

THE CATHOLIC UNIVERSITY OF AMERICA  
DEPARTMENT OF ELECTRICAL ENGINEERING

Semiannual Progress Report

*GRANT  
IN-63-CR*

EXPERIMENTAL INVESTIGATION OF  
ADAPTIVE CONTROL OF A  
PARALLEL MANIPULATOR

Grant Number NAG 5-780

*109128*

**Charles C. Nguyen**  
Principal Investigator and Professor  
and

*P.24*

**Sami S. Antrazi**  
Graduate Research Assistant

submitted to  
Dr. Charles E. Campbell, Jr.  
Code 714.1  
Goddard Space Flight Center (NASA)  
Greenbelt, Maryland

July 1992

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>The Stewart Platform-based Manipulator</b>	<b>3</b>
<b>3</b>	<b>SPBM Kinematic Analysis</b>	<b>4</b>
3.1	SPBM Inverse Kinematics . . . . .	4
3.2	SPBM Forward Kinematics . . . . .	6
<b>4</b>	<b>The Joint-Space Adaptive Control Scheme</b>	<b>9</b>
<b>5</b>	<b>Experimental Verification</b>	<b>12</b>
<b>6</b>	<b>Concluding Remarks</b>	<b>15</b>

## SUMMARY

*This report presents the implementation of a joint-space adaptive control scheme used to control non-compliant motion of a Stewart Platform-based Manipulator (SPBM) which is used in a facility called the Hardware Real-Time Emulator (HRTE) developed at Goddard Space Flight Center to emulate space operations. The SPBM is comprised of two platforms and six linear actuators driven by dc motors, and possesses six degrees of freedom. The report briefly reviews the development of the adaptive control scheme which is composed of proportional-derivative (PD) controllers whose gains are adjusted by an adaptation law driven by the errors between the desired and actual trajectories of the SPBM actuator lengths. The derivation of the adaptation law is based on the concept of model reference adaptive control (MRAC) and Lyapunov direct method under the assumption that SPBM motion is slow as compared to the controller adaptation rate. An experimental study is conducted to evaluate the performance of the adaptive control scheme implemented to control the SPBM to track a vertical and circular paths under step changes in payload. Experimental results show that the adaptive control scheme provides superior tracking capability as compared to fixed-gain controllers.*

## 1 Introduction

Closed-kinematic chain manipulators that comprise two platforms coupled together by six linear parallel actuators whose length variations (shortening and extension) produce the motion of one platform relative to another are classified as *Stewart Platform-based (SPB) manipulators*. The development of the above mechanism proposed by Stewart [1] in the design of an aircraft simulator was motivated by disadvantages suffered by conventional anthropomorphic open-kinematic chain (OKC) manipulators whose joints and links are actuated in series. OKC manipulators generally have long reach, large workspace and are capable of entering small spaces because of their compactness. However, they have low stiffness and undesired dynamic characteristics, especially at high speed and large payload mainly due to the cantilever-like structure. In addition, the nonuniform distribution of payload to actuators causes OKC manipulators to have low strength-to-weight ratios. Finally the serial accumulation of joint errors throughout the OKC mechanism results in a relatively large position error on the last link of the manipulator. On the other hand, SPB manipulators whose mechanism are parallel, have been proven to be capable of high precision positioning due to their high structural rigidity and non-serial accumulation of joint errors.

Since its introduction, the Stewart Platform has attracted tremendous attention from researchers [2]-[23] involving robot manipulators, robotic end-effectors and robotic devices for high precision robotic tasks where the requirements of accuracy and sturdiness are more essential than those of large workspace and maneuverability. Dieudonne et al. [2] obtained an actuator extension transformation for a SPB simulator built at NASA Langley Research Center to train aircraft operators. Hunt [3] conducted a systematic study of in-parallel-actuated robot arms and McCallion and Truong [4] considered the mechanism of the Stewart Platform in the design of an automatic assembly table. Sugimoto and Duffy [5] proposed a general technique to describe the instantaneous link motion of a single closed-loop mechanism by employing linear algebra elements to screw systems. Premack and his coworkers [6] built a SPB robot end-effector serving as a testbed for studying autonomous assembly for space operations and the compliance control problem of this end-effector was later investigated by Nguyen and others [9]. Yang and Lee [7] and Fichter [8] studied the kinematic problems and practical construction of SPB manipulators. Sugimoto [10] conducted studies of kinematics and dynamics of parallel manipulators. Nguyen and Pooran [14] conducted kinematic analyses and proposed an algorithm determining the reachable workspace of SPB end-effectors used to perform high and precise motion. Closed-form solutions for forward kinematics were derived by Griffis and Duffy [12] and by Nanua et al. [16] for a class of Stewart Platforms. Waldron et al [13] discussed the kinematics of a hybrid series/parallel six-degree-of-freedom manipulator (DOF) system. Nguyen and Pooran [15] performed a dynamic analysis for a general SPB manipulator. Gosselin and Angeles [17] provided a general classification of different kinds of singularities encountered in closed-kinematic chain mechanism. Nguyen and his coworkers [21] developed computationally efficient kinematic equations and

presented a hardware implementation for a 6 DOF robotic wrist recently built at NASA/Goddard Space Flight Center. In addition, they developed force and kinematic transformations for a force/torque sensor whose implementation was based on the Stewart Platform mechanism [23]. Although much of the research in the literature has devoted extensive effort to the kinematics, dynamics and mechanism design of SPB manipulators, little attention has been paid to the control problem of this type of manipulators. Reboulet and Pigeyre [19] investigated the force/position control of a six DOF SPB micromanipulator. Nguyen, Pooran and Premack [9] proposed a control scheme providing active compliance to SPB manipulators and presented computer simulation results of a 2 DOF parallel manipulator. They then developed a learning control scheme [18] and several trajectory planning schemes [20] for SPB manipulators. In order to effectively react to uncertainties in dynamic modeling and payload encountered in the control of a SPB manipulator, adaptive control schemes whose controller gains are regulated by an adaptation law should be employed in lieu of a fixed-gain control scheme. Numerous adaptive control schemes were developed for OKC manipulators [24]-[28], and only a few for SPB manipulators [11]. Using the concepts of model reference adaptive control (MRAC) and Lyapunov direct method, Nguyen et al. developed a computationally efficient adaptive control scheme for compliant motion control [11] of SPB manipulators and end-effectors. They later extended the developed adaptive control scheme to motion control of redundant manipulators [22].

This report deals with adaptive control of a Stewart Platform-based Manipulator (SPBM) which is an integral part of a facility called the Hardware Real-Time Emulator (HRTE) recently developed at Goddard Space Flight Center (GSFC) to study and emulate space operations. The organization of the report is described as follows. First, the main components of the HRTE and the SPBM are described in the next section. Then, a kinematic analysis is performed for the SPBM to provide a closed-form solution to the inverse kinematics and a computationally efficient solution to the forward kinematics employing the Newton-Raphson Method. After that, the development of a joint-space adaptive control scheme based on the concepts of model reference adaptive control and Lyapunov direct method will be briefly reviewed. The final part of the report presents results of experimental study conducted to evaluate the performance of the adaptive control scheme implemented to control the SPBM to track several test paths.

Matrix and vector notations used in this report are listed below:

- $M^T$ : transpose of the matrix  $M$
- $0_n$ : ( $n \times n$ ) matrix whose elements are all zero
- $I_n$ : ( $n \times n$ ) identity matrix
- $tr[M]$ : trace of matrix  $M$ .
- $diag(\omega_i)$  for  $i=1,2,\dots,6$  = diagonal matrix whose diagonal elements are  $\omega_i$ , for  $i=1,2,\dots,6$ .

## 2 The Stewart Platform-based Manipulator

Figure 1 presents the Hardware Real-Time Emulator (HRTE) developed at the Intelligent Robotic Laboratory (IRL) of the Goddard Space Flight Center (GSFC), to emulate space operations such as berthing and docking performed by a telerobot or by the Space Shuttle Remote Manipulator System (SSRMS). The HRTE is basically composed of a Cincinnati Milacron T3 robot possessing six DOFs and a six DOF Stewart Platform-based Manipulator (SPBM). The SPBM is mounted to the last link of the T3 robot such that one DOF of the T3 coincides with one of the SPBM, yielding a total of 11 DOFs to the HRTE. Currently the HRTE is planned to emulate space hardware docking and berthing performed by the SSRMS. The T3 robot will be used to emulate SSRMS mechanical vibrations produced by external disturbances, and the SPBM to emulate the vibration compensation of a mechanism attached between the SSRMS and the hardware so that a successful berthing/docking can be carried out. The HRTE is currently used as a testbed to investigate the feasibility of a series/parallel manipulator system in which

the T3 robot serving as a serial manipulator is mainly responsible for emulating gross motion while the SPBM serving as a parallel manipulator for emulating fine and precise motion. Testing of autonomous assembly of parts, mating of space-rated connectors and docking of *spline-locking screws* are under way.

Figure 2 shows the main components of the SPBM whose design is based on the Stewart Platform mechanism. It comprises a payload platform, a base platform, six linear actuators and an end-effector attached to the payload platform. Six linear actuators, each of which is composed of a ballscrew assembly mounted axially with a dc motor, link the payload platform and the base platform together. Motion of the payload platform with respect to the base platform is produced by driving the motors to shorten or extend the actuator lengths. Proper coordination of the actuator length trajectories enables the payload platform to perform complex maneuvers. Position sensing is provided by six linear displacement transducers (LDT) mounted along the linear actuators to measure their lengths. Forces/torques exerted by the end-effector are measured through a JR<sup>3</sup> Universal Force-Moment Sensor System mounted between the end-effector base and the payload platform. 2DOF universal joints are used to attach the actuator links to the platforms. The LDT signals are sent to the IRL Local Area Network (LAN) via an Ethernet board and a Data Translation input board which reside in a personal computer (PC) 80486. An IRIS workstation takes the LDT signals off the LAN by means of another Ethernet board, performs necessary coordinate transformations and forward kinematics and then graphically displays in real-time the payload platform pose of the SPBM relative to the base platform. Based upon the desired Cartesian path to be tracked by the SPBM, an Apollo workstation carries out the Cartesian trajectory planning and the SPBM inverse kinematics and sends the desired actuator length trajectories down to the PC. The PC then implements a selected control scheme, such as the joint-space adaptive control scheme presented in the next section, which is driven by tracking errors computed according to the LDT signals and the desired actuator lengths to produce the actuating signals that are finally sent to the dc motor drives via a Data Translation output board.

