Parameterization and Adaptive Control of Space Robot Systems

YANGSHENG XU, Member, IEEE
HEUNG-YEUNG SHUM, Student Member, IEEE
TAKEO KANADE, Fellow, IEEE
Carnegie Mellon University
JU-JANG LEE, Member, IEEE
Korea Advanced Institute of Science and Technology

In space application, robot systems are subject to unknown or unmodeled dynamics, for example, in the tasks of transporting an unknown payload or catching an unmodeled moving object. We discuss the parameterization problem in dynamic structure and adaptive control of a space robot system with an attitude-controlled base to which the robot is attached. We first derive the system kinematic and dynamic equations based on Lagrangian dynamics and the linear momentum conservation law. Based on the dynamic model developed, we discuss the problem of linear parameterization in terms of dynamic parameters, and find that, in joint space, the dynamics can be linearized by a set of combined dynamic parameters; however, in inertial space linear parameterization is impossible in general. Then we propose an adaptive control scheme in joint space, and present a simulation study to demonstrate its effectiveness and computational procedure. Because most takes are specified in inertial space instead of joint space, we discuss the issues associated to adaptive control in inertial space and identify two potential problems: unavailability of joint trajectory because the mapping from inertial space trajectory is dynamic-dependent and subject to uncertainty; and nonlinear parameterization in inertial space. We approach the problem by making use of the proposed joint space adaptive controller and updating the joint trajectory by the estimated dynamic parameters and given trajectory in inertial space.

Manuscript received November 17, 1991; revised January 20, 1993.

IEEE Log No. T-AES/30/2/15488.

Authors' addresses: Y. Xu, H-Y. Shum, T. Kanade, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213; J-J. Lee, Dept. of Electrical Engineering, Korea Advanced Institute of Science and Technology.

I. INTRODUCTION

Considerable research efforts have been directed to some primary functions of robots in space applications such as material transport [19], simple manipulation [5], manipulation coordination and navigation [16, 21], basic locomotion [15], and inspection and maintenance of the space station and satellites [2, 5]. The adaptive control is critical for the robot system subject to dynamic uncertainty in these tasks.

For material transport and manipulation tasks, space robots have to face uncertainty about the parameters describing the dynamic properties of the grasped load such as moments of inertia or exact position of the mass center. In most cases, these parameters are unknown and thus they cannot be specified off-line in inverse dynamics for feedforward compensation in any model-based control scheme. In catching a moving object [16], the robot is expected to be capable of adaptation to the dynamics change at the moment of catching operation. On the other hand, most space robots are designed to be light-weight and thus low-powered, for the zero gravity environment and energy-efficiency concern. As a result, joint friction and damping are much more significant in space robots than in industrial robots. These friction and damping effects are neither negligible nor easy to model. Adaptive control may provide a feasible solution to these system dynamics uncertainties. Adaptive control will also be able to accommodate various unmodeled disturbances, such as base disturbance, microgravity effect, sensor and actuator noise due to extremes of temperature and glare, or impact effect during the docking or rendezvous process.

Most existing adaptive control algorithms have the following shortcomings which cause their applications in space robots to be unrealistic: the use of joint acceleration measurement, the need of inversion of inertia matrix, high gain feedback and considerable computational cost. The first two must be avoided even for fixed-based industrial robots, because of lack of a joint acceleration sensor and the complexity of inversion of the inertia matrix. Slotine and Li [12] have tackled these problems successfully. A high gain feedback is extremely harmful for a space robot which is usually light-weight and low-powered. Considerable computation also needs to be avoided for allowable packaging of self-contained space robots.

This work focuses on the robot system where the base attitude is controlled by either thrust jets or reaction wheels. The reaction wheels are arranged in orthogonal directions, and the number of reaction wheels can be three or two depending on different tasks. A standard reaction wheel configuration can be found in [8]. When the attitude of the base is controlled, the orientation and position of both robot and base are no longer free, and the

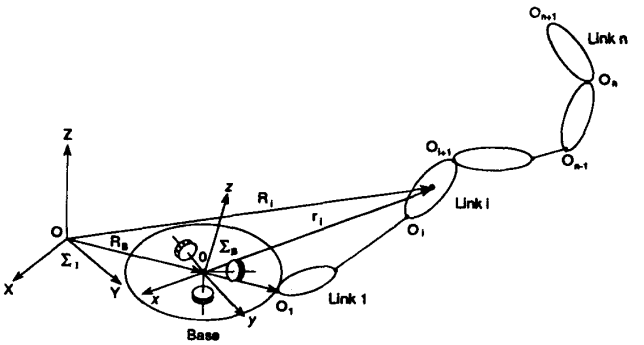


Fig. 1. Space robot system with an attitude-controlled base.

dynamic interaction results in the dynamic-dependent kinematics, i.e., the kinematics is in relation to the mass/inertia property of the base and robot. Control is not only applied to robot joint angles, but also to three orientations of the base.

Based on linear momentum conservation law and Lagrangian dynamics, we first formulate kinematics and dynamics equations of the space robot system with an attitude-controlled base in a systematic way. Based on the dynamic model developed, we study the linear parameterization problem, i.e., dynamics can be linearly expressed in terms of dynamic parameters such as mass and inertia. We find that for the space robot system with an attitude-control base, the linear parameterization is valid in joint space, while it is not valid in inertial space, which can be viewed as Cartesian space for Earth-based robots.

Using the dynamic model, we propose an adaptive control scheme in joint space. The scheme does not need to measure accelerations in joint space, and a high feedback gain is not required. Since, in most applications, the tasks are normally specified in inertial space instead of in joint space, we discuss the issues in relation to implementation of adaptive control in inertial space, and we identify two main problems. The first problem occurs when the joint adaptive control is executed. The required joint trajectory cannot be accurately generated by the given trajectory in inertial space due to the parameter uncertainty in the kinematic mapping which is dynamics dependent. The second problem, nonlinear parameterization in inertial space, makes it impossible to implement the same structured adaptive control as that possible in joint space. We approach this problem by making use of a joint space adaptive controller and by updating the joint trajectory from identified kinematic mapping and the given trajectory in inertial space.

In the simulation study we investigate the linear parameterization problem of robot system dynamics, and illustrate the validity and effectiveness of the proposed adaptive control schemes both in joint space and inertial space.

II. KINEMATICS RELATIONSHIP

In this section, we discuss the kinematics of the space robot system when the orientation of the base is controlled and the translation of the base is free. The relationship between the robot hand motion in inertial space and robot joint motion is derived using the linear momentum conservation law.

As shown in Fig. 1, a space robot system with an attitude-controlled base can be modeled as a multibody chain composed of $n+1$ rigid bodies connected by n joints, which are numbered from 1 to n. Each body is numbered from 0 to n, and the base is denoted by **B** in particular. The mass and inertia of ith body are denoted by m_i and $\mathbf{I}_i$, respectively. A joint variable vector $\mathbf{q} = (q_1, q_2, \ldots, q_n)^{\mathrm{T}}$ is used to represent those joint displacements. The orientation of the base is represented by a vector $\mathbf{q}_B = (q_{B1}, q_{B2}, q_{B3})^{\mathrm{T}}$.

Two coordinate frames are defined: the inertial coordinate $\sum_I$ on the orbit, and the base coordinate $\sum_B$ attached to the base body with its origin at the centroid of the base. As shown in Fig. 1, let $\mathbf{R}_i$ and $\mathbf{r}_i$ be the position vectors pointing the centroid of ith body with reference to $\sum_I$ and $\sum_B$, respectively, then

$$\mathbf{R}_i = \mathbf{r}_i + \mathbf{R}_B \tag{1}$$

where $\mathbf{R}_B$ is the position vector pointing the centroid of the base with reference to $\sum_I$. Let $\mathbf{V}_i$ and $\mathbf{\Omega}_i$ be linear and angular velocities of ith body with respect to $\sum_I$, $\mathbf{v}_i$ and ω_i with respect to $\sum_B$. Then we have

$$\begin{aligned} \mathbf{V}_i &= \mathbf{v}_i + \mathbf{V}_B + \mathbf{\Omega}_B \times \mathbf{r}_i \\ \mathbf{\Omega}_i &= \omega_i + \mathbf{\Omega}_B \end{aligned} \tag{2}$$

where $\mathbf{V}_B$ and $\mathbf{\Omega}_B$ are linear and angular velocities of the centroid of the base with respect to $\sum_I$, and operator "×" represents outer product of R^3 vector. The velocities $\mathbf{v}_i$ and ω_i in base coordinates can be represented by

$$\mathbf{v}_i = \mathbf{J}_{Li}(\mathbf{q})\dot{\mathbf{q}} \tag{3}$$

$$\omega_i = \mathbf{J}_{Ai}(\mathbf{q})\dot{\mathbf{q}} \tag{4}$$

where $\mathbf{J}_{Li}(\mathbf{q})$ and $\mathbf{J}_{Ai}(\mathbf{q})$ are the submatrices of Jacobian of the ith body representing linear part and angular part, respectively. The centroid of the total system can be determined by

$$m_c = \sum_{i=0}^{n} m_i \tag{5}$$

$$\mathbf{I}_c = \sum_{i=0}^{n} \mathbf{I}_i \tag{6}$$

$$\mathbf{r}_c = \sum_{i=0}^{n} m_i \mathbf{r}_i / m_c \tag{7}$$

$$\mathbf{J}_c = \sum_{i=1}^{n} m_i \mathbf{J}_{Li} / m_c. \tag{8}$$

The linear momentum can be determined [22] by

$$\mathbf{P} = [\mathbf{H}_V, \mathbf{H}_{V\Omega}] \begin{bmatrix} \mathbf{V}_B \\ \mathbf{\Omega}_B \end{bmatrix} + \mathbf{H}_{Vq}\dot{\mathbf{q}}$$

$$= \mathbf{H}_V \mathbf{V}_B + \mathbf{H}_{V\Omega}\mathbf{\Omega}_B + \mathbf{H}_{Vq}\dot{\mathbf{q}} \tag{9}$$

where

$$\mathbf{H}_V = m_c \mathbf{U}_3 \tag{10}$$

$$\mathbf{H}_{V\Omega} = -m_c[\mathbf{r}_c \times] \tag{11}$$

$$\mathbf{H}_{Vq} = m_c \mathbf{J}_c \tag{12}$$

and $\mathbf{U}_3$ is a 3×3 unity matrix. The matrix function $[\mathbf{r}\times]$ for a vector $\mathbf{r} = [r_x, r_y, r_z]^{\mathrm{T}}$ is defined as

$$[\mathbf{r}\times] = \begin{bmatrix} 0 & -r_z & r_y \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{bmatrix}. \tag{13}$$

