JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 28, No. 4, July–August 2005

Free-Flying Robot Tested on Parabolic Flights: Kinematic Control

C. Menon,* A. Aboudan,† S. Cocuzza,‡ A. Bulgarelli,‡ and F. Angrilli§
University of Padova, 35131 Padua, Italy

During servicing operations, free-flying robots equipped with robotic arms are required to operate without disturbing the nearby environment through unwanted base oscillations. We present underlying theory, kinematic control software, and experimental results of a three-dimensional free-flying robot prototype able to follow a given trajectory with its end-effector while base movements are minimized. Experiments were performed in a microgravitational environment obtained during parabolic flight tests. For the purposes of this study, only rotation of the base around its yaw axis was minimized, and a robotic arm with four degrees of freedom was used. The displacements of the robot base were monitored by an inertial platform positioned on the base. Results obtained during experiments prove the feasibility of testing such a robot during parabolic flights, show the effectiveness of the kinematic control software developed to minimize base displacements, and suggest directions for future development.

Nomenclature

$\boldsymbol{e}_a$	=	error, defined as difference between desired position and real position of end effector
$\dot{\boldsymbol{e}}_a$	=	time derivative of $\boldsymbol{e}_a$
$\boldsymbol{e}_\varphi$	=	error related to base yaw angle
$\boldsymbol{F}$	=	matrix relating $\dot{\boldsymbol{q}}_b$ to $\dot{\boldsymbol{q}}_{\bar{a}}$
$\boldsymbol{H}$	=	homogeneous system matrix related to conservation of momentum and of angular momentum of free-flying robot
$\boldsymbol{H}_a$	=	submatrix of H related to $\dot{\boldsymbol{q}}_a$
$\boldsymbol{H}_b$	=	submatrix of H related to $\dot{\boldsymbol{q}}_b$
$\boldsymbol{H}_1$	=	submatrix of $\boldsymbol{H}$ related to conservation of momentum
$\boldsymbol{H}_2$	=	submatrix of $\boldsymbol{H}$ related to conservation of angular momentum
$\boldsymbol{J}$	=	Jacobian matrix
$\boldsymbol{J}_a$	=	last four columns of Jacobian matrix (related to robot arm)
$\boldsymbol{J}_b$	=	first six columns of Jacobian matrix (related to robot base)
$\boldsymbol{J}_e$	=	transformation matrix related to $\dot{\boldsymbol{q}}_a$
$\boldsymbol{J}_G$	=	generalized Jacobian matrix
$\boldsymbol{J}_G^T$	=	transpose of generalized Jacobian matrix
$\tilde{\boldsymbol{J}}_G$	=	first three rows of generalized Jacobian matrix
$\tilde{\boldsymbol{J}}_G^T$	=	transpose of $\tilde{\boldsymbol{J}}_G$
$\tilde{\boldsymbol{J}}_G^\dagger$	=	pseudoinverse matrix of $\tilde{\boldsymbol{J}}_G$
$\boldsymbol{J}_{\mathrm{MC}}$	=	Jacobian matrix relating $\dot{\boldsymbol{q}}$ to $\dot{\boldsymbol{r}}$
$\boldsymbol{J}_{\mathrm{RF}}$	=	Jacobian matrix relating $\dot{\boldsymbol{q}}$ to velocities of robot reference frames
$\tilde{\boldsymbol{J}}_{\mathrm{RF}}$	=	last three rows of $\boldsymbol{J}_{\mathrm{RF}}$
$\boldsymbol{J}_s$	=	transformation matrix related to $\boldsymbol{\omega}_b$
$\boldsymbol{J}_v$	=	Jacobian matrix relating $\dot{\boldsymbol{q}}_a$ to $\boldsymbol{v}_b$
$\boldsymbol{J}_\omega$	=	Jacobian matrix relating $\dot{\boldsymbol{q}}_a$ to $\boldsymbol{\omega}_b$
$\boldsymbol{K}_a$	=	gain related to $\boldsymbol{e}_a$
$\boldsymbol{K}_d$	=	damping gain matrix
$\boldsymbol{K}_\varphi$	=	gain related to base yaw angle
$\dot{\boldsymbol{q}}$	=	velocity vector of all joints in system
$\dot{\boldsymbol{q}}_a$	=	velocity vector of arm joints
$\dot{\boldsymbol{q}}_{\bar{a}}$	=	vector containing possible constraints that might be chosen for $\dot{\boldsymbol{q}}_a$
$\dot{\boldsymbol{q}}_b$	=	velocity vector of fictitious arm joints (linking inertial reference frame to base reference frame)
$\dot{\boldsymbol{r}}$	=	linear velocity vector of centers of mass of robot links
$\boldsymbol{v}$	=	vector containing linear and angular velocities of end effector
$\boldsymbol{v}_b$	=	base linear velocity vector
$\boldsymbol{x}$	=	end-effector position
$\dot{\boldsymbol{x}}$	=	time derivative of end-effector position
$\boldsymbol{x}_d$	=	desired end-effector position
$\dot{\boldsymbol{x}}_d$	=	time derivative of desired end-effector position
$\boldsymbol{\Phi}$	=	matrix related to minimization of $\boldsymbol{e}_\varphi$
φ	=	base yaw angle
φ_d	=	desired base yaw angle
$\boldsymbol{\omega}$	=	angular velocity vector of centers of mass of robot links
$\boldsymbol{\omega}_b$	=	base angular velocity vector

Received 22 February 2004; revision received 24 August 2004; accepted for publication 25 August 2004. Copyright © 2004 by the American Institute of Aeronautics and Astronautics, Inc. All rights reserved. Copies of this paper may be made for personal or internal use, on condition that the copier pay the $10.00 per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923; include the code 0731-5090/05 $10.00 in correspondence with the CCC.

*Ph.D. Student, CISAS "G.Colombo," Department of Mechanical Engineering, Via Venezia 15; menoncarlo@stargatenet.it. Student Member AIAA.

†Ph.D. Student, Department of Mechanical Engineering, Via Venezia 1; alessio.aboudan@unipd.it.

‡Ph.D. Student, CISAS "G.Colombo," Department of Mechanical Engineering, Via Venezia 15.

§Supervising Professor, Director of CISAS and Professor, Via Venezia 1.

Introduction

FREE-FLYING robots have received considerable research attention, thanks to the promising applications they have in space. During extravehicular activity, robots can take the place of astronauts and spare humans from the hostile space environment. Although astronauts are only exposed to the space environment for short periods of time, high-energy radiation caused by charged particles inside Van Allen belts and solar flares can be harmful. When the exposure time increases, this danger is even greater, and the use of robots becomes highly advisable. In the case of intravehicular activity, a free-flying robot can be used to operate on artifacts using extended mechanical arms or, as was done in the Personal Satellite Assistant project,[1] monitor and inspect internal space environments. However, the most compelling application of a free-flying robot is perhaps represented by servicing missions. NASA, the Canadian Space Agency (CSA), and many other space agencies have funded important programs for developing space robotic arms operating in servicing operations[2–9]; remarkable servicing missions have been designed, such as those for repairing the Hubble Space Telescope.[10–12] In such scenarios, free-flying robots would successfully inspect and work in multipurpose operations, taking advantage of their capability of flying in free space.

