The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

Holistic design and analysis for the human-friendly robotic co-worker

Sami Haddadin, Sven Parusel, Rico Belder, Jörn Vogel, Tim Rokahr, Alin Albu-Schäffer, and Gerd Hirzinger

Abstract— **In this overview paper we present current work on safety analysis for physical Human-Robot Interaction (pHRI) and motion control methods for robotic co-workers. In particular, we introduce the analysis tools for investigating the potential injury a human would suffer during robot-human impacts. Furthermore, we outline our concept for establishing a procedure towards standardized crash testing in robotics with automobile crash-test dummies. Since it is only possible to investigate blunt impacts with these devices, we developed a drop testing setup for analyzing soft-tissue injury in robotics from a biomechanics perspective. In the second part of the paper, some of our methods for task preserving and task relaxing motion schemes are described, which enable collision avoidance in real-time. The algorithms are well suited to work in an integrated fashion with the soft robotics control developed for the DLR Lightweight Robot III (LWR-III). In addition, it is shown how the torque sensing capabilities of the robot can be used to support reactive motion schemes. Finally, an overview of our human-friendly control architecture for the LWR-III is given, which unifies the rich bundle of developed methods for this manipulator.**

I. Introduction

pHRI is expected to be of large benefit for future advanced robotic applications where the segregation of human and robot workspace will vanish. In order to achieve such an ambitious goal, there are currently large efforts undertaken in the robotics research community to develop novel designs and control schemes for robots with strong interaction capabilities [1], [2], [3], [4], [5], [6], [7], [8]. Most of the work deals with the physical problems of interaction, especially by means of robot design and control. They focus on the realization of so called human-friendly robots by combining in a bottom-up approach suitable actuation technologies with advanced control algorithms, reactive motion generators, and path planning algorithms. The aim is to achieve safe, intuitive, and high performance physical interaction schemes.

In the line of this research robot safety is of highest priority and still needs large research efforts. Up to now, robot safety was mostly an exclusive topic for applications involving heavy machinery and no physical human-robot interaction. In direct consequence current standards are tailored to certainly exclude the human from the robot workspace and solve the safety problem by segregation. However, recently first efforts were undertaken to shift focus and consider the close cooperation of human and robot. This necessitates fundamentally different approaches and forces the standardization bodies to specify new standards, which are suited for regulating HRI. In this sense we show our latest setups for driving the biomechanical investigation of human injury in robotics further and extending our previous work in this field [9]. We believe it is absolutely crucial to provide this basic research in order to enable safe human-robot interaction and finally release new robot standards that are suitable for pHRI.

Apart from these biomechanical aspects, which directly aim at an investigation of injury caused by human-robot impacts, we present our latest results for reactive obstacle avoidance.

In addition to generating the basic analysis and the motion control methods, a concept for a human-friendly architecture for the DLR Lightweight Robot III is given. The focus of this line of research is to bring a set of sophisticated control features to such integration and versatility that they can be used effectively in a complex application. This incorporates seamless switching between autonomy without human presence, physical Human-Robot Interaction (pHRI), and autonomy under human presence.

The paper is organized as follows. Section II surveys our current efforts in investigating human injury in robotics. Section III describes methods for reactive collision avoidance, followed by Section IV that presents the human-friendly architecture of the LWR-III. Finally, Section V concludes the paper.

II. Injury analysis

In previous work [9], [10], [11] we analyzed the effects of robot mass and inertia during human-robot impacts for various situations and were able to gain fundamental insight into the problem. In order to extend the analysis and continue our efforts to provide extensive data basis to the robotics community, we have set up two different impact testing setups in our lab. These serve the general analysis of human injury in the robotics context.

A. Towards standardized crash-testing

The worst-case injury of humans during blunt robot-human collisions was comprehensively investigated in [9], [10], [11]. However, there are still numerous open issues and it is still an unsolved problem how to standardize safety related analysis of a robot. In order to qualify a robot for industrial and domestic applications incorporating human-robot interaction, we intend to widen our previous analysis such that all relevant factors will be taken into consideration. Unfortunately, it is very time consuming to use standard automobile measurement tools for our purposes since they were not designed for robotic requirements. Conventional

S. Haddadin, S. Parusel, J. Vogel, T. Rokahr, R. Belder, A. Albu-Schäffer, and G. Hirzinger are with Institute of Robotics and Mechatronics, DLR - German Aerospace Center, Wessling, Germany sami.haddadin@dlr.de

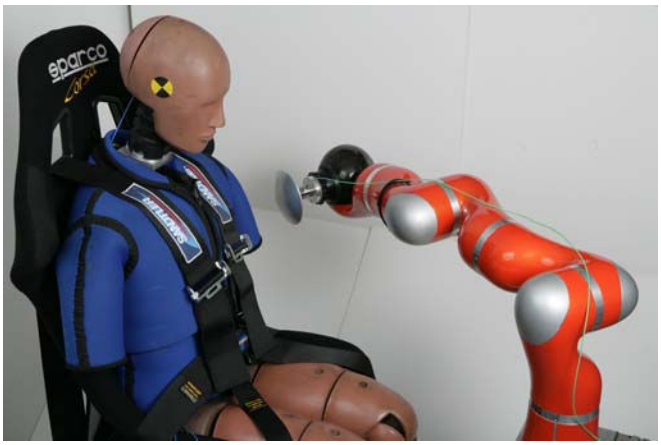

Fig. 1. Side impact dummy with LWR-III.