### 3 SPBM Kinematic Analysis

This section will use vector algebra to attain a closed-form solution for the SPBM inverse kinematics. Then the Newton Raphson's Method will be applied to the nonlinear kinematic equations to obtain a computationally efficient solution for the SPBM forward kinematics.

#### 3.1 SPBM Inverse Kinematics

The inverse kinematics of the SPBM can be formulated so as to determine the required actuator lengths for a given pose <sup>1</sup> of the payload platform with respect to the base platform. Figure 3 shows that two coordinate frames {P} and {B} are assigned to the payload and base platforms of the SPBM, respectively. The origin of Frame {P} is located at the centroid P of the payload platform, the  $z_P$ -axis is pointing outward and the  $x_P$ -axis is perpendicular to the line connecting the two attachment points  $P_1$  and  $P_6$ . The angle between  $P_1$  and  $P_2$  is denoted by  $\theta_P$ . A symmetrical distribution of joints on the payload platform is achieved by setting the angles between  $P_1$  and  $P_3$  and between  $P_3$  and  $P_5$  to  $120^\circ$ . Similarly, Frame {B} has its origin at the centroid B of the base platform. The  $x_B$ -axis is perpendicular to the line connecting the two attachment points  $B_1$  and  $B_6$ , the angle between  $B_1$  and  $B_2$  is denoted by  $\theta_B$ . Moreover, the angles between  $B_1$  and  $B_3$  and between  $B_3$  and  $B_5$  are set to  $120^\circ$  in order to symmetrically distribute the joints on the base platform. The pose of the payload platform is specified by the orientation of Frame {P} with respect to Frame {B} and the position of the origin of Frame {P} with respect to Frame {B}. Now denoting the angle between  $PP_i$  and  $x_P$  by  $\lambda_i$ , and the angle between  $BB_i$  and  $x_B$  by  $\Lambda_i$  for  $i=1,2,\dots,6$ , we obtain

$$\Lambda_i = 60i^\circ - \frac{\theta_B}{2}; \lambda_i = 60i^\circ - \frac{\theta_P}{2}, \text{ for } i = 1, 3, 5 \quad (1)$$

<sup>1</sup>In this report, *pose* implies both Cartesian position and orientation.

and

$$\Lambda_i = \Lambda_{i-1} + \theta_B; \lambda_i = \lambda_{i-1} + \theta_P, \text{ for } i = 2, 4, 6. \quad (2)$$

Furthermore, if Vector  ${}^P\mathbf{p}_i = (p_{ix} \ p_{iy} \ p_{iz})^T$  describes the position of the attachment point  $P_i$  with respect to Frame  $\{\mathbf{P}\}$ , and Vector  ${}^B\mathbf{b}_i = (b_{ix} \ b_{iy} \ b_{iz})^T$  the position of the attachment point  $B_i$  with respect to Frame  $\{\mathbf{B}\}$ , then they can be written as

$${}^P\mathbf{p}_i = [ r_P \cos(\lambda_i) \ r_P \sin(\lambda_i) \ 0 ]^T \quad (3)$$

and

$${}^B\mathbf{b}_i = [ r_B \cos(\Lambda_i) \ r_B \sin(\Lambda_i) \ 0 ]^T \quad (4)$$

for  $i=1,2,\dots,6$  where  $r_P$  and  $r_B$  represent the radii of the payload and base platforms, respectively.

We proceed to consider the vector diagram for an  $i$ th actuator given in Figure 4. The position of Frame  $\{\mathbf{P}\}$  is represented by Vector  ${}^B\mathbf{d} = [x \ y \ z]^T$  which contains the Cartesian coordinates  $x, y, z$  of the origin of Frame  $\{\mathbf{P}\}$  with respect to Frame  $\{\mathbf{B}\}$ . The length vector  ${}^B\mathbf{q}_i = (q_{ix} \ q_{iy} \ q_{iz})^T$ , expressed with respect to Frame  $\{\mathbf{B}\}$  can be computed by

$${}^B\mathbf{q}_i = {}^B\mathbf{x}_i + {}^B\mathbf{p}_i \quad (5)$$

where

$${}^B\mathbf{x}_i = {}^B\mathbf{d} - {}^B\mathbf{b}_i \quad (6)$$

$$= \begin{bmatrix} x - b_{ix} \\ y - b_{iy} \\ z - b_{iz} \end{bmatrix} = \begin{bmatrix} x - b_{ix} \\ y - b_{iy} \\ z \end{bmatrix} = \begin{bmatrix} \bar{x}_i \\ \bar{y}_i \\ \bar{z}_i \end{bmatrix} \quad (7)$$

which is a shifted vector of  ${}^B\mathbf{d}$  and

$${}^B\mathbf{p}_i = {}^B\mathbf{R} \ {}^P\mathbf{p}_i \quad (8)$$

$$= \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} p_{ix} \\ p_{iy} \\ p_{iz} \end{bmatrix} = \begin{bmatrix} r_{11}p_{ix} + r_{12}p_{iy} \\ r_{21}p_{ix} + r_{22}p_{iy} \\ r_{31}p_{ix} + r_{32}p_{iy} \end{bmatrix} = \begin{bmatrix} u_i \\ v_i \\ w_i \end{bmatrix} \quad (9)$$

which is the representation of  ${}^B\mathbf{p}_i$  in Frame  $\{\mathbf{B}\}$  and  ${}^B\mathbf{R}$  is the *Orientation Matrix* specifying the orientation of Frame  $\{\mathbf{P}\}$  with respect to Frame  $\{\mathbf{B}\}$ .

Thus the length  $l_i$  of Vector  ${}^B\mathbf{q}_i$  can be computed from its components as

$$l_i = \sqrt{q_{ix}^2 + q_{iy}^2 + q_{iz}^2}. \quad (10)$$

or

$$l_i = \sqrt{(\bar{x}_i + u_i)^2 + (\bar{y}_i + v_i)^2 + (\bar{z}_i + w_i)^2} \quad (11)$$

We obtain from (3)-(4)

$$p_{ix}^2 + p_{iy}^2 + p_{iz}^2 = r_P^2, \quad (12)$$

$$b_{ix}^2 + b_{iy}^2 + b_{iz}^2 = r_B^2. \quad (13)$$

and from the properties of orientation matrix

$$r_{11}^2 + r_{21}^2 + r_{31}^2 = r_{12}^2 + r_{22}^2 + r_{32}^2 = r_{13}^2 + r_{23}^2 + r_{33}^2 = 1 \quad (14)$$

and

$$\begin{aligned} r_{11}r_{12} + r_{21}r_{22} + r_{31}r_{32} &= 0 \\ r_{11}r_{13} + r_{21}r_{23} + r_{31}r_{33} &= 0 \\ r_{11}r_{13} + r_{22}r_{23} + r_{32}r_{32} &= 0. \end{aligned} \quad (15)$$

Employing (12)-(15), (10) can be rewritten as

$$l_i^2 = x^2 + y^2 + z^2 + r_P^2 + r_B^2 + 2(r_{11}p_{ix} + r_{12}p_{iy})(x - b_{ix}) + 2(r_{21}p_{ix} + r_{22}p_{iy})(y - b_{iy}) + 2(r_{31}p_{ix} + r_{32}p_{iy})z - 2(xb_{ix} + yb_{iy}), \quad (16)$$

for  $i=1,2,\dots,6$ .

Equation (16) represents the *closed-form* solution to the inverse kinematic problem in the sense that required actuator lengths  $l_i$  for  $i=1,2,\dots,6$  can be determined using (16) to yield a given pose of Frame  $\{\mathbf{P}\}$  with respect to Frame  $\{\mathbf{B}\}$ .

The orientation of Frame  $\{\mathbf{P}\}$  with respect to Frame  $\{\mathbf{B}\}$  can be described by the orientation matrix  ${}^B_P\mathbf{R}$  as shown in (9) which requires nine variables  $r_{ij}$  for  $i,j=1,2,3$  from which six are redundant because only three are needed to specify an orientation. There exist several ways to specify an orientation using three variables, but the most widely used one is the Roll-Pitch-Yaw angles  $\alpha$ ,  $\beta$ , and  $\gamma$ , which represent the orientation of Frame  $\{\mathbf{P}\}$ , obtained after the following sequence of rotations from Frame  $\{\mathbf{B}\}$ :

1. First rotate Frame  $\{\mathbf{B}\}$  about the  $\mathbf{x}_B$ -axis an angle  $\gamma$  (*Yaw*)
2. Then rotate the resulting frame about the  $\mathbf{y}_B$ -axis an angle  $\beta$  (*Pitch*)
3. Finally rotate the resulting frame about the  $\mathbf{z}_B$ -axis an angle  $\alpha$  (*Roll*).

The orientation represented by the above Roll-Pitch-Yaw angles is given by <sup>2</sup>

$${}^B_P\mathbf{R} = \mathbf{R}_{RPY} = \begin{bmatrix} c\alpha c\beta & c\alpha s\beta s\gamma - s\alpha c\gamma & c\alpha s\beta c\gamma + s\alpha s\gamma \\ s\alpha c\beta & s\alpha s\beta s\gamma + c\alpha c\gamma & s\alpha s\beta c\gamma - c\alpha s\gamma \\ -s\beta & c\beta s\gamma & c\beta c\gamma \end{bmatrix}. \quad (17)$$

### 3.2 SPBM Forward Kinematics

The forward kinematics of the SPBM deals with the determination of the pose of the payload platform with respect to the base platform based on the actuator lengths  $l_i$  for  $i=1,2,\dots,6$  measured by the six LDTs. In other words, the SPBM forward kinematic problem can be formulated as finding a Cartesian position specified by  $x$ ,  $y$ ,  $z$  and an orientation specified by Roll-Pitch-Yaw angles  $\alpha$ ,  $\beta$ , and  $\gamma$  to satisfy Equation (16) for a given set of actuator lengths  $l_i$  for  $i=1,2,\dots,6$ . The fact that Equation (16) represents a set of six highly nonlinear simultaneous equations with six unknowns results in no closed-form solution for the forward kinematic problem. Consequently, iterative numerical methods must be employed to solve the above set of nonlinear equations. In the following, the implementation of Newton-Raphson method for solving the forward kinematic problem is presented.