A key challenge in the design of free-flying robots with extended arms is to minimize perturbations on the robot's base caused by movements of its arm during operations. Minimizing base disturbances is of primary importance in such situations, because

uncontrolled base movements can cause unpredictable and detrimental collisions. A common method for reducing undesired displacements of the robot base is to employ thrusters mounted on the base to counterbalance forces induced by the arm. However, this solution requires extra fuel consumption, which can significantly shorten the duration of the operation, because refuelling is typically not an option. As an alternative, researchers[13–23] have developed kinematic control strategies that produce optimal path trajectories for the arm without inducing unwarranted displacements of the base. In particular, Caccavale and Siciliano[24] have presented an effective approach based on a redundant arm used to guide the six degrees of freedom (DOF) of the end effector while controlling the attitude of the base.

In addition to the preceding theoretical developments, several groups have built and tested free-flying robots in various test environments. For almost 45 years, space agencies and government laboratories have developed facilities where experiments were performed on air-tables,[25] perhaps the most effective planar systems for testing space robots. Also, many universities have developed air-bearing simulators, for example, Virginia Polytechnic Institute and State University,[25] University of Maryland,[26] Stanford University,[27,28] University of Victoria,[8,9,29,30] Tokyo Institute of Technology,[31,32] Naval Postgraduate School,[33,34] Massachusetts Institute of Technology,[35] Carnegie Mellon University and Texas Robotics and Automation Center,[36] University of Padova,[37] Georgia Institute of Technology, and many others.[25]

Because air-tables are planar systems, they are inherently not suitable for robots operating in three-dimensional space, such as free-flying robots. To overcome this limitation, test beds based on suspension systems have been developed.[38–45] Although they are suitable for most space manipulators, free-flying robots with extended arms are not commonly tested in these facilities because unwarranted disturbances are induced to their bases. As an alternative, underwater test environments, which better simulate unconstrained three-dimensional space, have been developed and applied.[46–52] However, the drag force induced by water on the robot during arm manipulations introduces another source of disturbance not present in the space environment. The team led by Yoshida,[53] therefore, presented an experiment performed in low Earth orbit. Their system, called Engineering Test Satellite VII (ETS VII), successfully demonstrated that a free-flying robot, in this case a satellite with a robotic arm, can be used for docking operations in space.

The work presented here addresses the problem of minimizing the base displacements of a free-flying robot while guiding its arm to reach a prefixed target. Its main contribution is the development of a kinematic control algorithm, which was implemented on a custom-built free-flying robot tested in microgravity conditions obtained during parabolic flights. The free-flying robot prototype, shown in Fig. 1, was built with particular geometry and specifications in order to test the performance of the kinematic algorithm: the mass and inertia of the robot's base were comparable with those of the arm, increasing the effect of coupling between base displacements and arm movements; the robotic arm had four DOFs, making it possible to keep the yaw angle of the base stationary while controlling the three position variables of the end effector. With regard to control strategies, the kinematic control algorithm was developed on the basis of prior theoretical concepts on the subject, in particular those presented in Ref. 24. Conservation of momentum and of angular momentum was taken into account in the Jacobian matrix of the robot, and redundant manipulator theory[54] was used to control both end-effector position and base orientation.

This paper is organized as follows. The first section describes how the free-flying robot prototype was modeled. The two subsequent sections discuss underlying theory, introducing the reader to the kinematic inversion control section. Robot hardware is then presented, and experimental results are shown and discussed. In the last section, conclusions and future directions are presented.

Reference Frames of Robot

The choice of the position and orientation of the reference frames of a robot is an important issue, especially if the robot has a free-floating base. In this paper, the entire system, base and robotic arm, is modeled as a single robot with a fixed base. The position of the base is computed using the concept of fictitious arm, a sequence of frames and joints linking the inertial reference frame to the base itself. Because the free-flying base has six DOFs, the fictitious arm chosen for calculating the position and orientation of the base has three prismatic joints plus three revolute joints. The real four-DOF robotic arm is connected to the fictitious arm and is represented by a sequence of four links plus a rigid end-effector link.

The robot is sketched in Fig. 2. The first reference frame, $\sum_0$, is the inertial frame, and the last one, $\sum_T$, represents the position of the end-effector tip (Tool). For the first three DOFs of the fictitious arm, the prismatic joints are positioned just before reference frames $\sum_1$, $\sum_2$, and $\sum_3$, along their z axes. The following joints are all revolute joints and are also positioned just before reference frames $\sum_4$, $\sum_5$, $\sum_6$, $\sum_7$, $\sum_8$, $\sum_9$, and $\sum_{10}$ along their z axes. $\sum_0$ and $\sum_6$ overlap at starting time T_0.

Generalized Jacobian Matrix

Using the model described in the preceding section and representing the entire robotic system, base and robotic arm, as a single robot with a fixed base makes it possible to write a single Jacobian matrix $\boldsymbol{J}$, which can be used to express the end-effector velocity as follows:

$$\boldsymbol{v} = \boldsymbol{J} \cdot \dot{\boldsymbol{q}} \tag{1}$$

Vector $\dot{\boldsymbol{q}}$ can be divided into the joint variables of the base $\dot{\boldsymbol{q}}_b$ (the first six elements of $\dot{\boldsymbol{q}}$) and those of the arm $\dot{\boldsymbol{q}}_a$ (the last four elements

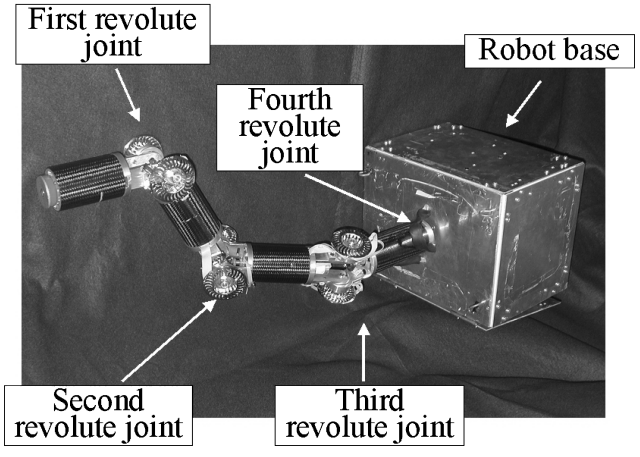

Fig. 1 Free-flying robot prototype with four-DOF arm.

