COORDINATED MOTION CONTROL OF MULTIPLE MANIPULATOR SPACE FREE-FLYERS

S. Ali A. Moosavian[1], **Evangelos Papadopoulos**[2]

In this paper, coordination between a spacecraft and its several manipulators is investigated, during a capture maneuver of a moving object in space. Using a general Lagrangian formulation, the dynamics model of the system is derived, and the results are summarized in an explicit dynamics model of multiple manipulator space free-flyers. The system dynamics is also formulated on the basis of choosing *Euler parameters* for orientation representation. This selection introduces algebraic constraints to the system, and the *Natural Orthogonal Complement Method* is applied to obtain independent system of equations of motion. Two model-based control algorithms, based on an *Euler angle* and an *Euler parameter* description of the orientation, are developed that allow coordinated control of the manipulators and the spacecraft, to track the planned trajectories. These trajectories ensure smooth operation, and reduce disturbances on the spacecraft and on the object just before grasping. The performance of the two model-based algorithms is compared to that of a transposed Jacobian controller, by a 3-D simulation. It is shown that both model-based algorithms result in smaller errors, as long as model uncertainties are limited. However, the Euler angle model-based control algorithm (MB1) presents the inconvenience of representational singularities at some orientations, while the one based on Euler parameters (MB2) overcomes this problem.

I. INTRODUCTION

Space Free-Flying Robots (SFFRs) in which manipulators are mounted on a thruster-equipped spacecraft, have been proposed to increase the mobility of robotic systems in space [1]. Unlike fixed-based robots, the base body of SFFR is disturbed by the dynamic reaction forces due to the arms motion. The kinematics and dynamics of a free-floating space manipulator system was described using the *Virtual Manipulator Approach* [2]. No external forces act on the system, and so the system center of mass is fixed in inertial space, enabling them to represent a free-floating system by one with a virtual fixed base. A

[1] Graduate Research Assistant, Department of Mechanical Engineering & Centre for Intelligent Machines, McGill University, Montreal, QC, Canada H3A 2A7

[2] Assistant Professor, Department of Mechanical Engineering & Centre for Intelligent Machines, McGill University, Montreal, QC, Canada H3A 2A7

barycentric vector approach was employed to study kinematics and dynamics of a single arm SFFR in free-floating mode [3]. Taking the center of mass of the whole system as a representative point for the translational motion, and using barycentric vectors which reflect both geometric configuration and mass distribution of the system, results in a decoupling of the total linear and angular motion from the rest of the equations. Also, a *Generalized Jacobian Matrix* for a free-floating system was presented [4]. Assuming that no external forces are applied on a rigid robotic system with revolute joints, they derive a generalized Jacobian matrix which reflects both momentum conservation laws and kinematic relations. The proposed generalized Jacobian matrix converges to the conventional Jacobian, when the base body is relatively massive.

Although dynamics modelling of SFFR is still an ongoing subject of research, control of these free-flying manipulators to perform precise tasks in space, has already received some attention. In order to control such a system, it is essential to consider the dynamic coupling between the arms and the base [5]. A resolved motion rate control was developed to compensate for spacecraft motion [6], and applied to the control of a multiple arm system [7]. The extended operational-space method was developed to control the motion of a SFFR based on a reference model [8]. Efficient algorithms for computing the generalized Jacobian matrix was studied to present the resolved acceleration control for multiple arm space robots [9]. Control strategies for changing the configuration of all joints of an underactuated space manipulator was also studied [10].

This paper studies coordination between a spacecraft and its several manipulators, during a capture maneuver of a moving object in space. First, the dynamics model of the system is derived, and an explicit dynamics model of multiple manipulator space free-flyers is presented. The use of *Natural Orthogonal Complement Method* in formulating the system dynamics on the basis of choosing *Euler parameters* for orientation representation is briefly discussed. Then, two model-based control algorithms, based on an *Euler angle* and an *Euler parameter* description of the orientation, and a transpose Jacobian algorithm are developed. To ensure smooth operation and reduced disturbances on the spacecraft and on the object, appropriate trajectories are planned. Finally, the performance of alternative control algorithms is compared to each other, by a 3-D simulation, and the results are discussed.

II. DYNAMICS MODELLING

In this section, using a set of body-fixed geometric vectors, the dynamics of a rigid multiple arm free-flying space robotic system is developed. The motion of the spacecraft center of mass (CM) is used to describe the system global translation with respect to an inertial frame of reference XYZ, Figure 1. An alternative approach using the system CM to describe translation is described in [11], [12].

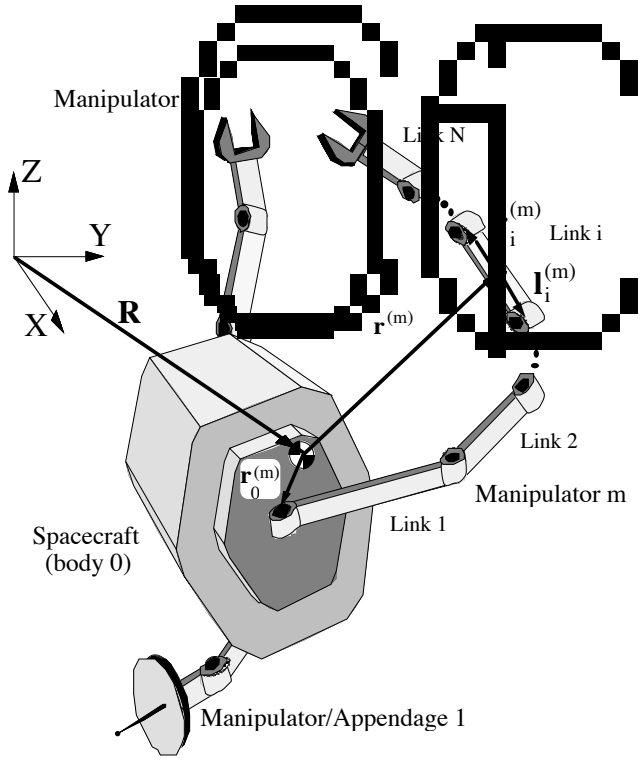


Figure 1: A free-flying space robotic system with n manipulators.

