DARKO
Dynamic Agile Production Robots That Learn and Optimise Knowledge and Operations

H2020-ICT-2020-2 Grant agreement no: 101017274

DELIVERABLE 6.4
Safety for wheeled mobile robots

Dissemination Level: PUBLIC

Due date: month 43 (July 2024)

Deliverable type: Report

Lead beneficiary: TUM

Contents

1 Introduction **3**

2 Theoretical Framework and Implementation Details of Safety Unit **3**
2.1 Hardware and software system setup 4
2.2 vSMU operation modes 5
2.3 vSMU algorithm 6
2.4 Standardization overview 7

3 Human Robot Spatial Interaction **14**

4 Perception **14**
4.1 Human perception stack 14

5 Motion planning Architecture **16**
5.1 Architectural Layers 17

6 Robot intent **18**

7 Planning and Control Framework for Manipulator **19**
7.1 Control algorithm 19
7.2 Human-Like motion planning algorithm 20
7.3 Integration with vSMU 22

8 Whole body Impedance controller **22**

9 Scenario testing **24**

10 Conclusions **27**

1 Introduction

Novel development for industrial robotics are continuously introducing new challenges for co-existence of humans and machines in industrial settings. Although, general tendency is achieving high level of automatization, many argue that the intermediate step of a collaboration between human and robots is unavoidable. Thus, safety is a crucial aspect of development of every robotic based system. In comparison to fully automatized production, human-in-the-loop introduces additional challenges and safety margins to be considered, as the price for the robot repair can be estimated - although highly costly in many situations, but price tag can't be set for the human life or injury.

First steps towards considering a possibility of safe interaction in manipulation robotics are coined by foundational works such as [1], where one can see injury analysis and gain better understanding on safe robot speed for human-robot interaction in manipulation. Since then, collaborative robots, or famously called co-bots are becoming an integral part of the industrial cycles, spanning from factory production markets up to a consumer oriented applications such as retail or hospitality businesses, etc. To tackle challenges of Physical human-robot interaction (pHRI) and ensure systematic approach towards avoidance of undesired and potentially dangerous situations, including harmful collisions, one can refer to the latest industry standards (e.g. ISO10218/TS15066 for industrial robots). However, many aspects remain undefined because of on-gong development of newest standards to follow industrial and research developments in this area. Here, where possible, a line of conformity towards existing and latest standards will be made. Otherwise, novel capabilities, requirements will be highlighted with respect to safety of Darko or similar mobile robotic solution. Its worth noting, large step in direction of standardisation of relevant sector was recent introduction of standards ANSI R15.08 Part 1 and ANSI R15.08 Part 2.

At the beginning, general overview of the platform is given, highlighting various aspects related to integration of necessary components. Since the utilized platform is a modification of industrial robot, it already contains an important set of hardware built-in features. In addition to that, the focus of this contribution are in particular safety aspects for tasks related to manipulation and mobile platform control on autonomous or collaboration/pHRI level. The safety is deployed as one unit for the whole system - Vehicle Safe Motion Unit (vSMU). Where, based on the various available inputs, it provides speed limits. As such, the collision in worst case can not exceed the safety limits set by the standard or other safety mappings chosen to be respected. It will be clarified about safety mapping used for this scenario. Further, an outlook is provided on various components available on the robot that are utilizing these safety margins. A simple mockup-up scenario is presented as a showcase of some of the capabilities.

2 Theoretical Framework and Implementation Details of Safety Unit

A unified approach for ensuring human safety during pHRI of stationary robot arms relies on an injury data-driven safety framework termed the Safe Motion Unit (SMU), introduced in the previous work [2]. It relies on the data of human relative position to the robot to introduce speed limits either in Cartesian space or at joint level of the manipulator. Main criteria for establishing the metric of the injury is relation between the reflected mass of the robot in the place characterised by most likelihood for collision, termed Point of Interest (PoI). This highly reduces computation time, since inspection of the whole robot mesh is not necessary. Further, one can increase number of samples to be checked for better resolution. Although, in practice it was sufficient to take several points at the critical places on the robot.

Reflected mass of a point is a measure of the mass as perceived externally at that point, in the Cartesian unit direction u:

$$m_u(q) = 1/(u^\intercal \Lambda_v(q)^{-1} u)^{-1} \tag{1}$$

where Λ_v is an operational space kinetic energy matrix. Its relation to dynamic parameters of the robot is a clear indication that once the mobile arm base is not stationary (i.e. arm is mounted on the mobile robot), the reflected mass of the PoI will be influenced by the platform inertia as well. Inertia dependencies can be combined in a rather intuitive manner (from the decoupled components) through modular approach towards derivation of generalized Mass matrix, as shown in Figure 1.

One simplification taken into account for the safety unit is that the base is modeled as a floating base. Indeed, thats a valid simplification for the used platform since it accommodates Omni-wheel drives (characterized by holonomic motion).

$$M = \begin{bmatrix} \boldsymbol{M_v}' & \boldsymbol{M_{va}} \\ \boldsymbol{M_{va}^\intercal} & \boldsymbol{M_a} \end{bmatrix}$$

Figure 1: Modularity of mass matrix for robot consisting of mobile platform (vehicle) and arm. Mass matrix corresponding to vehicle marked with v and to arm marked with a. [3]

Based on the observed relative human-robot distance and their velocity information vSMU is setting the limit for robot maximum speed. Correlation between the motion speed and safety is based on a rich injury database for various impact curvatures - the vSMU shapes the robot motion such that the impact energy of possible dynamic collisions will not lead to serious injury in the worst case.

To summarize, vSMU for input requires:

- Massmatrix
- POIs
- Relative human position, speed
- Robot frame transformations
- Dynamic and kinematic data of the robot
- An injury database for characterized POIs

2.1 Hardware and software system setup

There were several iterations of the system for testing the vSMU algorithm:

- First tests were performed in ROS-based Gazebo simulation environment featuring integrated industrial mobile platform and Franka Emika manipulator as shown on Figure 2.
- Next iteration (in parallel of further effort to include more features in simulation) was done on the initial hardware platform featuring Omni-wheeled Mobile Platform and Franka Emika Panda Manipulator.

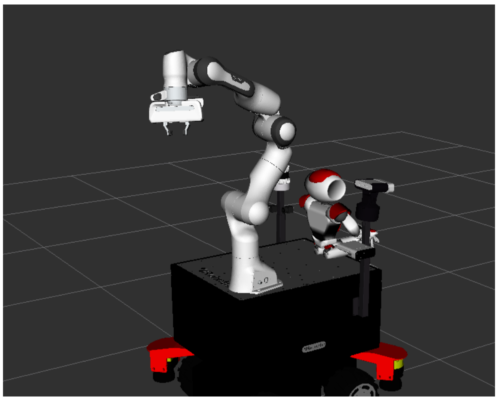

Figure 2: Robot in the Gazebo simulation featuring Mobile platform and Franka Emika manipulator as an combined system.

For the integration purposes, the efforts were placed on refining all the features together with partners, keeping in mind compatibility and integration aspects for the future implementation within in-house manipulator. As an example of the compatibility between current development and the future requirements, for general low-level controller for the platform control, an Whole Body Impedance controller is adopted, as explained in Section 8. Control bandwidth limitation of the SEA actuator, have rather positive characteristic for safety due to the high bandwidth for compliance (mechanical spring present in series) - thus ensuring better safety performance for manipulation (in relatively low speed motion, in comparison to the throwing mode speeds).

Further, we don't make a distinction between the arm only, or the base only, but rather refer to the whole integrated system as a Robot.

2.2 vSMU operation modes

It is important to distinguish several safety operational modes in general [4]:

- Autonomous mode: no shared workspace
- Coexistence: human and the robot share the workspace, but do not have a shared task purpose
- Cooperation: coexistance with a shared task purpose
- Collaboration: cooperation that allows contact between the human partner and the robot

Due to the project use-case, the focus is rather set on the first 3 modes (Autonomous, Coexistance and Cooperation). This can be seen as a simplification due to the rather different approach for safety in Collaboration (in contrast to perceiving relative velocities for the human to robot in general - as between two unit masses, here, one should rather focus on velocities of individual parts).

Within previously mentioned scenario, it is important to distinguish different situations:

- Robot and human are moving towards each other
- Both are moving in the same direction with the robot being behind the human and moving faster

- Both are moving in the same direction but the human is behind the robot

For case 3 it is expected that human will not try to intentionally cause the collision with the robot, thus relative task velocity stays unaltered while the direction stays consistent. For case 2, V_{smu} is expected to actively control the maximum speed of the robot, while for the metric, its rather that the absolute speeds are used (robot not taking advantage of additional speed margin available due to the human going away). Regarding case 1, the relative speeds are taken into consideration, otherwise the derived injury metric would not be valid. This approach has been introduced as a trade-off between safety and task execution efficiency.

In seminal paper for the vSMU modes [4], several approaches towards evaluation of relative speeds are taken into account. Among them best trade-off between control effort and efficiency can be observed for the V_{motion}, where relative velocity between Robot and human is taken into account as projected in the direction of the PoI's motion. One can also encode PoI characteristic by specifying pointing direction (e.g. direction at which the PoI is for example facing the sharper corner, etc.). Further, this direction can be taken for evaluation of relative speeds, for consideration of previously mentioned PoI characteristic.

2.3 vSMU algorithm

Schematic representation available in Fig. 3. A more detailed procedural description of vSMU itself can be found in Algorithm 1.

Algorithm 1 The vehicle-extended safe motion unit (vSMU) algorithm for ensuring human-safe task trajectory control for mobile manipulators

---

Assign a set of POIs encoding surface curvatures at key points on both the base vehicle and manipulator arm structures. Alongside position, directional normalized (unit) Cartesian vector u is defined perpendicular to the object surface in the Cartesian world frame (characterising PoI, in this case with most dangerous direction).

Obtain the velocity of each POI using robot kinematics, for the case of requested mobile manipulator's end-effector velocity.

