2011 IEEE International Conference on Robotics and Automation
Shanghai International Conference Center
May 9-13, 2011, Shanghai, China

The DLR Robot Motion Simulator Part I: Design and Setup

Tobias Bellmann, Johann Heindl, Matthias Hellerer, Richard Kuchar, Karan Sharma and Gerd Hirzinger

Abstract— **In recent years a new generation of motion simulators, based on serial kinematics industrial robots, emerged as alternative to the currently prevalent Steward-platforms. This paper presents the newest addition to this: The DLR Robot Motion Simulator.**

Part I covers the design process and gives a detailed introduction of the setup. The overall layout of the simulation platform and its parts is introduced. To meet the requirements of an interactive simulation, a new piloting cell had to be designed using light-weight construction and real time vehicle simulations were required. Another key issue of the whole design is the guarantee of safety. The simulator utilizes a 10m linear axis to increase its workspace to provide a better simulation experience compared to previous designs of motion simulators with serial or parallel kinematics.

Part II introduces a new path planning algorithm for the kinematically redundant simulator platform, required for the generation of appropriate motion cues. Furthermore the application artistic-flight simulation is demonstrated and validated.

I. INTRODUCTION

Motion simulators aim to create the sensation of operating or being on a real vehicle within a simulation environment. To achieve this goal, the human senses have to be provided with sensory input as real as possible. Typically, simulations focus on vision and hearing and in the case of motion simulators also include motion cues.

In the history of motion simulators a plurality of mechanical configurations have been developed and used for the creation of motion cues, to generate the impression of large scale vehicle movements. Since the presentation of the parallel Stewart-platform [1] in 1965 it has become the prevalent configuration for motion simulators, widely used as training simulators or for fun-rides. Parallel configurations can handle heavy loads e.g. complete cars [2] or aircraft cabins, but their workspace is severely limited.

Motion simulators are already in wide spread use for a multitude of purposes. They provide a safe, cheap, flexible and easily controllable surrounding to train military and civilian pilots on all kinds of vehicles in both normal operation and the handling of extreme cases, which could not be performed safely in a real world setting. Furthermore they're a common tool for researchers to find out how accelerations and angular velocities are percepted by human beings, to test new designs for vehicles and many more scientific applications. Recently they have also found application as fun-rides in amusement parks.

The authors are with Institute of Robotics and Mechatronics, German Aerospace Center, 82234 Oberpfaffenhofen, Germany Tobias.Bellmann@DLR.de

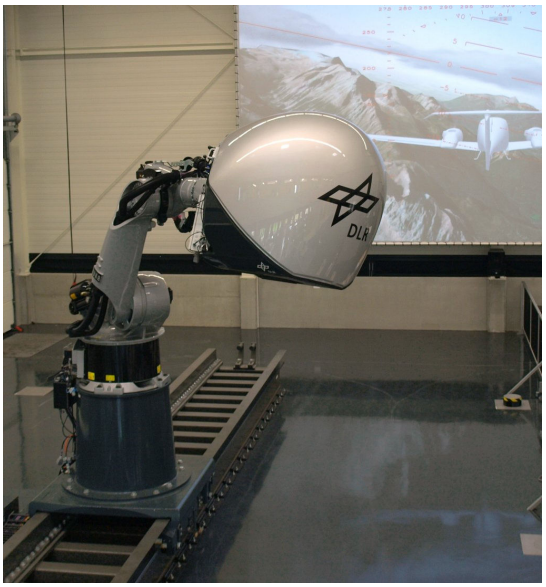

Fig. 1. The DLR Robot Motion Simulator during operation. The projected video in the background shows the simulation environment as seen by the simulator pilot.

Serial configurations used as simulator mechanics are relatively new in the field of simulator engineering. On the Automatica exhibition 2004 the German Aerospace Center (Deutsches Zentrum für Luft- und Raumfahrt - DLR) presented a non-interactive simulator, based on a KUKA KR500/2 TÜV industrial robot with 6 DOFs [3] in cooperation with KUKA Robotics. In the following years, path-planning algorithms for interactive motion simulation with non redundant industrial robots have been the focus of several publications [4][5] and driving simulations have been implemented as use-cases [6][7].

Serial mechanical configurations like industrial robots, while on the one hand having a large workspace, are limited in size and weight of their payload on the other hand (e.g. 1000kg KUKA KR1000 Titan [8]), compared to parallel configurations. Fortunately, industrial robots are mass products and therefore they're available at much lower cost than Stewart-platforms. Yet they introduce a series of new challenges, where the most demanding one is the path-planning for the robot to generate the correct motion cues for the pilot. A solution for this problem is presented in part II of the paper. This first part presents a novel design for a redundant serial kinematics based motion simulator: The DLR Robot Motion Simulator.