In order to apply the Newton-Raphson method, first from (11) we define 6 scalar functions

$$f_i(\mathbf{a}) = (\bar{x}_i + u_i)^2 + (\bar{y}_i + v_i)^2 + (\bar{z}_i + w_i)^2 - l_i^2 = 0 \quad (18)$$

for  $i=1,2,\dots,6$ , where the vector  $\mathbf{a}$  is defined as

$$\mathbf{a} = [ a_1 \ a_2 \ a_3 \ a_4 \ a_5 \ a_6 ]^T = [ x \ y \ z \ \alpha \ \beta \ \gamma ]^T, \quad (19)$$

and then employ the following algorithm to solve for  $\mathbf{a}$ :

#### SPBM Forward Kinematics Algorithm

**Step 1:** Select an initial guess  $\mathbf{a}$ .

**Step 2:** Compute the elements  $r_{ij}$  of  ${}^B_P\mathbf{R}$  using (17) for  $i,j=1,2,3$ .

<sup>2</sup> $c\alpha \equiv \cos \alpha$ , and  $s\alpha \equiv \sin \alpha$ .

**Step 3:** Compute  $\bar{x}_i, \bar{y}_i, \bar{z}_i$ , using (7) and  $u_i, v_i, w_i$  using (9) for  $i=1,2,\dots,6$ .

**Step 4:** Compute  $f_i(\mathbf{a})$  and  $A_{ij} = \frac{\partial f_i}{\partial a_j}$  using (18) for  $i, j=1,2,\dots,6$ .

**Step 5:** Compute  $B_i = -f_i(\mathbf{a})$  for  $i=1,2,\dots,6$ . If  $\sum_{j=1}^6 |B_j| < \text{tol}$  (tolerance), stop and select  $\mathbf{a}$  as the solution.

**Step 6:** Solve  $\sum_{j=1}^6 A_{ij} \delta a_j = B_i$  for  $\delta a_j$  for  $i,j=1,2,\dots,6$  using LU decomposition. If  $\sum_{j=1}^6 \delta a_j < \text{tola}$  (tolerance), stop and select  $\mathbf{a}$  as the solution.

**Step 7:** Select  $\mathbf{a}^{new} = \mathbf{a} + \delta \mathbf{a}$  and repeat Steps 1-7.

It is still unsolved as to how to select an initial guess which ensures convergence of the algorithm. However according to the experiences gained from computer simulation presented in [21] any nonzero initial guess within the reachable workspace of the SPBM will make the algorithm converge. This is probably one of the properties of the Stewart Platform-based mechanisms. Furthermore, the above algorithm as manifested by numerous experimental studies [22], has worked well in a real-time tracking problem where it is employed to compute the actual pose of the payload platform with respect to the base platform based on the actuator lengths. This occurs because the current guess is based on the previous actual pose which is very close to the correct solution provided that the SPBM is tracking the desired path very closely.

In order to minimize the computational time of the SPBM Forward Kinematics Algorithm, the expressions for computing the partial derivatives in Step 4 of the algorithm should be simplified. First using (9) and (17), the partial derivatives of  $u_i, v_i$ , and  $w_i$  with respect to the angles  $\alpha, \beta$ , and  $\gamma$  can be computed as follows:

$$\frac{\partial u_i}{\partial \alpha} = -v_i; \quad \frac{\partial u_i}{\partial \beta} = c\alpha w_i; \quad \frac{\partial u_i}{\partial \gamma} = p_{iy} r_{13}, \quad (20)$$

$$\frac{\partial v_i}{\partial \alpha} = u_i; \quad \frac{\partial v_i}{\partial \beta} = s\alpha w_i; \quad \frac{\partial v_i}{\partial \gamma} = p_{iy} r_{23}, \quad (21)$$

$$\frac{\partial w_i}{\partial \alpha} = 0; \quad \frac{\partial w_i}{\partial \beta} = -(c\beta p_{ix} + s\beta s\gamma p_{iy}); \quad \frac{\partial w_i}{\partial \gamma} = p_{iy} r_{33}. \quad (22)$$

From (7), we note that

$$\frac{\partial \bar{x}_i}{\partial x} = \frac{\partial \bar{y}_i}{\partial y} = \frac{\partial \bar{z}_i}{\partial z} = 1. \quad (23)$$

Employing (20)-(23), we obtain after intensive simplification

$$\frac{\partial f_i}{\partial a_1} = \frac{\partial f_i}{\partial x} = \frac{\partial f_i}{\partial \bar{x}_i} = 2(\bar{x}_i + u_i), \quad (24)$$

$$\frac{\partial f_i}{\partial a_2} = \frac{\partial f_i}{\partial y} = \frac{\partial f_i}{\partial \bar{y}_i} = 2(\bar{y}_i + v_i), \quad (25)$$

$$\frac{\partial f_i}{\partial a_3} = \frac{\partial f_i}{\partial z} = \frac{\partial f_i}{\partial \bar{z}_i} = 2(\bar{z}_i + w_i), \quad (26)$$

$$\frac{\partial f_i}{\partial a_4} = \frac{\partial f_i}{\partial \alpha} = 2(-\bar{x}_i v_i + \bar{y}_i u_i), \quad (27)$$

$$\frac{\partial f_i}{\partial a_5} = \frac{\partial f_i}{\partial \beta} = 2[(-\bar{x}_i c\alpha + \bar{y}_i s\alpha)w_i - (p_{ix} c\beta + p_{iy} s\beta s\gamma)\bar{z}_i] \quad (28)$$

$$\frac{\partial f_i}{\partial a_6} = \frac{\partial f_i}{\partial \gamma} = 2p_{iy}(\bar{x}_i r_{13} + \bar{y}_i r_{23} + \bar{z}_i r_{33}). \quad (29)$$

Conventionally the Manipulator Jacobian Matrix is defined as a matrix relating joint velocities to Cartesian velocities composed of translational velocities and rotational velocities. For the robot wrist, since actuator lengths are selected as joint variables, the time rates of change of actuator lengths  $\dot{l}_1, \dot{l}_2, \dots, \dot{l}_6$  are joint velocities. However in order to utilize the partial derivatives computed for the forward kinematic transformation, we define here the velocities of Cartesian positions of the payload platform with Frame  $\{\mathbf{B}\}$ , namely  $\dot{x}, \dot{y}$  and  $\dot{z}$  as the translational velocities and the velocities of the Roll-Pitch-Yaw angles  $\dot{\alpha}, \dot{\beta}$ , and  $\dot{\gamma}$  as the *rotational velocities*. The matrix  $\mathbf{J}_M$  which relates the length velocities to translation velocities and Roll-Pitch-Yaw angle velocities is therefore called *The Modified Jacobian Matrix*. Denoting

$$\dot{\mathbf{a}} = (\dot{a}_1 \ \dot{a}_2 \ \dot{a}_3 \ \dot{a}_4 \ \dot{a}_5 \ \dot{a}_6)^T = (\dot{x} \ \dot{y} \ \dot{z} \ \dot{\alpha} \ \dot{\beta} \ \dot{\gamma})^T, \quad (30)$$

and

$$\dot{\mathbf{l}} = (\dot{l}_1 \ \dot{l}_2 \ \dot{l}_3 \ \dot{l}_4 \ \dot{l}_5 \ \dot{l}_6)^T, \quad (31)$$

we obtain

$$\dot{\mathbf{a}} = \mathbf{J}_M \dot{\mathbf{l}}, \quad (32)$$

or

$$\dot{\mathbf{l}} = \mathbf{J}_M^{-1} \dot{\mathbf{a}}. \quad (33)$$

where  $\mathbf{J}_M$  is the Modified Jacobian Matrix. Calling  $k_{ij} = \frac{\partial l_i}{\partial a_j}$ , the  $ij$ -element of  $\mathbf{J}_M^{-1}$ , from (33) we have

$$\dot{l}_i = \sum_{j=1}^6 k_{ij} \dot{a}_j = \sum_{j=1}^6 \frac{\partial l_i}{\partial a_j} \dot{a}_j. \quad (34)$$

Now solving for  $l_i^2$  in (18) yields

$$l_i^2 = (\bar{x}_i + u_i)^2 + (\bar{y}_i + v_i)^2 + (\bar{z}_i + w_i)^2 = \bar{f}_i \quad (35)$$

for  $i=1,2,\dots,6$ . Recognizing that  $\bar{f}_i$  is a function of  $\bar{x}_i, \bar{y}_i, \bar{z}_i, \alpha, \beta$ , and  $\gamma$ , and using (23), we differentiate both sides of (34) with respect to time to obtain

$$2l_i \dot{l}_i = \sum_{j=1}^6 \frac{\partial \bar{f}_i}{\partial a_j} \dot{a}_j \quad (36)$$

from which solving for  $\dot{l}_i$  yields

$$\dot{l}_i = \sum_{j=1}^6 \frac{1}{2l_i} \frac{\partial \bar{f}_i}{\partial a_j} \dot{a}_j. \quad (37)$$

Now comparing (34) and (37) and noting from (35) and (18) that  $\frac{\partial \bar{f}_i}{\partial a_j} = \frac{\partial f_i}{\partial a_j}$ , we arrive at

$$k_{ij} = \frac{1}{2l_i} \frac{\partial f_i}{\partial a_j} \quad (38)$$

where  $\frac{\partial f_i}{\partial a_j}$  can be obtained from Step 4 of the Newton-Raphson Algorithm using (24)-(29). In other words, we just showed that the inverse of the Modified Jacobian Matrix can be computed using the results of the forward kinematic transformation.

## 4 The Joint-Space Adaptive Control Scheme

The fact that the SPBM inverse kinematics has a closed-form solution motivates the use of a joint-space control scheme for the SPBM instead of a Cartesian-space control scheme in order to avoid problems associated with the use of the SPBM Forward Kinematics Algorithm in converting LDT signals into actual payload platform pose in the case of a Cartesian-space control scheme. To effectively react to the nonlinearity of the manipulator dynamics, errors in dynamic modeling, and sudden changes in payloads, adaptive controllers are selected instead of fixed-gain controllers which work well only when the manipulator stays within the linearized operating region. Figure 5 illustrates the structure of the adaptive control scheme implemented to control the noncompliant motion of the SPBM. As the figure shows, a Cartesian trajectory planner converts a desired Cartesian path which is specified by initial and final poses and pose velocities and to be carried out by the SPBM into six desired Cartesian trajectories. Then the SPBM inverse kinematics is used to transform the Cartesian trajectories into six desired trajectories of the SPBM actuator lengths which are then compared with the actual length trajectories measured by the LDTs to determine the length errors. Based upon the length errors, proportional-derivative (PD) controllers whose gains are adjusted by an adaptation law to be derived below, control the joint forces of the SPBM actuators such that the length errors remain zero all the time or existing errors decay to zero as quickly as possible.