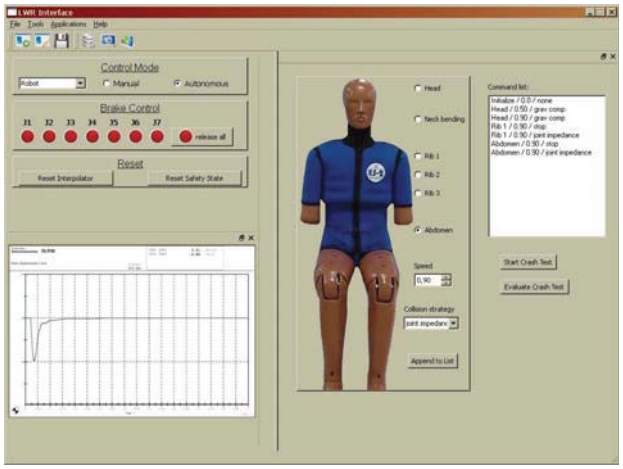

Fig. 2. Graphical user interface for automatic robot crash-testing.

automobile crash-tests are timely limited one-shot sensing processes (a car is only crashed once and is then at least partially damaged) and each test is an entire measurement session. For systematically analyzing the injury potential of a robot with crash-test dummies, it is important to be able to launch longer testing sequences of recurring patterns and adjustable experiment parameters. Therefore, we developed a setup for carrying out large series impact tests with a standard Euro-Sid II crash-test dummy and a LWR-III. This includes automated robot crash-testing and evaluation that is well suited for our demands, c.f. Fig. 1 -3.

The side impact crash-test dummy is equipped with numerous high speed sensors for measuring injury severity during impacts for particular body parts. They range from head acceleration to rib deflection and abdominal contact forces. The sensor signals are recorded at a rate of 20 kHz via a National Instruments DAQ Card running on a QNX real-time pc. In order to unify crash-test procedures in our setup, there are three parameters used for specification of the crash-test scenario. **First**, it is defined which body part of the dummy the robot shall collide with. Available impact regions are:

- head
- head impact with neck bending
- upper rib
- middle rib
- lower rib
- abdomen

The **second** parameter determines the impact velocity, ranging up to ≈ 2.5 m/s tip velocity.

As described in [12] we developed different strategies for collision reaction tailored and implemented for the LWR-III:

- proceed desired motion
- stop
- switch to torque control with gravity compensation
- admittance based strategy
- joint impedance control
- Cartesian impedance control

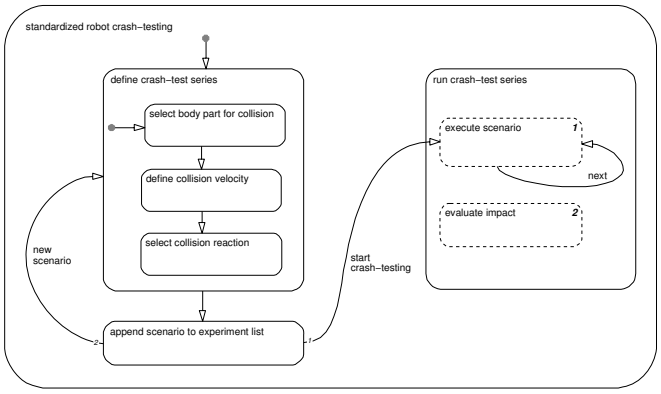

Fig. 3. State machine of the LWR-III crash-testing session.

The **third** parameter activates one of the available strategies. These settings are defined by the user with the help of a graphical user interface (GUI) shown in Fig. 2. Besides the crash-test parameterization, the GUI provides the user an easy to use access to the main functionalities of the LWR-III. The crash-test parameterization dialog is simple to use and designed for automatic processing and analysis of large-scale test series.

To ease the process of sequential crash-test execution, it is possible to set up a list of individually parameterized scenarios. The currently selected parameterization can be appended to the experiment list. By initiating the testing process, the robot motion control and data recording are automatically started.

The recorded data is then transferred from the local harddrive of the real-time system to a target pc. After the crash-tests are finished, the sensor data is used to calculate injury severity indices, which are then displayed and stored as well. The overall work flow is depicted in Fig. 3.

To sum up, the developed setup makes it possible to execute large scale robot crash-test in a simple to use and reproducible fashion.

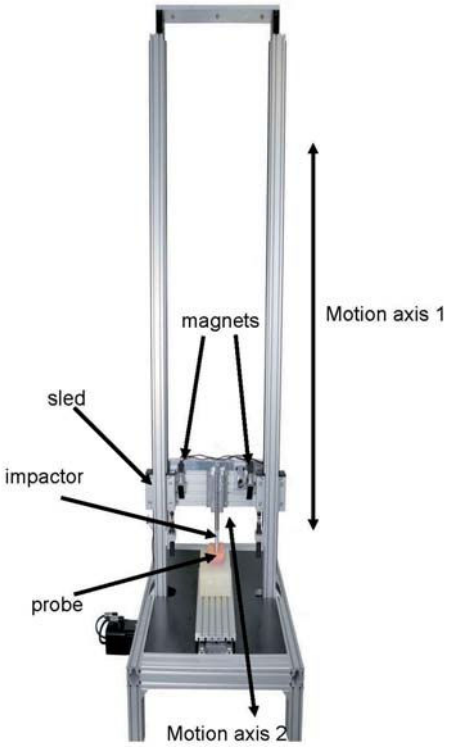

Fig. 4. Drop testing setup for analysis of soft-tissue injury.

