Remote Operation with Supervised Autonomy (ROSA)

Ross Gillett\textsuperscript{a}  Michael Greenspan\textsuperscript{b}  Leo Hartman\textsuperscript{c}  Erick Dupuis\textsuperscript{c}  Demetri Terzopoulos\textsuperscript{d}

\textsuperscript{a}MD Robotics Limited,  
9445 Airport Road, Brampton ON, Canada, L6S 4J3,  
rgillett@mdrobotics.ca  
\textsuperscript{b}National Research Council of Canada,  
Bldg M-50, Montreal Road, Ottawa ON, Canada, K1A 0R6,  
michael.greenspan@nrc.ca  
\textsuperscript{c}Canadian Space Agency,  
6767 route de l’aéroport, St-Hubert QC, Canada, J3Y 8Y9,  
leo.hartman@space.gc.ca, erick.dupuis@space.gc.ca  
\textsuperscript{d}University of Toronto,  
Dept of Computer Science, 10 King’s College Road, Toronto ON, Canada, M5S 3G4,  
dt@cs.toronto.edu

Keywords autonomous robotics, space robotics, behaviour-based robotics, cognitive modelling, satellite servicing

Abstract

The goal of the ROSA project is to develop a variable-autonomy architecture to support ground-based control of space-deployed robotic systems in dynamic workspace environments. This capability is aimed at unmanned satellite servicing systems that must rendezvous with, capture and service free flying objects. As data communication limitations force the system to perform free flyer capture autonomously, the ROSA project will implement a system employing a combined cognitive modelling and behaviour-based control approach that uses machine vision as the primary sensor.

1 Introduction

Robots are used extensively in manufacturing, where they excel at performing repetitive tasks. These tasks are exclusively performed in highly structured environments and are restricted to a small number of well-defined and rehearsed activities. Otherwise, autonomous robots still have not established a significant presence in other types of activities that are unstructured or performed in dynamic worksites. Wherever possible, humans still directly control manipulator systems.

The best examples of remotely controlled robot and manipulator systems used to perform non-repetitive and complex tasks are those used in assisting manned space flight operations or in space exploration. Space-deployed robots have been used for many years in manned space flight and in planetary exploration systems. In manned space flight, the Canadarm and Canadarm2 remain the best-known examples of directly controlled telemanipulator systems.

Both of these systems are designed for human-in-the-loop control; a human operator directly issues motion commands via handcontrollers with negligibly small latency between commands and response of the system. These systems rely on human intelligence, judgement, and motor skills to perform the desired operation. In the case of the Canadarm, the operator in the space shuttle can directly observe the operation through the shuttle windows. The Canadarm2 operator may be more removed from the robot and may be limited to camera views only to observe the system under his/her control, however the command/control and response latencies are still negligibly low.

Ground operation of space robotics introduces a number of challenges at the system level. These generally arise from the bandwidth and latency limitations of the data communication channel connecting the operator control station to the remote system. Additionally, these challenges include the problem of limited operator situational awareness that occurs whenever line of sight viewing is removed.
2 IIRO: Control From A Distance For Static Worksites

Previous work in the Interactive Intelligent Remote Operations (IIRO) project was aimed at developing techniques that can enable control of robots over extreme distances and overcome the associated data communications limitations [1]. Figure 1 contains a simplified diagram of the IIRO architecture.

The IIRO project addressed these challenges by using robot-mounted sensing to construct an accurate model of the desired portions of the robot equipment surroundings at the operator station. The robot system was then represented by a computer model correctly posed and registered within the modelled surroundings.

Controlling the virtual equipment within this world model, it was then possible to generate motion script data. Tools were provided to permit complex script construction, script editing and rehearsal, and to introduce discrete-event decision capabilities. Only after a script had been thoroughly rehearsed would it be uploaded for execution, at which time the operator could monitor the resulting operation on virtual displays and monitors.