978-1-61284-385-8/11/$26.00 ©2011 IEEE

II. DESIGN MOTIVATIONS

The design of the DLR Robot Motion Simulator is the direct result of the experiences made with earlier versions of the simulator at the institute. Although the workspace for linear movements is considerably larger compared to the workspace of Stewart platforms of the same weight and size, an additional linear axis is an easily available way to increase the workspace in a defined direction. Those linear axes are an omnipresent element of robotic production lines and are available even for heavy robots usable for motion simulation. The experiences made with the development of simulator cells for offline simulators showed the necessity for a cell concept with easy accessible control instruments. In the existing simulator cell, a safety harness provides a firm hold on the pilot, but also restricts its movability and possibilities to interact with instruments. Furthermore, the plurality of applications demand a freely configurable instrument package for a multitude of simulation scenarios.

In this paper, both the existing and a newly implemented cell concept covering these issues, will be shown. In order to allow the simulation of different scenarios and vehicle types, the software system controlling the robot must be modular. The previous simulator applications have led to the presented software system design, using modular simulation components, which can be replaced and reused easily. The transportation of human beings demands highest standards from the safety systems controlling the simulator. To assure the safety of the pilot and the people in the vicinity of the simulator, several security systems have been implemented during the development of the earlier offline simulator applications. As these systems are used for public entertainment with fixed scenarios, the security systems for the new design had to be extended in order to fit the needs of an interactive motion simulator within a laboratory environment.

III. THE DLR ROBOT MOTION SIMULATOR SETUP

The general setup can be divided into four major components, connected with each other: Control room, motion platform, piloting cell, safety systems.

A. Control room

In the control room the entire control and simulation infrastructure is concentrated. The vehicle dynamics simulation is performed here as well as the path planning and its multimedia augmentation. In case it is required by the installed simulation cell instrumentation package, further hardware can be integrated easily. Primarily the control room is used to initiate and shut down both the robot and the simulation. During the motion simulator run, the operator also monitors the current state of all components and the pilot as well as the safety systems from here.

B. Motion platform

The main purpose of the DLR Robot Motion Simulator is the transportation and interaction with living persons. To ensure their safety and thereby changing the robot's legal definition under german law from "six axes kinematics" ISO EN 10218 [9] to "fun ride as part of a temporary building" DIN 4112 [10]), certain criteria have to be certified by the german Technical Inspection Authority (Technischer Überwachungs-Verein - TÜV). At the time when this paper was written, only one commercially available robot has the required certification: the KUKA KR500/2 TÜV called Robocoaster [11]. For this reason the Robocoaster was chosen to be the main mechanical component of the motion platform.

However the standard Robocoaster working space is rather limited for safety reasons. Since the path planning requires more flexibility, the original hardware stops had to be moved. This was done in close cooperation with KUKA Robotics so the safety would not be affected. The new joint limits are listed in table I, the according locations and turning directions are explained in figure 3.

Additionally, the robot has been mounted on a KUKA KL3000 linear axis, with a total length of 10m, to increase the workspace for linear movements in a selected direction dramatically.

Fig. 2. The DLR Robot Motion Simulator is supervised from the control room. On the left side, the safety control console with emergency stop is visible, on the right side the KUKA Robot Control (KRC) is running a program.

TABLE I

ENHANCED ROBOCOASTER SOFTWARE JOINT LIMITS

Axis	minimum joint limit	maximum joint limit	maximum acceleration	maximum speed
q_1	$-133°$	$+133°$	$97\ °/s^2$	$70\ °/s$
q_2	$-128°$	$-42°$	$72\ °/s^2$	$56\ °/s$
q_3	$-30°$	$+77°$	$128\ °/s^2$	$69\ °/s$
q_4	$-90°$	$+90°$	$42\ °/s^2$	$76\ °/s$
q_5	$-58°$	$+58°$	$95\ °/s^2$	$76\ °/s$
q_6	$-90°$	$+90°$	$67\ °/s^2$	$120\ °/s$
q_7	$-9.41m$	$+0.431m$	$1.2\ m/s^2$	$1.5\ m/s$

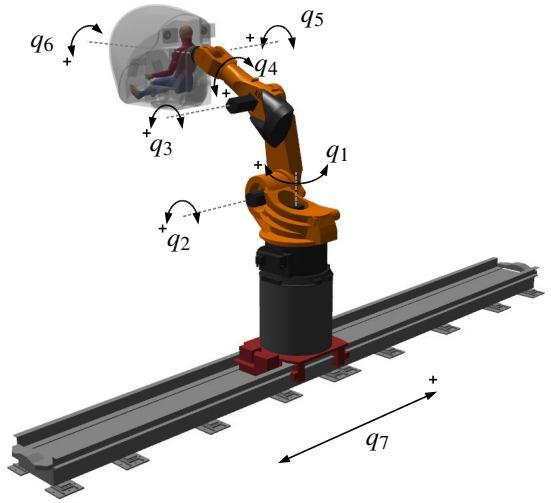

Fig. 3. Schematic showing joint locations and turning directions of the KUKA KR500/2 [11]

C. Simulator cell

The simulator cells are mounted at the end-effector of the robot system, and are connected via Ethernet to the simulation and control systems in the control room. A detailed description of the available simulator cells can be found in section IV.

