



System for interactive scientific driving simulation with haptic information

Miha Ambrož^{a,*}, Ivan Prebil^a, Roman Kamnik^b, Marko Munih^b

^a University of Ljubljana, Faculty of Mechanical Engineering, Aškerčeva cesta 6, SI-1000 Ljubljana, Slovenia

^b University of Ljubljana, Faculty of Electrical Engineering, Tržaška cesta 25, SI-1000 Ljubljana, Slovenia

ARTICLE INFO

Article history:

Received 17 July 2009

Received in revised form 18 June 2011

Accepted 29 September 2011

Available online 10 November 2011

Keywords:

Vehicle simulator

Virtual 3D environment

Software application

Haptic interface

Active seat

Active steering wheel

ABSTRACT

An interactive wheeled vehicle simulator is described, consisting of a software application for simulating vehicle dynamics and presenting the results in virtual 3D environment, and a haptic interface. The latter consists of a hydraulically actuated active seat and an electrically activated active steering wheel. The system makes it possible to simulate wheeled vehicles of various configurations on arbitrary terrains. The design of the system is presented and the specific considerations are discussed. The system was experimentally verified by measurements on a real off-road vehicle. The kinematic values measured on the real vehicle and those, measured on the haptic interface were compared. The results are discussed and are generally found to correspond well, making the system usable for studying vehicle performance and training its operators.

© 2011 Elsevier Ltd. All rights reserved.

1. Introduction

Interactive simulation of driving dynamics is a useful tool in many fields of human–vehicle–terrain systems analysis. The software applications for such simulations can present different levels of interactivity [1–4]. These range from merely navigating the virtual 3D environment to full control of the simulated vehicle and all its systems. Software tools for scientific interactive driving simulations are a complement to other vehicle analysis related simulation tools. While the renowned specialised tools for driving dynamics and vehicle systems simulations [6] are used when the need for accurate results prevails over the requirements of fast and easy simulation, the tools for interactive driving simulation find their use whenever the simulated vehicle needs to be operated in real time by people without profound technical knowledge about the aspects of vehicle design.

Contemporary implementations of driving simulators differ considerably in their complexity. The simplest among them consist of a computer screen and an input device and mostly do not provide any other feedback to the user apart from visual and aural information. It is, however, desirable that a driving simulator provide as much feedback to the user's senses as possible. This is why the development of driving simulators is headed towards the systems that attempt to provide the user not only with visual and aural information, but also with information about acceleration of the simulated vehicle and forces on its control systems [2–5]. Simulators developed along

these lines range from low-cost solutions for home entertainment use [7–10], to highly versatile systems for simulating high-speed high-acceleration motion of various types of land and air vehicles [11–13].

The aim of the work presented in this paper was to develop a software application with a level of interactivity that will, together with driven hardware, provide real-time interactive simulation of a wheeled vehicle on an arbitrary terrain with haptic information presented to the user in form of active seat motion and active steering wheel torque. The application was named i3Drive (for Interactive 3D Driving simulation). In its current state of development i3Drive is capable of simulating vehicles with up to eight wheels on an arbitrary polygon-mesh terrain model.

2. Simulator design

2.1. Vehicle dynamics mechanical model

Multibody system simulation was chosen as the means of simulating the vehicle dynamics. The mechanical model of the simulated vehicle was composed in a way that permits real-time simulation on readily available personal computers [1] whilst still retaining the complexity, required for sufficiently accurate simulation. To achieve this delicate balance, the mechanical model of the vehicle consists of bodies that represent the vehicle chassis and individual wheels together with suspension elements. An example of a six-wheel vehicle is shown in Fig. 1. The bodies in the system are connected by joints with individually adjustable characteristics. Such a model corresponds with the “lumped-mass-model”

* Corresponding author. Tel.: +386 1 4771 186; fax: +386 1 4771 178.

E-mail address: miha.ambroz@fs.uni-lj.si (M. Ambrož).

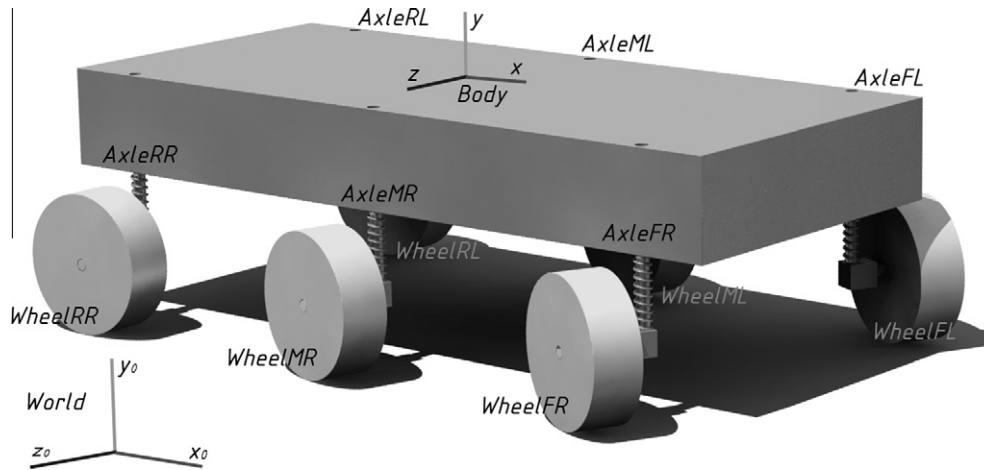


Fig. 1. Example mechanical model of a six-wheeled vehicle.

as described in [14]. The simulated vehicle can have 2, 3 or 4 axles (4, 6 or 8 wheels). All the multibody model characteristics, which include mass and inertial properties of the bodies and spring and damping characteristics of the joints, are parametric and can be set up upon model creation. Some of them can even be adjusted during the simulation.

Numerical simulation of the driving dynamics model is performed by the Open Dynamics Engine physics simulation library [16].

The parameters for the particular test vehicle (a light armoured vehicle Valuk 6 × 6) were obtained by direct measuring of its geometrical, mass and inertial properties in cooperation with the vehicle manufacturer. Due to limitations of the mathematical model and to achieve real-time simulation with a reasonably short simulation interval some of the suspension elements (notably the damping characteristics) have been simplified by linearising their characteristics. The consequence of this is that the vehicle motion at high vertical velocity gradients may induce inaccuracies into simulation. Nonetheless, the simulation results prove accurate enough when compared to measurements on a real vehicle for all the chosen test scenarios.

2.2. Vehicle drivetrain model

Vehicle drivetrain consists of the engine as the variable source of torque, the power transmission elements and the arbitrary number of driven wheels that transfer the engine torque to the driving

surface. The number and the design of the power transmission elements can vary significantly between different types of vehicles [15]. The goal was to be able to simulate various types of vehicles, which is why the model of power transmission includes all the elements found on a generic vehicle. These include clutch, main gearbox, reduction gearbox, axle gears and hub gears (Fig. 2). Each of these elements converts torque and angular velocity according to its transfer function. The transfer functions can have user input as parameter and can be continuous, such as that of clutch, discrete, such as that of a multi-speed gearbox or constant, such as that of an axle- or hub gear. As not all of the transmission elements are present in all types of vehicles, any one of them can be omitted from the model by assigning a constant value of 1 to its transfer function.