Calculate the reflected mass of the joined manipulator and vehicle system at each POI, considering the impact direction u, oriented towards the human:

$$m_{POI} = 1/(c^{\intercal} \Lambda_{v,POI}^{-1} u) \tag{2}$$

Find the maximum permissible velocity v_{max} that is bio-mechanically safe for the corresponding reflected mass of POI from the suitable safety curves of the pain/injury

Obtaining the human velocity consider the cases:

if human behind the robot **then**

  use v_{task}

else if human in front of the robot **then**

  apply safety margin

else if human moving towards the robot **then**

  update v_{max} by subtracting the value of the v_{human}

end if

Scale v_{max} with the projection of arm POI velocity in the impact u direction to get the new POI velocity $v_{POIsafe}$

Calculate the new, safe velocity limit of the platform v_{safe}. It is supplied to the control as the new desired task velocity.

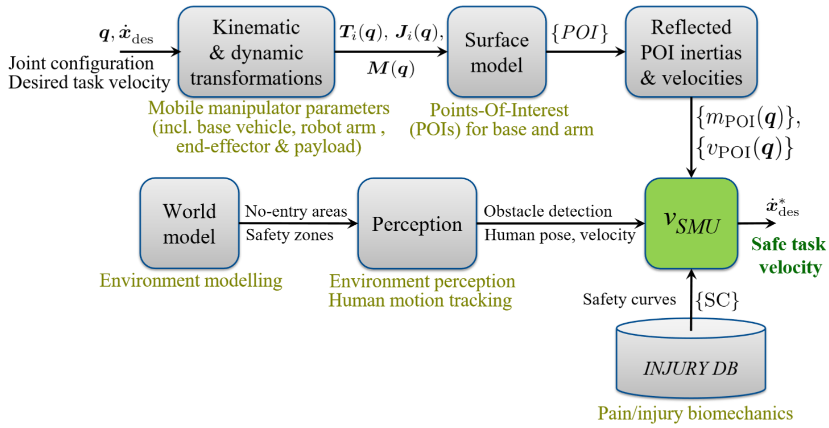

Figure 3: Schematic representation of vSMU module integration and key working principles.

2.4 Standardization overview

Recent industrial robotics developments posed need for the safety standardization. Embodiment of the aforementioned is the relatively recent release of the Eurpoean standards ISO 10218-1 and ISO 10218-2, as well as American ANSI/RIA R15.08-1 and ANSI/RIA R15.08-2. The novel aspect of the ANSI standard is introduction of Industrial Mobile Robots (IMR), that is highly applicable to the usecase presented within this project framework.

Reference from relevant Standard	**Level of maturity on Darko platform (comments when relevant)**
ref. ANSI/RIA R15.08-1-2020: 5.1.7.1 Navigation and control General requirements for navigation functions Navigation can be a safety function. An IMR may re-plan its trajectory for the purpose of: — obstacle avoidance (e.g., changing direction and/or speed as new information about the environment becomes available); — simultaneous motion (e.g., coordinating motions of the mobile platform and the manipulator for IMRs Type C); and/or — collision avoidance. Such trajectory re-planning shall not introduce new hazards.	Robot, as referenced in Section 5, is capable of dynamic obstacle avoidance. For the purpose of collision avoidance, it takes into account the maximum speed provided by the safety unit. Simultaneous motion algorithm is present for the mobile base and arm - ref Section 8, but it has only been tested in simulation so far.
ref. ANSI/RIA R15.08-1-2020: 5.1.7.2 Collision avoidance A collision avoidance safety function shall either prevent a collision by stopping the IMR, or reduce the severity of a collision when contact cannot be avoided (e.g., insufficient time to stop after an obstacle is detected). NOTE: If an established means for testing the severity reduction of collisions is not available, ASTM F3265-17 provides example test methods for this purpose. **5.2.7.2 Collision avoidance** In addition to the requirements of Clause 5.1.7.2, the following shall apply. For IMR configurations in which component(s) of the attachment(s) (including manipulator(s)) can extend beyond the base footprint of the mobile platform, the IMR shall be capable of monitoring the position of such component(s) and adjusting its base footprint accordingly, to include the component(s) in the IMR's collision avoidance function(s).	Safety function takes into account the whole robot kinematics and based on the injury data analysis is providing safety speed limit for the injury prevention.

ref. ANSI/RIA R15.08-1-2020: 5.1.7.3 Velocity limiting The IMR may have the capability of velocity limiting, such that risks from tip-over hazards are eliminated or reduced. Velocity of part(s) of the IMR may exceed the velocity limits if necessary to prevent tip-over hazards and only if no additional hazards are introduced. If the IMR is capable of velocity limiting, its velocity limit settings shall be configurable by authorized individuals. NOTE: Velocity can include, e.g., linear velocity, angular velocity, rotational velocity. **5.2.7.3 Velocity limiting** In addition to the requirements in Clause 5.1.7.3, the following shall apply. For IMR configurations supporting simultaneous motions of the attachment(s) (including manipulator(s)) and the mobile platform, the IMR shall be capable of modulating these velocities. If the active attachment and the mobile platform are controlled independently, and if synchronized motion is required, one of the following shall be provided: a) both components shall assume the worst-case scenario based on the maximum velocity of the other; or b) both components shall provide bidirectional signals such that the velocity and direction of travel may be reported to and/or controlled by the other. If the IMR can tow or lift a payload, the mobile platform shall have velocity control commensurate with the stability profile of a lifted or towed payload.	Since the Safety Motion Unit calculations are based on the dynamical parameters, its easy to calculate and take into account any external payload. Tip-over is not a hazard for the used platform so far.

ref. ANSI/RIA R15.08-1-2020: 5.2.7.4 Control of simultaneous motion When an IMR is configured to enable simultaneous motion of active attachment(s) and mobile platform, such simultaneous motion shall not create additional hazards. Simultaneous motion of attachment(s) and mobile platform can be either coordinated or independent. NOTE 1: Coordinated simultaneous motion can be achieved by (1) the attachment being capable of sending its kinematic data to the mobile platform and receiving control inputs from the mobile platform; (2) the mobile platform being capable of sending positional data to the attachment and receiving control inputs from the attachment; or (3) both attachment and mobile platform being controlled by an intermediate controller, based on positional and/or kinematic data provided by both the attachment and platform. **ref. ANSI/A3 R15.08-2-2023: 5.2.3.3 End-effectors for IMR Type C** End-effectors shall be integrated in such a way that the dynamic forces created by the end-effector and its payload do not compromise the stability of the IMR. IMR systems may operate with simultaneous motion of the mobile platform and active attachment(s) when the IMRs comply with ANSI/RIA R15.08-1-2020, Clause **5.2.7.4.** If simultaneous motion results in risks that are not reduced in accordance with Clause 4, then the active attachment(s) shall be stowed in a stow position when the mobile platform(s) is(are) in motion, in accordance with ANSI/RIA R15.08-1-2020, Clause **5.2.9.2.** NOTE: Simultaneous motion of mobile platform and attachment(s) can be either coordinated or independent; for more information, see ANSI/RIA R15.08-1-2020, Clause 5.2.7.4.	Whole body impedance controller has been tested in simulation as a unified framework for controlling the mobile platform and attachment simultaneously. Refer to Section 8 for details on mathematical implementation.

ref. ANSI/RIA R15.08-1-2020: 5.2.18 Collaborative operations IMRs Type C are permitted limited utility in collaborative operations in accordance with RIA TR R15.606, which describes safety requirements for collaborative operation of robot systems that incorporate manipulators (e.g., power and force limits, or safe separation distances). Where collaborative operation includes expected physical contact, hazardous contact shall be prevented between any part of an IMR (including active attachments) and a person, as determined by the risk assessment and in accordance with the requirements of power and force limiting in RIA TR R15.606. Where collaborative operation does not include physical contact, the requirements for contact prevention, as described in the section on speed and separation monitoring in RIA TR R15.606, shall be applied to the motions of both the mobile platform and any active attachments for any IMR capable of such collaborative operations. NOTE: The use of a manipulator intended for collaborative applications is not sufficient, by itself, for the resulting IMR Type C to be considered in accordance with RIA TR R15.606. The evaluation of the IMR's speed and position shall include the speed and direction of motion of all parts of the IMR (including the mobile platform and any active attachments).	Collaborative applications are not part of the direct use-case for the robot, however in the operating environment where an active presence of humans is expected, this can be an important perspective. It is yet to be further refined, but the nature of the controller (Impedance based) and availability of the torque sensing in the links can provide much-needed tools for human-robot close interaction.
ref. ANSI/RIA R15.08-1-2020: C.2.2.1 Ranges for the estimation of the severity of harm **S3 serious** Injuries which require treatment by a medical practitioner but do not lead to permanent impairment; or, injuries which lead to the loss or permanent damage of parts of the human body (but not total loss) with reversible medical condition. **C.2.6.1 Ranges of the possibility of avoidance or limiting the harm** **Avoidable:** Some conditions that allow the avoidance of harm are present but not always (e.g., as skilled workers, slow movements, infrequent intervention, low-complexity processes, and no sudden or unexpected movements with high acceleration).	Mentioned severity of harm and ranges of possibility of avoidance (which are taken as they are considered to best correspond to the scenario of the project) indicate that the system safety should not be taken for granted. Safety unit is able to counteract and prevent hazardous situations.
ref. ANSI/A3 R15.08-2-2023: 3.1.1.8.2 Monitored-speed safety function - safety function that limits the speed to a configured value	Seams, that speed safety function, as explained in the standard, is rather focused on reducing the speed to a specific configured value. It's important to mention that in our case, it's also responsible for the speed analysis for providing safety limits and trying not to introduce unnecessary sacrifices of productivity.