Since a typical maneuver of SFFR is of relatively short length and duration, microgravity and dynamical effects due to orbital mechanics are negligible, compared to control forces. Therefore, the motion of the system is considered with respect to an in-orbit inertial frame of reference (XYZ), and the system potential energy is taken equal to zero.

Using Euler Angles as Orientational Coordinates. Choosing the vector of generalized coordinates as

$$\mathbf{q} = (\mathbf{R}_{C_0}^T, \boldsymbol{\delta}_0^T, \boldsymbol{\theta}^T)^T \tag{1}$$

and using the general Lagrangian formulation, the equations of motion are obtained as

$$\mathbf{H}(\boldsymbol{\delta}_0, \boldsymbol{\theta})\ddot{\mathbf{q}} + \mathbf{C}(\boldsymbol{\delta}_0, \dot{\boldsymbol{\delta}}_0, \boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = \mathbf{Q}(\boldsymbol{\delta}_0, \boldsymbol{\theta}) \tag{2a}$$

where $\mathbf{R}_{C_0}$ is the inertial position of the spacecraft CM, $\boldsymbol{\delta}_0$ is a set of Euler angles that describe the orientation of the spacecraft, $\boldsymbol{\theta} = \left(\boldsymbol{\theta}^{(1)T}, \boldsymbol{\theta}^{(2)T}, \cdots, \boldsymbol{\theta}^{(n)T}\right)^T$ represents the joint angles of *n* manipulators, **H** is an N×N mass matrix, **C** is an N×1 vector which contains all the nonlinear velocity terms (in a microgravity environment), and **Q** is the N×1 vector of generalized forces given by

$$\mathbf{Q} = \begin{Bmatrix} \mathbf{0}_{6\times 1} \\ \boldsymbol{\tau}_{K\times 1} \end{Bmatrix} + \sum_{p=1}^{i_f} \mathbf{J}_{0,p}{}^T \mathbf{F}_{0,p} + \sum_{m=1}^{n} \sum_{i=1}^{N_m} \sum_{p=1}^{i_f} \mathbf{J}_{i,p}^{(m)\,T} \mathbf{F}_{i,p}^{(m)} \tag{2b}$$

in which $\mathbf{F}_{0,p}$ 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.

The elements of the mass matrix $\mathbf{H}$ can be explicitly determined as [13]

$$\begin{aligned} H_{ij} = {} & M\frac{\partial \mathbf{R}_{C_0}}{\partial q_i}\cdot\frac{\partial \mathbf{R}_{C_0}}{\partial q_j} + \frac{{}^0\partial \boldsymbol{\omega}_0}{\partial \dot{q}_i}\cdot\mathbf{I}_0\cdot\frac{{}^0\partial \boldsymbol{\omega}_0}{\partial \dot{q}_j} + \\ & \sum_{m=1}^{n}\sum_{k=1}^{N_m}\left(m_k^{(m)}\frac{\partial \mathbf{r}_{C_k}^{(m)}}{\partial q_i}\cdot\frac{\partial \mathbf{r}_{C_k}^{(m)}}{\partial q_j} + \frac{{}^k\partial \boldsymbol{\omega}_k^{(m)}}{\partial \dot{q}_i}\cdot\mathbf{I}_k^{(m)}\cdot\frac{{}^k\partial \boldsymbol{\omega}_k^{(m)}}{\partial \dot{q}_j}\right) + \\ & \left(\sum_{m=1}^{n}\sum_{k=1}^{N_m} m_k^{(m)}\frac{\partial \mathbf{r}_{C_k}^{(m)}}{\partial q_i}\right)\cdot\frac{\partial \mathbf{R}_{C_0}}{\partial q_j} + \left(\sum_{m=1}^{n}\sum_{k=1}^{N_m} m_k^{(m)}\frac{\partial \mathbf{r}_{C_k}^{(m)}}{\partial q_j}\right)\cdot\frac{\partial \mathbf{R}_{C_0}}{\partial q_i} \end{aligned} \tag{3}$$

where $\boldsymbol{\omega}_k^{(m)}$ is the angular velocity of the k-th body of the m-th manipulator, and $\mathbf{r}_{C_k}^{(m)}$ is the CM position vector of the i-th body with respect to the spacecraft CM, computed as

$$\mathbf{r}_{C_i}^{(m)} = \mathbf{r}_0^{(m)} + \sum_{k=1}^{i-1}(\mathbf{r}_k^{(m)} - \mathbf{l}_k^{(m)}) - \mathbf{l}_i^{(m)} \qquad \begin{cases} m = 1, \cdots, n \\ i = 1, \cdots, N_m \end{cases} \tag{4}$$

where vectors $\mathbf{l}_i^{(m)}$ and $\mathbf{r}_i^{(m)}$ are body-fixed vectors which describe the position of joints i and i+1 with respect to C_i, see Figure 1. The vector of nonlinear velocity terms in Eq. (2), $\mathbf{C}$, can be obtained as [13]

$$\mathbf{C}(\boldsymbol{\delta}_0, \dot{\boldsymbol{\delta}}_0, \boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = \mathbf{C}_1(\boldsymbol{\delta}_0, \dot{\boldsymbol{\delta}}_0, \boldsymbol{\theta}, \dot{\boldsymbol{\theta}})\,\dot{\mathbf{q}} + \mathbf{C}_2(\boldsymbol{\delta}_0, \dot{\boldsymbol{\delta}}_0, \boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) \tag{5a}$$

where $\dot{\mathbf{q}} = (\dot{\mathbf{R}}_{C_0}^T, \dot{\boldsymbol{\delta}}_0^T, \dot{\boldsymbol{\theta}}^T)^T$, and

