Proceeding of the 7th International Symposium on Artificial Intelligence, Robotics and Automation in Space:
i-SAIRAS 2003, NARA, Japan, May 19-23, 2003

On-Orbit MSS Training Simulator

Régent L'Archevêque Serguei Bedziouk Eric Martin Michel Doyon József Kövecses Pierre Allard
Canadian Space Agency,
6767 Route de l'Aéroport, St-Hubert, Québec, Canada, J3Y 8Y9
regent.larcheveque@space.gc.ca

Keywords Simulator, Training, Mobile Servicing System, International Space Station, Symofros

Abstract

Experimental tests and analysis have shown that the capture of free-flyers is the most complicated task to be performed by a robotic operator on board of the International Space Station (ISS). The understanding of Mobile Servicing System (MSS) and free-flyer dynamics require highly qualified and well-trained operators. The dexterity and accuracy of the astronauts may decrease over time if they are not trained on-board. It was an obvious choice to have a simulator on-orbit to keep the skills of the astronauts at the required level. In order to support the training scenarios required by the on-orbit training, the SMP[1] simulator has been developed. This paper presents the goals and the architecture of SMP and provides some results obtained with the system.

1. Introduction

The MSS is a critical element of the ISS and will be used to assemble and maintain the Station throughout its lifetime in orbit. A number of factors are known to affect MRO performance on-orbit: psychological and physiological stress of space flight, pre-flight training fidelity, skill degradation over time, man-machine compatibility, etc. While pre-flight training ensures that the operators performance is reliable at launch, many months can pass before crew get to use the MSS. A system capable of identifying operator weakness and providing the necessary remedial training on-board would be required to decrease the probability of crew error.

[1] System for Maintaining, Monitoring MRO Performance on board the ISS

2. On-Orbit Training & Simulators

The safety and success of MSS operations will depend on reliable performance by the MSS Robotics Operator (MRO). This is particularly important for the Human-in-loop modes involved in such critical tasks as Free Flyer capture, payload berthing/un-berthing, capture/release, EVA support, and Special Purpose Dextrous Manipulator (SPDM) positioning.

Crew time being an expensive resource on the ISS, training needs to be adapted to each individual, performed only when needed, and focused on the skills that degrades the most on-orbit. As a result of analysis of operators' pre-flight performance during training using the MSS Operation and Training Simulator (MOTS) and on-orbit operation using the MSS on the ISS, critical MRO skills have been identified and prioritized.

At the top of the list are the skills required to perform a Free Flyer capture. A Free Flyer capture involves significant target velocity and strict execution time constraint that requires excellent hand controller skills and complex tri-dimensional transformations from the operator.

SMP uses a new concept of on-board training developed at CSA. It includes the monitoring of pre-selected MRO critical skills using simulator exercises to identify MRO weaknesses based on analysis of on-orbit performance against pre-flight baseline. Based on such analysis, the training is adjusted and repeated until the operator performance recovers to the baseline level. In order to identify skill degradation, SMP implements the following elements:

1. Use of a skill-centered task (Free Flyer capture);
2. Objective performance index;

Figure 1: SMP Hardware Components

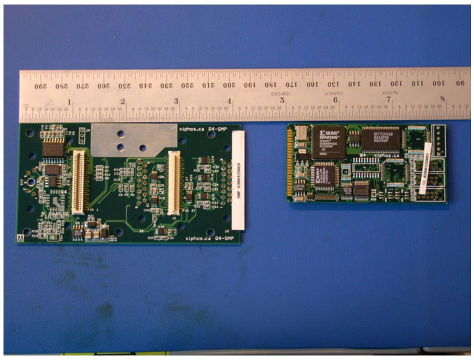

Figure 2: SMP Daugtherboard (left) and Q4 (right)

3. Immediate user performance feedback on a task by task basis;
4. A trend analysis of performance over time;

Implementing these elements requires a realistic on board simulation of the Free Flyer capture, the capability to record performance data on both pre-flight and on-orbit exercises, and a simple-to-use analysis tools to provide feedback to the crew as well as detailed analysis to the training community on the ground.