D. Safety systems

The safety system's primary goal is to protect both the simulator pilot and people nearby from being harmed by the robot. If this can be ensured, it's secondary goal is to protect the robot hardware from being damaged - only if these two points are insured, normal operation is permitted.

To prevent any emergency situations in the first place, the robot workspace is secured by a fence preventing nearby people from entering the workspace. During operation, the gates in the fence are automatically locked. The workspace is also limited by deformable hardware stops at each joint, making it physically impossible for the robot to crash into itself, the surroundings and the floor.

Three actions are possible in the case of an emergency, depending on its cause: If an emergency switch is triggered manually or a person forcefully enters the robot's work space by prying open a gate, the emergency shutdown is triggered. This is the most forceful kind of abort, causing all of the robot's mechanical brakes to be applied instantly. Further steps can then be taken by the operator driving the robot manually. In cases where there is no immediate danger to a person nearby and the simulation pilot is to be evacuated from the piloting cell, the robot returns to its entry position. This can be triggered either by the pilot or operator pressing the return button or by one of the safety sensor for smoke and temperature in the piloting cell. In these last two cases, the power supply to the simulator cell is shut down as well, leaving only the independent intercom system operational. The last safety system installed prevents the robot from leaving its entry position, if any of the following conditions is present: The pilot's safety belt is not fastened, the entry platform is not secured outside the robot workspace or the cell is not closed correctly.

In case of a complete power failure, the robot could get stuck in a position holding the simulator cell upside down. In order to rescue the pilot from this inconvenient position rapidly, the joints can be moved, using a battery powered manipulator, returning the pilot to an upright position from where he can be evacuated using the mobile entry platform. The intercom system in the piloting cell is also backed up by a battery of its own to stay operational in this case.

The values commanded to the robot by the path-planning module are independently checked by two safety systems. First the values sent by the path-planning algorithm are passed to a custom watchdog application. This watchdog is checking, whether or not the remaining distance between the current position and the workspace limit is sufficient for a safe full stop and initiates this full stop the moment the criterion is about to be violated. The second monitoring system is implemented as application for the KRC robot controller and besides double checking the workspace limits also checks maximum joint speeds and accelerations.

IV. SIMULATOR CELLS

In this section, the general design parameters for simulation cells used with the DLR Robot Motion Simulator are introduced. Furthermore, the two existing cell variants are presented and explained in detail.

A. Design constraints

The two crucial design constraints applying to the simulator cell are the limitation of weight and the spatial constraints. The first one is introduced by the specification of the robot system and limited to 500 kg, for the KR500/2 TÜV.

The second constraint depends on the maxmimum and minimum joint angle limits and the working height of the robot above ground. As the cell can rotate around axis 6, a cylinder mounted coaxial to axis 6 has been used to define the allowed dimensions of the cell. For the joint limits of the used robot system (Table I), two critical configurations exist, defining the radius and length of this cylinder :

- Axis 3 and 5 are at their lowest configuration, defining the radius of the cylinder from the endeffector to the robot arm.
- Axis 2 and 3 are at their lowest configuration, defining the length of the cylinder, from the flange to the ground.

Given the maximum axes joint angles $q_{2,max}$, $q_{3,max}$, $q_{5,max}$, the dimensions of the robot l_{23}, l_{31}, l_{35} , l_{56}, r_{arm} and the height of axis 2 above ground h_0, both width and length of the enveloping cylinder can be calculated (see Fig. 4):

$$\begin{aligned} d_{cell} = \quad & \cos(\pi - q_3) l_{31} + cos(\pi/2 - q_3) l_{35} + \\ & cos(\pi/2 - q_3 - q_5) l_{56} - r_{arm} \\ r_{cell} = \quad & d_{cell} / \cos(\pi - q_3 - q_5); \end{aligned}$$

