e. the combination of mobile robots with one or more robot manipulators. Ideally, researchers should recognize the opportunity to evaluate and assess their robots and their research approaches in an independent, but well-defined setting. Developers, manufacturers and investors in industrial robotics area should recognize the opportunity to objectively assess and compare different engineering solutions for robot hardware and software design problems in an independent and objective manner, or to use the results obtained by third parties doing such assessments.

1.1 Purpose and Objectives of RoCKIn@Work

The main purpose of RoCKIn@Work is to *foster innovation in industrial service robotics*.

The trend towards globalization of the world economy is now ongoing for more than two decades, and European companies find themselves in and ever-growing competitive position in these global markets. While sufficiently large companies, which are already multinational or global players, are tempted to relocate manufacturing, logistics, and services to countries outside of Europe in order to regain competitiveness, small- and medium-sized companies often do not have this option.

An alternative to relocating enterprise operations to foreign countries is to improve productivity in the native country and to continue to rely on the often excellent trained and experienced workforce in Europe. A large unused potential for a higher degree of automation lies within small- and medium-sized enterprises, and in concepts for more flexible manufacturing and intelligent logistics throughout industry. Robots play an essential role in this process. While pure automation is highly inflexible and requires enormous investments in capital, time, and effort for setting it up, robot-based automation approaches are more flexible and can be easier adjusted to new situations. The robot industry is currently at the brink of continuing this trend by introducing *mobile manipulators* as components in manufacturing and logistics processes. New and innovative applications allow the development of new services.

A commonly agreed-on deficit by the robotics community is its sub-par use of rigid evaluation and assessment procedures in order to compare alternative methods and approaches and to effectively find out which ones are advanced and robust enough for use in developing innovative applications. Benchmarking is one element to improve evaluation and assessment, competitions are another. While benchmarking focuses on the rigidity and validity of the results, competitions can do a great job to raise the interest in solving particular problems and to create more research and development activity in an area. RoCKIn strives to combine both elements in the competitions it designs and organizes.

Another barrier for higher market penetration of mobile manipulators is commonly seen in the lack of standards. While it is not an immediate objective of RoCKIn to work on standardization in robotics, the project still makes significant contributions towards this effect. This is achieved e.g. by identifying *typical scenarios* and *typical tasks* relevant for innovative industrial applications, and by developing *methods and techniques to evaluate and assess various abilities* of such robots. This is an essential step towards standardization.

1.2 The RoCKIn Approach

The RoCKIn project uses a systematic approach to achieve its goals. This approach is illustrated in Figure 1.1 and briefly explained the following five paragraphs. For more details, please refer to Deliverable D-1.1 [1].

(1) Formulation of challenge problems: As a first step, challenge problems need to be formulated. This is a difficult yet critical step, because the challenge problems need to satisfy various constraints, such as being new and interesting enough to raise the interest of the general public, the media, and relevant societal and political stakeholders. Furthermore, the challenge problems should have an interesting application potential in domains with significant market potential to raise the interest of industry. The challenge problems should pose interesting and new research questions to raise the interest of the research community, but solutions need to be within reach in a foreseeable period of two to five years.

(2) Definition of a scenario: Once a challenge problem has been identified and described, we need to define a scenario for it. A scenario description includes all relevant aspects of the *task* to be performed, of the *environment* in which the task is to be performed, and of the *robots* to be used.

For each of these three major aspect categories, a set of relevant features[1] are identified. For each feature, a feature domain, i.e. a range of possible instances of the feature, is defined. Features can be mandatory or optional in a scenario. Thus, while the set of features describes the *structure of a scenario*, the feature domains span a space of possible parameter values and thereby define the *variability of the scenario*.

An interesting aspect of scenario definitions is their relationship with the functionalities that the robots are required to have. One of the major objectives of the RoCKIn project is to better understand how the performance of particular functional subsystems, e.g. for mapping, navigation, grasping, taking orders from humans, etc. correlates with the overall performance of the robot executing a particular non-trivial task, e.g. "*Deliver this parcel to the post office!*". To that extent the scenario definition should include a list of *abstract* functionalities which seem to be implied by the specification of the scenario and its major subcategories task, environment, and robot.

[1] Most of the features can be viewed as *parameters* in a scientific experiment.

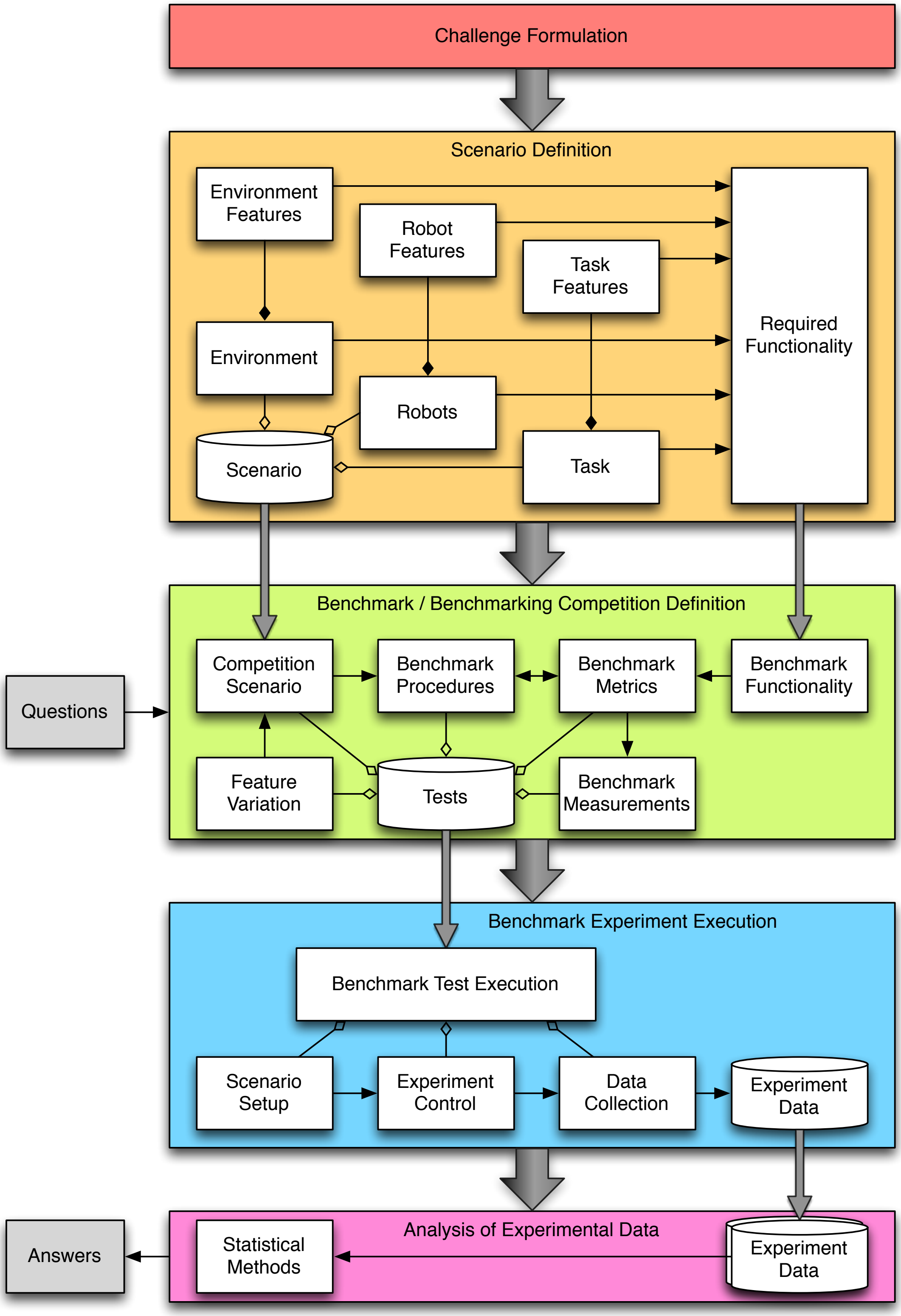


Figure 1.1: Overview of the RoCKIn approach.

(3) Definition of a benchmarking competition: While scenario definitions are an excellent means to characterize a particular application domain, they usually leave an immense variability in various aspects of task, environment, and robot. When performing a benchmark or scientific competition we like to make it as much a scientific experiment as we can.

Aside of identifying the functionalities, defining measures and metrics, and suitably constraining the scenario, we need to specify precise procedures for executing the experiments. In scientific competitions this includes timing, setup of the scenario, initiation of a test run, any kind of interaction with the robots during a run, controlling any environment or task features (e.g. supplying an object at a particular instant of time), scoring, measuring and recording of data, and so on.

(4) Execution of the benchmarking competition: After a benchmark or scientific competition has been defined and a well-specified set of tests is provided, the tests need to be executed. This requires the actual setup of the scenarios specified in the tests, carefully executing the procedures for experiment control, and diligently recording all required data.

(5) Analysis of the results: After execution of the experiments, the recorded data can be analyzed by applying the appropriate statistical methods in order to provide the answers that motivated the benchmark or competition.

1.3 Brief Outline of the RoCKIn@Work User Story

This section provides a brief overview of the competition scenario and will describe – in general terms – the kind of robots targeted and the tasks the robots are expected to perform. Further details on the scenario are deferred to Chapter 2.

The user story is looking into a factory which is specialized in production of small to medium size mechanical parts and mechatronic subassemblies. The factory provides custom products and parts directly over the Internet. The assembly process in the production section depends on the actual demand which changes on daily basis. Thousand of assembled products have to be shipped every day, with each shipment containing between one and a dozen assembled parts. About one third of the shipment sent out will be returned, e.g. because a shipped item was damaged during shipping, is not working properly, is not fitting, or simply not liked by the customer.