ref. ANSI/A3 R15.08-2-2023: 4.2.1 General IMRs used for integration into an IMR system shall comply with ANSI/RIA R15.08-1. The hazards associated with the IMR system and IMR application(s) (Annex A, Table A.1, sections 1 - 8 and 10) shall be considered before the hazards of the DOE (Annex A, Table A.1, section 9). **The risk assessment shall consider:** a) IMR systems, their component part(s) and their payloads are capable of high-energy movement through a large space (e.g., due to intended movement or falling); b) difficulty of predicting the initiation of movement and the trajectories of the various IMRs in the system, since they can vary; EXAMPLE: Paths are re-planned dynamically based on obstacle avoidance and destination. c) the working space of the IMR and its payload can overlap with hazard zones of other machines and task zones of personnel; d) IMR(s) interfacing with station(s); e) for IMR Type C, the end-effector selection and design as it interacts with workpieces and other parts of the IMR application;	With the respect to the mentioned hazards, it is important to note that the strategies for safety should be rather created as a functions, instead of hard-coded limits, etc. For example, mentioned correlation between speed, perceived mass and injuries is done in order to have an function for introducing adequate limits online.
ref. ANSI/A3 R15.08-2-2023: 5.2 Design of IMR systems **5.2.1 General** Consideration shall be given to items that affect the safety of the system as it is deployed into its DOE, and risks shall be reduced. These items can include: a) modifications to existing equipment interacting with IMRs, b) modifications to the facility (addition of guidepaths, floor markings, etc.). If required for risk reduction, IMR systems shall have a means to provide status (e.g., “in station”, “ready for interaction”) to affected persons and/or peripheral equipment. The information provided may be application specific. Any risks associated with an attachment, including an end-effector in the case of an IMR Type C, and/or workpiece or payload, shall be evaluated and reduced in accordance with clause 4.	For this particular scenario, it is worth mentioning that part of the deployed system is the small Humanoid robot located on the mobile platform - near to the manipulator. It is responsible for providing various information about the robot state and its intent. More details on it is provided in Section 6.

ref. ANSI/A3 R15.08-2-2023: 5.6.3 Safety-related parts of control systems Safety-related parts of control systems (SRP/CS) shall be designed according to ISO 13849-1 or IEC 62061. **ref. ANSI/A3 R15.08-2-2023: 5.6.5 Performance** Safety-Related Parts of the Control System (SRP/CS) The results of a risk assessment performed on the IMR system and its DOE shall determine the required performance of the SRP/CS for each required safety function according to Annex C. The IMR system shall have the required SRP/CS for each safety function according to the determined performance requirements. The SRP/CS performance of all safety functions shall be provided in the information for use, including specific identification of the criteria used to select the required performance for each safety function.	**From ISO 13849-1:2023:** ISO 13849 is highlighting an important metric - evaluation of required performance level for safety functions. Several risk parameters are identified as relevant for that considered platform and deployment environment: a) S2: serious(normally irreversible injury or death) b) F1: if F1 may be chosen if the accumulated exposure time frequent is not higher than once per 15 min and does not exceed 1/20 of operating time. c) Determination of P based on the 5 factors leads to the P2 selection: - since Use of machine by unskilled person, also means that the unskilled person can interact with the machine or approach it. (category B) - Medium speed event (category B) - Occasionally/rarely possible to escape. (category B) 3 or more B categories that are describing robot setup summarize in overall score of P2 (scarcely possible of avoiding or limiting harm). Thus, previously mentioned classification leads to the required performance level d from the safety device on board according to this standard.

<table>
<tr><td>ref. ANSI/A3 R15.08-2-2023: 5.6.8 Velocity limiting

A velocity limiting safety function shall be provided. This safety function may be provided by the mobile platform, IMR, IMR system, or a combination of a platform and an attachment. If the payload or workpiece could present hazards due to their movement, risk reduction for such hazards shall be provided.
The velocity limiting safety function parameter or limit value shall be configured and implemented to reduce risks in the application. There may be multiple velocity limiting safety functions and/or multiple parameters.
NOTE 1: A zero (0) velocity setting can be used and this can be used with or without a standstill safety function.
If a monitored-speed safety function is provided with the IMR, it shall be configured. NOTE 2: Monitored-speed safety function is relative to the IMR's own coordinates in the deployed operating environment. The cumulative effect of the speeds of multiple IMRs relative to one another is not considered.

ref. ANSI/A3 R15.08-2-2023: 5.11.3.2 Velocity limiting

Velocity limiting for collaborative applications shall comply with 5.6.8. If velocity limiting is used to reduce risks of energy transfer during contact events, motions of relevant parts of the IMR system (e.g., active attachments, held workpieces, payload) shall be limited to a reduced velocity to minimize energy transfer during contact events.</td><td>For safety implemented on board, one can also refer to this part of standard, since it corresponds to the functional requirements of the robot. It is important to note that the safety unit is taking into account both, motion of platform and robot motion.</td></tr>
<tr><td>ref. ANSI/A3 R15.08-2-2023: 5.8.2.2 Automatic mode
In automatic mode, the IMR system shall execute the operations automatically, and the risk reduction measures associated with automatic mode shall be active. Hazardous motion shall be prevented according to 5.9.4, if a stop condition is active.

5.8.2.3 Semi-automatic mode

IMR functions that are available to an operator in semi-automatic mode shall have access restricted to authorized personnel only. Operation of these IMR functions shall require continuous actuation of a hold-to-run or enabling control device.</td><td>Safety unit doesn't distinguish between automatic and semi-automatic mode, since it runs at the low level controller on the robot, thus equally applicable to the both.</td></tr>
<tr><td>ref. ANSI/A3 R15.08-2-2023: 5.11.3.1 Risk reduction

Power and force limiting Risk of collaborative applications shall be reduced in accordance with Clause 4 and ISO/TS 15066.
NOTE 1: RIA TR R15.606 is an identical national adoption of ISO/TS 15066 as a technical report. Technical reports are informative. RIA TR R15.606 explains that the use of “shall” and “should” indicate priority and not requirements.
NOTE 2: Where contact force(s) or pressure(s) exceed limits defined in ISO/TS 15066, Annex A, some examples of technical means to address unacceptable force or pressure and to reduce impact energy transfer include:
— safety functions such as
— velocity limiting of the IMR system,
– power or force limiting of the IMR system, or
– presence-sensing to prevent hazardous contact,</td><td>Due to the abundance of sensors (RGBD cameras, depth cameras, lidars, joint force sensors), as well as their processing (perception, mapping, localization) one can extract information on humans and their intents, obstacles, robot state. Where, through data aggregation and vertical integration of different control components one can ensure safe robot operation.</td></tr>
</table>

Table 1: Cross-reference of developed features with **relevant standards**.

Figure 4: (Left) Image from THÖR dataset. (Right) Risk analysis performed between the selected agent (red) and obstacle (black).

3 Human Robot Spatial Interaction

To additionally increase the efficiency of the safety algorithms and avoid unnecessary velocity reduction, one can utilise the risk evaluation approach studied in [5, 6] as a part of T5.4. In these studies, the risk evaluation is part of a causal inference framework that aims to predict potential human reactions in response to robot actions. This framework can be used to provide human intention predictions in human robot space interactions (HRSI), which can then be input into the vSMU to enhance its performance. As proposed in [5], the risk of collision for a selected agent is evaluated using the Velocity Obstacle (VO) strategy with respect to the closest obstacle (Figure 4 right). The risk is a function of two parameters: d_{OP}, which measures the time available for the selected agent to avoid a collision, and d_{BP}, which indicates the steering effort by the same agent to avoid such collision. This risk is significant only if the closest obstacle is within a certain threshold distance (d_{thres}=1.5m). Its value is computed as follows:

$$\begin{cases} risk = e^{d_{OP}+d_{BP}+v} & d_{obs} \leq d_{thres} \\ risk = e^{v} & else \end{cases} \tag{3}$$

As shown in the equations, the $risk$ depends on the selected agent's velocity v and its proximity to the obstacle d_{obs}.

This risk estimation has been tested using data from a public dataset: THÖR [7] (Fig. 4 left). A detailed explanation of the mentioned approach and how it is exploited in [5, 6] will be presented in the dedicated deliverable D5.3.

4 Perception

4.1 Human perception stack

For the perception of humans around the robot using its onboard sensors, Bosch has developed a modular perception pipeline. This pipeline is described in more detail in DARKO deliverables D2.1 and D2.2. Here, we just briefly summarize the main components.

First, humans are detected around the robot using a robust deep learning-based detector, which estimates their 3D centroid coordinates. For RGB-D data from the Azure Kinect camera on the robot's pan-tilt unit, we use our well-proven RGB-D YOLO approach [8]. Further detection methods on other modalities, including 3D lidar, have been benchmarked in a DARKO publication at the beginning of the project [9]. Current research in WP2

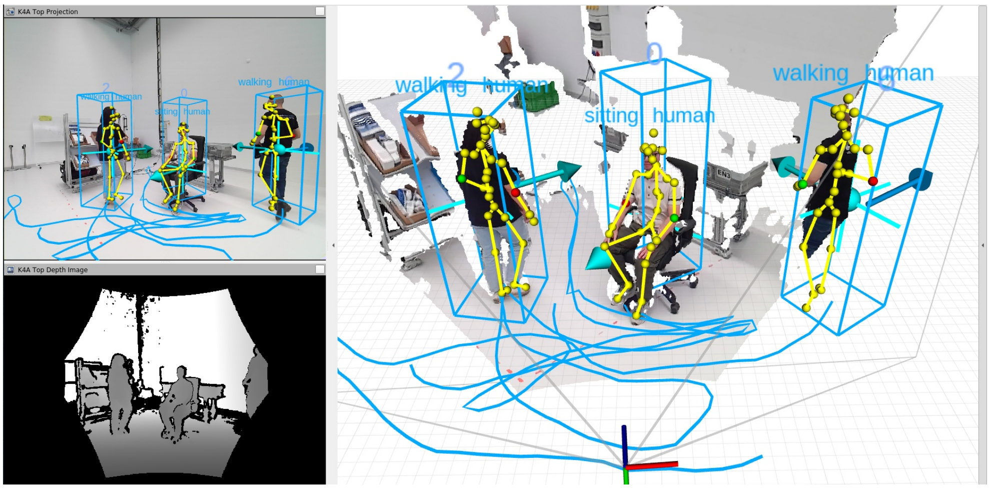

Figure 5: Visualization of the integrated human perception stack that was deployed on the DARKO robot during the MS2 milestone demonstration at ARENA 2036. A single Azure Kinect RGB-D camera was used for human perception during this milestone, its color and depth images are shown on the left. Bounding boxes have a fixed scale and indicate tracked persons, trailing blue lines their past trajectories, blue arrows their velocity vector. Cyan crosses represent oriented 3D centroids from the RGB-D YOLO detector.

