

Dynamics & Control of Space Free-Flyers with Multiple Manipulators

Evangelos Papadopoulos, S. Ali A. Moosavian

Department of Mechanical Engineering & Centre for Intelligent Machines

McGill University

Montreal, PQ, Canada H3A 2A7

ABSTRACT

This paper studies the motion control of a multiple manipulator free-flying space robot chasing a passive object in near proximity. Free-flyer kinematics are developed using a minimum set of body-fixed barycentric vectors. Using a general and a quasi-coordinate Lagrangian formulation, equations of motion for model-based controllers are derived. Two model-based and one transposed Jacobian control algorithms are developed that allow coordinated tracking control of the manipulators and the spacecraft. In particular, an Euler parameter model-based control algorithm is presented that overcomes the non-physical singularities due to Euler angle representation of attitude. To ensure smooth operation, and reduce disturbances on the spacecraft and on the object just before grasping, appropriate trajectories for the motion of spacecraft/manipulators are planned. The performance of model-based algorithms is compared, by simulation, to that of a transposed Jacobian algorithm. Results show that due to the complexity of space robotic systems, a drastic deterioration in the performance of model-based algorithms in the presence of model uncertainties results. In such cases, a simple transposed Jacobian algorithm yields comparable results with much reduced computational burden, an issue which is very important in space.

I. INTRODUCTION

As space commercialization materializes, space structures and satellites will proliferate. Extending the life of such systems, and therefore reducing the associated costs, will require extensive inspection, assembly and maintenance capabilities in orbit. Astronaut Extra Vehicular Activities (EVA) can be valuable in meeting these requirements. However, the cost of human life support facilities, the limited time available for astronaut EVA, and the high risks involved, make space robotic devices candidate astronaut assistants or alternatives. To increase the mobility of such devices, free-flying space robotic systems in which manipulators are mounted on a thruster-equipped spacecraft, have been proposed [1-3].

Control schemes that allow the spacecraft to be uncontrolled (free-floating mode operation), have been studied to eliminate the use of reaction jet fuel [4-8]. Control schemes for the capture of targets within a free-floating system's workspace were presented in [9,10]. However, the workspace of free-floating systems is restricted by their inability to have their system Center of Mass (CM) translated, and by the existence of workspace dynamic singularities [11]. To achieve an unlimited workspace, a control scheme that treats a free-flyer as a redundant manipulator, and is based on a pseudo-inverse Jacobian controller has been proposed [12]. In this scheme, an operator will command the manipulator's end-effector only; the spacecraft position and orientation, and the manipulator configuration will change in an uncontrolled way. In another study, a coordinated controller was designed so that both an end-effector and the spacecraft can be controlled [13]. This control scheme allows commanding a desirable manipulator configuration, and planning of a system's motion.

In this paper, the dynamics of multi-manipulator free-flying space robots are developed. The system CM position is used to represent the system translational Degrees-of-Freedom (DOF). The kinematic and dynamic quantities are expressed using a set of body-fixed barycentric vectors. As studied in [14, 15], this results in decoupling the total linear and angular motion from the rest of the equations, when no external forces/torques are applied on the system. Next, two model-based control algorithms, based on an Euler angle and an Euler parameter description of the orientation, and a transposed Jacobian control algorithm are developed. These algorithms permit control of both the

spacecraft and its appendages in their task space. Euler angle model-based control algorithm (MB1) presents the inconvenience of representational singularities, while Euler parameter model-based control algorithm (MB2) overcomes these non-physical singularities. To ensure smooth operation, and reduce disturbances on the spacecraft and also on the object just before grasping, appropriate trajectories for the spacecraft/manipulators motion are planned which lead to capture of moving objects in space. These trajectories take into account the relative target motion, and thruster/actuator saturation limits. The developed control laws are evaluated using a three manipulator/appendages free-flyer example. It is found that if dynamic properties are accurately known, model-based controllers provide good tracking, but are computationally expensive. On the other hand, the simple Jacobian-based algorithm, when used with appropriate gains, provides an acceptable and computationally inexpensive controller.

II. DYNAMICS MODELLING

In this section, the equations of motion of a rigid multiple arm free-flying space robotic system, are obtained. The travel of the system is assumed of relatively short length and duration and therefore dynamical effects due to orbital mechanics are neglected. The motion of the system CM is used to describe its translation with respect to an in-orbit inertial frame of reference (XYZ), and all the kinematic and dynamic quantities are written in terms of a set of body-fixed barycentric vectors. The body 0 in Figure 1 represents the spacecraft of the free-flyer, which is connected to n manipulators or appendages, each with N_m links. Manipulator joints are revolute and have a single DOF. The joint angles and rates are represented by $N \times 1$ column vectors $\boldsymbol{\theta}$, and $\dot{\boldsymbol{\theta}}$. The total DOF of the system are $N = K + 6$, where $K = \sum_{m=1}^n N_m$.

The inertial position of a point P, \mathbf{R}_p , can be written as

$$\mathbf{R}_p = \mathbf{R}_{CM} + \mathbf{r}_p \quad (1)$$

$$\mathbf{r}_p = \mathbf{r}_{C_i} + \mathbf{r}_{p/C_i} \quad (2)$$

where \mathbf{r}_p is the position vector of P with respect to the system CM, \mathbf{R}_{CM} is the inertial position of the system CM, C_i is the CM of the i-th body, \mathbf{r}_{C_i} is its position vector with respect to the system CM, and \mathbf{r}_{P/C_i} is the position vector of P with respect to C_i . Vectors \mathbf{r}_{C_i} can be written as follows

$$\mathbf{r}_{C_0} = \tilde{\mathbf{e}}_0 + \sum_{m=1}^n \sum_{k=1}^{N_m} \tilde{\mathbf{l}}_k^{(m)} \quad (3)$$

$$\mathbf{r}_{C_i}^{(m)} = \tilde{\mathbf{r}}_0^{(m)} + \sum_{\substack{j=1 \\ j \neq m}}^n \sum_{k=1}^{N_j} \tilde{\mathbf{l}}_k^{(j)} + \sum_{k=1}^{N_m} \tilde{\mathbf{v}}_{ki}^{(m)} \quad \begin{cases} m = 1, \dots, n \\ i = 1, \dots, N_m \end{cases} \quad (4)$$

where the superscript ‘‘m’’ corresponds to the m-th manipulator, the subscript ‘‘i’’ refers to the i-th body of that manipulator, and $(\tilde{\cdot})$ denotes body-fixed barycentric vectors defined in Appendix A.

To obtain the inertial velocity of point P, Equation (1) is differentiated and yields

$$\dot{\mathbf{R}}_p = \dot{\mathbf{R}}_{CM} + \dot{\mathbf{r}}_{C_i} + \boldsymbol{\omega}_i \times \mathbf{r}_{P/C_i} \quad (5)$$

Differentiation of Eqs. (3) and (4) yields

$$\dot{\mathbf{r}}_{C_0} = \boldsymbol{\omega}_0 \times \tilde{\mathbf{e}}_0 + \sum_{m=1}^n \sum_{k=1}^{N_m} \boldsymbol{\omega}_k^{(m)} \times \tilde{\mathbf{l}}_k^{(m)} \quad (6a)$$

$$\dot{\mathbf{r}}_{C_i}^{(m)} = \boldsymbol{\omega}_0 \times \tilde{\mathbf{r}}_0^{(m)} + \sum_{\substack{j=1 \\ j \neq m}}^n \sum_{k=1}^{N_j} \boldsymbol{\omega}_k^{(j)} \times \tilde{\mathbf{l}}_k^{(j)} + \sum_{k=1}^{N_m} \boldsymbol{\omega}_k^{(m)} \times \tilde{\mathbf{v}}_{ki}^{(m)} \quad \begin{cases} m = 1, \dots, n \\ i = 1, \dots, N_m \end{cases} \quad (6b)$$

where \boldsymbol{w} 's are angular velocities of individual bodies, which for single DOF joints are written as

$$\boldsymbol{\omega}_i^{(m)} = \boldsymbol{\omega}_0 + \sum_{k=1}^i \dot{\mathbf{q}}_k^{(m)} \mathbf{z}_k^{(m)} \quad \begin{cases} m = 1, \dots, n \\ i = 1, \dots, N_m \end{cases} \quad (7)$$

the $\mathbf{z}_k^{(m)}$ is a unit vector along axis of rotation of the k-th joint of the m-th manipulator, and $\mathbf{q}_k^{(m)}$ is the corresponding joint angle. To obtain scalar equations, appropriate transformation matrices for each term must be employed.