2.3. Vehicle braking system model

The braking system of the vehicle is modelled as a source of tangential force on the braked wheels. The magnitude of the force depends on the position of the brake pedal as user input. Braking force can be individually applied to each of the wheels according to vehicle parameters. The simulator provides a basic model of braking force reduction on the rear axle depending on the normal force on the rear axle. A model of slip-detection-based anti-lock brake system (Fig. 3) is also implemented. The anti-lock braking system model can be switched on or off by an input parameter. When switched on, the model monitors the angular velocity of the wheels and calculates longitudinal wheel slip s . Whenever its

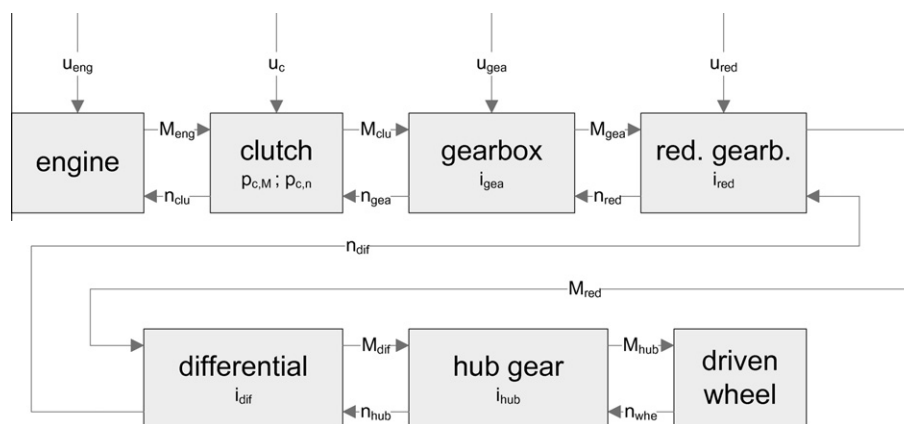


Fig. 2. Vehicle drivetrain model schematics.

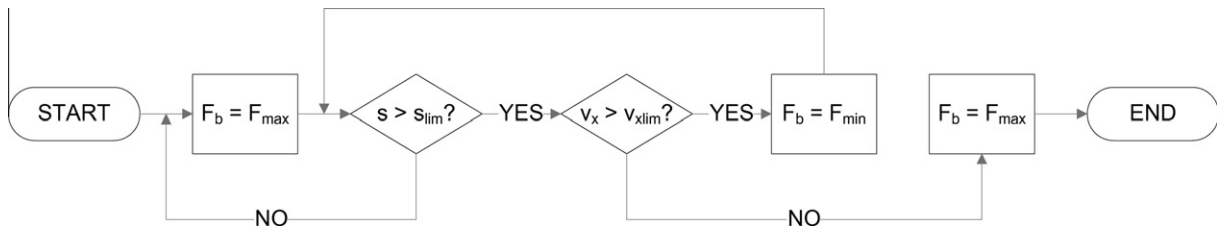


Fig. 3. Anti-lock brake system schematics.

value rises above the specified threshold value s_{lim} , the braking force F_b is reduced from maximal braking force F_{max} to minimal braking force F_{min} to allow the wheel to start rotating again. This way the simulated vehicle brakes with wheel slip in the interval, where the friction is the highest, and retains its steering abilities even during braking with maximal force on the brake pedal.

2.4. Vehicle steering model

The steering system of the vehicle is modelled with a transfer function that converts the steering wheel rotation angle as user input to rotation angles of each individual steered wheel. The model provides means of modelling steering mechanism kinematics to achieve steering angle difference between wheels on either side of the same axle. Each individual axle on the simulated vehicle can be steered or unsteered. The unsteered axles have their steering transfer function constant and set to zero. Each steered axle can have its own different steering model parameters. The relation connecting the steering wheel angle φ as user input to the wheel angles on the left (δ_l) and on the right (δ_r) steered wheel is modelled using the following transfer functions:

$$\begin{aligned} \delta_l &= \varphi^2 \cdot f_\delta \cdot f_s + \varphi \cdot f_s \cdot C \\ \delta_r &= \varphi^2 \cdot f_\delta \cdot f_s - \varphi \cdot f_s \cdot C \end{aligned} \quad (1)$$

where f_s is the steering system gain factor, f_δ is the steering angle difference factor and C is the geometry-dependent constant.

This way it is possible to model steering mechanisms that fulfil the Ackermann condition [14,15], which has to be met in order to

provide rolling wheel motion without sideways sliding when cornering at low speed. Such a setup makes it possible to model vehicles with various different steering systems, including rear-axle-steered forklifts, four-wheel-steered off-road vehicles, wheeled vehicles with multiple steered axles (Fig. 4) as well as special-purpose vehicles with unconventional steering systems [17].

2.5. Interaction between vehicle tyres and driving surface

The driving surface is modelled as a triangle mesh and used in collision algorithm to excitate the simulated vehicle wheels with forces. The ODE collision detection system [16] is used to determine the contacts between vehicle parts and the terrain geometry and to calculate the contact forces. The shape of the terrain triangle mesh is arbitrary and without restrictions for triangle dimensions. This way the memory required for storing triangle data and the number of vertices used in collision detection can be effectively reduced in the largely flat areas of the terrain, whilst the more agitated areas can be represented by a larger number of triangles providing more accurate simulation. The point data to generate the terrain mesh can be obtained from various sources. These include field measurements and GIS data for existing real terrains, or different 3D modelling techniques for artificial terrains. The terrain data is stored in files that contain point and surface data and are read-in prior to simulation start.

Apart from the driving surface model the terrain can include an arbitrary number of additional objects that can either be freely moving (i.e. interacting with other objects only through collisions)

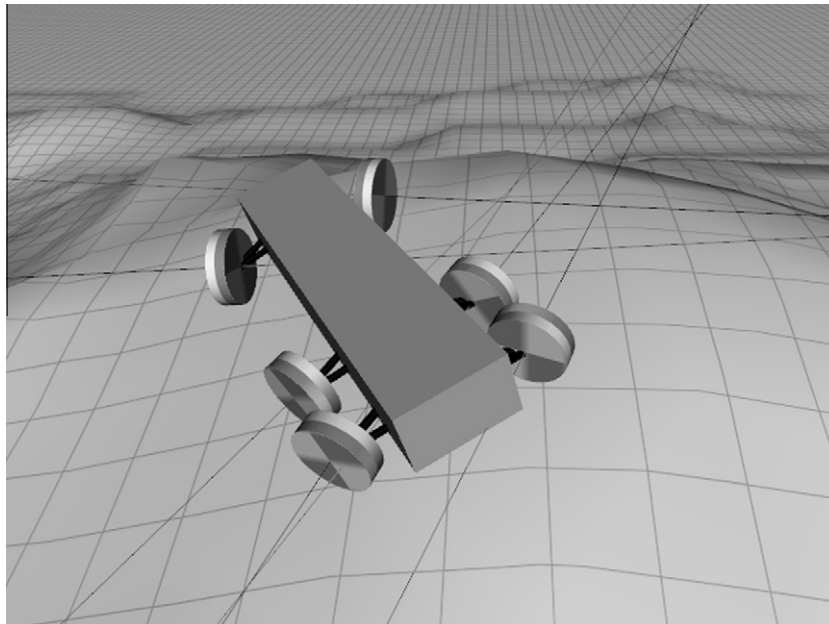


Fig. 4. Model of a vehicle with multiple steered axles.

or connected to terrain and other objects by means of various types of mechanical joints.

The forces that occur on the tyre – driving surface contact depend on many factors [19], of which the most significant are the longitudinal wheel slip s and lateral wheel slip angle α as defined by

$$s = \frac{|\vec{\omega}| \cdot r_d - |\vec{v}_x|}{|\vec{v}_x|} \quad (2)$$

$$\alpha = \arccos \frac{\vec{v}_x \cdot \vec{u}_x}{|\vec{v}_x| \cdot |\vec{u}_x|} \quad (3)$$

where $\vec{\omega}$ represents angular velocity of a particular wheel, r_d its dynamic radius, \vec{v}_x the longitudinal velocity of its centre and \vec{u}_x the radial unit vector from the centre of the wheel pointing forward.

These two quantities are computed for each wheel in every simulation step. The forces are modelled separately for the longitudinal and the lateral direction of the wheel using the simplified formulae (4) and (5) as proposed by [18]

$$\mu(s) = \mu_0 \cdot A \cdot \sin(B \cdot \arctan(C \cdot (s + D))) \quad (4)$$

$$\mu(\alpha) = \mu_0 \cdot E \cdot \sin(F \cdot \arctan(G \cdot (\alpha))) \quad (5)$$

where A , B , C , D , E and F are constants, the values of which are obtained from experiments and depend on the type and geometry of the particular tyre.

The code also includes statements for computing the aligning torque M_z on each steered wheel using a simplified formula (6) as per [18].

$$M_z(\alpha) = D_r \cdot \sin(C_r \cdot \arctan(B_r \cdot \alpha)) \quad (6)$$

where D_r , C_r and B_r are again constants dependent on the tyre properties and obtained experimentally.