B. Analysis of soft-tissue injury

In [13] we have analyzed the performance of our collision detection and reaction schemes [14], [12] with swine experiments and analyzed the basic behavior of soft-tissue collisions during cutting and stabbing. In future work we extend our safety investigations in this field to other usually non-lethal injuries as abrasions and contusions. For this purpose a drop test setup was developed (see Fig. 4) for analyzing the effect different contact characteristics have on biological soft-tissue and use the outcome of these tests for further improving the safety characteristics of robots. The setup serves as a general platform for acquiring biomechanical injury data, which relates typical soft-tissue injury to contact force, impact stress, velocity, acceleration, and impact mass. In order to vary the contact dynamics, it is possible to mount different contact profiles and attach additional masses. The sled itself is mechanically decoupled from the impactor and therefore does not affect the collision dynamics. The setup is designed such that one may run with the motor directly moving the sled, making it possible to penetrate the tissue with desired velocity profiles. Alternatively, the sled is automatically released at a freely chosen height of up to $\approx$ 2 m in order to achieve higher impact velocities. Furthermore, a second motor moves the probe table in horizontal direction, such that for every test an unaffected tissue area is collided with. The generated insights will serve as a foundation for further experiments with the LWR-III to complete the analysis on soft-tissue injury given in [13].

Apart from investigating of intrinsic collision behavior during robot-human impacts and the analysis of collision detection and reaction schemes for their quantitative safety evaluation, it is important to provide the robot with reactive real-time motion schemes. These have to be capable to use different sensor inputs for avoiding collisions, or in case of unwanted contact retract from it. In the next section we describe two of our methods for collision avoidance.

III. Reactive real-time motion generation for collision avoidance

The first method we discuss is trajectory scaling. This is an algorithm providing geometric task consistency during obstacle avoidance, i.e. the robot stays on its desired trajectory when avoiding contact. The second method is a modified version of the circular field method [15], which we use for calculating virtual forces that can be used with the task relaxing motion scheme developed in [16]. This method produces more intuitive behavior compared to classical potential field methods [17], since it does not inherently create repulsive forces independently from the relative velocity between robot and object. For both methods we present measurements, outlining the effectiveness for real-time collision avoidance. In the last part of this section we show how the tactile information, provided by the joint torque sensors of the LWR-III, can be used to place virtual objects based on collision information. After this update in the environment model the robot can perform a collision free path.

A. Timely trajectory scaling

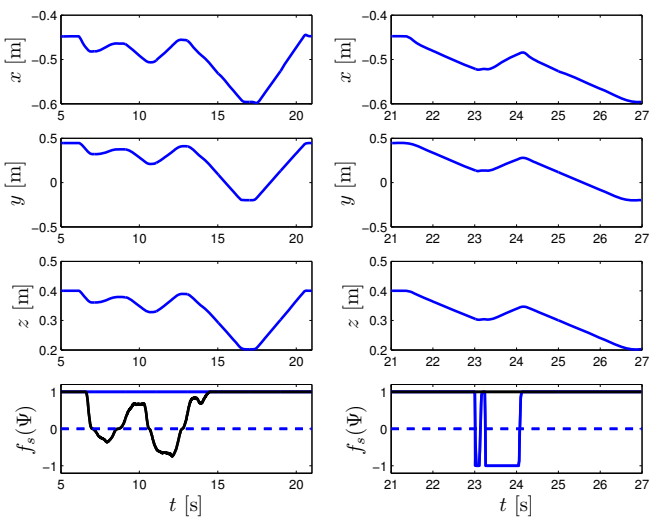

Fig. 5. Trajectory scaling with physical (left) and virtual (left) residuals.

Apart from utilizing physical contact signals as e.g. in [14], [12], [18] for collision retraction/recovery, it is important to use the proximity information of the robot structure with respect to its environment and the human. As described in [12] trajectory scaling can be used to have task consistent compliant behavior during a motion while

the robot approaches the human or vice versa. We used the human proximity and the external forces, estimated by our disturbance observer [12] in combination with a properly designed shaping function, to scale the dynamics of the trajectory execution. This means to continuously slow down, stop, and revert the robot along its desired trajectory, depending on the magnitude of the disturbance. This can simply be done by re-defining the interpolation time as

$$t_i := t_{i-1} + f_s(\Psi(\hat{\mathbf{r}}_i))\Delta t. \tag{1}$$

The trajectory scaling input $\Psi(\hat{\mathbf{r}}_i)$ is based on appropriately shaped residual $\hat{\mathbf{r}}_i$, which is a normalized representations of the disturbance signal[1]. As a disturbance we may e.g. use any distance between control points of interest. Since the plain distance is not an appropriate residual, we use the proximity

$$\rho_\gamma = \left\{ \begin{array}{cc} 0 & d_{\text{bound}} < d \\ d_{\text{bound}} - d & d_{\text{bound}} \geq d. \end{array} \right) \tag{2}$$

