AIAA 04–22682

CONTROL OF SPACECRAFT SIMULATORS USING IMMERSIVE VIRTUAL ENVIRONMENTS

Michael A. Shoemaker
Virginia Polytechnic Institute and State University,
Blacksburg, VA 24061

The 2004 AIAA Region I-MA
Student Conference
April 16–18, 2004/Blacksburg, VA

CONTROL OF SPACECRAFT SIMULATORS USING IMMERSIVE VIRTUAL ENVIRONMENTS

Michael A. Shoemaker*
Virginia Polytechnic Institute and State University, Blacksburg, VA 24061

Tracking control laws are applied to spacecraft simulators to study the effect of immersive virtual environments on the ability of a human operator to perform manual attitude control tasks. The three degree of freedom (DOF) spacecraft simulator, located in the Virginia Tech Space Systems Simulation Laboratory (SSSL), consists of a test-bed mounted on a low friction, hemispherical air-bearing, allowing full rotational freedom about one axis and constrained rotation about two axes. Actuation of the simulator is achieved with three rigid, axisymmetric wheels controlled by axial torques and six cold-gas thrusters. Solid-state rate gyros are used to estimate the attitude of the spacecraft simulator. The human operator controls the simulation while viewing a visual interface from within the Virginia Tech Cave Automatic Virtual Environment (VT-CAVE). The operator uses the motion of a six DOF tracked input device to specify the torque level to apply to a reference body, in this case an ideal rigid body making up the spacecraft simulator. Modified Rodriquez Parameters (MRPs) are used to describe the attitude kinematics. The nonlinear tracking control law computes the axial motor torques needed to track the motion of the reference body with the spacecraft simulator, using the angular velocity error and MRP error. The overall effectiveness of the system is evaluated and recommendations given for future improvements.

Introduction

As space missions grow in scope and complexity, spacecraft attitude control requirements become increasingly important. The accuracy needed for typical spacecraft attitude determination and control tasks is often a fraction of a degree.[1] In addition, spacecraft operating in orbit have constraints placed on their available communications load, as well as time delays due to the distance from Earth. These are some of the reasons why unmanned spacecraft, such as communications satellites or interplanetary probes, are usually automatically controlled by onboard computers.

Remote control of spacecraft using "tele-presence" is one alternative to fully autonomous vehicles. Telepresence is "a form of electronically mediated presence providing high-fidelity remote control by projecting natural human capabilities to distant work sites".[2] Such systems usually employ a combination of visual and haptic interfaces, such as stereo vision and force feedback, to give the operator the sensation of being present at a remote location.

The focus of this paper is the use of an immersive virtual environment (VE) to allow a human operator to overcome the usual drawbacks associated with manual control of spacecraft attitude. The overall goal is to develop a system utilizing both the spacecraft simulators located in the Virginia Tech Space Systems Simulation Laboratory (SSSL) and the Virginia Tech Cave Automatic Virtual Environment (VT-CAVE). This paper describes the groundwork currently being developed, leading up to fully operational system with which to test different control and visualization methods.

First, we present the results of a literature review on innovative uses of visual displays related to space missions and remotely operated vehicles, as well as past studies on manual control input devices. Second, we give an overview of the SSSL spacecraft simulators and the VT-CAVE. Next, we describe a tracking control law that is being investigated for use in controlling the attitude of one of the spacecraft simulators. Emphasis is given to the issues associated with translating the human operator's desired control input from the CAVE into usable control variables related to this specific controller. We then describe computer simulation results of the controller, and future implementations on the hardware simulators.

Literature Review

Both visual displays and manual control methods have been the topic of a number of papers, dealing with varying fields such as robotics, undersea vehicles, and space missions. Much of the past research has been from a human factors perspective, with questions of the form: given different display types or input methods, how well do human subjects perform a specified

*Undergraduate Student, Department of Aerospace and Ocean Engineering, Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061, Student Member AIAA.

control task? The evaluation criteria are often psychological in nature, which can be difficult to extrapolate to other systems. However, the following examples are useful in placing this paper in context.

Visual Displays

As mentioned in the introduction, there is certainly a strong case for the use of automatic spacecraft control in some missions. Conversely, some examples in the literature where manual control is preferred involve maneuvering a spacecraft in close relation to another object in orbit, such as for docking, rendezvous, rescue, and repair[3] While these tasks typically concern manned spacecraft, the control issues could also be applied to a tele-presence remote control system. Difficulties arise during these so-called "proximity operations" (PROX-OPS) due to the nature of orbital mechanics; trajectories viewed between two moving bodies result in counterintuitive motion with a lack of stable reference points. Also, numerous burns are required to complete a single task: departing, maneuvering, and braking. The operator must be aware of various safety constraints: clearance from structures, allowable approach velocities, fuel consumption, and thruster plume impingement on sensitive equipment.[4] Such issues would be of concern for attitude control as well as orbit maneuver, especially in the case of immobile thruster nozzles. PROX-OPS are usually planned well in advance, but emergency situations might arise, dictating the need for a human operator to take command.[5]

There are other examples besides spacecraft in orbit where human operators need an increased understanding of the environment in which a remote vehicle is operating. One such case was the Telepresence-controlled Remotely Operated underwater Vehicle (TROV).[6] This system consisted of an undersea vehicle piloted remotely to explore the marine environment beneath Antarctic ice. The motivation was to test future space systems and their control interfaces in environments and situations analogous to those which they might encounter during operation. One method of remotely controlling the TROV was accomplished with the Virtual Reality Vehicle Interface (VEVI), in which an operator located at NASA Ames Research Center in California controlled the actual TROV from within the graphical simulation (see Figure 1). The undersea terrain models in VEVI were generated using real sensor data from the TROV, and the operator viewed the simulation using either a Head-Mounted Display (HMD) or a stereo monitor. The results gathered from this system showed that the immersive VE allowed the operator to achieve a level of spatial orientation and understanding despite little or no foreknowledge of the operating environment.

Another example of an unconventional display system used to control space related hardware is the Mars Pathfinder mission.[7] The martian lander used a high resolution narrow field of view stereo-camera called the Imager for Mars Pathfinder to send images of the landing site back to Earth. A 3-D terrain map was generated by comparing the stereo images, and the data feed into a computer generated graphical environment. The controllers could then navigate in this VE and specify science targets or movement commands to the rover (see Figure 2). Like the TROV, this system allowed the controllers increased situational awareness despite the extreme distances and unknown environment. Another noteworthy result was that even though the rover was not directly controlled, the interface allowed high-level commands to be issued to the rover with confidence while the onboard autonomous control system managed the low-level control tasks.