The code statements for computing the aligning torque can be included into the simulation loop and executed once every simulation step if steering wheel self-centring or torque feedback (e.g. for controlling an active steering) is required.

3. Software application for interactive driving dynamics simulation

3.1. The application structure

The simulation models as described in the previous chapter, together with components for display in virtual 3D environment, modules for polling control devices, a graphical user interface to control the application behaviour and components for simulation results export, are incorporated into a software application named “i3Drive”. The main purpose of the application is providing an environment for scientific interactive driving simulation. By using it, the user can analyse vehicle movement in different conditions. With user access to all the relevant simulation parameters it is possible to realistically simulate real vehicles on real terrains, which also makes the application useful for vehicle operator training. The structure of the application is schematically shown in Fig. 5.

The simulation results are presented as an interactive animation in virtual 3D environment. The application provides the user with tools to observe the animation through the eyes of a virtual person (avatar), which can be moved through the virtual environment, rotated in respect to it, or attached to various positions on objects (e.g. vehicle or driving surface). Together with possibility of adjusting the viewpoint parameters, this provides the user with a realistic environment experience.

The application is controlled through graphical user interface, which includes elements for setting the parameters of included simulation models, the virtual 3D display, the input and output devices and data transfer files. The main part of the graphical user interface is the main application window as shown in Fig. 6. It provides the user with display in virtual 3D environment, main information about application status and the user interface elements for application control such as menu bar and toolbar (Fig. 7).

Data entry and parameter adjustments are performed through separate dialogue windows (Fig. 8) that are specific to each task and among others include the vehicle settings dialogue, the terrain settings dialogue and the simulation settings dialogue.

3.2. Required simulation input data

To simulate the vehicle dynamic and control the application needs a set of input data. These include the vehicle property data, the vehicle geometry data, the terrain property and geometry data, data about objects in the environment and the vehicle control data.

The vehicle property data is organised in text files that are read-in before the simulation is started. The files contain vehicle properties in human-readable format one per line and can either be created from scratch in a text editor or exported from the i3Drive application itself. The vehicle property data-reading routines take care of substituting the defaults for any data that might be missing from the file.

The vehicle geometry is stored in VRML 2.0-compliant files that follow some additional rules in order to represent the model appearance. These include lines with metamarkers in commented lines that designate special sections of the VRML 2.0 file (the vehicle body colour, the light-emitting bodies on the vehicle, etc.).

The terrain data in general contains the geometry and friction properties of the terrain. The i3Drive application can import the terrain geometry either from VRML 2.0 files (where geometry is represented by a specially designated IndexedFaceSet node) or from proprietary ADAMS Road Files (where geometry is represented by arrays of points and surfaces defined by them). The terrain friction and softness properties are controlled globally by a set of user parameters, adjustable in the “Terrain settings” window.

The data about the objects included into the simulation environment are stored in separate text files that contain object descriptions and properties in human-readable format. Currently the objects can be one of the geometrical primitives (sphere, cylinder, or box) or arbitrary triangle meshes.

The vehicle control data consists of states of the systems that are controlled by the vehicle operator and can, in general, change in every simulation interval. The control data includes the following quantities: engine throttle pedal position, main brake pedal position, steering wheel angle, selected gear ratio, state of engine clutch, state of the auxiliary brake handle. The i3Drive application offers two modes of simulation: in the interactive mode the vehicle control data is delivered into the model of driving dynamics from the control device by the user in real-time, whilst in the non-interactive mode the data is read from a file one line per simulation step.

3.3. Export of simulation results

During the simulation, the i3Drive application can export the simulation results in various formats. The values of simulated vehicle parameters (including kinematic values of all the vehicle bodies, engine properties and body contact properties) can be written to a tab-separated text file for post-simulation analyses. The animation itself can also be written either as an interactive VRML 2.0 animation that faithfully retains the kinematics as simulated, or as digital video encoded exactly as displayed on main window display area.

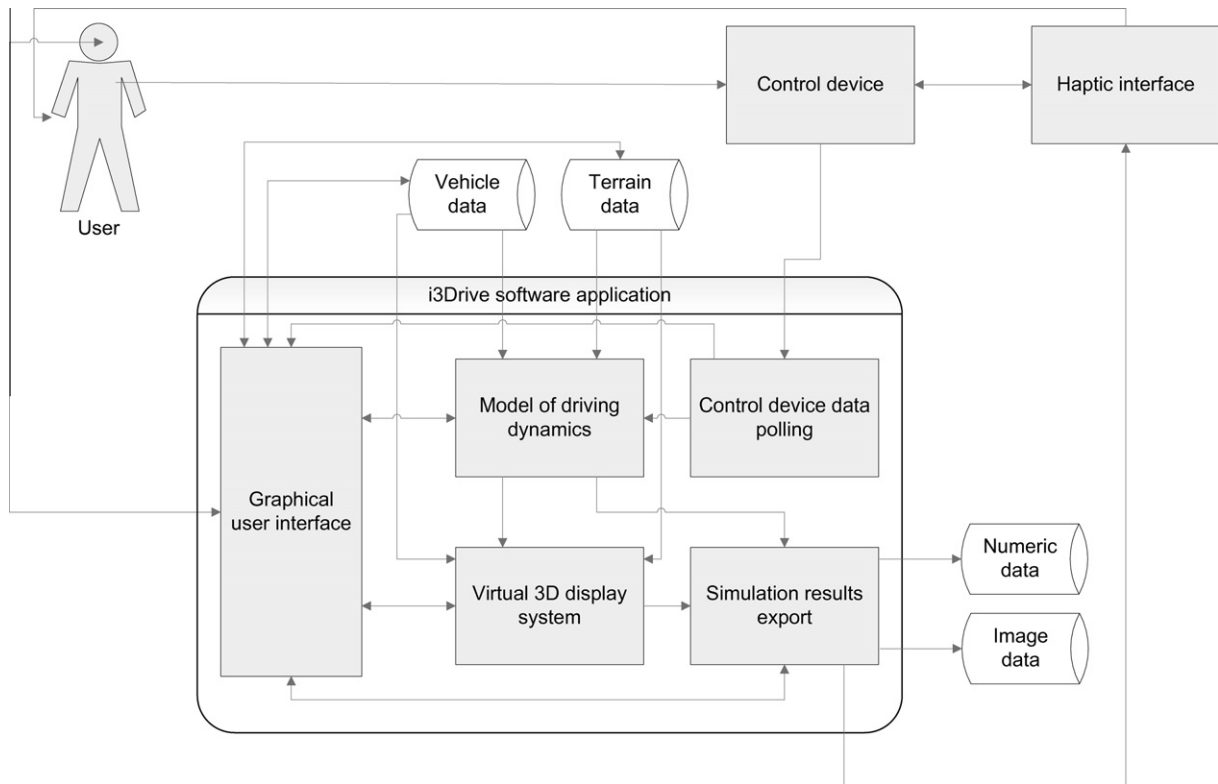


Fig. 5. Schematics of the i3Drive application structure.