3. Hardware Architecture

As shown in Figure 1, SMP comprises an IBM Thinkpad Laptop (P-III 800MHz), hand controllers (similar to the ones used to operate the MSS) and an electronic interface module acting as a Universal Serial Bus (USB) interface on the computer side and as a data acquisition interface on the hand controller side.

The use of USB has numerous advantages, one of the most interesting being that connected devices can draw power from the host, thus not requiring separate power supply. The standard is widely in use in industry and thus has widespread, extensive software support. The controlling entity in the interface is the Q-Card. It supplies power to the controllers, samples the analog data as well as their digital button outputs and controls a USB link layer that handles the low-level protocol issues. The Q4-SMP USB interface unit consists of two circuit boards: a Q4 and a purpose-built daughterboard (Figure 2).

The Q4 is a COTS logic-based control device that was developed for aerospace control and data

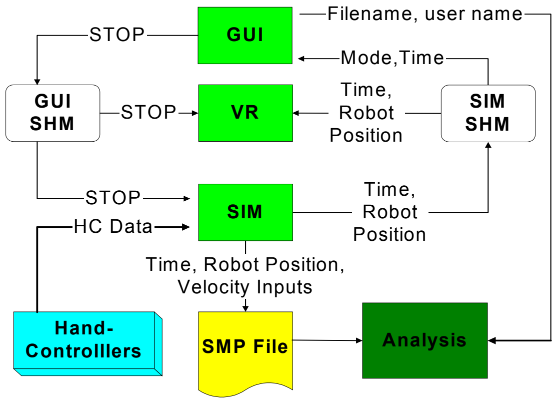

Figure 3: SMP Software Architecture

acquisition tasks. It is used unmodified in this system. The Q4 provides a 4.096V reference to the hand controllers, which is amplified on the daughterboard. Based on this voltage, it reads the analog position from each of the 6 axes, as well as a logic value from each of the 3 switches. These voltages are appropriately range-scaled, converted to the Human Interface Device (HID) format, and then passed to the USB link layer device (via I2C bus) for transmission to the host. The Q4 also handles the initial handshaking and identification with the host.

The daughterboard provides the USB link layer services, a current-limited power input stage, and connectors that interface to the existing joystick units. It also supports a serial interface for debugging, and an onboard temperature sensor for evaluation and test.

4. Software Architecture

The simulator includes four modules, the Graphical User Interface (GUI), the Analysis Module, the Visual Renderer (VR) and the Dynamic Simulator (SIM)

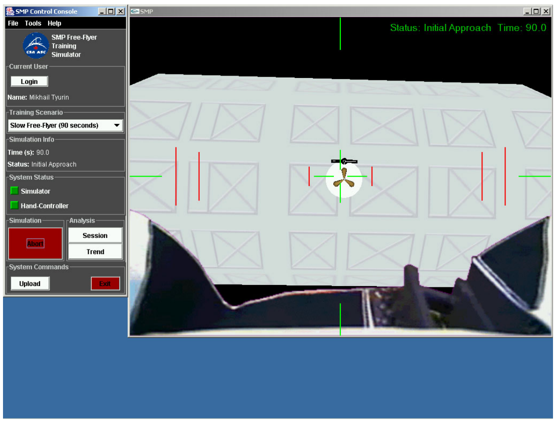

Figure 4: SMP Graphical User Interface

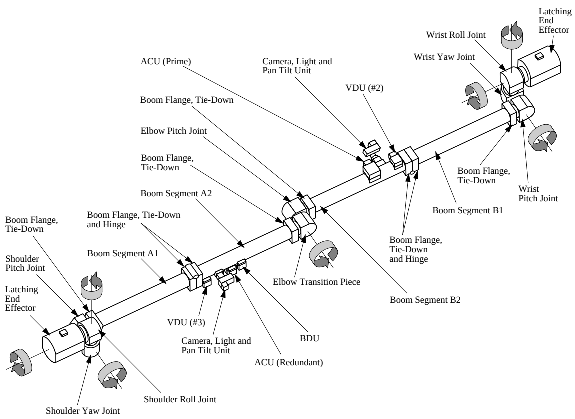

Figure 5: Overall SSRMS Architecture.

(Figure 3). It has the same architecture as the Basic Operations Robotic Instructional System (BORIS) simulator used to provide generic robotic training to the astronauts [2].