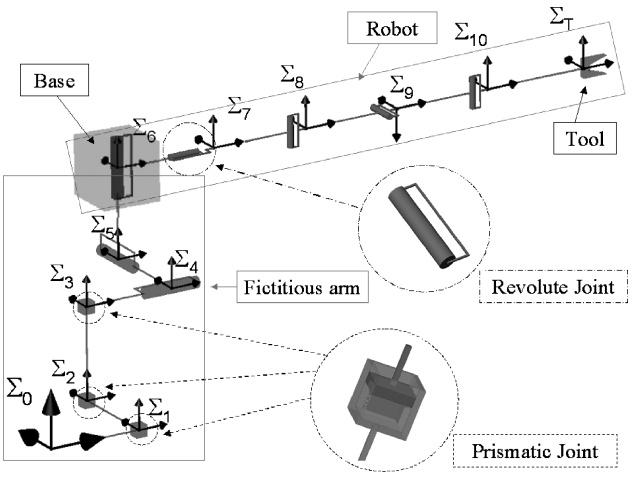

Fig. 2 Reference frame.

of $\dot{\boldsymbol{q}}$). Similarly, in the Jacobian matrix of Eq. (1) the first six columns are related to base $\boldsymbol{J}_b$ and the last four to arm joint variables $\boldsymbol{J}_a$. Thus, Eq. (1) can be written in the following form:

$$\boldsymbol{v} = \boldsymbol{J}_b \cdot \dot{\boldsymbol{q}}_b + \boldsymbol{J}_a \cdot \dot{\boldsymbol{q}}_a \tag{2}$$

Because the robot does not have a base fixed to an inertial reference frame, a particular Jacobian matrix, generalized Jacobian matrix $\boldsymbol{J}_G$, which takes into account the conservation of momentum (CM) and conservation of angular momentum (CAM), must be computed. If, at starting time T_0, the system is stationary, CM and CAM are expressed by the following equations:

$$\boldsymbol{M} \cdot \dot{\boldsymbol{r}} = 0 \tag{3}$$

$$\boldsymbol{I} \cdot \boldsymbol{\omega} + \boldsymbol{M} \cdot \boldsymbol{r} \wedge \dot{\boldsymbol{r}} = 0 \tag{4}$$

where $\boldsymbol{I}$ is the inertia matrix, $\boldsymbol{M}$ the mass matrix, and $\boldsymbol{r}$ the vector of the distances of the centers of mass of the robot links from the origin of the inertial reference frame. From Eq. (4), and defining $\boldsymbol{J}_\omega$ as in Ref. 24, base angular velocity $\boldsymbol{\omega}_b$ is obtained as follows:

$$\boldsymbol{\omega}_b = \boldsymbol{J}_\omega \cdot \dot{\boldsymbol{q}}_a \tag{5}$$

Equation (5) is of fundamental importance because it states that the angular velocity of the floating base can be guided by the velocities of the arm joints.

From Eq. (3), and defining $\boldsymbol{J}_s$ and $\boldsymbol{J}_e$ as suggested in Ref. 24, the velocity vector of the end-effector can be derived:

$$\boldsymbol{v} = \boldsymbol{J}_s \cdot \boldsymbol{\omega}_b + \boldsymbol{J}_e \cdot \dot{\boldsymbol{q}}_a \tag{6}$$

Substituting Eq. (5) into Eq. (6) yields

$$\boldsymbol{v} = \boldsymbol{J}_G \cdot \dot{\boldsymbol{q}}_a \tag{7}$$

Computation of $\boldsymbol{J}_G$

This section describes the algorithm used to compute the $\boldsymbol{J}_G$ matrix in the software developed for the parabolic flight tests. The main characteristic of this algorithm is that it is efficient and suitable for software implementation.

The first step consists of writing Eqs. (3) and (4) as a function of vector $\dot{\boldsymbol{q}}$. Using matrices $\boldsymbol{J}_{\mathrm{RF}}$, $\tilde{\boldsymbol{J}}_{\mathrm{RF}}$, and $\boldsymbol{J}_{\mathrm{MC}}$ defined in the Nomenclature, Eqs. (3) and (4) can be reduced to the fundamental equations (8) and (9):

$$\boldsymbol{M} \cdot \boldsymbol{J}_{\mathrm{MC}} \cdot \dot{\boldsymbol{q}} = 0 \tag{8}$$

$$\boldsymbol{I} \cdot \tilde{\boldsymbol{J}}_{\mathrm{RF}} \cdot \dot{\boldsymbol{q}} + \boldsymbol{M} \cdot \boldsymbol{r} \wedge \boldsymbol{J}_{\mathrm{MC}} \cdot \dot{\boldsymbol{q}} = 0 \tag{9}$$

which can be rewritten, respectively, as

$$\boldsymbol{H}_1 \cdot \dot{\boldsymbol{q}} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \tag{10}$$

$$\boldsymbol{H}_2 \cdot \dot{\boldsymbol{q}} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \tag{11}$$

or, in compact form,

$$\boldsymbol{H} \cdot \dot{\boldsymbol{q}} = 0 \tag{12}$$

where $\boldsymbol{H}$ is written as

$$\boldsymbol{H} = \begin{pmatrix} \boldsymbol{H}_1 \\ \boldsymbol{H}_2 \end{pmatrix} \tag{13}$$

Equation (12) represents conditions CM and CAM in a compact equation. To obtain the generalized Jacobian matrix, $\boldsymbol{H}$ is separated into submatrices relative to the base and the arm:

$$(\boldsymbol{H}_b \quad \boldsymbol{H}_a) \cdot \begin{pmatrix} \dot{\boldsymbol{q}}_b \\ \dot{\boldsymbol{q}}_a \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix} \tag{14}$$

Deriving $\dot{\boldsymbol{q}}_b$ from Eq. (14) as follows:

$$\dot{\boldsymbol{q}}_b = \left(-\boldsymbol{H}_b^{-1} \cdot \boldsymbol{H}_a\right) \cdot \dot{\boldsymbol{q}}_a \tag{15}$$

and substituting Eq. (15) in Eq. (2), the end-effector velocities are expressed as

$$\boldsymbol{v} = \boldsymbol{J}_b \cdot \left(-\boldsymbol{H}_b^{-1} \cdot \boldsymbol{H}_a\right) \cdot \dot{\boldsymbol{q}}_a + \boldsymbol{J}_a \cdot \dot{\boldsymbol{q}}_a \tag{16}$$

Defining the generalized Jacobian matrix as

$$\boldsymbol{J}_G = \boldsymbol{J}_a + \boldsymbol{J}_b \cdot \left(-\boldsymbol{H}_b^{-1} \cdot \boldsymbol{H}_a\right) \tag{17}$$

and substituting $\boldsymbol{J}_G$ in Eq. (16), the end-effector velocities can be expressed, as in Eq. (7), as a function only of $\dot{\boldsymbol{q}}_a$.

The last two terms of Eq. (17) define matrix $\boldsymbol{F}$

$$\boldsymbol{F} = -\boldsymbol{H}_b^{-1} \cdot \boldsymbol{H}_a \tag{18}$$

which plays a crucial role in the kinematic inversion algorithm, as described in the next section.

Kinematic Inversion

This section describes the kinematic inversion algorithm implemented in the free-flying robot prototype. As noted earlier, only the position of the end effector is controlled, and thus only the first three rows of the generalized Jacobian matrix, namely, the 3×4 matrix $\tilde{\boldsymbol{J}}_G$, are considered. Because $\tilde{\boldsymbol{J}}_G$ is not invertible, a pseudoinversion is computed. To solve singularity problems, a damping gain matrix $\boldsymbol{K}_d$ is introduced, and, as suggested in Ref. 24, a damped pseudoinverse of $\tilde{\boldsymbol{J}}_G$ is computed as

$$\tilde{\boldsymbol{J}}_G^{\dagger} = \boldsymbol{J}_G^T \cdot \left(\tilde{\boldsymbol{J}}_G \cdot \tilde{\boldsymbol{J}}_G^T + \boldsymbol{K}_d \cdot \boldsymbol{I}\right)^{-1} \tag{19}$$

To compute the kinematic inversion, an algorithm based on minimization of the error vector is used. The error is defined as the difference between the desired position and the real position of the end effector:

$$\boldsymbol{e}_a = \boldsymbol{x}_d - \boldsymbol{x} \tag{20}$$

and its derivative is

$$\dot{\boldsymbol{e}}_a = \dot{\boldsymbol{x}}_d - \dot{\boldsymbol{x}} \tag{21}$$

According to the following equation, in which $\dot{\boldsymbol{x}}$ is related to $\boldsymbol{e}_a$,

$$\dot{\boldsymbol{x}} = \dot{\boldsymbol{x}}_d + \boldsymbol{K}_a \cdot \boldsymbol{e}_a \tag{22}$$

the temporal evolution of the error is represented by

$$\dot{\boldsymbol{e}}_a + \boldsymbol{K}_a \cdot \boldsymbol{e} = 0 \tag{23}$$

which exponentially converges to zero.