The kinetic energy of the system, T , is found using Eqs. (5-7) as

$$T = \int_M \dot{\mathbf{R}}_p \cdot \dot{\mathbf{R}}_p dM = T_0 + T_1 \quad (8a)$$

with

$$T_0 = \frac{1}{2} M (\dot{\mathbf{R}}_{CM} \cdot \dot{\mathbf{R}}_{CM}) \quad (8b)$$

$$T_1 = \frac{1}{2} \{ m_0 \dot{\mathbf{r}}_{C_0} \cdot \dot{\mathbf{r}}_{C_0} + \boldsymbol{\omega}_0 \cdot \mathbf{I}_0 \cdot \boldsymbol{\omega}_0 + \sum_{m=1}^n \sum_{i=1}^{N_m} (m_i^{(m)} \dot{\mathbf{r}}_{C_i}^{(m)} \cdot \dot{\mathbf{r}}_{C_i}^{(m)} + \boldsymbol{\omega}_i^{(m)} \cdot \mathbf{I}_i^{(m)} \cdot \boldsymbol{\omega}_i^{(m)}) \} \quad (8c)$$

where $\mathbf{I}_i^{(m)}$ is the inertia dyadic of the i -th body of the m -th manipulator with respect to its CM. Using Eqs. (6-8), the kinetic energy of the system can be written as

$$T = \frac{1}{2} \mathbf{v}^T \mathbf{H}(\boldsymbol{\theta}) \mathbf{v} \quad (9)$$

where $\mathbf{v} = (\dot{\mathbf{R}}_{CM}^T, {}^0\boldsymbol{\omega}_0^T, \dot{\boldsymbol{\theta}}^T)^T$ is the vector of generalized velocities, and \mathbf{H} is an $N \times N$ positive definite mass matrix. The vector ${}^0\boldsymbol{\omega}_0$ is the spacecraft angular velocity expressed in its frame of reference. For the free-flyer, the microgravity effects compared to control forces are very small and hence they are neglected; the system potential energy is taken equal to zero. Using the expression for the kinetic energy given by Equation (9), and a quasi-Lagrangian approach [16], a set of dynamical equations is obtained in the form

$$\mathbf{H}(\boldsymbol{\theta}) \dot{\mathbf{v}} + \mathbf{C}({}^0\boldsymbol{\omega}_0, \boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = \mathbf{Q} \quad (10)$$

where \mathbf{C} contains all the nonlinear velocity terms, and \mathbf{Q} is the vector of generalized forces given by

$$\mathbf{Q} = \begin{Bmatrix} \mathbf{0}_{6 \times 1} \\ \boldsymbol{\tau}_{K \times 1} \end{Bmatrix} + \sum_{p=1}^{i_f} \mathbf{J}_{0,p}^T \mathbf{F}_{0,p} + \sum_{m=1}^n \sum_{i=1}^{N_m} \sum_{p=1}^{i_f} \mathbf{J}_{i,p}^{(m)T} \mathbf{F}_{i,p}^{(m)} \quad (11)$$

The vector \mathbf{t} contains joint torques, $\mathbf{F}_{0,p}$ is the p -th external force/moment applied on the spacecraft, $\mathbf{F}_{i,p}^{(m)}$ is the p -th external force/moment applied on the i -th body of the m -th manipulator, i_f is the number of applied forces/moments on the corresponding body, and $\mathbf{J}_{i,p}^{(m)}$ is a Jacobian matrix corresponding to the point of force/moment application as

$$\mathbf{J}_{i,p}^{(m)} = \begin{bmatrix} \mathbf{1}_{3 \times 3} & \mathbf{J}_1 & \mathbf{J}_2 \\ \mathbf{0}_{3 \times 3} & \mathbf{1}_{3 \times 3} & \mathbf{J}_3 \end{bmatrix}_{6 \times N} \quad (12)$$

and, similarly, $\mathbf{J}_{0,p}$ is defined for the one corresponding to the spacecraft, where

$$\mathbf{J}_1 = - \left[\mathbf{T}_0 \mathbf{v}_{0i,p} + \sum_{k=1}^n \sum_{j=1}^{N_k} [\mathbf{T}_j^{(k)} \mathbf{v}_{ji,p}^{(k)}] \right]^\times \quad (13a)$$

$$\mathbf{J}_2 = - \sum_{k=1}^n \sum_{j=1}^{N_k} [\mathbf{T}_j^{(k)} \mathbf{v}_{ji,p}^{(k)}]^\times \mathbf{E}_j \quad (13b)$$

$$\mathbf{J}_3 = \mathbf{E}_i^{(m)} \quad (13c)$$

The \mathbf{T}_0 and $\mathbf{T}_j^{(k)}$ are rotation matrices between body-fixed frames and the inertial frame, $[\cdot]^\times$ is the *cross product operator*, and

$$\mathbf{v}_{ji,p}^{(k)} = \tilde{\mathbf{v}}_{ji}^{(k)} + \delta_{ji} \mathbf{r}_{p|c_i} \quad (13d)$$

$$\mathbf{E}_j^{(k)} = \left[\mathbf{0}_{3 \times a} \quad \mathbf{T}_1^{(k)1} \mathbf{z}_1^{(k)} \quad \dots \quad \mathbf{T}_j^{(k)j} \mathbf{z}_j^{(k)} \quad \mathbf{0}_{3 \times b} \right]_{3 \times K} \quad (13e)$$

δ_{ji} is *Kronecker delta*, $a = \sum_{l=1}^{k-1} N_l$, and ${}^j \mathbf{z}_j^{(k)}$ is a unit vector along axis of rotation of the j -th joint of the k -th manipulator expressed in its own body-fixed frame. Equation (11) can be rearranged, so that actuator forces/torques are displayed explicitly. If the only external forces that act on the system are the net force ${}^0 \mathbf{f}_s$ and torque ${}^0 \mathbf{n}_s$ applied on the spacecraft, then \mathbf{Q} can be written as

$$\mathbf{Q} = \mathbf{J}_Q^T \begin{Bmatrix} {}^0 \mathbf{f}_s \\ {}^0 \mathbf{n}_s \\ \boldsymbol{\tau}_{K \times 1} \end{Bmatrix} \quad (14)$$

where \mathbf{J}_Q is an $N \times N$ Jacobian matrix which can be written based on the above definition for $\mathbf{J}_{i,p}^{(m)}$. For a well designed system, \mathbf{J}_Q is nonsingular, that is any required \mathbf{Q} vector can be produced by the system's actuators.

The form of equations in (10) is useful in designing an Euler-parameter based control algorithm, as discussed in more detail in Section III. For control reasons, it is also beneficial to obtain the equations of motion using as the vector of generalized coordinates $\mathbf{q} = (\mathbf{R}_{CM}^T, \boldsymbol{\delta}^T, \boldsymbol{\theta}^T)^T$, where $\boldsymbol{\delta}$ is a set of Euler angles that describe the orientation of the spacecraft. The spacecraft angular velocity can be expressed in terms of the Euler rates as [16]

$${}^0\boldsymbol{\omega}_0 = \mathbf{S}_0(\boldsymbol{\delta})\dot{\boldsymbol{\delta}} \quad (15)$$

where $\mathbf{S}_0(\boldsymbol{\delta})$ is a 3×3 matrix, function of the attitude $\boldsymbol{\delta}$. Then the kinetic energy is written as

$$T = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{H}_{\boldsymbol{\delta}}(\boldsymbol{\delta}, \boldsymbol{\theta}) \dot{\mathbf{q}} \quad (16)$$

Applying Lagrange's equations to Equation (16)

$$\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{q}_i} \right) - \left(\frac{\partial T}{\partial q_i} \right) = Q_{\boldsymbol{\delta}, i} \quad i = 1, \dots, N \quad (17)$$

results in the equations of motion

$$\mathbf{H}_{\boldsymbol{\delta}}(\boldsymbol{\delta}, \boldsymbol{\theta}) \ddot{\mathbf{q}} + \mathbf{C}_{\boldsymbol{\delta}}(\boldsymbol{\delta}, \dot{\boldsymbol{\delta}}, \boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = \mathbf{Q}_{\boldsymbol{\delta}} \quad (18)$$

The vector $\mathbf{Q}_{\boldsymbol{\delta}}$ is related to \mathbf{Q} by a simple transformation as

$$\mathbf{Q}_{\boldsymbol{\delta}} = \begin{bmatrix} \mathbf{1}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times K} \\ \mathbf{0}_{3 \times 3} & \mathbf{S}_0^T & \mathbf{0}_{3 \times K} \\ \mathbf{0}_{K \times 3} & \mathbf{0}_{K \times 3} & \mathbf{1}_{K \times K} \end{bmatrix} \mathbf{Q} \quad (19)$$

The equations of motion derived in this section will be used in designing control algorithms, the topic of the next section.

III. CONTROL DESIGN

Controlling a free-flying space robot requires definition of the controlled system outputs, and design of a control law which can guarantee that these outputs will track asymptotically desired trajectories. A designer is faced with many options for the controlled outputs. These include joint space variables, Cartesian (task) space variables, and others. The various orientation representations further increase the available options. In this paper, the focus is in controlling the Cartesian position/orientation of the spacecraft and the end-effectors of its manipulators. The coordination between the spacecraft motion and several end-effectors is investigated under different control laws. Two model-based control algorithms, based on an Euler angle and on an Euler parameter description of the orientation, and a

transposed Jacobian control algorithm are developed. Euler angle model-based control algorithm (MB1) presents the inconvenience of representational singularities, i.e. inversion of relation between angular velocity and Euler rates, Equation (15), is not possible at some orientations. Such inversion is required when actuator forces/torques \mathbf{Q} are to be calculated from the the control command \mathbf{Q}_δ , using Equation (19). As the system gets closer to these representational singularities, the orientational error grows, and if it passes through them, the control algorithm fails computationally, see Fig. 2. To avoid this situation, a different set of Euler angles must be used at such points.

It is known that such singularities will appear whenever a three-parameter description of the orientation is employed. However, a great improvement can occur if a singularity appears at some *attitude error* and not at some *attitude*. An Euler parameter model-based control algorithm that achieves this condition has been presented for the attitude control of a single rigid body [17]. This algorithm is adapted here to the coordinated control of a multiple arm free-flyer robot, and is presented as the second model-based control algorithm (MB2). The simpler transposed Jacobian (TJ) controller can be employed, if computational power is limited. In the following, these three algorithms are developed and compared.