Because there is no external force applied to the system, the linear momentum is conserved. However, the angular momentum is not conserved, for attitude-control torques are applied. The linear momentum is zero, assuming stationary initial condition.

$$\mathbf{P} = 0. \tag{14}$$

Therefore, we may represent the base linear velocities by base angular velocities and robot joint velocities, i.e.,

$$\mathbf{V}_B = -1/m_c[\mathbf{H}_{V\Omega}, \mathbf{H}_{Vq}] \begin{bmatrix} \mathbf{\Omega}_B \\ \dot{\mathbf{q}} \end{bmatrix}. \tag{15}$$

Now we derive the relationship between the motion rate in inertial space and that in joint space. For position control tasks, we are interested in controlling three orientations of the base, and six generalized displacements of the robot end-effector simultaneously. Control actions are instead applied at n robot joints and three base attitudes. We therefore define $\mathbf{V}$ and $\dot{\theta}$ as generalized velocities in inertial space and joint space,

$$\mathbf{V} = [\mathbf{\Omega}_B, \mathbf{V}_E]^{\mathrm{T}} \tag{16}$$

$$\dot{\theta} = [\mathbf{\Omega}_B, \dot{\mathbf{q}}]^{\mathrm{T}} \tag{17}$$

where $\mathbf{V}_E$ is the velocity of the robot end-effector in inertial space.

$$\mathbf{V}_E = \mathbf{v}_E + \mathbf{V}_B + \Omega_B \times \mathbf{r}_E \tag{18}$$

Since the velocity of the end-effector in the base coordinates is determined by

$$\mathbf{v}_E = \mathbf{J}_E \dot{\mathbf{q}} \tag{19}$$

where $\mathbf{J}_E$ is the manipulator Jacobian with respect to the base coordinates,

$$\mathbf{V}_E = \mathbf{J}_E\dot{\mathbf{q}} - 1/m_c[\mathbf{H}_{V\Omega}, \mathbf{H}_{Vq}] \begin{bmatrix} \mathbf{\Omega}_B \\ \dot{\mathbf{q}} \end{bmatrix} - [\mathbf{r}_E \times]\mathbf{\Omega}_B$$

$$= \left[-\frac{1}{m_c}\mathbf{H}_{V\Omega} - [\mathbf{r}_E\times], \mathbf{J}_E - \frac{1}{m_c}\mathbf{H}_{Vq}\right] \begin{bmatrix} \mathbf{\Omega}_B \\ \dot{\mathbf{q}} \end{bmatrix}$$

$$= [[(\mathbf{r}_c - \mathbf{r}_E)\times], \mathbf{J}_E - \mathbf{J}_c] \begin{bmatrix} \mathbf{\Omega}_B \\ \dot{\mathbf{q}} \end{bmatrix}. \tag{20}$$

Therefore, the motion rate relationship between joint space and inertial space can be obtained by introducing a special Jacobian matrix $\mathbf{N}$ which differs from the Jacobian in a fixed-base robot or the generalized Jacobian in a completely free-flying space robot system.

$$\begin{bmatrix} \mathbf{\Omega}_B \\ \mathbf{V}_E \end{bmatrix} = \mathbf{N} \begin{bmatrix} \mathbf{\Omega}_b \\ \dot{\mathbf{q}} \end{bmatrix} \tag{21}$$

$$\mathbf{N} = \begin{bmatrix} \mathbf{U}_3 & \mathbf{O}_3 \\ \mathbf{J}_{rr} & \mathbf{J}_{rE} \end{bmatrix} \tag{22}$$

where

$$\mathbf{J}_{rr} = [(\mathbf{r}_c - \mathbf{r}_E)\times] \tag{23}$$

$$\mathbf{J}_{rE} = \mathbf{J}_E - \mathbf{J}_c \tag{24}$$

and $\mathbf{O}_3$ is a 3×3 zero matrix.

III. DYNAMICS AND PARAMETERIZATION

In this section, we derive a dynamic equation of the space robot system with an attitude-controlled base. Based on the structure of dynamics equation, we discuss the property of linear parameterization of the system.

The total system kinetic energy is represented by [22]

$$
\begin{aligned}
\mathbf{T} &= 1/2\mathbf{V}_B^T\mathbf{H}_V\mathbf{V}_B + \mathbf{V}_B^T[\mathbf{H}_{V\Omega},\mathbf{H}_{Vq}]\begin{bmatrix}\boldsymbol{\Omega}_B\\ \dot{\mathbf{q}}\end{bmatrix}\\
&\quad + 1/2[\boldsymbol{\Omega}_B,\dot{\mathbf{q}}]\begin{bmatrix}\mathbf{H}_\Omega & \mathbf{H}_{\Omega q}\\ \mathbf{H}_{Tq} & \mathbf{H}_q\end{bmatrix}\begin{bmatrix}\boldsymbol{\Omega}_B\\ \dot{\mathbf{q}}\end{bmatrix}\\
&= 1/(2m_c^2)[\boldsymbol{\Omega}_B,\dot{\mathbf{q}}]\begin{bmatrix}\mathbf{H}_{V\Omega}\\ \mathbf{H}_{Vq}\end{bmatrix}\mathbf{H}_V[\mathbf{H}_{V\Omega},\mathbf{H}_{Vq}]\begin{bmatrix}\boldsymbol{\Omega}_B\\ \dot{\mathbf{q}}\end{bmatrix}\\
&\quad - 1/m_c[\boldsymbol{\Omega}_B,\dot{\mathbf{q}}]\begin{bmatrix}\mathbf{H}_{V\Omega}\\ \mathbf{H}_{Vq}\end{bmatrix}[\mathbf{H}_{V\Omega}\mathbf{H}_{Vq}]\begin{bmatrix}\boldsymbol{\Omega}_B\\ \dot{\mathbf{q}}\end{bmatrix}\\
&\quad + 1/2[\boldsymbol{\Omega}_B,\dot{\mathbf{q}}]\begin{bmatrix}\mathbf{H}_\Omega & \mathbf{H}_{\Omega q}\\ \mathbf{H}_{\Omega q}^T & \mathbf{H}_q\end{bmatrix}\begin{bmatrix}\boldsymbol{\Omega}_B\\ \dot{\mathbf{q}}\end{bmatrix}\\
&= 1/2[\boldsymbol{\Omega}_B,\dot{\mathbf{q}}]\begin{bmatrix}\mathbf{H}_\Omega - \mathbf{H}_{V\Omega}^T\mathbf{H}_{V\Omega}/m_c & \mathbf{H}_{\Omega q} - \mathbf{H}_{V\Omega}^T\mathbf{H}_{Vq}/m_c\\ \mathbf{H}_{\Omega q} - \mathbf{H}_{V\Omega}^T\mathbf{H}_{Vq}/m_c & \mathbf{H}_q - \mathbf{H}_{Vq}^T\mathbf{H}_{Vq}/m_c\end{bmatrix}\begin{bmatrix}\boldsymbol{\Omega}_B\\ \dot{\mathbf{q}}\end{bmatrix}\\
&= 1/2[\boldsymbol{\Omega}_B,\dot{\mathbf{q}}]\mathbf{M}(\theta)\begin{bmatrix}\boldsymbol{\Omega}_B\\ \dot{\mathbf{q}}\end{bmatrix}\\
&= 1/2\dot{\theta}^T\mathbf{M}(\theta)\dot{\theta} \qquad (25)
\end{aligned}
$$

where $\mathbf{M}$ is the inertia matrix of the system, $\mathbf{H}_q$ is the robot inertia matrix in base coordinate, i.e., fixed-base inertia matrix, and

$$\theta = [\mathbf{q}_B,\mathbf{q}]^T \qquad (26)$$

$$\mathbf{H}_\Omega = \mathbf{I}_c + \sum_{i=1}^{n}\mathbf{D}(\mathbf{r}_i)m_i \qquad (27)$$

$$\mathbf{H}_{\Omega q} = \sum_{i=1}^{n}(\mathbf{I}_i\mathbf{J}_{Ai} + m_i[\mathbf{r}_i]\mathbf{J}_{Li}) \qquad (28)$$

$$\mathbf{M}(\theta) = \begin{bmatrix}\mathbf{M}_{11} & \mathbf{M}_{12}\\ \mathbf{M}_{21} & \mathbf{M}_{22}\end{bmatrix} \qquad (29)$$

$$\mathbf{M}_{11} = \mathbf{H}_\Omega - \mathbf{H}_{V\Omega}^T\mathbf{H}_{V\Omega}/m_c \qquad (30)$$

$$\mathbf{M}_{12} = \mathbf{H}_{\Omega q} - \mathbf{H}_{V\Omega}^T\mathbf{H}_{Vq}/m_c \qquad (31)$$

$$\mathbf{M}_{21} = \mathbf{M}_{12}^T \qquad (32)$$

$$\mathbf{M}_{22} = \mathbf{H}_q - \mathbf{H}_{Vq}^T\mathbf{H}_{Vq}/m_c. \qquad (33)$$

The property of linear parameterization to dynamic parameters is one of the prerequisite conditions under which most adaptive and nonlinear dynamic control schemes are developed. It has been shown [22] that the problem of parameterization linearity in dynamics can be reduced to the problem of parameterization linearity in an inertia matrix. Therefore, in order to study whether linear parameterization is valid for the space robot system with an attitude-controlled base, we need to show whether the inertia matrix $\mathbf{M}$ can be linearly represented by a set of properly chosen combinations of dynamic parameters.

