In Proceedings of the 9th ESA Workshop on Advanced Space Technologies for Robotics and Automation
'ASTRA 2006' ESTEC, Noordwijk, The Netherlands, November 28-30, 2006

A GROUND STATION FOR HUMAN-ROBOT-INTERACTION RESEARCH WITH EXOSKELETON

A. Schiele [(1)], M. De Bartolomei[(1)]

[(1)]European Space Agency
Mechanical Engineering Department, Automation & Robotics Section
Keplerlaan 1, 2201 AZ Noordwijk, The Netherlands
Email: Andre.Schiele@esa.int

INTRODUCTION

The European Space Agency aims at developing a humanoid-like assistant robot that can be used to support crew on the International Space Station and during possible future manned exploration missions to the Moon or Mars. This robot is called Eurobot [1]. Eurobot, equipped with anthropomorphic 7 degree of freedom (d.o.f.) arms, will be controllable in two major modes. During autonomous mode, the robot will perform pre-planned activities with support of its on-board autonomy system. The on-board autonomy system will monitor the robot environment and will help avoiding collisions with known or unforeseen obstacles.

In remote-manipulation mode, crew or operators from ground will be able to control the arms of the robot in direct tele-manipulation. The choice of appropriate man-machine interfaces for these tasks is important and many different technologies exist and are proposed. However, in space robotics, choice of input devices is mostly rather conservative. Control of presently available space manipulators, such as SSRMS on the space station, or RMS on the space shuttle is currently restricted to using a set of two 3 d.o.f. joysticks, without force-feedback. With the emergence of a number of smaller and more dexterous manipulators developed for space applications, such as DEXTRE, the Robonaut [2] or Eurobot, we have to reconsider whether conservative man-machine interfaces alone are still appropriate. The question arises whether interfaces such as joysticks, are sufficiently capable to exploit all functional advantages offered to the tele-operators by modern space robots. Novel input devices, such as haptic masters or exoskeletons might be better suited for execution of certain remote-control tasks. At ESA, currently two devices are considered for manual control of Eurobot, an exoskeleton and a conventional joystick workstation. It is still under discussion whether to use force-feedback or not. While exoskeletons can allow redundancy control on the fly, facilitate commanding of three dimensional trajectories and might ultimately lead to saving precious crew-time in operation and training, they also might have disadvantages related to controllability, safety or the ability to perform precise and confined tasks.

In order to rigorously trade off advantages and disadvantages of both systems, we decided to embark on a series of performance tests with both types of man-machine interfaces. We therefore started developing a ground demonstration facility that allows integrating a Joystick man-machine interface (MMI) and an Exoskeleton MMI with a prototype of the Eurobot robot.

It is our goal to perform a variety of realistic tasks with Eurobot in remote control to analyze the advantages and disadvantages of each system by means of performance metrics. Potentially, such analysis will provide us with data that can be used for man-machine interface design in the future. If advantages of conservative approaches and novel devices, such as exoskeletons, are merged, we believe, a truly intuitive system could be developed for robot control. We decided to investigate remote-control without force-feedback first, to investigate the fundamental differences between the devices. However, the facility used as a basis for those tests can be extended to include force-feedback control at a later stage.

It is the goal of this paper to provide a detailed description of the ground development facility, which is currently being developed to carry out performance analysis experiments with joysticks and exoskeletons for human-machine interaction. Moreover, a first pilot experiment is described and discussed, that demonstrates functionality of the currently available test system.

GROUND STATION IMPLEMENTATION

Overview

The ground test-bed developed at ESA consists of 4 major sub-systems. While some of them currently consist of commercially available items, in the near future they will be fully replaced with custom developed units.

A block diagram of the ground demonstration facility is presented in Fig. 1. The first major element is the Eurobot prototype with BarrettHand TM including their controllers. The second element is the robot workstation personal computer (PC), which contains the experiment software, the graphical user interface (GUI) for robot activation and monitoring and serves as central station to receive/distribute commands from/to the various man-machine interfaces.