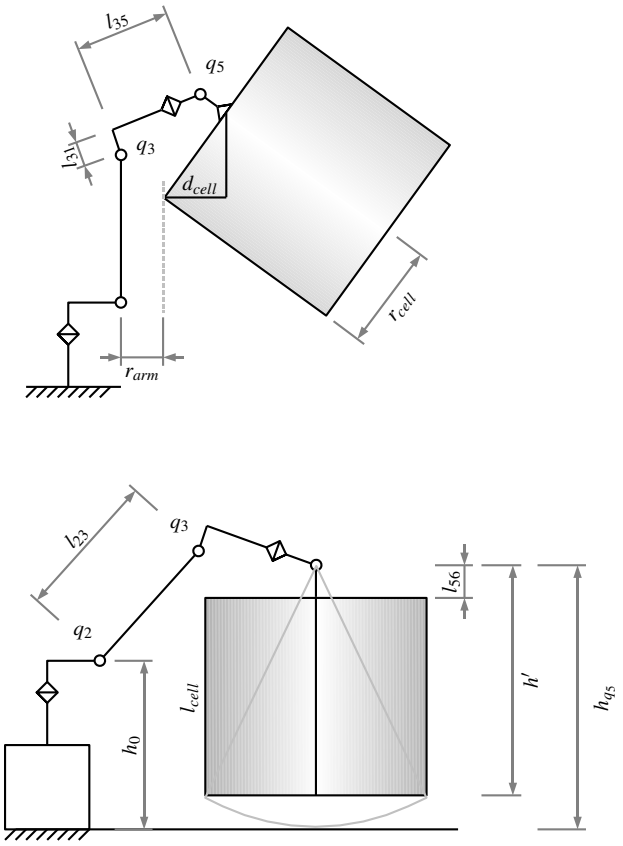

Fig. 4. The spatial constraints of the simulator cell depending on the robot joint angles and its dimensions.

$$
\begin{aligned}
h_{q_5} = \quad & h_0 + sin(-q_2)l_{23} + \\
& sin(-q2 - q3 + \pi/2)l_{31} + sin(-q2 - q3)l_{35} \\
h' = \quad & \sqrt{h_{q_5}^2 - r_{cell}^2} \\
l_{cell} = \quad & h' - l_{56}
\end{aligned}
$$

Using the joint limits from Table I, plus a safety margin considering the deformation of the hardware stops, the spatial constraints of the simulator cell result in $r_{cell} = 0.92m$ and $l_{cell} = 1.56m$.

B. KUKA/DLR Simulator hood

In 2007 DLR developed a dual simulator dome to use it in combination with the standard KUKA Robocoaster seats, shown in figure 5. This so called 4D Simulator has been commercialized and is now in use by amusement parks around the world. It features two seats, comparable to those of a roller coaster, and an integrated 20" TFT-screen as well as a stereo sound system, a surveillance camera with microphone, a smoke sensor and a ventilation system [12]. For interactive simulations this cell also has been equipped with a two axis side-stick.

However, the hood has been designed and optimized for a passive experience. It only covers the pilots head and parts of his or her torso while the limbs are locked outside of the

Fig. 5. The simulator hood as used by the KUKA 4D simulator, with additional side-stick for interactive simulations

dome. This arrangement severely limits the possibilities for the usage of control instruments, required by an interactive system. During the development phase of the DLR Robot Motion Simulator, heavy use has been made of this hood and it has been shown, that the mentioned limitations make a new cell design necessary.

C. Revised, modular design

Fig. 6. The new, modular simulator cell for the DLR Robot Motion Simulator, equipped with the car instrumentation package

In order to approach these problems, a new cell was designed from ground up considering the limitations introduced in section IV-A. To incorporate the overall modular

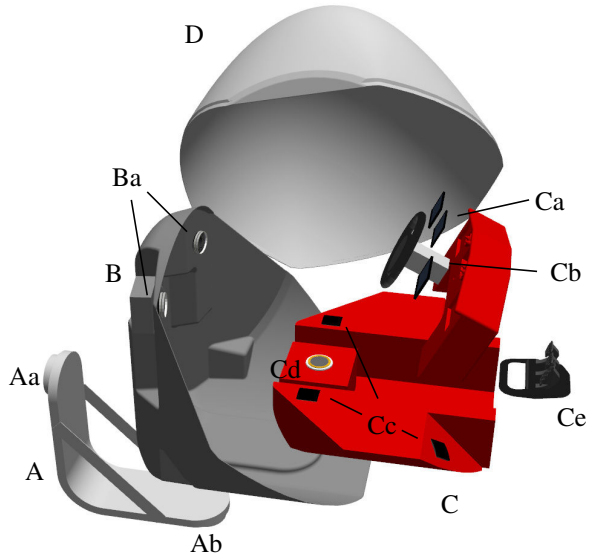

Fig. 7. Components of the new simulator cell (grey) and car instrumentation package (red).

design into the piloting cell it has been split up in two parts: The outer cabin and a replaceable instrumentation package, adaptable for different simulation types, for example driving- or flight-simulation.