The environment consists of several workstations, in which either human workers or robots pick up parts such as screws, nuts, and bolts, rings, etc. and assemble them based on the customer's demand. A central scheduler gets customer orders and assigns them dynamically to the workstations. Once it is known where an order will be assembled, the goods required for it need to be available at the workstation sufficiently on time such that no delays in the packaging process are caused. Each type of product requires a different set of component items for the assembly phase. Some of these items, such as screws, nuts, bolts and rings (*commodity items*) are used in all products, albeit in different numbers, while others are specific to the product type being assembled (*specific items*). All parts are kept in suitable storage containers and are delivered into adequately-sized boxes at the workstations. In addition, the factory periodically receives return shipments and exchange parts. The received packages arrive in different shape, size and condition. Each received package needs to be identified, examined and recorded in the database. Afterwards, the package's content will be extracted and inspected. Each part in the package should be delivered to its respective container and the damaged part will be sort out or forwarded to a human worker for further inspection.

The task to be solved by the robots falls into four categories:

Order Picking and Parcel Assembly: When an order has been assigned to a workstation, robots are requested to get the required goods from the storage. Goods are stored in containers in a warehouse, and have to be delivered to containers at the corresponding workstation. Initially, a human worker will pack the goods into a parcel to be shipped to the customer. In the future, a robot manipulator at the workstation can either assist a human (e.g. for heavy goods) or autonomously place goods into the parcels.

Return Shipment Handling: Return shipment parcels are opened by a human and placed into large boxes at a workstation. Robots are taking out the returned goods and determine the type of good returned. Depending on this type, the returned good will be handled appropriately, e.g. delivered directly to the warehouse, or brought to another workstation for further human inspection and processing.

Order Customizing: As an example for customized orders, we use gift boxes, which could contain anything from sets of cosmetics, body and hair care products, chocolate pralines or candies, toys, or similar items. This task is beyond pure order picking, because the picked products have to be assembled in a specific way in a gift box and must be nicely packaged and wrapped. Delivery and assembly are expected to be done by the robots, while packing and wrapping is expected to be performed by a human.

Warehouse Deliveries Processing: Deliveries to the warehouse usually consist of palettes of parcels, each of which contains multiple items of the same good. The goods have to be unpacked from the parcels and store in boxes in the warehouse.

The warehouse logistics scenario can come in a lot of variations, e.g. in the number of parts considered, their size, color, weight, and shape, the number of workstations, the number of robots available for performing logistics tasks, the storage arrangements for the component items, and the scheduling strategy of the job shop scheduler, to name just a few. While it is possible to start this idea with significantly constrained scenarios involving only a single robot, it should be clear that more realistic scenarios are bound to be inherently multi-robot. These robots will then not only be networked with each other, but also with other items in the environment, such as the scheduler, conveyor belts, electronic scales, item dispensers, etc.

It is possible to extend this scenario later to include the subsequent assembly process, such that semi-finished or finished products are produced as the outcome of this application scenario.

1.4 The RoCKIn@Work Competition

The RoCKIn project runs from 2013 through the end of 2015. The RoCKIn competitions will be held twice, in Fall 2014 and Fall 2015. Both RoCKIn and RoboCup have the intention to eventually merge the competitions in a joint event for the years beyond 2015.

The first RoCKIn event including the RoCKIn@Work competition will be held end of November 2014 at LAAS, Toulouse, France. Details on this competition will be supplied as they become known. Please refer to the web site of the RoCKIn project (`www.rockinrobotchallenge.eu`) and the web pages on RoCKIn@Work.

1.5 Commonalities and Differences with RoboCup

It is not a secret that RoCKIn@Work is inspired by RoboCup@Work, and the RoCKIn-@Work organizers hope that many if not all of the concepts and methods developed for RoCKIn@Work will eventually be picked up and integrated in RoboCup@Work. Nevertheless, it may be illustrative to briefly outline some commonalities and differences between RoCKIn@Work and RoboCup@Work, based on the 2013 rule book of RoboCup-@Work:

- Both RoCKIn@Work and RoboCup@Work aim toward innovative robotics scenarios with very high relevance for the robotics industry.
- RoboCup@Work developed the challenges for the competition from scenarios which are sufficiently general and independent of particular industrial applications (transportation, manipulation). On the other hand, RoCKIn@Work developed the challenges based on convincing and easy-to-communicate user stories, which catch the interest of both the general public and the scientific community.
- RoCKIn@Work emphasizes benchmarking.

Chapter 2

The RoCKIn@Work Challenge

The European robotics industry is convinced that recent developments in robotics, especially more lightweight manipulators, more dexterous hands, mobile manipulators, compliant manipulators, and bi-manual or multi-manual manipulators allow for many new and innovative industrial applications of their products in new markets. In order for these markets to develop, sufficiently realistic demonstrators need to be developed and publicised which show the capabilities of these new robot systems. An appropriate way to do this is by means of a competition such as RoCKIn@Work.

This chapter first sets the stage for RoCKIn@Work by describing in more detail the user story which serves as a basis to derive specific scenarios and tasks for the competition. The user story has a general part and a sequence of scenarios, each of which requires several tasks to be performed. Subsequent sections analyse the research and engineering challenges that need to be addressed when developing robots for solving these tasks, present the concepts and objectives behind RoCKIn@Work, and describe the differences and commonalities with RoboCup@Work in more detail.

2.1 The RoCKIn@Work User Story

The common, general part of this RoCKIn@Work user story is as follows:

The Factory RoCKIn 'N' RoLLIn is specialized in production of small to medium sized lots of mechanical parts and mechatronic assembled products. Fully automated processes are avoided for two reasons: the small lot sizes do not legitimate a fully automated production line and secondly some assembly-steps are very challenging. In this environment, our intelligent manufacturing robot, is an ideal assistant to support the production process with logistics and pre-assemblies.

Aside from providing mechanical parts and mechatronic products, RoCKIn 'N' RoLLIn also integrates the incoming shipments handling of damaged or unwanted products or incoming goods and raw material in their production line. Again, a complete automation is not suitable since the incoming shipments come intermittently in different size, shape and condition. So, to optimize the usage of its intelligent robot (and decrease operation cost), RoCKIn 'N' RoLLIn also assigns the robot with the task of handling received packages.

The production environment consists of several workstations, in which either human workers, robots or stationary machines work or assemble a variety of different goods. A central scheduler gets orders for these goods and assigns the specific assembly steps to the workstations. Once it is known where the next step is to be done, the parts required

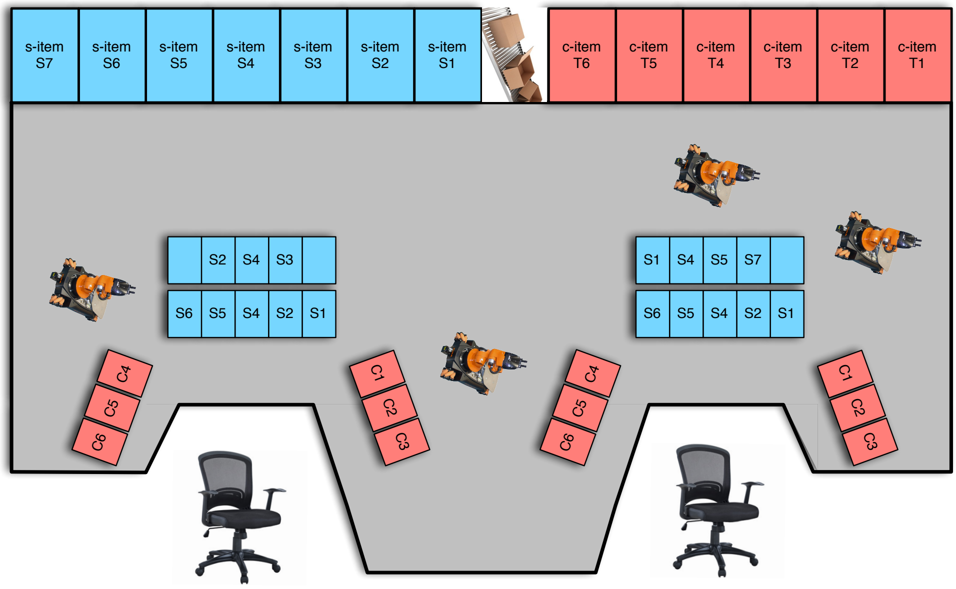


Figure 2.1: An environment for the manufacturing logistics and assembly support user story. Two workstations for human workers are shown on the bottom. The grey area above is all table area. To the left and right of each workplace there are boxes with commodity items (red). A revolving battery of boxes to be filled with specific items (blue) is right in front of the workplace. The robots maneuver on the free table area above the work places and fetch the requested items from storage areas with larger container boxes on the top. A conveyor belt in the center top carries completed items and received incoming package.

need to be available at the workstation sufficiently on time such that no delays in the production process are caused. The same procedure is executed for parts that have to be worked in stationary machines at a certain workstation. Each type of product requires a different set of items for the assembly phase:

- Some of these items, such as screws, nuts, and bolts, rings, etc. (commodity items) are used in many products, albeit in different numbers.
- others are specific to the product type being assembled (specific items)

All parts are kept in suitable storage containers and are delivered into adequately-sized boxes from stock to the workstations.