instead. d_{bound} is the safety boundary of a body and d its distance to another object of interest. The given formulation of proximity is utilized to calculate virtual forces, which are then transformed into the particular residuals. Figure 5 shows a measurement for a Cartesian motion with trajectory scaling being activated for both, contact force and human proximity to the Tool Center Point (TCP) of the robot. On the left side the translation behavior is shown for the consecutive goal configurations $\mathbf{x}_{d,1} = [-0.45\ 0.5\ 0.4]$ and $\mathbf{x}_{d,2} = [-0.6\ -0.2\ 0.2]$. During the motion towards the first goal a human pushes against the robot, causing the residual to decrease (slow down), become zero (stop) and then negative (drive backwards) according to the residual magnitude. A similar behavior is shown for proximity measurements using an ART[2] passive marker tracking system for feeding the robot with the human pose. If the residual $\hat{r}_i = 0$, the interpolation scaling $f_s = 1$, i.e. normal interpolation time is active. Important to notice is the coordinated behavior in all axes caused by the scaling of the scalar time. Since it is unique in the interpolation process, it consistently affects all degrees of freedom.

B. Circular fields

In commonly used motion generation methods like e.g. potential fields, obstacles as the one depicted in Fig. 7 usually lead to a dead-end in the created trajectory. By applying a modified version of the method based on circular fields [15] passing such obstacles can be tackled. The trajectory in Fig 7 (pink line), associated with a virtual mass particle linked to the goal configuration via a simple attractor, is computed in realtime. It is affected by virtual forces, which are calculated by the following algorithm.

[1]For details on the method please refer to [12].

[2]**A**dvanced **R**ealtime **T**racking

1) Algorithm: The force acting on the virtual particle with position $\mathbf{x}_d$, caused by the discretized virtual objects and the goal attractor, is defined as

$$\mathbf{F} = \mathbf{F}_a + \mathbf{F}_d + \sum_j \mathbf{F}_{ob,j}. \tag{3}$$

The attractor force $\mathbf{F}_a$ drives the robot to the target configuration $\mathbf{x}_d^*$. $\mathbf{F}_d$ is the damping force that is proportional to $\dot{\mathbf{x}}_d$. $\mathbf{F}_{ob,j}$ is the particle force in the circular field $\mathbf{B}$, which depends on $\dot{\mathbf{x}}_d$. Analogous to the Lorentz force in electrodynamics, the force $\mathbf{F}_{ob,j}$ is calculated from the cross product of particle velocity and the associated circular field. More complex shapes are defined by multiple surface elements, with each one having its own local circular field $\mathbf{B}_i$. The resulting force is defined as

$$\mathbf{F}_{ob,j} := \sum_i \mathbf{B}_i \times \dot{\mathbf{x}}_d. \tag{4}$$

The local circular field $\mathbf{B}_i$ of each surface element, acting on the virtual particle is defined as

$$\mathbf{B}_i := I_K \frac{(\mathbf{r}_j \times \mathbf{n}_i) \times \frac{\dot{\mathbf{x}}_d}{||\dot{\mathbf{x}}_d||}}{l_i^2}\, \mathrm{d}a_i, \tag{5}$$

where I_K is the virtual current, $\mathbf{n}_i$ is the normal of surface element i, $l_i = ||\mathbf{x}_d - \mathbf{x}_{\mathbf{n}_i}||$ is the distance of the current desired configuration $\mathbf{x}_d^*$ to the surface element, and $\mathbf{x}_{\mathbf{n}_i}$ is the position of the surface element. $\mathbf{r}_j$ is the field rotation vector for an obstacle, which is defined as

$$\mathbf{r}_j := \frac{\mathbf{d}_j \times \mathbf{b}}{||\mathbf{d}_j \times \mathbf{b}||}, \tag{6}$$

with $\mathbf{b}$ being the goal vector

$$\mathbf{b} := \mathbf{x}_d^* - \mathbf{x}_d. \tag{7}$$

Furthermore, $\mathbf{d}_j$ is the shortest distance between the center of mass of an object $\mathbf{c}_{og,j}$ and the goal vector $\mathbf{b}$:

$$\mathbf{d}_j = \mathbf{x}_d + \mathbf{b}\frac{(\mathbf{c}_{og,j} - \mathbf{x}_d) \cdot \mathbf{b}}{||\mathbf{b}||^2} - \mathbf{c}_{og,j} \tag{8}$$

The main difference between classical potential field approaches and the circular field method is that the former is a purely geometric scheme and does e.g. not take into account the velocity of the robot or other objects. Furthermore, the forces have always repulsive character. Circular fields on the other hand use also the dynamic state of the particle. In addition, the virtual forces act perpendicular to the velocity vector of the robot, therefore generating deviating behavior instead of purely repulsive forces.