The joystick and its interface are integrated with this sub-system directly. The robot workstation communicates with the robot controller via a proprietary fiber-optic link (ArcNetTM). The third major sub-system is the ESA exoskeleton (EXARM) together with its control station that communicates via Ethernet to the robot workstation. The EXARM has a direct link via analog and digital input lines to the exoskeleton workstation. A fourth element of the set-up is a graphical workstation PC used for displaying an animated 3D virtual reality model of the exoskeleton in real-time. Also this PC can be linked to the exoskeleton workstation via Ethernet.

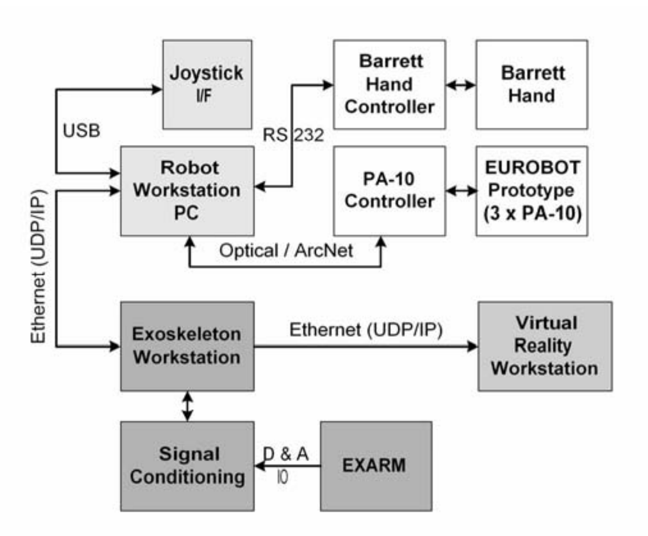
Eurobot Prototype

1) Mechanical Set-up: The Eurobot prototype currently consists of a triangular structural base, to which 3 Mitsubishi Heavy Ind., PA-10 robot arms are attached. The 7 d.o.f robots are attached in a symmetric configuration around the Eurobot body, at 120° distance intervals. The Eurobot structure is attached to the wrist of a large industrial 6 d.o.f. COMAU SMART-3 6.125A robot, which is used for weight support of the entire Eurobot prototype. The Comau robot is equipped with a wrist 6 d.o.f. force and torque sensor that will be used in the future for on-line gravity compensation during movement of the Eurobot prototype. However, the on-line compensation is not implemented at this time. One of the arms is equipped with a BarettHandTM end-effector [3], to allow grasping objects of regular and irregular shapes. This arm will be used for performing the man-machine interface comparison experiments. The other arms are equipped with TBK RH-707 Micro grippers. They can grasp robotically compatible interfaces. Pictures of the Eurobot prototype set-up are shown in Fig. 1 (right) and Fig. 4 (left).

2) Controllers: The PA-10 arms are controlled by their proprietary controllers at this time. They provide a software application program interface (API), that we used in the development of the experimental software on the robot workstation. The API allows a very low-level access to the robot controller functionality. Commanding of joint or Cartesian set-points at interpolator level is possible.
The BarrettHandTM is controlled via serial line (RS232) from the robot workstation PC. The hand motion-controller is located inside the Eurobot prototype body.
The COMAU robot controller is a derivative from a commercial version allowing low-level access as well, for flexible usage. It can be interfaced from a normal PC via a BIT3 VME/ISA bus adaptor.

Robot Workstation

1) Experiment Software: The overall architecture of the experiment software is shown in Fig. 2 (left). It has been developed under the Microsoft Visual C++ .NET environment and executes under Microsoft Windows XP on the robot workstation PC. The software is composed of a main GUI and several supporting software modules described hereafter.
The Motion Planner Module is in charge of controlling the PA-10 arm motion. This module exploits the PA-10 "Real Time Control Mode" in which Cartesian set-points are expected at a given frequency through the PA-10 C API I/F. The motion planner implements a periodic interpolator that runs every 40 ms. The interpolator is implemented as a function call-back attached to a timer.



