[SSC21-IX-03]

Advanced Technologies: 7 Degrees of Freedom Robotic Arm on Archinaut One

Kevin French
Redwire
8226 Philips Highway Suite 102, Jacksonville, FL 32256
kevin.french@madeinspace.us

ABSTRACT

In Space Manufacturing (ISM) combined with robotic In Space Assembly (ISA) enables autonomous construction of space structures including spin gravity habitation rings, precision synthetic aperture radar (SAR) structures, and trusses for kilometer scale solar sails. Archinaut One (AO), designated as OSAM-2 by NASA, will demonstrate ISM and ISA through the ability to 3D print and robotically manipulate meter scale structures in orbit all from an Evolved Expendable Launch Vehicle (EELV) Secondary Payload Adapter (ESPA) class satellite. This work focuses on challenges associated with autonomous robotic ISA and ISM operations. Limited computational resources, robot satellite coupled dynamics, contact dynamics, and short low bandwidth communication windows are all of major concern. Radiation hardened computational resources significantly lag terrestrial solutions often prohibiting the use of common robotic motion control systems. AO uses distributed computation and highly optimized controllers to accommodate the large computational cost of controlling the 7 degree of freedom robot arm. As a consequence of comparable masses, robot arm motion imparts significant satellite attitude disturbances. AO uses dynamics simulations with carefully pre-planned trajectories, and pre-planned interludes of active and inactive Attitude Control System (ACS) to avoid instability in the ACS, maintain communications and solar power. Contact is required for several ISA tasks, but small motion error can lead to rapid buildup of contact forces endangering critical systems. An admittance controller and strictly enforced speed limits maintain acceptable contact force bounds. AO's polar orbit and communication hardware limits communications to short low bandwidth windows. AO is endowed with sufficient autonomy required for faster than human-in-the-loop control and between communication windows. This paper discusses the approach toward robotic autonomy required for the AO mission. Modeling, simulations, hardware in the loop simulations, along with several representative ground based tests are outlined and prove the efficacy of these methods.

This paper will discuss the approach toward robotic autonomy required for the AO mission. Modeling and testing are outlined, as well as how the challenges of coupled dynamics, contact dynamics, and autonomy are addressed.

Introduction

Redwire's Archinaut One (AO) will demonstrate robotic On Orbit Servicing, Assembly and Manufacturing (OSAM) of meter long structures on orbit. AO's core technologies are additive manufacturing, robotic assembly, and in situ inspection. Ground based tests matured these technologies to technology-readiness-level (TRL) 6.[1] Now they will be demonstrated on orbit.

Two large beams will be 3D printed by the Extended Structure Additive Manufacturing Machine (ESAMM), as shown in figure 3. In situ inspection will be performed by a 7 seven degree of freedom (DoF) robot arm, designed by Motiv Space Systems fitted with a Redwire custom end effector. The end effector is complemented with grip force sensors and a camera. The robot arm's purpose is to reconfigure the ESAMM and manipulate the printed beams. This paper explains four challenges overcome by AO's robotic operations: coupled dynamics of the robot arm and the space vehicle, contact dynamics, limitations on communications, and computational limitations.

Coupled dynamics of the robot arm complicate space vehicle attitude control because the robot arm constitutes a significant fraction of the mass of the Evolved Expendable Launch Vehicle (EELV) Secondary Payload Adapter (ESPA) class space vehicle. When the arm moves, especially while grasping something, the entire space vehicle's attitude is significantly disturbed. This coupled dynamics may be too much for the Attitude Control System (ACS) to reject or the coupled dynamics may excited uncontrolled modes. Careful pre-planned trajectories, simulation, and ACS toggling prevent loss of atti-