(task T2.5) examines how to robustly perceive humans with full 360-degree horizontal coverage around the robot, e.g. using back-to-back fisheye cameras. Such a surround view 3D perception can be essential especially also for safety applications, given that the robot is able to drive not only forwards or backwards, but also sideways due to its omniwheel kinematics. In addition, knowledge not only of the humans' 3D centroids, or their 2D or 3D bounding boxes, may be important, but also more fine-grained information such as their articulated body poses; imagine e.g. a human stretching out the arm in front of the robot, causing a potential collision and resulting injury if undetected. The articulated 3D body pose estimation methods required for this task are also being developed within T2.5, and extended for wide-FOV sensors such as fisheye cameras or 3D lidar.

Temporal association and tracking of humans and their trajectories is achieved using an established nearest-neighbor tracker based upon Kalman filtering [10]. The same association used for detected 3D centroids may also be used to associate fully articulated 3D body poses. One challenge in tracking, especially for real-time safety-relevant applications, is latency. Since detection never takes zero time (spanning at least a few milliseconds, including data preprocessing), a certain latency can always be expected. To still be able to react to the "most current" situation, a joint trajectory and 3D pose prediction approach is being developed in collaboration of DARKO tasks T2.5 and T5.1.

The results from detection, tracking and body pose estimation are fed into a human activity classifier which classifies the human's current activity into a number of pre-defined classes (standing, sitting, walking, lying on the floor etc.). It may optionally be interfaced with object-level semantics (T2.1) to also recognize human-object interactions, e.g. sitting on a chair, or interacting with a computer.

To downstream components, the human perception pipeline provides the following information:

- Detected 3D centroids and 2D/3D bounding boxes
- Tracked human centroid trajectories

- Linear velocities of 3D human centroids (and optionally further body joints)
- Articulated 3D body poses covering around 20 body joints
- Classified activities (standing, sitting, walking, lying on the floor etc.)

An example scene with a visualization of outputs from DARKO's 3D human perception stack is shown in Figure 5.

5 Motion planning Architecture

In this section a high level overview of the planning architecture prototype developed so far in the project is provided. The ambition is to create a system that can handle different contextual cues of the environment, static and dynamic ones, by generating natural, smooth, legible and safe robot motion.

We have developed our architecture according to the following requirements:

- Make use of the several *contextual cues* that the DARKO robot can perceive (as output of the WP2, WP3, WP5, see Fig. 6). In particular we develop our planners to handle the output generated by scene understanding and human-robot spatial interaction components.
- Plan considering different *time-scales and representations*. Contextual cues and humans' dynamics are represented on different time scales. Usually semantics of the environments are reported in semi-static representations: maps that encode objects' semantics which are updated while the robot is fulfilling its tasks. Humans' dynamics are often handled according to two different time scales: semi-static representations where long-term human motion (or activity) patterns are reported (e.g., Maps of Dynamics – MoDs [11, 12]), dynamic and punctual ones where the current and short-term predicted behaviors are detailed (e.g., trajectory predictions).
- Consider *risk and safety aspects* when planning the robot operation and motion. The generated plans should reduce the risk of collision or in general robot unsafe operation at a minimum.

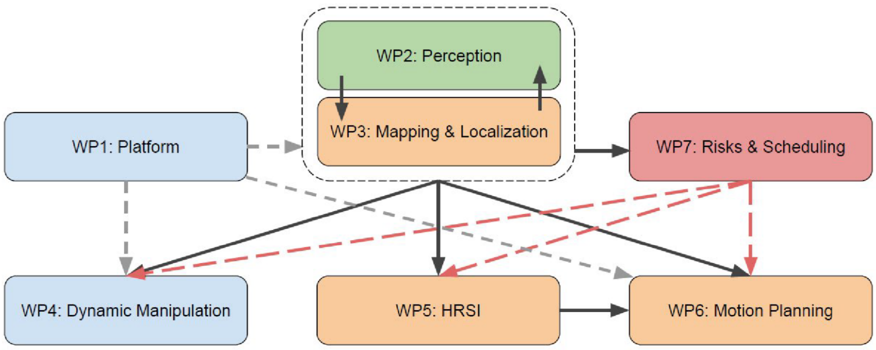

Figure 6: Overview of the project structure. In this deliverable we present the prototype of the navigation system, developed in WP6. Important inputs to the navigation system are generated in WP2 (Perception), WP3 (Mapping) and WP5 (Prediction).

During this first part of the project, we have developed a multi-layered architecture that addresses all the detailed requirements. A sketch of the architecture is reported in Fig. 7.

The architecture follows a *predictive planning* setup: differently from a traditional *sense-plan-act* one [13], our architecture plug-ins directly predictions of surrounding humans and dynamic objects inside the different layers. Differently from the state of the art, we include predictions at different levels of planning, considering different time scales. All planning layers use therefore predictions for generating robot motion. Only the safety layer considers no prediction in the current implementation.

The architecture is based on the open-source Robot Operating System (ROS) 1 middleware. As a consequence, the architecture is inherently component-based. We make use of the open-source navigation framework *move-base*.

5.1 Architectural Layers

Our architecture is composed of three main layers:

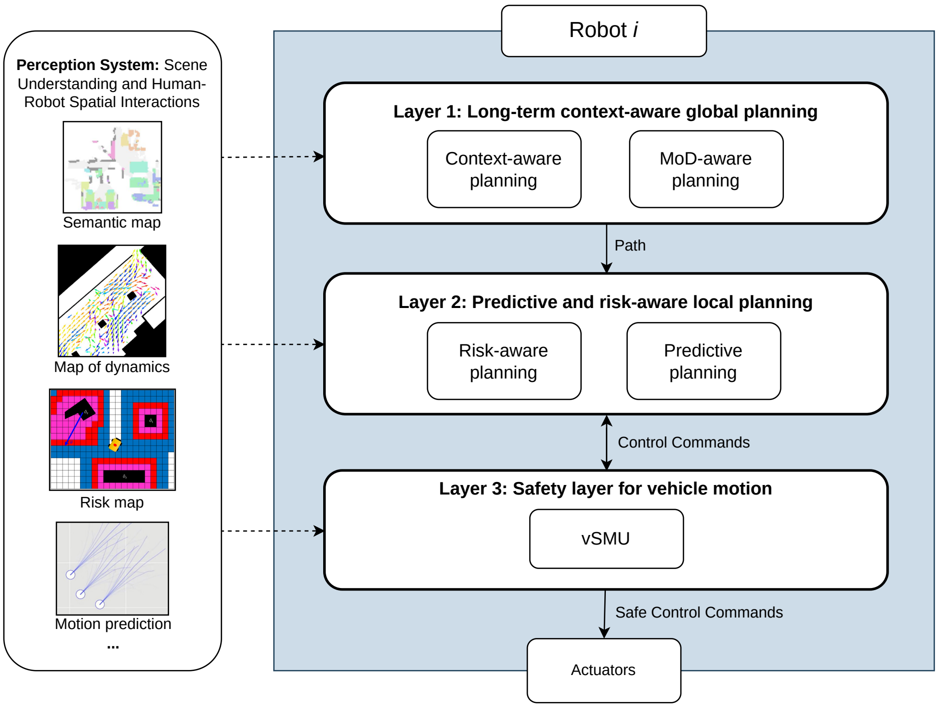

Figure 7: Motion planning system overview. The current system has 3 main layers. Each layer handles different contextual aspects of the environment.

- **Layer 1: Long-term context-aware global planning**. This layer is responsible to generate paths for legible robot operations considering long-term human behaviors and semantics of the environment. In DARKO, we encode long-term human behaviors as Maps of Dynamics (output of WP3): those maps encode usual flow of people navigating in a particular environment. Semantics can be represented as labeled data (i.e., text labels associated to geometric entities and positions) or as neural network architecture embeddings.

  Currently there are two components on this layer: a context-aware global navigation system and a MoD-aware path planner. The context-aware global planner enables reaching goals in partially unknown semantically-rich environments, bridging exploration with goal-directed navigation, considering primarily the static context.

Additionally, the dynamic long-term context for navigation in human environments, represented with Maps of Dynamics, is handled by the MoD-aware path planner.

- **Layer 2: Predictive and risk-aware local planning**. The paths computed by the long-term semantics-aware predictive planning unit are then forwarded to the predictive and risk-aware local planning layer. In this layer we adapt the former path according to different definitions of risk and by performing collision avoidance considering short-term predictions. Differently from the previously layer, these components run to a higher frequency (circa 20 Hz) thus allowing a quicker reaction to unforeseen events. The components will generate velocity commands for the robot platform.

  Currently there are two components on this layer: risk-aware planner, and predictive collision avoidance algorithm. The risk-aware planner, which is called at the end of every global planner iteration at Layer 1, takes as input the waypoints and returns an updated set of safe waypoints. The risk definition in this case includes the probability of collision and localization error. The updates set of waypoints follows to the predictive collision avoidance algorithm, which generates control commands considering short-term predictions for the nearby humans. The goal of this method is to generate efficient socially-compliant motion for the robot.

- **Layer 3: Safety layer for vehicle motion**. This layer is responsible for re-shaping the computed velocity commands based on the safety requirements. To reduce the hazards and harms to surrounding humans, the velocities of the robot platform are scaled based on bio-mechanical analysis of possible injuries with the nearby people. The safety components used in this layer are described in Section 2.