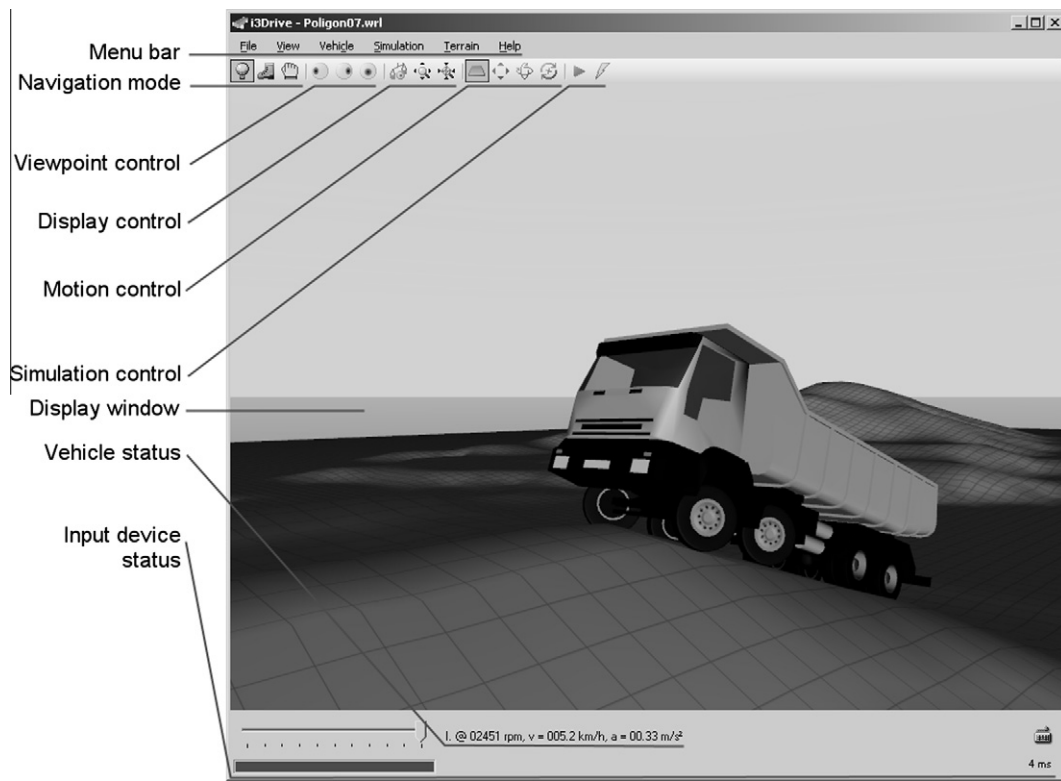


Fig. 6. i3Drive main application window.

The individual simulation results export options can be switched on simultaneously (if permitted by available computing

power of the host system) and are controlled by the user via the main window “File” menu.



Fig. 7. i3Drive main menu and toolbar.



Fig. 8. Dialogue windows for application control.

The application also provides means of exporting a single simulation frame as a static VRML 2.0 file or a still image of the display window in PNG format.

3.4. The help system

The i3Drive application is supported by a help system consisting of the “Online manual” in Microsoft CHM format and the “context help”, which is incorporated into the user interface itself.

The “Online manual” provides the user with basic data about the structure and operation of the application and gives a detailed reference about all the input and output file formats.

The “context help” provides the user with information about the user interface elements, their purpose and their accepted values.

4. Haptic information

The haptic interface consists of a hydraulically actuated active seat and an electrically actuated active steering wheel. The software application is connected to the hardware controller via ethernet and sends the information about seat position and cumulative wheel aligning torque as byte packets via TCP/IP network using the UDP protocol.

The application provides the user with means of adjusting the parameters of the transfer functions of both elements of the haptic interface.

4.1. Active driver seat

Active driver seat is realised in the simulation environment by the use of a robot device developed for training of standing-up

[20]. The robot device is a hydraulically driven 3-DOF mechanism, which in a way of supporting the subject, resembles half of a see-saw. The driver is seated on a seat mounted on the robot end-effector. The robot configuration enables an arbitrary seat motion restricted to a subject’s sagittal plane.

Positioning of the end-effector is accomplished by movement of the two robot segments. The first segment is rotating around its axis in a robot base, while the second translational segment is moving longitudinally along the first one. Both segments are driven by linear hydraulic actuators. At the robot end-effector the orientational mechanism is mounted assuring horizontal seat orientation in any robot position. Constant seat orientation is maintained by a passive hydraulic bilateral mechanism (Fig. 9).

The robot mechanism is driven by an electrohydraulic servosystem presented in Fig. 10. The system is powered by a hydraulic pump providing the pressure of 50 bar and the hydraulic current of 1 l/s. The pump performance and the mechanism design allow the maximal speed of the robot end-effector of up to 2 m/s. Two Moog D641-3 servovalves with incorporated electronics are used to control the pressure difference applied to the linear hydraulic cylinders driving both links. The robot device operates in the position control mode of operation. In this mode, the control objective is to guide the robot end-effector along the desired path (specified via TCP/IP connection) in terms of velocity and/or acceleration at each point.

Control system of the robotic device is implemented in two levels (see the upper part of Fig. 10). At the lower level, the hydraulic servosystem is controlled by a target computer built upon a 1 GHz PC Pentium III platform. On the platform, the xPC Target real time operating system is running at a constant sampling rate of 1 kHz. Two PCI interface boards are used in this controller to interface

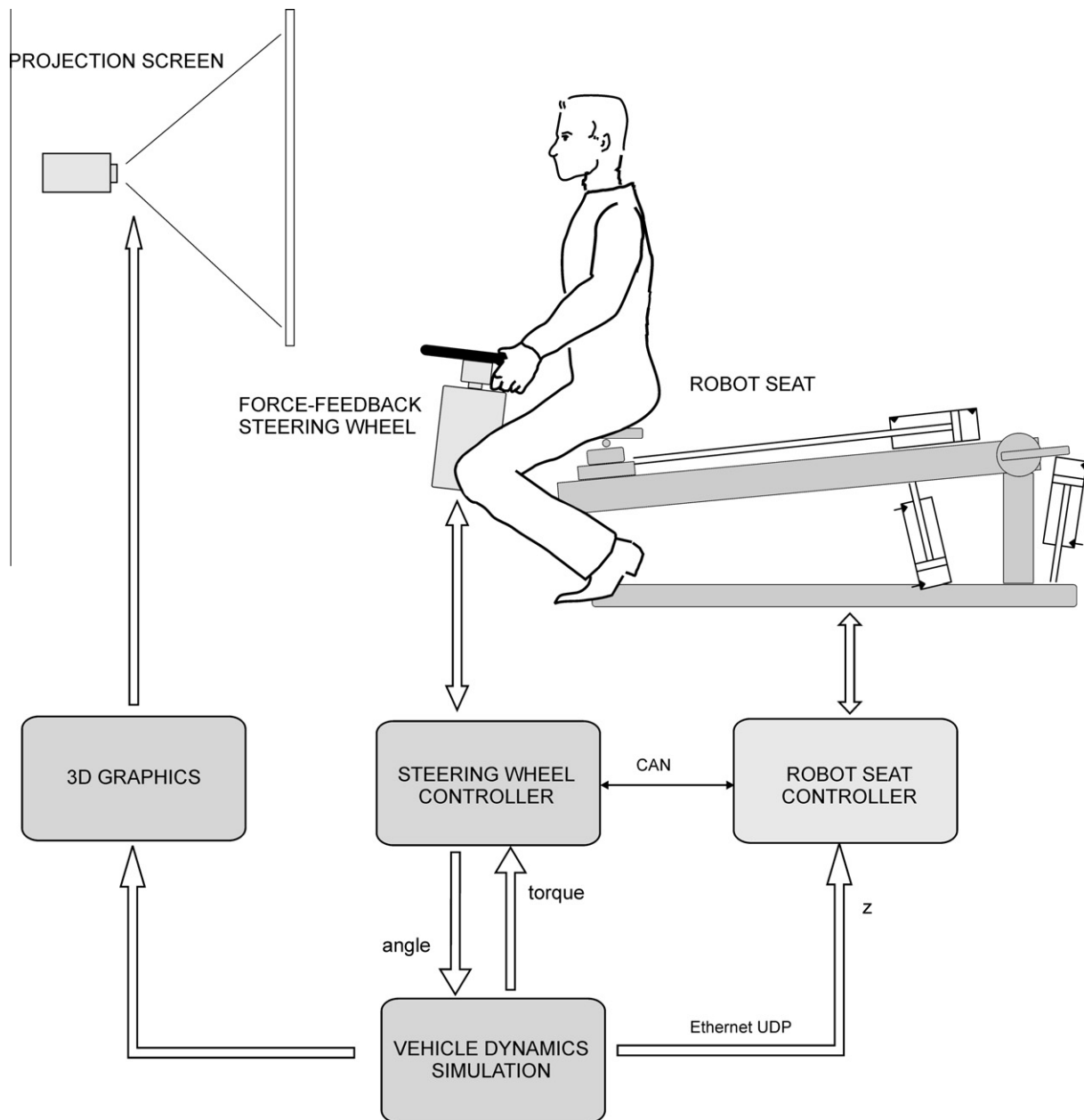


Fig. 9. Active driver seat schematics.

to the external hardware. The PCI-DDA08 board (Measurement Computing, Inc., Middleboro, USA) acquires the analogue force and pressure signals, and reads the joint positions via integrated encoder interfaces through its digital inputs. Another Measurement Computing board type D/A PCI-DAS1002 is employed to drive the hydraulic servovalves applying the output voltage in a range of ± 10 V. At the higher level, an additional computer is used as a host platform for robot system programming, supervision, data logging and control. The environment is based on Mathworks Matlab software with Simulink, Stateflow and xPC Target toolboxes. The configuration allows the development of control algorithms in graphical mode by building and connecting functional blocks, which provides user friendly graphical software development environment, optimal tuning of parameters, and easy acquisition and logging of signals.