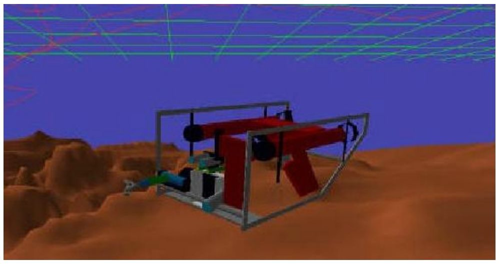

Fig. 1 Screenshot of virtual realty interface used with the TROV underwater vehicle.[6]

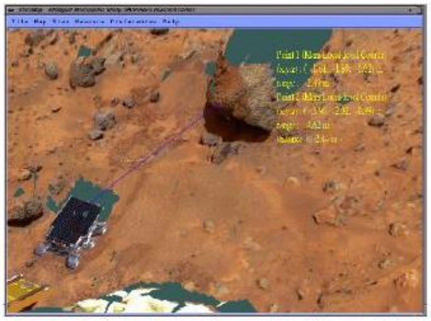

Fig. 2 Screenshot of virtual realty interface used with Mars Pathfinder mission.[7]

As a final example, researchers at Caltech developed an interactive trajectory planning tool to design orbits for the Terrestrial Planet Finder mission.[8] This astronomy mission called for transfer orbits from Earth to the L2 Lagrange point, where a formation of 5 satellites would enter a halo orbit and form an infrared interferometer. In order to better visualize and design these complex transfer orbits and satellite formations, a semi-immersive VE was created for use on a Responsive Workbench (see Figure 3). The designer views a 3-D representation of a region of low-energy transfer orbits, and selects different orbits using a tracked input device. Using this system allowed better understanding of the complex geometry associated with both the transfer orbit and the formation flying. Additionally, this tool was useful for conveying these unfamiliar orbits to others in industry.

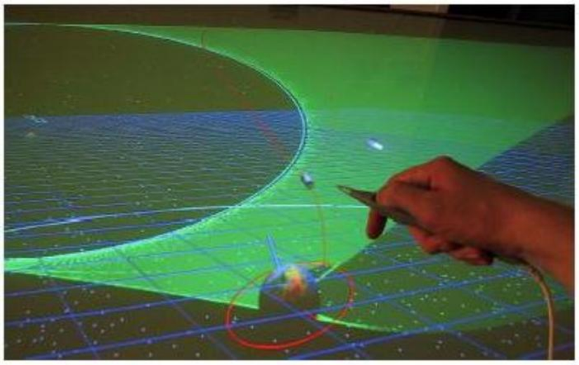

Fig. 3 Screenshot of semi-immersive trajectory planning tool.[8]

Control Input Methods

A recurring issue with manual spacecraft control is the selection of a proper input variable. The Lunar Module (LM) from the Apollo program, perhaps the most well documented manually controlled spacecraft from the 1960s and 1970s, was used extensively to study this problem. In one LM simulator system, the pilot used the flight stick to command the angular rates of the craft in an open-loop control system, where the deflection of the controller caused direct actuation of the attitude thrusters and a corresponding angular acceleration.[9]

In a separate system to simulate the control of a remote maneuvering craft for a satellite inspection mission, comparisons were done between controlling thrusters by issuing rate or acceleration commands with a hand controller, after which it was found that better performance was achieved through rate commands.[10]

Besides the type of control input, issues associated with translating the input into thruster actuation are often discussed in the literature. The fixed-base LM simulator system mentioned above tested two different methods: proportional thruster control and on-off thruster control.[9] Under the proportional controller, the thrust level was dependent on the error between the actual and commanded thrust levels. The commanded thrust level was determined by the amount of deflection in the flight stick, which was also evaluated under different levels of deadband. For the on-off controller, full thrust level was used at all times, resulting in overall fuel conservation but preventing delicate small angle maneuvers.

A later system developed for the LM also used rate commands, but did so using a combination of manual and automatic pilot control.[11] Namely, the descent portion of the lunar landing mission was divided into phases, with the autopilot controlling rate of descent and attitude control at higher altitudes and the human pilot controlling attitude and landing site selection at the final stage before touchdown. The "manual" phase was not entirely without computer assistance; the system employed a combination of human and computer control called "rate command/attitude hold." Here, the pilot issued rate commands with the flight stick to control the attitude directly, but when the flight stick was returned to the neutral position, the autopilot attempted to hold the current attitude to prevent drift. These improvements, which were practiced in simulators and flight-tested on Apollo 9, resulted in less fuel usage and better pilot ratings.

Lessons Learned

While these examples represent only some VE applications related to spacecraft control, they highlight important features and benefits. Clearly a well-designed VE can aid the operator in understanding a potentially complex control problem. Conversely, even a stunning graphical interface is useless if the control input method is difficult to use. Rather than use a purely manual control system, we investigate the use use of an automatic control system with direct human input. Regardless of the specific application, "the most successful visual interface will be the one that maximizes the operators ability to apply his or her superior perceptual and judgement abilities to the control problem at hand."[12]

Spacecraft Simulators

The SSSL consists of two spacecraft simulators making up the Distributed Spacecraft Attitude Control System Simulator (DSACSS).[13] These spacecraft simulators represent experimental test-beds with which to develop and test spacecraft attitude determination and control algorithms using hardware and software similar to that found on actual spacecraft. The presence of two simulators is especially useful for modelling spacecraft formation flying and distributed control laws.

The two simulators, "Whorl-I" and "Whorl-II", are shown in Figures 4 and 5, respectively. The main component of each is a low-friction air-bearing manufactured by Space Electronics, Inc. Whorl-I uses a "tabletop" configuration, with a main plate attached on top of a hemispherical air-bearing allowing full rotational freedom about one axis and ±5° rotation about the two table axes. Whorl-II uses a "dumbbell" configuration with two end plates, allowing full rotational freedom about two axes and ±30° rotation about the transverse axis. The air-bearing pedestal base on each prohibits any translational motion. The maximum payload capacity of each is 300 lb.

Both simulators use mostly the same hardware and software, but Whorl-I is chosen for the current study because it is further along in development. Whorl-I

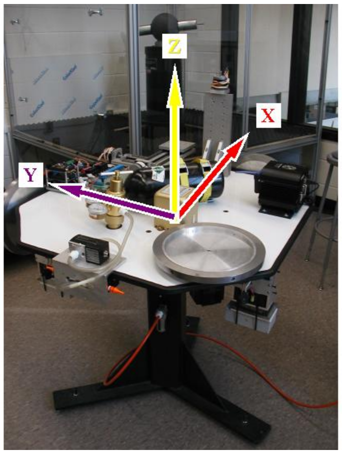

Fig. 4 Photo of the SSSL Whorl-I spacecraft simulator, with reference frame annotated.