1. Model-based Control Design Using Euler Angles (MB1)

Step 1. The equations of motion (18) are rewritten in terms of the output coordinates $\hat{\mathbf{q}}$ given by

$$\hat{\mathbf{q}} = [\mathbf{R}_0^T, \boldsymbol{\delta}^T, \mathbf{x}_E^{(1)T}, \boldsymbol{\delta}_E^{(1)T}, \dots, \mathbf{x}_E^{(n)T}, \boldsymbol{\delta}_E^{(n)T}]^T \quad (20)$$

where $\mathbf{x}_E^{(m)}$ and $\boldsymbol{\delta}_E^{(m)}$ correspond to the m-th end-effector position and orientation. If all manipulators have six DOF, then a system of n manipulators will have $6n+6$ DOF, and $\hat{\mathbf{q}}$ will be a $6n+6$ vector. The output velocities $\dot{\hat{\mathbf{q}}}$ are obtained from the generalized velocities $\dot{\mathbf{q}}$ using a square Jacobian $\mathbf{J}_{\hat{\mathbf{q}}}$

$$\dot{\hat{\mathbf{q}}} = \mathbf{J}_{\hat{\mathbf{q}}}(\boldsymbol{\delta}, \boldsymbol{\theta}) \dot{\mathbf{q}} \quad (21)$$

The Jacobian $\mathbf{J}_{\hat{q}}$ is not singular, except when a manipulator is at a singular configuration, or at a (non-physical) representation singularity due to the use of Euler angles. The latter can be avoided by switching to a different set of Euler angles. The equations of motion are then

$$\hat{\mathbf{H}}_{\delta} \ddot{\hat{\mathbf{q}}} + \hat{\mathbf{C}}_{\delta} = \mathbf{J}_{\hat{q}}^{-T} \mathbf{Q}_{\delta} \quad (22)$$

where $\hat{\mathbf{C}}_{\delta}$ and $\hat{\mathbf{H}}_{\delta}$ are given by

$$\hat{\mathbf{H}}_{\delta} = \mathbf{J}_{\hat{q}}^{-T} \mathbf{H}_{\delta} \mathbf{J}_{\hat{q}}^{-1} \quad (23a)$$

$$\hat{\mathbf{C}}_{\delta} = \mathbf{J}_{\hat{q}}^{-T} \mathbf{C} - \hat{\mathbf{H}} \dot{\mathbf{J}}_{\hat{q}} \dot{\hat{\mathbf{q}}} \quad (23b)$$

The new inertia matrix, $\hat{\mathbf{H}}_{\delta}$, is positive definite if $\mathbf{J}_{\hat{q}}$ is nonsingular.

Step 2. The following model-based control law is used

$$\mathbf{Q}_{\delta} = \mathbf{J}_{\hat{q}}^T \{ \hat{\mathbf{H}}_{\delta} \mathbf{u} + \hat{\mathbf{C}}_{\delta} \} \quad (24)$$

where it is assumed that the system geometric and mass properties are known, and where $\mathbf{u} = [\mathbf{u}_{\mathbf{R}}^T, \mathbf{u}_{\delta}^T, \mathbf{u}_{\mathbf{x}}^{(1)T}, \mathbf{u}_{\delta}^{(1)T}, \dots, \mathbf{u}_{\mathbf{x}}^{(n)T}, \mathbf{u}_{\delta}^{(n)T}]$ is an auxiliary control signal. This control law, similar to the Operational Space controller [18], linearizes and decouples the system equations to a set of second order differential equations

$$\ddot{\hat{\mathbf{q}}} = \mathbf{u} \quad (25)$$

Step 3. If \mathbf{u} is computed such that

$$\mathbf{u} = \mathbf{K}_p \mathbf{e} + \mathbf{K}_d \dot{\mathbf{e}} + \ddot{\hat{\mathbf{q}}}_{des} \quad (26)$$

where \mathbf{K}_p , and \mathbf{K}_d are positive definite diagonal matrices, and \mathbf{e} is the tracking error defined as

$$\mathbf{e} = \hat{\mathbf{q}}_{des} - \hat{\mathbf{q}} \quad (27)$$

then, the control law given by Equation (24) guarantees asymptotic convergence of the tracking error \mathbf{e} . The desired trajectory, $\hat{\mathbf{q}}_{des}$, is provided by a trajectory planner, while $\hat{\mathbf{q}}$ can be obtained from inertial measurements of the position and orientation of the spacecraft and of the end-effectors. If no such measurements are available, the error \mathbf{e} can be estimated by integrating the equations of motion in

real time, but then errors due to model uncertainties will be introduced. A mixed strategy can also be employed, e.g. inertial feedback may be available during a critical or terminal phase of a maneuver.

2. Transposed Jacobian Control Design (TJ)

If high enough gains are used, the simpler transposed Jacobian controller can be employed, [19]

$$\mathbf{Q}_\delta = \mathbf{J}_q^T \{ \mathbf{K}_p \mathbf{e} + \mathbf{K}_d \dot{\mathbf{e}} \} \quad (28)$$

This algorithm is quite simple to use. Its action can be understood by imagining generalized springs and dampers connected between the bodies under control and the desired trajectories; the stiffer the gains are, the better the tracking should be. Note that if a physical singularity is encountered, the controller given by Equation (28) will result in errors but will not fail computationally.

Table I reveals the efficiency of this algorithm, compared to the MB1 algorithm, in terms of the required computational operations, i.e. multiplication and summations required to follow the algorithm, for an N DOF system. It is assumed that the inverse of the Jacobian matrix and its time derivative, which are required for implementing MB algorithms, are available symbolically. Even with this assumption in favor of the model-based algorithm, implementation of TJ control significantly reduces the amount of required computations.

Table I. Comparison of the required computational operations.

Algorithm	Multiplications	Additions
TJ	3 N ²	3 N ² - 2 N
MB1	2 N ³ + 7 N ²	2 N ³ + 5 N ² - 4 N

3. Model-based Control using Euler Parameters (MB2)

Step 1. The equations of motion (10) are rewritten in terms of the output velocities $\hat{\mathbf{v}}$ given by

$$\hat{\mathbf{v}} = [\dot{\mathbf{R}}_0^T, {}^0\boldsymbol{\omega}_0^T, \dot{\mathbf{x}}_E^{(1)T}, {}^1\boldsymbol{\omega}_E^{(1)T}, \dots, \dot{\mathbf{x}}_E^{(n)T}, {}^n\boldsymbol{\omega}_E^{(n)T}]^T \quad (29)$$

where $\dot{\mathbf{x}}_E^{(m)}$ and ${}^m\boldsymbol{\omega}_E^{(m)T}$ are the m-th end-effector linear and angular inertial velocity, expressed in the inertial and m-th body frame, respectively. If all manipulators have six DOF, then a system of n

manipulators will have $6n+6$ DOF, and $\hat{\mathbf{v}}$ will be a $6n+6$ vector. The output velocities $\hat{\mathbf{v}}$ are obtained from the generalized velocities \mathbf{v} by a Jacobian $\mathbf{J}_{\hat{\mathbf{v}}}$

$$\hat{\mathbf{v}} = \mathbf{J}_{\hat{\mathbf{v}}}(\boldsymbol{\varepsilon}, n, \boldsymbol{\theta}) \mathbf{v} \quad (30)$$

where $\boldsymbol{\varepsilon}$ and n are the vector and scalar part of Euler parameters, representing the orientation of the spacecraft [20]. The equations of motion are then

$$\hat{\mathbf{H}} \dot{\hat{\mathbf{v}}} + \hat{\mathbf{C}} = \mathbf{J}_{\hat{\mathbf{v}}}^{-T} \mathbf{Q} \quad (31)$$

where $\hat{\mathbf{C}}$ contains the nonlinear terms, and $\hat{\mathbf{H}}$ is given by

$$\hat{\mathbf{H}} = \mathbf{J}_{\hat{\mathbf{v}}}^{-T} \mathbf{H} \mathbf{J}_{\hat{\mathbf{v}}}^{-1} \quad (32)$$

Step 2. The following model-based control law is used

$$\mathbf{Q} = \mathbf{J}_{\hat{\mathbf{v}}}^T \{ \hat{\mathbf{H}} \mathbf{u} + \hat{\mathbf{C}} \} \quad (33)$$

where \mathbf{u} is an auxiliary control input, under the assumption of knowledge of a system's properties. Applying this law to the equations of motion (31) results in the following decoupled system

$$\dot{\hat{\mathbf{v}}} = \mathbf{u} \quad (34)$$

Note that Equation (34) is expressed in terms of linear and angular velocities, and not in terms of positions and Euler angles as is the case in Equation (25).

