# A TEST-BED FOR THE DEMONSTRATION OF MSS GROUND CONTROL

É. Dupuis*, J.-C. Piedbœuf*, R. Gillett**, K. Landzettel***, B. Brunner***

*Canadian Space Agency, 6767 route de l'aéroport, St-Hubert (Qc) J3Y 8Y9, Canada, erick.dupuis@space.gc.ca
**MacDonald Detwiller Space and Advanced Robotics, 9445 Airport Road, Brampton (On) L6S 4J3, Canada
***Deutsches Zentrum für Luft und Raumfahrt, Postfach 1116, D-82230 Wessling, Germany

## ABSTRACT

A test-bed is being implemented by the Canadian Space Agency for the testing and validation of ground control technologies for MSS operations. The test-bed is built around the MSS Operations and Training Simulator. The first two ground control software implementations to be tested on the test-bed will be the Modular Architecture for Robot Control developed by DLR and the Intelligent Interactive Remote Operations system developed by a Canadian team that included the Canadian Space Agency and MacDonald Detwiller Space and Advanced Robotics.

## 1. INTRODUCTION

Canada's contribution to the International Space Station (ISS) is the Mobile Servicing System (MSS) which is composed of the Mobile Remote Servicer Base System (MBS), the Space Station Remote Manipulator System (SSRMS) and the Special Purpose Dextrous Manipulator (SPDM). The planned mode of operation for SSRMS and SPDM is teleoperation by an astronaut at the robotics workstation inside the ISS. Contrarily to SRMS operation in the Space Shuttle, the astronaut will not have any direct view of the manipulators. The visual feedback is obtained through cameras connected to three small LCD screens. It is predicted that the operation of MSS will consume a lot of crew time because of the low velocities at which the SSRMS and SPDM will be operated and because of the potentially large displacements to be performed by these manipulators.

As an alternative to reduce the load imposed on astronauts part of the MSS operations could be conducted from a ground station. SSRMS and SPDM provide control modes that would be suitable for ground operations and some of the hooks and scars necessary for ground control are already in the software. However, ground control is hampered by communication link limitations such as time delays and bandwidth and by the lack of good situational awareness of the operator. In this context, *situational awareness* refers to the operator's knowledge of the spatial relationships amongst the work site equipment, features and obstacles. Situational awareness is impeded in any remote operation where the operator is limited only to equipment-mounted camera views with which to perceive the work site. To ensure that MSS operations could be safely carried-out from the ground, it is necessary to conduct a demonstration of ground control technologies in a realistic environment.

The Modular Architecture for Robot Control (MARCO) developed by DLR is a spin-off of the ROTEX flight experiment conducted by DLR in 1992. Subsequent to the ROTEX experiment, the ground segment has been further developed to add more capabilities to the system. In 1999, MARCO was used to teleoperate the robot manipulator on the Japanese ETS-VII satellite (Ref. 2).

The Canadian Space Agency (CSA), MacDonald Detwiller Space and Advanced Robotics and a team of industry, universities and government organisations has developed an architecture for intelligent interactive remote operation (IIRO) of machinery. The IIRO project was developed as a generic architecture to satisfy the needs of both space and industrial mining operations. The IIRO system successfully demonstrated the remote operation of an excavator in Northern Canada and that of a laboratory robot in space-representative conditions from the Canadian Space Agency headquarters near Montréal. (Ref. 4)

In addition to the systems mentioned above, researchers and space agencies have developed ground control software. For example, ESA has been working in ground control technologies since 1989. Their most recent developments are SPARCO and FAMOUS, and the ground control of ETS-VII using VIABLE. ESA is currently working on the implementation of these technologies on the JERICO test-bed. NASDA has controlled the robotic manipulator on ETS-7 from Earth-based stations (Ref. 6) and a team from Texas A&M University are working on the development of ground control software for AERCam (Ref. 7)

To validate the concept of ground control of MSS in a representative environment, CSA is currently developing a ground-control test-bed on which the MARCO and IIRO systems will be tested. The objective is to faithfully reproduce the interface and dynamics of MSS as well as the communication limitations. One of the main components of the test-bed is the MSS Operation and Training Simulator (MOTS): a real-time simulator currently used for MSS operator training and for operation planning. To simulate MSS ground control from systems such as MARCO and IIRO, CSA will add an interface to MOTS that will allow it to receive commands and transmit telemetry in the same fashion as the MSS will through the ISS command and telemetry servers.