Based on the previous derivation, each member of matrix $\mathbf{M}$ can be further expanded in the following forms.

$$\mathbf{M}_{11} = \mathbf{I}_c + \sum_{i=1}^{n}\mathbf{D}(\mathbf{r}_i)m_i - \frac{1}{m_c}\begin{bmatrix}s_{11} & s_{12} & s_{13}\\ s_{21} & s_{22} & s_{23}\\ s_{31} & s_{32} & s_{33}\end{bmatrix} \qquad (34)$$

where $s_{ij} = s_{ji}$ and is determined by

$$s_{11} = \left(\sum_{i=1}^{n}m_ir_{iy}\right)^2 + \left(\sum_{i=1}^{n}m_ir_{iz}\right)^2 \qquad (35)$$

$$s_{22} = \left(\sum_{i=1}^{n}m_ir_{ix}\right)^2 + \left(\sum_{i=1}^{n}m_ir_{iz}\right)^2 \qquad (36)$$

$$s_{33} = \left(\sum_{i=1}^{n}m_ir_{ix}\right)^2 + \left(\sum_{i=1}^{n}m_ir_{iy}\right)^2 \qquad (37)$$

$$s_{12} = -\left(\sum_{i=1}^{n}m_ir_{ix}\right)\cdot\left(\sum_{i=1}^{n}m_ir_{iy}\right) \qquad (38)$$

$$s_{23} = -\left(\sum_{i=1}^{n}m_ir_{iy}\right)\cdot\left(\sum_{i=1}^{n}m_ir_{iz}\right) \qquad (39)$$

$$s_{13} = -\left(\sum_{i=1}^{n}m_ir_{ix}\right)\cdot\left(\sum_{i=1}^{n}m_ir_{iz}\right) \qquad (40)$$

and

$$\mathbf{D}(\mathbf{r}) = [\mathbf{r}\times]^T[\mathbf{r}\times] = \begin{bmatrix}r_y^2 + r_z^2 & -r_xr_y & -r_xr_z\\ -r_xr_y & r_x^2 + r_z^2 & -r_yr_z\\ -r_xr_z & -r_yr_z & r_x^2 + r_y^2\end{bmatrix}. \qquad (41)$$

Thus,

$$\mathbf{M}_{11} = \mathbf{I}_c + \sum_{i=1}^{n}\mathbf{D}(\mathbf{r}_i)m_i - \sum_{i=1}^{m}\sum_{j=1}^{n}R_{ij}m_im_j/m_c \qquad (42)$$

$$
\begin{aligned}
\mathbf{M}_{22} &= \mathbf{H}_q - \frac{1}{m_c}\left(\sum_{i=1}^{n}m_i\mathbf{J}_{Li}\right)^T\left(\sum_{i=1}^{n}m_i\mathbf{J}_{Li}\right)\\
&= \mathbf{H}_q + \sum_{i=1}^{n}\sum_{j=1}^{n}Q_{ij}m_im_j/m_c \qquad (43)
\end{aligned}
$$

$$\mathbf{M}_{12} = \sum_{i=1}^{n}\mathbf{I}_i\mathbf{J}_{Ai} + \sum_{i=1}^{n}[\mathbf{r}_i]\mathbf{J}_{Li}m_i + \sum_{i=1}^{n}\sum_{j=1}^{n}S_{ij}m_im_j/m_c \qquad (44)$$

where the matrices $[\mathbf{r}_i]\mathbf{J}_{Li}$, $\mathbf{J}_{Ai}$, R_i, Q_i, S_i are only functions of geometric parameters, i.e., independent of dynamic parameters. The above formulations imply that the inertia matrix can be linearly represented by a set of combination of dynamic parameters, m_k, $\mathbf{I}_k$, m_im_j/m_c, $i, j, k = 0,1,\ldots,n$.

The following two issues must be noted. First, from a control point of view, we only want to verify the linear parameterization in terms of any set of combination of dynamic parameters. However, from a parameter

identification point of view, the set of combination of dynamic parameters must be carefully selected [1, 10]. Second, the parameterization problem we discussed is specifically for dynamic parameters, i.e., mass/inertia parameters instead of geometric parameters. In space application, uncertainty in dynamic parameters is more important than geometric parameters, because not only do tasks involve unknown payload and unmodeled manipulation, but also dynamic parameters are usually unmeasurable. The geometric uncertainty may be significant in precision analysis and geometric design of robot configuration [20].

From the kinetic energy formulation, we can derive a dynamics equation by Lagrangian dynamics.

$$\mathbf{M}\ddot{\theta} + \mathbf{B}(\theta,\dot{\theta})\dot{\theta} = \tau \tag{45}$$

where

$$\mathbf{B}(\theta,\dot{\theta})\dot{\theta} = \dot{\mathbf{M}}\dot{\theta} - \frac{\partial}{\partial\theta}(\tfrac{1}{2}\dot{\theta}^T\mathbf{M}\dot{\theta}). \tag{46}$$

The corresponding dynamic equation in inertial space is

$$\mathbf{H}\ddot{\mathbf{x}} + \mathbf{C}(\mathbf{x},\dot{\mathbf{x}})\dot{\mathbf{x}} = \mathbf{F} \tag{47}$$

where

$$\mathbf{H} = \mathbf{N}^{-T}\mathbf{M}\mathbf{N}^{-1} \tag{48}$$

$$\mathbf{C} = \mathbf{N}^{-T}\mathbf{B}\mathbf{N}^{-1} - \mathbf{H}\dot{\mathbf{N}}\mathbf{N}^{-1}. \tag{49}$$

and N is a generalized Jacobian matrix and is dynamics dependent for the space robot system. The inertial space dynamic equation can be linearly expressed in terms of dynamic parameters if, and only if, the inertia matrix H can be linearly parameterized [22] since

$$\mathbf{C}(\mathbf{x},\dot{\mathbf{x}})\dot{\mathbf{x}} = \dot{\mathbf{H}}\dot{\mathbf{x}} - \frac{\partial}{\partial\mathbf{x}}(\tfrac{1}{2}\dot{\mathbf{x}}^T\mathbf{H}\dot{\mathbf{x}}). \tag{50}$$

We suppose $\mathbf{N}^{-1}$ exists, and

$$\mathbf{N}^{-1} = \frac{\mathbf{N}^*}{\det(\mathbf{N})} \tag{51}$$

where $\mathbf{N}^*$ and det(N) are the adjoint and determinant of the matrix N, then

$$\mathbf{H} = \frac{\mathbf{N}^{*T}\mathbf{M}\mathbf{N}^*}{[\det(\mathbf{N})]^2}. \tag{52}$$

In the above equation, the generalized Jacobian matrix $[\det(\mathbf{N})]^2$ appears as the denominator. From the deviation procedure of the N in the last section, it is clear that the N is time varying and highly coupled by dynamic parameters, i.e., mass/inertia. For such a complicated nonlinear, time-varying function combined with dynamic parameters and time-varying joint angles, it is impossible that every element of $\mathbf{N}^{*T}\mathbf{M}\mathbf{N}^*$ has the common factor $[\det(\mathbf{N})]^2$ at every instant.

Even if the above statement is true, there is still a possibility to linearly parameterize H, provided that the numerator can be linearly parameterized and the denominator can be expressed as a product of two scalar functions with only one containing dynamic parameters, i.e.,

$$\det(\mathbf{N}) = f_1(m_i, I_i)f_2(\theta_i) \tag{53}$$

where f_1 is a function of dynamic parameters which are unknown but constant, and f_2 is a function independent of any dynamic parameters. This, unfortunately, is impossible in general due to high coupling between dynamic parameters and joint variables. For example, two DOF generalized Jacobian may contain the following simple terms

$$\det(\mathbf{N}) = m_1\sin(\theta_1) + m_2\cos(\theta_2). \tag{54}$$

Even for such a simple form, det(N) cannot be decomposed as a product of two functions with one containing m_1 and m_2 only; nor can $[\det(\mathbf{N})]^2$.

The above discussion may raise a question why, for a fixed-base robot, the similar structured adaptive control can be implemented in Cartesian space. This is because the Jacobian in the fixed-base robots is only kinematic dependent, i.e., a function of geometric parameters and joint angles. Because of the dynamic interaction between the base and the robot, the generalized Jacobian for a space robot with an attitude control base is dynamics dependent, i.e., not only a function of geometric parameters and joint angles, but also a function of the dynamic parameters. It is these parameters that we aim to adapt in our problem. Therefore, the inertia matrix for the fixed-base robot can be linearly parameterized for dynamic parameters in Cartesian space, while for a space robot linear parameterization is impossible in inertial space.

Generally speaking, for a space robot with an attitude-controlled base, dynamics can be linearly parameterized in terms of dynamic parameters in joint space, but cannot be parameterized in inertial space.

IV. ADAPTIVE CONTROL IN JOINT SPACE

At an early state, adaptive control approaches for conventional fixed-base robot manipulators are based on unrealistic assumptions or approximations of local linearization, time-invariant, and decoupled dynamics [4, 7]. These assumptions or approximations are relaxed after some results are developed in the context of parameter estimation [10]. Based on the possibility of selecting a proper set of equivalent parameters such that the manipulator dynamics depends linearly on these parameters, research on adaptive robot control can now take full consideration of the nonlinear, time-varying and coupled robot dynamics. As stated in [9], all three kinds of adaptive controllers in use, i.e., direct [3, 12], indirect [11], and composite adaptive controllers [13], rely on the possibility of linear parameterization of manipulator dynamics.

From previous discussion, we learned that the dynamics of the space robot system in joint space is

linear in terms of a set of combinations of dynamic parameters. Therefore, this set of new combined parameters can be used in the design of our adaptive controller. This leads us to propose an adaptive control algorithm in joint space. Since a unique solution may be found from inverse kinematics of the robot system *with the attitude-controlled base*, an adaptive control algorithm in joint space is feasible. However, this is not true for a complete free-flying space robot system.

Recall the dynamic equation in joint space

$$\mathbf{M}\ddot{\theta} + \mathbf{B}(\theta,\dot{\theta})\dot{\theta} = \tau. \tag{55}$$

We define a composite error $\mathbf{s}$

$$\mathbf{s} = \dot{\mathbf{e}}_p + \zeta \mathbf{e}_p \tag{56}$$

$$\mathbf{e}_p = \theta_d - \theta \tag{57}$$

$$\dot{\mathbf{e}}_p = \dot{\theta}_d - \dot{\theta} \tag{58}$$

and we also define modified joint velocity

$$\theta' = \dot{\theta} + \mathbf{s} \tag{59}$$

and modified joint acceleration,

$$\theta'' = \frac{d}{dt}\theta' + \mathbf{s} \tag{60}$$

i.e.,

$$\begin{aligned}\theta'' &= \ddot{\theta} + \dot{\mathbf{s}} + \mathbf{s}\\ &= (\ddot{\theta}_d - \ddot{\mathbf{e}}_p) + (\ddot{\mathbf{e}}_p + \zeta\dot{\mathbf{e}}_p) + (\dot{\mathbf{e}}_p + \zeta\mathbf{e}_p)\\ &= \ddot{\theta}_d + (\zeta+1)\dot{\mathbf{e}}_p + \zeta\mathbf{e}_p = \ddot{\theta}_d + \mathbf{s} + \zeta\dot{\mathbf{e}}_p.\end{aligned} \tag{61}$$

If we apply the following control law in joint space,

$$\tau = \hat{\mathbf{M}}\theta'' + \hat{\mathbf{B}}\theta' \tag{62}$$

where $\hat{\mathbf{M}}$ and $\hat{\mathbf{B}}$ are the estimation of the matrices $\mathbf{M}$ and $\mathbf{B}$, then

$$\mathbf{M}\ddot{\theta} + \mathbf{B}(\theta,\dot{\theta})\dot{\theta} = \hat{\mathbf{M}}\theta'' + \hat{\mathbf{B}}\theta' \tag{63}$$

i.e.,

$$\mathbf{M}\ddot{\theta} = -\mathbf{B}(\theta,\dot{\theta})\dot{\theta} + \hat{\mathbf{M}}\theta'' + \hat{\mathbf{B}}\theta'. \tag{64}$$