On the other side of the stock, incoming shipments are processed by the robots. Received parcels need to be categorized and identified for the factory's database. The packages need to be unfolded, content has to be analysed and transported to specific locations in stock or for following manual examination.

A list of task these robots are supposed to perform includes the following activities

1. **Logistic support:** Maintaining the supply level of parts for assembly process to keep the production flow.
2. **Assembly assistance:** Helping assembly process through cooperative work or providing support in delivering the required parts and tools.
3. **Quality assessment:** Ensuring the quality of parts and assembled products through observation of dent or other damage mark.

4. **Shipment handling:** Identifying shipment box through barcode or QR code which is followed by the extraction and delivery of its content to the relevant workstation.

5. **Inventory management:** Updating inventory for inspected items, assembled parts and received shipment.

(a) Subaru transport robot [2]

(b) KUKA LWR

Figure 2.2: Robot assistants in logistics and pre-assemblies.

The RoCKIn@Work User Story is categorized into different episodes, which will be described in the following sections.

2.1.1 Episode 1: Assisting Assembly and Manufacturing

RoCKIn 'N' RoLLIn's produces wheels and drive axles of a mobile robotic platform. Both assemblies are finished in this factory and then send to the robot manufacturer, who integrates them into the platform. The production process is divided into the following steps:

1. Machining of several parts for the assembly

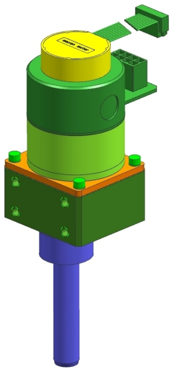
2. Assembling the drive axle

3. Assembling wheels ☐ To be determined. ☐

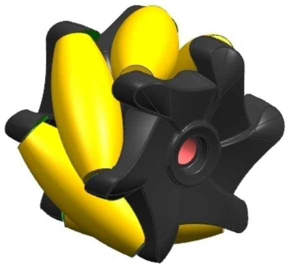
4. make package of wheels and axles for shipping

The robot assistants main task is to support the assembly and production line. It supplies the workstations with parts assists human workers with their work steps and assembles parts of the products.

Remark: The assembly of the drive axle serves as an initial example for this episode. More examples of assembling, handling and machining groups of parts will be inserted here in the future to enlarge the variety of handled objects. Assemblies can also be proposed in the future by external partners.

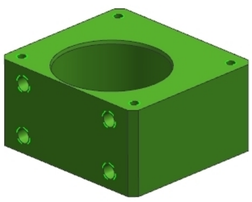


(a) Drive axle

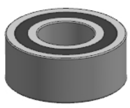


(b) Wheel

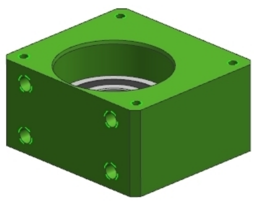
Figure 2.3: Products of Assembly



(a) Bearing box



(b) Bearing



(c) Finished force fitting

Figure 2.4: Task "*Prepare assemble-aid-tray for force fitting*"

2.1.1.1 Task "*Prepare assemble-aid-tray for force fitting*"

The robot collects the bearing box (part-ID: AX-01) from stock and delivers them to workstation 1. For the following production steps the bearing boxes have to be inserted into an assembly-aid-tray at this workstation. The trays have an identifier (QR, RFID) and have to be scanned in the beginning to start with the lot. In the following production steps the identifier is reused to monitor the production process and to be able to react fast in case of occurring errors. The assembly-aid-tray is plugged into a rig at the workstation and then filled with bearing boxes by the robot. The tray has a loose fit for easy filling. The robot supplies workstation 1 with bearings. The tray is then inserted into the machine

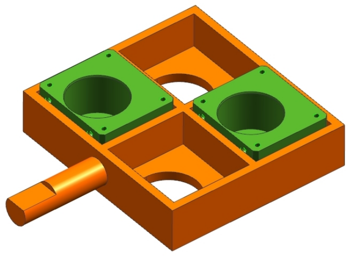


Figure 2.5: Assembly-aid-tray for bearing boxes

and the bearings are then force fitted into the bearing boxes by the machine (or by human) of workstation 1. After finishing one tray the machine has to be emptied and the tray has to be transported to the next workstation or to the stock, if workstations have no more capacity. The bearing boxes with mounted bearings stay in the tray for nearly the rest of the production. Workstation 1 can hold only a limited number of unfinished and finished trays in special rigs.

2.1.1.2 Task "*Cooperative assembly of axle and bearing*"

The tray has to be delivered to workstation 2 and plugged into the workstation rig. Hereupon, the robot takes the axis (part-ID) and inserts it one by one from the lower side of the tray into the bearing boxes. A human has to screw the shaft nut onto the axis to mount it to the bearing. The robot has to counter the axis until it is fixed. After mounting one axis, the worker has to check if the axis is of easy motion.

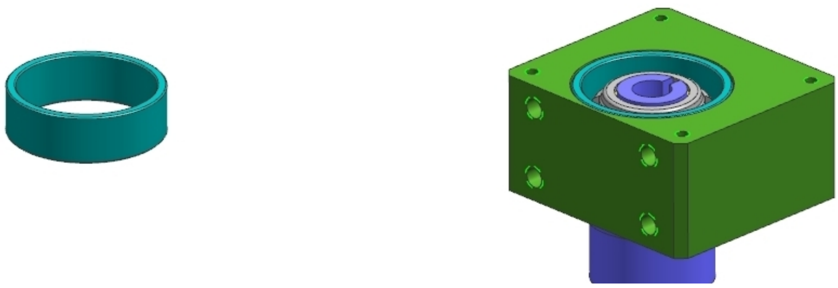
2.1.1.3 Task "*Assemble the distance tube*"

At the next workstation (no. 3) the robot has to perform an assembly. The distance tube distance has to secure the bearing's position after final assembly. For the robot it is a very small fit. The distance tube has to be positioned and inserted very carefully an in the correct angle. Failing this, it would cant and not be possible to assemble it any more. If the robot fails, the error has to be reported immediately and the tray has to be transported to a special rig for manual inspection.

□ An open question is whether to consider the error handling in each task or whether to make it a more general top-level task. □

(a) Axis (b) Shaft nut (c) Finished cooperative assembly

Figure 2.6: Task "*Cooperative assembly of axle and bearing*"



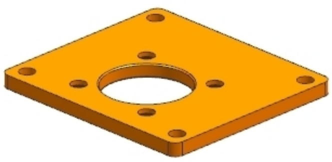
(a) Distance tube (b) Shaft nut

Figure 2.7: Task "*Assemble the distance tube*"

2.1.1.4 Task "*Plate drilling*"

This task simulates an incomplete or faulty delivery from an external component supplier.

The cover plate of the bearing box has eight holes for connecting the motor with the bearing box. The four central holes need to have a cone sink. Unfortunately the supplier forgot to drill these sinks. The robot has to transport a bundle of cover plates in a



(a) Cover plate

(b) Cone sink driller

Figure 2.8: Task "*plate drilling*"

customized box to workstation 4. The plates are stored in this box like file cards. Firstly, the robot has to execute a quality control procedure. The cover plate is taken out from the box and put on the workstation table. Then the robot has to detect the existence of the eight holes. Secondly, if the quality process was successful, the cover plate is inserted into a rig for drilling the four missing cone sinks. After the drilling process, the robot has to take the cover plate out of the rig and into a new customized box.

2.1.1.5 Task "*Fill a box with parts for a manual assembly*"

The robot composes boxes with parts for the manual, final assembly of the drive axle. The boxes have no special subdivisions. They only have foam material t the bottom to guarantee a safe transport. Therefore, the robot has to plan the collecting to arrange the parts next to each other. Needed parts:

- the motor with gearbox and encoder, part ID
- machined cover plate
- pre-assembled bearing box (taken from the aid-tray now) (part-ID: AX-08)

In addition to that, screws are needed for this assembly step. These commodity items have to be always available in sufficient quantity. The robot picks up a new empty box, scans its code. Then he collects the needed parts and delivers the box to workstation 5. At the workstation a worker takes care of the assembly. After finishing the drive axle, the worker puts it back to the same box and triggers the status to the central scheduler. The robots scans the box and takes it back to the stock.

2.1.1.6 Task "*Fill a parcel for a customized order*"

Drive axles are shipped out in combination with wheels or as a single spare part. These are a few of the optional combinations of the shipped products:

- four drive axles, two left wheels, two right wheels
- one drive axle with right wheel
- one drive axle with left wheel
- one drive axle
- one left wheel
- one right wheel

Depending on what package has to be composed, the robot has to choose the correct parcel from stock, which has already a code to scan and place the needed products into it. To create no damage at shipment, the robot has to place packaging material between the products or put over carton gauges over them first. The parcel is left open and transported to the stock. The next steps of this task are to be done at the shipment area of the factory.

2.1.1.7 Task "*Clean up the floor*"

While the robot is moving from one station to the other or to the stock, obstacles can appear. Obstacles can be of the following categories:

- trash or dirt like parts of packaging material or metal swarf
- tools from workers
- parts of the assembly process
- commodity items (e.g. screws, rings, bolts, nuts)

The robot has to be able to detect these obstacles and avoid collisions. Furthermore it has to classify the objects into a category. Depending on more criteria (e.g. the number of objects) it has to report the issue to the central scheduler and the area is marked as locked for passing. A new task is then assigned to the robot (or his successor): Depending on the category of the obstacle, the robot has to remove the trash, collect the parts and bring it to a workstation for manual quality control or scans the code on the tool and transports it to its belonging worker at the workstation. If the tool has no code to scan, it is delivered to the closed workstation with a human worker.