This paper describes the requirements and implementation of a test-bed used to validate ground control technologies in the context of MSS operations. It also provides an overview of two ground control software implementations that will first be integrated into the test-bed.

## 2. TEST-BED REQUIREMENTS

The test-bed will be used to validate the concept of ground control and compare various technologies in the context of MSS operations. Third party ground control software will be connected or integrated into the test-bed for testing and validation.

It must therefore provide a high fidelity simulation of MSS operations accurately emulating the rigid and flexible body dynamics of MSS, its control software including the relevant control modes and features as well as all relevant environmental effects.

In addition, the test-bed must emulate the communication conditions between a potential ground station and the on-orbit system including time delays, jitter, band-width limitations, blackouts and routing of commands and telemetry.

Finally, the test-bed must provide camera views that are representative of the images down-linked from on-orbit cameras including all camera and lighting effects.

To be a useful validation tool, the test-bed must support known and future approaches to ground control (predictive displays, supervisory control, tele-sensory programming) to the extent that the MSS will support them. It must have the ability to receive commands from and send telemetry to third party ground control software. Finally, it should provide a real-time dynamic engine for driving predictive simulations.

A difficulty in ground control of MSS is the acceptance of the concept by the astronaut. As an intermediate step to facilitate the acceptance, the test-bed should allow the implementation of the ground control system directly on the Robotics Workstation. In this mode, the controller will reduce the complexity of the task to be done by the astronaut.

## 3. TEST-BED SYSTEM ARCHITECTURE

The test-bed architecture is illustrated in Figure 1. It features a dynamic simulation engine built around the MSS Operations and Training Simulator (MOTS) and it emulates the communication link between the ISS and a ground station. For the moment, it will incorporate ground control technologies from previous projects (IIRO and MARCO). The architecture is flexible and will allow for the testing and validation of various ground control technologies.

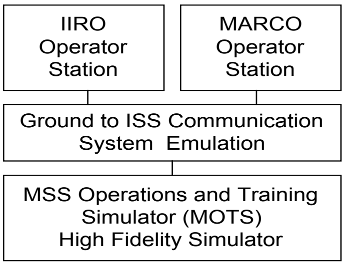

Figure 1 - Test-bed Architecture

### 3.1 MSS Operations and Training Simulator

The MSS Operations Training Simulator (MOTS) facility is a high fidelity real-time simulation environment coupled with monitoring and control functions for effective training, operations development and verification. To support training activities, MOTS features replicas of the MSS Crew Station (CS) hardware including operational hand controllers, real-time simulation of the CS Human-Computer Interface (HCI), video displays, lighting, audio and failure mode conditions.

The dynamics engine simulates the rigid and flexible body dynamics of MSS and of the ISS in real-time at an update rate of 1000 Hz. The control system model runs both the MSS flight code and a validated simulation model. In the absence of flight data, the open loop and closed loop responses of the simulator have been validated by comparing them to the MSS dynamics reference model.

Since the MOTS is used for operator training, it includes a high-fidelity graphics engine generating 3D views that are representative of on-orbit conditions. The virtual cameras have the same properties as their on-orbit counter parts and are commanded using the same user-interfaces.

The MOTS can be used to generate telemetry using the same protocol as will be used for the distribution of data to ground stations.

### 3.2 Modular Architecture for Robot Control (MARCO)

Over the last years, DLR has focussed its work in space robotics on the design and implementation of a high-level task-oriented robot programming and control system. The goal was to develop a unified concept for a flexible, highly interactive, on-line programmable teleoperation ground station as well as an off-line programming system, which includes all the sensor-based control features partly tested in the ROTEX (Ref. 1) scenario. But in addition it should have the capability to program a robot system at an implicit, task-directed level, including a high degree of on-board autonomy.

The current system provides a very flexible architecture, which can easily be adapted to application specific requirements. To get the robots more and more intelligent, the programming and control methodology is based on an extensive usage of sensors, such as cameras, laser range finders, and force-torque sensors. It combines sensor-based teleprogramming (as the basis for on-board autonomy) with the features of teleoperation under time delays (shared control via operator intervention). Applications in a well-known environment, e.g. to support or even replace an astronaut in intra-vehicular activities, can be fully pre-programmed and verified on-ground – including the sensory feedback loops – for further sensor-based execution autonomously on-board. A payload user, who has normally no expertise in robotics, can easily compose the desired tasks in a virtual world. As man machine interface, a sophisticated VR-environment with DataGlove and high-performance graphics is provided. Service tasks, e.g. assembling and maintenance of ISS modules or catching and repairing a failed satellite, require a high amount of flexibility in programming and controlling.