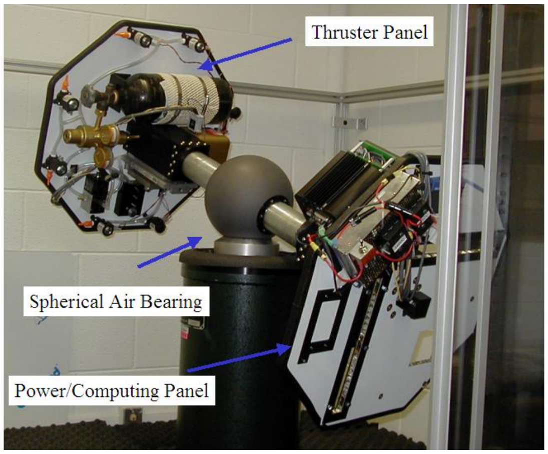

Fig. 5 Photo of the SSSL Whorl-II spacecraft simulator highlighting different components.

is capable of using 3 momentum wheels, 6 cold-gas thrusters, and a control moment gyro for attitude control. The momentum wheels consist of custom-made aluminum and steel flywheels attached to Animatics SM3430D SmartMotors. The thrusters are Evolutionary Concepts 654-2207 solenoid valves, using compressed nitrogen. Attitude determination is accomplished with 3 solid-state rate-gyros and 3 linear accelerometers housed in a Systron Donner Inertial Division MotionPakII. Future implementations of Whorl-I will include absolute attitude sensors as well. Linear actuators with movable masses can be used to balance the simulator and adjust the center of mass.

Whorl-I is equipped with a PC/104 form-factor computer as the onboard flight computer. This computer has a 32-bit, 133 MHz processor with 64 MB of RAM, and a 288 MB “Disk-on-chip” architecture, thus avoiding the spinning parts associated with typical hard drives. Linux is used as the operating system, and communicates through a 11 Mbps IEEE 802.11b wireless network, or W-LAN, with the other spacecraft simulator and a desktop running Linux. Besides the wireless network, the entire spacecraft simulator is wireless, using rechargeable batteries as the power supply.

VT-CAVE

The VT-CAVE is part of the the University Visualization and Animation Group (UVAG),[14] and is one of a growing number of CAVEs across the country. The VT-CAVE consists of 3 back-projected walls and a top-down projected floor, making up a partial cube approximately 10 feet on each side. Each screen uses an Electrohome Marquis 8000 projector to display full-color (1024x768 stereo) images at 96 Hz. Around the top of the CAVE are located a series of Intersense brand ultrasonic transmitters that work with small receivers to detect the position of objects in the CAVE. The objects currently capable of being tracked include a “wand”, or 6-DOF input device, and a 6-DOF pair of LCD shuttered glasses.

The immersive 3-D effect of the CAVE is created because the graphical images are displayed in stereo, or “double-vision”. When the user stands in the CAVE and wears the LCD glasses, the shutters flash at the same rate as the stereo image, and each eye sees only the correct image, thus producing the stereo effect. Because the glasses are tracked, the user can move anywhere within the 10 ft by 10 ft area and always have the correct perspective. The fact that the CAVE encloses such a large area compared to a standard monitor allows large environments to be visualized, or smaller objects to be manipulated and viewed from any angle.

Figure 6 shows a screenshot of a 3-D model of Whorl-I loaded into the CAVE. The current simulation places the model of Whorl-I in the center of the CAVE, with about the correct proportions as the actual Whorl-I spacecraft simulator. During the simulation, the user stands next to this “virtual” Whorl-I and uses the wand to issue input commands. The model is animated using a “3-1-2” Euler angle sequence, or heading-pitch-roll combination of rotations.[15]

Figure 7 shows a screenshot of what a fully operational system might look like, showing the virtual Whorl-I in orbit about the Earth. Such a system would be useful for simulating PROX-OPS scenarios, where the user takes advantage of the increased spatial awareness and sense of immersion provided by the CAVE. The present study is only considering the attitude control of a single spacecraft in a simple environment.

System Model

We make some assumptions in order to apply spacecraft dynamics equations to Whorl-I, which is subject to different dynamics on the surface of the Earth. First, Whorl-I is modelled as a perfectly rigid body. The main plate mounted on the air-bearing is sufficiently rigid such that this will not lead to major

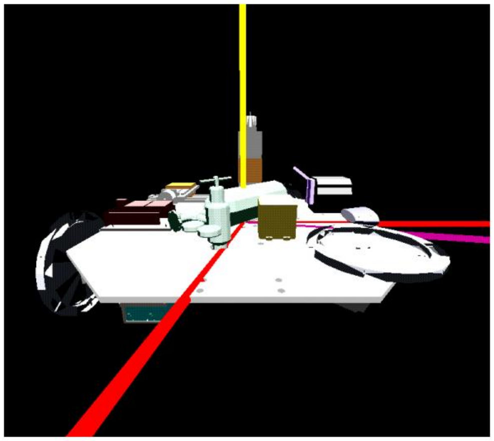

Fig. 6 Screenshot of Whorl-I model loaded into VT-CAVE graphical environment.

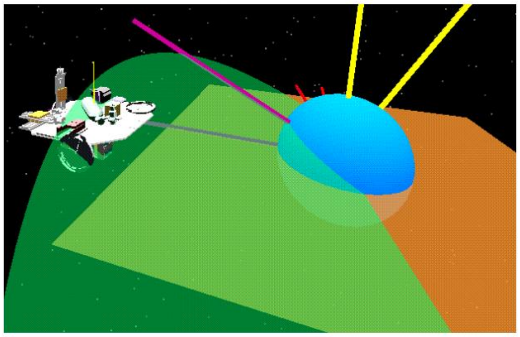

Fig. 7 Screenshot of sample orbital environment with Whorl-I model, reference frames, and simplified Earth.

errors for the present study. We also neglect any environmental forces that might act to perturb the simulator motion, such as friction from the air-bearing, laboratory air currents, and gravitational effects. Friction from the air-bearing is significantly low, and air currents from the room are troublesome but can be reduced to some extent.

Gravitational effects are neglected by assuming that the center of mass, center of gravity (cg), and center of rotation of Whorl-I are all coincident at the center of the air-bearing. This constraint is the most difficult to meet, since the simulator must be very carefully balanced. The linear actuators will be capable of doing this automatically once operational. Such a system would be necessary for any robust controller to correct for mass changes, especially while using the cold-gas thrusters. In the present study, we assume that the mass of Whorl-I remains constant, since any thruster firings would be short in duration.