4.2. Force feedback steering

For the implementation of force feedback steering an Iskra AML 7103 induction motor is directly coupled to the steering wheel. This direct drive configuration, shown in Fig. 11, is able to produce high torques with zero backlash enabling a realistic driving sensation. Traditionally, separately excited direct current motors (SEPEX) are being used in high power applications where accurate torque control is required. Since the commutator introduces considerable issues (noise, sparks, frequent maintenance, high rotor inertia, lower efficiency), the SEPEX motors are being replaced by contactless motors. For our application, an induction motor was chosen as at higher powers induction motors are cheaper and more robust. However, the torque control of an induction motor is more complex. As an accurate torque control around zero speed region is

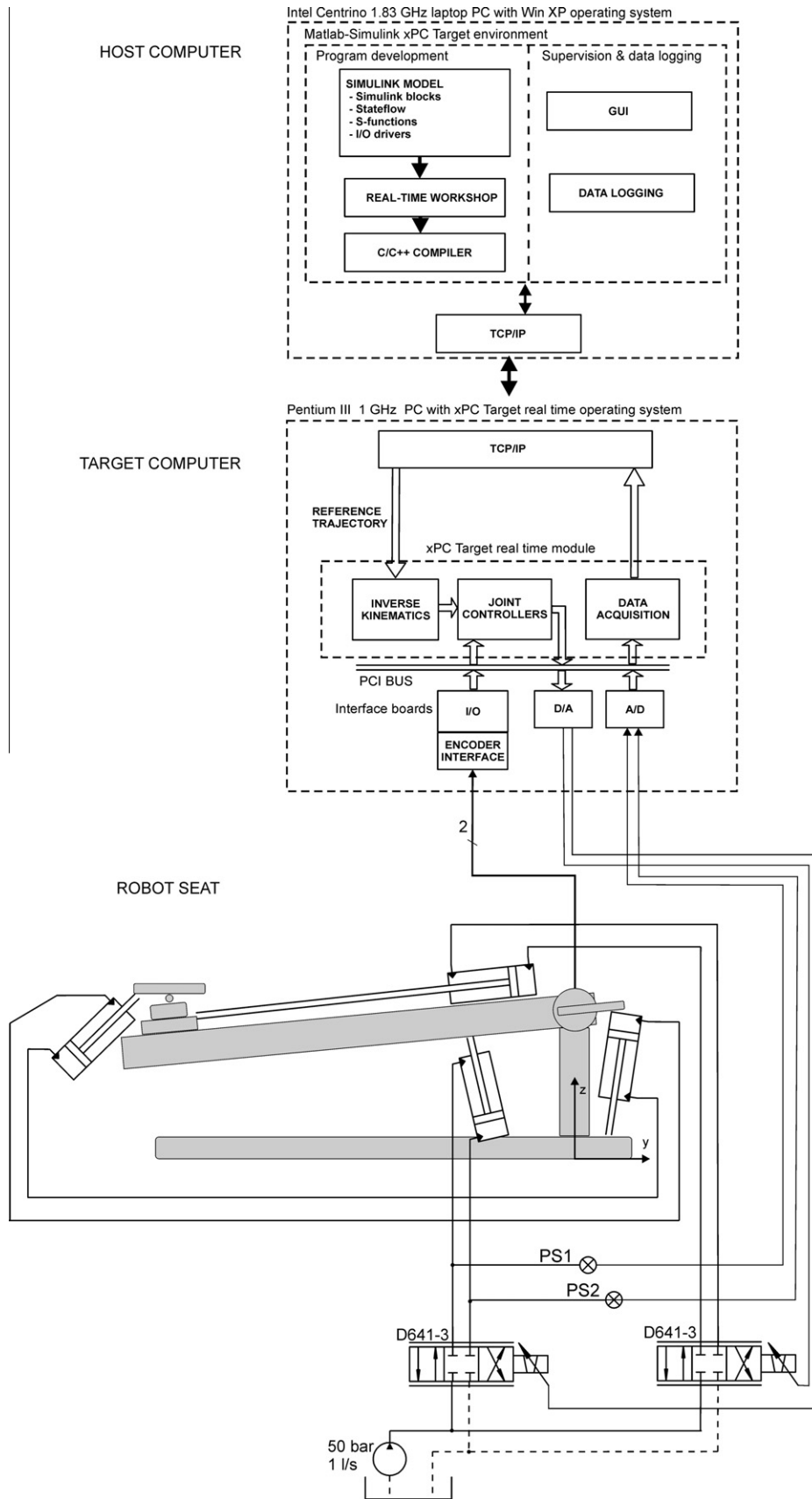


Fig. 10. Electrohydraulic servosystem for driving the active seat.

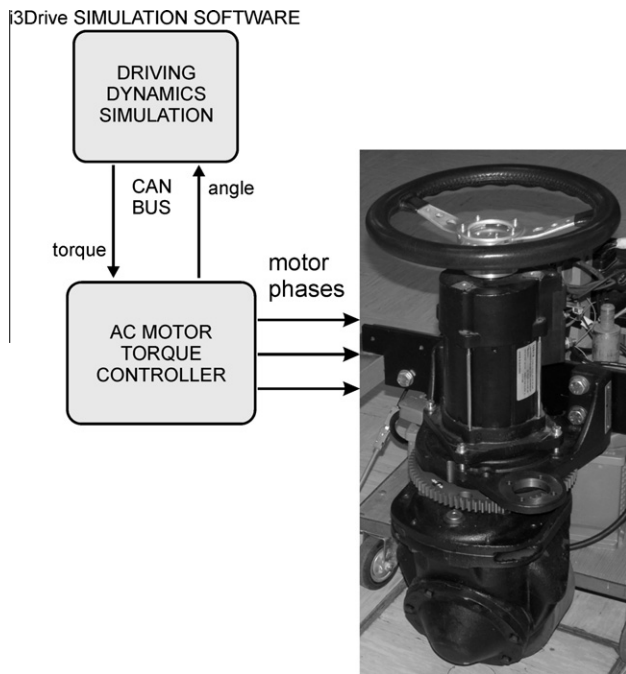


Fig. 11. Active steering wheel.

necessary, an indirect field oriented control scheme (FOC) is implemented [21]. For this purpose, an Iskra AES1134 controller which can produce up to 400 A current is used. The controller performs torque control at 10 kHz by the help of a fixed point Texas Instruments TMS320LF2406 DSP microcontroller. The particular motor-controller pair allows for up to 38 Nm of torque.

Fast CAN interface is implemented for communication between motor controller and superior xPC Target controller. The application uses master–slave principle of communication where the target computer behaves as a master device and the AC motor behaves as a slave device on the bus. CAN messages, that can be up to 8 bytes of data long, are transferred with maximum speed of 1 Mbit/s, allowing to the xPC Target operating system to run at a sampling rate of 1 kHz. Via CAN bus the reference torque value is transferred to the AC motor controller, while the steering wheel angle value is transferred in the opposite direction.

5. Experimental verification of simulation results

To verify the behaviour of the system, certain kinematic values were measured on the real vehicle during test driving, and on the haptic interface active seat and steering wheel. The sets of measured values were compared to each other and to simulated kinematic values in order to evaluate the agreement between the motion of the real vehicle and that of the simulated vehicle with haptic interface.