Defining $\tilde{\mathbf{M}} = \hat{\mathbf{M}} - \mathbf{M}$, $\tilde{\mathbf{B}} = \hat{\mathbf{B}} - \mathbf{B}$, we have

$$\begin{aligned}\mathbf{M}\ddot{\mathbf{e}}_p &= \mathbf{M}\ddot{\theta}_d - \mathbf{M}\ddot{\theta}\\ &= \mathbf{M}[\theta'' - \mathbf{s} - \zeta\dot{\mathbf{e}}_p] - [-\mathbf{B}\dot{\theta} + \hat{\mathbf{M}}\theta'' + \hat{\mathbf{B}}\theta']\\ &= \mathbf{M}[\theta'' - \mathbf{s} - \zeta\dot{\mathbf{e}}_p] - [-\mathbf{B}(\theta' - \mathbf{s}) + \hat{\mathbf{M}}\theta'' + \hat{\mathbf{B}}\theta']\\ &= -\tilde{\mathbf{M}}\theta'' - \tilde{\mathbf{B}}\theta' - (\mathbf{M}+\mathbf{B})\mathbf{s} - M\zeta\dot{\mathbf{e}}_p\\ &= -\mathbf{Y}(\theta,\dot{\theta},\theta_d,\dot{\theta}_d,\ddot{\theta}_d)\tilde{\mathbf{a}} - (\mathbf{M}+\mathbf{B})\mathbf{s} - \mathbf{M}\zeta\dot{\mathbf{e}}_p\end{aligned}$$

where

$$\mathbf{Y}\tilde{\mathbf{a}} = \tilde{\mathbf{M}}\theta'' + \tilde{\mathbf{B}}\theta' \tag{65}$$

$$\tilde{\mathbf{a}} = \hat{\mathbf{a}} - \mathbf{a} \tag{66}$$

and $\hat{\mathbf{a}}$ is the estimation of the unknown dynamic parameters of the space robot system including the robot, the base, and probably the payload which is being manipulated.

We now design our adaptive control algorithm using the Lyapunov function candidate

$$V = 1/2\mathbf{s}^T\mathbf{M}\mathbf{s} + 1/2\tilde{\mathbf{a}}^T\Gamma\tilde{\mathbf{a}} \tag{67}$$

where the matrix Γ is diagonal and positive definite. This yields

$$\begin{aligned}\dot{V} &= 1/2\mathbf{s}^T\dot{\mathbf{M}}\mathbf{s} + \mathbf{s}^T\mathbf{M}\dot{\mathbf{s}} + \tilde{\mathbf{a}}^T\Gamma\dot{\tilde{\mathbf{a}}}\\ &= 1/2\mathbf{s}^T\dot{\mathbf{M}}\mathbf{s} + \mathbf{s}^T\mathbf{M}(\ddot{\mathbf{e}}_p + \zeta\dot{\mathbf{e}}_p) + \tilde{\mathbf{a}}^T\Gamma\dot{\tilde{\mathbf{a}}}\\ &= -\mathbf{s}^T\mathbf{Y}\tilde{\mathbf{a}} - \mathbf{s}^T(\mathbf{M}+\mathbf{B})\mathbf{s} + 1/2\mathbf{s}^T\dot{\mathbf{M}}\mathbf{s} + \tilde{\mathbf{a}}^T\Gamma\dot{\tilde{\mathbf{a}}}\\ &= -\mathbf{s}^T\mathbf{M}\mathbf{s} + 1/2\mathbf{s}^T(\dot{\mathbf{M}} - 2\mathbf{B})\mathbf{s} + \tilde{\mathbf{a}}^T(\Gamma\dot{\tilde{\mathbf{a}}} - \mathbf{Y}^T\mathbf{s}).\end{aligned}$$

If we use adaptation law

$$\dot{\tilde{\mathbf{a}}} = \Gamma^{-1}\mathbf{Y}^T\mathbf{s} \tag{68}$$

then

$$\dot{V} = -\mathbf{s}^T\mathbf{M}\mathbf{s} \le 0 \tag{69}$$

due to the fact that the matrix $\dot{\mathbf{M}} - 2\mathbf{B}$ is skew-symmetric, and $\mathbf{M}$ is positive definite. Therefore, the system is stable in the sense of Lyapunov, because V is a positive, nonincreasing function bounded below by zero. $\mathbf{s}(t)$ and $\tilde{\mathbf{a}}(t)$ are then bounded, and $\mathbf{s}(t)$ is a so-called square integrable or L_2 function [14]. Provided that the function $\mathbf{Y}$ is bounded, this is sufficient for the purpose of control because $\mathbf{s}(t)$ converges to zero as the L_2 function must converge to zero as $t \to \infty$. The parameter estimation error $\tilde{\mathbf{a}}(t)$ converges to zero only if persistent excited input is utilized.

The output error

$$\mathbf{s} = \dot{\mathbf{e}}_p + \zeta\mathbf{e}_p \tag{70}$$

converges to zero, which in turn implies that $\mathbf{e}_p \to 0$ as $t \to \infty$ since ζ is positive. We can now readily state our adaptive control algorithm in the following theorem.

THEOREM 1 *For the dynamic system* (55), *the adaptive control law defined by* (62) *and* (68) *is globally stable and guarantees zero steady-state error in joint space.*

The composite error $\mathbf{s}$ is of PD-type structure which is the same as the composite error defined by Slotine and Li [12]. In general, the PD structure control adds damping to the system, but the steady-state response is not affected. The PI structure adds damping and improves the steady-state error at the same time, but rising time and settling time are penalized. To improve the system steady-state error in the proposed adaptive control algorithm, the PID type $\mathbf{s}$ can also be used. Since the use of the PID type $\mathbf{s}$ causes the order and type of the system to increase by one, the steady-state error is decreased, and thus the system is more robust to parameter uncertainties which usually cause a significant steady-state error. Moreover, the PID type $\mathbf{s}$ allows two parameters, instead of one, to be adjustable in order to achieve a desired system performance. In the following, we

discuss the stability of the control scheme when the PID type s is employed.

Define

$$\mathbf{s} = \dot{\mathbf{e}}_p + \zeta_1 \mathbf{e}_p + \zeta_2 \int_{t_0}^{t} \mathbf{e}_p \, d\tau \tag{71}$$

and the gains ζ_1 and ζ_2 can be selected such that the eigenvalues of the tracking error equation

$$\ddot{\mathbf{e}}_p + \zeta_1 \dot{\mathbf{e}}_p + \zeta_2 \mathbf{e}_p = 0 \tag{72}$$

have negative real parts. This ensures the global stability of the system when s converges to zero.

Using the PID type s and the same definitions of θ' and θ'', we can derive that

$$\begin{aligned} \mathbf{M}\ddot{\mathbf{e}}_p &= \mathbf{M}\ddot{\theta}_d - \mathbf{M}\ddot{\theta} \\ &= \mathbf{M}[\theta'' - \mathbf{s} - \zeta_1 \dot{\mathbf{e}}_p - \zeta_2 \mathbf{e}_p] - [-\mathbf{B}\dot{\theta} + \hat{\mathbf{M}}\mathbf{q}'' + \hat{\mathbf{B}}\theta'] \\ &= -\mathbf{Y}(\theta, \dot{\theta}, \dot{\theta}_d, \ddot{\theta}_d)\tilde{\mathbf{a}} - (\mathbf{M} + \mathbf{B})\mathbf{s} - \mathbf{M}\zeta_1 \dot{\mathbf{e}}_p - \mathbf{M}\zeta_2 \mathbf{e}_p \end{aligned}$$

where

$$\mathbf{Y}\tilde{\mathbf{a}} = \tilde{\mathbf{M}}\theta'' + \tilde{\mathbf{B}}\theta' \tag{73}$$

$$\tilde{\mathbf{a}} = \hat{\mathbf{a}} - \mathbf{a}. \tag{74}$$

When the same type of Lyapunov function is used

$$V = 1/2\mathbf{s}^T \mathbf{M}\mathbf{s} + 1/2\tilde{\mathbf{a}}^T \Gamma \tilde{\mathbf{a}} \tag{75}$$

then,

$$\begin{aligned} \dot{V} &= 1/2\mathbf{s}^T \dot{\mathbf{M}}\mathbf{s} + \mathbf{s}^T \mathbf{M}\dot{\mathbf{s}} + \tilde{\mathbf{a}}^T \Gamma \dot{\tilde{\mathbf{a}}} \\ &= 1/2\mathbf{s}^T \dot{\mathbf{M}}\mathbf{s} + \mathbf{s}^T M(\ddot{\mathbf{e}}_p + \zeta_1 \dot{\mathbf{e}}_p + \zeta_2 \mathbf{e}_p) + \tilde{\mathbf{a}}^T \Gamma \dot{\tilde{\mathbf{a}}} \\ &= -\mathbf{s}^T \mathbf{M}\mathbf{s} + 1/2\mathbf{s}^T(\dot{\mathbf{M}} - 2\mathbf{B})\mathbf{s} + \tilde{\mathbf{a}}^T(\Gamma \dot{\tilde{\mathbf{a}}} - \mathbf{Y}^T \mathbf{s}). \end{aligned}$$

If adaptation law

$$\dot{\tilde{\mathbf{a}}} = \Gamma^{-1} \mathbf{Y}^T \mathbf{s} \tag{76}$$

is used, then

$$\dot{V} = -\mathbf{s}^T \mathbf{M}\mathbf{s} \leq 0 \tag{77}$$

for all s due to the fact that the matrix $\dot{\mathbf{M}} - 2\mathbf{B}$ is skew-symmetric, and $\zeta_1, \zeta_2 > 0$, and $\mathbf{M}$ is positive definite.

A block diagram of the proposed control algorithm with PD type s is shown in Fig. 2. Our adaptive controller is conceptually simple and easy to implement. This approach does not require the use of joint accelerations and inversion of the inertia matrix. Its computational cost is low because it can be implemented through the use of Newton–Euler recursive formulation. It can be seen from (62), which has the same structure as the computed torque method, that the control law can be computed efficiently using a Newton–Euler formulation once $\hat{\mathbf{a}}$ has been specified. A high gain feedback is not a must for the system stability. With a slight modification, this adaptive approach is applicable for the fixed-base industrial robot control.

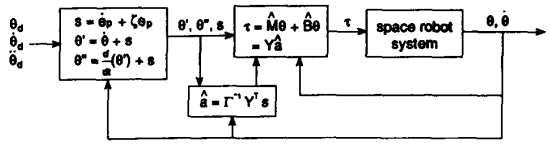


Fig. 2. Block diagram of adaptive controller in joint space.