Substituting the end-effector linear velocity of Eq. (22) in Eq. (7) and using the damped pseudoinverse of the generalized Jacobian matrix defined in Eq. (19), the velocities of the arm joint variables are computed as

$$\dot{\boldsymbol{q}}_a = \tilde{\boldsymbol{J}}_G^{\dagger} \cdot (\dot{\boldsymbol{x}}_d + \boldsymbol{K}_a \cdot \boldsymbol{e}_a) \tag{24}$$

To introduce a mathematical condition for the robot base position in Eq. (24), redundant manipulator theory is used,[54] and $\dot{\boldsymbol{q}}_a$ is thus expressed as

$$\dot{\boldsymbol{q}}_a = \tilde{\boldsymbol{J}}_G^{\dagger} \cdot (\dot{\boldsymbol{x}}_d + \boldsymbol{K}_a \cdot \boldsymbol{e}_a) + (\boldsymbol{I} - \tilde{\boldsymbol{J}}_G^{\dagger} \cdot \tilde{\boldsymbol{J}}_G) \cdot \dot{\boldsymbol{q}}_{\bar{a}} \tag{25}$$

where $\dot{\boldsymbol{q}}_{\bar{a}}$ is the vector of the arbitrary joint velocities of the arm.[54]

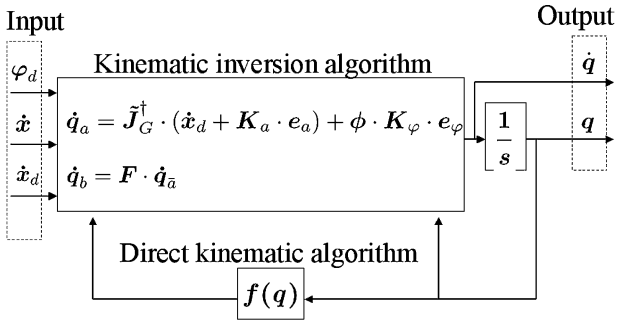

Fig. 3 Kinematic loop.

For the purposes of this study, only the yaw angle of base φ is taken into account. This angle corresponds to the sixth variable of the fictitious arm, and therefore only the sixth row of matrix $\boldsymbol{F}$, called $\boldsymbol{F}_6$, is considered. The joint variable vector of the arm is obtained by computing the transpose of matrix $\boldsymbol{F}_6$ (see Appendix):

$$\dot{\boldsymbol{q}}_{\bar{a}} = \boldsymbol{F}_6^T \cdot \boldsymbol{K}_\varphi \cdot \boldsymbol{e}_\varphi \tag{26}$$

where

$$\boldsymbol{e}_\varphi = \varphi_d - \varphi \tag{27}$$

Equations (25) and (26) yield the final equation of the kinematic control:

$$\dot{\boldsymbol{q}}_a = \tilde{\boldsymbol{J}}_G^{\dagger} \cdot (\dot{\boldsymbol{x}}_d + \boldsymbol{K}_a \cdot \boldsymbol{e}_a) + \boldsymbol{\Phi} \cdot \boldsymbol{K}_\varphi \cdot \boldsymbol{e}_\varphi \tag{28}$$

where $\boldsymbol{\Phi}$ is defined by

$$\boldsymbol{\Phi} = \left(\boldsymbol{I} - \tilde{\boldsymbol{J}}_G^{\dagger} \cdot \tilde{\boldsymbol{J}}_G\right) \cdot \boldsymbol{F}_6^T \tag{29}$$

Equation (28) makes it possible to control the four-DOF arm of the free-flying robot prototype, guiding its end effector to desired position $\boldsymbol{x}_d$ while keeping the yaw angle of the base at desired position φ_d. To compute the whole joint variable vector $\dot{\boldsymbol{q}}$, vector $\dot{\boldsymbol{q}}_b$ is calculated, according to Eqs. (15) and (18), as

$$\dot{\boldsymbol{q}}_b = \boldsymbol{F} \cdot \dot{\boldsymbol{q}}_{\bar{a}} \tag{30}$$

Figure 3 shows the kinematic loop. Note that dynamic terms are computed in the loop. In fact, output $\dot{\boldsymbol{q}}_b$ is also used as an input for computing matrix $\tilde{\boldsymbol{J}}_G^{\dagger}$.

The software developed for guiding the free-flying robot comprises two parts: the kinematic inversion loop, just described, and the centralized joint control. One of the main characteristics of this software is its ability to run the kinematic inversion off-line, thereby saving computational resources for real-time centralized joint control. To better understand the architecture of this software, the centralized joint control is briefly described next.

The equations of motion of the free-flying system are

$$\boldsymbol{M}_g(\boldsymbol{q}) \cdot \ddot{\boldsymbol{q}}_a + \boldsymbol{N}_g(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{\tau}_a \tag{31}$$

where $\boldsymbol{M}_g$ and $\boldsymbol{N}_g$ are, respectively, the generalized mass and dynamic coupling matrices of the system.

Using the robust control suggested by Lewis et al.,[55] matrix $\boldsymbol{M}_g$ is considered diagonal and constant, whereas $\boldsymbol{N}_g$ coincides with the void matrix. Therefore, Eq. (31) does not depend on entire vector $\boldsymbol{q}$ but only on vector $\boldsymbol{q}_a$. The input for the centralized joint control is thus reduced to the vector of the arm joint, making it possible to run the kinematic inversion off-line. It can, therefore, run very fast in a real-time loop, without compromising the efficiency of the robot's software as a whole.

The trajectory of the end effector, the input of the kinematic inversion loop, must be carefully chosen in order to guide a free-flying robot successfully. As opposed to a fixed-base robot, the working space of a free-flying robot depends on the initial position of the robotic arm. Examples are given in Figs. 4 and 5, which show a free-flying robot in its initial and final positions. The robotic arm is assumed to be mechanically constrained between points A and B. In Fig. 4, the robotic arm cannot reach the target (hatched area), whereas in Fig. 5 the different initial position of the arm does allow this. Feasible trajectories of the end effector must therefore be correctly computed. A suitable trajectory can be designed using an algorithm, which, starting from a prefixed initial position, computes the three-dimensional working space of the robot. However, if only the kinematic loop is tested a simpler procedure is used: a time function is assigned to the arm joint variables, and the trajectory of the end effector is then computed by means of the direct kinematics of the robot. Following this procedure, a multibody dynamic software package, such as Adams or VisualNastran, can be used to obtain vector $\boldsymbol{\nu}$, the input of the kinematic control.