The remainder of the system model and control laws are based on a nonlinear tracking controller found in the literature.[16] We let Whorl-I be a rigid spacecraft $\mathcal{B}$ with N rigid, axisymmetric wheels. We take $\mathcal{F}_B$ to be the body-fixed reference frame attached to $\mathcal{B}$, and specify the inertial frame $\mathcal{F}_N$ as having constant orientation with respect to the lab. Note that this is not a true inertial frame, but we can use it as such if we neglect accelerations acting on $\mathcal{F}_B$ and $\mathcal{F}_N$ from the rotation of the Earth.

We also define a reference spacecraft $\mathcal{R}$, with identical mass properties as $\mathcal{B}$. The controller is designed such that $\mathcal{F}_B$ will asymptotically track $\mathcal{F}_R$, the reference frame attached to $\mathcal{R}$.

Dynamics

We define $\mathbf{I}$ as the moment of inertia of Whorl-I, which includes the momentum wheels and thrusters. The axial moments of inertia of the wheels are expressed as

$$\mathbf{I}_s = \text{diag}\left[\mathbf{I}_{s1}, \cdots, \mathbf{I}_{sN}\right] \tag{1}$$

where $\mathbf{I}_{si}, i = 1, 2, \cdots, N$ denotes the axial moments of inertia of the wheels.

The dynamics of this system are described as[16]

$$\dot{\mathbf{h}}_B = \mathbf{h}_B^\times \mathbf{J}^{-1}(\mathbf{h}_B - \mathbf{A}\mathbf{h}_a) + \mathbf{g}_e \tag{2}$$

$$\dot{\mathbf{h}}_a = \mathbf{g}_a \tag{3}$$

where $\mathbf{h}_B$ is the angular momentum vector of the system in $\mathcal{F}_B$, defined as

$$\mathbf{h}_B = \mathbf{I}\boldsymbol{\omega}_B + \mathbf{A}\mathbf{I}_s\boldsymbol{\omega}_s \tag{4}$$

and $\mathbf{h}_a$ is the $N \times 1$ matrix containing the axial momenta of the N-wheels. Also in these equations, $\mathbf{g}_e$ is the 3×1 matrix of the external torques generated by the thrusters, and $\mathbf{g}_a$ is the $N \times 1$ matrix of the torques applied by the individual motors to the wheels. The matrix $\mathbf{A}$ is a $3 \times N$ matrix consisting of the axial unit vectors of the wheels, and describes their orientation in the system. The matrix $\mathbf{J}$ is defined as

$$\mathbf{J} = \mathbf{I} - \mathbf{A}\mathbf{I}_s\mathbf{A}^T \tag{5}$$

making $\mathbf{J}$ a positive definite matrix with "inertia-like" units. The angular velocity vector of $\mathcal{F}_B$ can be obtained from equations (4) and (5) as

$$\boldsymbol{\omega}_B = \mathbf{J}^{-1}\left(\mathbf{h}_B - \mathbf{A}\mathbf{h}_a\right) \tag{6}$$

Lastly, the axial angular momenta of the wheels is given by

$$\mathbf{h}_a = \mathbf{I}_s\mathbf{A}^T\boldsymbol{\omega}_B + \mathbf{I}_s\boldsymbol{\omega}_s \tag{7}$$

where $\boldsymbol{\omega}_s$ is the $N \times 1$ vector containing the axial angular velocities of the wheels with respect to the body.

Kinematics

For this control law, Modified Rodrigues Parameters (MRPs) are used to represent the attitude of the spacecraft. The MRPs are defined as

$$\boldsymbol{\sigma} = \hat{\mathbf{e}}\tan(\Phi/4) \tag{8}$$

where $\hat{\mathbf{e}}$ is the unit vector along the Euler axis, and Φ is the Euler angle. These components are sometimes called the Euler principal axis and Euler principal rotation angle. MRPs have a kinematic singularity at $\Phi = 2\pi$, which must be avoided. We can protect against this singularity in the future by incorporating other attitude representations, such as quaternions, into our simulation. The current simulations will only consider small rotations.

The kinematics differential equation using MRPs can be written as[15]

$$\dot{\boldsymbol{\sigma}} = G(\boldsymbol{\sigma})\boldsymbol{\omega} \tag{9}$$

where

$$G(\boldsymbol{\sigma}) = \frac{1}{2}\left(I_3 + \boldsymbol{\sigma}^\times + \boldsymbol{\sigma}\boldsymbol{\sigma}^T - \frac{1+\boldsymbol{\sigma}^T\boldsymbol{\sigma}}{2} I_3\right) \tag{10}$$

In equation (10), I_3 is the 3×3 identity matrix.

These definitions for the kinematics are used to describe the orientation of both the body frame and the reference frame. The equation for $\mathcal{F}_B$, or the "actual spacecraft" (Whorl-I), is

$$\dot{\boldsymbol{\sigma}}_B = G(\boldsymbol{\sigma}_B)\boldsymbol{\omega}_B \tag{11}$$

and the equation for $\mathcal{F}_R$, or the "virtual spacecraft", is

$$\dot{\boldsymbol{\sigma}}_R = G(\boldsymbol{\sigma}_R)\boldsymbol{\omega}_{\mathbf{R}} \tag{12}$$

When speaking of the reference frame motion, some dynamics must be generated in order to produce that reference motion. Here, the virtual spacecraft is assumed to have the same mass and momenta properties as the actual spacecraft, with non-spinning momentum wheels. Then, the reference motion dynamics are given by

$$\dot{\mathbf{h}}_R = \mathbf{h}_R^\times \mathbf{I}^{-1}\mathbf{h}_R + \mathbf{g}_R \tag{13}$$

$$\mathbf{h}_{\mathbf{R}} = \mathbf{I}\boldsymbol{\omega}_{\mathbf{R}} \tag{14}$$

Thus, if the nominal control toque $\mathbf{g}_R$ in equation (13) is applied to the actual spacecraft with the same initial conditions and non-spinning wheels, the actual spacecraft would follow the desired attitude motion.