V. ADAPTIVE CONTROL IN INERTIA SPACE

Conceptually, for most applications, the desired robot hand trajectory, i.e., position, velocity, and acceleration must be specified in inertial space. For example, for catching a moving object, the desired trajectory after catching depends upon the tasks and the motion trajectory of the object before catching, and thus must be specified in inertial space. In other words, as in the case of fixed-base robots for which tasks are normally specified in Cartesian space, tasks in space applications are unlikely to be specified in joint space. Fortunately, the mapping from robot hand displacement in inertial space to joint angles can be uniquely determined for a space robot system when the base attitude is controlled. When the base is completely free flying, this mapping is not uniquely determined [22].

However, the unique kinematics relationship can only be determined when dynamic parameters are given, because this relationship is indeed dynamic-dependent. When some dynamic parameters are unknown, which is indeed the reason why the adaptive control is needed, the mapping cannot be determined. Therefore, the primary difficulty of extending our approach from joint space to inertial space is that the desired trajectory in inertial space cannot be transformed to the desired trajectory in joint space because some dynamic parameters are unknown. In previous discussion, we have utilized a desired trajectory in joint space, as other researchers have done [18], without giving any explanation about how the trajectory is generated.

The problem can be resolved if the same structured adaptive control scheme can be implemented in inertial space. The adaptive control with the same structure, however, is not feasible because the scheme requires that the dynamic model must be linearly parameterized. As has been known, the dynamic related generalized Jacobian of a space robot makes it impossible to suitably choose a set of dynamic parameters such that the inertial space system dynamics can be linearized. That is why the structure of the adaptive controller in joint space is not feasible for adaptive control in inertial space.

We approach the problem in the following way. First, given a trajectory in inertial space, we use an initial estimation of dynamic parameters to compute initial joint trajectory. Then we use the initial joint trajectory and dynamic parameters in the proposed

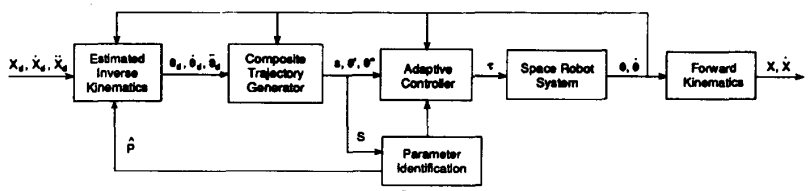


Fig. 3. Block diagram of adaptive control scheme in inertial space.

joint space adaptive control algorithm. After a certain period of time, we update the system dynamic parameters by using new estimated ones in the outer loop of our controller. We can then specify a more precise joint space trajectory based on these new parameters and the inertial space trajectory. Since the inertial space trajectory is uniquely determined by the joint space trajectory and dynamic parameters, it can be shown from the Jacobian relationship that position error in inertial space converges to a given boundary if position errors in joint space and parameter errors are bounded, provided that the robot is not in its singularity configuration. The control scheme is illustrated in Fig. 3.

It is worthwhile to discuss two issues in the implementation of the proposed control scheme. First, to accurately estimate unknown parameters, a persistent excitation (PE) trajectory is required to drive the robot joints. PE trajectories in joint space and in inertial space are not equivalent, because the spectrum of trajectory signal in inertial space is different from the spectrum of the same signal in joint space due to nonlinear kinematic transformation. Therefore, it is of importance to carefully choose an initial trajectory in inertial space such that the same trajectory in joint space is PE. If the PE condition is not satisfied, parameter identification error occurs, although the joint space position errors may still converge.

Second, the updating time for inverse kinematics using the estimated parameters in the outer loop of our controller must be slow enough to maintain system stability. The outer loop, as shown in Fig. 3, is used to update the inverse kinematics and therefore the desired joint trajectory which is used in the joint space adaptive controller. A fast update, especially one using incorrect parameters $\hat{p}_i$, may not guarantee the convergence of parameter errors. In the simulation, the updating time for inverse kinematics is set to 10 s. Simulation results have shown that position errors in inertial space converge to zero as errors in joint space converge to zero and estimated parameters converge to their true values.

In fact, if the updating time for inverse kinematics is long enough, we can also view the control scheme as a two-phase approach: the parameter identification phase and the control phase. That is, estimate dynamic parameter in joint space using the joint space trajectory transformed by the given inertia space trajectory and initial guess of parameters, and then control the system in inertial space once the dynamic parameters have been correctly identified. If the dynamic parameters are estimated ideally, the control phase may also be executed using the model-based dynamic control algorithm such as the one given in [22].

The above approach is computationally simple and efficient in implementation. Although it requires a slow control loop because of parameter estimation, the approach is feasible for space tasks for which the required motion is not too high in general. At the time this study was done, there have been some adaptive control schemes appeared based on the description in inertial space directly, such as the one by Walker and Wee [17] and Gu and Xu [6].

VI. SIMULATION STUDY

In this section, we conduct a case study to show the computation of the proposed algorithms and their feasibility in robot motion control. Though the following discussion is confined to adaptation to mass variation only, our algorithm is also applicable to other parameter adaptation, provided that a set of combinations of those parameters can be chosen such that the dynamics can be linearly expressed in terms of the parameters in which we are interested.

A two-DOF revolute manipulator with link length given by l_1 and l_2 ($l_1 = l_2 = l$) is considered as a lumped-parameter model with point mass m_1 and m_2 at the end of each link. For simplicity, we assume that the base attitude can be successfully controlled so that we need only consider the control of the robot itself. However, it must be pointed out that our adaptive control algorithm can be applied to control the robot motion and the base orientation simultaneously, albeit a more complicated computation. The system model for simulation study is shown in Fig. 4.

At initialization, m_c and $\mathbf{R}_c$ are computed, and they remain unchanged unless a load is added.

$$m_c = m_0 + m_1 + m_2 \tag{78}$$

$$m_c\mathbf{R}_c = m_0\mathbf{R}_0 + m_1\mathbf{R}_1 + m_2\mathbf{R}_2 \tag{79}$$

$$\mathbf{R}_1 = \mathbf{R}_0 + \mathbf{r}_1 \tag{80}$$

$$\mathbf{R}_2 = \mathbf{R}_0 + \mathbf{r}_2. \tag{81}$$

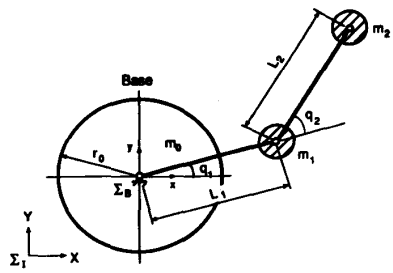


Fig. 4. Planar space robot system model.

When the robot is in motion,

$$\mathbf{R}_0 = \mathbf{R}_c - \frac{m_1}{m_c}\mathbf{r}_1 - \frac{m_2}{m_c}\mathbf{r}_2 \tag{82}$$

$$\mathbf{R}_E = \mathbf{R}_2. \tag{83}$$

The generalized Jacobian is

$$\mathbf{N} = \mathbf{J}_E - \mathbf{J}_c \tag{84}$$

and

$$\mathbf{J}_E = \mathbf{J}_2 \tag{85}$$

$$\mathbf{J}_c = \frac{m_1}{m_c}\mathbf{J}_1 + \frac{m_2}{m_c}\mathbf{J}_2 \tag{86}$$

$$= \frac{l}{m_c}\begin{bmatrix} -(m_1+m_2)s_1 - m_2 s_{12} & -m_2 s_{12} \\ (m_1+m_2)c_1 + m_2 c_{12} & m_2 c_{12} \end{bmatrix} \tag{87}$$

where s and c stand for sine and cosine, e.g., $s_1 = \sin(q_1)$, $c_{12} = \cos(q_1 + q_2)$. The system dynamics has the following form,

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{B}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} = \tau \tag{88}$$

where

$$\mathbf{M} = \mathbf{M}_q - \mathbf{M}_2 \tag{89}$$

$$\mathbf{M}_q = l^2\begin{bmatrix} m_1 + 2m_2(1+c_2) & m_2(1+c_2) \\ m_2(1+c_2) & m_2 \end{bmatrix} \tag{90}$$

$$\mathbf{M}_2 = m_c \mathbf{J}_c^T \mathbf{J}_c \tag{91}$$

$$= \frac{l^2}{m_c}\begin{bmatrix} m^2 + 2m_2(m_1+m_2)(1+c_2) & m_2^2(1+c_2) + m_1 m_2 c_2 \\ m_2^2(1+c_2) + m_1 m_2 c_2 & m_2^2 \end{bmatrix} \tag{92}$$

therefore,

$$\mathbf{M} = \frac{l^2}{m_c}\begin{bmatrix} m_0 m_1 + m_1 m_2 + 2m_0 m_2(1+c_2) & m_1 m_2 + m_0 m_2(1+c_2) \\ m_1 m_2 + m_0 m_2(1+c_2) & m_1 m_2 + m_2 m_0 \end{bmatrix} \tag{93}$$

$$= p_1\mathbf{R}_1 + p_2\mathbf{R}_2 + p_3\mathbf{R}_3 \tag{94}$$

where

$$p_1 = \frac{m_0 m_1}{m_c} \tag{95}$$

$$p_2 = \frac{m_1 m_2}{m_c} \tag{96}$$

$$p_3 = \frac{m_0 m_2}{m_c} \tag{97}$$

$$\mathbf{R}_1 = \begin{bmatrix} l^2 & 0 \\ 0 & 0 \end{bmatrix} \qquad \mathbf{R}_2 = \begin{bmatrix} l^2 & l^2 \\ l^2 & l^2 \end{bmatrix}$$

$$\mathbf{R}_3 = \begin{bmatrix} 2(1+c_2)l^2 & (1+c_2)l^2 \\ (1+c_2)l^2 & l^2 \end{bmatrix}. \tag{98}$$

It is noted that $\mathbf{M}$ is linear in terms of combined dynamic parameters p_1, p_2, and p_3. This example shows that dynamics of the space robot system with an attitude-controlled base can be linearly parameterized in joint space. We also note that m_0, m_1, and m_2 can be uniquely determined by p_1, p_2, and p_3,

$$m_1 = p_1 p_2\left(\frac{1}{p_1} + \frac{1}{p_2} + \frac{1}{p_3}\right) \tag{99}$$

$$m_2 = p_2 p_3\left(\frac{1}{p_1} + \frac{1}{p_2} + \frac{1}{p_3}\right) \tag{100}$$

$$m_0 = p_1 p_3\left(\frac{1}{p_1} + \frac{1}{p_2} + \frac{1}{p_3}\right). \tag{101}$$