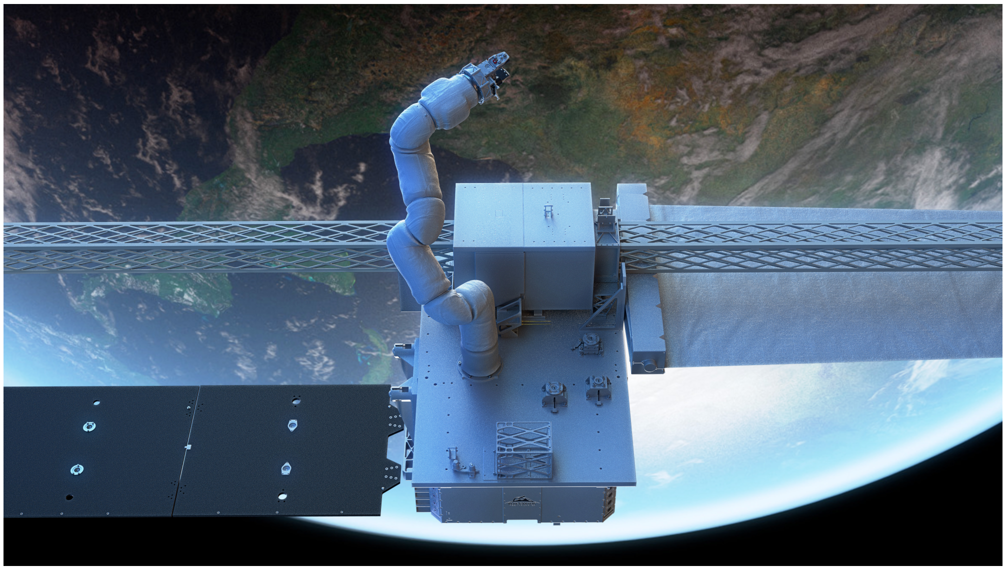

Figure 1: Archinaut One. Archinaut One orbits earth after completing its In Space Assembly and In Space Manufacturing tasks.

tude control.

In Space Assembly (ISA) tasks require that the robot arm make contact with various surfaces. The contact force must be limited to protect the space vehicle and sensitive components like the robot arm end effector's Force Torque Sensor (FTS). AO uses an admittance controller along with precision trajectories to minimize the risk of excessive forces.

AO will have short communication windows of between 3m to 15m that occur no more than 4 times per 90 min orbit. This is particularly challenging for an operation where the robot arm must reconfigure the ESAMM withing a maximum time window due to thermal constraints. Time critical robot arm operations must occur without a human in the loop, and when they are in the loop, a human must be able to provide complex actions in short time frames. Sufficient autonomy provides just enough intelligence for the robot arm operations, while adding minimal risk of failure. High level commands enable operators to perform the mission in the short communication time windows.

The need for radiation hardened components limits the compute resources that can be used. Control algorithms for the 7 DoF robot arm are computationally expensive, making it difficult to achieve sufficient update rates for safe operations. AO uses a slow soft processor realized on a radiation hardened FPGA along with a fast MIL-SPEC co-processor to meet the computational demands. Efficient versions of controls algorithms are implemented and optimized to meet the tight requirements to maintain safe robotic operations.

Robot Arm Controller Architecture

The Robot Arm Controller (RAC), depicted in Figure 2, allows the robot arm to track joint space and workspace motions and comply to forces distal to the FTS. There are four major components of the RAC; joint control, position control, compliance control, and trajectory generation.

Joint Control

Joint control refers to the independent control of each joint actuator of the robot arm. Motiv Space Systems provided Redwire with cascaded proportional integral derivative (PID) joint controllers. There are three PID loops; current, velocity, and position. The output of the position loop is the velocity set-point which in turn outputs the set-point for the current PID loop. Each PID has independent feedback provided by appropriate sensors. The primary benefits of this architecture is responsiveness and disturbance rejection.[2]

The controller may be operated to control joint position, velocity, or current. It is safest to operate the arm in position mode as, if a failure occurs, the arm will hold position, rather than moving and possibly colliding.