Step 3. An auxiliary control signal \mathbf{u} is used, and is partitioned as $\mathbf{u} = [\mathbf{u}_{\mathbf{R}}^T, \mathbf{u}_{\boldsymbol{\omega}}^T, \mathbf{u}_{\dot{\mathbf{x}}}^{(1)T}, \mathbf{u}_{\boldsymbol{\omega}}^{(1)T}, \dots, \mathbf{u}_{\dot{\mathbf{x}}}^{(n)T}, \mathbf{u}_{\boldsymbol{\omega}}^{(n)T}]^T$, where the partition follows that of $\hat{\mathbf{v}}$. The acceleration terms in Equation (34) that correspond to linear motions are controlled similar to Equation (26). For example, $\mathbf{u}_{\mathbf{R}}$ is given by

$$\mathbf{u}_{\mathbf{R}} = \mathbf{K}_{p,\mathbf{R}} \mathbf{e}_{\mathbf{R}} + \mathbf{K}_{d,\mathbf{R}} \dot{\mathbf{e}}_{\mathbf{R}} + \ddot{\mathbf{R}}_{0,des} \quad (35)$$

where

$$\mathbf{e}_{\mathbf{R}} = \mathbf{R}_{0,des} - \mathbf{R}_0 \quad (36)$$

However, the terms that correspond to angular velocities are controlled using

$$\mathbf{u}_\omega = \mathbf{R}_e \dot{\boldsymbol{\omega}}_{des} + \boldsymbol{\omega}^\times \boldsymbol{\omega}_e - \mathbf{K}_v \boldsymbol{\omega}_e - 2(\mathbf{K}_p - \boldsymbol{\omega}_e^T \boldsymbol{\omega}_e / 4) \boldsymbol{\epsilon}_e / n_e \quad (37)$$

In the above law, subscripts e and des correspond to error and desired quantities respectively. All \mathbf{w} 's are expressed in the corresponding body frame. Detailed expressions for computing Equation (37) are given in Appendix B. Applying the control law given by Equation (33) guarantees asymptotic convergence for the positional errors, and asymptotic convergence for the attitude error expressed in terms of Euler parameters. Note that due to the form of Equation (37), singularities occur only when n_e is zero, that is when the *attitude error* angle is π rad about any eigen axis. This problem can be tackled by a simple modification [17].

Note that all the above algorithms employ PD action; however, integral action can be easily incorporated. Also note that the above control approaches allow one to compute a set of generalized forces that will reduce the tracking error. The reaction jet forces and torques and the joint torques can be found by inverting an equation relating generalized forces to actuator forces, for example Equation (14).

IV. SIMULATIONS & COMPARISONS

In this section, the control algorithms developed in Section III are compared and evaluated. To this end, a planar free-flyer chasing a moving point target, is employed. The free-flyer includes three open chain appendages, two of which are two-link manipulators, while the third is a communications antenna, see Figure 3.

The spacecraft is equipped with reaction jets which provide the required control forces and torques up to some limited values. The system geometric parameters and mass properties, and the maximum available actuator forces/torques are displayed in Table II. The origin of the inertial frame coincides with the initial position of the system CM.

Table II-a. Spacecraft parameters and actuator limits.

$r_0^{(1)}$ (m)	$r_0^{(2)}$ (m)	$r_0^{(3)}$ (m)	m_0 (kg)	I_0 (kg m ²)	F_x (N)	F_y (N)	t_0 (N-m)
0.5	0.5	0.5	50.0	10.0	20.0	20.0	10.0

Table II-b. Manipulator parameters and joint actuator limits.

Appendage	i-th body	$r_i^{(m)}$ (m)	$l_i^{(m)}$ (m)	$m_i^{(m)}$ (kg)	$I_i^{(m)}$ (kgm^2)	$t_i^{(m)}$ (N-m)
1	1	0.50	0.50	4.0	0.50	7.0
1	2	0.50	0.50	3.0	0.25	5.0
2	1	0.50	0.50	4.0	0.50	7.0
2	2	0.50	0.50	3.0	0.25	5.0
3	1	0.25	0.25	5.0	2.00	7.0

The vector of generalized coordinates for this 8-DOF system is

$$\mathbf{q} = [x_{CM}, y_{CM}, \theta_0, \theta_1^{(1)}, \theta_2^{(1)}, \theta_1^{(2)}, \theta_2^{(2)}, \theta_1^{(3)}]^T \quad (38)$$

while the vector of coordinates to be controlled is

$$\hat{\mathbf{q}} = [x_0, y_0, \theta_0, x_E^{(1)}, y_E^{(1)}, x_E^{(2)}, y_E^{(2)}, \delta^{(3)}]^T \quad (39)$$

where x_{CM} and y_{CM} are the inertial coordinates of the system CM, x_0 and y_0 are the inertial coordinates of the spacecraft CM, θ_0 is the spacecraft attitude, $\theta_i^{(j)}$ is the i-th joint angle of the j-th manipulator, and $x_E^{(i)}$, $y_E^{(i)}$, and $\delta^{(i)}$ are the inertial coordinates and attitude of the i-th end-effector.

To ensure smooth operation, appropriate trajectories for the spacecraft motion are planned. It is assumed that the target is in the vicinity of the robotic system and it is a passive object, i.e. drifting at some constant speed, and that its trajectory is measured by such feedback devices as on-board cameras. Hence, the position and velocity of the target is available in the spacecraft frame.

1. Trajectory Planning. For the spacecraft motion, in both translation and rotation, parabolic trajectories made of constant acceleration, constant velocity, and constant deceleration segments are planned. These trajectories are first planned in the spacecraft frame at initial time, ${}^0\mathbf{x}(t) = [{}^0x_0, {}^0y_0]^T$, then will be transformed to the inertial space, $\mathbf{x}(t) = [x_0, y_0]^T$. To plan the desired trajectories, a motion final time, t_f , is first selected. Then, the required spacecraft position at t_f is found as follows. If ${}^0\mathbf{x}_{obj}(0)$ and ${}^0\mathbf{v}_{obj}(0)$ are the position and velocity of the object as measured *from (with respect to)* the spacecraft at initial time, then the final position of the spacecraft CM, ${}^0\mathbf{x}_f$, is given by

$${}^0\mathbf{x}_f = {}^0\mathbf{x}_{obj}(0) + [{}^0\mathbf{v}_{obj}(0) + {}^0\mathbf{v}_0(0)] t_f + {}^0\mathbf{r} \quad (40)$$

where ${}^0\mathbf{v}_0(0)$ is the initial velocity of the spacecraft, and ${}^0\mathbf{r}$ defines the relative position of the spacecraft CM and the object at time t_f . The direction of ${}^0\mathbf{r}$ is along the line connecting the spacecraft CM at initial time with the object location at t_f , and its magnitude is such that the manipulators can reach the object. During capture, it is desired to have the object stationary in the spacecraft frame, so the final spacecraft velocity, ${}^0\mathbf{v}_f$, is chosen as

$${}^0\mathbf{v}_f = {}^0\mathbf{v}_{obj}(0) + {}^0\mathbf{v}_0(0) \quad (41)$$

Next, parabolic trajectories made of constant acceleration, constant velocity, and constant deceleration segments, are planned to yield final position equal to ${}^0\mathbf{x}_f$, and final velocity equal to ${}^0\mathbf{v}_f$. Given the maximum acceleration \mathbf{a}_1 and deceleration \mathbf{a}_2 , using the above expressions yields time t_1 at which the acceleration segment ends, and time t_2 at which the deceleration segment starts, as follows

$$t_2 = \frac{{}^0\mathbf{v}_{obj_i}(0) - \mathbf{a}_{1i} t_1}{\mathbf{a}_{2i}} + t_f \quad (42a)$$

$$t_1 = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad (42b)$$

where

$$a = 0.5(\mathbf{a}_{1i} + \frac{\mathbf{a}_{1i}^2}{\mathbf{a}_{2i}}), \quad b = -\mathbf{a}_{1i} t_f - \frac{{}^0\mathbf{v}_{obj_i}(0) \mathbf{a}_{1i}}{\mathbf{a}_{2i}}, \quad c = {}^0\mathbf{x}_{obj_i}(0) + {}^0\mathbf{v}_{obj_i}(0) t_f - {}^0\mathbf{r} + 0.5 \frac{{}^0\mathbf{v}_{obj_i}(0)^2}{\mathbf{a}_{2i}} \quad (42c)$$

Note that the off/on times, t_1 and t_2 , are not necessarily equal for all three axes, ($i = 1, 2, 3$), corresponding to the spacecraft's position and orientation in planar motion. Also, in the case of having two positive solutions for t_1 , the smaller one is chosen to minimize energy consumption. Estimates for \mathbf{a}_1 and \mathbf{a}_2 can be obtained using thruster force capabilities and the mass properties of the system.

After computing the desired trajectory in the spacecraft frame, ${}^0\mathbf{x}(t)$, the trajectory in inertial space is computed by

$$\mathbf{x}(t) = \mathbf{x}(0) + \mathbf{T}_0(0) {}^0\mathbf{x}(t) \quad (43)$$

where $\mathbf{T}_0(0)$ is the transformation matrix between the spacecraft frame at initial time and the inertial frame, $\mathbf{x}(0)$ is the inertial position of the spacecraft CM at initial time, and $\mathbf{x}(t)$ is the inertial trajectory. In practice, the object would be under observation during the chase phase. Should its trajectory change significantly, a new spacecraft chase trajectory would be re-planned following the same procedure.

The desired trajectory for the orientation of the spacecraft, is similarly planned. The final orientation is chosen so as to provide an approximately symmetric motion of the manipulators during capture, since this strategy can minimize spacecraft disturbances.

The manipulators remain in their home configuration as long as the final position of the object is not in their fixed-base reachable workspace. During that period, a joint-space controller acting as a brake, is used. When the object enters the reachable workspace of an end-effector, a quintic trajectory [19], is planned in the task space for that end-effector. This trajectory provides position, velocity, and acceleration continuity throughout the motion. During this phase, a task-space control algorithm is applied. For the third appendage, i.e. the communications antenna, a constant attitude is commanded throughout the maneuver.