2) 2D fields: In Figure 6 a 2D example of a narrow passage problem is shown. The polygons are randomly generated and six sample steps from the full simulation were chosen to indicate the performance of the method. The virtual particle has a limited view range, which is indicated by the virtual forces calculated for the discretized surface elements. The resulting external force (red arrow) acting on the virtual particle and the attractor force (green) are shown. Furthermore, the associated current direction is indicated

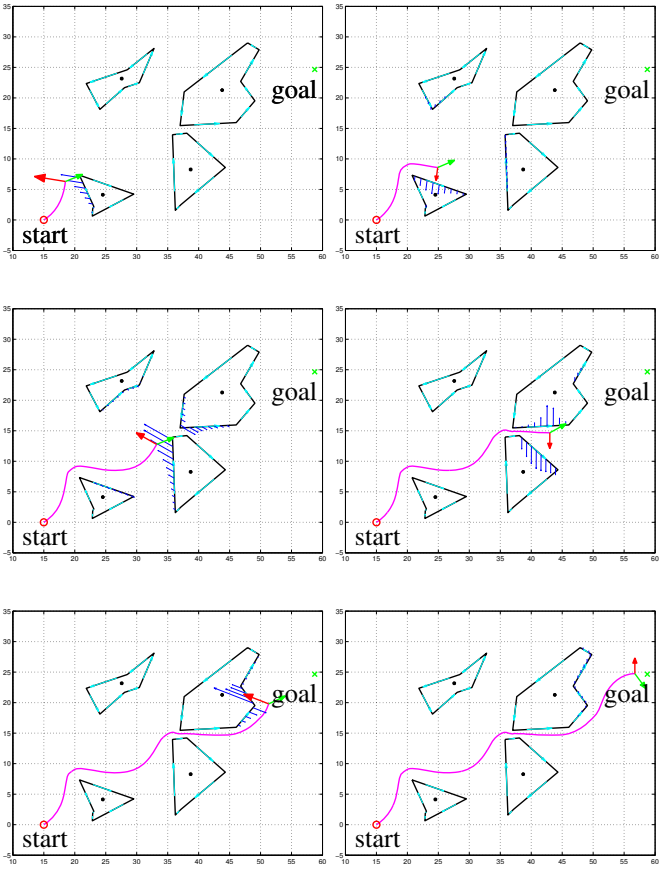

Fig. 6. 2D example for the circular field method surpassing a narrow passage.

on the object borders. In contrast to potential fields, which always generate repulsive forces in normal direction to the surface element, this approach generates a very intuitive force response for the presented problem, smoothly guiding the robot to the goal.

3) 3D fields: To show the capabilities of this algorithm in 3D, we simulated more complex problems, c.f. Fig. 7-Fig. 9.

Figure 7 shows a 3D trap from different angles. The robot has a limited view and thus enters the trap. However, after it is able to sense the walls of the object, the robot escapes and converges to the goal.

Figure 8 and Fig. 9 depict a box with only a single small entry. Even though the robot has only very limited view (indicated by the sparsely visualized grey spheres), it is able to escape and converge very smoothly to the goal.

As one can see from the presented simulations, the collision avoidance based on circular fields is also able to cope with more complex obstacles with local minima and non trivial geometries.

Finally, we have carried out the measurements depicted in Fig. 10, showing the full generated path of the robot and the virtual forces that are calculated for a single time step. The object is a human wrist, whose position was again measured via passive marker tracking. The field of view

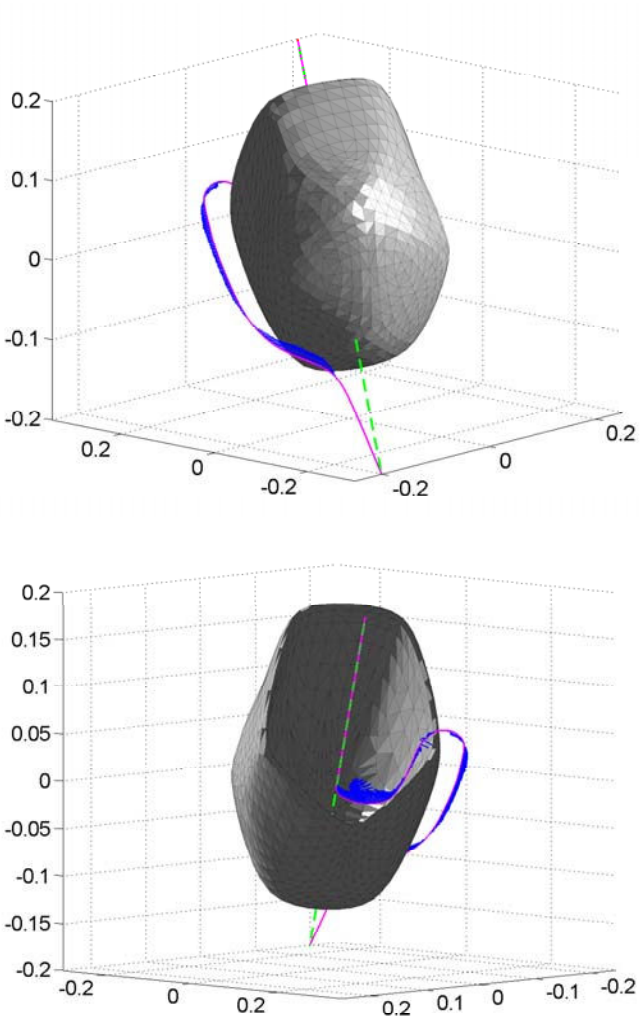

Fig. 7. Usage of circular fields to reactively pass a dead end.