The GUI has been developed with Java and runs on Windows operating systems (Figure 4). It has the role of managing users logging, user session files, and user profiles. It also spawns other SMP modules and displays data during simulation or analysis sessions. At last it controls SMP modules execution by using the shared memory named *GUI SHM*.

The *VR* provides a virtual environment that models a free-flyer, the Space Station Remote Manipulator System (SSRMS) end-effector camera view and the SSMRS capture overlays (Figure 4). The engine is driven by the data generated in real-time by the simulator during simulation sessions.

4.1 SSRMS Modelling with Symofros

The SSRMS is a 7 DOF manipulator that is used for

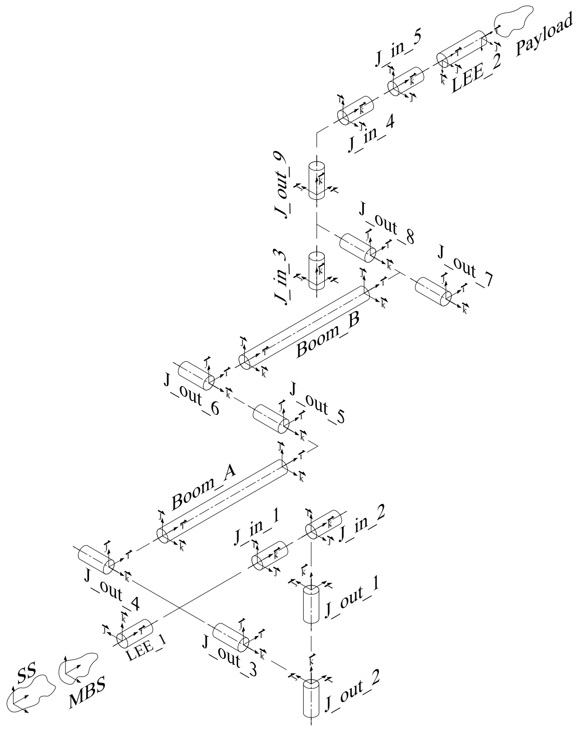

Figure 6: SSRMS links Synthesis.

the Space Station assembly. It is a relocating manipulator that can be attached to various points on the ISS by both ends terminated by Latching End Effectors (LEE). For that reason, the manipulator is symmetric with respect to the elbow joint. The overall architecture of the SSRMS is shown in Figure 5.

The SSRMS is composed of 8 links whose model can be synthesised by assembling standardised elementary body components. These body components can be classified in 5 categories: LEE, Joint_in, Joint_out, Boom A, Boom B. The 8 links of the SSRMS can be modelled by assembling specific body components as shown in Figure 6. These bodies are connected together using elastic joints. The elasticity in the SSRMS joints is significant due to the use of gearbox with high gear ratios of about 2000. The Boom A and Boom B components are flexible beams. To simplify the dynamics model, these bodies were modelled as rigid bodies and their elasticity was lumped in the joint. Again, to reduce the complexity of the model, the elasticity of the three last joints was not considered and the joints were modelled as rigid. This assumption can be made since the mass of the three last joints is small compared to the overall weight of the manipulator. Moreover, it is not intended to manipulate any payload

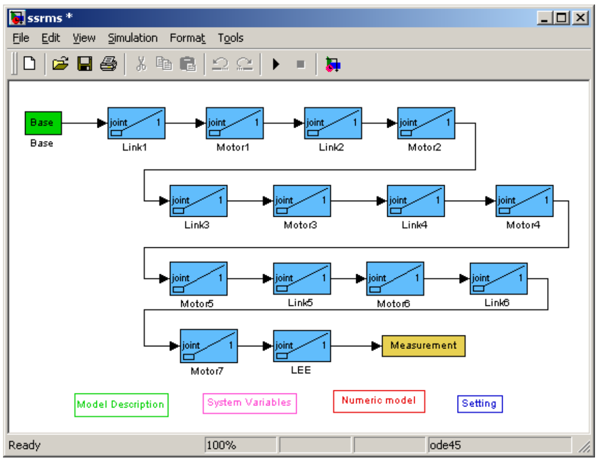

Figure 7: Graphical representation of the Symofros model.

with this SSRMS model and thus the effect of the elasticity in the last three joints will remain small. The elasticity of the four first joints was chosen to match the first four natural frequencies of the SSRMS in order to make sure that the dynamics behaviour of the tip of the arm is representative of the real arm.