2.1.1.8 Task "*Learn assembly by watching humans*"

A human worker assembles a group of mechanical parts. The robot has to recognize, remember and realize the same assembly itself. This can be a measure of a part in the assembly, that has to be handled more carefully.

□ INSERT HERE: rest of manufacturing Logistics tasks: report more errors, clear empty trays and boxes □

(a) Different sizes of parcels

(b) Parcel transportation flow

Figure 2.9: Received shipment

2.1.2 Episode 2: Handling Incoming Shipments

RoCKIn 'N' RoLLIn receives returned shipments or incoming goods in daily basis. The received package consists of raw material and parts which will be processed (examined for damage and identified for the factory database).

The operation for each received shipment includes the following routine steps:

1. Delivering the shipment to examination workstation.

2. Identification of the package received and examination of its condition.

3. Unfolding the package and delivering the contents to its respective containers.

The robot task is to performs the routine steps for each received shipment. The operation consist of the following tasks.

2.1.2.1 Task "*Transport parcels*"

The received shipments come in different sizes and shapes (Figure 2.9). When a parcel arrived on the conveyor belt, the robot collects the parcel and delivers it to the parcel inspection workstation for further processing. Each parcel should be delivered individually since the inspection workstation can only process one parcel at a time.

2.1.2.2 Task "*Inspect parcels*"

The condition of the parcel will be checked in the inspection workstation. Any damage marks detected on the package will be documented before the parcel is opened. Additionally, every return parcel should have a sticker with a barcode ID provided beforehand by RoCKIn 'N' RoLLIn to its customers. There are several different possible locations for the barcode placement on the parcel. The robot task is to locate the barcode location on the package and send the information to the RoCKIn 'N' RoLLIn's database along with the information on the parcel condition. Parcels without any barcode ID and parcels which are seriously damaged will be sorted out. After the inspection procedure is completed, the parcel will be forwarded to the unpacking procedure.

2.1.2.3 Task "*Unpack parcel contents*"

In this operation, the parcel will be opened and its content will be analysed. Different types of parcels require different type of tools for the unfolding process. These tools will be provided in the unfolding workstation. The robot is tasked to grasp the correct tools for each parcel and proceed with the task of unfolding the parcel. After the parcel is opened, the robot continues with examining its content. The robot identifies and delivers each item in the parcel to its respective containers and updates the RoCKIn 'N' RoLLIn's factory database.

2.1.3 Episode 3: Manufacturing Logistics

□ To be revised and included for a future revision 2.0. □

□ This episode is to some extent already covered in RoboCup@Work scenarios; we will eventually include a revised version here. □

The robots have to observe the production schedule for each workstation and ensure that required specific parts will be there on time and placed in the correct, pre-specified places. Furthermore, the robots must ensure that a sufficient supply level for all commodity items is maintained.

2.2 Analysis of Challenges in RoCKIn@Work

□ This section is planned for inclusion in revision 2.0 □

2.2.1 Robot System Design Challenges

2.2.2 Interaction Challenges

2.2.3 Perceptual Challenges

2.2.4 Robot Motion Challenges

2.2.5 Cognitive Challenges

2.2.6 Software Engineering Challenges

2.3 Concepts and Objectives Behind RoCKIn@Work

□ This section is planned for inclusion in revision 2.0 □

2.3.1 Supporting Replication and Repeatability

2.3.2 Boosting Benchmarking

2.3.3 Contributing To Standardization

2.3.4 Stimulating Cooperative Team Work

2.4 RoCKIn@Work and RoboCup@Work: Commonalities and Differences

□ This section is planned for inclusion in revision 2.0 □

To round up the description of the RoCKIn@Work competition, we briefly oultine the commonalities and differences of the RoCKIn@Work and RoboCup@Work competitions. The comparison is based on the information documented the 2013 Rule Book for RoboCup@Work.

2.4.1 Commonalities

2.4.2 Modified Commonalities

2.4.3 Differences

Chapter 3

The RoCKIn@Work Scenario

The following chapter describes in more detail the design of the arena with its facility and the used robots. Furthermore, the tasks and their conditions is presented in more detail.

3.1 The RoCKIn@Work Environment

The RoCKIn@Work Environment is designed to consider the following conditions

- close to a real factory environment
- clearly arranged and sized for an audience
- easy to clone/adapt to a team's laboratory regarding size and cost
- multiple teams should be able to compete in the arena at the same time.

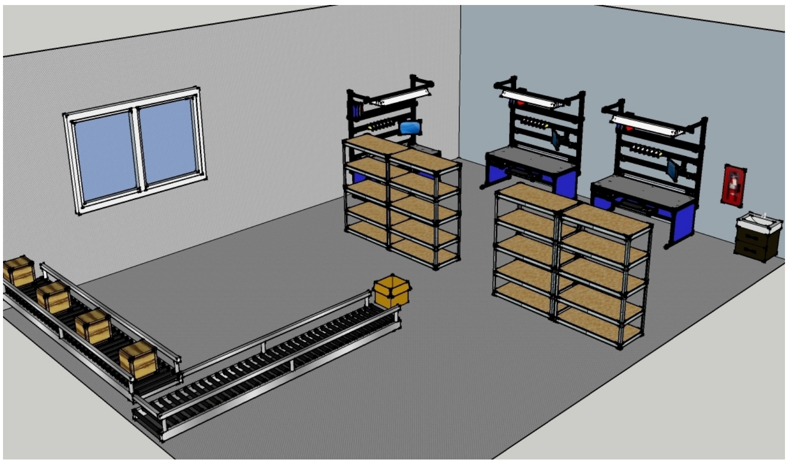


Figure 3.1: RoCKIn N' RoLLIn factory

Therefore the arena is chosen as a ground-level area and represents a factory (or modules of it) described in the precedent chapter with an industrial look-a-like. The arena is build in a scaled down environment for better visibility and for better installing in competitors laboratories.

The arena is split into two areas: one warehouse and logistics area and one assembly and production area. The two areas are separated by one or more shelfs realizing the supply with material, parts and finished products. This separation makes it possible to have more than one team in the arena at a time, besides the possibility of enlargement of the whole arena for multiple robots.

3.1.1 Environmental Elements

3.1.1.1 Floor

Scenario Specification 3.1 (*Floor Condition*)

The floor of the environment can be a ground level floor or an elevated environment, to let scaled-down robots better interact with human workstations.

Scenario Specification 3.2 (*Floor Material*)

The floor planes are made of flat, solid material and should have characteristics comparable to floors in factories. The floor has a mostly uniform and light color, e.g. grey or white. The floor can be made of laminated boards, plywood, PVC, or similar. There is no lose material (like sand) on the floor.

Scenario Specification 3.3 (*Floor Slope*)

Every plane in the environment has a slope of approximately $0°$*. Spots of unevenness can occur of up to 10mm in any direction. The environment has no stairs.*

Scenario Specification 3.4 (*Ramps*)

Ramps with up to 5 *percent slope may be included in the environment.*

3.1.1.2 Walls

The walls purpose is to keep the audience away from the arena and in a safe distance from the robots. Furthermore it limits the area where the robots have to navigate.

Scenario Specification 3.5 (*Walls Material*)

The walls are made of a solid, flat material to enable navigation with sensors (e.g. laser-scanner).

The walls surround the complete arena with exception of where workstations, shelf or similar are located. They serve as a barrier for the robot to not collide directly with e.g. conveyor belts or not to fall down the elevated environment.

3.1.1.3 Workstations

Workstations are coming in different variations depending on the application that is realized. The simplest version is an elevated field (table) where an assembly is processed. Other variations can have a fence with an automated robot or machine cell inside. this cell has to be supplied with parts or trays through a gate (with safety facility with e.g. light barrier). A manual workstation (with a human) can be a simple field or some sort of table where the robot and the worker can work together.

Scenario Specification 3.6 (*Workstations Referencing System*)

The workstation may have or have not a referencing system and can be of different *shape (rectangular, round).*

3.1.1.4 Shelves

In the center and on the logistics area shelves are needed to realize the transportation storage and supply of all goods an raw material.

Scenario Specification 3.7 (*Shelf Variations*)

Shelf can appear in different versions: simple shelf to freely arrange boxes or parts on it, shelves with facilities to take special sized trays or boxes or shelf with roller-systems.

(a) Simple shelf (b) Shelf for boxes (c) Shelf with slope and rollers

Figure 3.2: Products of Assembly

The shelves can be mobile as well and moved freely or on tracks. Conveyor belts transport incoming shipments to the sorting station and can be connected to mobile shelves.

Include specs of dimension, color, and material.

3.1.1.5 Obstacles

Obstacles can be part of a task or in general work as an non-projectable challenge. In the real factory environment, workers not necessarily arrange objects in a nice rectangular fashion and put them almost anywhere, or they unintentionally drop trash.

add images of obstacles

Examples of obstacles include palettes, trash, tool trolleys, and boxes.

3.1.2 Common Environment Features

The following list describes a set of environment features that are not specific to particular spatial areas and apply to the whole environment. Each of them should be measured and recorded for all experiments performed in the scenario environment.

Scenario Constraint 3.1 (*Air Temperature*)

The air temperature of the environment is measured in degrees Celsius (°C*). The air temperature should not be less than* 16 °C *and no more than* 26 °C*; it should typically be between* 20 °C *and* 22 °C*.*

The air temperature is usually quite evenly distributed through environment space, but temperature deviations may occur near heat sources (machines, workstations, conveyor belt motors, lamps, etc) or heat sinks. If such elements are used in an environment, the air temperature in their vicinity should be measured and recorded for each experiment.