Next, the tracking error of the angular velocity expressed in $\mathcal{F}_B$ is

$$\delta\boldsymbol{\omega} = \boldsymbol{\omega}_B - C_R^B(\delta\boldsymbol{\sigma})\boldsymbol{\omega}_R \tag{15}$$

where $C_R^B(\delta\boldsymbol{\sigma})$ is the rotation matrix from $\mathcal{F}_R$ to $\mathcal{F}_B$. The term $\delta\boldsymbol{\sigma}$, representing the kinematics error between these two frames, is defined by

$$C_R^B(\delta\boldsymbol{\sigma}) = C_N^B(\boldsymbol{\sigma}_B)C_N^R(\boldsymbol{\sigma}_R)^T \tag{16}$$

The differential equation for the error kinematics, similar to the other differential equations (11) and (12), is written as

$$\delta\dot{\boldsymbol{\sigma}} = G(\delta\boldsymbol{\sigma})\delta\boldsymbol{\omega} \tag{17}$$

From the dynamic equation for the virtual body, equation (13), we know that

$$\dot{\boldsymbol{\omega}}_R = \mathbf{I}^{-1}\mathbf{h}_R^\times \mathbf{I}^{-1}\mathbf{h}_R + \mathbf{I}^{-1}\mathbf{g}_R \tag{18}$$

and following a more thorough derivation in the literature,[16] it can be shown that the error dynamics are given as

$$\begin{aligned} \delta\dot{\mathbf{h}} &= \mathbf{h}_B^\times \mathbf{J}^{-1}(\mathbf{h}_B - \mathbf{A}\mathbf{h}_a) + \dots \\ &\dots + \mathbf{g}_e - \mathbf{J}\boldsymbol{\omega}_B^\times \delta\boldsymbol{\omega} - \mathbf{J}C_R^B(\delta\boldsymbol{\omega})\dot{\boldsymbol{\omega}}_R \end{aligned} \tag{19}$$

Equation (19) also makes use of a proof found in the literature.[16]

Controller

The controller described below is used to track the reference motion of the virtual spacecraft with the actual spacecraft. The objective of the control law is to asymptotically drive the tracking error terms $\delta\boldsymbol{\sigma}$ and $\delta\boldsymbol{\omega}$ to zero. The controller makes use of the candidate Liapunov function

$$V = \frac{1}{2}\delta\boldsymbol{\omega}^T\mathbf{K}\delta\boldsymbol{\omega} + 2k_2\ln(1 + \delta\boldsymbol{\sigma}^T\delta\boldsymbol{\sigma}) \tag{20}$$

which is positive definite and radially unbounded, with $\mathbf{K} = \mathbf{K}^T > 0$, and $k_2 > 0$. Choosing $\mathbf{K} = \mathbf{J}$, and using equations (19) and (11), the derivative of the candidate Liapunov function can be written as

$$\begin{array}{lcr} & \dot{V} = \delta\boldsymbol{\omega}^T[\mathbf{h}_B^\times\mathbf{J}^{-1}(\mathbf{h}_B - \mathbf{A}\mathbf{h}_a) + & \dots \\ \dots & + \mathbf{g}_e - \mathbf{J}\boldsymbol{\omega}_B^\times\delta\boldsymbol{\omega} - \mathbf{J}C_R^B(\delta\boldsymbol{\sigma})\mathbf{I}^{-1}\mathbf{h}_R^\times\mathbf{I}^{-1}\mathbf{h}_R + & \dots \\ \dots & - \mathbf{J}C_R^B(\delta\boldsymbol{\sigma})\mathbf{I}^{-1}\mathbf{g}_R - \mathbf{A}\mathbf{g}_a + k_2\delta\boldsymbol{\sigma}] & \end{array} \tag{21}$$

Thus, by selecting control torques $\mathbf{g}_a$ and $\mathbf{g}_e$ to satisfy $\dot{V} < 0$ whenever the tracking error is not zero, the desired control can be achieved.

One controller that uses this approach deals with combined thruster and reaction wheel control. The thrusters on the virtual spacecraft are assumed to the bang-bang type, and this thruster control directly determines the thruster control on the actual spacecraft with

$$\mathbf{g}_e = \mathbf{g}_R \tag{22}$$

Meanwhile the reaction wheels on the actual spacecraft correct for any tracking errors that might exist between the two trajectories. Thus, this reaction wheel feedback controller is found by rewriting equation (21) as

$$\begin{array}{lcr} \mathbf{A}\mathbf{g}_a & = \mathbf{h}_B^\times\mathbf{J}^{-1}(\mathbf{h}_B - \mathbf{A}\mathbf{h}_a) + \mathbf{g}_R - \mathbf{J}\boldsymbol{\omega}_B^\times\delta\boldsymbol{\omega} + & \dots \\ \dots & - \mathbf{J}C_R^B(\delta\boldsymbol{\sigma})\mathbf{I}^{-1}\mathbf{h}_R^\times\mathbf{I}^{-1}\mathbf{h}_R + & \dots \\ \dots & - \mathbf{J}C_R^B(\delta\boldsymbol{\sigma})\mathbf{I}^{-1}\mathbf{g}_R + k_1\delta\boldsymbol{\omega} + k_2\delta\boldsymbol{\sigma} & \end{array} \tag{23}$$

which gives

$$\dot{V} = -k_1\delta\boldsymbol{\omega}^T\delta\boldsymbol{\omega} \leq 0 \tag{24}$$

where $k_1 > 0$. This control law, when used with the error dynamics defined previously, can then be shown to be asymptotically stable.[16]

Simulation

Two different implementations of the above control law are currently under investigation. The first uses the attitude of the wand to represent $\boldsymbol{\sigma}_R$. The second uses the wand as a 3-DOF joystick, where the rotation of the wand represents a desired application of $\mathbf{g}_R$. Preliminary computer simulations have been performed to evaluate each method before implementation on the spacecraft simulator hardware.

The first step in the current simulation procedure involves recording the wand attitude while the operator commands a hypothetical slew maneuver. This task is done while viewing the graphical display shown in Figure 6. Figure 8 shows a plot of a the wand data as the operator slowly rotates the wand about the yaw axis, with slight movement about the other two axes. This data is then post-processed in Matlab.