The dynamics of the SPBM can be described by [15]:

$$\boldsymbol{\tau}(t) = \mathbf{M}(\mathbf{q}, \dot{\mathbf{q}}) \ddot{\mathbf{q}}(t) + \mathbf{N}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}}(t) + \mathbf{G}(\mathbf{q}, \dot{\mathbf{q}}) \mathbf{q}(t) \quad (39)$$

where  $\mathbf{q}(t)$  denotes (6x1) actuator length vector of the SPBM,  $\boldsymbol{\tau}(t)$ , the (6x1) joint force vector,  $\mathbf{M}(\mathbf{q}, \dot{\mathbf{q}})$ , the SPBM (6x6) mass matrix which is symmetric positive-definite,  $\mathbf{N}(\mathbf{q}, \dot{\mathbf{q}})$  and  $\mathbf{G}(\mathbf{q}, \dot{\mathbf{q}})$  are (6x6) matrices whose elements are highly complex nonlinear functions of  $\mathbf{q}$  and  $\dot{\mathbf{q}}$ .

In the right-hand side of (39), under the assumption that the joint friction is negligible, the second term represents the centrifugal and Coriolis forces, and the third term the gravity forces.

From Figure 5, we obtain

$$\boldsymbol{\tau}(t) = \mathbf{K}_p(t) \mathbf{q}_e(t) + \mathbf{K}_d(t) \dot{\mathbf{q}}_e(t) \quad (40)$$

where

$$\mathbf{q}_e(t) = \mathbf{q}_d(t) - \mathbf{q}(t) \quad (41)$$

represents the error vector between the actual length vector  $\mathbf{q}(t)$  and the desired length vector  $\mathbf{q}_d(t)$ ,  $\mathbf{K}_p(t)$  and  $\mathbf{K}_d(t)$  are the matrices of the proportional and derivative terms, respectively, of the adaptive controller.

Substituting (40) into (39) yields

$$\mathbf{M} \ddot{\mathbf{q}}_e + (\mathbf{N} + \mathbf{K}_d) \dot{\mathbf{q}}_e + (\mathbf{G} + \mathbf{K}_p) \mathbf{q}_e = \mathbf{M} \ddot{\mathbf{q}}_d + \mathbf{N} \dot{\mathbf{q}}_d + \mathbf{G} \mathbf{q}_d \quad (42)$$

where the dependent variables of the matrices and vectors were dropped for simplicity.

In order to transform (42) into a state space form, we proceed to define a (12x1) state vector  $\mathbf{z}(t)$  such that

$$\mathbf{z}(t) = [\mathbf{q}_e^T(t) \quad \dot{\mathbf{q}}_e^T(t)]^T, \quad (43)$$

which converts (42) into

$$\dot{\mathbf{z}}(t) = \begin{bmatrix} \mathbf{0}_6 & \mathbf{I}_6 \\ -\mathbf{A}_1 & -\mathbf{A}_2 \end{bmatrix} \mathbf{z}(t) + \begin{bmatrix} \mathbf{0}_6 & \mathbf{0}_6 & \mathbf{0}_6 \\ \mathbf{A}_3 & \mathbf{A}_4 & \mathbf{I}_6 \end{bmatrix} \mathbf{u}(t) \quad (44)$$

where

$$\mathbf{A}_1 = \mathbf{M}^{-1}(\mathbf{G} + \mathbf{K}_p), \quad \mathbf{A}_2 = \mathbf{M}^{-1}(\mathbf{N} + \mathbf{K}_d), \quad (45)$$

and

$$\mathbf{A}_3 = \mathbf{M}^{-1} \mathbf{G}, \quad \mathbf{A}_4 = \mathbf{M}^{-1} \mathbf{N}, \quad (46)$$

and

$$\mathbf{u}(t) = [\mathbf{q}_d^T(t) \quad \dot{\mathbf{q}}_d^T(t) \quad \ddot{\mathbf{q}}_d^T(t)]^T. \quad (47)$$

In the framework of MRAC, Equation (44) represents the *adjustable system*. The desired performance of the SPBM motion can be specified by a *reference model* in terms of the tracking error vector  $\mathbf{q}_e(t) = [q_{e1}(t) \ q_{e2}(t) \ \dots \ q_{e6}(t)]^T$ . Suppose the tracking errors  $q_{ei}(t)$  for  $i=1,2,\dots,6$ , are decoupled from each other, and satisfy

$$\ddot{q}_{ei}(t) + 2 \xi_i \omega_i \dot{q}_{ei}(t) + \omega_i^2 q_{ei}(t) = 0 \quad (48)$$

for  $i=1,2,\dots,6$ , where  $\xi_i$  and  $\omega_i$  denote the damping ratio and the natural frequency of  $q_{ei}$ , respectively. Then the dynamics of the reference model can be represented by

$$\dot{\mathbf{z}}_m(t) = \mathbf{D} \mathbf{z}_m(t) = \begin{bmatrix} \mathbf{0}_6 & \mathbf{I}_6 \\ -\mathbf{D}_1 & -\mathbf{D}_2 \end{bmatrix} \mathbf{z}_m(t), \quad (49)$$

where  $\mathbf{D}_1 = \text{diag}(\omega_i^2)$  and  $\mathbf{D}_2 = \text{diag}(2\xi_i\omega_i)$  are constant (6x6) diagonal matrices, and

$$\mathbf{z}_m(t) = [\mathbf{q}_m^T(t) \quad \dot{\mathbf{q}}_m^T(t)]^T \quad (50)$$

with

$$\mathbf{q}_m = (q_{e1} \ q_{e2} \ \dots \ q_{e6})^T. \quad (51)$$

Solving (49), we obtain

$$\mathbf{z}_m(t) = \exp(\mathbf{D}t) \mathbf{z}_m(0). \quad (52)$$

We note from (52), that if  $\mathbf{z}_m(0) = 0$ , i.e. the initial values of the actual and reference length vectors are identical, then  $\mathbf{z}_m(t) = 0$ .

Now if  $\mathbf{e}(t)$ , the adaptation error vector is defined as

$$\mathbf{e}(t) = \mathbf{z}_m(t) - \mathbf{z}(t), \quad (53)$$

then from (44) and (49), we obtain an error system described by

$$\begin{aligned} \dot{\mathbf{e}}(t) = & \begin{bmatrix} \mathbf{0}_6 & \mathbf{I}_6 \\ -\mathbf{D}_1 & -\mathbf{D}_2 \end{bmatrix} \mathbf{e}(t) + \begin{bmatrix} \mathbf{0}_6 & \mathbf{0}_6 \\ \mathbf{A}_1 - \mathbf{D}_1 & \mathbf{A}_2 - \mathbf{D}_2 \end{bmatrix} \mathbf{z}(t) \\ & + \begin{bmatrix} \mathbf{0}_6 & \mathbf{0}_6 & \mathbf{0}_6 \\ -\mathbf{A}_3 & -\mathbf{A}_4 & -\mathbf{I}_6 \end{bmatrix} \mathbf{u}(t). \end{aligned} \quad (54)$$

We proceed to select a Lyapunov function candidate  $v(t)$  such that

$$\begin{aligned} v(t) = & \mathbf{e}^T \mathbf{P} \mathbf{e} + \text{tr} [(\mathbf{A}_1 - \mathbf{D}_1)^T \mathbf{\Pi}_1 (\mathbf{A}_1 - \mathbf{D}_1)] \\ & + \text{tr} [(\mathbf{A}_2 - \mathbf{D}_2)^T \mathbf{\Pi}_2 (\mathbf{A}_2 - \mathbf{D}_2)] \\ & + \text{tr} [\mathbf{A}_3^T \mathbf{\Pi}_3 \mathbf{A}_3] + \text{tr} [\mathbf{A}_4^T \mathbf{\Pi}_4 \mathbf{A}_4], \end{aligned} \quad (55)$$

where  $\text{tr}[\mathbf{M}]$  is the trace of matrix  $\mathbf{M}$ ,  $\mathbf{P}$  and  $\mathbf{\Pi}_i$  for  $i=1,2,\dots,4$ , are positive definite matrices to be determined later.

Taking the time derivative of (55) and simplifying the resulting expression, we obtain

$$\begin{aligned} \dot{v}(t) = & \mathbf{e}^T (\mathbf{P} \mathbf{D} + \mathbf{D}^T \mathbf{P}) \mathbf{e} \\ & + 2 \text{tr} [(\mathbf{A}_1 - \mathbf{D}_1)^T (\Omega \dot{\mathbf{q}}_e^T + \mathbf{\Pi}_1 \dot{\mathbf{A}}_1)] \\ & + 2 \text{tr} [(\mathbf{A}_2 - \mathbf{D}_2)^T (\Omega \dot{\mathbf{q}}_e^T + \mathbf{\Pi}_2 \dot{\mathbf{A}}_2)] \\ & - 2 \text{tr} [\mathbf{A}_3^T (\Omega \dot{\mathbf{q}}_d^T - \mathbf{\Pi}_3 \dot{\mathbf{A}}_3)] \\ & - 2 \text{tr} [\mathbf{A}_4^T (\Omega \dot{\mathbf{q}}_d^T - \mathbf{\Pi}_4 \dot{\mathbf{A}}_4)] \end{aligned} \quad (56)$$

where

$$\Omega = [P_2 \ P_3]e(t) = -[P_2 \ P_3]z(t) = -P_2q_e - P_3\dot{q}_e \quad (57)$$

and  $\mathbf{P}$  is given by

$$\mathbf{P} = \begin{bmatrix} P_1 & P_2 \\ P_2 & P_3 \end{bmatrix} \quad (58)$$

and it is noted that  $e(t) = -z(t)$  since  $z_m(t) = 0$ .