Feedforward terms can be provided to the controller for faster response times.

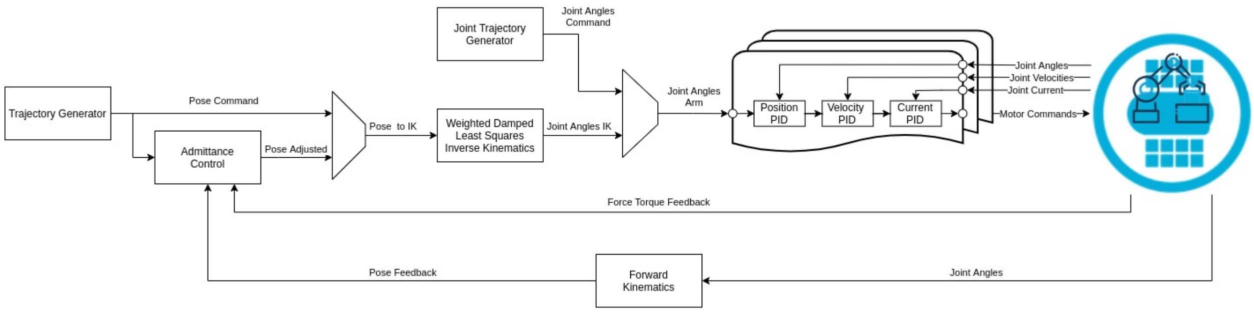

Figure 2: Robot Arm Controller. Seven cascaded position velocity current PID loops are controlled in parallel by the inverse kinematics outer loop. An optional admittance controller may be used to add virtual compliance to the robot arm.

Position Control

Position control moves the robot arm end effector to a desired pose (position and orientation). Inverse Kinematics (IK), provides position control by finding a set of joint states that achieve a desired pose.

The RAC uses a numerical IK, Weighted Damped Least Squares Inverse Kinematics (WDLSIK).[3] WDLSIK addresses the issue of numerical instability near singularities by adding a damping term to the optimization. The damping penalizes rapid changes in the output joint angles that occur near singularities. The weighting terms control both which joints will be used most and allows more or less importance to be given to particular degrees of freedom of the 6 DoF error. The RAC adds weight linearly from the first joint to the last joint such that the distal joints are used more than the proximal joints. This helps minimize disturbances to the space vehicle and power usage because the most distal joints move less mass. Weighting is also used to preferentially correct for position over orientation.

Compliance Control

Compliance is the ability for a material to undergo a deformation when a force is applied. Compliance can be modeled as a spring-mass-damper system. Compliance control uses sensors to measure properties of contact and actuate the robot arm to impart a desired virtual compliance with the desired spring-mass-damper properties at the contact.

There are two inversely proportional measurements of compliance, admittance and impedance. Admittance is a measure of how much motion occurs per unit of force. Impedance is a measure of how much force occurs per unit of motion.

Compliance control tracks a desired virtual compliance making the robot arm act as if it were a spring-mass-damper system. There are three major forms of compliance control for robot arms: admittance, impedance, and passive. Admittance controllers use FTS to measure applied force and torque and applies an offset velocity or position to achieve the desired compliance properties. Impedance control is the inverse of admittance, displacement is measured, and force is controlled to achieve the desired compliance properties. Passive control uses mechanical compliance, such as real spring dampers, to achieve compliance.

The RAC uses an admittance controller. Admittance control is less computationally intensive than impedance control which requires Inverse Dynamics (ID) and does not require joint friction models.[4] Admittance control also takes advantage of the RAC's existing high bandwidth position control loops. Admittance control suffers from poor performance when interacting with high impedance surfaces;[4] however, this disadvantage can be compensated for with slow motion, high damping and high bandwidth.

Trajectory Generation

Trajectory planners generate constrained position and velocity profiles for the robot arm to follow. Constraints may include joint limits, collision avoidance, or other user defined constraints. Trajectories may be in joint space or in the workspace.