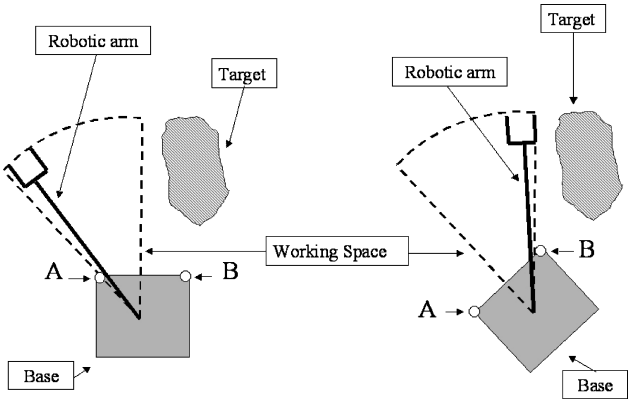

Fig. 4 First possible starting position.

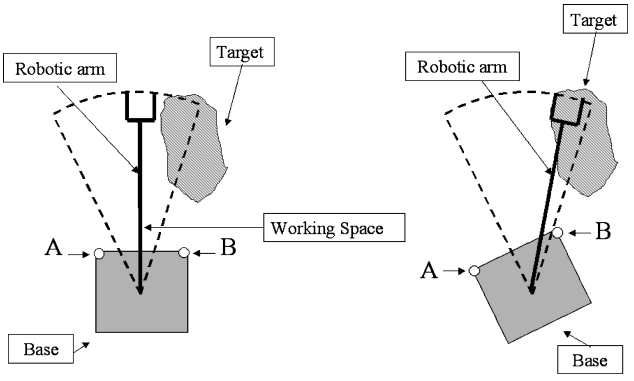

Fig. 5 Second possible starting position.

Robot Subsystems

The kinematic control just described was one subsystem of a complex system comprising 1) a tailored robot, with a lightweight structure made of competitive materials such as carbon-fiber-reinforced plastics and aluminum alloy (Al7075-T6); 2) an electronic system—PC104 GEODE 277MHz 128 Mb, IDE Flash Disk 512 Mb, dc/dc Converter 12-5V, and two RSC629 Driver Boards, all mounted on the robot base; 3) a power supply system—three 12V 1.2Ah batteries; 4) four 4.2-W CC micromotors with preloaded planetary gearboxes; 5) sensors—optical encoders, limit switch devices for arm link movements, and one Microstrain 3DM-G inertial platform mounted inside the base; 6) a real-time operative system, QNX; 7) kinematic inversion software; and 8) real-time centralized joint control system composed of a) "onboard data handling" for telemetry and external communication, complying with ESA Packet Telemetry Standard,[56] and b) robotic system controller, complying with NASA/NASREM software architecture.[57]

All of these subsystems were developed to perform the parabolic flight tests, which proved the feasibility of a free-flying three-dimensional robot prototype. The main results are shown in the next section.

Tests and Future Improvements

Kinematic control performance was demonstrated during a parabolic flight in July 2003. A microgravitational environment made it possible to verify the feasibility of guiding a three-dimensional free-flying robot. For the same end-effector trajectory, two types of experiments were performed, to show the possibility of minimizing rotation around the yaw axis of the base.

In the first type of experiment, gain matrix $\boldsymbol{K}_\varphi$ of Eq. (28) coincides with the identity matrix. In Fig. 6, the simulated temporal evolution of the yaw angle of the base is represented by the dashed line; the continuous line shows experimental results for the same gain matrix. Figure 6 shows that the range of the yaw angle is about 12 deg. The temporal evolution of the error of the experimental data with respect to the simulated results is shown in Fig. 7. As the inertial platform positioned on the base of the robot had an accuracy of 1 deg, Fig. 7 shows consistency between theoretical and experimental data and reveals the effectiveness of the guiding control and the quality of the whole system. Theoretical and experimental roll and pitch angles are shown in Figs. 8 and 9, in which dashed lines represent simulated results and continuous lines experimental data. The temporal evolution of the error is shown in Figs. 10 and 11.

In the second type of experiment, matrix $\boldsymbol{K}_\varphi$ was increased by a factor of thousand, that is, $\boldsymbol{K}_\varphi = 1000 \cdot \boldsymbol{I}$. Figures 12–14 present the yaw, roll, and pitch angles, and Figs. 15–17 present the temporal evolution of the error for this new gain matrix. These figures show that the control system was able to follow the desired trajectories correctly.

The most important result of the parabolic flight experiments can be seen by comparing Figs. 6 and 12. As indicated by Eq. (28), if matrix $\boldsymbol{K}_\varphi$ is increased, the amplitude of one of the generalized displacements of the robot base can be minimized. In Fig. 12, the amplitude of the yaw angle, which was the variable minimized, is one order of magnitude smaller than that of the same angle in Fig. 6. This important result proves the real possibility of keeping the robot base stationary during arm operations. This aspect is of great interest, especially for space servicing missions.

For the future, many improvements are foreseen. In this work, parabolic flight tests showed the difficulty of obtaining repeatable data. The inevitably imprecise movements of the operators who carried out the experiments and the disturbances produced on the free-flying robot by aircraft oscillations compromised most of the results. An automatic releasing system is therefore of great interest,

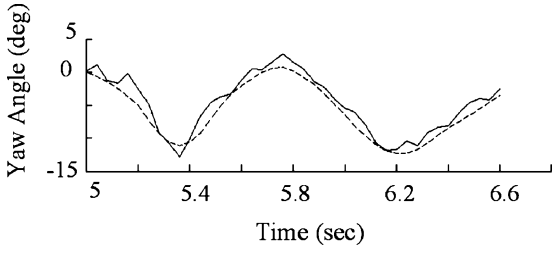

Fig. 6 Yaw angle of robot base for $K_\varphi = I$.

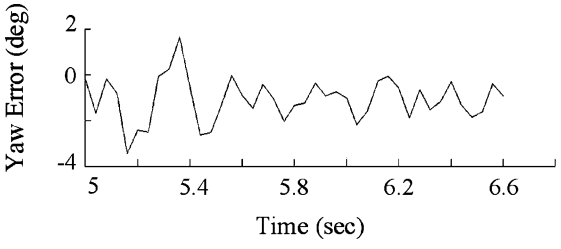

Fig. 7 Temporal evolution of error between simulated and experimental yaw angle for $K_\varphi = I$.

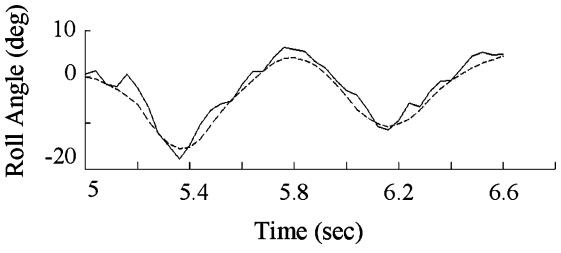

Fig. 8 Roll angle of robot base for $K_\varphi = I$.

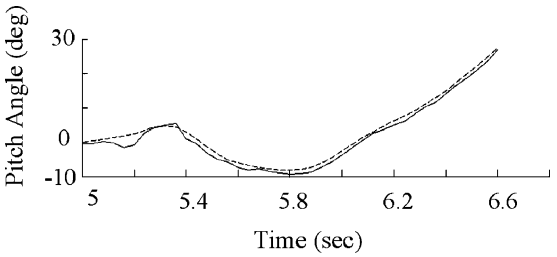

Fig. 9 Pitch angle of robot base for $K_\varphi = I$.