The matrix $\mathbf{B}$ is determined by

$$\mathbf{B} = \frac{m_0 m_2}{m_c}\begin{bmatrix} -2l^2 s_2 \dot{q}_2 & -l^2 s_2 \dot{q}_2 \\ l^2 s_2 \dot{q}_1 & 0 \end{bmatrix} = p_3\mathbf{R}_4 \tag{102}$$

where

$$\mathbf{R}_4 = \begin{bmatrix} -2l^2 s_2 \dot{q}_2 & -l^2 s_2 \dot{q}_2 \\ l^2 s_2 \dot{q}_1 & 0 \end{bmatrix}. \tag{103}$$

Our adaptive control law is

$$\tau = \hat{\mathbf{M}}\mathbf{q}'' + \hat{\mathbf{B}}\mathbf{q}' = \mathbf{Y}\hat{\mathbf{a}} \tag{104}$$

$$\mathbf{Y} = [\mathbf{R}_1\mathbf{q}'' \ \mathbf{R}_2\mathbf{q}'' \ \mathbf{R}_3\mathbf{q}'' + \mathbf{R}_4\mathbf{q}'] \tag{105}$$

$$\hat{\mathbf{a}} = \begin{bmatrix} \hat{p}_1 \\ \hat{p}_2 \\ \hat{p}_3 \end{bmatrix} \tag{106}$$

with the following adaptation law

$$\dot{\hat{\mathbf{a}}} = \begin{bmatrix} \gamma_1 \mathbf{s}^T \mathbf{R}_1 \mathbf{q}'' \\ \gamma_2 \mathbf{s}^T \mathbf{R}_2 \mathbf{q}'' \\ \gamma_3 \mathbf{s}^T (\mathbf{R}_3\mathbf{q}'' + \mathbf{R}_4\mathbf{q}') \end{bmatrix}. \tag{107}$$

To study the proposed adaptive algorithms, we use the following common set of conditions:

$$q_{1d} = \frac{\pi}{180}(54 + 6(\sin(t) + \cos(4t))) \tag{108}$$

$$q_{2d} = \frac{\pi}{180}(-126 + 6(\sin(2t) + \cos(6t))) \tag{109}$$

$$\zeta = 10. \tag{110}$$

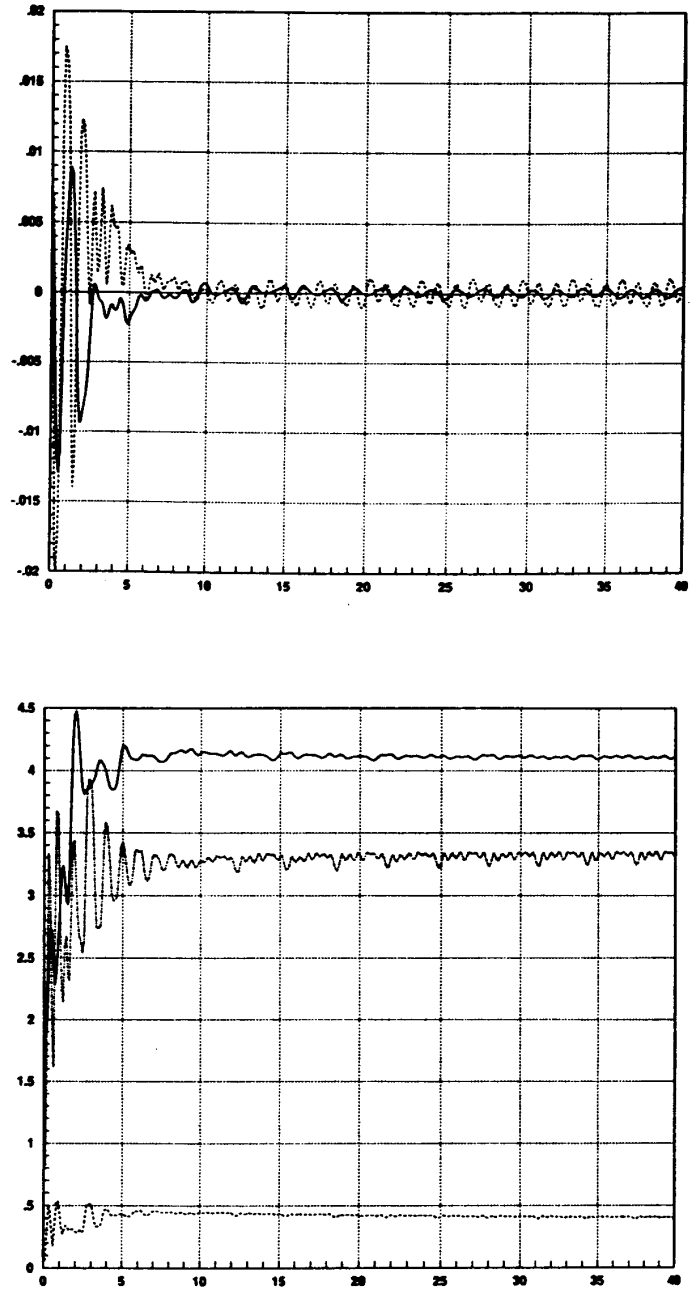


Fig. 5. Tracking errors and parameter estimations using joint space adaptive control.

In the first case we used the following mass parameters, $m_0 = 41$ kg, $m_1 = 5$ kg, $m_2 = 4$ kg, and the initial guess of all three parameters is set to 50% of their true values. It can be found from Fig. 5 that joint errors converge to zero and all parameters converge to their true values 4.1, 0.4, and 3.28 (with small relative errors 1.2%, 2.1%, 2.5%, respectively) after a transient period (approximately 10 s). The results showed the validity and efficiency of the adaptive algorithm proposed.

We then compared the performances of the adaptive controller and dynamic controller without adaptation when there is uncertainty in dynamic parameters. In order to make the dynamic control more favorable, we use 80% of true values as initial estimates of those dynamic parameters. The dynamic control algorithm

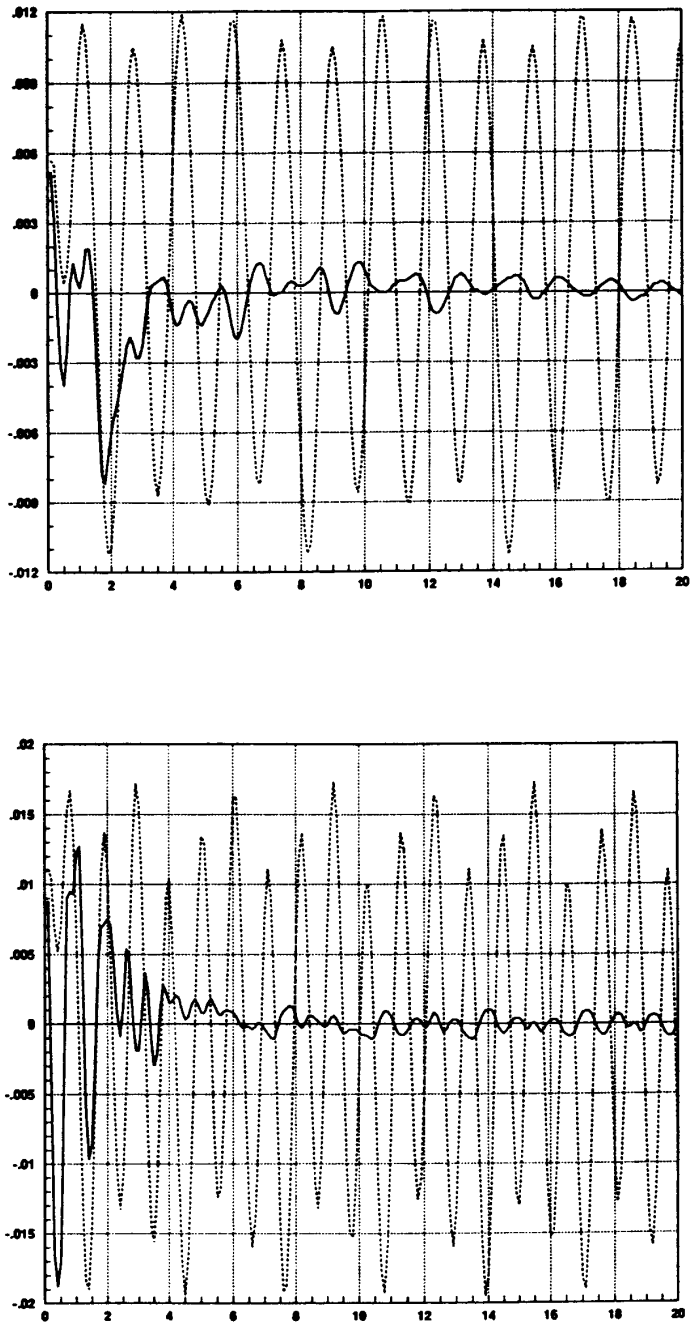


Fig. 6. Comparison between adaptive control and dynamic control.

is based on PD-type structure in joint space without consideration of parameter uncertainty [22]. Fig. 6 gives plots of the variations of two joint position errors by using adaptive control and dynamic control. The adaptive control performance is distinctly superior to the dynamic control response.

To study the effect of mass ratio of the base with respect to the robot, we performed simulation when the base mass is sufficiently large compared with that of the robot. Fig. 7 gives the simulation results when the base mass is 50000 kg. The results have shown that the performance is not sensitive to the mass ratio, and also have shown that the proposed control algorithm is applicable to fixed-base robots.