2. Simulation Results. For the simulation results that follow, the initial values are taken as

$${}^0\mathbf{x}_{obj}(0) = [3.0, 4.0]^T \text{ (m)}$$

$${}^0\mathbf{v}_{obj}(0) = [0.05, 0.1]^T \text{ (m/s)}$$

$$[x_0(0), y_0(0), \theta_0(0)]^T = [-0.0485\text{m}, -0.0659\text{m}, -\pi/6 \text{ rad}]^T$$

$$[{}^0\dot{x}_0(0), {}^0\dot{y}_0(0), \dot{\theta}_0(0)]^T = (0.01\text{m/s}, 0.01\text{m/s}, 0.001 \text{ rad/s})^T$$

$$\mathbf{q}(0) = [0, 0, -30^\circ, 45^\circ, 90^\circ, 135^\circ, -90^\circ, 30^\circ]^T$$

The final time for the linear motion, t_f , is 15.0 sec. To ensure a symmetric grasp, which causes minimum disturbances on the spacecraft, the final time for the rotational motion is chosen equal to $0.7t_f$. Taking into account the mass properties of the system and the available thruster forces/torques, the maximum acceleration and deceleration was set to $\mathbf{a}_1 = [0.2, 0.2]^T \text{ m/s}^2$, $\mathbf{a}_2 = 0.2\mathbf{a}_1$ for the linear

motion, and $a_1 = 0.05 \text{ rad/s}^2$, $a_2 = 0.5a_1$ for the rotational motion. There are two main reasons for taking the maximum deceleration significantly less than the maximum acceleration. First, because this requires smaller thruster forces before the grasp, therefore results in smaller object disturbances. Second, it provides a longer duration for manipulators to catch the object. This is due to the fact that in the above strategy, manipulators remain in their home configuration as long as the final position of the object is not in their fixed-base reachable workspace. Providing longer duration for manipulators motion reduces their speed which is desirable in space to prevent flexible mode excitations. The importance of symmetric grasp, and the importance of acceleration/deceleration ratio is investigated below, by simulation.

Figure 4 depicts manipulator joint trajectories, and an animated view of the corresponding system maneuver. Note that the joint angles for the two-link manipulators remain constant during the chase phase (in home configuration), and that they change smoothly during the capture phase (object in manipulator workspace). The joint angle for the third appendage, i.e. the antenna, changes smoothly so that a fixed inertial orientation is maintained during the maneuver.

For this system, the two model-based control algorithms (MB1, MB2) yield almost identical results, and so only those corresponding to MB1 are presented here. A comparison between these two laws in the 3-dimensional case has been given in [14]. Note that to compute the model based part in Equation (33), Equation (31) must be written in terms of Euler parameters and their rates. This can be done by expressing the Lagrangian in Equation (9) in terms of Euler parameters and their rates. However, the resulting equations of motion must be appended with algebraic constraints introduced by the four-parameter Euler parameter representation of rotations. A straightforward way to eliminate these constraints and obtain Equation (31), is to apply the so-called Natural Orthogonal Complement method [21].

To include the effects of model uncertainties in the MB laws, the mass properties of the model used in the control algorithm were perturbed with respect to the “true” parameters by up to 10%. The gains used for the MB1 controller are

$$\mathbf{K}_p = \text{diag}(70,70,100,100,100,100,100,70)$$

$$\mathbf{K}_d = \text{diag}(15,15,15,15,15,15,15,15)$$

while for the TJ controller these are

$$\mathbf{K}_p = \text{diag}(100,100,80,80,80,80,80,80)$$

$$\mathbf{K}_d = \text{diag}(150,150,100,100,100,100,100,100)$$

The gain selection for the model-based control was based on error equation settling time and damping criteria, while for the TJ control a heuristic approach was used.

Before going through comparisons between MB and TJ algorithms, the importance of symmetric grasp, and the ratio of acceleration/deceleration is investigated by simulation. To this end, the MB1 algorithm as described above is used. Figure 5, shows the profile of applied torque on the spacecraft for different grasp strategies, i.e. (a) symmetric and, (b) non-symmetric grasps. In Figure 5(a), i.e. symmetric grasp, the final orientation is chosen so that the axis of symmetry for the spacecraft is aligned with the direction of the object motion, while in Figure 5(b) a misalignment of just 5.0° between these directions is composed. As it is seen, during capture phase, $11.0 < t < 15.0$, the torque peak for symmetric grasp is almost half of the one for non-symmetric grasp. Therefore it can be concluded that a symmetric grasp reduces disturbances on the spacecraft.

As discussed earlier, there are two main reasons for choosing the maximum deceleration less than maximum acceleration in an on-off strategy, where total duration of the maneuver is given. Figure 6, demonstrates some consequences of this choice, by comparison between the two cases of $\mathbf{a}_2 = 0.2\mathbf{a}_1$ and $\mathbf{a}_2 = \mathbf{a}_1$. As shown in part (a), the former results in lower thruster forces before the grasp, therefore causes less disturbance on the object. This, $\mathbf{a}_2 = 0.2\mathbf{a}_1$, also requires lower torque on the spacecraft to track the desired trajectory. Considering the fact that an additional reaction torque is required in response to the manipulators motion, due to dynamics coupling, it results in a smoother profile for the applied torque on the spacecraft, see Figure 6(b). Also, since this provides a longer duration for manipulators to catch the object, tracking errors are reduced almost 50% with respect to the ones of $\mathbf{a}_2 = \mathbf{a}_1$, Figure 6(c).

Figure 7 can be used to compare and evaluate MB1 and TJ algorithms. Figure 7(a) displays the tracking error for the first manipulator end-effector in the task space. During the chase phase, this error is almost zero for MB1, as the manipulators are kept fixed at their home positions. When the object enters the manipulator workspace, the manipulators start moving, and tracking errors appear due to dynamic coupling and to transition to the task-space control phase. Note that in the absence of parameter uncertainties, i.e. for a perfect model-based control, feedback linearization results in zero tracking errors. However, the performance of the algorithm deteriorates if model uncertainties exist. These errors decrease with time and eventually vanish, in both MB and TJ algorithms.

Comparison of the maximum values of the tracking errors for the two algorithms shows that the errors occurring with TJ are about forty times larger than the errors with MB1, Figure 7(a). However, their absolute magnitude is small enough for executing an ordinary task in space. Comparison of the spacecraft thruster forces, shows that the required forces are about the same for both algorithms, Figure 7(b). However, in most parts of the maneuver, for MB algorithm the profile is staircase, while TJ does not yield such a profile. This is because the TJ algorithm does not use any knowledge of the dynamical behavior of the system, and therefore its operation is basically quasi-static. The required joint torques are lower in MB1, see Figure 7(c). The variation of the applied joint torques follows the variation of spacecraft attitude and also tracking errors, which is due to the same reasons, as above.

As shown by simulation, model-based algorithms result in smaller errors and required torques, and yield better results as long as model uncertainties are limited. Since torques are lower, smaller actuators are required, resulting in reduced system weight, an important issue in space. However, the performance of these algorithms deteriorates in the presence of larger model uncertainties. Also, implementing a model-based control requires increased computational burden, which may not be available, while at the same time it reduces closed-loop bandwidth. On the other hand, TJ control yields acceptable results without requiring a dynamics model of the system, and can be considered a good control algorithm candidate, especially when larger bandwidths and low computational costs are required.

V. CONCLUSIONS

In this paper, the motion control of a multiple arm free-flying space robot chasing a passive object in near proximity was studied. Using a minimum set of body-fixed barycentric vectors, and a general and a quasi-coordinate Lagrangian formulation, two dynamics models were derived. Control algorithms were developed that allow coordinated tracking control of the manipulators and the spacecraft. In particular, an Euler parameter model-based control algorithm was presented that overcomes the non-physical singularities due to Euler angle representation of attitude. To ensure smooth operation, reduced disturbances on the spacecraft and also on the object just before grasp, appropriate trajectories for the spacecraft/manipulators motion were planned. The importance of symmetric grasp, and the ratio of acceleration/deceleration in an on-off strategy, were investigated by simulation. Next, the performance of model-based algorithms was compared, by simulation, to that of a transposed Jacobian algorithm. Results show that, because of the complexity involved in the space robotic systems, there is a significant deterioration in the performance of model-based algorithms in the presence of model uncertainties. In such cases a simple transposed Jacobian algorithm can yield comparable results with much reduced computational burden which is an important issue in space.

VI. ACKNOWLEDGMENTS

The support of this work by the Natural Sciences and Engineering Council of Canada (NSERC), and by the Fonds pour la Formation de Chercheurs et l'Aide a la Recherche (FCAR), is acknowledged. We also wish to acknowledge the support to the second author by the Iran Ministry of Higher Education.

REFERENCES

- [1] Bronez, M. A., Clarke, M. M., and Quinn A., "Requirements Development for a Free-Flying Robot - the ROBIN," in *Proc. of the IEEE International Conference on Robotics and Automation*, San Fransisco, CA, April 1986.