$$\begin{aligned} \mathrm{C}_{1ij} = {} & \frac{{}^0\partial \boldsymbol{\omega}_0}{\partial \dot{q}_i}\cdot\mathbf{I}_0\cdot\frac{{}^0\partial \boldsymbol{\omega}_0}{\partial q_j} + \boldsymbol{\omega}_0\cdot\mathbf{I}_0\cdot\frac{{}^0\partial^2 \boldsymbol{\omega}_0}{\partial \dot{q}_i \partial q_j} + \frac{\partial \mathbf{R}_{C_0}}{\partial q_i}\cdot\sum_{m=1}^{n}\sum_{k=1}^{N_m}\left(m_k^{(m)}\sum_{s=1}^{N}\frac{\partial^2 \mathbf{r}_{C_k}^{(m)}}{\partial q_s \partial q_j}\dot{q}_s\right) + \\ & \sum_{m=1}^{n}\sum_{k=1}^{N_m}\left(m_k^{(m)}\frac{\partial \mathbf{r}_{C_k}^{(m)}}{\partial q_i}\cdot\left(\sum_{s=1}^{N}\frac{\partial^2 \mathbf{r}_{C_k}^{(m)}}{\partial q_s \partial q_j}\dot{q}_s\right) + \frac{{}^k\partial \boldsymbol{\omega}_k^{(m)}}{\partial \dot{q}_i}\cdot\mathbf{I}_k^{(m)}\cdot\frac{{}^k\partial \boldsymbol{\omega}_k^{(m)}}{\partial q_j} + \boldsymbol{\omega}_k^{(m)}\cdot\mathbf{I}_k^{(m)}\cdot\frac{{}^k\partial^2 \boldsymbol{\omega}_k^{(m)}}{\partial \dot{q}_i \partial q_j}\right) \end{aligned} \tag{5b}$$

$$\mathrm{C}_{2i} = -\left(\boldsymbol{\omega}_0\cdot\mathbf{I}_0\cdot\frac{{}^0\partial \boldsymbol{\omega}_0}{\partial q_i} + \sum_{m=1}^{n}\sum_{k=1}^{N_m}\boldsymbol{\omega}_k^{(m)}\cdot\mathbf{I}_k^{(m)}\cdot\frac{{}^k\partial \boldsymbol{\omega}_k^{(m)}}{\partial q_i}\right) \tag{5c}$$

Using Euler Parameters as Orientational Coordinates. Choosing *Euler parameters* for orientation representation introduces algebraic constraints to the system dynamics. This is due to the fact that these four parameters are not independent, and obey an algebraic constraint. An independent system of equations of motion can be obtained

using the *Natural Orthogonal Complement Method* [14], which is briefly described here. The vector of generalized coordinates is defined as

$$\mathbf{q} = (\mathbf{R}_{C_0}^T, \boldsymbol{\kappa}^T, \boldsymbol{\theta}^T)^T \tag{6}$$

where $\boldsymbol{\kappa} = \left(\boldsymbol{\varepsilon}^T, \eta\right)^T$ is the vector of Euler parameters describing the spacecraft attitude. Accordingly, the vector of generalized speeds is selected as $\boldsymbol{\nu} = (\mathbf{R}_{C_0}^T, {}^0\boldsymbol{\omega}_0^T, \dot{\boldsymbol{\theta}}^T)^T$. It can be shown that [15]

$$\boldsymbol{\omega}_0 = \mathbf{E}\,\dot{\boldsymbol{\kappa}} \tag{7a}$$

where

$$\mathbf{E} = 2\left[\breve{\mathbf{E}} \quad -\boldsymbol{\varepsilon}\right]_{3\times 4} \tag{7b}$$

$$\breve{\mathbf{E}} = \eta\,\mathbf{1} + [\boldsymbol{\varepsilon}]^{\times} \tag{7c}$$

Based on Eqs. (7), it can be written that

$$\dot{\mathbf{q}} = \boldsymbol{\Phi}\boldsymbol{\nu} \tag{8a}$$

where

$$\boldsymbol{\Phi} = \begin{bmatrix} \mathbf{1}_{3\times 3} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times K} \\ \mathbf{0}_{4\times 3} & \tfrac{1}{4}\mathbf{E}^T & \mathbf{0}_{4\times K} \\ \mathbf{0}_{K\times 3} & \mathbf{0}_{K\times 3} & \mathbf{1}_{K\times K} \end{bmatrix}_{(N+1)\times N} \tag{8b}$$

Similarly, $\boldsymbol{\nu} = \boldsymbol{\Psi}\dot{\mathbf{q}}$ can be defined, where

$$\boldsymbol{\Phi}\boldsymbol{\Psi} = \mathbf{1}_{(N+1)\times(N+1)} \quad \& \quad \boldsymbol{\Psi}\boldsymbol{\Phi} = \mathbf{1}_{N\times N} \tag{9}$$

Then, the equations of motion can be obtained as [13]

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \boldsymbol{\nu}}\right) + \left(\boldsymbol{\Psi}\dot{\boldsymbol{\Phi}}\right)^T \frac{\partial T}{\partial \boldsymbol{\nu}} - \boldsymbol{\Phi}^T\left(\frac{\partial T}{\partial \mathbf{q}}\right) = \boldsymbol{\Phi}^T\mathbf{Q} \tag{10}$$

which is a set of N independent equations, and represents the system dynamics in terms generalized speeds selected as $\boldsymbol{\nu} = (\mathbf{R}_{C_0}^T, {}^0\boldsymbol{\omega}_0^T, \dot{\boldsymbol{\theta}}^T)^T$, and the generalized coordinates as defined in Eq. (6). This is a proper dynamics model to be used in the implementation of the second model-based control algorithm developed in the following sections.

Computation of the obtained dynamics can be done either by *numerical* or *symbolical* programming tools. It was shown that preparation of each term for numerical programming requires cumbersome calculations, while by means of the symbolical tools, each term can be analytically calculated [13]. Also, using various mathematical identities and factorization

techniques, the result can be simplified to reduce the obtained analytical expressions. Next, planning appropriate trajectories is discussed.