Translating wand motion into reference attitude

The method using the wand attitude for $\boldsymbol{\sigma}_R$ requires more data manipulation than would likely be feasible for a real-time system, despite the fact that the control of the virtual spacecraft is more intuitive. Namely, the operator simply rotates the wand as they would the virtual spacecraft, since the attitude of the wand is feed directly to the graphical simulation to animate the model of Whorl-I in the CAVE. The drawback is that other variables besides $\boldsymbol{\sigma}_R$ are needed to calculate the reaction wheel torques $\mathbf{g}_a$, as is evident from equation (23). The present data analysis procedure using this method is

- Fit a high-order polynomial to the wand attitude data curves. This allows a good estimate of $\boldsymbol{\theta}_R = f(t)$ given the noisy nature of the data.
- Differentiate to obtain $\dot{\boldsymbol{\theta}}_R$.
- Calculate $\boldsymbol{\sigma}_R = f(t)$ from $\boldsymbol{\theta}_R = f(t)$.
- Calculate $\boldsymbol{\omega}_R = f(t)$ from the relation $\boldsymbol{\omega}_R = S(\boldsymbol{\theta}_R)\dot{\boldsymbol{\theta}}_R$ where

$$S(\boldsymbol{\theta}_R) = \begin{bmatrix} -c\theta_2 s\theta_3 & c\theta_3 & 0 \\ s\theta_2 & 0 & 1 \\ c\theta_2 c\theta_3 & s\theta_3 & 0 \end{bmatrix} \tag{25}$$

  and the shorthand notation refers to $c\theta_i = \cos\theta_i$ and $s\theta_i = \sin\theta_i$.

- Differentiate $\boldsymbol{\omega}_R = f(t)$ to obtain $\dot{\boldsymbol{\omega}}_R = f(t)$.
- Calculate $\mathbf{g}_R$ from Euler's Law

$$\mathbf{g}_R = \mathbf{I}\dot{\boldsymbol{\omega}}_R + \boldsymbol{\omega}_R^{\times}\mathbf{I}\boldsymbol{\omega}_R \tag{26}$$

Knowing σ_R, ω_R, and $\mathbf{g}_R$, we can numerically integrate the equations of motion to simulate the operation of the controller given the initial conditions. While straight forward to describe, this method is difficult to implement in a operational system. The raw wand data is too noisy (see Figure 8) to numerically differentiate, and the proposed method of polynomial fitting is easy to perform during post-processing in Matlab, but impractical during actual operation. A possible solution would be to use more sophisticated filters or smoothers, but this option is not considered in the current study.

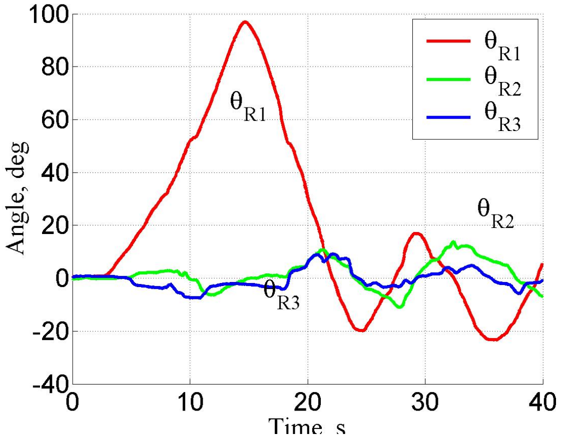

Fig. 8 Saved attitude data from wand.

Translating wand motion into reference torque

The second method is more practical to implement in a working system, but the resulting reference spacecraft motion is less intuitive for the operator to anticipate than the previous method. As the wand is rotated, a finite torque is generated (see Figure 9). This method is analogous to the actuation of a on-off thruster using a flight stick, as was discussed in the literature review. Whereas the visualization of the reference motion in the graphical simulation requires additional steps (since the attitude of the wand is no longer directly driving the attitude of the graphical model), this method is a more realistic option to implement in an operational system.

The torque profile shown in Figure 9 was generated directly from the rotation of the wand from the neutral position (i.e., $\boldsymbol{\theta}_R = \mathbf{0}$). A constant torque of 0.6 N-m was produced for 2 seconds, with a sign determined by the direction of rotation of the wand. This estimated torque value is an estimate of the external torque available to Whorl-I from the cold-gas thrusters. Preliminary studies being conducted in the SSSL on the thruster characterization were used to generate this estimate. Note that we only produce a torque about the $y-$ axis of the simulator, since the wand motion about the other two axes is small in this case. The use of such a deadband prevents very small wand movements from causing unexpected thruster firing.

The current analysis of this control method is performed using computer simulation in Matlab. We let

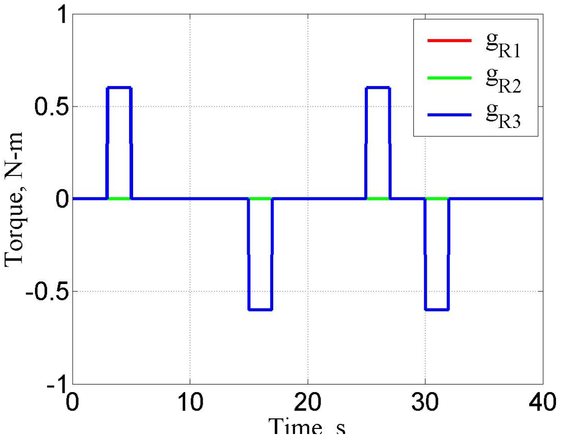

Fig. 9 On-off torque profile generated by wand motion in second control method.

the state vector be

$$\mathbf{X} = \left[\begin{array}{ccccc} \boldsymbol{\omega}_R & \boldsymbol{\sigma}_R & \boldsymbol{\omega}_B & \boldsymbol{\sigma}_B & \mathbf{A}\mathbf{h}_a \end{array} \right]^T \tag{27}$$

Equations (11),(12),(18),(23) and

$$\dot{\boldsymbol{\omega}}_B = \mathbf{J}^{-1}\left[\mathbf{J}\boldsymbol{\omega}_B^{\times}\delta\boldsymbol{\omega} + \mathbf{J}C_R^B\boldsymbol{\omega}_R - k_1\delta\boldsymbol{\omega} - k_2\delta\boldsymbol{\sigma}\right] \tag{28}$$

the are then used as the state vector differential equations that define

$$\dot{\mathbf{X}} = \left[\begin{array}{ccccc} \dot{\boldsymbol{\omega}}_R & \dot{\boldsymbol{\sigma}}_R & \dot{\boldsymbol{\omega}}_B & \dot{\boldsymbol{\sigma}}_B & \mathbf{A}\dot{\mathbf{h}}_a \end{array} \right]^T \tag{29}$$

which represents a coupled system of equations that must be solved using a numerical integration algorithm and given initial conditions. We use the "ode45" function in Matlab, which uses an explicit one-step Runge Kutta medium-order method,[17] to obtain numerical solutions to the set of ordinary differential equations.