Fig. 8 shows identification of combined parameters p_1, p_2, and p_3, and the resultant mass m_1, m_2, and

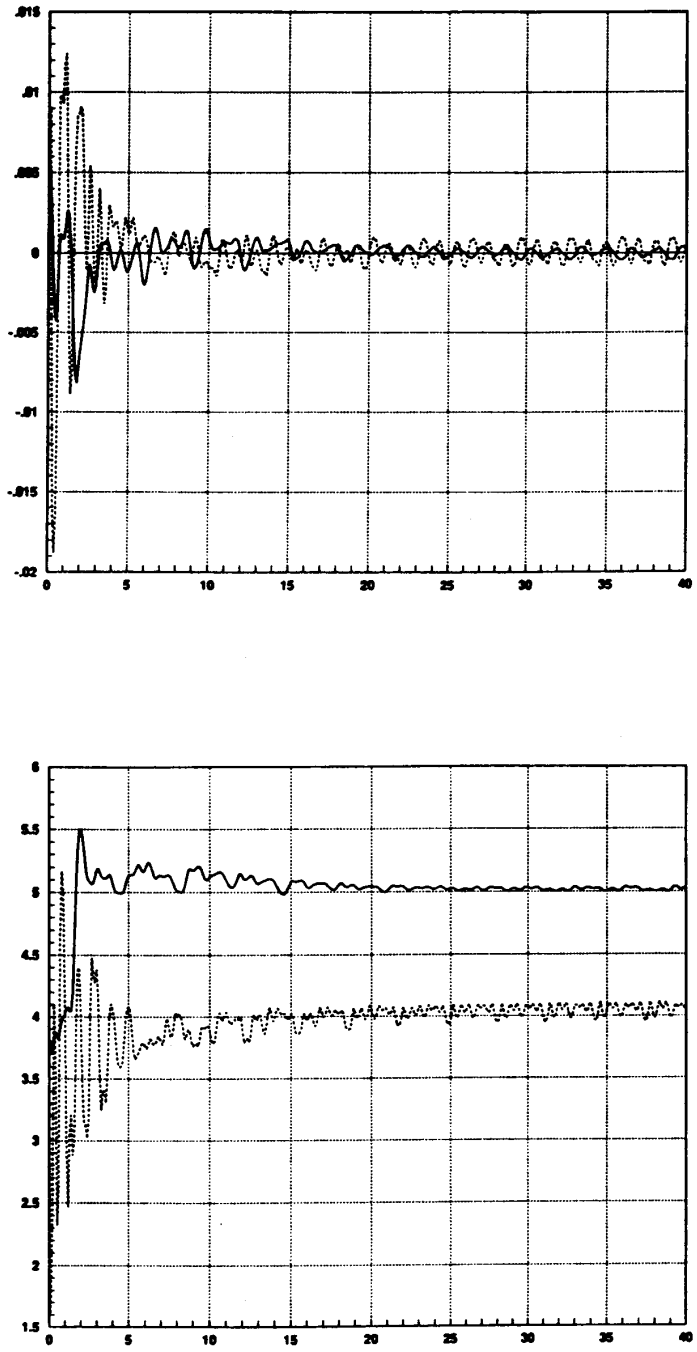


Fig. 7. Example of adaptive control for fixed-based robot.

m_0, in the above case. From Fig. 8 we found that estimation of all parameters m_1, m_2, and m_0 are very close to their true values. This demonstrated that identification of combined dynamic parameters is equivalent to the identification of dynamic parameters m_1, m_2, and m_0, as we discussed previously. It is interesting to note that in Fig. 8 the estimation of nonlinear dynamic parameters p_1, and p_3 converge to m_1 and m_2 due to the fact that the base mass is almost infinite.

In order to compare two different adaptive control algorithms (PD type and PID type), various cases have been tested. For a PE trajectory, both algorithms presented almost identical performances. For a non-PE

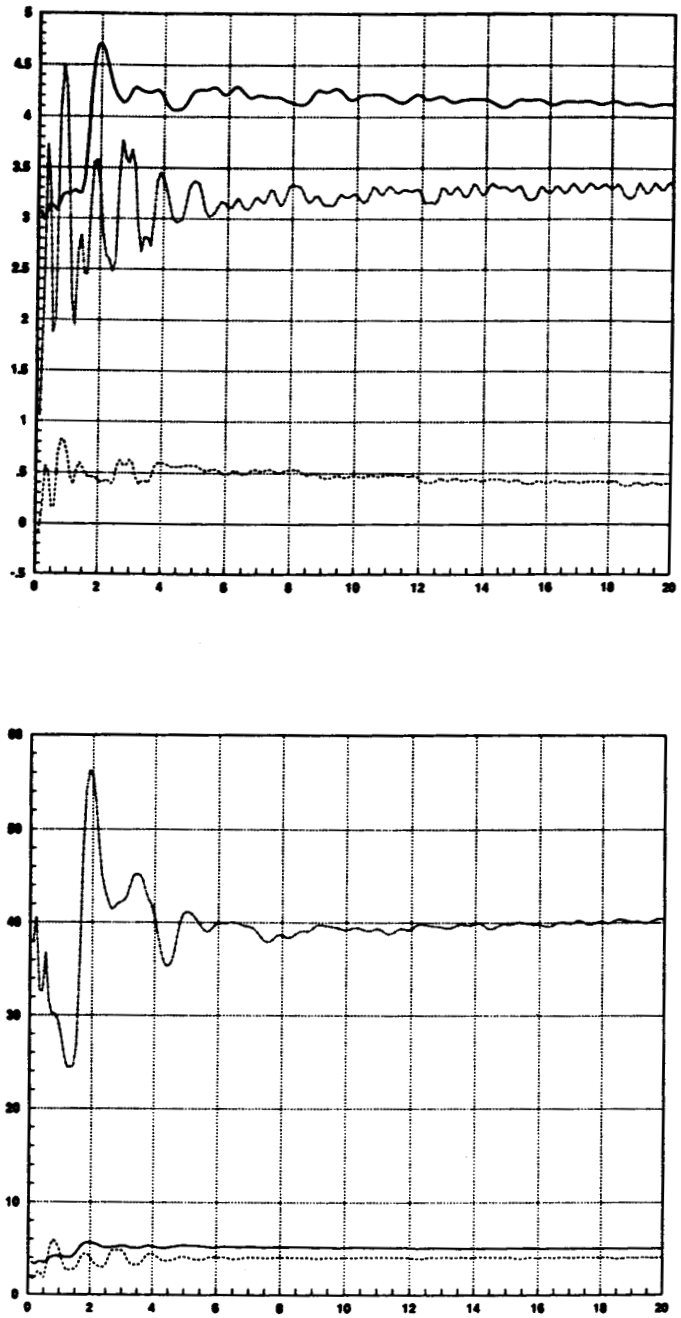


Fig. 8. Illustration of combined dynamic parameter identification.

trajectory, such as

$$q_{1d} = \frac{\pi}{180}(60 - t + 0.05t^2) \tag{111}$$

$$q_{2d} = \frac{\pi}{180}(-120 - t + 0.05t^2) \tag{112}$$

the steady-state performance is improved significantly using PID type adaptive controller, as shown in Fig. 9.

For inertial space adaptive controller, an initial guess of the updating parameters is set to 80% of the true value. The inertial space trajectory and joint space trajectory employed in the simulation are shown

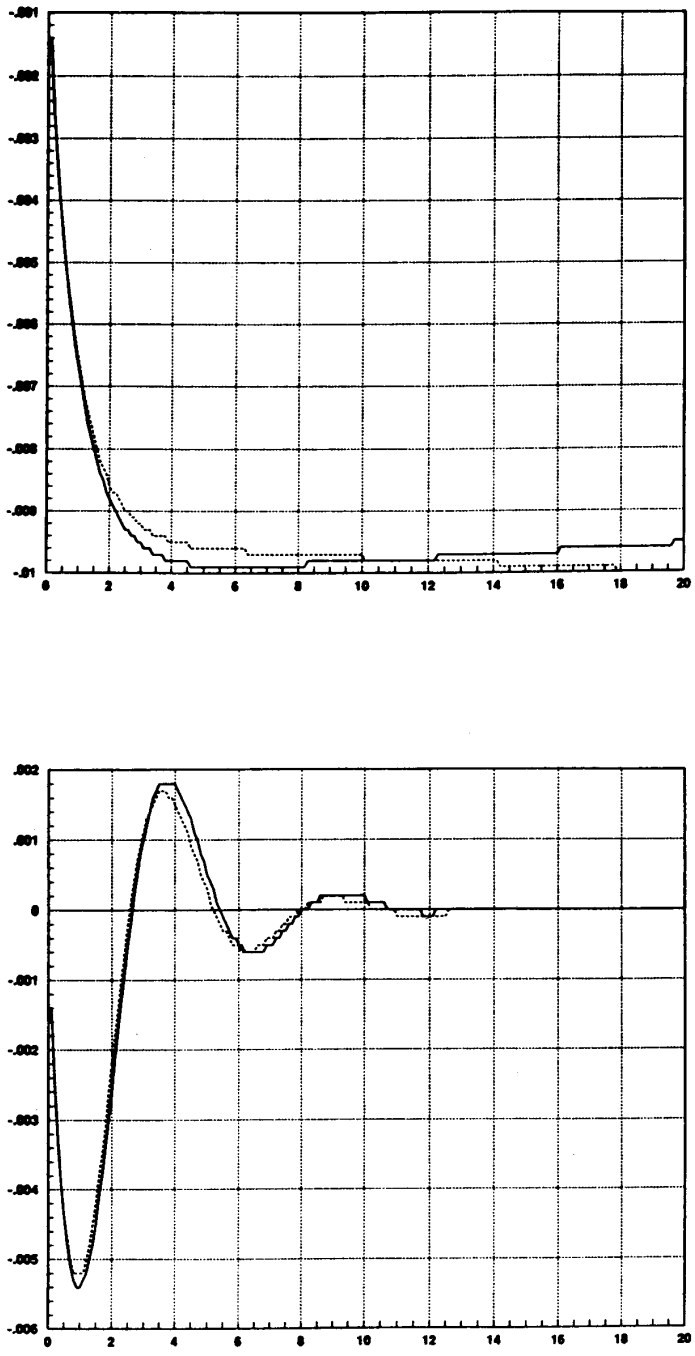


Fig. 9. Comparison between PD type and PID type adaptive control schemes.

in Fig. 10. We used 10 s as updating time for inverse kinematics. The effectiveness of this adaptive scheme has been verified by the tracking errors shown in Fig. 11. It is found that position errors in inertial space converge to zero as errors in joint space converge to zero and estimated parameters converge to their true values.

VII. CONCLUSIONS

Adaptive control is critical for various robotic applications in space, such as material transport and light manipulation, in which robots have to face uncertainty on the dynamic parameters of the load or the structure. Based on Lagrangian dynamics and

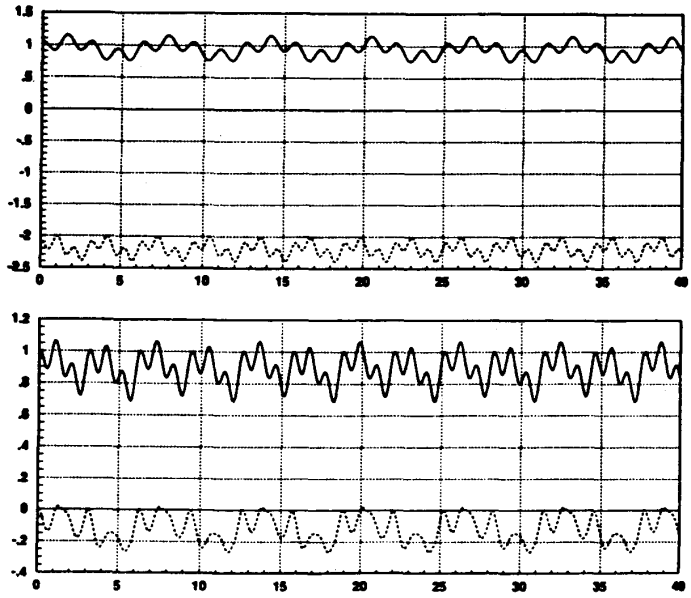


Fig. 10. Trajectories in joint space and in inertia space using inertia space adaptive controller.