III. TRAJECTORY PLANNING

In this section, planning appropriate trajectories for the spacecraft and its manipulators to result in capturing moving space objects, assumed to be passive, is briefly described. These trajectories ensure smooth operation, and reduce disturbances on the spacecraft and on the object just before grasping. All of these trajectories take into account the relative target motion, and thruster or actuator saturation limits.

For the spacecraft motion, in both translation and rotation, parabolic trajectories made of constant acceleration, constant velocity, and constant deceleration segments are planned. Since the object detecting sensors are usually on board, and thruster capabilities can be directly converted to the spacecraft maximum acceleration and deceleration magnitudes in the body frame, the desired trajectories are first planned in the spacecraft frame at initial time. These trajectories are subsequently transformed to the inertial space. The final spacecraft orientation is chosen so as to provide an approximately symmetric motion of the manipulators during capture, since this strategy can minimize spacecraft disturbances. To ensure this symmetric motion, the final time for orientational motion is chosen to be smaller than the final time used for the translational motion. Then the desired rotation matrix at final time is assembled such that an axis of symmetry for the spacecraft is aligned with the direction of the object motion. To position the end-effectors, this constraint yields an infinite number of solutions. Therefore another constraint should be added, e.g. keeping the spacecraft roll angle (if the attitude is described by Euler angles) constant during the maneuver. Then, the corresponding parameters for the spacecraft final attitude are extracted from the desired rotation matrix. Having these values, the desired trajectory for the orientation of the spacecraft can be planned [13].

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[1], $t = t_r$, a quintic trajectory is planned in the task space for that end-effector, and accordingly a task-space control algorithm is applied. For instance, to plan the desired trajectory for end-effector position, six coefficients have to be determined for each component. First, the end-effector position, linear velocity, and acceleration at starting time ($t = t_r$) are computed based on the current spacecraft position/orientation, and its linear and angular velocity and acceleration. The final values are also computed based on final position

1- The planned trajectory for the spacecraft rotation aims to provide a symmetric grasp of the object, by two participating manipulators. Therefore, the object enters the fixed-base workspace of both end-effectors, almost at the same time.

and velocity of the object. Then, the six coefficients of the desired quintic trajectory can be computed based on end-effector position, linear velocity, and acceleration at initial and final time [16]. The result provides continuity of end-effector position, linear velocity, and acceleration, throughout the motion. The desired trajectory for end-effector orientation, can be similarly planned. For some appendages, e.g. the communications antenna, a constant attitude in the inertial frame is commanded throughout the maneuver.

IV. DESIGN OF CONTROL LAWS

The first model-based control algorithm (MB1) is based on using Euler angles and suffers from the inconvenience of representational singularities. It is expected that such non-physical singularities will occur whenever a three-parameter description of the orientation such as Euler angles, is employed. This is due to the fact that inversion of the relation between angular velocity and Euler rates is not possible at some orientations. Note that such an inversion is required in calculating actuator forces/torques based on the control command [13]. So, the orientational error grows as the system approaches these singularities, Figure 2, and if it goes through these points, the control system fails. Therefore, at such points, a different set of Euler angles must be used. 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 as part of a coordination scheme to control a multiple arm free-flyer robot, and is presented as the second model-based control algorithm (MB2).

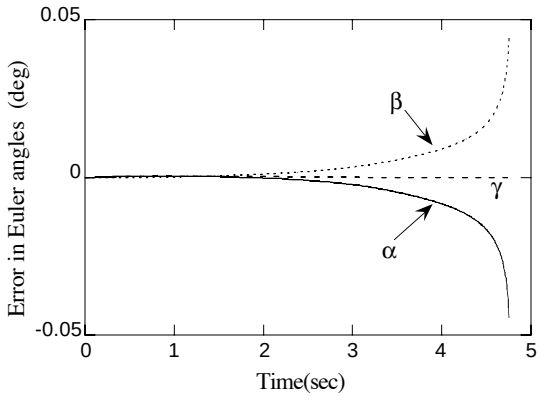


Figure 2: Errors in spacecraft orientation encountering a non-physical singularity at time = 4.75 sec.

Using Euler Angles (MB1). Defining the output variables as

$$\mathbf{q} = [\mathbf{R}_{C_0}^T, \boldsymbol{\delta}_0^T, \mathbf{x}_E^{(1)T}, \boldsymbol{\delta}_E^{(1)T}, \cdots, \mathbf{x}_E^{(n)T}, \boldsymbol{\delta}_E^{(n)T}]^T \tag{11}$$

where $\mathbf{x}_E^{(m)}$ and $\boldsymbol{\delta}_E^{(m)}$ correspond to the m-th end-effector position and orientation, the dynamics model of Eq. (2) can be rewritten as

$$\hat{\mathbf{H}}_\delta \, \ddot{\hat{\mathbf{q}}} + \hat{\mathbf{C}}_\delta = \hat{\mathbf{Q}}_\delta \tag{12}$$

Then, the model-based control law (MB1), under the assumption of knowledge of the system's properties, is presented as

$$\hat{\mathbf{Q}}_\delta = \hat{\mathbf{H}}_\delta \, \mathbf{u} + \hat{\mathbf{C}}_\delta \tag{13a}$$

The auxiliary control signal $\mathbf{u}$ can be computed as

$$\mathbf{u} = \mathbf{K}_p \, \mathbf{e} + \mathbf{K}_d \, \dot{\mathbf{e}} + \ddot{\mathbf{q}}_{des} \tag{13b}$$

where $\mathbf{K}_p$, and $\mathbf{K}_d$ are chosen as *positive definite* matrices, to result in a guaranteed stable error behavior, and $\mathbf{e}$ is the tracking error. Applying the control law of Eq. (13) results in

$$\ddot{\mathbf{e}} + \mathbf{K}_d \, \dot{\mathbf{e}} + \mathbf{K}_p \, \mathbf{e} = \mathbf{0} \tag{14}$$

which guarantees asymptotic convergence of the tracking error $\mathbf{e}$ to zero.