We specify the initial conditions

$$\mathbf{X}(t=0) = \left[\begin{array}{c} \boldsymbol{\omega}_R = \mathbf{0} \\ \boldsymbol{\sigma}_R = \mathbf{0} \\ \boldsymbol{\omega}_B = \mathbf{0} + \mathbf{x} \\ \boldsymbol{\sigma}_B = \mathbf{0} + \mathbf{y} \\ \mathbf{A}\mathbf{h}_a = \mathbf{0} \end{array} \right]^T \tag{30}$$

where

$$\mathbf{x} = \left[\begin{array}{ccc} 0.05 & 0.5 & -0.07 \end{array} \right] \text{ rad/s} \tag{31}$$

$$\mathbf{y} = \left[\begin{array}{ccc} 0.1 & -0.1 & 0.75 \end{array} \right] \tag{32}$$

represent arbitrary errors on the initial conditions between the reference and actual spacecraft. The moment of inertia $\mathbf{I}$ of the spacecraft is taken as an estimate of Whorl-I's mass properties. The computer aided design (CAD) software AutoCAD 2002 Mechanical Desktop is used to generate an estimate of the mass properties of the spacecraft simulator given the location and mass of each part

$$\mathbf{I} = \left[\begin{array}{ccc} 5.8 & 0.0 & 0.0 \\ 0.0 & 8.1 & 0.0 \\ 0.0 & 0.0 & 12.7 \end{array} \right] \text{ kg-m}^2 \tag{33}$$

where we assume that the principal axes are aligned with $\mathcal{F}_B$. The values of the gains given in the reference were obtained using a commercial software package to computer optimal gains for given bang-bang maneuvers. Here we select arbitrary gains of $k_1 = k_2 = 5$ for the time being, with the only criteria being that the resulting motor torque commands (i.e., $\mathbf{g}_a$) are less than the maximum available torques of the motors on Whorl-I, which is approximately 4 N-m peak torque. Lastly, the reaction wheels are arranged such that $\mathbf{A}$ is simply the 3×3 identity matrix, and the $\mathbf{I}_s$ of each wheel is 0.075 kg-m^2.

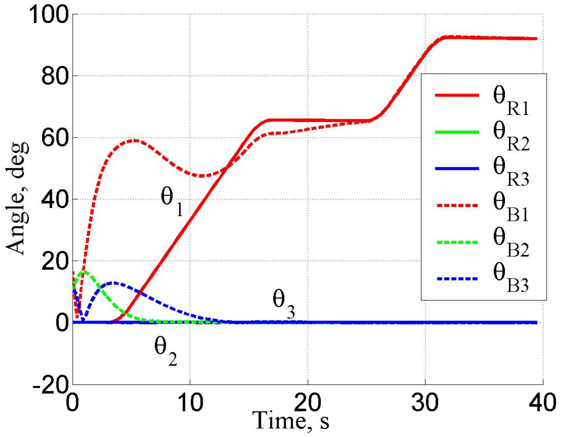

Fig. 10 Predicted attitude (in Euler angles) of reference and actual bodies under the current computer simulation of the second control method.

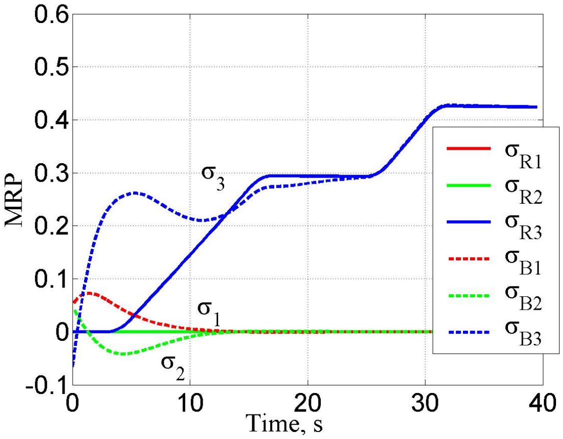

Fig. 11 Predicted attitude (in MRPs) of reference and actual bodies under the current computer simulation of the second control method.

Figures 10 and 11 show the calculated attitude motion of the reference and actual bodies with time, expressed in both MRPs and Euler angles. Figure 12 shows the corresponding angular velocity of these frames. The affect of the on-off thruster firings is clearly visible in these plots. We see periods of coasting, especially for $\mathcal{F}_B$, corresponding to "off" thruster conditions, and periods of acceleration during the "on"

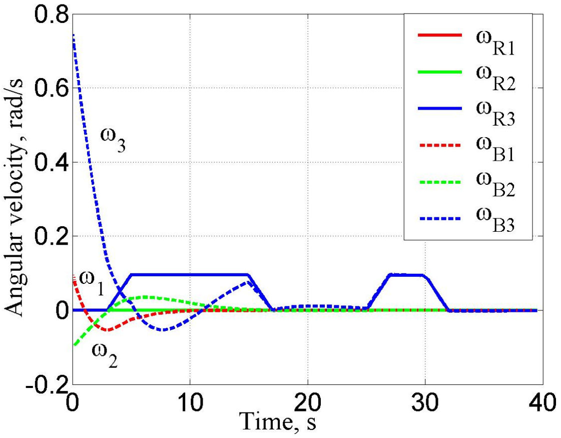

Fig. 12 Predicted angular velocity of reference and actual bodies under the current computer simulation of the second control method.

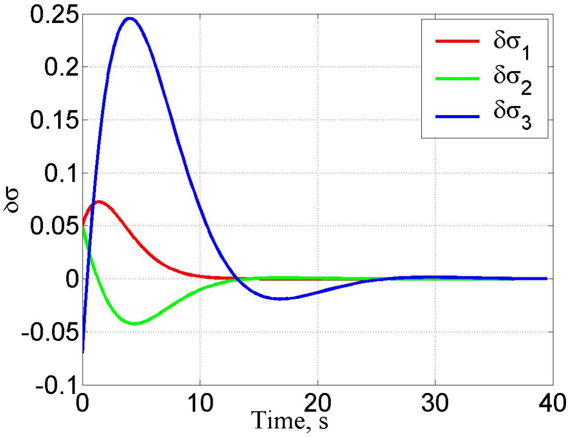

Fig. 13 Plot of attitude error for computer simulation of second control method.

conditions.

We see in Figures 13 and 14 that the attitude and angular velocity error, respectively, rapidly approach zero during the early portions of the trial. The majority of the error also appears to be attributed to initial condition errors. In fact, if x amd y in equations (31) and (32) were zero, theoretically $\mathcal{F}_B$ would track $\mathcal{F}_R$ perfectly.[16] The application of this controller to the real Whorl-I would always have some errors, either from initial condition uncertainties or errors in thruster characterization, motor torque, mass properties, etc.

Lastly, we see that the motor torques did not exceed the 4 N-m limit given the current gains. However, with changes in $\mathbf{g}_a$, say from pressure loss in the nitrogen tank after repeated use, the motor torque would eventually increase until all of the control authority was supplied from the wheels. Further simulations should be conducted to find proper levels for all of these values.