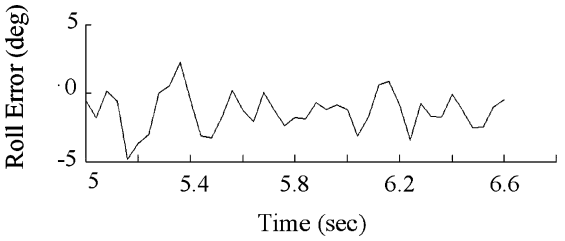

Fig. 10 Temporal evolution of error between simulated and experimental roll angles for $K_\varphi = I$.

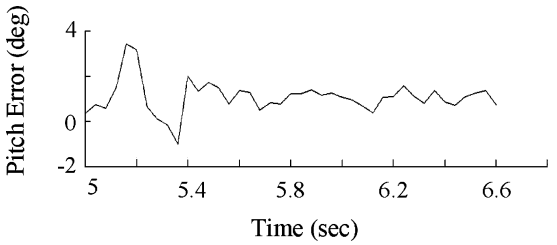

Fig. 11 Temporal evolution of error between simulated and experimental pitch angles for $K_\varphi = I$.

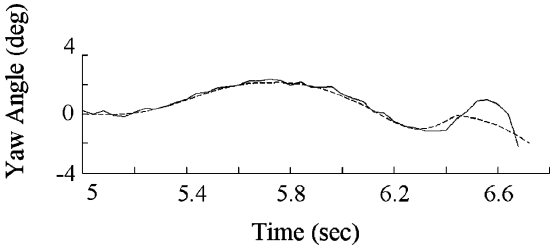

Fig. 12 Yaw angle of robot base for $K_\varphi = 1000 \cdot I$.

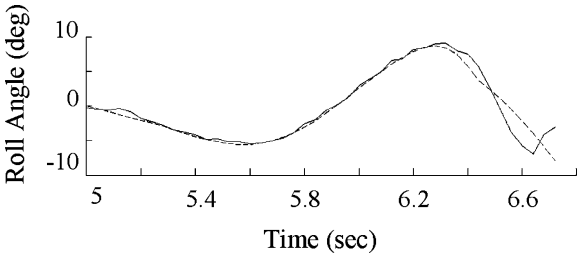

Fig. 13 Roll angle of robot base for $K_\varphi = 1000 \cdot I$.

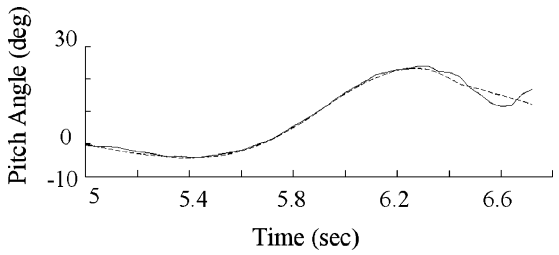

Fig. 14 Pitch angle of robot base for $K_\varphi = 1000 \cdot I$.

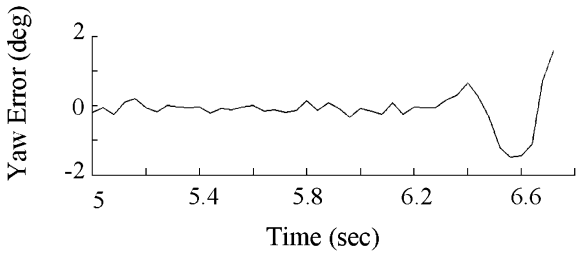

Fig. 15 Temporal evolution of error between simulated and experimental yaw angles for $K_\varphi = 1000 \cdot I$.

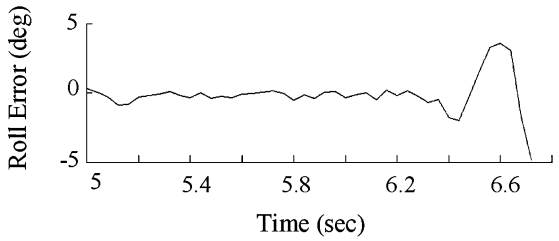

Fig. 16 Temporal evolution of error between simulated and experimental roll angles for $K_\varphi = 1000 \cdot I$.

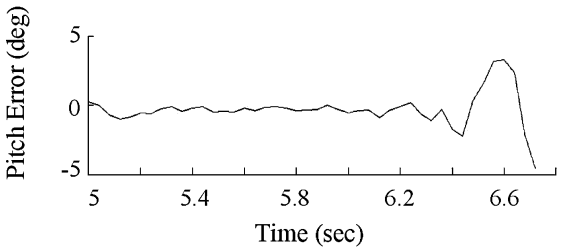

Fig. 17 Temporal evolution of error between simulated and experimental pitch angles for $K_\varphi = 1000 \cdot I$.

so that the system will be able to position the robot in the center of the aircraft and release it during the microgravitational parabolic phase, an operation that was conducted manually in previous flight experiments. For future experiments, to validate the implemented algorithm for more general cases, all of the generalized displacements of the robot base will be minimized. Despite the considerable disturbances that occurred during the parabolic flight tests, experimental results did show that the kinematic control system described here performed well while minimizing the yaw rotation of the base.

Conclusions

Control software developed for guiding a three-dimensional free-flying robot is presented in this paper, together with experimental results obtained during a parabolic flight test campaign. According to experimental data, the software performed well, showing the possibility of minimizing one of the generalized displacements of the free-flying base during robotic arm movements. The control algorithm was able to limit the base yaw angle to 2 deg, which corresponds to one order-of-magnitude improvement when compared with the case in which the base motion was not minimized. In addition, theoretical results were consistent with experimental data, proving the reliability of the implemented algorithm, as well as the quality of the structure and design of the free-flying robot prototype. Experiments also highlighted the need for an automated release system aimed at increasing the duration of tests during the parabolic microgravitational phase while reducing operator disturbances on the robot attitude.

Appendix: Transpose of the Jacobian Matrix

This Appendix gives the mathematical procedure to obtain Eq. (26). If $\boldsymbol{y}$ is a vector defined as

$$\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{q}) \tag{A1}$$

and if its desired value is $\boldsymbol{y}_d$, the following equation can be written:

$$\boldsymbol{e}_c = \boldsymbol{y}_d - \boldsymbol{y} \tag{A2}$$

A positive definite Lyapunov function is defined:

$$\boldsymbol{V} = \tfrac{1}{2} \cdot \boldsymbol{e}_c^T \cdot \boldsymbol{K} \cdot \boldsymbol{e}_c \tag{A3}$$

Its derivative is

$$\dot{\boldsymbol{V}} = \boldsymbol{e}_c^T \cdot \boldsymbol{K} \cdot \dot{\boldsymbol{e}}_c \tag{A4}$$

By substituting Eq. (A2) into Eq. (A4), the following equation can be written:

$$\dot{\boldsymbol{V}} = \boldsymbol{e}_c^T \cdot \boldsymbol{K} \cdot \dot{\boldsymbol{y}}_d - \boldsymbol{e}_c^T \cdot \boldsymbol{K} \cdot \dot{\boldsymbol{y}} \tag{A5}$$