The flexibility of the system, to control nearly each kind of space robotics applications, has been demonstrated in ROTEX 1993 and in spring 1999, where the Japanese ETS-VII (Ref. 2) space robot was controlled from ground by this system.
Currently, the programming and control system is applied to the ESA funded mars rover study ROBUST, where the autonomy of the flight system has been increased even further, due to the very large time delays.

#### 3.2.1 Programming and Control Methodology

A non-specialist user – e.g. a payload expert – should be able to remotely control the robot system in case of internal servicing in a space station (i.e. in a well-known environment). However, for external servicing (e.g. the repair of a defect satellite) high interactivity between man and machine is demanded. To fulfil these requirements, the design of the programming system is based on a 2 in 2-layer concept (See Figure 2), which represents the hierarchical control structure from the planning to the executive layer:

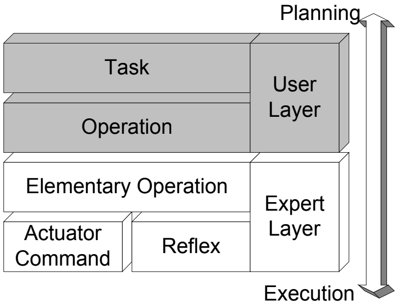

Figure 2 - MARCO Control Methodology

On the user layer the instruction set is reduced to **what** has to be done (planning level). No specific robot actions will be considered at this task-oriented level. On the other hand the robot system has to know **how** the task can be successfully executed, which is described in the expert layer (execution level). For details see (Ref. 3).

### 3.2.2 Expert Layer

At the lowest system level, the sensor control mechanism is active. The teaching by showing paradigm is used at this layer to show the reference situation, which the robot should reach, from the sensor's view: in the virtual environment the nominal sensory patterns are stored and appropriate reactions (of robot movements) on deviations in the sensor space are generated. This programming layer is completed by the Elemental Operation (*ElemOp*) level. It integrates the sensor control facilities with position and end-effector control.
In telemanipulation mode, the user generates position commands and selects the appropriate sensor control strategies for path refinement (shared control).

### 3.2.3 User Layer

This task-directed level provides a powerful man-machine-interface for the robotics user. An *Operation* is characterised by a sequence of ElemOps, which hides the robot-dependent actions. For the user of an Operation the manipulator is fully transparent. To apply the Operation level, the user has to select the object/place, he wants to handle, and to start the Object-/Place-Operation. Via a 3D-interface (DataGlove or SpaceMouse) an object can be grasped and moved to an appropriate place. After the user has moved all the objects to their target locations, the execution of the generated *Task* can be started. The system provides status information and comprehensive quick look displays (See Figure 3) for task execution monitoring purposes.

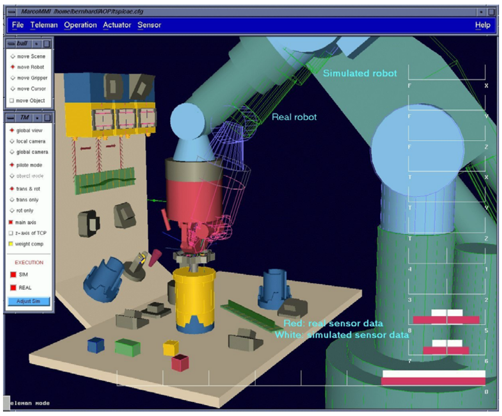

Figure 3 - 3D-Viewer and quick look display

### 3.2.4 System interfaces

In a first implementation phase DLR's telerobotic system will be interfaced to MOTS via an ISP client to establish the command and telemetry interfaces. Therefore the MARCO interface data structures have to be adapted to the existing MOTS interfaces and their timing. A second implementation phase is to allow MARCO to use MOTS as a dynamic engine to close control loops in the ground simulation. The current input devices will be complemented by two joysticks, one for position and the other for orientation control of the manipulator, to build a high fidelity replica of the on board user interface.