6 Robot intent

Even the robot motion itself can be seen as an important communication channel, but rather for experienced and trained personnel. For inexperienced participants (with respect to the peculiarities of the DARKO system), robot motion and other external signs, such as LED stripes, sound indicators, etc., might sometimes be overwhelming or confusing, leading to an inefficient and possibly more dangerous collaboration. Research has mostly explored the possibility of engaging more feedback methods that are commonly available on industrial robotic platforms, such as, for example, in-the-floor projections [14]. Despite all the advancements, the need for approaches that can be validated and used across a range of mobile robots remains open [15, 16]. The need for more efficient deployment and communication has led us to develop the concept of the "Anthropomorphic Robotic Mock Driver" (ARMoD) as an easily deployable platform for facilitating intuitive communication between non-humanoid robots and human co-workers in workplace settings [17] (presented within D5.2).

As indicated previously, communicating intent is an important part of the safety system on the DARKO platform. In particular, communicating intent can create a better way to collaborate with humans, warn about potentially dangerous situations, etc. The ARMoD concept involves integrating a humanoid robot onto the DARKO robot to leverage anthropomorphic features for more natural and intuitive communication. Mounting an ARMoD on the DARKO robot (see Figure 8) provides additional communication channels such as a Head for robotic gaze, a Text-To-Speech Engine for Dialogues, Additional LEDs and Arms for pointing, and other gestures. This approach was initially inspired by ongoing work at ORU on human-robot interaction and the promising results of established research such as [18, 19].

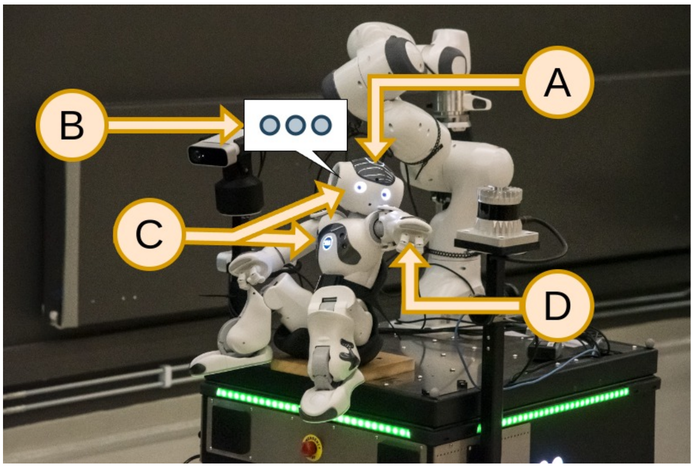

Figure 8: Additional Intent Communication Channels with an ARMoD: Equipping an ARMoD provides additional channels to leverage its anthropomorphic features to communicate intent. A: Head for robotic gaze, B: Text-To-Speech for Dialogues, C: Additional LEDs, D: Arms for pointing and other gestures.

The ARMoD can use anthropomorphic features such as head movements, robotic gaze, and gestures to communicate intent, providing implicit cues similar to those used in human communication, enhancing the clarity and naturalness of a robot's communication in shared spaces. Especially naturalistic cues, like gaze direction, might be more easily interpreted than traditional signals, such as LED turn signals, providing a more intuitive and effective means of communication [20]. Additionally, the ARMoD system has been designed to be easily programmed to perform a wide range of communicative behaviors independently and in parallel to the DARKO robot, allowing it to adapt to various situational needs.

7 Planning and Control Framework for Manipulator

The planning and control framework designed for the robotic arm in DARKO offers an extensive array of solutions for issuing commands to the manipulator. This choice was made to take into account the various tasks that we have to accomplish inside the presented scenario (such as simple repositioning of the arm, object picking, throwing, etc...). For the first integration, it was decided to start integrating the vSMU with the planning and control at the Cartesian level, which will be the most used modality to perform pick and place actions. For the sake of brevity, in this section, we will describe only the part related to the integration, while we refer the interested reader to Deliverable D4.1 "Preliminary planning and control software developments for efficient manipulation" for a deeper description of all the planning and control framework.

7.1 Control algorithm

The control law chosen to follow a Cartesian reference is a classical Cartesian impedance controller [21]. This is done to ensure a safer interaction with the environment in comparison to the classical position controllers which requires some policies to avoid the generation of high interaction forces. Given a desired Cartesian trajectory x_{des} and its time derivative $\dot{x}_{des}$, we can define the error between the actual and desired Cartesian position and the velocity of the manipulator as $e = x_{des} - x$ and $\dot{e} = \dot{x}_{des} - \dot{x}$. From these values, we can compute the joint torque necessary to follow the desired trajectory:

$$\tau = J^T(q)(K_p e + K_v \dot{e}) + G(q) + C(q, \dot{q})\dot{q} \tag{4}$$

where $J(q)$ is the Jacobian, $G(q)$ is the gravitational term, $C(q)\dot{q}$ is the Coriolis term, and K_p and K_v are the matrices that define the impedance behaviour of the controller.

7.2 Human-Like motion planning algorithm

Regarding the computation of the trajectory, the proposed technique is an algorithm able to generate motion similar to the ones produced by humans. The embedding of this behaviour in the robot motion is a key factor in increasing the predictability and the acceptability during human-robot interaction [22].

To achieve this goal, the proposed technique is to exploit the functional Principal Component Analysis (fPCA) to extract a representation of the human motion features using a reduced number of functional elements. Different works in the literature prove the effectiveness of this type of analysis in extracting a reduced basis of functions related to the human arm movements both in the joint [23] and the Cartesian domain [24], and also in using the output of this analysis to design a planning algorithm to generate Human-Like trajectories [25]. Following this line, in [26] the authors have presented a novel Human-Like motion planning algorithm defined directly in the Cartesian domain starting from the analysis of the human hand motions. This solution, given its abstraction from the kinematics of the human arm, allows for easier integration with any manipulators. In the next sections, we will provide a brief explanation of the theory behind the fPCA and how it is exploited for the development of the planning algorithm. For further details regarding the validation of this approach, we refer the interested reader to the original paper.

7.2.1 Functional Principal Component Analysis

Functional Principal Component Analysis (fPCA) is a statistical method to identify a geometrical basis of functions whose elements can be combined to reconstruct time series. In this section, we will provide a brief introduction to the underpinning theory and its application - without loss of generality - to the description of hand trajectories (i.e. the trajectories of the end-effector of the upper limb kinematic chain), while referring the interested reader to [27] for more details. Given a dataset of hand motions, the generic motion $x(t)$ can be represented as a weighted sum of a set of basis functions $S_i(t)$, or functional Principal Components (fPCs) extracted from the dataset, that is:

$$x(t) \simeq \bar{x} + S_0(t) + \sum_{i=1}^{s_{max}} \alpha_i \circ S_i(t) \tag{5}$$

where $\bar{x}$ is the average pose of the hand, $S_0(t)$ is the average trajectory across all the trajectories in the dataset, α_i is a vector of weights, s_{max} is the number of basis elements, $S_i(t)$ is the i^{th} basis element, the symbol $\circ$ represents the Hadamard product (i.e. the element-wise product) and $t \in [0, 1]$ is the normalized time axis.

The first element of the functional basis or first fPC can be computed from the R motions of the dataset as:

$$\max_{S_1} \sum_{j=1}^{R} \left(\int S_1(t) x_j(t) dt \right)^2 \tag{6}$$

subject to

$$||S_1(t)||_2^2 = 1 \tag{7}$$

The other components $S_i(t)$ can be computed as:

$$\max_{S_i} \sum_{j=1}^{R} \left(\int S_i(t) x_j(t) dt \right)^2 \tag{8}$$

subject to

$$||S_i(t)||_2^2 = 1 \tag{9}$$

$$\int_0^{t_{end}} S_i(t) S_k(t) dt = 0, \forall k \in \{1, ..., i-1\} \tag{10}$$

In this manner, we can identify a basis of functional elements, ordered in terms of the explained variance that each element accounts for.

For our purpose, to obtain a general representation of the human hand motion, we used the dataset proposed in [28], containing the recording of 30 different activities of daily living performed by 30 different subjects belonging to three main classes of actions: Transitive, Intransitive and Tool-mediated, which were assumed to be representative of the human example [23].

7.2.2 Planning algorithm

The fPCs extracted from a dataset that can be considered representative of the most common upper limb movements can be used to plan trajectories that intrinsically embed Human-Likeness. In the following section, we provide a formalization of the planning problem. Of-note, fPCA is performed for each Degree of Freedom (DoF) separately that, in our case, are the Cartesian position and orientation of the end-effector with respect to the base frame. In the following, we report the equations for a single DoF of the end effector, while the extension to multiple DoFs (e.g. the six DoFs describing the pose of the end effector) is trivial.

The reconstruction of the single DoF trajectory can be attained as:

$$x(t) \simeq \bar{x} + S_0(t) + \sum_{i=1}^{s_{max}} \alpha_i S_i(t) \tag{11}$$

To find the coefficients $\bar{x}$ and α_i given a set of constraints to be satisfied we can define an equation system to obtain the desired trajectory to be planned. For example, setting the initial and final position, velocity and acceleration, the following equation system is defined:

$$\begin{bmatrix} 1 & S_1(t_0) & \dots & S_5(t_0) \\ 1 & S_1(t_f) & \dots & S_5(t_f) \\ 0 & \dot{S}_1(t_0) & \dots & \dot{S}_5(t_0) \\ 0 & \dot{S}_1(t_f) & \dots & \dot{S}_5(t_f) \\ 0 & \ddot{S}_1(t_0) & \dots & \ddot{S}_5(t_0) \\ 0 & \ddot{S}_1(t_f) & \dots & \ddot{S}_5(t_f) \end{bmatrix} \begin{bmatrix} \bar{x} \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \alpha_5 \end{bmatrix} = \begin{bmatrix} x(t_0) - S_0(t_0) \\ x(t_f) - S_0(t_f) \\ \dot{x}(t_0) - \dot{S}_0(t_0) \\ \dot{x}(t_f) - \dot{S}_0(t_f) \\ \ddot{x}(t_0) - \ddot{S}_0(t_0) \\ \ddot{x}(t_f) - \ddot{S}_0(t_f) \end{bmatrix} \tag{12}$$

by solving the system we can obtain the desired planned trajectory

$$x(t) = \bar{x} + S_0(t) + \sum_{i=1}^{5} \alpha_i S_i(t) \tag{13}$$

7.3 Integration with vSMU

As previously mentioned, the vSMU determines the maximum permissible velocity to avoid severe injuries in the event of a collision between a robot and a human. Therefore, the arm motion planning algorithm must be crafted to modify its planned movements in line with the speed constraints set by the vSMU. There are three methods to achieve this:

1. vSMU can be a middleman between the Planner and Controller. In this case, the Planner can not foresee and avoid regions with lower speeds (it can not adjust for the speed limitation online). vSMU rather just clamps the speed to ensure safe execution.