5.1. Measurements on the real vehicle

The measurements were carried out on a six-by-six armoured vehicle during tests on three differently shaped test tracks. The first set of measurements was done on a half-cylinder test track (Fig. 12) where the vehicle response to short single-peak input was observed. The second set of measurements was done on a “washboard” test track with sine waves in longitudinal direction (Fig. 13) to test the vehicle response to periodic excitation at different speeds. The third set of measurements was done on a level surface according to [22]. Its purpose was to determine the steering wheel torque due to wheel self aligning in a curve.

For the first two tests the vehicle was equipped with five MEMS accelerometers in several places on the vehicle body, its suspension elements, the unsprung part of the driver’s seat and the driver’s helmet (Fig. 14). The data acquisition at 1000 samples per second and data storage was performed by the computer system in the vehicle. In addition to the accelerometer system, a high-speed imaging system with a rate of 500 frames per second was used on the half-cylinder test track to determine the time curve of the driver’s seat position by means of motion image analysis.

The vehicle driver’s seat kinematics (acceleration and displacement) were chosen as the quantities to compare to those simulated on the robotic simulator, since they are directly comparable and straightforward to measure on both systems.

The tests on the half-cylinder test track were performed at target velocities of 10 km/h, 15 km/h and 20 km/h. The vehicle was driven over the half-cylinder obstacle with all its wheels without applying the brakes until the vehicle had entirely passed the obstacle. The time curves of the vertical acceleration on the vehicle driver’s seat at different target velocities are shown in Fig. 15. The driver’s seat displacement time curve at 10 km/h (as obtained from



Fig. 12. Vehicle under test on a half-cylinder test track.



Fig. 13. Vehicle under test on the “washboard” test track.

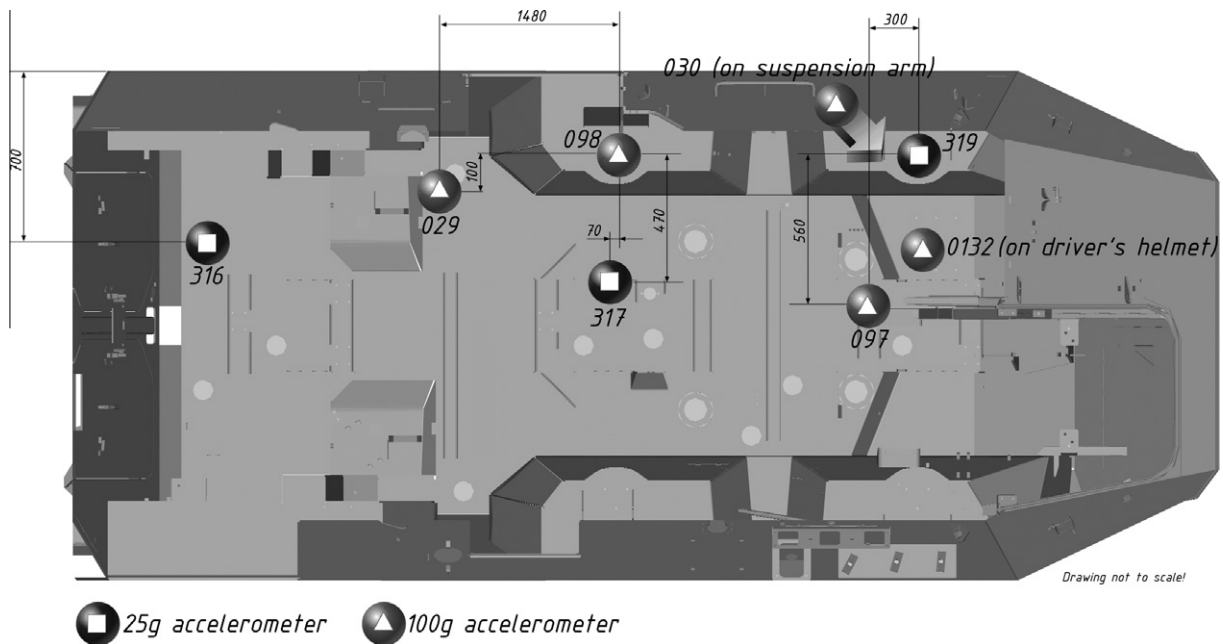


Fig. 14. Position of the accelerometers during testing.

motion image analysis) is shown in Fig. 16. It is worthwhile to note that the fourth peak in the vertical acceleration time curve is due to unequal second wheelbase, which is a design property of the particular vehicle.

The tests on the “washboard” test track were also performed at target velocities of 10 km/h, 15 km/h and 20 km/h. A major resonant frequency of the entire vehicle as a system on the washboard test track was experimentally determined to appear at vehicle velocity of approximately 17 km/h, which means that during the tests at 20 km/h the vehicle was operating above the mechanical resonance of its suspension system. Shown in Fig. 17 are extracts of driver’s seat

vertical acceleration time curves in the part of the course where the vehicle velocity had settled at the desired target velocity.

For the third test the vehicle was equipped with a longitudinal velocity sensor and a strain-gauge-based steering wheel torque measuring device (Fig. 18). The steering wheel torque time curve was measured during driving along a semi-circular path of 30 m radius with constant target longitudinal velocity of 16 km/h as per [22]. The measured path curve in the x - y plane is shown in Fig. 19 and the measured steering wheel torque time curve $M_{Wz}(t)$ in Fig. 20. Slight oscillations in the measured steering wheel torque are due to driver’s corrections of steering wheel angle in order to

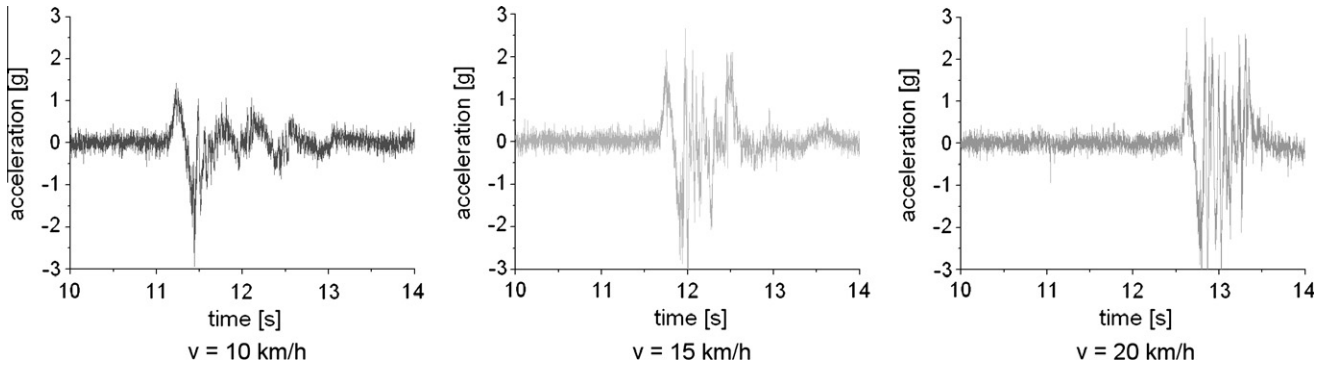


Fig. 15. Vertical accelerations in g ($1\text{ g} = 9.806\text{ m/s}^2$) of driver's seat on the half-cylinder test track at different velocities.

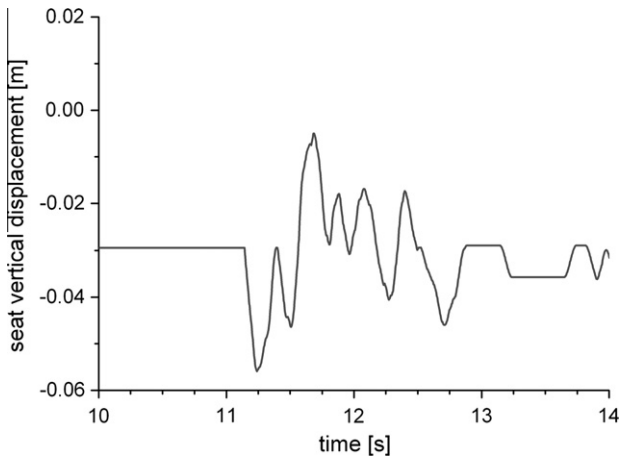


Fig. 16. Driver's seat displacement on the half-cylinder obstacle at 10 km/h.

keep the vehicle on the designated curve, marked on the driving surface.