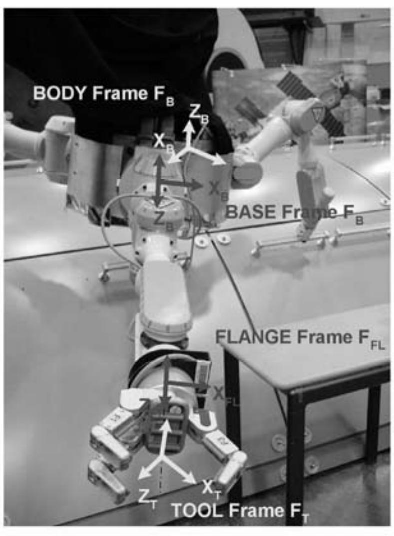


Fig. 1: (left) Block diagram of test-bed for prototyping and operation testing of man-machine interfaces for robot control. (right) Coordinate frame assignment of the Eurobot prototype (PA-10 standard frames shown in dark, added frames shown in light colour).

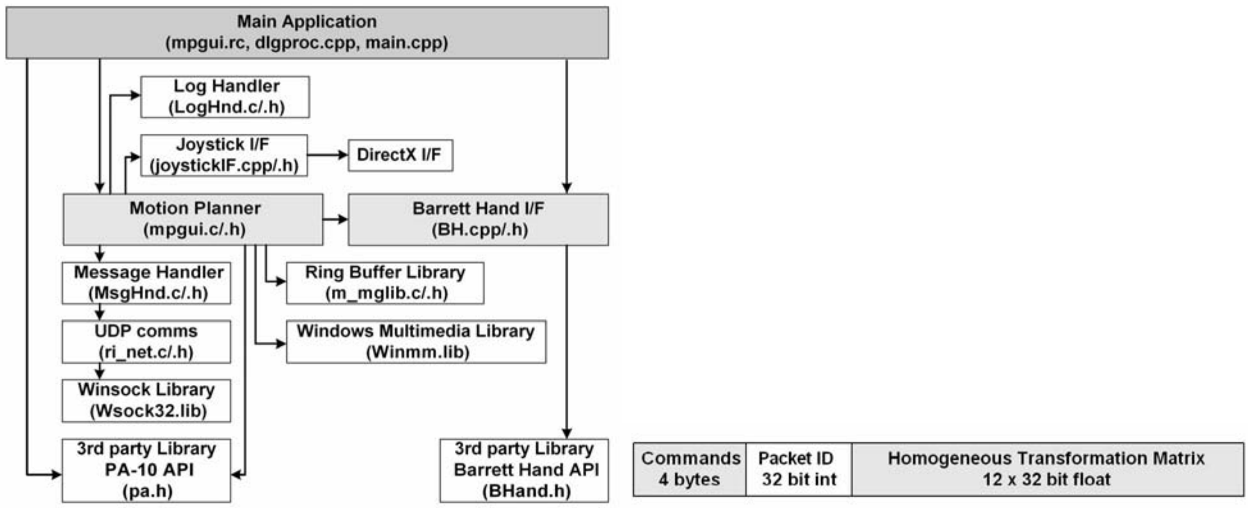


Fig. 2: (left) Block diagram describing architecture of the experiment software implemented on robot workstation PC. The multi-threading architecture provides real-time capability for robot control. (right) Implemented Datagram of UDP/IP protocol.

The timer makes use of the Microsoft Windows multimedia library. This timer is very precise and runs the call-back function at the operating system (OS) kernel priority. This ensures real time performance to the robot control. The interpolator can run in different modes that can be selected via a command or buttons on the GUI:

In HOLD mode, the same set-point is continuously sent and the arm remains still.