- [2] Reuter, G. J., et al. "An Intelligent, Free-Flying Robot," in "Space Station Automation IV," Chiou, W.C. ed., *SPIE Proceedings Series*, Vol. 1006, November 1988, pp. 20-27.
- [3] Space Robots R&D in Japan, Special Issue of *New Technology Japan*, JETRO, Tokyo, Japan, 1990.
- [4] Alexander, H. and Cannon, R., "Experiments on the Control of a Satellite Manipulator," *Proc. 1987 American Control Conf.*, Seattle, WA, June 1987.
- [5] Umetani, Y. and Yoshida, K., "Resolved Motion Control of Space Manipulators with Generalized Jacobian Matrix," *IEEE Trans. on Robotics and Automation*, Vol. 5, No. 3, 1989, pp. 303-314.
- [6] Vafa, Z. and Dubowsky, S., "On the Dynamics of Space Manipulators Using the Virtual Manipulator, with Applications to Path Planning," *J. Astr. Sciences*, Vol. 38, No. 4, 1990, pp. 441-472.
- [7] Masutani, Y., Miyazaki, F., and Arimoto, S., "Sensory Feedback Control for Space Manipulators," *Proc. IEEE Int. Conf. on Robotics & Automation*, Scottsdale, AZ, May 1989.
- [8] Papadopoulos, E. and Dubowsky, S., "On the Nature of Control Algorithms for Free-floating Space Manipulators," *IEEE Trans. on Robotics and Automation*, Vol. 7, No. 6, Dec. 1991, pp. 750-758.
- [9] Fujii, H., et al., "Capture Control for Manipulator Arm of Free-flying Space Robot," *AIAA Paper*, AIAA-90-3432-CP, 1990.
- [10] Yoshida, K., "Space Structure Capturing and Assembling by Experimental Free-floating Robot Satellite (EFFORTS) Simulators," *Proc. Dynamics and Control in Space*, Cranfield, UK, Sep. 1993.
- [11] Papadopoulos, E. and Dubowsky, S., "Dynamic Singularities in the Control of Free-Floating Space Manipulators," *ASME Journal of Dynamic Systems, Measurement and Control*, Vol. 115, No. 1, March 1993, pp. 44-52. Also in *Space Robotics: Dynamics and Control*, Y. Xu, and T. Kanade eds., Kluwer, Boston, MA, 1993, pp. 77-100.

- [12] Spofford, J., and Akin, D., "Redundancy Control of a Free-Flying Telerobot," *Proc. of the AIAA Guidance, Navigation and Control Conference*, Minneapolis, MN, August 1988.
- [13] Papadopoulos, E. and Dubowsky, S., "Coordinated Manipulator/Spacecraft Motion Control for Space Robotic Systems," *Proc. of the IEEE Int. Conf. on Robotics and Automation*, Sacramento, CA, April 1991, pp. 1696-1701.
- [14] Papadopoulos, E. and Moosavian, S. Ali A., "Trajectory Planning & Control of Multiple Arm Space Free-Flyers," *Proc. of the 2nd Workshop on Robotics in Space*, International Advanced Robotics Program, Montreal, Canada, July 6-8, 1994.
- [15] Papadopoulos, E. and Moosavian, S. Ali A., "Dynamics & Control of Multi-arm Space Robots During Chase & Capture Operations," *Proc. Int. Conf. on Intelligent Robots and Systems (IROS '94)*, Munich, Germany, Sept. 12-16, 1994.
- [16] Meirovitch, L., *Methods of Analytical Dynamics*, McGraw-Hill, 1970.
- [17] Paielli, R. A. and Bach, R. E., "Attitude Control with Realization of Linear Error Dynamics," *J. of Guidance, Control, and Dynamics*, Vol. 16, No. 1, 1993, pp. 182-189.
- [18] Khatib, O., "A Unified Approach for Motion and Force Control of Robot Manipulators: The Operational Space Formulation," *IEEE J of Robotics & Automation*, Vol. RA-3, No. 1, Feb. 1987.
- [19] Craig, J., *Introduction to Robotics, Mechanics and Control*, Addison Wesley, Reading, MA, 1989.
- [20] Hughes, P.C., *Spacecraft Attitude Dynamics*, John Wiley, New York, NY, 1986.
- [21] Saha, S. K., Angeles J., "Dynamics of Nonholonomic Mechanical Systems Using a Natural Orthogonal Complement," *ASME J. of Applied Mechanics*, Vol. 58, March 1991, pp. 238-244.

APPENDIX A

Body-fixed barycentric vectors are given in [11], and can be written for a multiple arm free-flyer as

$$\tilde{\mathbf{v}}_{ki}^{(m)} = \begin{cases} \tilde{\mathbf{r}}_k^{(m)} = \mathbf{r}_i^{(m)} - \mathbf{e}_k^{(m)} & k < i \\ \tilde{\mathbf{e}}_k^{(m)} = -\mathbf{e}_k^{(m)} & k = i \\ \tilde{\mathbf{l}}_k^{(m)} = \mathbf{l}_i^{(m)} - \mathbf{e}_k^{(m)} & k > i \end{cases} \begin{cases} m = 1, \dots, n \\ i = 1, \dots, N_m \end{cases} \quad (\text{A1})$$

where vectors $\mathbf{r}_i^{(m)}$ and $\mathbf{l}_i^{(m)}$ are defined in Figure 1, and $\mathbf{e}_i^{(m)}$ are given by

$$\mathbf{e}_i^{(m)} = \mathbf{l}_i^{(m)}(1 - \mu_i^{(m)}) + \mathbf{r}_i^{(m)}\mu_{i+1}^{(m)} \quad (\text{A2})$$

$$\mathbf{e}_0 = \sum_{m=1}^n \mathbf{r}_0^{(m)} \mu_1^{(m)} \quad (\text{A3})$$

The quantity $\mu_i^{(m)}$ is the outboard mass after joint i in manipulator m , and is given by

$$\mu_i^{(m)} = \sum_{k=i}^{N_m} \frac{m_k^{(m)}}{M} \quad i = 1, \dots, N_m \quad \text{and} \quad \mu_{N_m+1}^{(m)} = 0 \quad (\text{A4})$$

Finally, M is the total mass of the system, and $m_k^{(m)}$ is the mass of the k -th body of the m -th manipulator.

APPENDIX B

The auxiliary control \mathbf{u}_ω computed according to Equation (37), is repeated here for completeness

$$\mathbf{u}_\omega = \mathbf{R}_e \dot{\boldsymbol{\omega}}_{des} + \boldsymbol{\omega}^\times \boldsymbol{\omega}_e - \mathbf{K}_v \boldsymbol{\omega}_e - 2(\mathbf{K}_p - \boldsymbol{\omega}_e^T \boldsymbol{\omega}_e / 4) \boldsymbol{\epsilon}_e / n_e \quad (\text{B1})$$

The matrix \mathbf{R}_e is a rotation matrix expressing the error between the desired and current attitude and is defined as

$$\mathbf{R}_e = \mathbf{R} \mathbf{R}_{des}^T \quad (\text{B2})$$

The matrix \mathbf{R} is the rotation matrix which corresponds to the orientation of a body with respect to the inertial frame, and \mathbf{R}_{des} corresponds to the desired orientation. Similarly, the angular velocity $\boldsymbol{\omega}_e$ is the error in angular velocities, expressed in the body-fixed frame

$$\boldsymbol{\omega}_e = \boldsymbol{\omega} - \mathbf{R}_e \boldsymbol{\omega}_{des} \quad (\text{B3})$$

where $\boldsymbol{\omega}$ is the body angular velocity and $\boldsymbol{\omega}_{des}$ the desired one, expressed in the desired orientation frame. Finally, $\boldsymbol{\epsilon}_e$ and n_e correspond to the error in attitude as expressed by Euler parameters

$$\boldsymbol{\epsilon}_e = \mathbf{T}_{des}^T \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_{des} n \quad (\text{B4})$$

$$n_e = \boldsymbol{\epsilon}_{des}^T \boldsymbol{\epsilon} + n_{des} n \quad (\text{B5})$$

with

$$\mathbf{T} = n \mathbf{I} + \boldsymbol{\epsilon}^\times \quad (\text{B6})$$

where \mathbf{I} is a 3×3 unit matrix, and n and $\boldsymbol{\epsilon}$ are the current Euler parameters [17].

As shown in [16], applying the control law given by Equation (37), the attitude error is governed by an homogeneous linear second order differential equation, which guarantees that the error will converge to zero asymptotically

$$\ddot{\boldsymbol{\epsilon}}_e + \mathbf{K}_d \dot{\boldsymbol{\epsilon}}_e + \mathbf{K}_p \boldsymbol{\epsilon}_e = \mathbf{0} \quad (\text{B7})$$