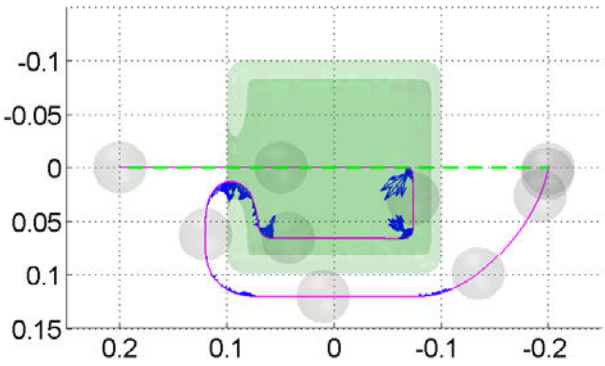

Fig. 8. Usage of circular fields to reactively pass a complex dead end (top view).

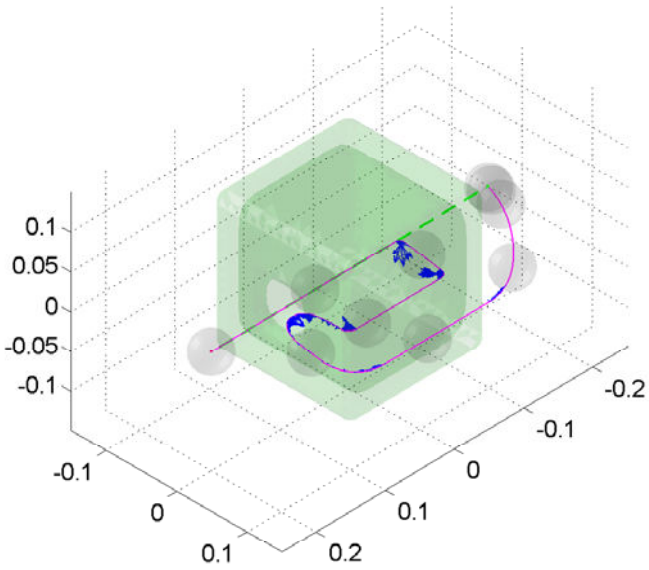

Fig. 9. Usage of circular fields to reactively pass a complex dead end (3D view).

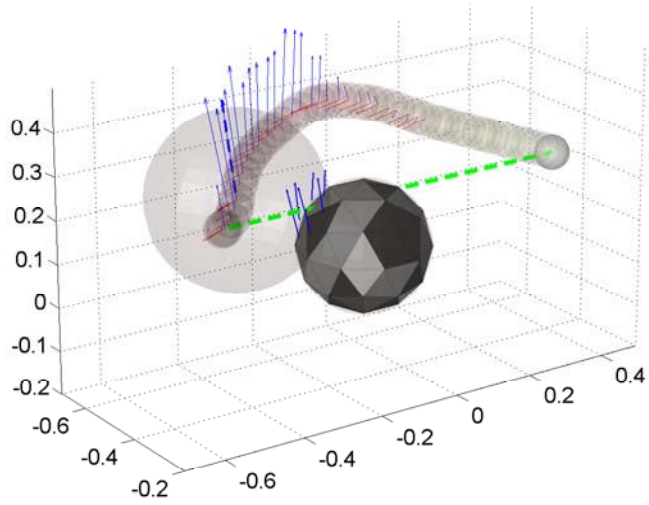

Fig. 10. Circular fields for reactively avoiding visually tracked objects in the robots workspace.

Fig. 11. Tactile exploration

for this instant is the grey sphere. As also shown in the accompanying video, the robot is able to surpass the human very nicely and carries out no unexpected or sudden jerky motions, providing subjectively very convenient avoidance behavior.

Next, we discuss the usage of force information for tactile exploration and its embedding into collision avoidance.

C. Tactile exploration

The contact forces along the structure are provided via an accurate estimation by a nonlinear disturbance observer unit for estimating the external joint torques $\boldsymbol{\tau}_{\mathbf{ext}} \in \mathbb{R}^n$. Its output is the first order filtered version of $\hat{\boldsymbol{\tau}}_{\mathbf{ext}} \in \mathbb{R}^n$, see [18], [12]. These torques can then be transformed into estimations of external forces by

$$\hat{\mathbf{f}}_{\text{ext}} = (J^T)^{\#} \hat{\boldsymbol{\tau}}_{\mathbf{ext}}, \tag{9}$$

where $J \in \mathbb{R}^{n \times m}$ is the contact Jacobian. This force estimation is now available for integration into the task space avoidance since we can identify the magnitude and direction of the contact force.

Figure 11 shows an image series of a collision experiment with the LWR-III and an aluminum block.

Because the robot is not aware of the object, they collide (3rd image). In order to prevent further impact with the object, the robot places virtual objects in the direction of the external force. These ensure that no second collision in the vicinity of the preceding one may take place anymore. However, since no prior knowledge is given, the robot tries

to circumvent the virtual objects and collides several times into the obstacle. After building up an increasing number of virtual objects, the robot is finally able to reach its goal configuration. From now on one will be able to prevent any further collision with the aluminum block.

Next, we explain the main ideas of our human-friendly control architecture for the LWR-III.