This scenario introduces a new problem for a robot system – a non-static worksite. As already stated, the IIRO approach – to model the worksite for the purpose of planning the detailed motion prior to executing this motion – will not work since the worksite model will no longer be valid by the time a plan has been generated. Specifically, a target satellite prior to capture will not remain perfectly motionless relative to the servicing spacecraft.

If we consider the communication latency from ground to orbit, on the order of ten seconds, it becomes apparent that the robot system must respond to this worksite variability without the benefit of human judgement or direct human-in-the-loop control. That is to say the system must be capable of tracking the changes in location of the satellite and capturing the satellite autonomously. Once captured and rigidised,
the worksite is once again static and it is possible to again decide whether or not to employ high or low levels of autonomy.

4 The ROSA Variable Autonomy Architecture

The mixed needs of capturing a floating object, necessitating a high degree of autonomy, and subsequently performing servicing, involving a complex series of deterministic activities that may best be directed more closely by a human, points to a need for variable levels of autonomy.

The ROSA team is implementing an architecture that will allow operators to decide which portions of an operation may be autonomous, and which require detailed instructions generated by the operator, and mix these seamlessly within a single mission script. The lowest levels of autonomy are based on the IIRO system architecture already outlined. The higher levels of autonomy are implemented in two distinct approaches: an Inference Engine employing a hierarchical task network, and an Intelligent Robotic Mission Supervisor employing an autonomous control architecture. The autonomous control architecture comprises a behavioural subsystem and a cognitive subsystem that are being designed according to approaches developed at the University of Toronto [3,4,5].

A diagram of the complete ROSA architecture is shown in Figure 2. The operator is provided with the tools to generate and rehearse a series of mission scripts in the manner developed in the IIRO project. As in IIRO, these mission scripts can contain low-level commands or motion script segments as desired. This includes the capability to define logical branching and pre-emptive reflex actions based on sensory events. Additionally, the operator is able to enter higher level commands into the script. These high level commands may be Tasks or Goals. Tasks are commands targeted at an Inference Engine high level controller, and Goals are commands targeted at the Intelligent Robotic Mission Supervisor. Both of these controllers employ behaviours and artificial vision functions, as defined in the following subsections.

![Figure 2 The ROSA Architecture](image-url)

4.1 Vision Sensing

A system requires sensing of the external environment in order to interact with it dynamically and exhibit intelligent behaviour. Autonomous tracking, rendezvous and capture of a free flying object such as a satellite require some form of non-contact sensing. The ROSA architecture will rely primarily on machine vision techniques that employ two on-board stereo camera systems, one medium range and one short range [6]. Other sensors, such as monocular video and range sensors, are also possible within the concept of ROSA as the architecture does not preclude any sensory data source.

The medium range (1-20 m) and short range (< 1 m) vision sensing functions within ROSA are used to search for and identify the satellite pose and motion parameters and subsequently to search for and identify the pose and motion parameters of the docking feature (located on the satellite) respectively. The system has three modes of operation: Monitoring, Acquisition, and Tracking. In the Monitoring mode, which is invoked in the medium or possibly far range, the relative motion of satellite is estimated from either a monocular or a stereo video sequence. The motion parameters are estimated using structure from motion techniques that combine Projective Vision and Robust Statistics [7]. These methods allow correspondences between feature points extracted from successive video images to reliably determine the rigid motion of the satellite with respect to the sensor.

The Monitoring mode determines the relative motion between the sensor and the satellite. It does not, however, resolve the absolute position of the satellite within the motion sequence. In the Acquisition mode, the pose of the satellite is estimated from the acquired 3D data points returned from the stereo vision system. Full 6 degrees of freedom (6 dof) pose estimation is, in
general, a difficult problem. In this case the 3D point data is also relatively sparse, which can further complicate the problem. It is somewhat simplified, however, by the relatively easy segmentation of the satellite from the background. Also, specific attributes of the satellite global shape can be used to resolve some of the degrees of freedom. A Geometric Probing method [8] is currently being investigated to solve the satellite pose estimation.