Using Euler Parameters (MB2). To develop the MB2 controller, the angular velocity of the spacecraft/end-effectors is included in the vector of generalized speeds, instead of corresponding Euler rates, and the system dynamics is formulated on the basis of choosing Euler parameters for orientation representation as described in Section II. Therefore, the output speeds are defined as

$$\hat{\mathbf{v}} = [\mathbf{R}_{C_0}^T, \, {}^0\boldsymbol{\omega}_0^T, \, \mathbf{x}_E^{(1)T}, \, {}^1\boldsymbol{\omega}_E^{(1)T}, \cdots, \mathbf{x}_E^{(n)T}, \, {}^n\boldsymbol{\omega}_E^{(n)T}]^T \tag{15}$$

Then, the dynamics model of Eq. (10) can be rewritten as

$$\hat{\mathbf{H}}_{\hat{v}} \, \dot{\hat{\mathbf{v}}} + \hat{\mathbf{C}}_{\hat{v}} = \hat{\mathbf{Q}}_{\hat{v}} \tag{16}$$

and the model-based control law (MB2) is presented as

$$\hat{\mathbf{Q}}_{\hat{v}} = \hat{\mathbf{H}}_{\hat{v}} \, \mathbf{u} + \hat{\mathbf{C}}_{\hat{v}} \tag{17a}$$

The auxiliary control signal $\mathbf{u}$ is partitioned as

$$\mathbf{u} = [\mathbf{u}_{\dot{\mathbf{R}}_0}^T, \mathbf{u}_{\omega_0}^T, \mathbf{u}_{\dot{\mathbf{x}}}^{(1)T}, \mathbf{u}_{\omega}^{(1)T}, \ldots, \mathbf{u}_{\dot{\mathbf{x}}}^{(n)T}, \mathbf{u}_{\omega}^{(n)T}]^T \tag{17b}$$

where the partition follows that of $\hat{\mathbf{v}}$, and

$$\mathbf{u}_{\mathbf{R}} = \mathbf{K}_{p,\mathbf{R}} \, \mathbf{e}_{\mathbf{R}} + \mathbf{K}_{d,\mathbf{R}} \, \dot{\mathbf{e}}_{\mathbf{R}} + \ddot{\mathbf{R}}_{C_0,des} \tag{17c}$$

$$\mathbf{u}_{\omega} = \mathbf{T}_e \, \dot{\boldsymbol{\omega}}_{des} + [\boldsymbol{\omega}]^{\times} \mathbf{e}_{\omega} - \mathrm{K}_{d,\omega} \, \mathbf{e}_{\omega} - 2(\mathrm{K}_{p,\omega} - \mathbf{e}_{\omega}^T \mathbf{e}_{\omega} / 4) \mathbf{e}_{\varepsilon} / e_{\eta} \tag{17d}$$

The matrix $\mathbf{T}_e$ relates the error between the desired and current attitude in terms of rotation matrices. In fact, it is a rotation matrix which maps the body frame with desired orientation to the actual body frame, and is defined as

$$\mathbf{T} = \mathbf{T}_e \, \mathbf{T}_{des} \tag{18a}$$

or

$$\mathbf{T}_e = \mathbf{T} \, \mathbf{T}_{des}^T \tag{18b}$$

The matrix $\mathbf{T}$ is a rotation matrix which corresponds to the current body orientation with respect to the inertial frame, and $\mathbf{T}_{des}$ is the one which corresponds to the desired orientation. The vector $\mathbf{e}_\omega$ is the error in angular velocity, expressed in the actual body-fixed frame

$$\mathbf{e}_\omega = \boldsymbol{\omega} - \mathbf{T}_e \, \boldsymbol{\omega}_{des} \tag{19}$$

where $\boldsymbol{\omega}$ is the current angular velocity of the corresponding body expressed in its own body fixed frame, and $\boldsymbol{\omega}_{des}$ is the desired angular velocity, expressed in the desired orientation frame. So, the term $\mathbf{T}_e \, \boldsymbol{\omega}_{des}$ represents the desired angular velocity resolved in the actual body frame, and the subtraction in Eq. (19) is in terms of consistent coordinates. Finally, $\mathbf{e}_\varepsilon$ and e_η, which correspond to the error in attitude as expressed by Euler parameters, are defined as

$$\mathbf{e}_\varepsilon = \breve{\mathbf{E}}_{des}^T \boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}_{des} \eta \tag{20}$$

$$e_\eta = \boldsymbol{\varepsilon}_{des}^T \boldsymbol{\varepsilon} + \eta_{des} \eta \tag{21}$$

where $\breve{\mathbf{E}}$ has been already defined in Eq. (7c).

Considering Eqs. (20, 21), for perfect tracking it can be obtained

$$\left(\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}_{des} \quad \& \quad \eta = \eta_{des}\right) \Rightarrow \left(\mathbf{e}_\varepsilon = \mathbf{0} \quad \& \quad e_\eta = 1\right) \tag{22}$$

It should be noted that assuming the same axis of rotation (for the desired and actual orientations), the above definitions given for $\mathbf{e}_\varepsilon$ and e_η result in $\|\mathbf{e}_\varepsilon\| = \sin(e_{\theta_0}/2)$ and $e_\eta = \cos(e_{\theta_0}/2)$ where θ_0 describes a simple rotation about axis of rotation, and e_{θ_0} is error in θ_0. Therefore, these definitions are geometrically meaningful, rather than $\left(\boldsymbol{\varepsilon}_{des} - \boldsymbol{\varepsilon}\right)$ and $\left(\eta_{des} - \eta\right)$ which do not have any physical interpretation. Also note that due to the form of Eq. (17d), singularities occur only when e_η is zero, that is when the *attitude error* angle is π rad about the eigen axis, i.e. $e_\eta = \cos(e_{\theta_0}/2) = \cos(\pi/2) = 0$.