In SET-POINT mode, a destination frame can be specified via the GUI that must be reached by the robot TOOL frame in a given amount of time. The destination frame is reached via a linear trajectory. In JOG mode, it is possible to jog the arm via the GUI. The movement occurs linearly. It is possible to select the axis along/around which to translate/rotate the arm. It is furthermore possible to specify velocity and the reference frame in which motion occurs. In TRACKING mode, a dedicated software thread is started to deal with UDP / IP communications. This thread uses the Message Handler Module which in turn uses the UDP Communication Module. The tracking thread opens an UDP socket and waits for an incoming UDP datagram. When a datagram arrives, the message formatting is checked. If it complies with the message structure, the set-point inside the message is written into a FIFO ring buffer that is implemented in the Ring Buffer Module. This buffer is read by the motion interpolator that sends the set-point to the robot arm. The UDP message structure illustrated in Fig. 2 (right) has been implemented. The 1st byte in the command field denotes the tracking command. If this byte is set to 2, it notifies that the packet contains a set-point. The second byte contains the command for the BarrettHandTM (0 for Open, 1 for Close). The Packet ID is a monotonically increasing integer, starting from 0 at start-up. Assuming that the sender (e.g. the exoskeleton workstation) delivers the packets at the same frequency than the motion interpolator (of the PA-10), then, the packet ID can be used for communication acquisition and synchronization.

In JOYSTICK-CONTROL mode, a dedicated software thread is activated to read periodically the joystick inputs via the dedicated Joystick Interface Module. The inputs are acquired through the standard Microsoft DirectX interface. These inputs are sent, through the ring buffer, to the motion interpolator.

In TRACKING mode, the set-points sent to the arm are directly received from the UDP interface. In all other modes, it is the task of the motion interpolator to compute the following set-point to be sent to the robot manipulator. However, in all modes a velocity scaling algorithm is applied on the set-points prior to transmitting them to the PA-10. The algorithm works as follows: Let T be the interpolator period, $P(t-T)$ the arm pose reached at previous run of the interpolator and $P(t)$ the current set-point, then, the algorithm calculates translation and rotation magnitude to move from $P(t-T)$ to $P(t)$. For translational magnitude, a vector V is defined that describes the translation from $P(t-T)$ to $P(t)$. For the rotation, an angle Θ is defined that describes the rotation of $P(t)$ with respect to $P(t-T)$ around a vector R. R is formulated in angle-axis notation and derived from the set-point rotation matrices transmitted through the network. If speed limits are exceeded, the algorithm applies the following reductions: For translation speed, if the norm of V between two set-points is greater than the maximum allowed translation VMAX (user defined variable), a new translation component V' is recalculated, having same direction of V but magnitude equal to VMAX. For angular speed, if Θ is greater than the maximum allowed rotational THETAMAX (user defined variable) between two consecutive set-points, a new rotation component is recalculated as a rotation of THETAMAX around the same vector R. The motion planner module allows changing the maximum rotation and translation speeds at any time, even during motion, via the GUI or a joystick button input.

The BarrettHandTM is controlled via the Barrett Hand Interface Module. This module implements a dedicated low priority thread dealing with hand command requests (e.g. Init, Open, Close). This additional interface is necessary because the C API provided with the BarrettHandTM is composed by blocking functions, i.e. they do not return until the command is completed.

The Log Handler Module has been implemented for recording of the experimental data. It allocates memory for data storage in real time. It is possible to save data to files in several formats. The log handler is configurable and can record the set-points, elapsed experiment time, certain software state changes, the network performance by monitoring packet loss, BarrettHandTM status and others.

2) Control Approach: Both input devices, joystick and exoskeleton, generate Cartesian set-points that are exchanged as homogeneous transformation matrices between the sub-systems. The matrices are interpreted by the robot as translations and rotations of the robot TOOL coordinate frame with respect to the robot BODY frame. Fig. 1 (right) illustrates the assignment of coordinate frames to the Eurobot arm. The two additional frames TOOL and BODY have been defined in the experimental software, in order to allow issuing more generic motion commands to the robot. The relations between BODY-BASE and FLANGE-TOOL frames are configuration parameters in our software.
Kinematic inversion is computed by the PA-10 controller. The controller is configured to exploit the kinematic redundancy of the PA-10 to keep the joints as far from their stroke ends as possible during movement.
Motion commands are issued as follows: In TRACKING and SET-POINT modes, absolute commanding is used. The target set-point is then the desired position of the TOOL frame with respect to the BODY frame. In JOG and JOYSTICK control modes, relative commanding is used. Motion occurs with respect to the selected reference frame, which can be either the BODY or the TOOL Frame.