D. Cabin

The cabin provides the supporting structure and some basic equipment, common to all possible types of simulations. The simulator forces are applied on the cell by an steel angle bracket, with two flanges used for mounting the cell on the robot, depicted in figure 7 as part A and the flanges as Aa and Ab. The flanges make it possible for the cabin to be mounted in different ways. In earlier drafts the connector system was designed to be even more flexible and possibly motorized, yet these plans were dropped later to reduce weight and to increase the available space. The reason for wanting a flexible connection lies in the robot's serial kinematics structure: The closer a joint is to the cabin, the faster it can move the cell. Attached through the rear flange next to axis 6, the cell is optimized for fast roll movements. Attached through the bottom connector, fast yaw movements can be performed, without reconfiguring other joints. The light weight design and distribution of weight were a crucial part of the cabins design. The weight had to be reduced as far as possible and the remaining weight shifted toward the flanges to reduce the cabins inertia and thereby increasing the simulator's performance.

Hookups for a 5-point safety belt, attached directly to the solid steel frame for maximum robustness are also provided by the cabin, while the seating is part of the instrumentation package.

Additionally the cabin features a smoke detector, a temperature sensor, an intercom system, ventilation and visualization. For the visualization, the dome (part D) double-functions as screen for two projectors, installed at the rear side of the cabin, depicted in figure 7 as Ba. To compensate for the dome's concave, irregular form, the projector images need to be corrected geometrically, this is done by an real-time image processor [13], integrated in the projector compartment. The large screen allows a viewing angle of about 120 [deg] in horizontal direction. Furthermore, 3D stereo vision is supported by using wavelength multiplexing of RGB color triplets in combination with interference filtering glasses [14].

E. Instrumentation package

For different simulation scenarios, different kinds and arrangements of control intruments and gauges are needed for realism. In the new cell design, the cockpit instrumentation is an independent package, that can easily be exchanged for different simulations providing the according control instruments. This allows the DLR Robot Motion Simulator to be used for a multitude of scenarios while implementing a complete replica of the simulated cockpit. The only two connections of the instrument package are defined as 230V AC power supply and an 1000BASE-T ethernet line. All necessary hardware to control the instruments, as computer systems, power supplies etc. are included into the instrumentation package. This allows the double use of the instrumentation packages as fixed space, stand-alone simulators without the motion simulator platform by simply adding a visualization screen in front of it.

While other instrumentation packages are currently designed, the first one is already available: The car simulation instrumentation package, depicted as part C in figure 7. It features three TFT-Touchscreens (Ca), a high torque force-feedback steering wheel (Cb), a 5.1 surround sound system (Cc), a seat-shaker(Cd), three pedals (Ce). The seat-shaker simulates the high frequency agitations, the motion platform is unable to perform, for example the vibrations expirienced when driving on cobble stone pavement.

V. SIMULATOR SOFTWARE INFRASTRUCTURE

The modular design of the DLR Robot Motion Simulator enforces also modularity in the software infrastructure. The software framework used for the control of the robot, the simulation of the vehicles and the visualization of the environment is the subject of this section.

A. Framework

The evaluation of different designs, the simulation of vehicle dynamics and the path planning for the DLR Robot Motion Simulator are done using detailed computer models based on the Modelica modelling language [15]. Furthermore, during the design process, the robot and all of it's parts were completely simulated, including its kinematics and its dynamic behavior. Reusable Modelica libraries, both commercial and open-source, have been developed during the simulator design process. This libraries cover fields as vehicle dynamics (see V-B), inter-system communication, robot dynamics and more. For visualization of the environment and to facilitate the development process by getting

a better overview of the simulations being performed, the *DLR Visualization* Modelica library, was used [16]. This library provides an external viewer software capable of stereo visualization, *DLR SimVis*. A simulation of a hybrid car visualized with *DLR SimVis* can be seen in figure 8.

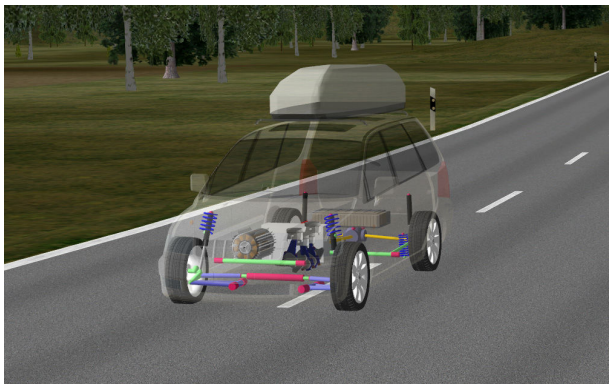

Fig. 8. Visualization of a hybrid car with chassis, suspension, drivetrain and engines being simulated.

All models are subdivided in functional entities in an hierarchical structure. This allows one entity to be easily replaced by another one, inherited by the same base class. For example the path-planning module can be supplied by a file containing test cases or by manual input using a SpaceMouse without any further changes to any other part of the module.