5.2. Measurements on the robotic simulator

In the experimental evaluation of the robot environment for combat vehicle driving simulation the experimental setup was tested. The evaluation protocol incorporated the manoeuvring of combat vehicle in the same testing scenarios that were assessed in real vehicle. The driver manoeuvred the vehicle by the steering wheel system, while the vehicle motion dynamics and its interaction with the environment were calculated on-line by the i3Drive software. Haptic information about the vehicle vertical motion

and wheels aligning torque were delivered to the driver by the help of robot seat and force feedback steering system. A back-projection screen and a LCD projector was used for providing visual feedback about the vehicle position in virtual 3D space.

5.3. Comparison of results

The driver seat vertical displacements (Fig. 21) and seat vertical acceleration (Fig. 22) are presented comparing the situation in simulated case and in real vehicle during controlled manoeuvring. The testing scenario was the driving over a half cylinder obstacle with target velocity of 10 km/h. Figures present the seat vertical motion in real vehicle assessed by the optical measurement system and measured acceleration compared to motion calculated by simulation software and motion accomplished by the robot seat. As it can be seen from the graphs, the simulation model of the vehicle well approximates the real vehicle behaviour, while the robot controller provides good tracking of the calculated reference signal. The haptic seat acceleration peaks correspond with terrain features without noticeable delay and the acceleration magnitudes are comparable between real vehicle and simulation, which is both essential to achieve the subjective feeling of motion that is comparable between driving a real vehicle and simulating driving with the simulation system. The haptic seat displacements are slightly larger than those in real vehicle, especially when their magnitude is small. This is due to the haptic seat design and was not noticeable by test subjects during tests.

Fig. 23 presents the steering wheel torque during driving on a semi-circular path accomplished with a constant steering wheel angle and a constant vehicle velocity as per [22]. The graph compares the torque measured on the real vehicle, the torque calculated in a simulated run, and the torque generated at the active steering wheel system in driving simulator. The steering wheel

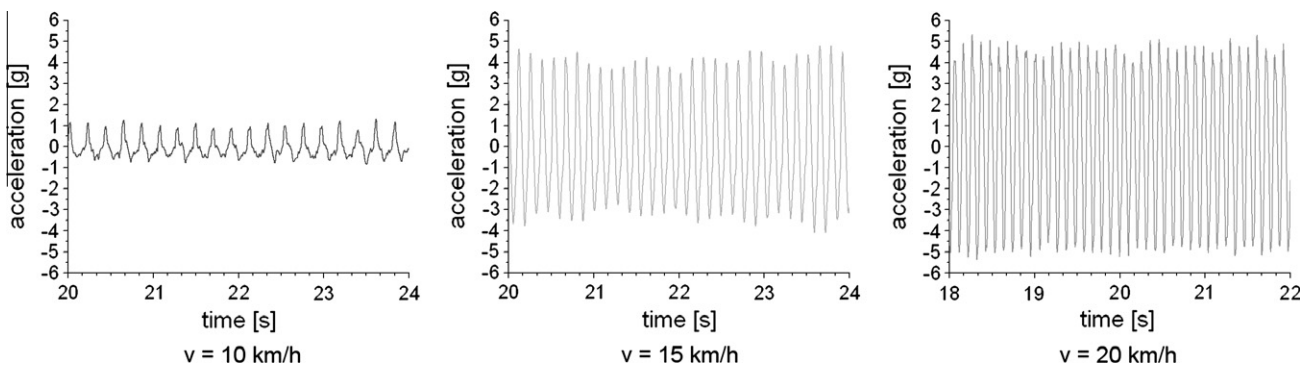


Fig. 17. Vertical accelerations in g ($1\text{ g} = 9.806\text{ m/s}^2$) of driver's seat on the "washboard" test track at different velocities.



Fig. 18. Device for measuring the steering wheel torque.

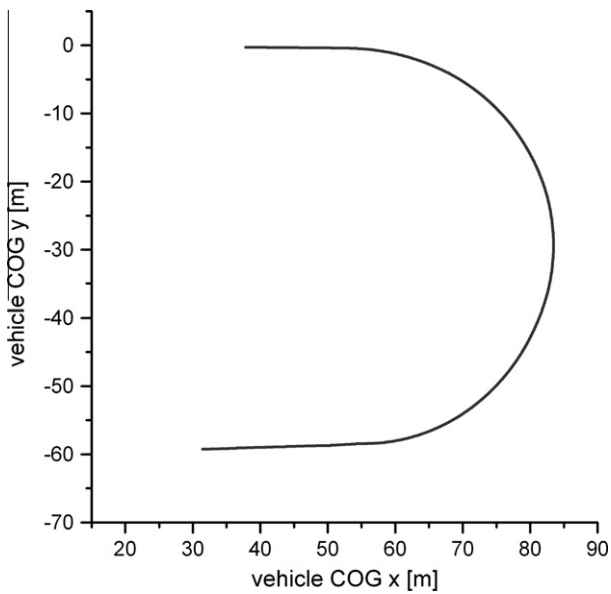


Fig. 19. The vehicle path in x–y plane during steering effort test.

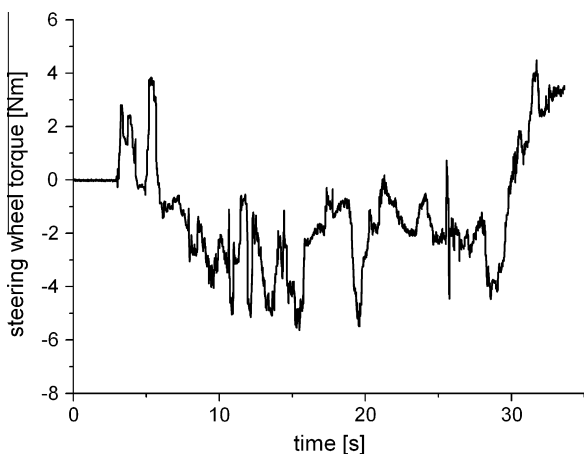


Fig. 20. Steering wheel torque during steering-effort test.

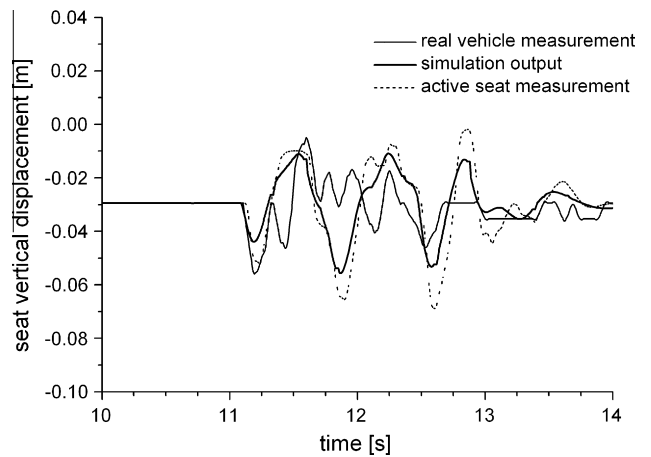


Fig. 21. Comparison of driver's seat vertical displacements during driving over the half-cylinder obstacle at 10 km/h.