The graphical representation of the SSRMS model in the Symofros Model Editor is shown in Figure 7 [3],[4]. It is composed of fourteen rigid bodies to model the seven moving links of the SSRMS and one Base to represent the non-moving LEE body. The Link1, Link2, Link3 and Link4 blocks contains elastic degree-of-freedom (dof) while the Link5, Link6 and Link7 blocks does not have any dof. The Motor1 to Motor7 blocks all contain a rigid dof. Therefore, the total number of dof for this model is eleven. The graphical representation of Figure 7 is then used by Symofros to generate the dynamics model using Maple and in a second step this model is saved in a C format for integration in the simulator.

4.2 Simulation Validation

The SSRMS model developed with Symofros and described above has been validated using the Manipulator Development and Simulation Facility (MDSF). MDSF is the official robotics simulation environment of the space station robotics system. It was developed by MD Robotics, and has been used extensively to support robotics operations at the International Space Station. MDSF includes the model of the SSRMS, which has been identified and validated

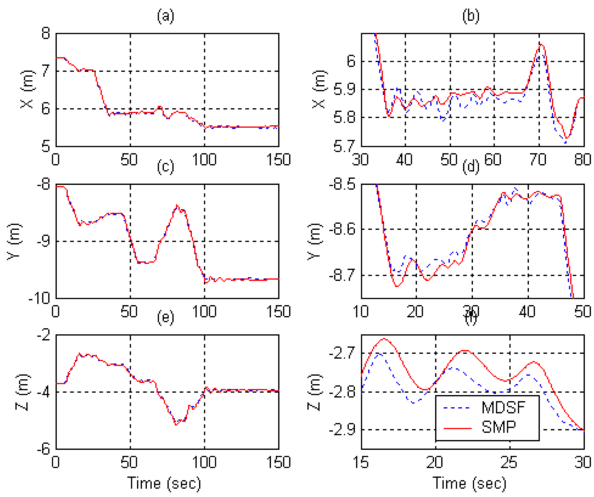

Figure 8: Cartesian tip position for MDSF and SMP.

against flight data and considered as a truth model of the system. This model includes all the relevant physical characteristics of the SSRMS: detailed joint models that account for friction, nonlinear stiffness, backlash and damping; flexibility of the two long booms, validated mass and inertial parameters; models of the controllers and algorithms. The simplified SSRMS model used in the SMP simulator has been compared to the MDSF model in terms of the fundamental structural frequencies, and based on simulating the motion for end effector trajectories. It was found that there is a good agreement between the first few fundamental frequencies. These govern the response of the robot for the majority of robotics operations. The agreement in performance was also confirmed based on comparing the simulated responses for end-effector trajectories. The trajectories were obtained by generating step input commands of the hand-controller from 0 to maximum vernier velocity for the 6 axes. These inputs represent worst-case scenarios for typical SMP simulation runs since usually the HC inputs looks more as ramp functions than step functions. The results for the X, Y and Z cartesian position are presented in Figure 8(a), (c) and (e), respectively. Figure 8(b), (d) and (f) are zooms of interesting portions of the outputs. It is obvious from these plots that the response of both simulators are very similar and that the manipulators are oscillating at about the same frequency. Finally, it is important to note that the validation was performed only for one configuration of the SSRMS. Since the simulator