Scenario Constraint 3.2 (*Atmospheric Pressure*)

The atmospheric pressure in the environment is measured in hectopascals (hPa). The atmospheric pressure should be no less than 950 hPa *and no more than* 1050 hPa*; it should typically be between* 980 hPa *and* 1020 hPa*.*

The atmosperic pressure is evenly distributed through the environment space.

The previous two features are known to influence e.g. the performance of acoustic time-of-flight sensors such as sonars. In addition, they also influence the next feature, relative humidity, which may prevent safe operation of the robot (due to constraints imposed by its electronic components) or influence the robot's ability to perform manipulation tasks.

Scenario Constraint 3.3 (*Relative Humidity*)

The relative humidity in the environment is measured in percent (%*). The relative humidity should be no less than* 10 % *and no more than* 90 %*; it should typically be between* 30 % *and* 60 %*.*

The relative humidity is usually quite evenly distributed through environment space, but deviations may occur if water supplies are in the vicinity of heat sources or near heat sinks.

The next two feature obviously influence the performance of optical sensor system, such as cameras, 3D cameras, or infrared sensors. Illuminance is "the total luminous flux incident on a surface" [3] and a measure of the intensity (aka brightness) of lighting in the environment. Color temperature is a measure to quantify the color of a light source, sometimes also referred to as the "warmth". Both features influence the behavior of optical sensors and how colored surfaces will be perceived.

Scenario Constraint 3.4 (*Illuminance*)

Illuminance is measured in lux (lx). Illuminance in the environment can be as low as 0 lx *and should not exceed* 10000 lx*; the typical range should be between* 500 lx *and*1500 lx*.*

Illuminance cannot be expected to be evenly distributed throughout the environment. The specific interior design, the placement of furniture, the configuration and placement of lamps and other light sources may create significant variance of illuminance in the environment (dark corners, bright spots). Daylight in the environment would be desirable, even if it is rapidly changing due to weather conditions, but it is not required. If daylight influence is present, the scenario environment should foresee a device to record illuminance levels during experiments.

Scenario Constraint 3.5 (*Color Temperature*)

> *Color temperature is measured in Kelvin (K), the unit of absolute temperature. The color temperature in the environment will not be less than 1000 K and not exceed 8000 K; the typical range will be between 1500 K and 5000 K, with up to 8000 K with daylight influence.*

In speech-based human-robot interaction, the robot's performance may critically depend on the level of acoustic noise present in the environment.

Scenario Constraint 3.6 (*Acoustic Noise*)

> *Acoustic noise will be measured in decibels (dB). The noise level in the environment can be as low as 0 db and should under no circumstance exceed 100 db, the typical range should be between 40 db and 85 db (this is the limit, a worker can act without ear protection), with occasional exceptions (alarm signal, working close to a running machine) of up to 90 db.*

The robot and other devices may (have to) use the local network infrastructure, and both performance and robustness of the robot's services may critically depend on the situation on the network. In particular, utilization of the network for other purposes than the operation of the robot and servicing its functioning, may reduce the effectively available bandwidth usable by the robot, and it may not any more be able to deliver certain services in a robust manner.

Scenario Constraint 3.7 (*Wireless Communication Noise*)

> *The wireless communication noise will be measured in kilobytes per second (Kb/s) and will be recorded both as absolute value and as relative value (percentage) of the maximum available bandwith. The wireless communication noise level may range from 0 % to 100 % of the available bandwidth. The typical range will be between 10 % and 50 %.*

When wireless communication is relevant for particular experiments, these experiments should foresee test runs using different situations, e.g.

- **Negligible Traffic** Almost no traffic is detected on the network.
- **Noise Traffic** Erratic random noise is detected at different levels.

3.2 The RoCKIn@Work Robots

As the arena is designed to build up a scaled-down environment, it is useful to have robots, that can fit in this small-sized factory. The design of RoCKIn@Work will be such that it is possible to participate and successfully solve the tasks using mobile manipulators such as the KUKA youBot, the Little Helper robot by Aalborg University, or similar designs. Nevertheless, workstations that interact with the robots can be of normal scale, providing a possibility to fill them by scaled down robot assistants.

3.2.1 Features of the RoCKIn@Work Robots

The robots have to be able to drive from one workstation to the other, from one shelf to a conveyor belt. They are mobile platforms (on wheels) and can move on the floor described in 3.1.1.1. The robots need at least one manipulator to pick up or process objects.

□ A minimal and maximal size should be specified for objects to be gripped. □

A gripper or robotic hand is able to grasp objects belonging to the production or logistics process. Optional: There are several tools used for different production steps. The robot should be able to pick them up and use them the right way.

3.2.2 Sensors of the RoCKIn@Work Robots

The robots use sensors. Sensors are not restricted to specified technology. Some tasks can require a special sensor to interact with a workstation (e.g. RFID). The use of an external sensor system (e.g. ground-support external camera system) can be restricted.

3.3 The RoCKIn@Work Tasks

This section analyzes the RoCKIn@Work tasks and their sub-tasks. For each (sub-)task the initial and final situation, the activities to be performed, the quality criteria, constraints and features are investigated below.

3.3.1 Assisting Assembly

3.3.1.1 Task "Fill a tray for an automated process"

This task is derived from the description in Section 2.1.1.1. The robot received command from the user on which system needs to be assembled. The task focus is on parts placement and efficient navigation. The task "Fill a tray for an automated process" consists of the following sub-tasks.

1. planning and scheduling
2. parts detection
3. parts acquisition and placing
4. planning and failure detection

Initial state

- The robot is presented with a custom-tray where placement of each part can be identified (i.e. label).
- The location of containers are known to the robot.
- The assembly process that will be performed by the user is unknown to the robot before the test.

Final state

- The robot placed every part properly to the tray and inform user on the status of missing parts (i.e. empty container, unreachable workstation).

Activities

- The robot identified the required parts from labels in the tray.
- The robot perform planning for moving through different containers to collect the parts efficiently.
- The robot detects the part in the storage container.
- The robot acquire the part from the storage container.
- The robot placed each part to the tray.
- The robot perform reasoning and make decision on the task status (success and failure). For example, the empty storage containers means that the task cannot be executed and it should be reported to the user.

Constraint

- The combination of required parts are unknown to the robot (the robot only know the label place in the tray).
- All parts required to be collected efficiently (duration will be an important evaluation aspect).

3.3.1.2 Task "Cooperative assembly"

This task is derived from the description in Section 2.1.1.2. In the assembly work station, the robot received command from the user for cooperative work. The task focus on user interaction and compliance control. The task "Cooperative assembly" consists of the following sub-tasks.

1. user interaction.
2. object detection.
3. object manipulation.

Initial state

- The user is sitting in the workstation.
- The parts (axle and bearing) is placed in the workstation.

Final state

- The parts is assembled appropriately through cooperative work between the human user and the robot.

Activities

- The robot receives command from the user (speech recognition is optional).
- The robot detects the parts in the tray.
- The robot collects the required part for the assembly process based on the user request (i.e. "hold the bearing" or "insert the axle").

Constraint

- The parts which the robot handle during the cooperative work is unknown.
- While performing the task, the robot should adjust its motion based on the user position and gesture (optional).

3.3.1.3 Task "Assisting machinery process"

This task is derived from the description in Section 2.1.1.4. Nearby to a machinery, the robot received a stack of plates which are going to be processed by the machine. The robot task is to inspect the state of each plate and determined whether the plate satisfy the condition for further processing with the machinery. The task focus on observation and precise placement. The task "Assisting machinery process" consists of the following sub-tasks.

1. Object grasping.
2. faulty detection.
3. object placement.

Initial state

- The robot is presented with a stack of plates.
- Each plates have different state.

Final state

- The faulty plates are sorted out from the good plates.
- The good plates are processed by the machinery (optional).

Activities For each plate in the stack, the robot performs the following activities:

- The robot grasp one of the available plate.
- The robot observed the condition of the plate.
- The robot determined the condition of the plate. Every faulty plate is discarded in a container and every good plate is placed in the machine.
- After placing the good plate in the machinery, the robot proceed with operating the machinery (optional).

Constraint

- The ratio of good plate and faulty plate are unknown to the robot.

3.3.1.4 Task "Assemble a mechanical group of parts"

This task is derived from the description in Section 2.1.1.3. In the assembly work station, the robot received command from the user to assemble parts on the assembly tray. The task focus on sequential object manipulation. The task "assemble a mechanical group of parts" consists of the following sub-tasks.

1. parts detection
2. parts handling
3. planning

Initial state

- The robot is presented with a completed assembly tray (screws, rings and bolts).

Final state

- All the parts in the tray is assembled appropriately.

Activities

- The robot receive command from the user (speech recognition is optional).
- The robot detects the parts in the tray.
- The robot perform the assembly process.

Constraint

- The combination of parts in the assembly tray is unknown.
- The rings, nuts and bolts in the assembly tray have different size.
- (Optional) The combination of ring and nut in each bolt can differ depending on user request.
- (Optional) There is a possibility of an unused or unknown part in the tray.
- (Optional) There is a possibility of missing parts.

3.3.1.5 Task "Fill a box with parts for manual assembly"

This task is derived from the description in Section 2.1.1.5 and 2.1.1.6. The robot received command from the user on which system needs to be assembled or which parts require for the assembly process. The task focus on assisting assembly and sequential delivery (i.e. screwdriver and wrench should not be delivered last). The task "Fill a box for a manual (human) assembly step" consists of the following sub-tasks.