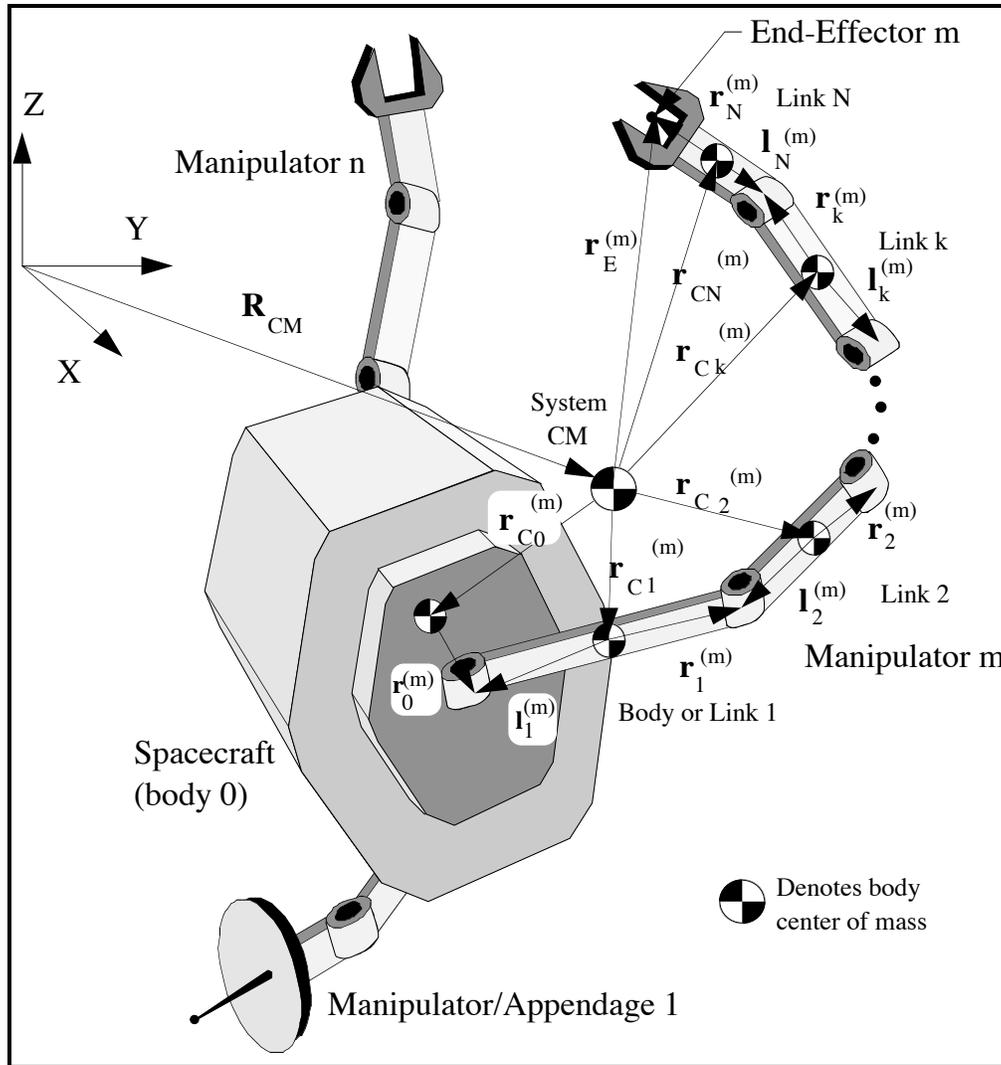


Figure 1. A free-flying space robotic system with n manipulators.

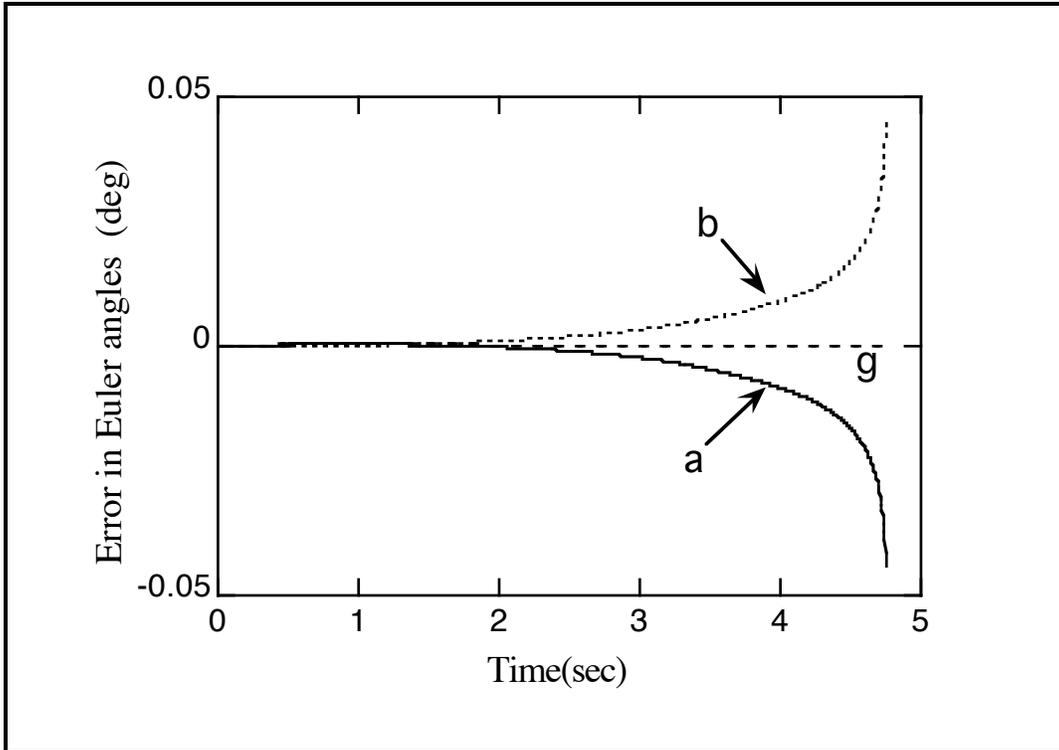


Figure 2. Errors in spacecraft orientation during a 3-d maneuver passing through a non-physical representational singularity at time = 4.75 sec.

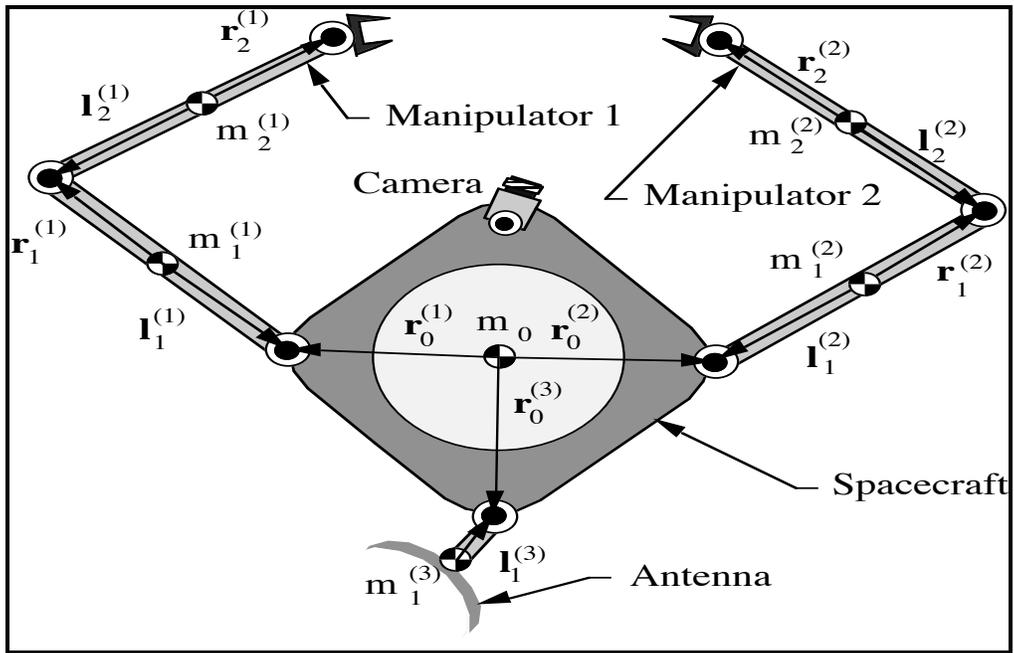
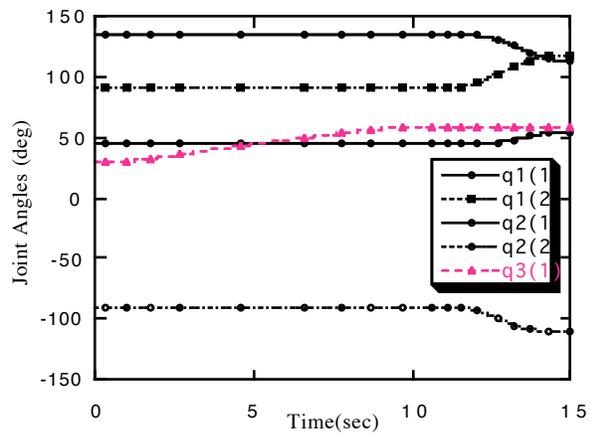


Figure 3. A planar three manipulator / appendage free-flyer.

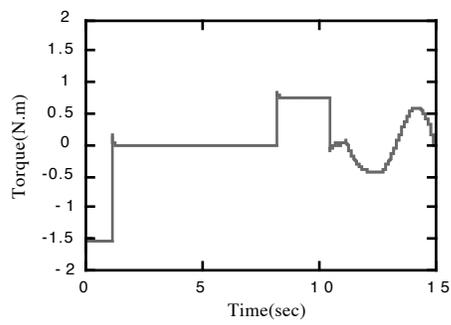


File Name : animation
 Title : MATLAB graph
 Creator : MATLAB, The Mathworks, Inc.
 CreationDate : 05/05/94 13:33:10
 Pages : 1

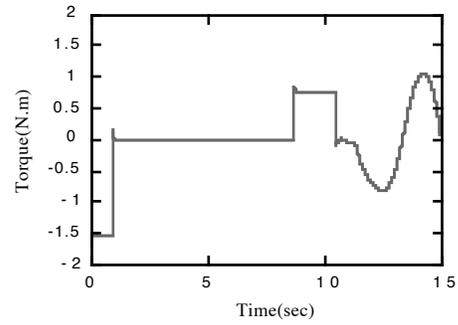
(a)

(b)

Figure 4. (a) Joint angle histories for the two manipulators and the antenna, (b) Animated view of the maneuver.