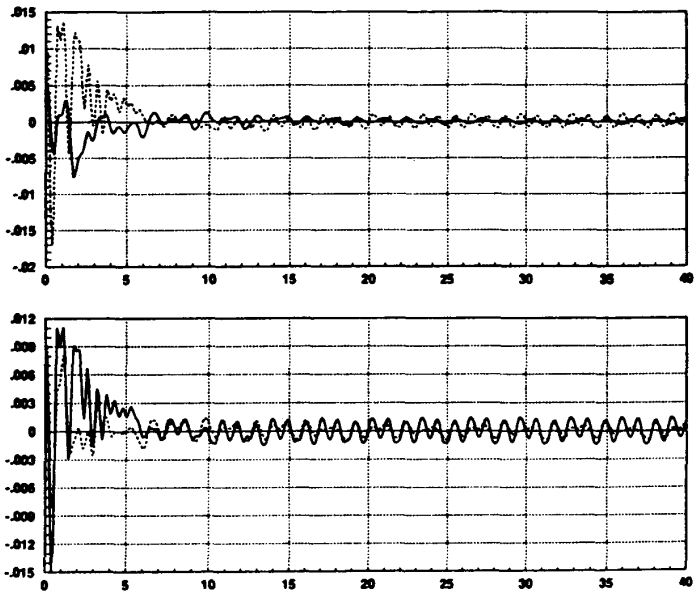


Fig. 11. Tracking errors in joint space and in inertia space using inertia space adaptive controller.

linear momentum conservation law, we derived system dynamic equations. Then we showed that the system dynamics in joint space can be linearly parameterized, i.e., the structure of dynamics equation in joint space can be linearly represented by a set of combined mass/inertia parameters, but the dynamics in inertial space cannot be linearly parameterized.

We proposed an adaptive control scheme in joint space to cope with dynamic uncertainties based on the dynamic model developed. The scheme is effective and feasible for space robot applications because it eliminates the use of joint acceleration measurement, inversion of the inertial matrix, high gain feedback, and considerable computation cost.

Considering that most tasks in space are specified in inertial space, we discussed the issues of adaptive control of the robot for the tasks that must be fulfilled in inertial space. Two main problems have been identified. If the joint adaptive control is implemented, the desired joint trajectory cannot be generated from the given inertia space trajectory since kinematic mapping is dynamics dependent, and thus is subject

to uncertainty in parameters. Moreover, the same structured adaptive control as used in joint space is not feasible for inertial space due to nonlinear parameterization in inertial space. We approached this problem by making use of the proposed joint space adaptive controller while updating the joint trajectory by using the estimated dynamic parameters and the given trajectory in inertial space. In the simulation study, we showed the effectiveness of the proposed method, illustrated the procedure to design the controller, and discussed the implementation issues such as parameter estimation and updating time.

REFERENCES

[1] Atkeson, C. G., An, C. G., and Hollerbach, J. M. (1985)
Estimation of inertial parameters of manipulator loads and links.
Robotics Research 3, 1985.

[2] Bejczy, A. K., and Hannaford, B. (1988)
Man-machine interaction in space telerobotics.
In *Proceedings of the International Symposium of Teleoperation and Control 1988*, 1988.

[3] Craig, J. J., Hsu, P., and Sastry, S. S. (1987)
Adaptive control of mechanical manipulators.
International Journal of Robotics Research, **6** (2), 1987.

[4] Dubowsky, S., and DesForges, D. (1979)
The application of model-referenced adaptive control to robotic manipulators.
ASME Journal of Dynamic Systems, Measurement, and Control, 1979.

[5] Butler, G. (1988)
The 21st century in space: Advances in the astronautical sciences.
American Astronautical Society, 1988.

[6] Gu, Y.-L., and Xu, Y. (1991)
Internal dynamics and adaptive control of a space robot system.
Submitted to *IEEE Transactions on Robotics and Automation*, 1991.

[7] Horowitz, R., and Tomizuka, M. (1980)
An adaptive control scheme for mechanical manipulators.
In *ASME Paper 80-WA/DSC-6*, 1980.

[8] Junkins, J. L. (1990)
Mechanics and control of large flexible structures.
American Institute of Aeronautics and Astronautics, 1990.

[9] Khatib, O., Craig, J. J., and Lozano-Perez, T. (1989)
The Robotics Review.
Cambridge, MA: MIT Press, 1989.

[10] Khosla, P., and Kanade, T. (1985)
Parameter identification of robot dynamics.
In *Proceedings of IEEE International Conference on Decision and Control*, 1985.

[11] Middleton, R. H., and Goodwin, G. C. (1986)
Adaptive computed torque control for rigid link manipulators.
In *Proceedings of IEEE International Conference on Decision and Control*, 1986.

[12] Slotine, J. J., and Li, W. (1987)
On the adaptive control of robot manipulators.
International Journal of Robotics Research, **6** (3), 1987.

[13] Slotine, J. J., and Li, W. (1991)
Applied Nonlinear Control.
Englewood Cliffs, NJ: Prentice-Hall, 1991.

[14] Spong, M. W., and Vidyasagar, M. (1989)
Robot Dynamics and Control.
New York: Wiley, 1989.

[15] Ueno, H., Xu, Y., and et al. (1990)
On control and planning of a space robot walker.
In *Proceedings of 1990 IEEE International Conference on System Engineering*, 1990.

[16] Ullman, M., and Cannon, R. (1989)
Experiments in global navigation and control of free-flying space robot.
In *Proceedings of ASME Winter Conference*, 1989.

[17] Walker, M. W., and Wee, L.-B. (1991)
Adaptive Cartesian coordinates control of space based robot manipulators.
In *Proceedings of American Control Conference*, 1991.

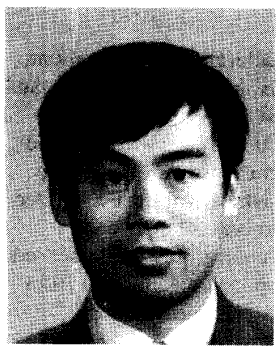
[18] Walker, M. W., and Wee, L.-B. (1991)
An adaptive control strategy for space based robot manipulators.
In *Proceedings of IEEE Conference on Robotics and Automation*, 1991.

[19] Whittaker, W. L., and Kanade, T. (1991)
Space robotics in Japan.
Loyola College, Baltimore, MD, 1991.

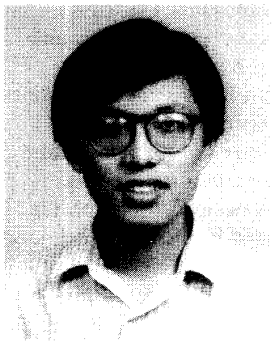
[20] Wiens, G. J., Zarrugh, M. Y., and Scott, R. A. (1992)
Effects of geometric parameters on manipulator dynamic performance.
ASME Journal of Mechanical Design, **114** (1992).

[21] Xu, Y., and Kanade, T. (1992)
Space Robotics: Dynamics and Control.
Boston: Kluwer Academic Publishers, 1992.

[22] Xu, Y., and Shum, H.-Y. (1991)
Dynamic control of a space robot system with no thrust jets controlled base.
Technical report CMU-RI-TR-91-34, The Robotics Institute, Carnegie Mellon University, Philadelphia, PA, 1991.



Yangsheng Xu (S'59—M'90) received his Ph.D. degree from the University of Pennsylvania, PA, in 1989. From 1987 to 1989, he has been associated with the research in the GRASP laboratory at Penn where he developed a robotic compliant wrist system and hybrid position/force control algorithm. Since November 1989, he has been working with the Robotics Institute of Carnegie Mellon University where he is currently a research scientist. At CMU, he has developed a control system for the Self-Mobile Space Manipulator (SM^2) and has been working on the issues related to dynamics and control of space robots, learning control and skill learning, and assembly planning. His current interests include space robotics, intelligent control, teleoperation, and manufacturing.



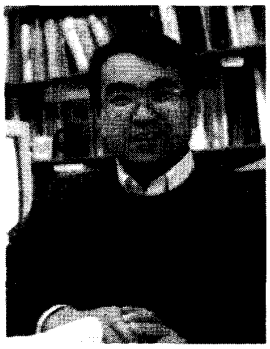
Heung-Yeung Shum (S'93) received the M.Phil. degree from Dept. of Electrical and Electronic Engineering, University of Hong Kong, in 1991.

He is currently a graduate student in the Robotics Ph.D. program, Carnegie Mellon University, Pittsburgh, PA. His research interests include space robotics, 3D robot vision, computer graphics, and virtual reality.

Ju-Jang Lee (M'86) was born in Seoul, Korea, on November 14, 1948. He received the B.S. and M.S. degrees in electrical engineering, both from the Seoul National University in 1973 and 1977, respectively, and the Ph.D. degree in electrical engineering from the University of Wisconsin, Madison, in 1984.

From 1977 to 1978 he was a research engineer at Korean Electric Research & Testing Institute in Korea. From 1978 to 1979 he was a design and processing engineer at the G.T.E. Automatic Electric Co., U.S.A. In 1983 he worked briefly with the Research and Development Department of the Wisconsin Electric Power Co., U.S.A. as a project engineer. Since 1984 he has been working with the Department of Electrical Engineering, Korea Advanced Institute of Science and Technology, where he currently is an Associate Professor. His research interests are in the areas of space robotics, flexible manipulators, variable structure control, intelligent robot control, electronic control unit for automobile, and power system stabilizer.

Dr. Lee is a member of the IEEE Industrial Electronics Society, KIEE, KITE, and KISS.



Takeo Kanade (M'80—SM'88—F'92) received his Doctoral degree in electrical engineering from Kyoto University, Japan, in 1974.

After holding a faculty position at the Dept. of Information Science, Kyoto University, he joined Carnegie Mellon University, Pittsburgh, PA, in 1980, where he is currently Professor of Computer Science and Director of the Robotics Institute. For education in robotics, he established the Robotics Ph.D. program at CMU and is currently the Chairman of the program.

Dr. Kanade is a Founding Fellow of American Association of Artificial Intelligence, the founding editor of *International Journal of Computer Vision*, and an Administrative Committee member of IEEE Robotics and Automation Society. He has received several awards, including the Memorial Paper Award for the 25th Year of Information Processing Society of Japan, and the Marr Prize Award in 1990. Dr. Kanade has served for many government, industry, and university advisory or consultant committees, including the Aeronautics and Space Engineering Board (ASEB) of the National Research Council, NASA's Advanced Technology Advisory Committee (Congressionally mandated committee) and the Advisory Board of the Canadian Institute for Advanced Research.