Applying the control law given by Eq. (17), the attitude error is governed by a homogeneous linear second order differential equation, which guarantees that the error will converge asymptotically to zero

$$\ddot{\mathbf{e}}_\varepsilon + \mathrm{K}_{d,\omega}\,\dot{\mathbf{e}}_\varepsilon + \mathrm{K}_{p,\omega}\,\mathbf{e}_\varepsilon = \mathbf{0} \tag{23}$$

Therefore, it can be concluded that applying the control law given by Eq. (17) guarantees asymptotic convergence for the position errors, and attitude error expressed in terms of Euler parameters.

Transpose Jacobian Algorithm (TJ). If high enough gains are used, the simpler transpose Jacobian controller (TJ) can be employed as

$$\mathbf{Q} = \mathbf{J}_{C_\delta}^{T} \left\{ \mathbf{K}_p\,\mathbf{e} + \mathbf{K}_d\,\dot{\mathbf{e}} \right\} \tag{24}$$

This algorithm is quite simple to use with no significant computational burden, and without requiring a priori knowledge of plant dynamics. In fact, for slow trajectories this algorithm approximates the behavior of the model-based algorithms.

Having a mathematical model of the system dynamics, developed control laws, and desired trajectories for every output variable, the system performance can now be simulated. This is to be discussed next.

V. SIMULATION RESULTS

The performance of the presented Euler parameter model-based controller (MB2) is now compared to that of alternative algorithms, i.e. an Euler angle model-based control algorithm (MB1) and a transposed Jacobian controller (TJ). The system is a 14-DOF space free-flyer equipped with reaction jets on the base, chasing a moving target in 3-dimensional space [13]. The spacecraft includes three open chain appendages, two of which are three-DOF manipulators, while the third is a two-DOF communication antenna. Figure 3 shows the system general configuration, and Figure 4 depicts the planned path for the spacecraft center of mass and the two end-effectors.

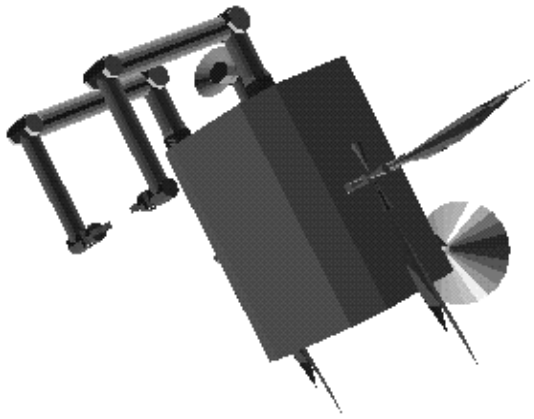


Figure 3: A three manipulator and appendage free-flyer.

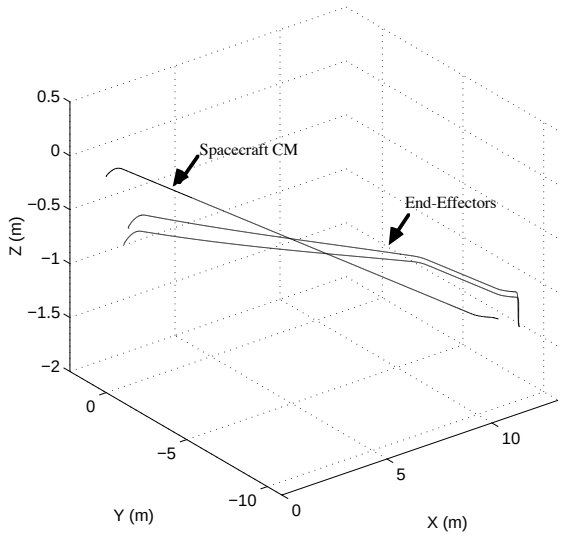


Figure 4: The desired path for the spacecraft center of mass and the two end-effectors.

To include the effects of model uncertainties in the MB laws, the mass properties of the model used in the control circuit were perturbed with respect to the "true" parameters by up to 30%. The gains used for the MB controllers are $\mathbf{K}_p$=diag(80,...,80,50,50), and $\mathbf{K}_d$=diag(150,...,150,100,100), while for the TJ controller the gains are $\mathbf{K}_p$=diag(300, 300,300,200,...,200,100,100), and $\mathbf{K}_d$=diag(600,600,600,400,...,400,200, 200). The gain selection for the model-based control was based on error equation settling time and damping criteria, while for the TJ control on heuristics.

Tracking error for the position of the first manipulator end-effector is shown in Figure 5. Other tracking errors (e.g. spacecraft CM position, second manipulator end-effector, etc.) behave similarly. So, Figure 5 represents typical error characteristics of the implemented algorithms. During the chase phase ($0< t <58$), the error for MB algorithms is almost zero, as the manipulators are kept fixed at their home configurations and the whole system moves like a single rigid body. However, for the TJ algorithm, the error is considerable at the beginning of this phase, where the system is accelerating (i.e. $0< t <7$ sec). This is due to the fact that the TJ algorithm does not include dynamics terms in its structure. When the object enters the manipulator workspace, the manipulators start moving, and some errors appear due to the dynamic coupling and also transition from joint-space to task-space control phase.

Comparison of the maximum tracking errors for these algorithms shows that the errors occurring with the TJ are about two-five times higher than the errors with the MB

algorithms[2], although their absolute magnitude may be considered small enough for performing a wide range of tasks.