Once the initial pose of the satellite is determined, the Tracking mode is then invoked and initialized with this estimate. During tracking, the satellite pose and motion are computed with high precision and update rate from the 3D point data provided by the stereo vision system. A fast variation of the Iterative Closest Point Algorithm [9] has been developed toward this end. The method incrementally minimizes the fitting error between the acquired data points, and a model of the surface that is being imaged. In laboratory tests, the method has been able to perform with a high degree of accuracy up to approximately 1Hz update rate, with roughly one thousand 3D points.

4.2 Behaviour Executor

The ROSA system employs reactive behaviours, forming form a behavioural subsystem, to guide the robot between obstacles to rendezvous and capture the target satellite. As such, higher functions or command inputs within the system must be capable of invoking and terminating behaviours. The ROSA system provides two higher functions capable of this. The first is an Inference Engine that invokes and terminates behaviours as determined by an operator-generated mission script in conjunction with prescribed discrete operational and/or sensory events. The second is an Intelligent Robotic Mission Supervisor that performs cognitive modelling and planning.

ROSA’s behavioural subsystem supports the basic functionalities necessary to rendezvous with and capture target satellites. It takes the form of a repertoire of programmed routines that enable the system to react adaptively to sensed external events. These routines are known as reactive “behaviours”. The behaviours, which directly couple perception to action, characteristically involve some sort of actuation of the robot; e.g., a thruster burn to modify position, attitude, speed, etc., a manipulation, say, extending an articulated arm to grab the satellite, or a sensing actuation to, say, change the point of view of the on-board camera. The repertoire of behaviours is arranged in a loose hierarchy, with low-level behaviours supporting mid-level behaviours, which in turn support higher-level behaviours.

Given current perceptual information about the external world and about the internal state of the robot, an action selection mechanism (Inference Engine or Intelligent Supervisor) is responsible for invoking the appropriate behaviours. Each behaviour is nominally designed to monitor sensory data and control the robot for a relatively short period of time, typically on the order of a few seconds. When an invoked behaviour has satisfied its specialized short-term task, a subsequent behaviour is invoked to carry the mission forward. If an active behaviour fails due to changing conditions in the dynamic environment, the action selection mechanism passes control to a different, more suitable behaviour. With an adequately rich behavioural repertoire in place, it becomes possible for the robot to deal with a sufficiently broad variety of possible events.

Thus the ROSA system progresses towards the eventual satisfaction of its mission by executing an appropriate sequence of behaviours. When the Inference Engine is employed, this sequence is determined by a logical relationship embedded in the Mission Script that is uploaded to the on-orbit system. Hence selection of behaviours can follow only a finite number of explicitly-scripted alternatives. When the Intelligent Supervisor is employed, however, the action sequence in the system is not determined in advance. Rather, it adapts to the changing situation. The autonomy and rudimentary intelligence afforded by the behavioural subsystem is akin to that observed in lower animals such as insects or fish. It constitutes a reactive substrate that supports the second major component of the ROSA autonomous control architecture, the cognitive subsystem.

4.3 Inference Engine

The ROSA architecture includes an Inference Engine as the first level of autonomous behaviour control. The Inference Engine accepts operator-generated mission scripts that can contain medium level commands, categorised as “Tasks”, and low level commands. It then infers from the Task commands a series of behaviours that are deemed necessary to accomplish the task. This decomposition is based on deterministic logical relationships that are defined and instilled into the system. It defines all autonomous behaviours that the robot can execute for any mission scenario. The implementation of the inference engine is done using a hierarchical task network of finite state machines.
The Inference Engine in turn issues commands to the Behaviour Executor to invoke or terminate behaviours or to perform lower level actions such as deterministic motion commands. Alternatively, the operator can explicitly include behaviours in a mission script and not rely on this inference function. The Inference Engine also monitors the system status and other sensory data to decide when to terminate a behaviour and which behaviour or series of behaviours to invoke next.