Because $\boldsymbol{y}$ is defined as in Eq. (A1), its velocity can be written as follows:

$$\dot{\boldsymbol{y}} = \boldsymbol{J}_c \cdot \dot{\boldsymbol{q}} \tag{A6}$$

If the condition

$$\dot{\boldsymbol{y}}_d = 0 \tag{A7}$$

is chosen, Eq. (A5) can be written in the following way:

$$\dot{\boldsymbol{V}} = -\boldsymbol{e}_c^T \cdot \boldsymbol{K} \cdot \boldsymbol{J}_c \cdot \dot{\boldsymbol{q}} \tag{A8}$$

If the joint velocity vector is chosen as

$$\dot{\boldsymbol{q}} = \boldsymbol{J}_c^T \cdot \boldsymbol{K} \cdot \boldsymbol{e}_c \tag{A9}$$

the function

$$\dot{\boldsymbol{V}} = -\boldsymbol{e}_c^T \cdot \boldsymbol{K} \cdot \boldsymbol{J}_c \cdot \boldsymbol{J}_c^T \cdot \boldsymbol{K} \cdot \boldsymbol{e}_c \tag{A10}$$

is negative definite, and thus the error asymptotically converges to zero.

Because the aim of the experiment was to control the yaw angle of the robot base, the following substitutions can be made:

$$\boldsymbol{y} = \varphi \tag{A11}$$

$$\boldsymbol{J}_c = \boldsymbol{F}_6 \tag{A12}$$

which yields Eq. (26).

Acknowledgments

The experiments were carried out during the Sixth Student Parabolic Flight Campaign sponsored by the ESA. The realization of the free-flying robot prototype was founded by the Italian Space Agency and CISAS–University of Padova. The authors thank Claudio Bombardelli of the University of Padova and Levent Burak Kara of the Carnegie Mellon University for their valuable advice.

References

[1]Gawdiak, Y., Bradshaw, J. M., Williams, B., and Thomas, H., "R2D2 in a Softball: The Personal Satellite Assistant," *Proceedings of the ACM Conference on Intelligent User Interfaces (IUI)*, Association for Computing Machinery, New Orleans, LA, 2000, pp. 125–128.

[2]Gray, G. G. H., "Commercialization of Canada's Mobile Servicing System Technology," AIAA Paper 94-4462, Sept. 1994.

[3]Culbertson, F., "Operating the ISS: A World-Wide Team Effort," AIAA Paper 2003-2701, July 2003.

[4]Marzwell, N. J., "Revolutionary Concepts Through Evolutionary Progress—Modular Robotics in Space Exploration," AIAA Paper 2001-4635, Aug. 2001.

[5]Basu, S., Mast, T., and Miyata, G., "A Proposed Autonomously Assembled Space Telescope (AAST)," AIAA Paper 2003-6369, Sept. 2003.

[6]Rusconi, A., Finotello, R., Borghi, G., Mugnuolo, R., Olivieri, A., and Pasquali, F., "Europa (External Use of Robotics for Payloads Automation)," AIAA Paper 2001-5009, Oct. 2001.

[7]Nenchev, D. N., and Yoshida, K., "Singularity-Consistent Teleoperation Techniques for Redundant Free-Flying Robots," AIAA Paper 99-4305, Aug. 1999.

[8]Pond, B., and Sharf, I., "Experimental Evaluation of Flexible Manipulator Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 4, 2001, pp. 834–843.

[9]Pond, B., Van Vliet, J., and Sharf, I., "Prediction Tools for Active Damping and Motion Planning of Flexible Manipulators," *Journal of Guidance, Control, and Dynamics*, Vol. 26, No. 2, 2003, pp. 267–272.

[10]Polidan, R. S., "Hubble Space Telescope Overview," AIAA Paper 91-402, Jan. 1991.

[11]Lumpkin, F. E., III, Stuart, P. C., and LeBeau, G. G., "The Airlock Depressurization Plume Anomaly on the STS-82 Hubble Servicing Mission," AIAA Paper 2000-462, Jan. 2000.

[12]Akin, D., Roberts, B., Pilotte, K., and Baker, M., "Robotic Augmentation of EVA for Hubble Space Telescope Servicing," AIAA Paper 2003-6274, Sept. 2003.

[13]Umetani, Y., and Yoshida, K., "Resolved Motion Rate Control of Space Manipulators with Generalized Jacobian Matrix," *IEEE Transactions on Robotics and Automation*, Vol. 5, No. 3, 1989, pp. 303–314.

[14]Nakamura, Y., and Mukherjee, R., "Nonholonomic Path Planning of Space Robots via Bi-Directional Approach," *Transaction on Robotics and Automation*, Vol. 7, No. 4, 1991, pp. 500–514.

[15]Dubowsky, S., and Papadopoulos, E., "The Kinematics, Dynamics, and Control of Free-Flying and Free-Floating Space Robotic Systems," *Transactions on Robotics and Automation, Special Issue on Space Robotics*, Vol. 9, No. 5, 1993, pp. 531–543.

[16]Iwata, T., Toda, Y., and Machida, K., "Dynamic Control of Free Flying Robot for Capturing Maneuvers," AIAA Paper 91-2824, Aug. 1991.

[17]Fujii, H., Murayama, T., Nakajima, K., and Anazawa, S., "Capture Control for Manipulator Arm of Free-Flying Space Robot," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, Pt. 2, AIAA, Washington, DC, 1990, pp. 1056–1060.

[18]Torres, M. A., and Dubowsky, S., "Minimizing Spacecraft Attitude Disturbances in Space Manipulator Systems," *Journal of Guidance, Control, and Dynamics*, Vol. 15, No. 4, 1992, pp. 1010–1017.

[19]Quinn, R. D., Chen, J. L., and Lawrence, C., "Base Reaction Control for Space-Based Robots Operating in Microgravity Environment," *Journal of Guidance, Control, and Dynamics*, Vol. 17, No. 2, 1994, pp. 263–270.

[20]Mukherjee, R., and Zurowski, M., "Reorientation of a Structure in Space Using a Three-Link Rigid Manipulator," *Journal of Guidance, Control, and Dynamics*, Vol. 17, No. 4, 1994, pp. 840–847.

[21]Okubo, H., Nagano, N., Komatsu, N., and Tsumura, T., "Path Planning for Space Manipulators to Reduce Attitude Disturbances," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 3, 1994, pp. 609–611.

[22]Nakamura, Y., and Suzuki, T., "Planning Spiral Motions of Nonholonomic Free-Flying Space Robots," *Journal of Spacecraft and Rockets*, Vol. 34, No. 1, 1997, pp. 137–143.

[23]Senda, K., Nagaoka, H., and Murotsu, Y., "Adaptive Control of Free-Flying Space Robot with Position/Attitude Control System," *Journal of Guidance, Control, and Dynamics*, Vol. 22, No. 3, 1999, pp. 486–501.

[24]Caccavale, F., and Siciliano, B., "Kinematic Control of Redundant Free-Floating Robotic Systems," *Advanced Robotics*, Vol. 15, No. 4, 2001, pp. 429–448.

[25]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.

[26]Carignan, C., and Akin, D., "Using Robots for Astronaut Training," *Control Systems Magazine*, Vol. 23, No. 2, 2003, pp. 46–59.