3) Graphical User Interface: The GUI contains various fields, message boxes and buttons to activate the various control modes, to send commands and to receive and log adequate state feedback.

4) Joystick Interface: Currently, a multi-d.o.f. Logitech Dual ActionTM joystick has been integrated with the set-up. A picture of it is shown in Fig. 3. Motion can be commanded only if dead-man button 1 is pressed for translation or dead-man button 2 for rotation. Joystick A is used by the left thumb to control X and Y axis motion, while Joystick B is used by the right thumb to control the Z axis movement. Via Button 3 and 4, the operator can choose the reference frame with respect to which the robot will be commanded (Button 3 for BODY or button 4 for TOOL frame). Frames can only be chosen when no dead-man button is pressed. If one of the dead-man buttons is pressed, Buttons 3 and 4 will decrease or increase the maximum speed of the robot. The Barrett Hand control button allows opening or closing of the BarrettHandTM.

Exoskeleton

1) Mechanical System: The ESA EXARM exoskeleton [4] [5] [6] is used as intuitive man-machine interface. The exoskeleton is weight-suspended by an external cable and counter-mass system in order to simulate usage inside a low-gravity environment. A picture of the exoskeleton during operations is provided in Fig. 4. The counter balance system does not influence dexterity of the device but fights fatigue of the operators on ground.
The EXARM features an advanced kinematic design that allows the users full freedom of arm motion. With EXARM, about 90% of all natural arm postures can be commanded to the remotely controlled robot [7]. Furthermore, to wear the EXARM, no adjustments to individuals are necessary. This feature enables extremely short dress-on and dress-off times (i.e. about 20 sec. only), which is desirable if the device shall be used by astronauts inside the International Space Station. Currently, the EXARM is a sensing device only. However, a number of single-joint actuation units have been developed in parallel, to investigate the optimal actuator configuration for force-feedback in a master-slave scenario. A Hand-held push-button device allows the operator to transmit a set of commands to the robot workstation.

Fig. 3: Logitech Dual Action Joystick currently used as input device for robot control experiments. Shown are the buttons that have additional functionality

2) Exoskeleton Workstation: The EXARM has 16 angular sensors that interface to the exoskeleton control PC via analog input lines. All sensors are high-precision conductive plastic potentiometers. They are powered by independent stabilized +5 V supply lines and their output signals are conditioned by impedance changers and 5th order Butterworth anti-aliasing filters. Anti-aliasing protection proofed to be important when testing the system together with the PWM motor drives of the actuation units for force-feedback. The exoskeleton workstation operating system is the real-time XPC target™. It is a flexible and easy to program environment that integrates with MATLAB Simulink and Real-Time Workshop. After acquisition of the 16 position channels into the exoskeleton control PC, a 2nd order digital filter stage smoothes the signals for further processing. Data acquisition and all further processing are performed within 1 ms intervals. All joint position values are linearly interpolated at each time step with calibration values that are hard coded.
The interpolated values are used in a forward kinematic function to derive the homogeneous transformation matrix from exoskeleton CHEST coordinate frame to exoskeleton HAND frame. The matrix values are then filtered again (2nd order low pass) and send via UDP/IP to the robot controller. The UDP transmission occurs every 40 ms.
The robot controller interprets the received transformation matrix as transformation from robot BODY to robot TOOL frames (see Fig. 1 (right) for reference). In order to improve the geometrical match between the robot and human arm posture, the robot TOOL frame was rotated around the y-axis of the FLANGE frame.
Furthermore, all position elements of the matrix are scaled to appropriately match the size difference between the human arm and the PA-10 robot. The network transmission speed is limited mainly by the current robot workstation capabilities (Operating system). Transmission speed from the exoskeleton workstation could be significantly increased, up to about 1 kHz. This will be important when extending the facility to include force-feedback from the robot to the MMI.
The hand-held push-button device makes use of digital input lines, to change the command bytes in the UDP datagram sent to the robot controller. This way the operator wearing the exoskeleton can start and stop transmission of set-points at any time during the experiments. At each start-up, the robot aligns itself smoothly to the current position transmitted by the exoskeleton.