Coupled Dynamics

Planning and Simulation

On satellites with large masses compared to the dynamic loads the attitude error induced by the dynamic loads can easily be handled simply by treating the error as noise that the Attitude Control System (ACS) rejects. Small satellites like AO do not have

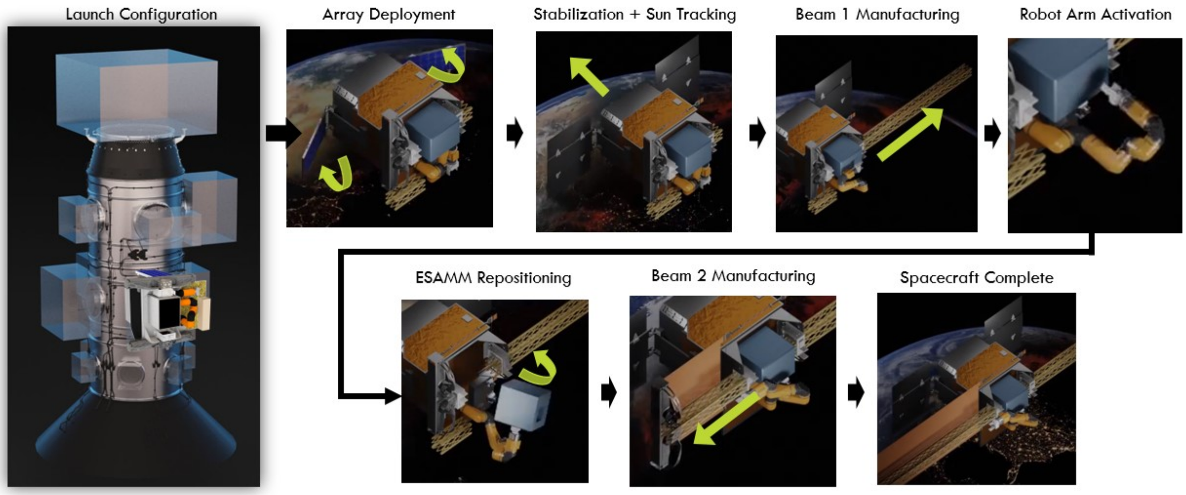

Figure 3: Mission Sequence. After achieving orbit and completing initial bus checkout, the first beam is printed. The robot arm then locks the beam in place and rotates the ESAMM. The second beam is then printed.

that luxury. Archinaut One's (AO) robot arm constitutes a significant fraction of the mass of the entire ESPA class space vehicle. When the arm moves, especially while grasping something, the entire space vehicle's attitude is significantly disturbed.

AO carefully pre-plans large motions with trajectory generators and simulates the disturbances to the space vehicle on the ground. A trajectory is hand crafted to minimizes the change in attitude of the space vehicle. For larger trajectories, the robot arm may need to move a large distance but subsequently returns to the original position. In these cyclical motions the ACS can be temporarily disabled. This way the ACS does not chase the space vehicle attitude back and forth, saves power, and avoids saturation of the reaction wheels in the ACS.

The nature of ISA requires masses to be moved around which disturbs the attitude of the space vehicle. For small masses this change is insignificant, but large masses must be considered separately. Again, AO uses pre-planned trajectories simulated on the ground in physics simulators and the ACS is disabled for cyclical motion.

To automate the construction of trajectories that minimize space vehicle disturbance the Reaction Null Space (RNS) is being considered.[5,6] The RNS is the subspace of motions that do not impart a disturbance on the space vehicle. Minimizing motion outside of the RNS will minimize the disturbances to the space vehicle. The RNS can be used in conjunction with a trajectory planner to design trajectories that minimize space vehicle disturbance. The RNS can also be used as part of the RAC's IK as a secondary objective to reduce space vehicle disturbance.