IV. Human-friendly control architecture

Currently, industrial settings incorporate, in most cases, simple sequences of tasks with static execution order, allowing sometimes some binary branching. Fault tolerance during task execution is, apart from certain counterexamples [3], usually not an issue due to the well designed environment. In [19] we proposed an integrative and flexible approach to carry out the desired task in a very robust yet efficient way. In this paper we give more details on the real-time robot control architecture, which was only surveyed in the cited work. Figure. 12 depicts the basic structure of our human-friendly control architecture. It consists of four central entities for robot control (the TCU is not shown for brevity):

1) Task control unit (TCU)
2) Robot control unit (RCU)
   a) Safety control unit (SCU)
   b) Motion control unit (MCU)

The first two units serve as the general interface to the robot and communicate with each other via asynchronous protocols. The TCU is the general state based control entity for gathering non-real-time data and providing the correct nominal behavior changes on an abstract level to the RCU. The RCU runs in the same clock rate as the robot, assigning control, motion generation and safety methods. Furthermore, it interprets and validates the selection of behaviors from the TCU, while preventing incorrect combinations with respect to the actual functional mode (see below). The SCU serves as an underlying safety layer below the RCU, which combines all low level safety behaviors and activates them consistently. The Motion Control Unit, which is supervised by the SCU, is responsible for appropriately changing the control and motion behavior of the robot. SCU and MCU are both implemented as hybrid state machines.The SCU supervises information from various sources and routes all control sinks. It decides state dependent whether environmental sensor input is used and which reactive schemes for environment and robot protection are activated. A central element used for the overall supervision process is the already mentioned disturbance observer and the associated collision detection unit. They provide information about external forces acting along the entire robot structure and are used to classify collisions into different fault categories.

In the SCU we embed the overall operational mode of the robot. This defines the set of feasible actions, mainly depending on the current human state. As shown in Fig. 12 we distinguish between four major functional modes of the robot in a co-worker scenario:

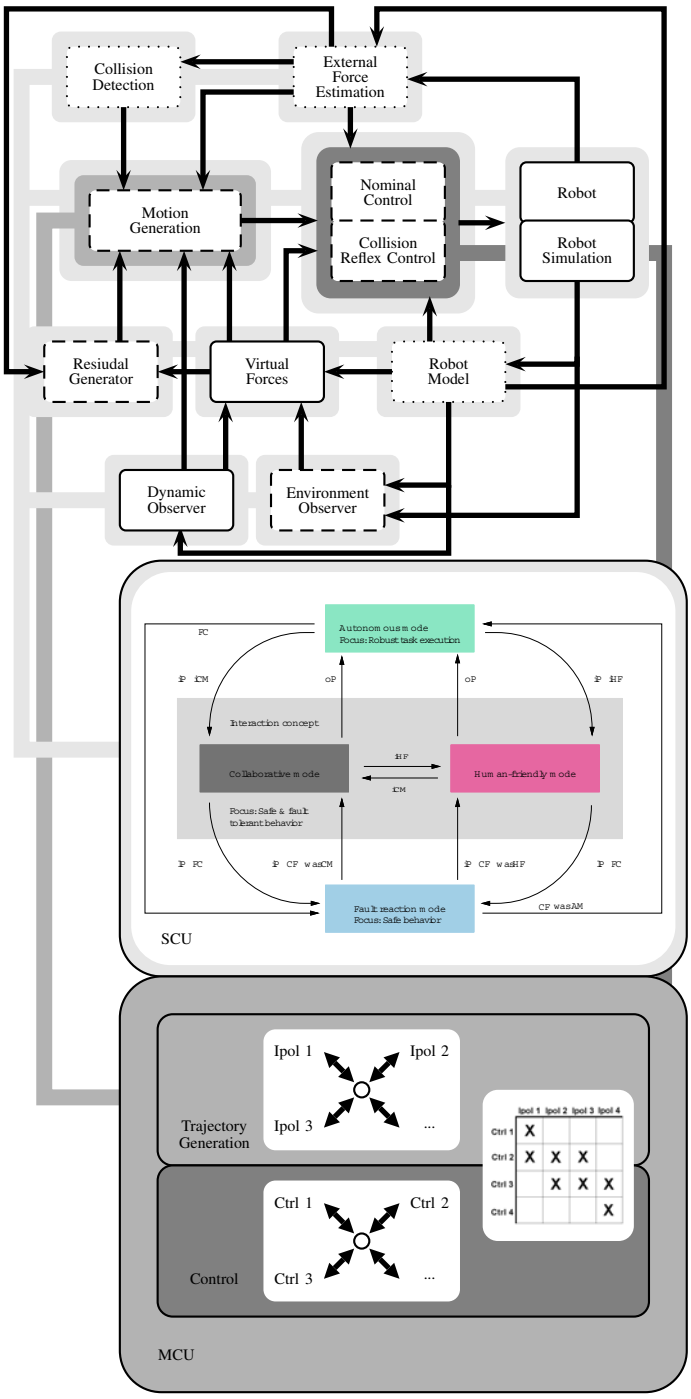

Fig. 12. Overview Robot Control Unit (RCU). In the upper part the available methods are shown. The thin borders indicate the type of the particular scheme with respect to their safety role: source (dotted line), sink (dashed line) or both (solid line). In the middle part the Safety Control Unit (SCU) is shown, which switches depending on the human information (FC = Fault Condition, CF = Confirm, oP = out of Perception, iP = in Perception, iCM = in Collaborative Mode, iHF = in Human Friendly Mode) between its operational modes. The Motion Control Unit (MCU) manages the consistent control of motion and control schemes.

[3]Checking for a successful grasp is e.g. commonly used.