Virtual Reality Workstation

The forward transformation matrices as well as the raw joint values of the exoskeleton are sent via UDP/IP to the Virtual Reality (VR) workstation PC. Transmit intervals are slower, at 500 ms, but sufficient for the graphical display.
The use of a 3D display workstation is mainly important during development of novel kinematic algorithms for exoskeleton-robot control. Visual feedback of device postures is crucial in the debugging phase of such developments. In Fig. 4 (right), a model of the exoskeleton with the operator is shown.
The model is based on an accurate kinematic description of the exoskeleton and was used to check the results generated by the forward kinematics functions. Each joint can be actuated by the acquired raw joint sensor signals of the real exoskeleton. Furthermore a helper coordinate frame (FKIN frame) attached to an easily visible yellow sphere is included. The helper-frame position and orientation is driven by the transmitted forward homogeneous transformation matrix that is computed in the exoskeleton workstation. The visual display of the computational results allows a quick check of correctness of calibration values and proper functioning of the algorithms.
Other visualizations are available, such as a graphical model of the PA-10, that will be used later during development of the exoskeleton.

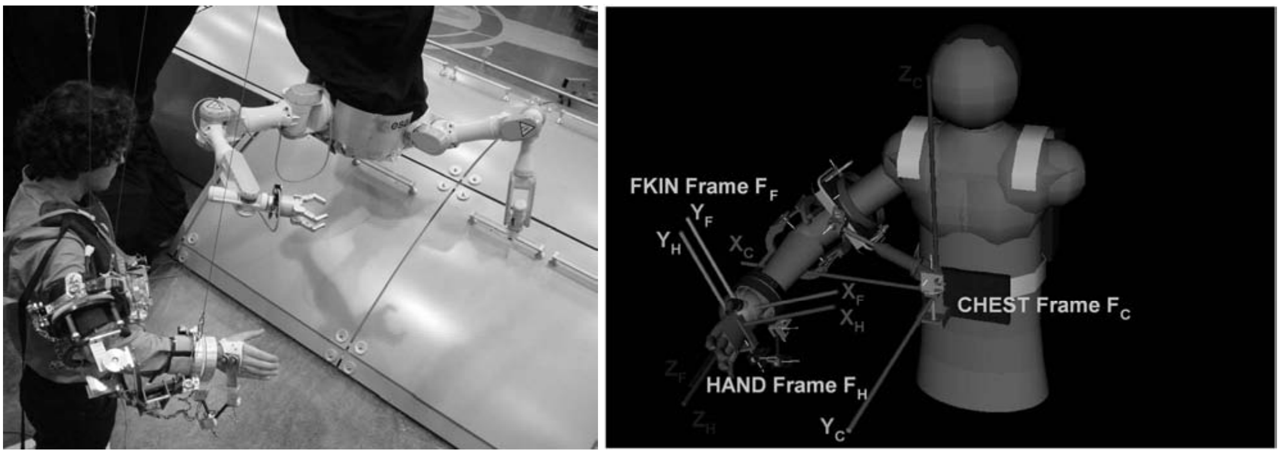


Fig. 4: (left) Overview of the Exoskeleton while being used to control the Eurobot prototype. The cables, used to counterbalance the exoskeleton are visible. (right) 3D Virtual Reality Model of the ESA Exoskeleton. The model can be controlled via Ethernet and can be used for development and debugging.

EXPERIMENTAL VALIDATION

Experimental Setup

In order to demonstrate proper functioning of the test-bed, a first set of pilot experiments was conducted with joystick and exoskeleton control. A simple tracking task was given to the operator, to see whether data can be properly logged with the experiment software and to test whether the motion planner module works smoothly together with all interfaces and the robot.