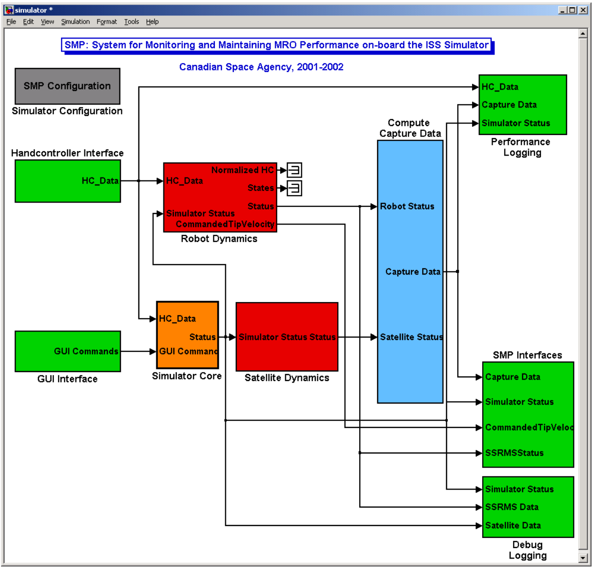

Figure 9: Simulator Architecture

always starts in this configuration and this is the only training scenario for SMP, this configuration was deemed sufficient for this validation of the simulator. However, we don't expect as good results for other configurations of the manipulator.

4.3 Simulator Engine

This module has been developed with Matlab/Simulink and simulates the motion of the SSRMS end-effector and the free-flyer (Figure 9). For the purpose of the free-flyer scenario, the astronauts will always use the Manual Augmented Mode (MAM). This is a cartesian velocity mode where the desired velocity signals come from the hand-controller inputs. The desired cartesian velocity is mapped in joint space using the Jacobian matrix of the manipulator. The arm having seven degrees of freedom, the Jacobian matrix is not square. In the first rate resolver mode, this matrix is inverted using the pseudo-inverse. In the second mode, the first joint is locked assuming a zero velocity command and the first column of the Jacobian is eliminated accordingly to obtain a square matrix, easily invertible. The third rate resolver mode consists in locking the second joint, again assuming a zero velocity command for that joint, and by inverting the Jacobian matrix with its second column removed. The output of the rate resolver, the desired joint velocity command, is tracked using a PI

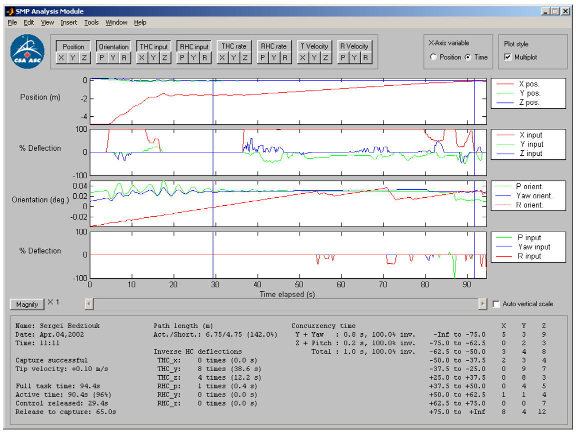

Figure 10: SMP Session Analysis Window

controller with collocated feedback of the actual joint velocity.

4.5 Analysis Module

The one of very important point of new concept and its implementation – multi level automated feedback – informs the MRO about his level of skills. This allows the MRO find out if he/she needs to continue training or not, and also to find what is the cause of a particularly success or mistake.

The Figure 10 shows that the session analysis provides information such as the hand-controller rates, the relative position and velocity between the end-effector and the free-flyer, and the capture status. Operational criteria and heuristics are used to provide a score, which allows the astronaut to have a good picture of his/her personal progress over time using a trend analysis. The astronaut can then determine if he/she needs more training or not.

5. Ground & On-Orbit Results

The skills required for Free Flyer capture have been found to be very close to those required for docking a Soyuz spacecraft (precise sensory-motor coordination and complicated cognitive spatial-logical transformations). Research conducted on docking skill degradation [1] has been used as the basis for the study of MRO performance.

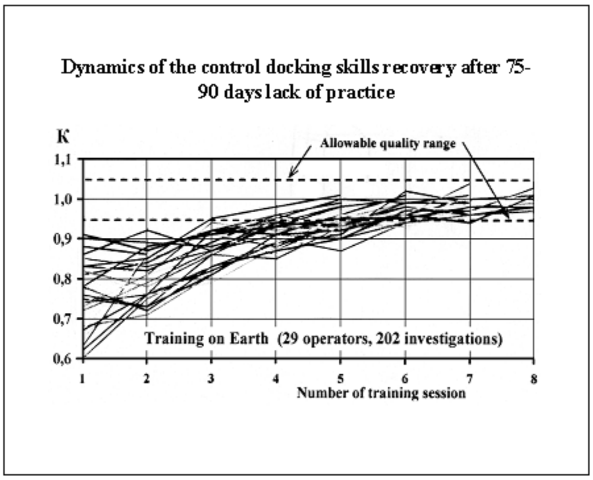

Figure 11. Soyuz docking skill degradation.