This decision-making capability is performed in a discrete-event manner using fixed logical relationships. These relationships may be explicitly contained within the operator-generated script or embedded in the hierarchical task network. Fault conditions such as failing to result in successful outcome within a given time limit will cause the Inference Engine take appropriate actions: it may, for example, terminate the script and await further command input from the ground-based operator. This represents a significantly higher level of autonomy in the system than the discrete-event controlled motion scripts employed in the IIRO project because this can invoke motion or higher level behaviours.

4.5 Intelligent Robotic Mission Supervisor

The autonomous control architecture being developed at the University of Toronto employs the suite of reactive behaviours described earlier and introduces two higher layers of control. First is a cognitive control layer that employs its own inference engine function but adds the ability to simulate, or 'think ahead' to achieve deliberative control. Above this lies an intelligent supervisor layer that controls the system at the highest level based on mission goals. The Intelligent Robotic Mission Supervisor is a separate function from the Inference Engine described earlier, and provides the highest level of autonomous behaviour control in the ROSA architecture.

The intelligent control abilities of the system can be bypassed, however, by issuing operator generated mission scripts directly to the action selection controller at the behavioral level. For example the operator can explicitly specify a sequence of behaviours in a mission script and not rely on the inference function to plan appropriate behaviour sequences. This achieves the variable autonomy identified as a need for the ROSA system

4.5.1 Cognitive Control Layer

The cognitive subsystem goes beyond simple reactive control to provide longer-term deliberative control. Analogous to the thinking abilities of higher animals (including humans), the cognitive subsystem is in principle capable of rudimentary contemplation of past, present, and future situations. The ability of the system to "think ahead" before deciding on the next course of action gives the ROSA system a substantial level of autonomy.

The cognitive subsystem explicitly represents a priori knowledge that the robot has. The knowledge base can take the form of an elementary set of facts about the robotic system, its environment and its task. A more elaborate knowledge base can also include an accurate physical simulator of the system and its environment. The available knowledge is exploited by the robot to predict the effect of its actions, to reason about its actions, and to plan action sequences.

Our approach is based on an artificial intelligence (AI) formalism known as the "situation calculus". The situation calculus explicitly represents "situations", which are snapshots of the state of the world, "actions" which produce changes in situations, and "fluents" which keep track of quantities in the world that can change over time. Knowledge is represented in terms of "axioms" within the logic formalism. An inference engine exploits this logic-based formalism to reason and plan actions. Perceptual sensing serves to reduce the robot's uncertainty about its world and is invoked whenever the uncertainty about the values of relevant fluents grows uncomfortably large.

4.5.2 Intelligent Supervisor Layer

The cognitive-level control enables the ROSA system to include an Intelligent Robotic Mission Supervisor (or "Intelligent Supervisor") that directs the system in terms of mission-specific goals. This supervisory control takes place at a higher level of abstraction than would be possible with just a purely reactive, behavioural level controller. To satisfy each goal, while monitoring the system status and performance, the inference engine at the cognitive level issues commands to the action selection controller at the behavioural level, which in turn initiates behaviours designed to perform specific lower level actions such as sensing, motion, and manipulation.
5 Operation Concept for On-Orbit Satellite Servicing Using ROSA

The ROSA project goal is to develop and demonstrate an architecture that will provide an operator with the ability to choose the level of autonomy with which to instruct the system to perform a mission, and to permit the operator to generate Mission Scripts that seamlessly incorporate mixtures of high and low level commands within a single mission. With this variability no single operation concept can fully define the use of the system. The team has selected a design reference mission from which to extract requirements.