The experimental task consisted of following a simple linear trajectory with the tip of a tool attached to the robots gripper. The trajectory was defined by four points along the edges of a foam parallelepiped placed inside the workspace of the robot arm. The task had to be completed in a limited and predefined time of less than 30 seconds. The position of the foam geometry with respect to the robot BODY frame was calibrated with an absolute position error of about 0.02 m. For the purpose of these first experiments the error was considered acceptable. The geometries location was identified by recording the positions of the robot when touching some of the way-points indicated on the object. Both man-machine interfaces, the joystick and the exoskeleton were used consecutively by the operator for commanding the robot. The trajectory set-points that the robot followed as well as the time required to follow the line was recorded. During usage of both man-machine interfaces, the operator was standing at the same location with respect to the Eurobot prototype and had the same view on the robot work-cell.

Discussion of First Results

In Fig. 5, two trajectories are shown, that have been recorded during the task execution of the operator. Absolute distances are shown with respect to the Eurobot BODY coordinate frame. It can be seen that the operator was able to fulfill the task and that the data logging function, the motion commanding and the interaction between joystick, exoskeleton and the robot functioned properly. Network related packet loss was monitored when using the Exoskeleton, but did basically not occur. Not a single Ethernet packet was lost in 10.000 transmitted packages. Considering the shape of the shown trajectories, it can be already easily noticed that significant differences exist in characteristics of the two man-machine interfaces. The Joystick can only command linear motions. This limitation requires a precise alignment of the TOOL frame to the geometry that needs to be followed. It can be seen, that the joystick was often moved into a completely wrong direction and then corrected. This is a result of the initial trial and error approach that operators follow when their own, visual reference coordinate system is not aligned with the TOOL coordinate frame of the robot. One reason for this strategy might be the limited time, which was available for this task. Otherwise the operator would probably check and compare the axis indicators on the Joystick and the robot, which is time consuming.

With the exoskeleton, the operator had no problem in following the right direction. However, it can clearly be seen that straight-line following is more difficult than expected. The trajectory is relatively 'bumpy' as a consequence. Both trajectories were recorded early after completion of the set-up. The operator was not trained. This could be one reason why the trajectories do not look very appealing. In any case, however, it is interesting to notice that successful execution of even such a simple task seems challenging at first, even with the intuitive exoskeleton interface.

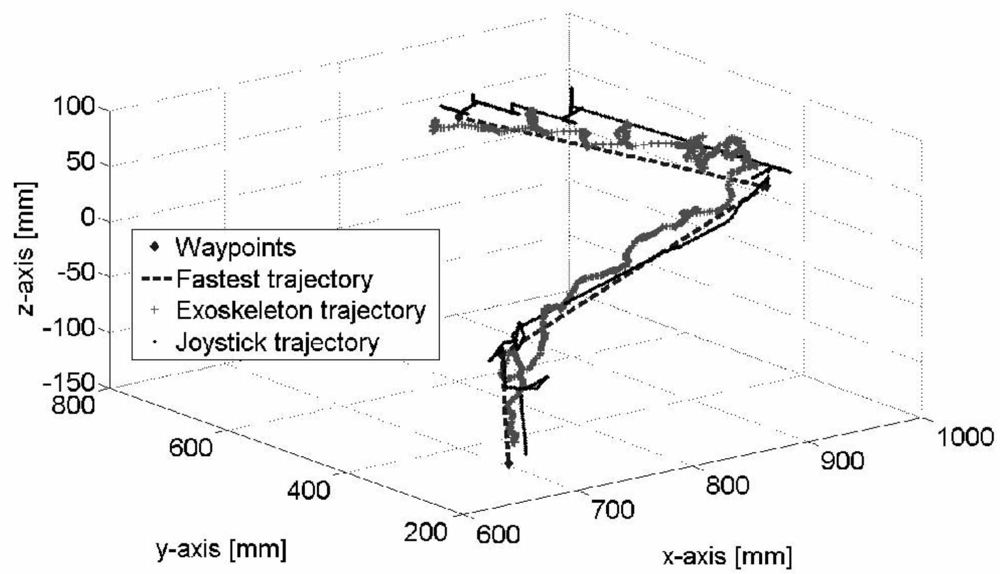


Fig. 5: Extract of experimental data from line-tracking experiment. Data is logged by experiment software and shows trajectories commanded with Joystick and Exoskeleton input devices.