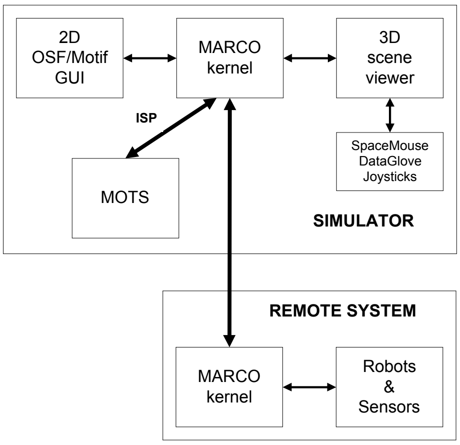

Figure 4 - Architecture: MARCO Demonstration

## 3.3 Intelligent Interactive Remote Operations (IIRO)

### 3.3.1 The Generic IIRO Concept

The IIRO project addressed the inherent challenges of control over long distances - operator situational awareness and communication limitations - in a generic manner. The system addressed modelling and rendering of highly unstructured and unknown environments, complex mission script generation and rehearsal, and programming of robot-generated event-driven decisions. This solution was aimed at remote operations across the earth, or in low earth orbit and beyond to planetary exploration (Ref. 4).

A high level diagram of the generic IIRO system concept is shown in Figure 5

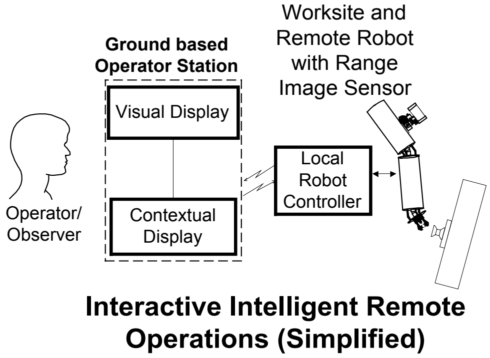

Figure 5 - Generic IIRO System Concept

The generic IIRO system begins by providing means for a remote operator to acquire a high level of situational awareness regarding the worksite surroundings of the robot.

Improved situational awareness is achieved by acquiring 3-D range image data from the surrounding workspace and by inserting a computer-generated CAD model of the manipulator, correctly registered within this model to create a virtualised reality model of the complete remote worksite.

Within this visually oriented, virtual environment display (Visual Display) the IIRO system employs motion scripts that are generated and rehearsed at the operator station; definition and creation of mission scripts is facilitated by an accompanying graphical display presenting an abstraction of missions scripts as a flowchart (Contextual Display). The Visual workstation allows the operator to view the remote equipment from any desired viewing angle, and to define motion sequences for this equipment by positioning the computer-modelled remote equipment and storing the successive positions as keyframes. Keyframes are then stored in groups as linked blocks in a flowchart display on the Contextual workstation. This workstation provides the capability to edit, copy and rearrange these blocks to form large and complex mission scripts. Equipment motion is rehearsed at the operator station using the remote equipment computer model.

Once completed and rehearsed to the satisfaction of the operator, the script information is uploaded from the Contextual workstation to the remote equipment on the actual work site, where the script is executed with the same interpolation function by the remote equipment controller. All equipment telemetry is displayed in context on the Visual workstation so that the operator can observe the progress of the mission.

The scripting capability also allows the introduction of programmable logical branching conditions that are used by the remote equipment to decide how to proceed at branch points as determined by local sensory information. Logical branching permits complex missions - and predetermined responses to local stimuli - to be executed autonomously by the machine, while maintaining tight operator control over the equipment behaviour. Local branching of control strategies is particularly useful for mission scripts that perform the complex task of autonomous digging. The IIRO project concluded with a demonstration of scripted excavation and precision grasping operations over an Internet connection (Ref. 5).

An example of the IIRO virtualised reality environment (Visual display) a seen through virtual camera views and an accompanying photograph of the actual laboratory work site are shown in Figure 6 and Figure 7.

```
1.25 Hz [  2.9  2.0  1.1] [ 92      2.1]
Angles [deg]:   0.67  0.01  78.13
Loc: [2.458 1.906 2.116] [0.000 0.000 0.000]
```

Figure 6 - IIRO Visual Display with Robot and Scene

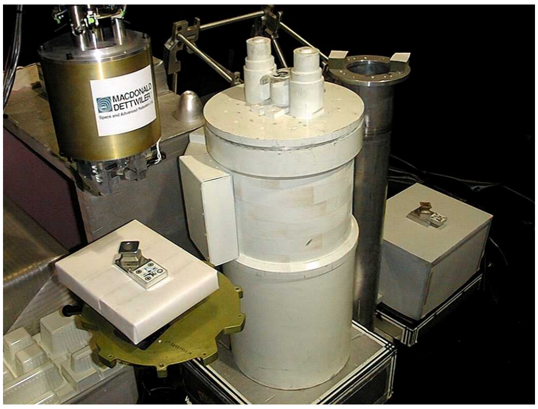

Figure 7 - Actual Laboratory Scene

### 3.3.2 IIRO Applied to MSS Ground Control

When the IIRO concept is applied to control MSS operations on the International Space Station from the ground, a subset of the above capabilities are needed due to the reduction of some of the challenges when compared to planetary robotic operations.