1. user interaction
2. navigation
3. parts detection
4. parts grasping and delivery
5. planning and failure detection

Initial state

- The robot has the knowledge on several assembly procedure, assembled system and parts.
- The location of container storage are known to the robot.
- The assembly process that will be performed by the user is unknown to the robot.

Final state

- The robot either: 1. delivered every part to the user sequentially. 2. inform user on the status of undelivered part (i.e. empty container, unreachable workstation) and request further instruction
- The robot acquire the knowledge on the status of each container storage (optional).

Activities

- The robot receive command from the user through speech.
- The robot move to different storage containers to collect the part sequentially and execute the plan.
- The robot detects the parts in the storage container.
- The robot acquire the parts from the storage container.
- The robot delivered the parts sequentially to a provided box in the assembly workstation.
- The robot perform reasoning and make decision on the task status (success and failure). For example, the empty storage containers means that the task cannot be executed and it should be reported to the user.

Constraint

- The status of each container is unknown to the robot.
- The command can be: 1. a list of required parts 2. the name of the assembled system
- The environment condition is unknown when receiving the command (obstacle exists).
- The parts needs to be delivered sequentially.

3.3.1.6 Task "Clean up the floor"

This task is derived from the description in Section 2.1.1.7. The robot received command from the user to clean up the workstation. The task focus on object detection and planning. The task of "Clean up the floor" consists of the following sub-tasks.

1. object detection
2. planning
3. object grasping and delivery
4. reporting

Initial state

- There are several objects on the factory floor.

Final state

- The robot clear up the factory floor and delivered each object to its relevant storage area (tools to the workstation, parts to the container).
- The robot make a record of all detected objects, returned objects and unknown objects.

Activities

- The robot can either use reactive search-and-deliver or perform complete observation first to determine the cleaning strategy.
- For each detected object, the robot grasp the object and deliver the object to its designated storage area.
- When completed, the robot informs the initial state and the final state of the factory floor.

Constraint

- The number of objects on the factory floor are unknown to the robot.
- The obstacle can be: 1. in places which prevents the robot from reaching one of the storage area for objects or tools. 2. is an unmovable object. and 3. is an unknown object.

3.3.1.7 Task "Learn assembly by watching humans"

This task is derived from the description in Section 2.1.1.8. In the assembly work station, the robot observed how a user perform an assembly process. The task focus on reasoning and mimic. The task "Learn assembly by watching humans" consists of the following sub-tasks.

1. parts detection
2. reasoning
3. parts handling

Initial state

- The robots are presented with two different part.

Final state

- The part has been assembled.
- The robot has the knowledge on assembly procedure.

Activities

- The robot receive command from the user (speech recognition is optional).
- The robot detects the parts in the tray.
- The user perform the assembly process.
- The robot detects the assembled parts.
- The robot perform the assembly process.

Constraint

- The knowledge is justified when the robot can reenact the assembly process.
- (Optional) the robot can perform decoupling of the assembled parts.

3.3.1.8 Task "Process parts for assembly" (optional)

The robot received command from the user on the required condition of the part (i.e. part for assembling pink mechanum wheel). The task focus on operating machinery and failure detection. The task "Prepare parts for assembly" consists of the following sub-tasks.

1. navigation
2. parts detection
3. parts acquisition and delivery
4. planning and failure detection

Initial state

- The robot has the knowledge of color as a state of the part.
- The location of container storage are known to the robot.
- The location of painting workstation is known to the robot.
- The robot has the knowledge on the procedure of operating the machinery.

Final state

- The robot either: 1. delivered the part to the user based on the desired condition 2. inform user on the status of undelivered part (i.e. non-operational machinery) and request further instruction

Activities

- The robot receive command from the user (speech recognition is optional).
- The robot move to the respective storage containers.
- The robot detects the parts in the storage container.
- The robot acquire the parts from the storage container.
- The robot delivered the parts to the painting workstation.
- (Optional) The robot perform checking for machine status (active, ink level, error indicator).
- The robot operates the machinery to execute the painting process.
- The robot perform analysis on change of the part condition.
- (Optional) The robot redo the painting process if the result is non-satisfactory.
- The robot delivered the finished part.

Constraint

- The status of each container is unknown to the robot.
- The command can be received will be in color space.
- The machine needs to be configured based on the desired color space.
- The machine state are unknown to the robot (ink level).
- The parts needs to be delivered with report on the task execution (i.e. the painting process was performed twice).

3.3.2 Handling Incoming Shipments

3.3.2.1 Task "Transport parcels"

This task is derived from the description in Section 2.1.2.1. In the factory, the robot is informed of an incoming package either through notification sound, user input or perception. The robot locates the package in a predefine package receiving area and delivered the package to examination workstation. The task focuses on vision, motion and grasp planning. The task "Package Transportation" consists of the following sub-tasks.

Sub-tasks

- Acknowledge incoming package.
- Locate the incoming package in the receiving area.
- Grasp the package.
- Deliver the package to the examination workstation.

Initial state

- A package is received either through a conveyor belt or placed by a user.
- The robot is informed that there is a new package received and knows that the package is placed within a predefine area.

Final state

- The package is placed in the examination workstation by the robot.

Activities

- The robot moves to the receiving area.
- The robot locates the package.
- The robot grasp the package.
- The robot delivered the package to the examination workstation.

Constraint

- The package precise location is unknown to the robot.
- The package size and shape is unknown to the robot.

3.3.2.2 Task "Inspect parcels"

This task is derived from the description in Section 2.1.2.2. In the examination workstation, the robot performs complete observation of one package and records all existing damage mark. After the observation process is completed, the robot record the identification number in package (barcode) and update the factory database. The task "Package Examination" focuses on perception.

Sub-tasks

- Damage mark identification
- Locating barcode id

Initial state

- The state and identification number of the package is unknown to the robot.

Final state

- The package information (condition and identification number) is stored in the factory database by the robot.

Activities

- The robot performs observation of the package and identify all existing damage mark.
- The robot locate and record the identification number of the package.
- The robot update the factory database with the package information

Constraint

- The package condition differs (perfect condition, several damage mark and beyond repair).
- The identification label position and orientation is unknown to the robot.
- There are packages without identification label.

3.3.2.3 Task "Unpack parcel contents"

This task is derived from the description in Section 2.1.2.3. In this task, the robot function is to assist human in extracting the package content. This includes providing the human with the required tools and identifying the content of the package. The task focuses on identifying parts and tool handling.

Sub-tasks

- Acquiring tools necessary for unfolding the package.
- Identifying the objects in the package.

Initial state

- Different tools is placed nearby the workstation and the location is known to the robot.
- The package is closed.
- The content of the package is unknown to the robot.

Final state

- The user received the required tool from the robot.
- The package is opened and the robot identify all the objects inside the package.

Activities

- The robot received commands from the user on which tool is required for the opening the package
- The robot acquired the requested tool and delivered it to the user.
- The robot identify all available objects in the package after the package is opened.

Constraint

- The tool required by the user is unknown to the robot and the user may request for an unknown tool.
- The number of item in the package differs and there are unknown objects.

3.4 Functionalities Required for RoCKIn@Work

In this section, the functionalities required for each task is presented collectively.

3.4.1 Assembly assistance

The required functionalities for all assembly subtask is:

- Grasping object in different placement and state (storage containers, assembly tray)
- Grasping different objects (bolt, nuts)
- Object placing and hand-over (assembly tray, assembly box, cooperative assembly)
- Compliant manipulation (operating switch)
- Scheduling (sequential parts delivery, cooperative work, parts placement in assembly tray)
- Object detection for identification of unknown parts, damaged part and colored part.
- Object categorization (different size of bolts, rings, and nuts).
- Failure detection and identification (missing parts, unreachable workstation).
- Label and sign identification (non-operational machinery, assembly tray labels).

Chapter 4

RoCKIn@Work Competition Design

4.1 Competition Elements

☐ To be included in a future revision: a general explanation of the competition elements, like competition stages, tests, technical challenges, reliability exercises, and open demonstrations. ☐

4.2 Stages

The stage system is unavoidable when there is a large number of participant. In accordance to the RoCKIn principle, the use of stage system in the RoCKIn@Work competition will be minimize (only in final stage). In the occasion that the stage system is being used, the OC and the TC will jointly determine which teams will be qualified to proceed further in the competition. The team qualification will be based on the following criteria:

- The score difference in team ranking.
- Innovation aspect of the solution shown in the earlier test.

In the case of final stage, the following additional criteria will be evaluated:

- The number of test solved with satisfactory result.
- The repeatability in performing the tests.
- The reliability in executing the task.

4.3 Tests

The subsequent sections describe tests to be included in RoCKIn@Work:

4.3.1 Assembly-aid-tray Test

This test is derived from the description in Section 3.3.1.1. In this test, the robot is tasked with detecting identifiers and delivering objects.

4.3.1.1 Mini-Story of the Test

This test targets the task ”fill a tray for an automated process”.

4.3.1.2 Functionality Benchmarked by the Test

- Signs detection
- Navigation and planning
- Object manipulation

4.3.1.3 Benchmark Metrics

4.3.1.4 Benchmark Scenario

- Environment 3.1
- Robot 3.2

4.3.1.5 Feature Variation

The test requires the robot to have object manipulation capability, vision system and mobility.

Complexity I

- Task features:
  - The robot detects the required parts through signs in the assembly tray.
  - There are unknown signs in the assembly tray.
- Environment feature
  - There is another robot in the environment.
  - Some parts are not available in its container.