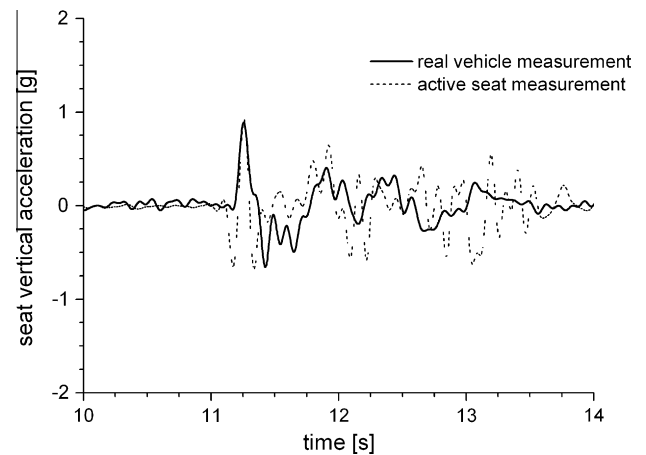


Fig. 22. Comparison of driver's seat vertical accelerations in g ($1\text{ g} = 9.806\text{ m/s}^2$) during the half-cylinder test.

torque in simulator was calculated on a basis of the model of wheels interaction with the terrain during cornering. The calculated wheel aligning torque was delivered to the steering wheel torque controller as the torque reference. Considering the fact that

the steering wheel angle of the real vehicle was being constantly adjusted by the driver during the test in order to follow the designated path curve, the results show good agreement between the steering wheel torque measured on the real vehicle and the simulator's active steering wheel. The haptic steering wheel torque was generally felt realistic by the test subjects. To further enhance realism the vibrations observed in the steering system

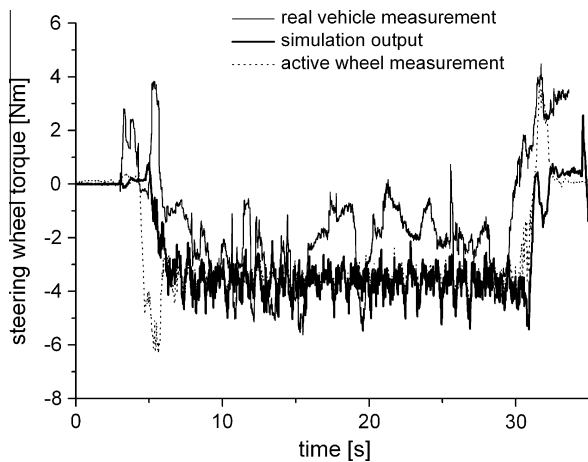


Fig. 23. Comparison of steering wheel torques during the steering-effort test.

of the real vehicle at certain vehicle speeds could be superimposed to the calculated wheel aligning torque.

6. Conclusions

Driving simulation in military applications provides a safe virtual environment in which driving behaviour experiments can be conducted that are difficult or even impossible to undertake in real environments because of potential risks to subjects or vehicles. Driving simulation is also important for virtual prototyping of new vehicle designs that, for example, arises from changes in vehicle configuration and driver training in such situations.

The paper describes the design and evaluation of a simulation environment intended for driving simulation of combat vehicle. The simulation environment is based on the detailed simulation model of combat vehicle dynamics, the virtual 3D graphics display, the robot seat and the force feedback steering system. The developed driving simulator incorporates a novel hydraulic robot device as the active driver's seat and utilises an induction motor as a direct driven force-feedback steering wheel. The evaluation results show good matching between responses measured in the real vehicle and those measured in the simulated environment.

Currently the simulator is designed to run on two personal computers, of which one performs the mechanical simulation and drives the virtual 3D display and the other controls the haptic interface controller. When running the simulation of a 6×6 vehicle on a terrain consisting of around 33,500 triangles, it is possible to achieve a 4 ms simulation interval and a display update of 30 fps on an Intel Core2 Quad CPU at 2.83 GHz.

The developed simulator serves as a prototype testbed for the study of simulation capabilities and the possibilities of software and hardware modules integration. Based on the presented results, it can be further developed into a high-fidelity combat vehicle simulator.

Further development will be focused on improving the mathematical models of vehicle propulsion, braking and steering systems, thus optimising them for speed and accuracy. Together with use of faster computer systems to run the simulation on, this will enable shortening the current simulation interval to provide

even smoother simulation. The development will also include a distributed version of the simulator application that will allow simulation of multiple vehicles in a multi-user environment, representing a step towards a tactical simulator.

Acknowledgements

The authors would like to acknowledge the Republic of Slovenia Ministry of Defence and Ministry of Education, Science and Sport Grants "System for Analysis of Exploitation Capabilities of Army Vehicles" (M2-0126), "Motion Analysis and Synthesis in Human and Machine" (P2-0228 C), "Modelling in Technics and Medicine" (P2-0109 C) and "Modelling of anatomic structures for analysis of occupant loads and injuries in traffic accidents" (J2-9536).

References

- [1] Ambrož Miha, Krašna Simon, Prebil Ivan. 3D road traffic situation simulation system. *Adv Eng Softw* 2005;36:77–86.
- [2] Wang Zheng, Liu Zhenyu, Tan Jianrong, Fu Yun, Wan Changjiang. A virtual environment simulator for mechanical system dynamics with online interactive control. *Adv Eng Softw* 2006;37:631–42 [Elsevier Ltd., 2006].
- [3] Shiiba Taichi, Suda Yoshihiro. Evaluation of driver's behavior with multibody-based driving simulator. *Multibody Syst Dyn* 2007;17:195–208 [Springer Science + Business Media B.V. 2007].
- [4] Lee Wong-Sung, Kim Jung-Ha, Cho Jun-Hee. A driving simulator as a virtual reality tool. In: *Proceedings of the 1998 IEEE international conference on robotics and automation*, Leuven, Belgium; May 1998.
- [5] Riggins Lt Col David W. The direction of virtual vehicle simulations for military training, virtual media for military applications. In: *Meeting proceedings RTO-MP-HFM-136*, paper 13, Neuilly-sur-Seine, France; 2006. p. 13–1–6.
- [6] MSC Software Adams. <<http://www.mscsoftware.com/products/adams.cfm>> [accessed 06.01.09].
- [7] Casolo F, Cinquemani S, Cocetta M. Functional mechanical design of a low cost driving simulator. In: *Proceedings of the 5th international symposium on mechatronics and its applications (ISMA08)*, Amman, Jordan; May 27–29, 2008.
- [8] Nehaoua L, Mohellebi H, Amouri A, Arioui H, Espié S, Kheddar A. Design and control of a small-clearance driving simulator. *IEEE Trans Vehicul Technol* 2008;57(2).
- [9] Force Dynamics. <<http://www.force-dynamics.com/>> [accessed 06.01.09].
- [10] Flight & Motion. <<http://www.flightmotion.com/>> [accessed 06.01.09].
- [11] Wentink Mark, Valente Pais Rita, Mayrhofer Michael, Feenstra Philippus, Bles Wim. First curve driving experiments in the DESDEMONA simulator. In: *DSC 2008 Europe*, Monaco; 31 January–1 February, 2008.
- [12] Roza Manfred, Wentink Mark, Feenstra Philippus. Performance testing of the desdemona motion system. In: *AIAA modeling and simulation technologies conference and exhibit*, Hilton Head, South Carolina; 20–23 August, 2007.
- [13] NASA Ames Research Center. <<http://www.aviationssystemsddivision.arc.nasa.gov/facilities/vms/index.shtml>> [accessed 06.01.09].
- [14] Genta Giancarlo. *Motor vehicle dynamics, modeling and simulation*. Singapore, New Jersey, London, Hong Kong: World Scientific Press; 1997.
- [15] Wong JY. *Theory of ground vehicles*. 4th ed. Hoboken, NJ, USA: John Wiley and Sons; 2008.
- [16] Smith Russell. Open dynamics engine. <<http://www.ode.org/>> [accessed 06.01.09].
- [17] Chicurel E. A 180° steering interval mechanism. *Mech Mach Theory* 1999;34:421–36 [Elsevier Science Ltd., 1999].
- [18] Pacejka Hans B. *Tire and vehicle dynamics*. Warrendale, USA: SAE; 2002.
- [19] Baffet Guillaume, Charara Ali, Stephant Joanny. Sideslip angle, lateral tire force and road friction estimation in simulations and experiments. In: *Proceedings of the 2006 IEEE international conference on control applications*, Munich, Germany; October 4–6, 2006.
- [20] Kamnik Roman, Bajd Tadej. Standing-up robot: an assistive rehabilitative device for training and assessment. *J Med Eng Technol* 2004;28(2):74–80 [Taylor and Francis Group, London, UK, 2004].
- [21] Blaschke F. The principle of field orientation applied to the new transvector closed-loop control system for rotating field machines. *Siem Rev* 1972;39:217–20 [Siemens AG, Erlangen, Germany, 1972].
- [22] NATO STANAG 4357 – AVTP "steering and manoeuvrability" (03-30), final ed., §4.4: steering effort. NATO Land Armaments Section, NATO HQ, Brussels, Belgium; 1991.