A high level view of two models used for the DLR Robot Motion Simulator is shown in figure 9. Each entity (boxed icons) shown here is again subdivided into multiple parts and replaceable by other functional units.

Switching from pure simulations to a full hardware in the loop control with only a view clicks, is one highlight of the modular design. At large this makes it possible to develop a new implementation for one part, test it in the simulation environment and, when the tests were successful, run it on the real hardware.

The high-level structure of the simulator software components is displayed in figure 10, showing the data flow between the modules. The primary data source is the pilot, generating control signals for vehicle dynamics simulation. The position data $\underline{x}$ and the accelerations of the vehicle $\underline{a}$ generated here are passed on to the multimedia augmentation, responsible for visual and audio rendering, and the path-planning module, generating the robot's joint trajectories $\underline{q}(t)$. These trajectories are passed through the controlling watchdog application before they are sent to the robot controller and executed by the hardware.

B. Vehicle dynamics models

Using the DLR Robot Motion Simulator accordingly requires the application of detailed dynamic system models as input to the trajectory planning process. Currently, two vehicle dynamics frameworks are available for the usage with the DLR Robot Motion Simulator:

A realistic and physically correct simulation of road vehicles has been presented in [6], using the *DLR Vehicle Controls*

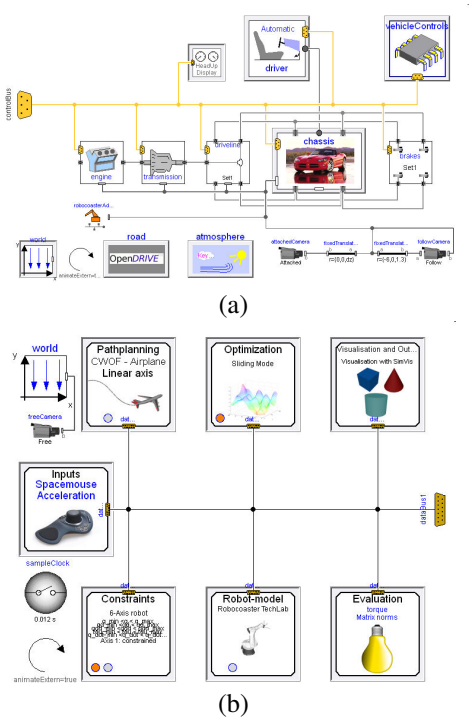

Fig. 9. High level view of two Modelica models. Model (a) shows the top level diagram of a vehicle dynamics model, with replaceable modules for engine, transmission, chassis, etc. Below, model (b) shows the top level diagram of the path-planning module, featuring adaptable washout-filters, robot configurations, visualizations, etc.

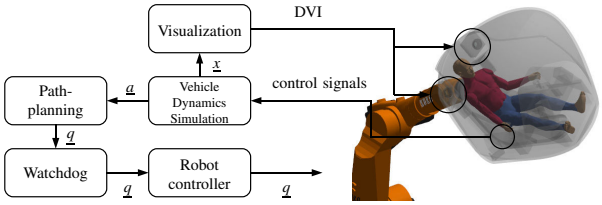

Fig. 10. System overview of the DLR Robot Motion Simulator, including the data flow between the single modules.

Modelica library for the simulation of cars, trucks and other road vehicles. The focus of this library is the accurate simulation of all car or truck components. It includes models for drive trains, engines, transmissions, road contact models and three-dimensional modeled chassis and suspensions.

The second vehicle dynamics library in use features real-time aircraft models, based on the DLR Flight Dynamics library [17][18]. The *DLR Flight Dynamics* library, is also based on the Modelica language and consists of numerous modules allowing the implementation of multi-disciplinary flight dynamics models for a wide variety of applications. These are ranging from real-time simulation models to the automatic generation of inverse models for the deduction of nonlinear control laws. Basic characteristics of the library are the interaction of flight mechanics, structural dynamics and sensor systems, as well as the capability to embed all available elements from other Modelica libraries into a single unified model. The library includes an airframe

module allowing the definition of properties for rigid or flexible airframe objects, aerodynamics-, propulsion- and gear modules are included for the complete description of the aircraft model. Additionally, the ability to specify the entire environmental surrounding of the aircraft (geodetic, gravitational-, magnetic- and atmospheric models), together with the capability to embed sensors for flight control (e.g. the Instrument Landing System - ILS) allows the rapid development of complete scenarios - possibly containing multiple aircraft models with their respective periphery. Models based on the library have been applied and tested in several projects at DLR - e.g. REAL (automatic landing), flight tests of the X31A aircraft with reduced vertical tail and recent flight tests of a new autopilot onboard the DLR ATTAS aircraft have been conducted.