Natural Frequencies

Gains of the RAC must be carefully tuned to avoid overlap of the controller's bandwidth and the natural frequencies of the space vehicle. If the position control bandwidth contains a natural frequency of another part of the space vehicle including the arm itself, then a positive feedback loop can form. In such a case the vibrations would induce error in the arm position at the natural frequency which, since the frequency is within (or close to) the bandwidth of the position controller, the arm will correct for. The arm then moves at the natural frequency of the space vehicle, increasing the amplitude of vibration. The arm then applies larger efforts to compensate. The positive feedback loop grows until either the arm or the space vehicle reaches a fault state and enters a safe mode.

Avoiding exciting natural frequencies and entering into a positive feedback loop begins by measuring and/or calculating all natural frequencies from models. Then the bandwidth of the RAC is tuned to ensure the bandwidth is less than half the smallest natural frequency. Therefore, any disturbance to the robot arm at those frequencies will be attenuated due to a gain of less than -3 decibels.

Contact Dynamics

On the space vehicle most surfaces are very stiff, tens to hundreds of Newtons per millimeter. The stiffness of the robot arm is in the same region (stiffness varies as a function of the robot arm configuration). When the robot arm contacts these surfaces undesirably large forces build up quickly. Even small errors impart a rapidly increasing force on the robot arm and surface. In the best case scenario these forces cause a fault and safe the system; in the worst, mission critical sensitive components like the FTS are damaged beyond repair. All of AO's grasping and peg-in-hole operations run this risk.

Risk is mitigated by using the admittance controller in the RAC. Before any such operation the admittance controller is activated. The admittance controller reacts to the forces imparted at the robot arm end effector and moves the robot arm such that it mimics a spring damper system. The various sources of error that may cause interference and large forces are, therefore, rejected.

Unfortunately, simply using an admittance controller does not solve the contact dynamics issues. In fact, admittance controllers have poor stability in contact with stiff environments.[4] See the compliance controller section for reasoning behind using the admittance controller. Since the surfaces on AO are stiff, the stability and safety of contact operations must be evaluated. Figure 4 compares the maximum end effector speed with the effective stiffness to the minimum update rate required to prevent more than 30 N of force from building up before the RAC can react and reduce the force. Effective stiffness is the combined stiffness of the end effector and surface when in contact.

$$\mathrm{Hz} = \frac{n k_{\mathrm{eff}} v_{\mathrm{ee}}}{f_{\max}} \tag{1}$$

Figure 4 graphs equation 1 where Hz is the minimum update rate, n is the number of update cycles before forces begin to reduce, k_{eff} is the combined stiffness of surface and end effector, v_{ee} is the maximum linear velocity of the end effector normal to contact, and $f_{\max}$ is the maximum allowable force buildup. In Figure 4, $n = 2$ and a factor of safety is applied by multiplying the resulting update rate by 2.

AO's main robot processor is a radiation hardened RTG4 with a soft Risk-V cpu implemented in the FPGA fabric. In addition, a MIL-SPEC co-processor runs the computationally intensive portions of the RAC with its faster clock speed and floating point unit hardware. The RTG4 verifies the results output by the co-processor and monitors for unexpected outputs in case radiation causes a problem. This architecture allows for a 50 Hz update rate of the RAC, and thus, by figure 4, can safely move the end effector at 4 mm/s.

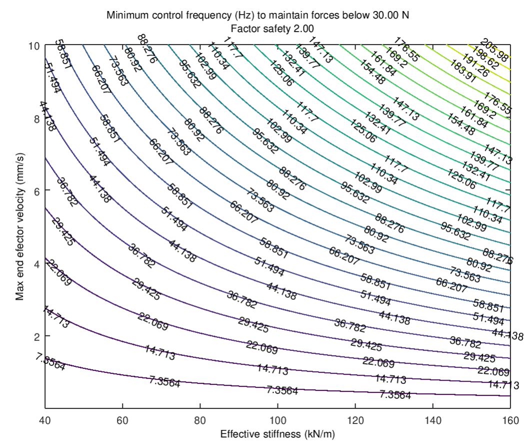

Figure 4: Minimum control frequency required to not exceed 30 N of unintentional force buildup. The level curves are calculated by assuming a perfectly elastic collision with the effective stiffness. It is assumed that the system can react and stop force build up within 2 update cycles of the control frequency. A factor of safety is used to ensure the frequency is sufficient.