CONCLUSIONS AND FUTURE WORK

The detailed design of a ground development facility was shown, that can be used for conducting tele-control experiments with a robot and two types of man-machine interfaces, joysticks and exoskeletons. The robot workstation can log the experimental data. Such data can be used to analyze task performance and establish performance metrics. MMI performance analysis is required to investigate the advantages and disadvantages of conventional joystick over exoskeleton-type man-machine interfaces. A simple experiment was conducted with both MMI's. The first results show that the facility functions properly and is ready to be used in a more comprehensive study of human-machine interaction performance.

At present, ESA performs a detailed definition of tasks, performance metrics and mental load assessment techniques that will allow comparing MMI performance during robotic remote-control. A series of detailed and controlled experiments with multiple subjects is currently under preparation. Investigation of the learning behavior, in dependence on the MMI-type, can give us important clues about how to design truly intuitive devices for our astronauts in the future. Also the hardware set-up of the facility will be modified in the coming months. A number of the commercially available items will be replaced with custom developed units from ESA. For instance, the commercial PA-10 controllers will be replaced with the ESA CONTEXT controller [8]. CONTEXT will fully replace the commercial controllers and offer simultaneous control of all 3 Eurobot arms in position, compliance or any combination of those. The CONTEXT hardware will be fully integrated into the mechanical structure of the Eurobot prototype. CONTEXT will also allow modification of the inverse kinematics functions, which will allow optimizing the geometric mapping between the exoskeleton and the robot arm. In the more distant future, the PA-10 robot arms will be exchanged with manipulators currently being developed for ESA in a dedicated research project [9] [10]. Furthermore, the commercial joystick will be replaced with a custom joystick station, similar to the one that is used in the Space Shuttle and on the ISS for manipulator control.

REFERENCES

[1] P. Schoonejans, et al., "Eurobot: EVA-assistant robot for ISS, Moon and Mars", Proceedings of 8th ESA Workshop on Advanced Space Technologies for Robotics and Automation, ASTRA, Noordwijk, November 2004.

[2] W. Bluethman, et al.,"Robonaut – A robot designed to work with humans in space", Autonomous Robots, Vol. 14, pp. 179 – 197, 2003.

[3] W.T. Townsend, "The BarrettHand grasper – programmably flexible part handling and assembly", Industrial Robot, Vol. 27, No. 3, pp. 181 – 188, 2000.

[4] A. Schiele, G. Visentin, "The ESA human arm exoskeleton for space robotics telepresence", 7th International Symposium on Artificial Intelligence, Robotics and Automation in Space, iSAIRAS, 2003.

[5] A. Schiele, G. Visentin, "Exosquelette pour bras humain, notamment pour des application spatialles", European Patent Pending, EP 1364755, 2003.

[6] A. Schiele, et al., "Exoskeleton for the human arm, in particular for space applications", U.S. Patent, US 2003223844, Dec. 2003.

[7] A. Schiele, F.C.T. van der Helm, "Kinematic design to improve ergonomics in human machine interaction", IEEE Transactions on Neural Systems and Rehabilitation Engineering, in press.

[8] P. Bologna, C. Mondellini, E. Crudo, F. Didot, L. Foresti, "CONTEXT – Space a&r controller capabilities extension", Proceedings of 8th ESA Workshop on Advanced Space Technologies for Robotics and Automation, ASTRA, Noordwijk, November 2004.

[9] A. Rusconi, et al., "DEXARM – A dextrous robot arm for space applications", Proceedings of 8th ESA Workshop on Advanced Space Technologies for Robotics and Automation, ASTRA, Noordwijk, November 2004.

[10] M. Hirt, G. Gruener, "Dextrous robot arm", Proceedings of 8th ESA Workshop on Advanced Space Technologies for Robotics and Automation, ASTRA, Noordwijk, November 2004.