VI. SAFETY TESTING

The simulator setup introduced in this paper was subject to various safety tests to determine the behavior of the safety setup and the resulting accelerations inside the simulator cell. For the same reason, an inertial measurement unit (IMU) was mounted near the usual human head position inside the simulator cell to record the accelerations acting on the head and neck. The most relevant tests, identifying the maximum accelerations during emergency shutdowns and the maximum braking distances are presented in brief as follows:

A. Test procedure

Test 1: Axis 1

A sinusoidal trajectory has been programmed for joint 1, to run at maximum speed. During the run, several emergency scenarios have been provoked, causing an automatic emergency braking by the robot control using the weaker mechanical brakes. The resulting accelerations have been recorded with a sample rate of 100Hz. The resulting acceleration magnitute $|\underline{a}_{sim}|$ can be seen in Fig. 11-Test 1, with accelerations not exceeding 2-3 g.

Test 2: Axis 2, 3 and 5 Sinusoidal trajectories with maximum speed have been programmed with the same frequency for Joints 2, 3 and 5, leading to a combined maximum cartesian velocity in the pilot's cranial direction (z-axis of cell 7). The results can be seen in Fig. 11-Test 2, with accelerations not exceeding 3g.

Test 3: All axes driven simultaneously

The same experiment as mentioned above has been performed with all 7 axis (Robot + Linear axis) of the motion simulator. The Sinusoidal trajectories run all axes at 150% maximum speed and with the same frequency. By commanding a joint speed amplitude higher than the allowed maximum speed (see I), an emergency braking is provoked. With about 5g, this test produced the highest acceleration measured (see Fig 11-Test 3).

Test 4: Braking distance of all axes

In order to test the safety of the software limits, experiments to determine the braking distances have been performed. The software limits have been parametrized in a safe distance to the hardware limits, and a controlled trajectory driving the according joint into the software limit with maximum joint speed has been planned. While the axes 1-6 of the robot have a braking distance of about 1 deg, the linear axis needs a braking distance from 0.4m - 0.6m subject to the motions of the other axes.

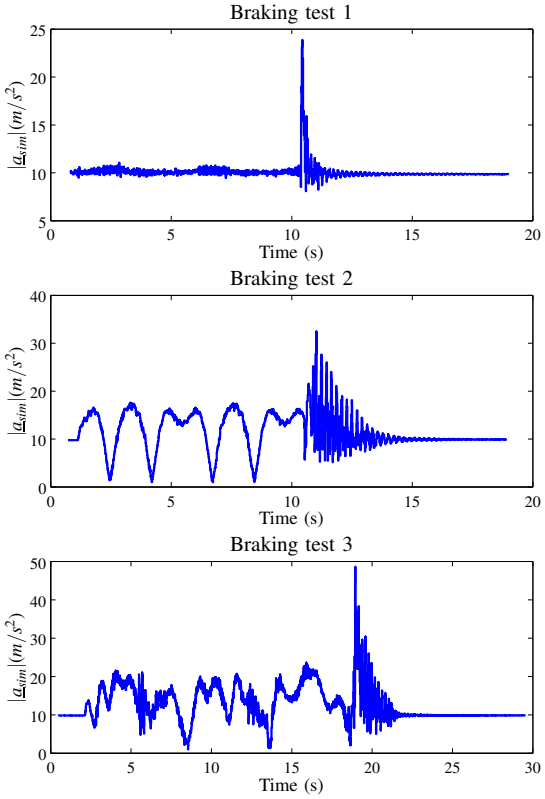

Fig. 11. Acceleration magnitudes at the pilots head position during safety tests. Upper diagram: Emergency stop of Axis 1, middle diagram: Emergency stop of Axes 2,3 and 5, lowest diagram: Emergency stop during movement of all axes.

B. Discussion and conclusions

The accelerations recorded during the tests 1-3 are less or equal 5g, a limit for harmless accelerations during car accidents, as defined by german jurisdiction in cases of whiplash associated disorders [19]. But as the sensor was mounted fixed on the simulator cell, effects induced by the movement of the cervical spine are not considered by these tests.

The Robot controller monitors the maximum acceleration and speed of each joint and when they exceed a certain value, the controller deploys the emergency brakes (as in test 3). However, it is still possible to exceed software limits, especially if the robot is being operated with high velocities and accelerations near these limits. The distance between the hardware limits and the software limits are 1 deg for the axes 2,3 and 5 and 1mm for the linear axis. Axis 1 has a reserve of about 47 [deg], and the axis 4 and 6 are mechanically unconstrained. Considering the results of test 4, the software limits have been reprogrammed to a distance

of 2 [deg] to the hardware limits for axis 2,3 and 0.8 m distance for the linear axis to prevent hardware damage. The software watchdog, introduced in III-D, was implemented as a consequence of these results.