Fiducial Markers

In addition to the admittance controller and the slow speeds, fiducial markers allow for precision alignment of the robot arm. The camera on the robot arm end effector takes a picture of one or more fiducial markers. Computer vision algorithms extracts the full 6 degree of freedom pose of the fiducial marker. Before launch precision measurements will be made so a known transformation exist between the fiducial marker and the target. Using these transforms the RAC determines a trajectory to the target and executes the corrective trajectory. This procedure reduces positional error and contact forces.

Autonomy

Autonomy is the ability to interpret sensory input and decide how to proceed without external control. Robotic autonomy methods come in many forms and are often mixed and matched. Common architectural components for autonomy include reactive, deliberative, learning, and cognitive components. Reactive components are mappings from sensory input that directly lead to action; for example, if a temperature sensor is too high shut off the heater. Deliberative components make plans, change reactive layers, achieve goals, etc. A STRIPS[7] planner may be used as a deliberative component of autonomy. Learning elements use experience to improve performance and domain knowledge. Reinforcement learning, Neural Networks, and Support Vector Machines are a popular choices for this application. Cognitive components are concerned with the acquisition representation and use of knowledge to pursue goals.[8] Soar[8] is an example of such a cognitive architecture.

AO orbits on a 90 minute sun synchronize polar Low Earth Orbit (LEO). Along the orbit there are various NASA Near Earth Network (NEN) ground stations; however, NEN is a shared resource and AO is only allocated small time windows. On a typical 90 minute orbit AO receives 2 to 4 up to 8 minute contact windows. All human in the loop operations are necessarily restricted to these short time windows. Human in the loop operations are, therefore, time constrained. AO adopts a methodology of sufficient autonomy to fill the voids between human contact and increase the efficiency of human in the loop operations, while minimizing risk.

Here sufficient autonomy is defined as a level 2 autonomous behaviors from Larry Young's work.[9] Level two autonomy is defined as "Ability to enable scripted contingency plans based upon predefined (well-posed) conditional logic conditions.".[9] The autonomy onboard AO is primarily behavioral based with the exception of the trajectory generators which are deliberative. While communications windows may be short, they are frequent. This means that a human operator can still act as a planner, learner, and cognitive partner for the space vehicle, saving computational load, algorithmic complexity, and risk.

The RAC monitors the state of the system for collisions, grasp misalignment, temperature, etc. If measurements exceed a predefined thresholds, a response is autonomously initiated. In general, the response is to engage the robot arm breaks and go to an idle state. However, this may not always be an option. During time critical operations, the RAC, under non-nominal conditions, must choose to continue the operation or if possible reverse the operation.

Modeling and Testing

Testing

AO has a comprehensive suite of ground tests built into the schedule. An end-to-end test will use engineering design units to test each phase of the mission. Likewise, every single mission requirement will be verified across multiple phases of AO's development. During development, several dummy components were made for testing. For example, volumetric mock-ups were made that weigh less than the actual part to allow the robot arm to pick the part up in Earth's gravity. These components allow thorough and comprehensive testing to be performed consistently.

Modeling

The robot arm operations are primarily simulated in CoppeliaSim.[10] CoppeliaSim offers a physics engine with many built in features such as Inverse Kinematics and Lua scripting. Rough estimates of space vehicle disturbances caused by robot arm motion can be generated. All of the robot arm operations are simulated. Telemetry can be fed into CoppeliaSim to provide situational awareness to the operators.