2. Similar to the previous case, but rather than limiting the speed itself, it constantly provides a maximum safe speed, taking into account the current robot configuration and human data.

3. vSMU can work asynchronously (on request). Meaning that one can feed various arm configurations to be checked for safety. This can be useful for sampling-based planning approaches, to try to actively avoid low-speed regions.

For this integration it was decided to focus on approach 2. To simplify, we consider only the manipulator's end effector as a dangerous point (POI) and try to limit the speed based on the robot's reflected mass at that point.

Since all system components are integrated within Robotic Operating System framework, it was decided to use a publisher/subscriber structure to pass the velocity limit from the vSMU to the motion planning node (as indicated in approach 2). In this way, the vSMU computes the safe velocity using the information regarding the actual state of the manipulator and its relative position with respect to the worker and makes it available to the others thorough a dedicated topic.

Regarding the motion planning side of this infrastructure, it is synchronously checking if the maximum planned velocity meets the constraint given by vSMU and, if not, slows down the trajectory. From a technical point of view, the node has a cycle inside that publishes each frame of the trajectory with the right sampling time to respect the desired behaviour. To integrate the information provided by vSMU, at every cycle, the node listens for the maximum safe velocity, compares this value with the actual desired velocity, and adjusts the sampling time of the trajectory to keep the movement inside the safety constraint.

In Figure 9 we can observe the results, in terms of end-effector Cartesian velocity, of an integration test between the human-like motion planner and the vSMU. In this test, we asked the planner to move the manipulator's end-effector forward and backwards along the same trajectory on the horizontal plane (plane x-y in the robot frame). At the time $t = 5s$ a human went in front of the manipulator, activating the vSMU. We can observe that when the human is near the manipulator, there is a decrease in the end-effector Cartesian velocity. Another interesting thing to notice is that the decrease is stronger when the x component of the velocity is positive. This happens because, in this case, when $v_x > 0$ the end-effector is moving toward the human and the vSMU sets a lower maximum allowed velocity to limit dangerous collision.

8 Whole body Impedance controller

In traditional mobile manipulator systems, the robotic arm and the mobile base are typically treated as two separate models. This streamlines the modeling process, ensuring stability and ease of implementation, as well as facilitating comprehension. However,

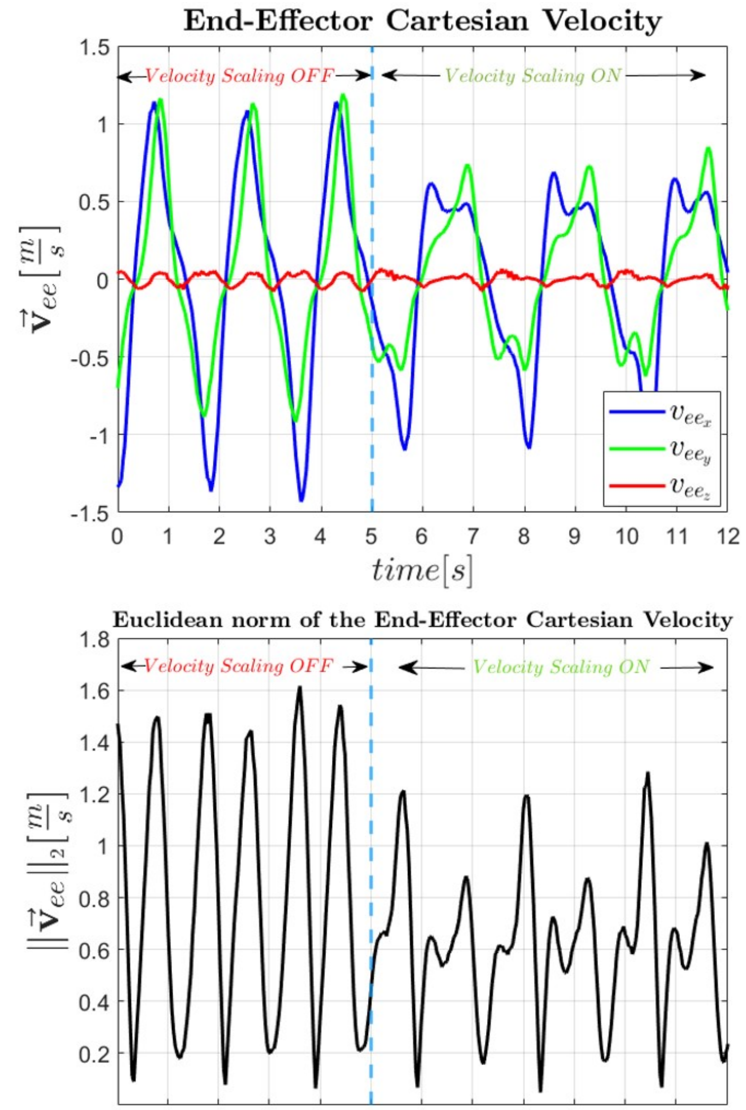

Figure 9: Velocity profile obtained during integration test between human-like motion planner and SMU. The manipulator moves the end-effector forward and backwards along the same trajectory on the horizontal plane. At $t = 5s$, a human goes in front of the manipulator activating the SMU velocity scaling. The plot on the left represents the single components of the Cartesian velocity of the end-effector during test: The plot on the right represent the norm of the Cartesian velocity of the end-effector.

during task execution, this approach prevents simultaneous operation of the mobile base and the robotic arm, resulting in a low efficiency and the inability to exhibit whole-body compliance behavior during interactions with the environment. Especially, keeping in mind the importance of compliant behaviour for safety. In this section, we extend the underlying Impedance controller, mentioned as a low-level controller within the Human-like planner in Section 7. We introduce a control strategy that takes into account the arm and mobile base as a whole-body. Based on this, a unified model is constructed that treats these two components as an integrated whole for the analysis of dynamics and kinematics.

To construct a whole-body control system, it is essential to develop a unified dynamic model that accounts for the coupling terms of inertia, Coriolis, and centrifugal torque

$$\begin{aligned}&\begin{bmatrix}\boldsymbol{M}_v+\boldsymbol{M}_{a,up} & \boldsymbol{M}_{va}\\ \boldsymbol{M}_{va}^T & \boldsymbol{M}_a\end{bmatrix}\begin{bmatrix}\ddot{\boldsymbol{q}}_v\\ \ddot{\boldsymbol{q}}_a\end{bmatrix}+\begin{bmatrix}\boldsymbol{C}_v & \boldsymbol{C}_{va}\\ \boldsymbol{C}_{av} & \boldsymbol{C}_a\end{bmatrix}\begin{bmatrix}\dot{\boldsymbol{q}}_v\\ \dot{\boldsymbol{q}}_a\end{bmatrix}+\begin{bmatrix}\boldsymbol{0}\\ \boldsymbol{g}_a\end{bmatrix}\\ &=\begin{bmatrix}\boldsymbol{E}_v(\boldsymbol{q}_v)\boldsymbol{\tau}_v\\ \boldsymbol{\tau}_a\end{bmatrix}+\begin{bmatrix}\boldsymbol{\tau}_v^{ext}\\ \boldsymbol{\tau}_a^{ext}\end{bmatrix}\end{aligned} \tag{14}$$

where $\boldsymbol{M}_{va} \in \mathbb{R}^{n_{q_a}\times n_{q_v}}$ is the inertial matrix which represents the effect of the mobile platform dynamics on the manipulator, $\boldsymbol{M}_{av}=\boldsymbol{M}_{va}^T \in \mathbb{R}^{n_{q_v}\times n_{q_a}}$ is the inertial matrix which reflects the dynamic effect of the manipulator motion on the mobile platform. $\boldsymbol{C}_{av} \in \mathbb{R}^{n_{q_a}}$ is Coriolis and centrifugal terms caused by angular motion of the mobile base, $\boldsymbol{C}_{va} \in \mathbb{R}^{n_{q_v}}$ denote Coriolis and centrifugal terms due to the presence of the manipulator, $\boldsymbol{M}_{v,up}$ is the extra inertial matrix, because the arm is mounted on the base, $\boldsymbol{E}_v \in \mathbb{R}^{n_{q_v}\times n_{q_v}}$ is the input transformation matrix.