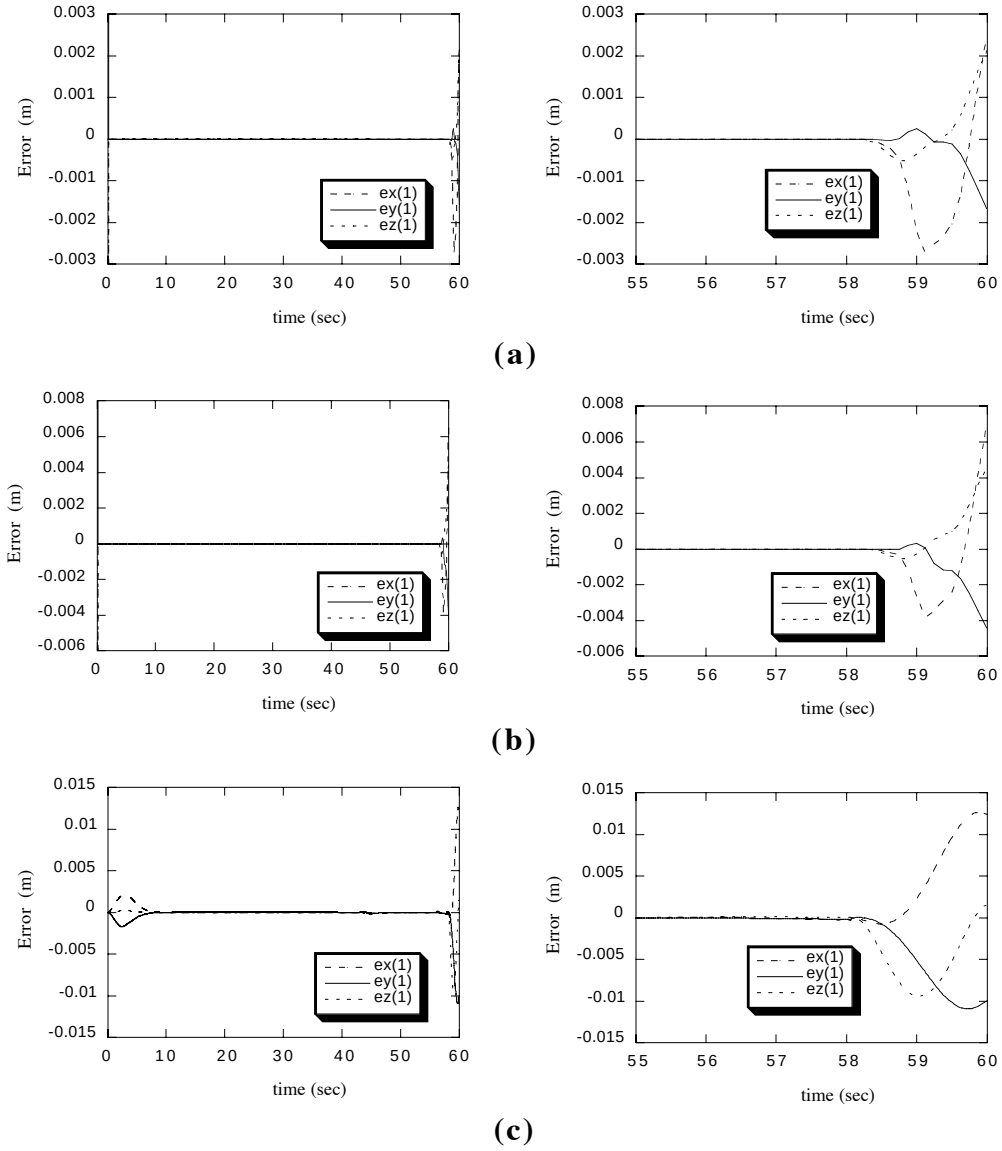
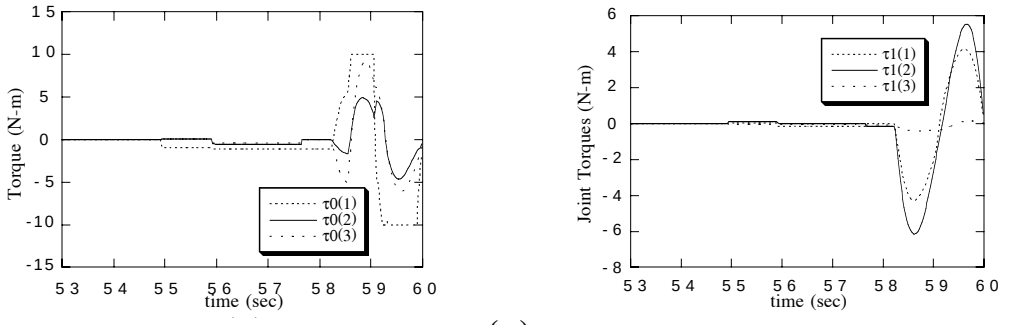


Figure 5: Tracking position errors for the first end-effector, (a) MB1. (b) MB2. (c) TJ.

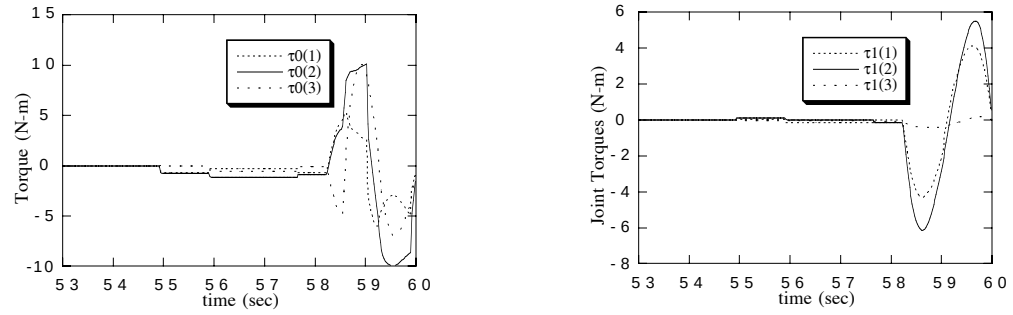
Figure 6 displays applied torques to control the spacecraft attitude and motion of the first manipulator, near the end of the maneuver (53.0< t <60.0). As shown in this figure, the required torques are almost the same for all algorithms, though MB2 is less demanding.

2- Note that to include the effects of model uncertainties in the MB laws, the mass properties of the model used in the control law were perturbed with respect to the "true" parameters by up to 30%. As expected and shown by simulation [13], the larger these uncertainties are the worse tracking is.