1) **Autonomous task execution:** autonomous mode in human absence
2) **Human-friendly behavior:** autonomous mode in human presence
3) **Co-Worker behavior:** cooperation with human in the loop
4) **Fault reaction behavior:** safe fault behavior with and without human in the loop

V. Conclusion

In this overview paper we outlined various aspects of our analysis for injury investigation in robotics, the reactive motion control methods we developed, and the human-friendly architecture of the LWR-III. Our goal is to provide a thorough analysis of the capabilities of the robot with respect to intrinsic impact characteristics and furthermore equip it with advanced control schemes for fault tolerant and human-friendly behavior. The chosen examples showcase the capabilities of the developed algorithms and how they contribute to more flexible and reactive robot behavior incorporating multiple sensor inputs. The described human-friendly architecture for unifying the large bundle of motion control features of the LWR-III visualizes our concept of how to use the complex sensory input and where to integrate it into the robot control architecture.

A **video** accompanying the paper shows some experiments described in the paper.

References

[1] A. Bicchi and G. Tonietti, "Fast and Soft Arm Tactics: Dealing with the Safety-Performance Trade-Off in Robot Arms Design and Control," *IEEE Robotics and Automation Mag.*, vol. 11, pp. 22–33, 2004.

[2] M. Zinn, O. Khatib, and B. Roth, "A New Actuation Approach for Human Friendly Robot Design," *Int. J. of Robotics Research*, vol. 23, pp. 379–398, 2004.

[3] H.-O. Lim and K. Tanie, "Human Safety Mechanisms of Human-Friendly Robots: Passive Viscoelastic Trunk and Passively Movable Base," *Int. J. of Robotics Research*, vol. 19, pp. 307–335, 2000.

[4] K. Ikuta, H. Ishii, and M. Nokata, "Safety Evaluation Method of Design and Control for Human-Care Robots," *Int. J. of Robotics Research*, vol. 22, no. 5, pp. 281–298, 2003.

[5] J. Heinzmann and A. Zelinsky, "Quantitative Safety Guarantees for Physical Human-Robot Interaction," *Int. J. of Robotics Research*, vol. 22, no. 7-8, pp. 479–504, 2003.

[6] A. Albu-Schäffer, C. Ott, and G. Hirzinger, "A Unified Passivity-based Control Framework for Position, Torque and Impedance Control of Flexible Joint Robots," *Int. J. of Robotics Research*, vol. 26, pp. 23–39, 2007.

[7] D. Ebert and D. Henrich, "Safe Human-Robot-Cooperation: Image-based Collision Detection for Industrial Robots," *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS2002), Lausanne, Swizerland*, pp. 239– 244, 2002.

[8] D. Kulic and E. Croft, "Pre-collision strategies for human robot interaction," *Autonomous Robots*, vol. 22, no. 2, pp. 149–164, 2007.

[9] S. Haddadin, A. Albu-Schäffer, and G. Hirzinger, "Requirements for Safe Robots: Measurements, Analysis & New Insights," *Int. J. of Robotics Research*, 2009.

[10] S. Haddadin, A. Albu-Schäffer, M. Frommberger, J. Rossmann, and G. Hirzinger, "The "DLR Crash Report": Towards a Standard Crash-Testing Protocol for Robot Safety - Part I: Results," in *IEEE Int. Conf. on Robotics and Automation (ICRA2008), Kobe, Japan*, 2009.

[11] ——, "The "DLR Crash Report": Towards a Standard Crash-Testing Protocol for Robot Safety - Part II: Discussions," *IEEE Int. Conf. on Robotics and Automation (ICRA2008), Kobe, Japan*, 2009.

[12] S. Haddadin, A. Albu-Schäffer, A. De Luca, and G. Hirzinger, "Collision Detection & Reaction: A Contribution to Safe Physical Human-Robot Interaction," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS2008), Nice, France*, 2008, pp. 3356–3363.

[13] S. Haddadin, A. Albu-Schäffer, and G. Hirzinger, "Soft-tissue injury in robotics," *IEEE Int. Conf. on Robotics and Automation (ICRA2010), Anchorage, Alaska*, 2010.

[14] A. De Luca, A. Albu-Schäffer, S. Haddadin, and G. Hirzinger, "Collision Detection and Safe Reaction with the DLR-III Lightweight Manipulator Arm," *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS2006), Beijing, China*, pp. 1623–1630, 2006.

[15] L. Singh, H. Stephanou, and J. Wen, "Real-time robot motion control with circulatory fields," *IEEE Int. Conf. on Robotics and Automation (ICRA1996), Minneapolis, USA*, 1996.

[16] S. Haddadin, H. Urbanek, S. Parusel, D. Burschka, J. Roßmann, A. Albu-Schäffer, and G. Hirzinger, "Real-time reactive motion generation based on variable attractor dynamics and shaped velocities," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS2010), Taipeh, Taiwan*, 2010.

[17] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," vol. 2, 1985, pp. 500–505.

[18] A. De Luca, F. Flacco, R. Schiavi, and A. Bicchi, "Nonlinear decoupled motion-stiffness control and collision detection/reaction for the vsa-ii variable stiffness device," *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS2009), St.Louis, USA*, pp. 5487–5494, 2009.

[19] S. Haddadin, M. Suppa, S. Fuchs, T. Bodenmuller, A. Albu-Schäffer, and G. Hirzinger, "Towards the robotic co-worker," in *International Symposium on Robotics Research (ISRR2009), Lucerne, Switzerland*, 2009.