The two most notable differences between planetary robotic operations and low earth orbit operations are lower latency/higher availability of communications in low earth orbit, and above all the need for safety when operating in a manned space flight environment. The reduced round trip latency for command and control communications allows MSS operations in LEO to proceed with higher levels of interaction between the ground-based operator and the on-orbit system.

In the case of MSS operations, a plethora of data is available regarding the equipment and the workspace surrounding the robots. The MSS operates in a highly structured environment about which a great deal of data exists regarding physical dimensions and relative locations of objects and features. However this data may contain inaccuracies that arise from thermal distortions and manufacturing/configuration discrepancies that introduce potentially significant errors in the computer models.

To correct for these error sources, the computer model will serve as the starting point and an assessment will then be made as to its accuracy and precision with respect to the actual ISS situation. Any errors will be corrected using computational stereo vision 3-D data that will supplant any erroneous portions of the model. While the entire scene can be reconstructed in this manner, it is anticipated that this step is needed only to correct for the location of robot interfaces for which this higher precision is required to support successful operation. The remainder of the scene model should simply need verification for acceptable tolerances.

Once constructed with suitable accuracy the virtualised reality model, in conjunction with the contextual script editor display, will provide the operator with the tools necessary to construct and upload for execution complex mission scripts that include error handling and recovery to maximise operational safety.

As an enhancement for ISS operations safety, during execution of an uploaded mission script existing ISS camera views of the work site will be fed to an on-orbit vision processor that will provide a background check on the manipulator configuration and its proximity to space station structures. This process will spontaneously issue warnings and potentially emergency stop commands upon detection of hardware failures or operator errors that could induce a collision.

## 4. CONCLUSION

This paper has described a test-bed being implemented by the Canadian Space Agency for the testing and validation of technologies for the ground control of the MSS. The test-bed is being built around the MSS Operations and Training Simulator (MOTS). This will ensure that the ground control technologies are tested under conditions that are representative of MSS on-orbit operations. This will ensure that the ground control systems take into account the capabilities and limitations of MSS operations.

At first, the test-bed will incorporate ground control technologies from the CSA and DLR in the form of an IIRO ground station and a MARCO ground station. These will be tailored to take into account the capabilities and limitations of the MSS in terms of control modes and operational procedures.

The first major milestone of the project will be a demonstration in June 2001 of the system being controlled both by an IIRO ground station and a MARCO ground station. This will allow for a first evaluation of complementary technologies.

The next step will be the development of a ground station that can be used to control the real MSS manipulators. The goal is to make a demonstration of these capabilities during a mission. As an intermediate step, a demonstration can be done using the Robotics Workstation inside the space station.

## 5. REFERENCES

1. Hirzinger G., et.al., Sensor-Based Space Robotics – ROTEX and its Telerobotic Features, *IEEE Transactions on Robotics and Automation*, Vol.9, No.5, Oct. 1993.
2. Landzettel, K. et.al., A Unified Ground Control and Programming Methodology for Space Robotics Applications — Demonstrations on ETS-VII, *31st International Symposium on Robotics*, May 14-17, 2000, Montréal, Canada.
3. Brunner, B., et.al., A Universal Task-Level Ground Control and Programming System for Space Robot Applications, *i-SAIRAS 5th International Symposium on Artificial Intelligence*, Robotics and Automation in Space, 1-3 June 1999, ESTEC, Noordwijk, The Netherlands.
4. Dupuis, E. et.al., Interactive, Intelligent Remote Operations: Application to Space Robotics, *Telemanipulator and Telepresence Technologies VI*, 1999.
5. Gillett R. and Lipsett M., Robotics Leading the Way to Sustained Human Presence in Space, *31st International Symposium on Robotics*, May 14-17, 2000, Montréal, Canada.
6. Kasai, T. et.al., Results of the ETS-7 Mission - Rendezvous Docking and Space Robotics Experiments, *i-SAIRAS 5th International Symposium on Artificial Intelligence*, Robotics and Automation in Space, 1-3 June 1999, ESTEC, Noordwijk, The Netherlands.
7. Borst, C.W. and Volz, R.A., Telerobotic Ground Control of a Free-Flying Space Camera, *Robotica,* vol. 18, pp. 361-367, 2000.