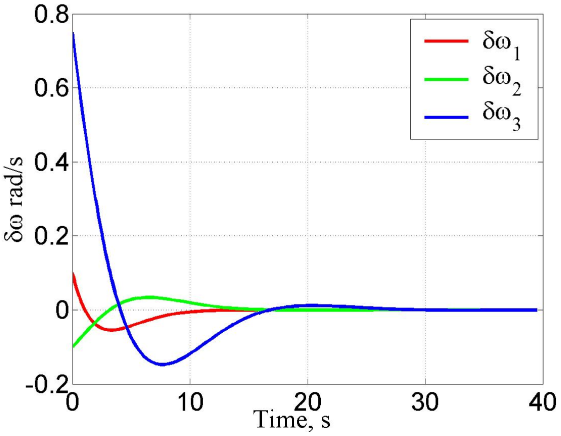

Fig. 14 Plot of angular velocity error for computer simulation of second control method.

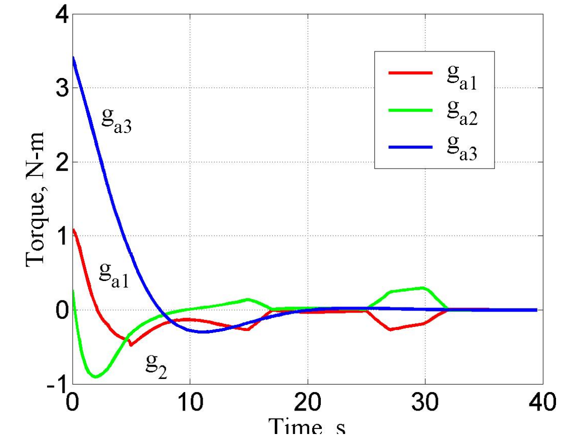

Fig. 15 Plot of axial torques applied to the reaction wheels during computer simulation of second control method.

CONCLUSION

We have identified the usefulness of a system whereby a human operator can use a CAVE to effectively stay "in-the-loop" of a spacecraft control system. Virtual environments can aid in the sensation of telepresence, making the human a more integral part of the system. We have described the potential for the spacecraft simulators located in the SSSL and the VT-CAVE to serve as effective test-beds for such a system. Despite the fact that the current study requires further work before being implemented on Whorl-I or Whorl-II, we have identified a potential control law to test the system. Computer simulations have shown that the 6-DOF wand can serve as an effective input device, with pros and cons to both methods of control. Using the wand to represent the desired spacecraft reference attitude allows for easy visualization of the resulting motion, but correlating this input with the chosen control laws is ill suited for real-time systems and requires more testing. Using the wand to represent the desired external torque on the reference spacecraft is easier to implement, but has challenges with useability in the virtual environment.

References

[1] Wertz, J. and Larson, W. e., *Space Mission Analysis and Design*, Microcosm Press and Kluwer Academic Publishers, 1999.

[2] McGreevy, M. A. and Stoker, C., "Telepresence for Planetary Exploraiton," *The International Society of Optical Engineering: Cooperative Intelligent Robotics in Space*, Vol. 1387, 1990, pp. 110–123.

[3] Brody, A. et al., "Interactive Displays for Trajectory Planning and Proximity Operations," *Journal of Spacecraft and Rockets*, Vol. 30, No. 4, Jul-Aug 1993, pp. 514–518.

[4] Grunwald, A. and Ellis, S., "Visual Display Aid for Orbital Maneuvering: Design Considerations," *Journal of Guidance, Control, and Dynamics*, Vol. 16, No. 1, Jan-Feb 1993, pp. 139–144.

[5] Brody, A. and Ellis, S., "Manual Control Aspects of Space Station Docking Maneuvers," *Proceedings of the 20th Intersociety Conference on Environmental Systems*, Society of Automotive Engineers, Williamsburg, VA, July 1990, pp. 1–6.

[6] Stoker, C., "From Antartica to Space: Use of Telepresence and Virtual Reality in Control of a Remote Underwater Vehicle," *Mobile Robots IX*, Vol. 2352, The International Society of Optical Engineering, 1994, pp. 288–299.

[7] Nguyen, L. et al., *Autonomous Robots*, Vol. 11, chap. Virtual Reality Interfaces for Visualization and Control of Remote Vehicles, Kluwer Academic Publishers, 2001, pp. 59–68.

[8] Museth, K. et al., "Semi-immersive Space Mission Design and Visualization: Case Study of the Terrestrial Planet Finder Mission," *SIGGRAPH Proceedings on Visualization*, Vol. 21-26, IEEE, October 2001, pp. 501–599.

[9] Cheatham, D. and Hackler, C., "Handling Qualities for Pilot Control of Apollo Lunar Landing Spacecraft," *Journal of Spacecraft and Rockets*, Vol. 3, No. 5, May 1966, pp. 632–628.

[10] Clark, H., "Control of a Remote Maneuvering Unit During Satellite Inspection," *Human Factors*, Vol. 8, December 1966, pp. 573–582.

[11] Stengel, R., "Manual Control of the Lunar Module," *Journal of Spacecraft and Rockets*, Vol. 7, No. 8, August 1970, pp. 941–948.

[12] Alexander, H., "Experiments in Teleoperation and Autonomous Control of Space Robotic Vehicles," *Proceedings of the 10th American Control Conference*, IEEE, Boston, MA, June 1991, pp. 1474–1477.

[13] Schwartz, J. L., Peck, M. A., and Hall, C. D., "Historical Review of Air-Bearing Spacecraft Simulators," *Journal of Guidance, Control and Dynamics*, Vol. 26, No. 4, 2003, pp. 513–522.

[14] Kelso, J. e. a., "DIVERSE: A Framework for Building Extensible and Reconfigurable Device-Independent Virtual Environments and Distributed Asynchronous Simulations," *Presence*, Vol. 12, No. 1, 36 2003, pp. 19.

[15] Schaub, H. and Junkins, J., *Analytical Mechanics of Space Systems*, AIAA Education Series, Reston, VA, 2003.

[16] Hall, C., Tsiotras, P., and Shen, H., "Tracking Rigid Body Motion Using Thrusters and Momentum Wheels," *The Journal of the Astronautical Sciences*, Vol. 50, No. 3, Jul-Sep 2002, pp. 311–323.

[17] Hanselman, D. and Littlefield, B., *Mastering Matlab 6: A Comprehensive Tutorial and Reference*, Prentice Hall, Upper Saddle River, New Jersey, 2001.