VII. Conclusions and future plans

In this part I, the design and setup of a novel, serial kinematics based motion simulator was presented. Based on a KUKA Robocoaster and prior experience with both offline and online simulations performed on it, we ran extensive simulations to design a new platform for online simulations, resulting in the implementation of the DLR Robot Motion Simulator. To adapt for different vehicles to be simulated, the whole design has been modularized and can now easily be customized to fit specific needs. In particular this required a new simulator cell to be designed from scratch, and the addition of a linear axis. The two introduced cell-designs show the evolution of the simulator concept, from an entertainment system to a fully operational simulator. While further scenarios are currently under development, simulations of fixed wing aircrafts and cars, based on detailed dynamics models, are readily available. During the whole design process, safety was a major concern. Multiple layers of protection systems have been included and detailed tests were successfully performed as shown in part VI.

Part II of this paper covers the path planning algorithm for the DLR Robotic Motion Simulator, required to mimic the accelerations on the pilot in a real vehicle and shows its application as an artistic flight-simulator capable of overhead-flights.

Near term future design improvements will focus on additional simulation scenarios and cockpits, such as the implementation of a realistic replica of a commercial aircraft cockpit, whereas on the long run we want to include a freely rotating axis q_1 to generate sustained accelerations. Furthermore teleoperation of unmanned vehicles is planned.

VIII. ACKNOWLEDGMENTS

The authors would like to thank KUKA Roboter GmbH for funding and supporting this work.

References

[1] D. Stewart, "A platform with six degrees of freedom," in *Proceedings of the Institution of Mechanical Engineers*, vol. 180, pp. 371–386, Professional Engineering Publishing, 1965.

[2] R. Suikat, "The new dynamic driving simulator at dlr," in *Driving Simulator Conference*, (Orlando, Florida, USA), pp. 374–381, 2005.

[3] J. Heindl, M. Otter, H. Hirschmueller, M. Frommberger, N. Sporer, F. Siegert, and H. Heinrich., "The robocoaster as simulation platform - experiences from the first authentic mars flight simulation," in *Proceedings of the 1st Motion Simulator Conference*, (Braunschweig, Germany), 2005.

[4] T. Bellmann, M. Otter, J. Heindl, and G. Hirzinger, "Real-time path planning for an interactive and industrial robot-based motion simulator," in *Proceedings of the 2nd Motion Simulator Conference*, (Braunschweig, Germany), 2007.

[5] L. Pollini, M. Innocenti, and A. Petrone., "Novel motion platform for flight simulators using an anthropomorphic robot," *Journal of Aerospace Computing, Information, and Communication*, vol. 5, 2008.

[6] T. Bellmann, "An innovative driving simulator: Robocoaster," in *Proceedings of FISITA, World Automotive Congress*, (Munich, Germany), FISITA 2008, 2008.

[7] P. Giordano, C. Masone, J. Tesch, M. Breidt, L. Pollini, and H. Buelthoff, "A novel framework for closed-loop robotic motion simulation-part i: Inverse kinematics design," 2010.

[8] KUKA Roboter GmbH, *KR 1000 titan, KR 1000 L750, Spezifications*, 2010.

[9] "Industrieroboter - Sicherheitanforderungen DIN EN ISO 10218," September 2009.

[10] "Fliegende Bauten; Richtlinie für Bemessung und Ausführung DIN 4112," Feburary 1983.

[11] KUKA Roboter GmbH, *Spez Robocoaster de/en/fr*, Feburary 2003.

[12] S. Schaetzle, C. Preusche, and G. Hirzinger, "Workspace optimization of the robocoaster used as a motion simulator," in *Proceedings of the 14th IASTED International Conference on Robotics and Applications*, (Cambridge, Massachusetts, USA), pp. 470–477, IASTED, November 2009.

[13] eyevis GmbH, *OpenWarp2 Datenblatt*, 2010.

[14] INFITEC GmbH, *3D-Visualization*, 2010.

[15] Modelica Association, *Modelica - A Unified Object-Oriented Language for Physical Systems Modeling - Language Specification*, May 2009.

[16] T. Bellmann, "Interactive simulations and advanced visualization with modelica," in *Proceedings of the 7th International Modelica Conference* (F. Casella, ed.), (Como, Italy), Linköping University Electronic Press, September 2009.

[17] G. Looye, S. Hecker, T. Kier, C. Reschke, and J. Bals, "Multi-disciplinary aircraft model development using object-oriented modelling techniques," in *DGLR Jahrestagung*, (Friedrichshafen, Germany), 2005.

[18] G. Looye, "The new dlr flight dynamics library," in *Proceedings of the 6th International Modelica Conference*, (Bielefeld, Germany), 2008.

[19] Regional court Bochum, "Decisision ref. 6 o 225/95," 22.5.1996.