Additionally, obtaining a whole-body Jacobian matrix is crucial, as it allows us to simultaneously account for the contributions of both the base and the arm movements to the end-effector's velocity. The calculation of the whole-body Jacobian matrix is as follows:

$$\begin{bmatrix}{}^S\boldsymbol{v}_e\\ {}^S\boldsymbol{\omega}_e\end{bmatrix}=\begin{bmatrix}\boldsymbol{I}_{3\times3} & -{}^S\boldsymbol{p}_{ve}\\ \boldsymbol{0}_{3\times3} & \boldsymbol{I}_{3\times3}\end{bmatrix}\begin{bmatrix}{}^S\boldsymbol{v}_v\\ {}^S\boldsymbol{\omega}_v\end{bmatrix}+\begin{bmatrix}{}^S_A\boldsymbol{R} & \boldsymbol{0}_{3\times3}\\ \boldsymbol{0}_{3\times3} & {}^S_A\boldsymbol{R}\end{bmatrix}\begin{bmatrix}{}^S\boldsymbol{v}_{e,a}\\ {}^S\boldsymbol{\omega}_{e,a}\end{bmatrix}=\begin{bmatrix}\boldsymbol{V}_v(\boldsymbol{q}_a){}^S\boldsymbol{J}_v & \boldsymbol{V}_a(\boldsymbol{q}_v){}^S\boldsymbol{J}_a\end{bmatrix}\begin{bmatrix}\boldsymbol{q}_v\\ \boldsymbol{q}_a\end{bmatrix} \tag{15}$$

where $\boldsymbol{V}_v(\boldsymbol{q}_a)=\begin{bmatrix}\boldsymbol{I}_{3\times3} & -{}^S\boldsymbol{p}_{ve}\\ \boldsymbol{0}_{3\times3} & \boldsymbol{I}_{3\times3}\end{bmatrix}$, $\boldsymbol{V}_a(\boldsymbol{q}_v)=\begin{bmatrix}{}^S_A\boldsymbol{R} & \boldsymbol{0}_{3\times3}\\ \boldsymbol{0}_{3\times3} & {}^S_A\boldsymbol{R}\end{bmatrix}$. In conclusion, the whole-body geometric Jacobian is formulated as:

$${}^S\boldsymbol{J}=\begin{bmatrix}\boldsymbol{V}_v(\boldsymbol{q}_a){}^S\boldsymbol{J}_v & \boldsymbol{V}_a(\boldsymbol{q}_v){}^S\boldsymbol{J}_a\end{bmatrix} \tag{16}$$

Eventually, we could get the control law of whole-body Impedance controller(with inertial shaping) by involving the whole-body kinematic and dynamic parameters:

$$\boldsymbol{\tau}=\boldsymbol{J}(\boldsymbol{q})^T\boldsymbol{F}_\tau=\boldsymbol{g}(\boldsymbol{q})+\boldsymbol{J}(\boldsymbol{q})^T(\boldsymbol{\Lambda}(\boldsymbol{x})\ddot{\boldsymbol{x}}_d+\boldsymbol{\mu}(\boldsymbol{x},\dot{\boldsymbol{x}})\dot{\boldsymbol{x}}_d-\boldsymbol{K}_d\widetilde{\boldsymbol{x}}-\boldsymbol{D}_d(\boldsymbol{x})\dot{\widetilde{\boldsymbol{x}}}) \tag{17}$$

where $\widetilde{\boldsymbol{x}}=\boldsymbol{x}-\boldsymbol{x}_d$ is the pose error, $\Lambda(\boldsymbol{x})$ and $\boldsymbol{\mu}(\boldsymbol{x},\dot{\boldsymbol{x}})$ are inertial and Coriolis/centrifugal matrix in Cartesian space. after applying this impedance controller to the darko dynamic system as described in 15, we can acquire the desired compliance behaviour to the external force

$$\boldsymbol{f}_{ext}=\boldsymbol{\Lambda}_d\ddot{\widetilde{\boldsymbol{x}}}+\boldsymbol{D}_d\dot{\widetilde{\boldsymbol{x}}}+\boldsymbol{K}_d\widetilde{\boldsymbol{x}} \tag{18}$$

9 Scenario testing

For the showcase scenario, the goal was to demonstrate a complete scenario engaging vSMU together with other necessary system components. Since all parts of the system are ROS compatible, it was easy to visualize all the incoming data in soft real-time using RVIZ,

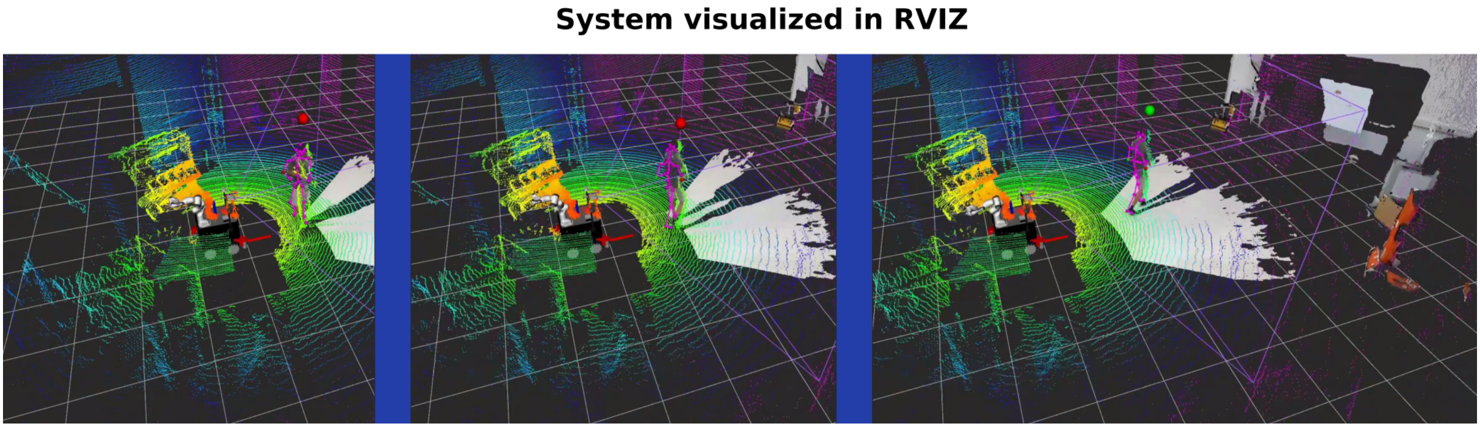

Figure 10: Robot perceived data visualization in the soft-realtime in RVIZ. Besides localizing the robot in the space, it can be seen that the human is correctly detected (together with detecting its orientation, skeleton fit, speed, etc.). The human position is classified as dangerous in the two left pictures (indicated by the floating red ball above the human representation) and as safe in the right picture (floating green ball). The pictures represent all the data available to the robot visualized during the tested scenario. Note that the robot arm position is not constant - it is moving between two points in Cartesian space.

Figure 11: Picture sequence of the mock-up scenario with the safety regions overlay. Data visualized during this scenario can be seen in Figure 10.

as shown in Figure 10. The chosen mock-up scenario is a simple representation of the use-case environment where worker can freely approach the robot. As visualized in RVIZ, worker classification (as dangerous or safe) is indicated above its representation (red or green accordingly) and is determined based on the detected human position. How the situation looked like in the mock-up environment settings, can be observed in Figure 11. Additionally, virtual safety zones have been added to the figure.

The robot has been set to move from the two positions in cartesian space (emulating a task execution). The human worker is being perceived by the perception, and its position (skeleton fit, orientation, speed, etc.) is correctly localized in the environment, as well as with respect to the robot position. As the human enters the pre-configured safety zone, the vSMU is triggered to limit the robot's speed. As can be seen in Figure 12, the speed reduction ratio is computed in the online control cycle and is not a constant value - it's a result of a comparison of the robot's current speed with the maximum safe speed at the current robot position (depending on the robot's perceived environment data).

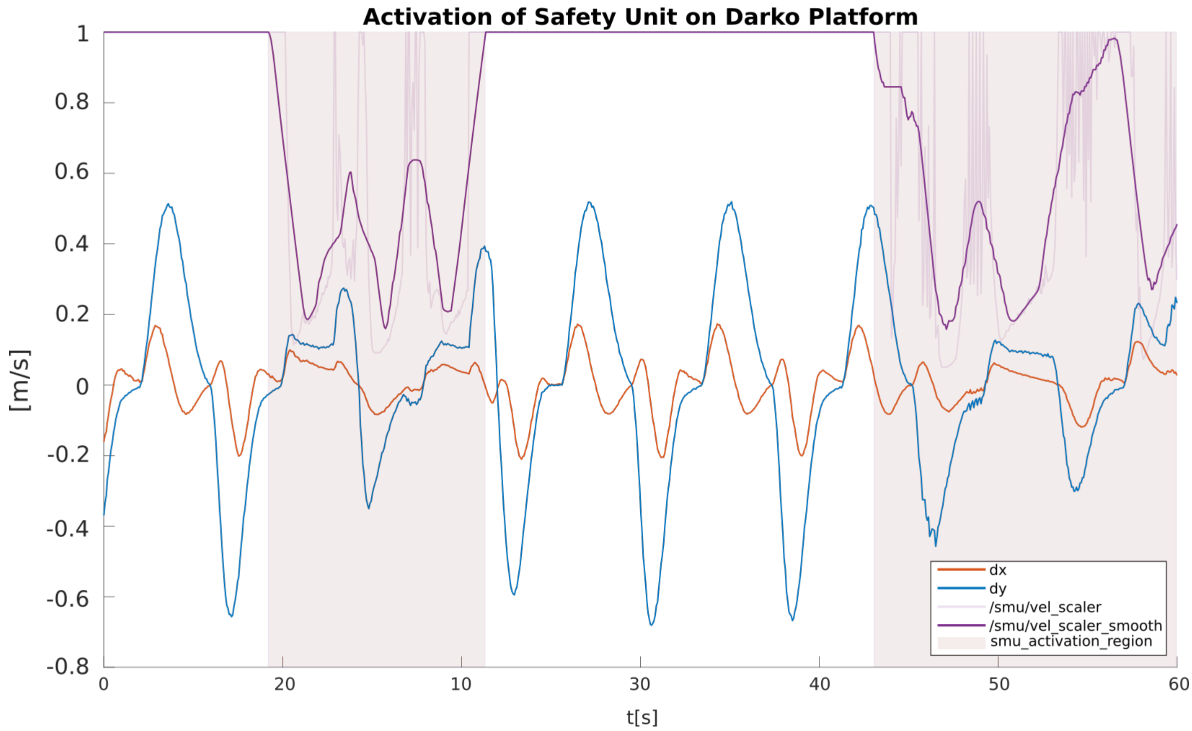

Figure 12: Recorded data of the Cartesian speed (dx and dy), alongside the safety velocity scaler value (taken into account by the robot control algorithms to reduce the speed with minimum sacrifices to the task execution performance). The safety activation region is when the scaler requires lowering the velocity. Note that the safety activation region doesn't always correspond to when the human is present in the unsafe zone - since, for example, velocity scaling is not required when the speed doesn't exceed the safe speed for the current robot configuration.