In (48)  $\xi_i$  and  $\omega_i$  can be selected so that  $\mathbf{D}$  is a matrix having stable eigenvalues, which is also called a *Hurwitz* matrix [24]. Therefore according to *Lyapunov Theorem*[25], for any given positive-definite symmetric matrix  $\mathbf{Q}$ , there exists a positive definite symmetric matrix  $\mathbf{P}$  that satisfies the Lyapunov equation

$$\mathbf{P}\mathbf{D} + \mathbf{D}^T\mathbf{P} = -\mathbf{Q}. \quad (59)$$

Indeed, if  $\mathbf{Q}$  is selected to be

$$\mathbf{Q} = \begin{bmatrix} 2Q_1 & 0_6 \\ 0_6 & 2Q_2 \end{bmatrix} \quad (60)$$

then the submatrices of  $\mathbf{Q}$  can be computed as follows [25]:

$$\mathbf{P}_1 = Q_1\mathbf{D}_2^{-1} + Q_1\mathbf{D}_1^{-1}\mathbf{D}_2 + Q_2\mathbf{D}_2^{-1}\mathbf{D}_1, \quad (61)$$

$$\mathbf{P}_2 = Q_1\mathbf{D}_1^{-1}, \quad (62)$$

$$\mathbf{P}_3 = Q_2\mathbf{D}_2^{-1} + Q_1\mathbf{D}_1^{-1}\mathbf{D}_2^{-1}. \quad (63)$$

Now in (56), if we set

$$\Omega q_e^T + \Pi_1 \dot{A}_1 = \Omega \dot{q}_e^T + \Pi_2 \dot{A}_2 = 0 \quad (64)$$

and

$$\Omega q_d^T - \Pi_3 \dot{A}_3 = \Omega \dot{q}_d^T - \Pi_4 \dot{A}_4 = 0, \quad (65)$$

then (56) becomes

$$\dot{v}(t) = -e^T \mathbf{Q} e \quad (66)$$

which is a negative definite function of  $e(t)$ . Furthermore, from (64)-(65), we obtain

$$\dot{A}_1 = -\Pi_1^{-1} \Omega q_e^T; \quad \dot{A}_2 = -\Pi_2^{-1} \Omega \dot{q}_e^T, \quad (67)$$

and

$$\dot{A}_3 = \Pi_3^{-1} \Omega q_e^T; \quad \dot{A}_4 = \Pi_4^{-1} \Omega \dot{q}_d^T. \quad (68)$$

We already showed that  $\mathbf{P}$  is a positive definite matrix. Now if we could show that  $\Pi_i$  for  $i=1,2,\dots,4$ , are also positive definite matrices, then the error system described in (54) is asymptotically stable, i.e.,  $e(t) \rightarrow 0$ , or  $z(t) \rightarrow z_m$  as  $t \rightarrow \infty$ .

Now assuming that the SPBM performs slowly varying motion,  $\mathbf{M}$ ,  $\mathbf{N}$  and  $\mathbf{G}$  are *slowly time-varying matrices* which can be considered as *nearly constant matrices*. Consequently from (45) and (46) we obtain

$$\dot{A}_1 \simeq \mathbf{M}^{-1} \dot{\mathbf{K}}_p; \quad \dot{A}_2 \simeq \mathbf{M}^{-1} \dot{\mathbf{K}}_d \quad (69)$$

and

$$\dot{A}_3 \simeq 0; \quad \dot{A}_4 \simeq 0. \quad (70)$$

Next substitution of (69)-(70) into (67)-(68) yields

$$\mathbf{M}^{-1} \dot{\mathbf{K}}_p = -\Pi_1^{-1} \Omega q_e^T; \quad \mathbf{M}^{-1} \dot{\mathbf{K}}_d = -\Pi_2^{-1} \Omega \dot{q}_e^T, \quad (71)$$

and

$$0 \simeq \Pi_3^{-1} \Omega q_e^T; \quad 0 \simeq \Pi_4^{-1} \Omega \dot{q}_d^T. \quad (72)$$

Now in (71), letting

$$\mathbf{\Pi}_1 = \frac{1}{\alpha_1} \mathbf{M}; \quad \mathbf{\Pi}_2 = \frac{1}{\alpha_2} \mathbf{M}, \quad (73)$$

where  $\alpha_1$  and  $\alpha_2$  are arbitrary positive scalars, and solving for  $\dot{\mathbf{K}}_p$  and  $\dot{\mathbf{K}}_d$ , we obtain from (71)

$$\dot{\mathbf{K}}_p = -\alpha_1 \mathbf{\Omega} \dot{\mathbf{q}}_e^T, \quad (74)$$

and

$$\dot{\mathbf{K}}_d = -\alpha_2 \mathbf{\Omega} \dot{\mathbf{q}}_e^T. \quad (75)$$

In (73), we note that  $\mathbf{\Pi}_1$  and  $\mathbf{\Pi}_2$  are positive definite matrices that can be considered as nearly constant because the SPBM mass matrix  $\mathbf{M}$  is positive definite and slowly time-varying. To satisfy (72),  $\mathbf{\Pi}_3$  and  $\mathbf{\Pi}_4$  should be chosen such that their determinants approach  $\infty$  in addition to the positive definite property. To achieve this, we can select  $\mathbf{\Pi}_3$  and  $\mathbf{\Pi}_4$  such that they are diagonal matrices whose main diagonal elements assume very large positive values.

Now integrating both sides of (74) and (75) and using (57), results in

$$\mathbf{K}_p(t) = \mathbf{K}_p(0) + \alpha_1 \int_0^t (\mathbf{P}_2 \mathbf{q}_e + \mathbf{P}_3 \dot{\mathbf{q}}_e) \dot{\mathbf{q}}_e^T dt \quad (76)$$

and

$$\mathbf{K}_d(t) = \mathbf{K}_d(0) + \alpha_2 \int_0^t (\mathbf{P}_2 \mathbf{q}_e + \mathbf{P}_3 \dot{\mathbf{q}}_e) \dot{\mathbf{q}}_e^T dt \quad (77)$$

where  $\mathbf{K}_p(0)$  and  $\mathbf{K}_d(0)$  are initial conditions of  $\mathbf{K}_p(t)$  and  $\mathbf{K}_d(t)$ , respectively and can be set arbitrarily.

We observe that (76) and (77) represent the solutions for the controller gain matrices of the joint-space adaptive control scheme which is as illustrated in Figure 5 mainly based on the errors of the actuator lengths of the SPBM and the submatrices of  $\mathbf{P}$ . The computation time required to calculate the adaptive control law given in (40) is relatively small because  $\mathbf{P}$  is a constant matrix and  $\mathbf{q}_e$  can be easily computed from the desired and actual lengths. From the fact that the adaptive control scheme is very computationally efficient, it can be implemented using very high sampling rates. Consequently the assumption of slowly varying motion of the SPBM stated before is valid since the SPBM dynamics is considered *constant* during each sampling interval. Finally, the implementation of the adaptive control scheme does not require on-line computation of the SPBM dynamics which is very computationally intensive.

## 5 Experimental Verification

This section present results of experiments conducted to evaluate the performance of the joint-space adaptive control scheme implemented to control the motion of the SPBM payload platform in the case of sudden change in payload and inertial disturbances. The experiment is arranged so that the payload platform is tied via a long string to a cinder block of about 15 lb. laying on a table. During an upward movement of the payload platform, sudden transition from no payload to full payload occurs at a vertical position which lifts the payload off the table. Reversely, during a downward movement, when the payload platform reaches a vertical position which sets the payload on the table, it creates a transition from full payload to no payload. Referring to Figure 5, for each of the following two study cases, the desired Cartesian path are first modeled by a set of desired Cartesian trajectories which are then converted to a set of desired trajectories of actuator lengths using the SPBM inverse kinematics. Then by *trial and error*, the controller gain matrices  $\mathbf{K}_p$  and  $\mathbf{K}_d$  are adjusted until the payload motion can be completed with the least tracking errors possible under no payload. The obtained controller gain matrices are then used as initial values  $\mathbf{K}_p(0)$  and  $\mathbf{K}_d(0)$  in Equations (76)-(77) which regulate the controller gain matrices

to effectively react to sudden payload changes and dynamic coupling effects. The fixed controller gain matrices used in the study cases are obtained as

$$\mathbf{K}_{p, fixed} = 2.68646 \mathbf{I}_6, \quad \mathbf{K}_{d, fixed} = 80.5 \mathbf{I}_6, \quad (78)$$

and the SPBM parameters are:  $r_P = 10.441 \text{ in.}$ ,  $r_B = 13.838 \text{ in.}$ ,  $\theta_P = 99.20^\circ$ , and  $\theta_B = 16^\circ$ .

Moreover, the desired performance of the payload platform is embodied in the specification of a reference model with the desired damping ratios and natural frequencies which are selected as

$$\omega_i = 10 \text{ rad/sec}; \quad \zeta_i = 0.7 \quad \text{for } i=1,2,\dots,6, \quad (79)$$

which yields  $\mathbf{D}_1 = 100 \mathbf{I}_6$  and  $\mathbf{D}_2 = 14 \mathbf{I}_6$ . Selecting  $\mathbf{Q} = 2 \mathbf{I}_{12}$  and applying Equations (62)-(63),  $\mathbf{P}_2$  and  $\mathbf{P}_3$  are computed as

$$\mathbf{P}_2 = 0.01 \mathbf{I}_6, \quad \mathbf{P}_3 = 0.0721 \mathbf{I}_6. \quad (80)$$

### Study Case 1: Vertical Motion

The objective of this study case is to evaluate the performance of the joint space adaptive control scheme in tracking a vertical motion under sudden changes in payload. The desired vertical path is modeled by

$$z(t) = z_0 + \frac{at^2}{2}, \quad \text{for } 0 \leq t \leq \frac{T}{6} \quad (81)$$

$$z(t) = vt + \frac{vT}{12} + z_0, \quad \text{for } \frac{T}{6} \leq t \leq \frac{5T}{6} \quad (82)$$

$$z(t) = z_0 + \frac{3}{4}vT + vt - \frac{at^2}{2} \quad \text{for } \frac{5T}{6} \leq t \leq T \quad (83)$$

where the travel time  $T = \frac{6}{5} \frac{\Delta}{v}$ , the travel distance  $\Delta z = |z_f - z_0|$ , the acceleration  $a = \frac{vT}{6}$ . The initial position  $z_0$ , the final position  $z_f$  and the velocity  $v$  are specified by the user. All above variables are expressed with respect to the base platform frame  $\{\mathbf{B}\}$ . For this study case we use:  $z_0 = 19.7 \text{ in.}$ ,  $z_f = 14.5 \text{ in.}$ , and  $v = 0.4 \text{ in./sec.}$