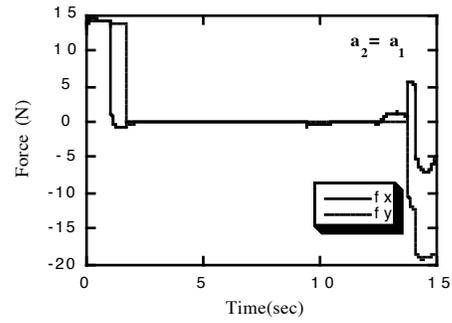
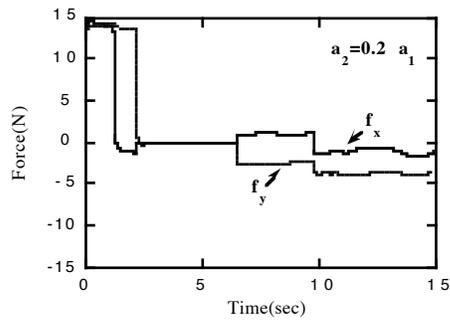


(a)

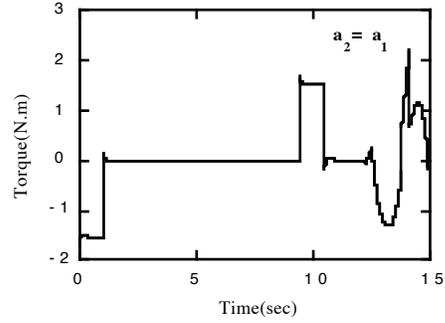
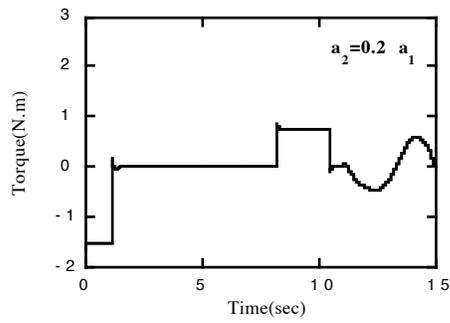


(b)

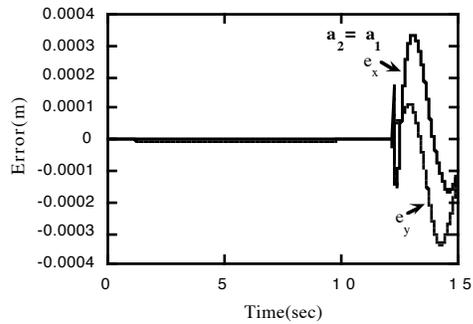
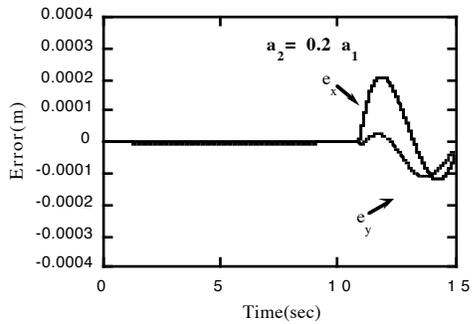
Figure 5. Applied torque on the spacecraft, (a) Symmetric grasp, (b) Non-symmetric grasp.



(a)

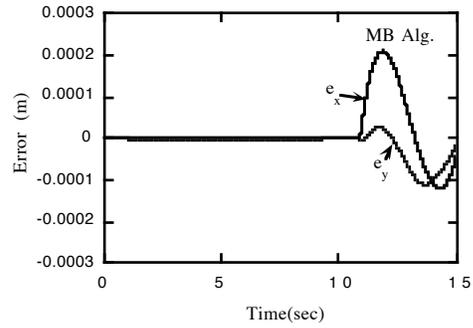
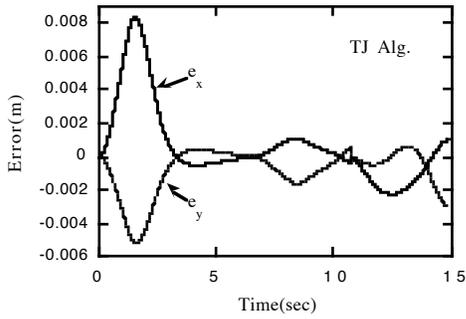


(b)

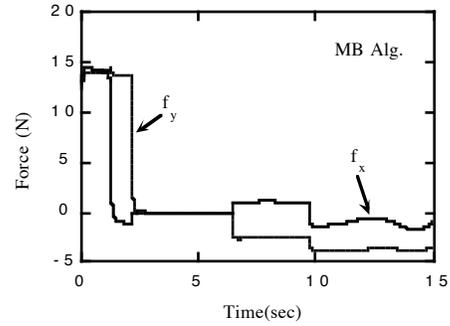
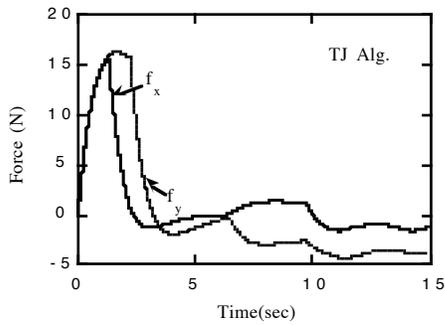


(c)

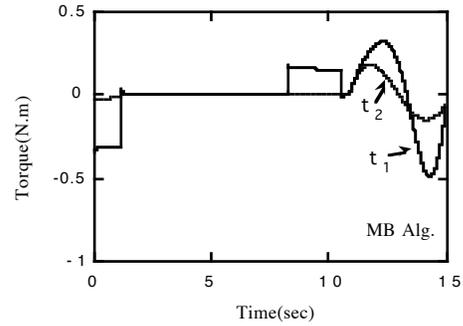
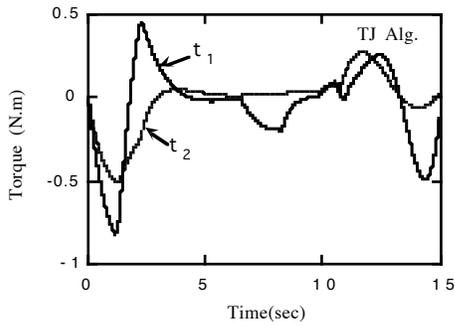
Figure 6. The effect of acceleration/deceleration ratio, (a) Thruster forces, (b) Applied torque on the spacecraft, (c) First end-effector positioning error.



(a)



(b)



(c)

Figure 7. Transposed Jacobian compared to Model-Based Control. (a) Tracking position errors for the first end-effector, (b) Thruster forces on the spacecraft, (c) Joint torques for the first manipulator.

Table I. Comparison of the required computational operations.

Algorithm	Multiplications	Additions
TJ	$3 N^2$	$3 N^2 - 2 N$
MB1	$2 N^3 + 7 N^2$	$2 N^3 + 5 N^2 - 4 N$

Table II-a. Spacecraft parameters and actuator limits.

$r_0^{(1)}$ (m)	$r_0^{(2)}$ (m)	$r_0^{(3)}$ (m)	m_0 (kg)	I_0 (kg m ²)	F_x (N)	F_y (N)	t_0 (N-m)
0.5	0.5	0.5	50.0	10.0	20.0	20.0	10.0

Table II-b. Manipulator parameters and joint actuator limits.

Appendage	i-th body	$r_i^{(m)}$ (m)	$l_i^{(m)}$ (m)	$m_i^{(m)}$ (kg)	$I_i^{(m)}$ (kgm ²)	$t_i^{(m)}$ (N-m)
1	1	0.50	0.50	4.0	0.50	7.0
1	2	0.50	0.50	3.0	0.25	5.0
2	1	0.50	0.50	4.0	0.50	7.0
2	2	0.50	0.50	3.0	0.25	5.0
3	1	0.25	0.25	5.0	2.00	7.0

List of Figures

- Figure 1. A free-flying space robotic system with n manipulators.
- Figure 2. Errors in spacecraft orientation during a 3-d maneuver passing through a non-physical representational singularity at time = 4.75 sec.
- Figure 3. A planar three manipulator / appendage free-flyer.
- Figure 4. (a) Joint angle histories for the two manipulators and the antenna, (b) Animated view of the maneuver.
- Figure 5. Applied torque on the spacecraft, (a) Symmetric grasp, (b) Non-symmetric grasp.
- Figure 6. The effect of acceleration/deceleration ratio, (a) Thruster forces, (b) Applied torque on the spacecraft, (c) First end-effector positioning error.
- Figure 7. Transposed Jacobian compared to Model-Based Control. (a) Tracking position errors for the first end-effector, (b) Thruster forces on the spacecraft, (c) Joint torques for the first manipulator.

Evangelos Papadopoulos received a Diploma in Mechanical Engineering from the National Technical University of Athens, Greece, in 1981, and M.S. and Ph.D. degrees in Mechanical Engineering from the Massachusetts Institute of Technology, in 1983 and 1990. He is an Assistant Professor of Mechanical Engineering at McGill University, and a member of the Centre for Intelligent Machines (CIM). Previously, in 1991, he was a Lecturer with the Department of Mechanical Engineering at MIT. From 1985 to 1987 he was a researcher at the Greek Navy Research and Technology Office.

Dr. Papadopoulos' main areas of interest are space robotics, mobile robotics, modelling and control of electromechanical systems, and control theory. He is a member of the IEEE, ASME, AIAA, and Sigma Xi.

S. Ali A. Moosavian received the B.Sc. degree in mechanical engineering in 1986 from Sharif University of Technology, and the M.Sc. degree in the same field in 1990 from Tarbiat Modaress University, both in Tehran. He is currently working on his Ph.D. in mechanical engineering at McGill University, Montreal. His dissertation research studies dynamics and control of space robotic systems, including their computational and graphical animation environments. Before attending McGill, he has been employed as a lecturer in Iran University of Science and Technology, Tehran. During his M.Sc. studies, he worked on boiler modeling and control at the Shahriar Steam Power Plant. He is a member of ASME and the CIM.