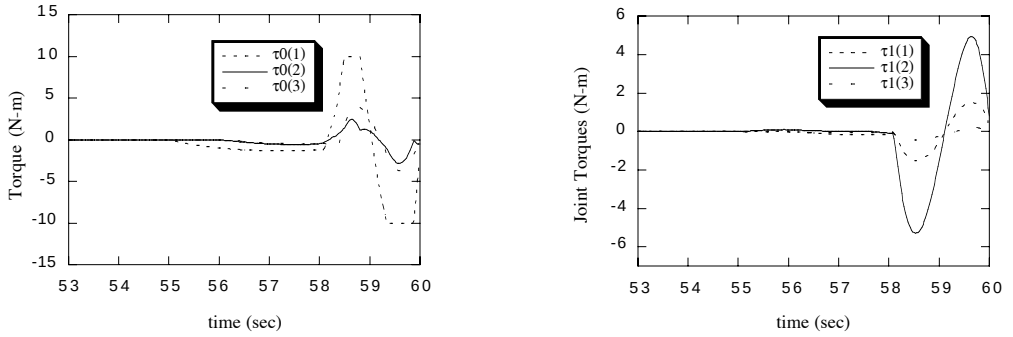
Note that the spacecraft torques for MB2 just touch the saturation limit (10 N-m), while for the others they remain saturated for a relatively long time.



(a)



(b)



(c)

Figure 6: Applied torques on the spacecraft (left) and joint torques for the first end-effector (right), (a) MB1, (b) MB2, (c) TJ Algorithm.

Comparing Figures 6a and 6c, it is interesting to note that profile of (c) is comparable to the profile of (a), despite the fact that the TJ does not include information about the system's dynamics.

As this general 3-dimensional maneuver reveals, the MB algorithms result in a better tracking and smaller errors, even in the presence of model uncertainties. The MB2 controller is preferred because as shown in the development of this algorithm, it overcomes the non-physical singularity problem.

VI. CONCLUSIONS

To obtain the dynamics model of a multiple manipulator SFFR with rigid elements, the general Lagrangian formulation was applied, and an explicit model was presented. Taking into account the object motion relative to the spacecraft, as well as thruster and actuator saturation limits, appropriate trajectories for the spacecraft and its manipulators motion were planned. Two model-based algorithms, and a transpose Jacobian control algorithm were developed. The Euler angle model-based control algorithm (MB1) presents the inconvenience of representational singularities at some orientations. To overcome this problem, an Euler parameter model-based control algorithm was proposed as the second model-based control algorithm (MB2). As shown by simulation, the model-based algorithms result in smaller errors, as long as model uncertainties are limited. However, due to the complexity of space robotic systems, the performance of these algorithms deteriorates if higher levels of model uncertainties exist. On the other hand, the TJ algorithm with relatively high gains, yields acceptable results (in terms of small errors and reasonable required forces/torques) for executing many tasks in space, without requiring knowledge of system dynamics.

VII. ACKNOWLEDGMENTS

The support of this work by the Natural Sciences and Engineering Council of Canada (NSERC) is acknowledged. We would also like to acknowledge support of the first author from 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 Int. Conf. on Robotics and Automation, San Francisco, CA, April 1986.
2. Vafa, Z. and Dubowsky, S., "On The Dynamics of Manipulators in Space Using The Virtual Manipulator Approach," *Proc. of IEEE Int. Conf. on Robotics and Automation*, April 1987, pp. 579-585.
3. Papadopoulos, E. and Dubowsky, S., "On The Nature of Control Algorithms for Free-Floating Space Manipulators," *IEEE Transactions on Robotics and Automation*, Vol. 7, No. 6, December 1991, pp. 750-758.
4. Umetani, Y. and Yoshida, K., "Continuous Path Control of Space Manipulators Mounted on OMV," *Acta Astronautica*, Vol. 15, No. 12, 1987, pp. 981-986.
5. Jaar, G., Cyril, X., and Misra, A. K., "Dynamical Modelling and Control of a Spacecraft-Mounted Manipulator Capturing a Spinning Satellite," *43rd Congress of The Int. Astronautical Federation*, September 1992, IAF-92-0029.

6. 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, June 1989, pp. 303-314.
7. Yoshida, K., Kurazume, R., and Umetani, Y., "Dual Arm Coordination in Space Free-Flying Robot," *Proc. of IEEE Int. Conf. on Robotics and Automation*, April 1991, pp. 2516-2521.
8. Alexander, H. and Cannon, R., "An Extended Operational-Space Control Algorithm for Satellite Manipulators," *The Journal of the Astronautical Sciences,* Vol. 38, No. 4, October-December 1990, pp. 473-486.
9. Yokokohji, Y., Toyoshima, T., and Yoshikawa, T., "Efficient Computational Algorithms for Trajectory Control of Free-Flying Space Robots with Multiple Arms," *IEEE Transactions on Robotics and Automation*, Vol. 9, No. 5, October 1993, pp. 571-580.
10. Mukherjee, R. and Chen, D., "Control of Free-Flying Underactuated Space Manipulators to Equilibrium Manifolds," *IEEE Transactions on Robotics and Automation*, Vol. 9, No. 5, October 1993, pp. 561-570.
11. 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.
12. Papadopoulos, E. and Moosavian, S. Ali A., "Dynamics & Control of Space Free-Flyers with Multiple Arms," *Journal of Advanced Robotics*, Vol. 9, No. 6, 1995, pp. 603-624.
13. Moosavian, S. Ali A., "Dynamics and Control of Free-Flying Manipulators Capturing Space Objects," *Ph.D. thesis*, McGill University, Montreal, Canada, June 1996.
14. Saha, S. K. and Angeles, J., "Dynamics of Nonholonomic Mechanical Systems Using a Natural Orthogonal Complement," *ASME Journal of Applied Mechanics*, Vol. 58, March 1991, pp. 238-244.
15. Angeles, J., *Rational Kinematics*, Springer-Verlag, New York, 1988.
16. Craig, J., *Introduction to Robotics, Mechanics and Control*, Addison Wesley, Reading, MA, 1989.
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.