Conclusion

Redwire's Archinaut One (AO) is a first of kind mission demonstrating robotic On Orbit Servicing, Assembly and Manufacturing (OSAM). AO's robot arm, designed by Motiv Space Systems, is controlled by the Robot Arm Controller (RAC). Cascaded PID loops control the joints, Weighted Damped Least Squares Inverse Kinematics control the robot arm end effector position and an admittance controller gives the robot arm virtual compliance. The RAC must overcome several challenges such as, coupled dynamics, contact dynamics, and autonomy. Disturbances to the space vehicle from robot arm space vehicle coupled dynamics is managed with pre-planned trajectories and simulation. Natural frequencies are avoided by bandwidth limiting of the RAC. Contact dynamics can lead to rapid buildup of damaging forces. RAC moves slow and uses an admittance controller running on the faster MIL-SPEC coprocessor. Autonomy is limited to a behavioral level autonomy and trajectory planners. Humans plan and the RAC executes the plans. AO will be comprehensively tested through end-to-end test, requirements verification, and operations tests. Robotic operations are simulated with CoppeliaSim.

Acknowledgments

The Archinaut Technology Demonstration Project is a Space Tech Public-Private partnership with funding from NASA's Space Technology Mission Directorate (STMD) tipping point grant, contract NNM17AA02C. The author is grateful for technical support from Subject Matter Experts at NASA and Motiv Space Systems.

References

[1] Simon Patane, John Schomer, and Michael Snyder. Design Reference Missions for Archinaut: A Roadmap for In-Space Robotic Manufacturing and Assembly. In *2018 AIAA SPACE and Astronautics Forum and Exposition*, Orlando, FL, September 2018. American Institute of Aeronautics and Astronautics.

[2] Bela G. Liptak. *Instrument Engineers' Handbook, Volume Two: Process Control and Optimization*. CRC Press, October 2018. Google-Books-ID: TxKynbyaIAMC.

[3] S. Chiaverini, B. Siciliano, and O. Egeland. Review of the damped least-squares inverse kinematics with experiments on an industrial robot manipulator. *IEEE Transactions on Control Systems Technology*, 2(2):123–134, June 1994. Conference Name: IEEE Transactions on Control Systems Technology.

[4] Andrea Calanca, Riccardo Muradore, and Paolo Fiorini. A Review of Algorithms for Compliant Control of Stiff and Fixed-Compliance Robots. *IEEE/ASME Transactions on Mechatronics*, 21(2):613–624, April 2016. Conference Name: IEEE/ASME Transactions on Mechatronics.

[5] Miguel A. Torres and Steven Dubowsky. Minimizing spacecraft attitude disturbances in space manipulator systems. *Journal of Guidance, Control, and Dynamics*, 15(4):1010–1017, July 1992.

[6] Kazuya Yoshida, Dragomir N. Nenchev, and Masaru Uchiyama. Vibration suppression and zero reaction maneuvers of flexible space structure mounted manipulators. *Smart Materials and Structures*, 8(6):847–856, December 1999. Publisher: IOP Publishing.

[7] Richard E. Fikes and Nils J. Nilsson. Strips: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 2(3-4):189–208, December 1971.

[8] John E. Laird. *The Soar Cognitive Architecture*. MIT Press, August 2019. Google-Books-ID: cLofEAAAQBAJ.

[9] Larry Young. System Analysis Applied to Autonomy: Application to Human-Rated Lunar/-Mars Landers. In *Space 2006*, AIAA SPACE Forum. American Institute of Aeronautics and Astronautics, September 2006.

[10] E. Rohmer, S. P. N. Singh, and M. Freese. Coppeliasim (formerly v-rep): a versatile and scalable robot simulation framework. In *Proc. of The International Conference on Intelligent Robots and Systems (IROS)*, 2013. www.coppeliarobotics.com.