Complexity II

- Task features:
  - The robot detects the required parts through signs in the assembly tray.
  - All the signs are known to the robot
- Environment feature:
  - There is only one robot in the environment
  - Some parts are not available in its container.

Complexity III

- Task features:
  - The robot detects the required parts through signs in the assembly tray.
  - All the signs are known to the robot
- Environment feature
  - There is only one robot in the environment
  - All parts are available in its container.

4.3.1.6 Test Procedures

1. A robot is presented with a tray marked with different signs representing different parts required for the assembly process.
2. The robot acknowledge the required parts through sending a message to the referee box.
3. The robot collects one of the required parts and delivered it to the assembly tray.
4. For each delivered part, the robot inform the referee box.

4.3.1.7 Rules

- Timing restriction.
- The tray can be presented to the robot based on the team preference.
- The robot can bring the tray through out the test, however the result is only valid when the tray is safely return to the user.

4.3.1.8 Acquisition and Logging of Measurements

Record status on:

- Unknown signs.
- Empty containers.
- Successfully collected parts.

4.3.1.9 Scoring

- Correctly understanding a sign: 50 points
- Incorrectly understanding a sign: -20 points
- Successfully collecting one part: 100 points
- Report status on an unknown signs: 20 points
- Report status on an empty containers: 20 points

- Time points (in seconds): (time limit - test duration) * 5 points.

The scoring apply to the relevant parts during each test. For example, reporting an empty containers for parts which are not required result in 0 points. Time points is added only when the task is completed.

4.3.2 Delivery Test

This test is derived from the description in Section 3.3.2.1. In this test, the robot task is to locate and deliver an object to the the designated workstation.

4.3.2.1 Mini-Story of the Test

4.3.2.2 Functionality Benchmarked by the Test

- Object detection.
- Object grasping.

4.3.2.3 Benchmark Metrics

4.3.2.4 Benchmark Scenario

- Environment 3.1.
- Robot 3.2.

4.3.2.5 Feature Variation

The test requires the robot to have object detection capability, grasping and mobility.

Complexity I

- Task features:
  - The object is delivered through conveyor belt.
  - There are unknown objects.

Complexity II

- Task features:
  - The object is positioned somewhere in a predefine area.
  - There are unknown objects.

Complexity III

- Task features:
  - The object is positioned somewhere in a predefine area.
  - All the objects are known to the robot.

4.3.2.6 Test Procedures

The test is performed with several objects. For each object, the test iterates the following steps:

1. The robot is informed by the referee box regarding an incoming object and its delivery destination (one of the available workstations or containers).

2. The robot proceed with finding the object and inform referee box when the object is detected.

3. After successfully grasping the object, the robot proceed with delivering the object to its delivery destination.

The number of iteration depends on how many successful delivery.

4.3.2.7 Rules

- Timing restriction (apply only for limiting the test duration).

- The object and its delivery destination differs for each iteration.

- Same object can be tested only when the robot had performed successful delivery for all available objects.

4.3.2.8 Acquisition and Logging of Measurements

Record status on:

- Successful object detection.

- Successfully delivered object.

4.3.2.9 Scoring

- Successful known object detection: 50 points

- Successful unknown object detection: 50 points

- Successful known object grasping: 50 points

- Successful unknown object grasping: 70 points

- Successful delivery: 20 points

For each repetition of one object, the points collected is half of the original points. For example, successful object detection will result in 25 point for the second iteration of the same object.

4.4 Technical Challenges

4.4.1 Tool handling

In this technical challenges, the teams are invited to presents their solution for handling industrial tools such as scissors, hammers, screwdrivers or electrical drills. This challenge includes the ability to grasp and use the tools accordingly. The solution will be evaluated based on its mechanical design, autonomous level and robustness.

Chapter 5

RoCKIn@Work Organization

5.1 RoCKIn@Work Management

The management structure of RoCKIn@Work has been divided into three committees, namely *Executive Committee*, *Technical Committee* and the *Organization Committee*. The rolls and responsibilities of those committees are described in the following paragraphs.

5.1.1 RoCKIn@Work Executive Committee

The Executive Committee (EC) is represented by the coordinators of each RoCKIn partner related to the respective activity area. The committee is mainly responsible for the overall coordination of RoCKIn@Work activities and especially for dissemination in the scientific community.

□ add further responsibilities as needed □ GKK:

- Daniele Nardi (Sapienza Università di Roma, Italy)
- Pedro Lima (Instituto Superior Técnico, Portugal)
- Gerhard Kraetzschmar (Bonn-Rhein-Sieg University, Germany)
- Matteo Matteucci (Politecnico di Milano, Italy)
- Rainer Bischoff (KUKA Laboratories GmbH, Germany)

5.1.2 RoCKIn@Work Technical Committee

The Technical Committee (TC) is responsible for the rules of the league. Each member of the committee is involved in maintaining and improving the current rule set and also in the adherence of these rules. Other responsibilities include the qualification of teams, general technical issue within the league as well as resolving any conflicts inside the league during a ongoing competition. The members of the committee are further responsible for maintaining the RoCKIn@Work Infrastructure described in Section 5.2.

The Technical Committee currently consists of the following members:

- Jakob Berghofer (KUKA Laboratories GmbH, Germany)
- Rhama Dwiputra (Bonn-Rhein-Sieg University, Germany)
- Aamir Ahmad (Instituto Superior Técnico, Portugal)
- Matteo Matteucci (Politecnico di Milano, Italy) □ To be confirmed. □

This committee can also include members of the Executive Committee.

5.1.3 RoCKIn@Work Organization Committee

The Organizing Committee (OC) is responsible for the actual implementation of the competition, i.e. providing everything what is required to perform the various tests. Specifically, this means providing setting up the test arena(s), providing any kind of objects (e.g. manipulation objects), scheduling the tests, assigning and instructing referees, recording and publishing (intermediate) competition results and any other kind of management and advertisement duties before, during and after the competition.

The Organizing Committee currently consists of the following members:

- **Chair:** Pedro Lima (Instituto Superior Técnico, Portugal)
- Frederik Hegger (Bonn-Rhein-Sieg University, Germany)
- Luca Iocchi (Sapienza Università di Roma, Italy)
- NN, (Politecnico di Milano, Italy) ☐ To be determined. ☐
- NN, (InnoCentive, U.K.) ☐ To be determined. ☐

This committee can also include members of the Executive and Technical Committee.

5.2 RoCKIn@Work Infrastructure

5.2.1 RoCKIn@Work Web Page

The official RoCKIn@Work website can be reached at

`http://rockinrobotchallenge.eu/work.php` ☐ To be confirmed. ☐

On those web pages, teams can find introductory information about the league itself as well as relevant information about upcoming events, the most recent version of the rulebook, videos and pictures of past competitions and links to further resources like the official mailing list or wiki.

5.2.2 RoCKIn@Work Mailing List

The official RoCKIn@Work mailing list maintained by the league is as follows

`rockin-at-work@rockinrobotchallenge.eu` ☐ To be confirmed. ☐

Anyone can subscribe by using the following subscription page.

`http://rockinrobotchallenge.eu/mailman/listinfo/rockin-at-work` ☐ To be confirmed. ☐

Every subscriber is requested to register either with an email address which already encodes the real name or alternatively specify it in the provided field at the subscription page. In order to prevent the mailing list from spammers, this mailing list is moderated.

The mailing list will be used for any kind of official announcement, e.g. upcoming RoCKIn@Work competitions, rule changes, registration deadlines or infrastructure changes. Teams are also welcome to raise any kind of question regarding the league on this list.

5.3 RoCKIn@Work Competition Organization

5.3.1 Qualification and Registration

Participation in RoCKIn@Work requires successfully passing a qualification procedure. This procedure is to ensure a well-organized competition event and the safety of participants. Depending on constraints imposed by a particular site or the number of teams interested to participate, it may not be possible to admit all interested teams to the competition.

All teams that intend to participate at the competition have to perform the following steps: ☐ To be confirmed. ☐

1. Preregistration (may be optional; currently by send an email to the TC)
2. Submission of qualification material (e.g. team description paper)
3. Final registration (qualified teams only)

5.3.1.1 Pre-Registration

All teams that intend to participate at the competition have to perform preregistration process using the following registration website:

`http://rockinrobotchallenge.eu/atwork-preregistration.php` ☐ To be confirmed. ☐

A team must provide the following information during the preregistration process:

- Team name
- Affiliation
- Country
- Team website (optional)
- Team leader name
- Team leader email address

This step can be considered as an *Intention of Participation* declaration and serves as planning basis for the Organizing Committee.

5.3.1.2 Qualification

The qualification process serves a dual purpose: It should allow the Technical Committee to assess the safety of the robots a team intents to bring to a competition, and it should allow to rank teams according to a set of evaluation criteria in order to select the most promising teams for a competition, if not all interested teams can be permitted. The TC will select the qualified teams according to the qualification material provided by the teams.

The evaluation criteria will include: ☐ To be confirmed. ☐

- Team description paper
- Team web site
- Relevant scientific contribution/publications
- Professional quality of robot and software
- Novelty of approach

- Relevance to industrial mobile robotics
- Performance in other competitions
- Contribution to RoCKIn@Work league (e.g. by organization of events or provision and sharing of knowledge)

The Team Description Paper (TDP) is a central element of the qualification process and has to be provided by each team as part of the qualification process. The TDP should at least contain the following information in the author/title section of the paper:

- Name of the team (title)
- Team members (authors), including the team leader
- Link to the team web site
- Contact information

The body of the TDP should contain information on the following: focus of research/research interests:

- Description of the hardware, including an image of the robot(s)
- Description of the software, esp. the functional and software architectures
- Innovative technology (if any)
- Reusability of the system or parts thereof
- Applicability and relevance to industrial tasks

The team description paper should cover in detail the technical and scientific approach, while the team web site should be designed for a broader audience. Both the web site and the TDP have to be written in English.