10 Conclusions

As has been presented, safety is not only defined through one protocol, but is rather embedded in different aspects of the system that are orchestrated to achieve reliable and safe performance of the robot. An unified approach for introducing safety for mobile platform with embedded manipulator, by primarily limiting the speed has been described. Further, different aspects for the platform system integration such as perception, prediction, features conveying robot intent, navigation and planning have been highlighted throughout the report. In particular, since safety is highly relying on accuracy of various inputs developed by the consortium (vision, human detection, localization, etc.), the latest relevant standards have been summarized to provide a perspective on how this report's development aligns with their requirements. The results demonstrate safety performance on the platform where the primary focus was to achieve an integrated demonstrator. Considerations with respect to the elastic arm features and future integration within the elastic platform are mentioned. We argue that in the current setting, where the elastic features are reserved for high-speed manipulation, this safety demonstration is valid for both - industrial and novel arm development. Especially since, in this respect, experiments on a rigid arm (joints made from classical actuators consisting of motor and gearbox) are equivalent to the demonstration on a newly developed arm without engaging various clutch modes and without dominant excitation of the spring dynamics. Further, this software stack is expected to be deployed and showcased in the final setup with newly developed manipulator. End-scenario from this report represents the successful integration and concurrent work of various required modules to ensure safety in the work environment.

References

[1] Sami Haddadin, Alin Albu-Schaffer, Mirko Frommberger, J. Rossmann, and Gerd Hirzinger. "The DLR Crash Report: Towards a Standard Crash-Testing Protocol for Robot Safety - Part I: Results". In: *Proceedings - IEEE International Conference on Robotics and Automation*. June 2009, pp. 272–279.

[2] Sami Haddadin, Simon Haddadin, Augusto Khoury, Tim Rokahr, Sven Parusel, Rainer Burgkart, Antonio Bicchi, and Alin Albu-Schäffer. "A truly safely moving robot has to know what injury it may cause". In: *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2012, pp. 5406–5413.

[3] Mazin Hamad, Alexander Kurdas, Nico Mansfeld, Saeed Abdolshah, and Sami Haddadin. "Modularize-and-Conquer: A Generalized Impact Dynamics and Safe Precollision Control Framework for Floating-Base Tree-Like Robots". In: *IEEE Transactions on Robotics* 39.4 (2023), pp. 3200–3221.

[4] Petr Svarny, Mazin Hamad, Alexander Kurdas, Matej Hoffmann, Saeed Abdolshah, and Sami Haddadin. "Functional Mode Switching for Safe and Efficient Human-Robot Interaction". In: *2022 IEEE-RAS 21st International Conference on Humanoid Robots (Humanoids)*. 2022, pp. 888–894.

[5] Luca Castri, Sariah Mghames, Marc Hanheide, and Nicola Bellotto. "Causal discovery of dynamic models for predicting human spatial interactions". In: *International Conference on Social Robotics*. Springer. 2022, pp. 154–164.

[6] Luca Castri, Sariah Mghames, Marc Hanheide, and Nicola Bellotto. "Enhancing Causal Discovery from Robot Sensor Data in Dynamic Scenarios". In: *Conference on Causal Learning and Reasoning*. PMLR. 2023, pp. 243–258.

[7] Andrey Rudenko, Tomasz P Kucner, Chittaranjan S Swaminathan, Ravi T Chadalavada, Kai O Arras, and Achim J Lilienthal. “THÖR: Human-Robot Navigation Data Collection and Accurate Motion Trajectories Dataset”. In: *IEEE Robotics & Automation Letters* (2020), pp. 676–682.

[8] Timm Linder, Kilian Y. Pfeiffer, Narunas Vaskevicius, Robert Schirmer, and Kai O. Arras. “Accurate Detection and 3D Localization of Humans Using a Novel YOLO-Based RGB-D Fusion Approach and Synthetic Training Data”. In: *IEEE International Conference on Robotics and Automation (ICRA)*. 2020.

[9] Timm Linder, Narunas Vaskevicius, Robert Schirmer, and Kai Oliver Arras. “Cross-Modal Analysis of Human Detection for Robotics: An Industrial Case Study”. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2021.

[10] T. Linder, S. Breuers, B. Leibe, and K. O. Arras. “On Multi-Modal People Tracking from Mobile Platforms in Very Crowded and Dynamic Environments”. In: *IEEE International Conference on Robotics and Automation (ICRA)*. 2016.

[11] Tomasz Piotr Kucner, Martin Magnusson, Erik Schaffernicht, Victor H. Bennetts, and Achim J. Lilienthal. “Enabling Flow Awareness for Mobile Robots in Partially Observable Environments”. In: 2.2 (Apr. 2017), pp. 1093–1100.

[12] Tomasz Piotr Kucner, Martin Magnusson, Sariah Mghames, Luigi Palmieri, Francesco Verdoja, Chittaranjan Srinivas Swaminathan, Tomáš Krajník, Erik Schaffernicht, Nicola Bellotto, Marc Hanheide, et al. “Survey of maps of dynamics for mobile robots”. In: *The International Journal of Robotics Research* (), p. 02783649231190428.

[13] Bruno Siciliano, Oussama Khatib, and Torsten Kröger. *Springer handbook of robotics*. Vol. 200. Springer, 2008.

[14] Ravi Teja Chadalavada, Henrik Andreasson, Maike Schindler, Rainer Palm, and Achim J Lilienthal. “Bi-directional navigation intent communication using spatial augmented reality and eye-tracking glasses for improved safety in human–robot interaction”. In: *Robotics and Computer-Integrated Manufacturing* 61 (2020), p. 101830.

[15] Elizabeth Cha, Yunkyung Kim, Terrence Fong, Maja J Mataric, et al. “A survey of nonverbal signaling methods for non-humanoid robots”. In: *Foundations and Trends® in Robotics* 6.4 (2018), pp. 211–323.

[16] Georgios Angelopoulos, Francesco Vigni, Alessandra Rossi, Giuseppina Russo, Mario Turco, and Silvia Rossi. “Familiar acoustic cues for legible service robots”. In: *2022 31st IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. IEEE. 2022, pp. 1187–1192.

[17] Tim Schreiter, Lucas Morillo-Mendez, Ravi T Chadalavada, Andrey Rudenko, Erik Billing, Martin Magnusson, Kai O Arras, and Achim J Lilienthal. “Advantages of Multimodal versus Verbal-Only Robot-to-Human Communication with an Anthropomorphic Robotic Mock Driver”. In: *2023 32nd IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. IEEE. 2023, pp. 293–300.

[18] Kerstin Severinson-Eklundh, Anders Green, and Helge Hüttenrauch. “Social and collaborative aspects of interaction with a service robot”. In: *Robotics and Autonomous systems* 42.3-4 (2003), pp. 223–234.

[19] Maha Salem, Katharina Rohlfing, Stefan Kopp, and Frank Joublin. “A friendly gesture: Investigating the effect of multimodal robot behavior in human-robot interaction”. In: *2011 ro-man*. IEEE. 2011, pp. 247–252.

[20] Justin Hart, Reuth Mirsky, Stone Tejeda, Bonny Mahajan, Jamin Goo, Kathryn Baldauf, Sydney Owen, and Peter Stone. "Unclogging our arteries: using human-inspired signals to disambiguate navigational intentions". In: *arXiv preprint arXiv:1909.06560* (2019).

[21] Christian Ott. "Cartesian impedance control: The rigid body case". In: *Cartesian Impedance Control of Redundant and Flexible-Joint Robots* (2008), pp. 29–44.

[22] Andrea Maria Zanchettin, Luca Bascetta, and Paolo Rocco. "Acceptability of robotic manipulators in shared working environments through human-like redundancy resolution". In: *Applied ergonomics* 44.6 (2013), pp. 982–989.

[23] Giuseppe Averta, Cosimo Della Santina, Edoardo Battaglia, Federica Felici, Matteo Bianchi, and Antonio Bicchi. "Unvealing the principal modes of human upper limb movements through functional analysis". In: *Frontiers in Robotics and AI* 4 (2017), p. 37.

[24] Marco Baracca, Paolo Bonifati, Ylenia Nisticò, Vincenzo Catrambone, Gaetano Valenza, A Bicchi, Giuseppe Averta, and Matteo Bianchi. "Functional analysis of upper-limb movements in the Cartesian domain". In: *Converging Clinical and Engineering Research on Neurorehabilitation IV: Proceedings of the 5th International Conference on Neurorehabilitation (ICNR2020), October 13–16, 2020*. Springer. 2022, pp. 339–343.

[25] Giuseppe Averta, Danilo Caporale, Cosimo Della Santina, Antonio Bicchi, and Matteo Bianchi. "A technical framework for human-like motion generation with autonomous anthropomorphic redundant manipulators". In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2020, pp. 3853–3859.

[26] Marco Baracca, Giuseppe Averta, and Matteo Bianchi. "A general approach for generating artificial human-like motions from functional components of human upper limb movements". In: *Control Engineering Practice* 148 (2024), p. 105968.

[27] James O Ramsay, Giles Hooker, and Spencer Graves. *Functional data analysis with R and MATLAB*. Springer Science & Business Media, 2009.

[28] Giuseppe Averta, Federica Barontini, Vincenzo Catrambone, Sami Haddadin, Giacomo Handjaras, Jeremia PO Held, Tingli Hu, Eike Jakubowitz, Christoph M Kanzler, Johannes Kühn, et al. "U-Limb: A multi-modal, multi-center database on arm motion control in healthy and post-stroke conditions". In: *GigaScience* 10.6 (2021), giab043.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101017274