Figure 6-13 present the experimental results obtained for the case in which the payload platform is controlled to perform the desired vertical motion under two different sudden payload changes, zero to full payload and full to zero payload. First starting at a vertical position  $z_0 = 19.7 \text{ in.}$ , the payload platform moves upwards and lifts the payload up at a vertical position  $z_m = 17 \text{ in.}$  After that, the payload platform continues to move upwards and stops at a vertical position  $z_f = 14.5 \text{ in.}$  Then the platform is controlled to move down on the table and sets the payload on the table when it reaches  $z_m$ . Figures 6 and 7 present the desired and actual trajectories of the vertical position  $z(t)$  when the fixed gain controller and adaptive controller are applied, respectively. Since the fixed-gain controller was by *trial and error* tailored to zero payload, it produces a steady-state tracking error of approximately  $-0.475 \text{ in.}$  in the case of full payload, as clearly shown in Figure 8. On the other hand, as shown in Figure 9, as soon as the adaptive PD controller senses the change in payload, it adjusts its gains accordingly so that the steady-state tracking error is reduced down to about  $-0.056 \text{ in.}$  According to experimental data obtained for tracking of the vertical position, the fixed-gain controller has a maximum absolute error of  $0.4990 \text{ in.}$ , and an average error of  $0.3859 \text{ in.}$  For the adaptive controller, the maximum absolute error and average error are  $0.1880 \text{ in.}$  and  $0.0950 \text{ in.}$ , respectively. The desired and actual trajectories of the length of the first actuator are presented in Figures 10 and 11, respectively. Figures 12 and 13 present their respective tracking errors. As manifested by Figures 10-13, the adaptive controller provides better performance as compared to the fixed-gain controller in tracking the desired actuator length trajectory. For this case, experimental data show that the fixed-gain controller has a maximum absolute error of  $0.3593 \text{ in.}$ , and an average error of  $0.2073 \text{ in.}$  For the adaptive controller, the maximum absolute error and average error are  $0.1796 \text{ in.}$  and  $0.0361 \text{ in.}$ , respectively. The experimental results of the above study case are tabulated in Table 1.

	Max Errors		Avg Errors	
	Fixed	Adaptive	Fixed	Adaptive
z [in.]	0.4990	0.1880	0.3859	0.0950
l <sub>1</sub> [in.]	0.3593	0.1796	0.2073	0.0361

Table 1: Errors for tracking the vertical motion

## Study Case 2: Circular Motion

The objective of this study case is to evaluate the performance of the joint space adaptive control scheme in tracking a circular motion under sudden change in payload. The circular motion consisting of three segments modeled by

$$x(t) = R \cos \Phi_i; \quad y(t) = R \sin \Phi_i \quad \text{for } t_{i-1} \leq t \leq t_i \quad \text{for } i = 1, 2, 3 \quad (84)$$

where the circular path radius  $R = 3$  in., and

$$\Phi_1(t) = \phi_0 + \frac{\beta}{2}t^2, \quad (85)$$

$$\Phi_2(t) = \phi_1 + \omega(t - t_1), \quad (86)$$

$$\Phi_3(t) = \phi_0 - \frac{\beta}{2}(t_3 - t)^2 \quad (87)$$

with  $\phi_0 = 0$  radian;  $\phi_1 = \Phi_1(t_1)$  radian, angular velocity  $\omega = \beta t_1 = 0.350$  radian/sec and the angular acceleration  $\beta = 2\pi/[t_1(t_3 - t_1)]$  radian/sec<sup>2</sup>.

For both fixed-gain and adaptive controllers, the payload is first placed on two supporting wooden blocks and tied to the payload platform by a string having enough slack so that the payload platform can initially move with zero payload. Then at about 10 seconds after the motion begins, the supporting wooden blocks are removed to produce a payload step change from zero to full payload.

Figure 14 shows the actual motions in the x-y plane tracked by the fixed-gain and adaptive controllers. As shown by the figure, during the period of zero payload, both fixed-gain and adaptive controllers track the desired circle relatively well. However after the introduction of full payload at about the completion of 1/4 of the circle, the fixed-gain controller starts degrading its performance and finally gets totally *off-track* after it completes half of the circle. On the other hand, the adaptive controller adjusts its gain to adapt to the full payload and tracks the circle with relatively small deviation from the desired circle until the end of the motion. Figures 15, 16, and 17 illustrate the tracking errors of coordinates  $z(t)$ ,  $y(t)$  and  $x(t)$ , respectively of the fixed-gain and the adaptive controllers. Computation of recorded data show that the maximum errors of  $z(t)$ ,  $y(t)$ , and  $x(t)$  are 0.2710 in., 0.3490 in., and 0.4370 in., respectively when the fixed-gain controller is applied. Also for the fixed-gain controller, the average errors of  $z(t)$ ,  $y(t)$ , and  $x(t)$  are 0.1261 in., 0.1391 in., and 0.1724 in., respectively. When the adaptive controller is applied, the maximum errors of  $z(t)$ ,  $y(t)$ , and  $x(t)$  are 0.1500 in., 0.2530 in., and 0.2930 in., respectively, and the average errors are 0.0744 in., 0.1060 in., and 0.0941 in., respectively. Figure 18 presents the length trajectories for the sixth actuator and Figure 19 its error trajectory. Recorded data show that for the case of fixed-gain controller, the maximum and average errors are 0.6620 in. and 0.1733 in., respectively. For the case of adaptive controller, the maximum and average errors are 0.1952 in. and 0.0790 in., respectively. We observe that the maximum and average errors of the adaptive controller are smaller than those of the fixed-gain controller. The experimental results of the above study case are tabulated in Table 2.

	Max Errors		Avg Errors	
	Fixed	Adaptive	Fixed	Adaptive
z [in.]	0.2710	0.1500	0.1261	0.0744
y [in.]	0.3490	0.2530	0.1391	0.1060
x [in.]	0.4370	0.2930	0.1724	0.0941
$l_6$ [in.]	0.6620	0.1952	0.1733	0.0790

Table 2: Errors for tracking the circular motion

## 6 Concluding Remarks

This report has dealt with the adaptive control of a Stewart Platform-based manipulator (SPBM) of the Hardware Real-Time Emulator (HIRTE) developed at Goddard Space Flight Center to study and emulate space operations. Kinematic analysis of the manipulator resulted in a closed-form solution for the inverse kinematics whose equations was then applied to develop an iterative solution for the forward kinematics using the Newton Raphson Method. Then a joint-space adaptive control scheme was developed using the concept of model reference adaptive control and the Lyapunov theorem under the assumption of slowly varying motion in the sense that the manipulator motion is slow compared to the adaptation rate of the controllers. The gains of the proportional and derivative controllers of the adaptive control scheme are adjusted by an adaptation law to effectively react to dynamic and static coupling between joints and sudden change in payloads. Being computationally efficient, the adaptation scheme can be implemented for real-time applications using personal computers or workstations. Experiments were conducted to study the performance of the adaptive control scheme in tracking a vertical motion and a circular motion. Experimental results showed that despite step changes in payload, the adaptive controller could track the desired motion with negligible tracking errors while the fixed-gain controller suffered from steady-state error when tracking the vertical motion and got *off-track* when tracking the circular motion. Current research activity focuses on extending the joint-space adaptive control scheme to a Cartesian-space force adaptive control scheme which can be used to provide active compliance to the manipulator. The SPBM is used as an active compliance wrist mounted between the T3 manipulator and a selected mechanism to emulate docking and berthing in space. Experimental results of the current study will be reported in future reports.

## References

- [1] Stewart, D., "A Platform with Six Degrees of Freedom," *Proc. Institute of Mechanical Engineering*, Vol. 180, Part 1, No. 5, pp. 371-386, 1965-1966.
- [2] Dieudonne, J.E. et al, "An Actuator Extension Transformation for a Motion Simulator and an Inverse Transformation Applying Newton-Raphson's Method," *NASA Technical Report D-7067*, 1972.
- [3] Hunt, K.H., *Kinematic Geometry of Mechanisms*, Oxford University, London 1978.
- [4] McCallion, H., and Truong, P.D., "The Analysis of a Six-Degree-of-Freedom Work Station for Mechanized Assembly," *Proc. The Fifth World Congress for the Theory of Machines and Mechanisms, an ASME Publication*, pp. 611-616, 1979.
- [5] Sugimoto, K. and Duffy, J., "Application of Linear Algebra to Screw Systems," *Mech. Mach. Theory*, Vol. 17, No. 1, pp. 73-83, 1982.
- [6] Premack, T. et al, "Design and Implementation of a Compliant Robot with Force Feedback and Strategy Planning Software," *NASA Technical Memorandum 86111*, 1984.
- [7] Yang, D.C. and Lee, T.W., "Feasibility Study of a Platform Type of Robotic Manipulators from a Kinematic Viewpoint," *Trans. ASME Journal of Mechanisms, Transmissions, and Automation in Design*, Vol. 106, pp. 191-198, June 1984.