These studies show that a period of 25 to 30 days without docking practice results in 25 to 20 % decrease in performance and that operators require 3 to 4 training sessions to recover. Period of 75 to 90 days typically cause performance degradation of up to 40%, as shown in Figure 11. The same trend, even more pronounced according to preliminary results, is expected for Free Flyer capture skills. Figure 11 shows the significant differences between subjects (each line represent one subject performance over training sessions) as for skill degradation level and also for they recovery rate.

The evaluation of Free Flyer Capture performance uses the following measurements:

1. **Operation effectiveness parameters**: final positioning accuracy, trajectory deviation (relative to the ideal), oscillation amplitudes, rate, acceleration, time of operation; and
2. **Human motor output parameters**: handcontroller's deflection direction, proportions of outputs, smoothness of deflection, concurrent and multi axes deflections.

A few cosmonauts, already experts in Soyuz docking, were trained for Free Flyer Capture on SMP on the ground. They considered the Free Flyer task more complicated than docking, which is confirmed by experimental results shown in Figure 12.

Figure 12 shows that the subject (a professional

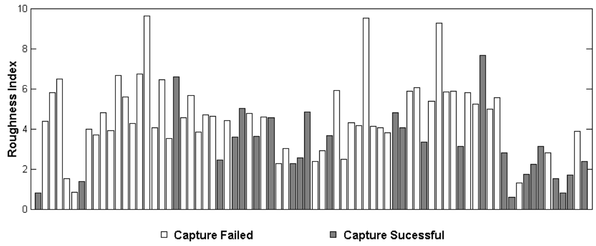

Figure 12. Free Flyer Capture skill acquisition dynamic.

cosmonaut) got to a certain level of performance stability (successful captures) after about 60 trials. While the average motion roughness decreased and success rate increased in the last few sessions, a few failed captures still occurred, showing that after 60 trials, the subject didn't reach yet a plateau in performance.

SMP having been launched to the ISS on February 2nd of this year, there are no on-orbit result available from SMP at this time.

6. Conclusion

In order to support the training of astronauts on board the ISS, CSA has developed the SMP project. In order to support training's scenarios, a simulator had to be developed.

Due to the complexity of the space robots, simulation tools were needed for appropriate modeling of these manipulators and their controllers. Based on previous work in the field of flexible robot modelling, SMP was developed. The use of an open architecture and of powerful development tools made the integration of a dynamic robot model and a 3D virtual environment into SMP possible.

Russian cosmonauts have used SMP officially and the first trainees have executed more than 75 simulation sessions. SMP was launched on February 2nd, 2003 on board the ISS. First on-orbit trials and sessions are planned in the middle of summer 2003.

References

[1] "Интегральная оценка и прогноз профессиональной надежности космонавтов в полете", Сальницкий В.П., Мясников В.И. С.В., Шевченко Л.Г. //Авиакосмическая и экологическая медицина –1999.- Т.33, №5.- С.16-22)

[2] R. L'Archevêque, Z. Joukakelian, P. Allard, "BORIS: A Simulator for Generic Robotic Training," in The 6th International Symposium on Artificial Intelligence and Robotics & Automation in Space: i-SAIRAS. Canadian Space Agency, St-Hubert, Canada, 22-26 May 2000.

[3] R. L'Archevêque, M. Doyon, J.-C. Piedbœuf, and Y. Gonthier, "SYMOFROS: Sofware architecture and real time issues," in DASIA 2000 – Data Systems in Aerospace, (Montreal, Canada), 22-26 May 2000.

[4] J.-C. Piedbœuf, "SYMOFROS: Symbolic modeling of flexible robots and simulation," in Proceedings of the AAS/AIAA Astrodynamics Conference, (Halifax, Nova-Scotia, Canada), pp. 949–968, August 14-17 1995. Paper AAS 95-359.