[27]Koningstein, R., and Cannon, R. H., "Experiments with Model-Simplified Computed-Torque Manipulator Controllers for Free-Flying Robots," *Journal of Guidance, Control, and Dynamics*, Vol. 18, No. 6, 1995, pp. 1387–1391.

[28]Robertson, A., Inalhan, G., and How, J. P., "Spacecraft Formation Flying Control Design for the Orion Mission," *Guidance, Navigation, and Control Conference and Exhibit*, Vol. 3, AIAA, Reston, VA, 1999, pp. 1562–1575.

[29]Nahon, M., Damaren, C., Bergen, A., and Goncalves, J., "A Test Facility for Multi-Armed Space-Based Manipulators," *Canadian Aeronautics and Space Journal*, Vol. 41, No. 4, 1995, pp. 150–162.

[30]Van Vliet, J., and Sharf, I., "Development of a Planar Macro-Micro Manipulator Facility—From Design Through Model Validation," *Canadian Aeronautics and Space Journal*, Vol. 44, No. 1, 1998, pp. 40–50.

[31]Yoshida, K., "Experimental Study on the Dynamics and Control of a Space Robot with Experimental Free-Floating Robot Satellite (EFFORTS) Simulators," *Advanced Robotics*, Vol. 9, No. 6, 1995, pp. 583–602.

[32]Matunaga, S., Yoshihara, K., Takahashi, T., Tsurumi, S., and Ui, K., "Ground Experiment System for Dual-Manipulator-Based Capture of Damaged Satellites," *Proceedings of the IEEE/RSJ*, International Conf. on Intelligent Robots and Systems, Kanagawa Univ., Takamatsu, Japan, 2000, pp. 1847–1852.

[33]Meyer, J. L., Harrington, W. B., Agrawal, B. N., and Song, G., "Application of Piezoceramics to Vibration Suppression of a Spacecraft Flexible Appendage," AIAA Paper 96-3761, July 1996.

[34]Spencer, M. G., "Development of a Servicing Satellite Simulator," AIAA Paper 2001-4529, Aug. 2001.

[35]Miller, D., Saenz-Otero, A., Wertz, J., Chen, A., Berkowski, G., Brodel, C., Carlson, S., Carpenter, D., Chen, S., Cheng, S., Feller, D., Jackson, S., Pitts, B., Perez, F., Szuminski, J., and Sell, S., "SPHERES: A Testbed for Long Duration Satellite Formation Flying In Micro-Gravity Conditions," American Astronautical Society, Paper 00-110, Jan. 2000.

[36]Choset, H., and Kortenkamp, D., "Path Planning and Control for Free-Flying Inspection Robot in Space," *Journal of Aerospace Engineering*, Vol. 12, No. 2, 2001, pp. 74–81.

[37]Bettanini, C. C., Aboudan, A., Francesconi, A., Marchesi, M., Menon, C., and Angrilli, F., "Improving the Free-Floater Space Robot Simulator for Intervention Missions," *Proceeding of the 7th International Symposium on Artificial Intelligence, Robotics and Automation in Space* [CD-ROM], 2003.

[38]Fujii, H. A., Uchiyama, K., and Yoneoka, H., "Ground-Based Simulation of Space Manipulators Using Test Bed with Suspension System," *Journal of Guidance, Control, and Dynamics*, Vol. 19, No. 5, 1996, pp. 985–991.

[39]Reaves, M. C., Chew, M. S., Juang, J. N., and Steven, H., "Dynamics and Control of a Large Displacement Suspension System for Ground Testing of Flexible Space Structures," AIAA Paper 92-1178, Feb. 1992.

[40]Fischer, A., and Pellegrino, S., "Interaction Between Gravity Compensation Suspension System and Deployable Structure," AIAA Paper 98-1835, April 1998.

[41]Hasselman, T. K., and Anderson, M. C., "Development of a Large Amplitude 3D Microgravity Suspension System," AIAA Paper 97-1566, April 1997.

[42]Uchiyama, K. F., Hironori, A., and Yoneoka, H., "Experiments on Dynamic Behavior of Space Manipulator Using Suspension System," AIAA Paper 95-3334, Aug. 1995.

[43]Woodard, S. E., and Housner, J. M., "The Nonlinear Behavior of a Passive Zero-Spring-Rate Suspension System," AIAA Paper 88-2316, April 1988.

[44]Kienholz, D. A., "A Suspension System for Simulating Unconstrained Boundary Conditions," *Proceedings of the 12th International Modal Analysis Conference*, CSA, Palo Alto, CA, 1994.

[45]Idle, M. K., and Cobb, R. G., "Use of a Zero-Gravity Suspension System for Testing a Vibration Isolation System," *17th Aerospace Testing Seminar*, The Aerospace Corp., Manhattan Beach, CA, 1997, pp. 79–84.

[46]Parrish, J. C., Sullivan, B. R., and Roberts, B. J., "Planning for the Ranger Telerobotic Shuttle Experiment On-Orbit Operations," AIAA Paper 2000-5291, Sept. 2000.

[47]Akin, D. L., Ranniger, C. U., and DeLevie, M., "Development and Testing of an EVA Simulation System for Neutral Buoyancy Operations," AIAA Paper 96-4223, Sept. 1996.

[48]Spofford, J. R., and Akin, D. L., "Redundancy Control of a Free-Flying Telerobot," *Journal of Guidance, Control, and Dynamics*, Vol. 13, No. 3, 1990, pp. 515–523.

[49]Heard, W. L., Jr., Bush, H. G., and Watson, J. J., "Astronaut/EVA Construction of Space Station," *Issues of the International Space Station, Conference*, AIAA, Washington, DC, 1988, pp. 39–46.

[50]Heard, W. L., Jr., Watson, J. J., Ross, J. L., Spring, S. C., and Cleave, M. L., "Results of the ACCESS Space Construction Shuttle Flight Experiment," *Space Systems Technology Conference*, AIAA, New York, 1986, pp. 118–125.

[51]Stokes, J. W., Engler, E. E., and Agan, W. E., "Neutral Buoyancy Test Results of a Deployable Space Beam," AIAA Paper 81-437, Feb. 1981.

[52]Spofford, J. R., and Akin, D. L., "Results of the M.I.T. Beam Assembly Teleoperator and Integrated Control Station," *Proceedings of Guidance and Control Conference*, AIAA, New York, 1984, pp. 351–359.

[53]Yoshida, K., "Engineering Test Satellite VII Flight Experiments for Space Robot Dynamics and Control: Theories on Laboratory Test Beds Ten Years Ago, Now in Orbit," *International Journal of Robotics Research*, Vol. 22, No. 5, 2003, pp. 321–335.

[54]Sciavicco, L., and Siciliano, B., *Modelling and Control of Robot Manipulators*, 2nd ed., Springer-Verlag, London, 2000, Chap. 3.

[55]Lewis, F. L., Abdallah, C. T., and Dawson, M. D., *Control of Robot Manipulators*, Macmillan, New York, 1993.

[56]"Packet Telemetry Standard," ESA, PSS 04-106, Issue 1, Jan. 1988.

[57]Albus, J. S., McCain, H. G., and Lumia, R., "NASA/NBS Standard Reference Model for Telerobot Control System Architecture (NASREM)," National Inst. of Standards and Technology, Tech. Note 1235, Gaithersburg, MD, July 1987.