The design reference mission consists of the rendezvous of an unmanned servicing vehicle with the target satellite system followed by tracking, capture, berthing and servicing, and finally release of the target satellite. Figure 4, 5, and 6 illustrate several phases of this mission. It is assumed that the Target Satellite system is designed for on-orbit servicing and as such is equipped with a docking feature with which the servicing manipulator has been designed to grasp the satellite, and a berthing fixture that includes a refuelling connection. The Target Satellite may have electronics implemented as orbit-replaceable units (ORUs) that will possess robotic interfaces compatible with tools that reside on the servicing satellite. It is also assumed that the target satellite and its docking feature are known and modelled with 3D computer models.

A Mission script can be generated any time prior to the mission, provided the 3D models and other pertinent data have been gathered for the satellite requiring servicing. Alternatively, the script can be generated or modified while the servicing spacecraft is on orbit if the required servicing of the target satellite is unknown or not fully established prior to rendezvous and inspection.

The Mission script will contain a mixture of Goals or Tasks and Low Level commands that control
deterministic events or motion sequences. This script is generated on the Ground based operator station (see Figure 2). The Visual Display and Kinematic simulator are used for explicit motion generation. Script segments are assembled in the Contextual Display and transformed into high level Goal/Task from which the final Mission Script is constructed.

In preparation for autonomous execution of the mission script, the spacecraft is configured by uploading the appropriate models and the script. The configuration of the system for a particular operation begins with loading the model of the target satellite, the satellite docking feature, and the docking feature target, as shown in the Figure [3]. These models are used by the vision system tracking functions as the manipulator approaches and finally docks with the docking feature.

During the execution of the script telemetry is returned to the ground station where the operator may observe the mission in near real-time. This will include all video data, manipulator motion telemetry data, and script execution status information indicating which commands or behaviours pertain to the motion telemetry. The ground operator then acts in a supervisory role: he may get prompted for confirmation at critical state transitions and he has the power to abort operations. However, he need not directly control the execution of the missions.

The above design reference mission will be performed with various environmental lighting challenges and anomalies to assess the performance and robustness of the various control approaches in controlled and repeatable conditions.

### 6 Demonstration System Hardware

The demonstration of the ROSA system architecture will be performed using a pair of Fanuc industrial robots in a laboratory installation called the Reusable Space Vehicle Payload Handling Simulator (RPHS) [10]. This facility will provide the relative motion capability needed to emulate the movements of the Servicing Manipulator and Target Satellite necessary to implement and evaluate the various behaviours and high level control approaches outlined for the project. The robots will be outfitted with a camera system and docking end effector prototype and a satellite mockup with docking feature to play the role of the servicing system and the target satellite respectively.

An important aspect of this laboratory is its ability to realistically emulate the lighting conditions that would be encountered in earth orbit - an essential element to recreate realistic challenges in vision system tracking.

### 7 Summary

The ROSA project will develop and demonstrate an architecture that will provide an operator with variable autonomy: the ability to choose the level of autonomy with which to instruct the system to perform a mission. At the highest level of autonomy the system operator is able to supervise the mission as it proceeds, and to intervene or confirm critical state transitions. Two different behaviour control approaches are planned for
the demonstration system: an Inference Engine that employs a hierarchical task network, and an Intelligent Supervisor that employs cognitive modelling and planning to complete the mission objectives. These two approaches will be compared in a series of controlled experiments based on a design reference mission.

The design reference mission consists of the rendezvous of a robot-equipped servicing vehicle with a target satellite system that requires repair or refuelling. This is followed by tracking, capture, berthing and servicing of the target satellite and finally release. This mission will be emulated using representative robot interface designs installed on a pair of industrial robots in a laboratory environment that attempts to realistically reproduce the lighting conditions in earth orbit. The ROSA architecture demonstration is scheduled to take place in March 2002.

Acknowledgements

The authors wish to thank Precarn Associates Incorporated for co-funding this collaborative research project. We also wish to express our thanks to Piotr Jasiobedzki, and Mike Brousseau for their valuable input to this paper.

References