- [8] Fichter, E.F., "A Stewart Platform-Based Manipulator: General Theory and Practical Construction," *Int. Journal of Robotics Research*, pp. 157-182, Summer 1986
- [9] Nguyen, C.C., Pooran, F.J., and Premack, T., "Control of Robot Manipulator Compliance," in *Recent Trends in Robotics: Modeling, Control and Education*, edited by M. Jamshidi, J.Y.S. Luh, and M. Shahinpoor, North Holland, New York, pp. 237-242, 1986.
- [10] Sugimoto, K., "Kinematic and Dynamic Analysis of Parallel Manipulators by Means of Motor Algebra," *ASME Journal of Mechanisms, Transmissions, and Automation in Design*, pp. 1-5, Dec. 1986.
- [11] Nguyen, C.C., Pooran, F.J., "Adaptive Force/Position Control of Robot Manipulators with Closed-Kinematic Chain Mechanism," in *Robotics and Manufacturing: Recent Trends in Research, Education, and Application*, edited by M. Jamshidi et al, ASME Press, New York, pp. 177-186, 1988.
- [12] Griffis, M., Duffy, J., "A Forward Displacement Analysis of a Class of Stewart Platforms," *Journal of Robotic Systems*, Vol. 6, pp. 703-720, 1989.
- [13] Waldron, K.J., Raghavan, M., and Roth, B., "Kinematics of a Hybrid Series-Parallel Manipulation System," *ASME Journal of Dynamic Systems, Measurement, and Control* Vol. 111, pp. 211-221, 1989.
- [14] Nguyen, C.C., and Pooran, F.J., "Kinematic Analysis and Workspace Determination of A 6 DOF CKCM Robot End-Effector," *Journal of Mechanical Working Technology*, Vol. 20, pp. 283-294, 1989.
- [15] Nguyen, C.C., and Pooran, F.J., "Dynamical Analysis of 6 DOF CKCM Robot End-Effector for Dual-Arm Telerobot Systems," *Journal of Robotics and Autonomous Systems*, Vol. 5, pp. 377-394, 1989.
- [16] Nanua, P., Waldron, K.J., Murthy, V., "Direct Kinematic Solution of a Stewart Platform," *IEEE Trans. Robotics and Automation*, Vol. 6, No. 4, pp. 438-444, 1990.
- [17] Gosselin, C. and Angeles, J., "Singularity Analysis of Closed-Loop Kinematic Chains," *IEEE Transactions on Robotics and Automation*, Vol. 6, No. 3, pp. 281-290, 1990.
- [18] Nguyen, C.C., and Pooran, F.J., "Learning-Based Control of a Closed-Kinematic Chain Robot End-Effector Performing Repetitive Tasks," *International Journal of Microcomputer Applications*, Vol. 9, No. 1, pp. 9-15, 1990.
- [19] Reboulet, C. and Pigeyre, R., "Hybrid Control of a Six-Degree-Of-Freedom In-Parallel Actuated Micro-Manipulator Mounted on a Scara Robot," *Robotics and Manufacturing, Recent Trends in Research, Education, and Applications* edited by Jamshidi, M. and Saif, M., ASME Press, Vol. 3, pp. 293-298, 1990.
- [20] Nguyen, C.C., Zhou, Z-L., Antrazi, S.S., Campbell, C.E., "Experimental Study of Motion Control and Trajectory Planning for a Stewart Platform Robot Manipulator," *Proc., IEEE International Conference on Robotics and Automation*, pp. 1873-1878, Sacramento, California, April 1991.
- [21] Nguyen, C.C., Antrazi, S., Zhou, Z-L, and Campbell, C.E., "Analysis and Implementation of a 6 DOF Stewart-Platform-Based Robotic Wrist," *Computers and Electrical Engineering: An International Journal*, Vol. 17, Number 3, pp. 191-204, 1991.
- [22] Nguyen, C.C., Zhou, Z.L., Mosier, G.E., "Cartesian-Space Control of Redundant Manipulators using a Computationally Efficient Adaptive Control Scheme," *Robotics and Computer-Integrated Manufacturing: An International Journal*, Volume 9, No. 2, 1992.
- [23] Nguyen, C.C., Antrazi, S.S., Park, J-Y., and Z-L. Zhou, "Analysis and Experimentation of a Stewart Platform-Based Force/Torque Sensor," *International Journal of Robotics and Automation*, Volume 7, No. 2, 1992.
- [24] Landau, Y.D., *Adaptive Control: The Model Reference Approach*, Marcell Dekker, 1979.
- [25] Seraji, H., "A New Approach to Adaptive Control of Manipulators," *ASME Journal of Dynamic Systems, Measurement, and Control*, Vol. 109, pp. 193-202, 1987.
- [26] Pourboghraat, F., "Virtual Adaptive Compliant Control for Robots," *International Journal of Robotics and Automation*, Vol. 4, No. 3, pp. 148-157, 1989.
- [27] Johansson, R., "Adaptive Control of Robot Manipulator Motion," *IEEE Transactions on Robotics and Automation*, Vol. 6, No. 4, pp. 483-490, 1990.
- [28] Walker, M.W. and Wee, L-B, "Adaptive Control of Space-Based Robot Manipulators," *IEEE Transactions on Robotics and Automation*, Vol. 7, No. 6, pp. 828-834, 1991.

Figure 1: The Hardware Real-Time Emulator (HRTE) →

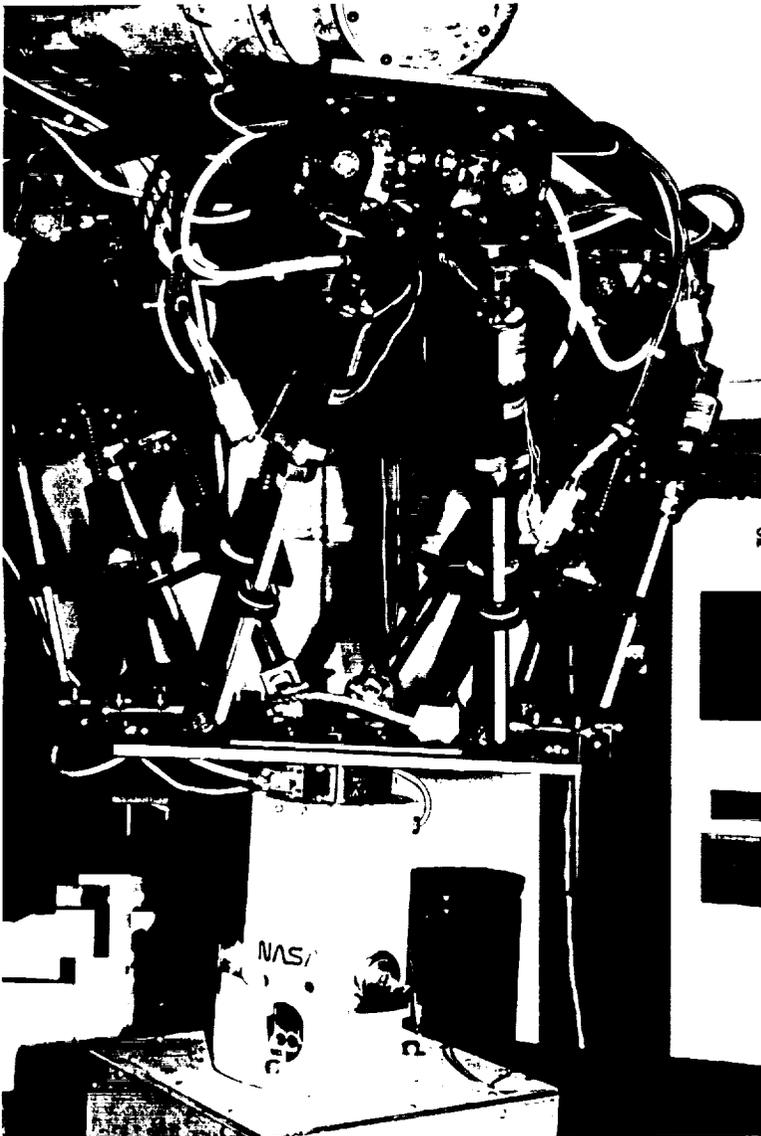
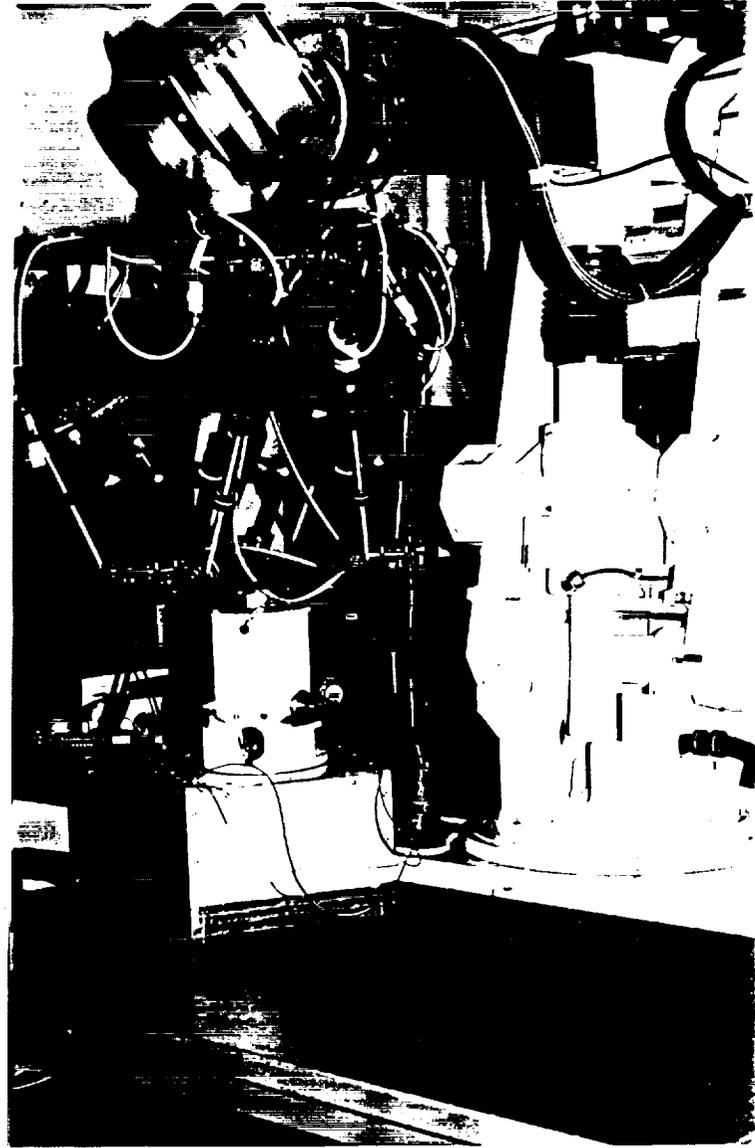