The length of the team description paper is limited to 6 pages and has to be to submitted in the IEEE Conference Proceedings format[1].

5.3.1.3 Registration

Only if team has passed the qualification procedure successfully it is allowed to register officially for the competition and has to provide additional information e.g. the exact number of team members. Further information about the registration procedure will be communicated through the mailing list mentioned in Section 5.2.2.

The number of people to register per team is not limited, but during the competition the organizers will provide space only for X persons to work at tables in the team area.

☐ Fill number X ☐

During the final registration, each team has to designate one member as team leader. A change of the team leader must be communicated to the Organizing Committee.

[1] http://www.ieee.org/conferences_events/conferences/publishing/templates.html

5.3.2 Setup and Schedule

□ The schedule is still under discussion. □

Setup: For the arrival, setup, and preparation of teams participating in the competition, the following procedures apply:

- A first draft of the rulebook will be made public X weeks before the actual event by the the TC.

  □ Fill number X □

- the final version of the rulebook will be made public X weeks before the actual event by the the TC.

  □ Fill number X □

- The competition side will be divided into a competition and a team area.
- The competition area consists of one or more testbeds (the arena) and is open for public.
- The arena must be kept clean and in a representable condition all the time.
- The team area is a dedicated area only for team members, no public access here.
- Each team will be assigned to a designated area with tables and chairs (based on the number of team members), with power sockets, a shelf internet connection and a reasonable area to park their robot and other equipment.

□ Revise the list below. Most of it does not belong here. This info given here should concern only on-site setup procedures for teams, e.g. mention how much time there is for unpacking robots, setting them up, and testing in the arena. □

Schedule: For the scheduling of particular stages, tests, and technical challenges of the competition the following applies:

- The exact schedule of task-/functionality tests will be announced one week before the actual competition by the OC on both, the website and the mailing list.
- In order to avoid to much "traffic" inside the testbed, an additional schedule only for test slots will be established on site by the OC.
- A set of test slots will be assigned to each team between the official test slots, where a team has exclusive access to the testbed without any other team/robot inside the arena.

5.3.3 Competition Execution

- Referees will be determined by the OC out of the group of team leaders and TCs.
- The referees ensure the correct execution of a test, are in charge of keeping the time and counting the scores.

- In case of any dangerous situation the referees are allowed to immediately stop a run and trigger the emergency stop functionality of the respective robot.
- The official language for all kind of communication within the league is English (e.g. team leader meetings, announcements, schedule, etc.)
- The order in which the teams have to perform a particular test will be determined by a draw through the OC.
- The order will be announced one the day before the actual test.
- No team members or other persons are allowed to be in the arena during an official test (only if the rulebook explicitly allows this).
- Regular team leader meetings (every day) will be organized and announced by the TC/OC during the competition in order to discuss open issues for upcoming tests.

5.3.4 Competition Evaluation and Assessment

5.3.4.1 Measurements Recording

5.3.4.2 Scoring

For each test the calculation of scores is defined individually (e.g. Section 4.3.1.9), comprising points for achieving certain subtasks, points for winning a run, negative points for reducing the difficulty, and penalty points. ☐ To be confirmed. ☐

5.3.5 Awards

- A ranking is established based on the scoring achieved in the particular task-/functionality tests described in Section 4.3. According to the ranking there will be a 1st, 2nd and 3rd place.
- Beside the ranking awards there are additional awards for the Technical Challenges specified in Section 4.4 (only one per challenge) and for the Open Demonstrations (only one per demonstration). ☐ To be confirmed. ☐

5.3.6 Post-Competition Procedures and Workshop

☐ To be decided ☐

Appendix A

RoCKIn@Work Environment Details

In this appendix, we provide further details on the environment used in the RoCKIn-@Work scenario.

□ The focus is on specifying first items that need to be known for preparing the construction of scenario environment and for building suitable simulators. Thus, we initially focus on items like walls, floors, ceilings, shelves, containers, workstations and infrastructure. For other items, such as objects (nuts, bolts, screws, angles, tools, etc.) to be manipulated, there exists — and will intentionally remain — a wide variability of which objects may be ultimately used in a particular environment. We will detail such objects in future versions of this document. □

Appendix B

Variations of the Environment

The role of this appendix is to indicate the variability of the scenario by illustrating possible, controllable variations of the environment.

B.1 Alternative Environment Layouts

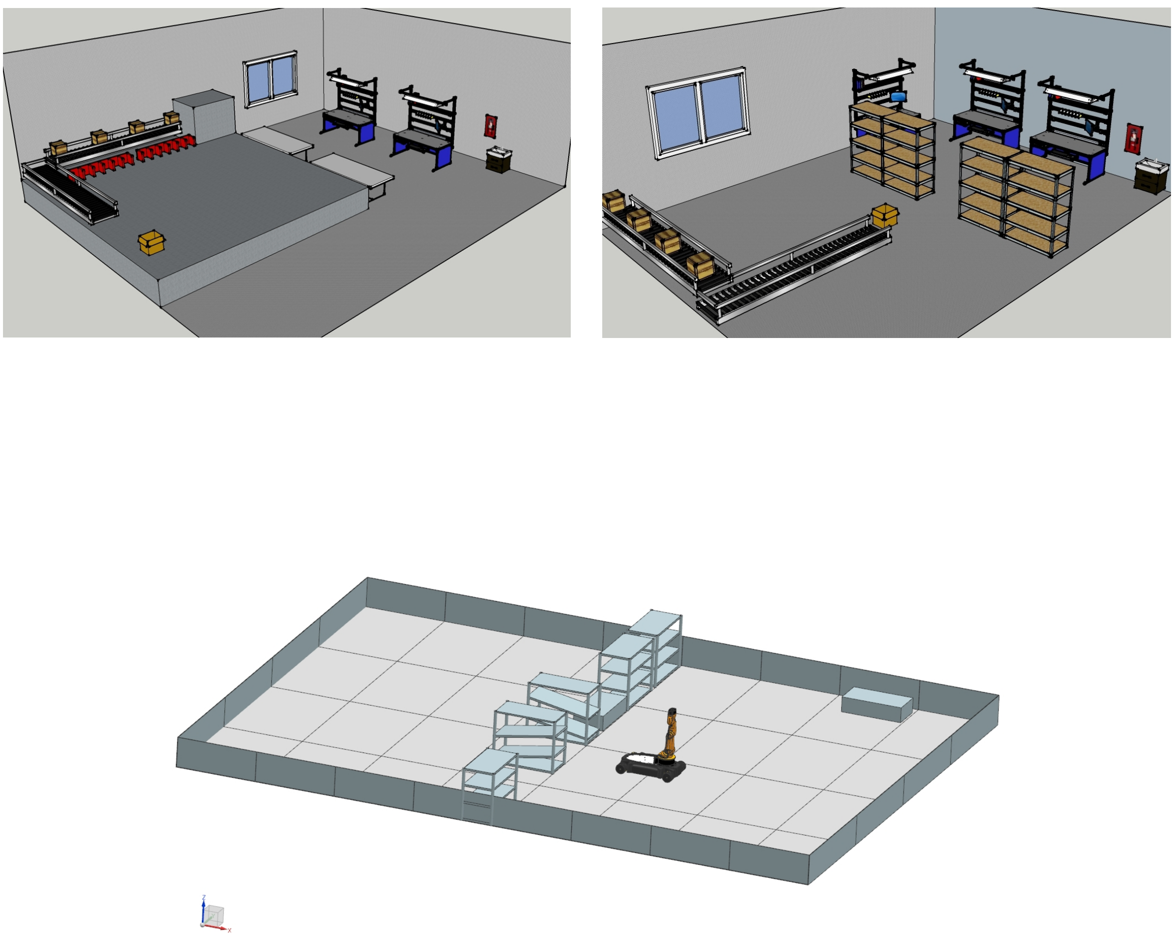


Figure B.1: Draft of construction, to be discussed; floorplanes have the size of an stndardized Euro palette (1200x800mm);

□ This appendix may be extended in future revisions. □

Appendix C

RoCKIn@Work Scenario Construction

☐ This section is planned for inclusion in revision 2.0 ☐

Appendix D

The Benchmarking Infrastructure for RoCKIn@Work

□ This section is planned for inclusion in revision 2.0 □

Appendix E

Library of Functionalities Deemed Useful for RoCKIn@Work

□ This section is planned for inclusion in revision 2.0 □

Bibliography

[1] A. Ahmad, I. Awaad, F. Amigoni, J. Berghofer, R. Bischoff, A. Bonarini, R. Dwiputra, G. Fontana, F. Hegger, N. Hochgeschwender, L. Iocchi, G. Kraetzschmar, P. Lima, M. Matteucci, D. Nardi, V. Schaffionati, and S. Schneider, "D1.1 - specification of general features of scenarios and robots for benchmarking through competitions," The RoCKIn Project (FP7-ICT-601012 grant agreement number 601012)," Public Deliverable, 2013.

[2] Subaru-cargo-photo_2.jpg. [Online]. Available: http://www.plasticpals.com/http://www.plasticpals.com/wp-content/uploads//2009/10/Subaru-cargo-photo_2.jpg

[3] Anonymous. Illuminance. [Online]. Available: www.wikipedia.org/illuminance

List of Figures

List of Scenario Specifications

List of Scenario Constraints

Todo list