Figure 2: The Stewart Platform-based Manipulator (SPBM) ←

ORIGINAL PAGE IS  
OF POOR QUALITY

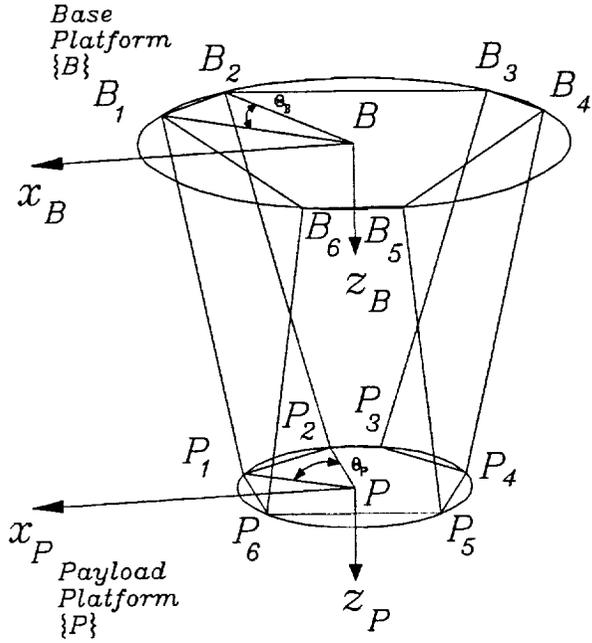


Figure 3: Platform frame assignment

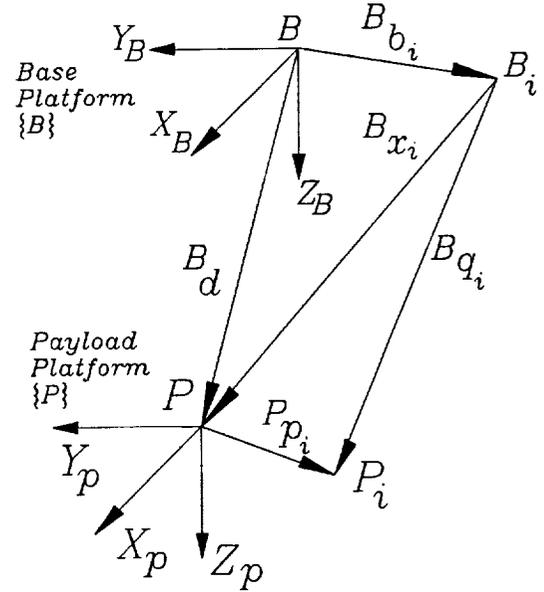


Figure 4: Vector diagram for the  $i$ th actuator

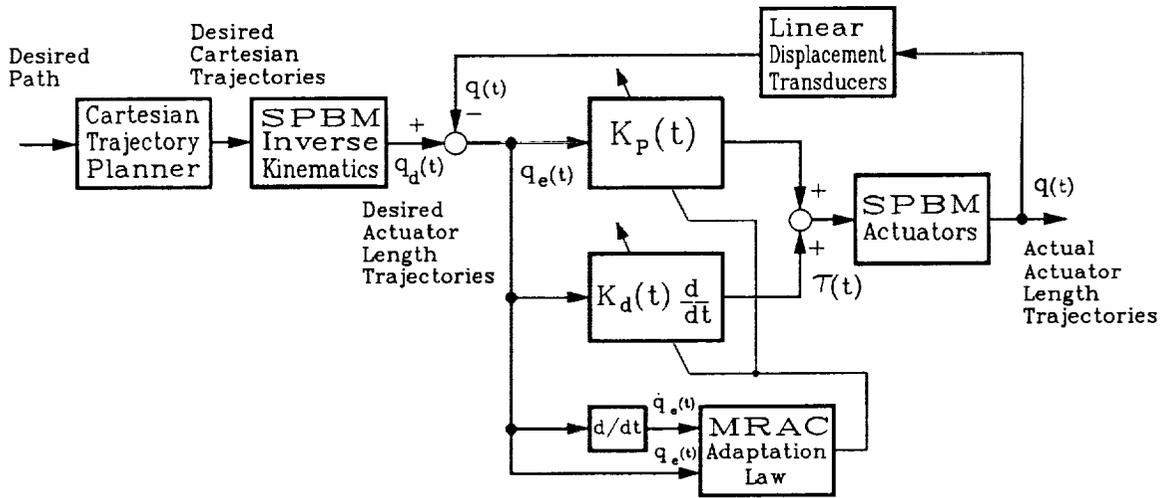
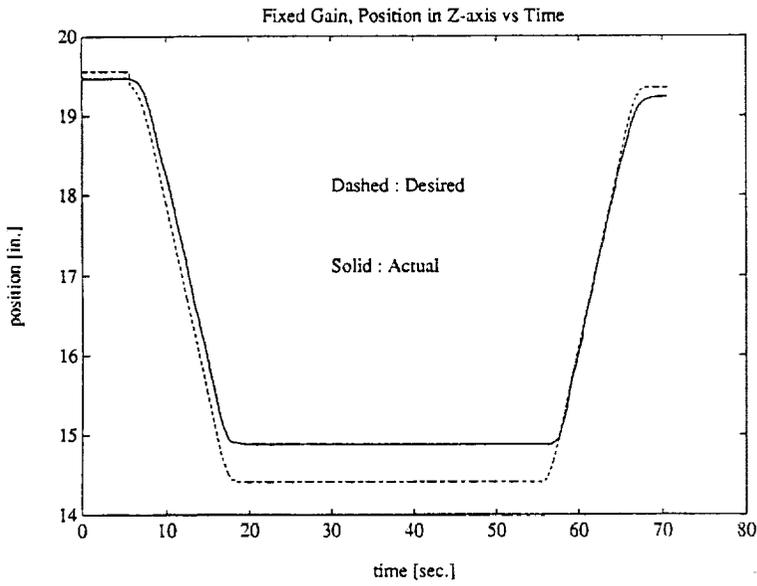
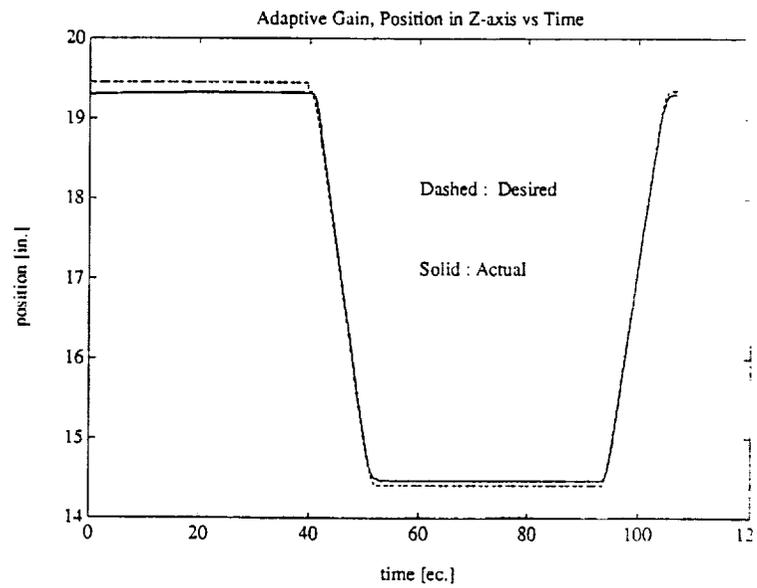


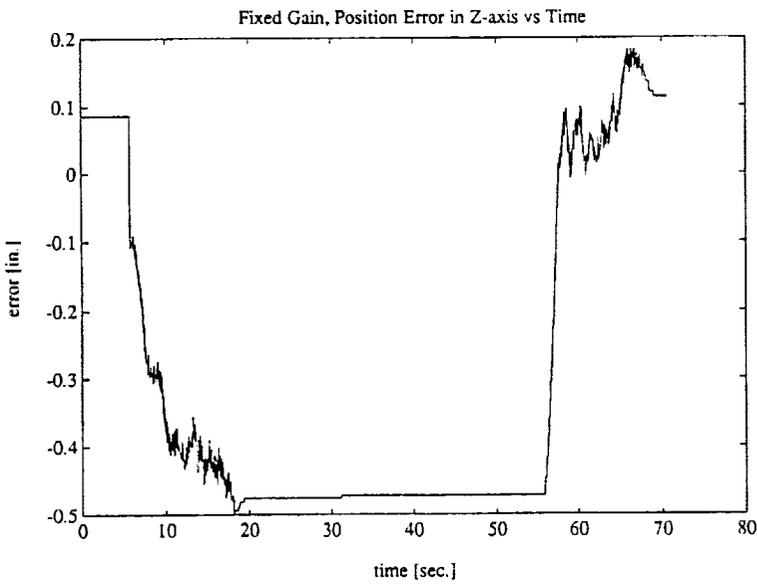
Figure 5: The joint-space adaptive control scheme for SPBM



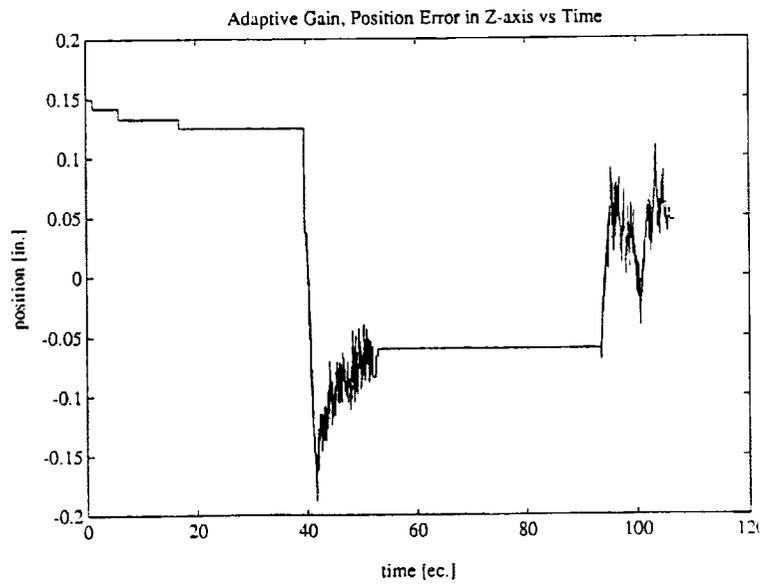
**Figure 6:** Vertical position trajectories (fixed-gain controller)



**Figure 7:** Vertical position trajectories (adaptive controller)



**Figure 8:** Vertical position error trajectories (fixed-gain controller)



**Figure 9:** Vertical position error trajectories (adaptive controller)

Fixed Gain Leg Length vs Time

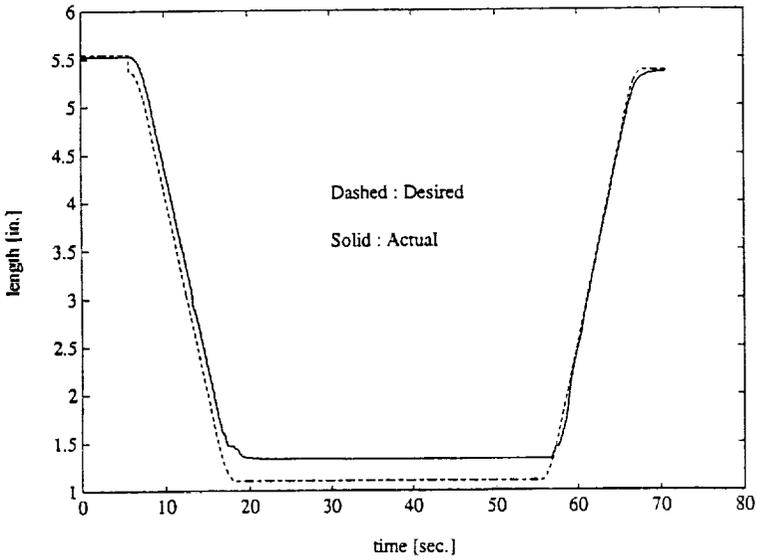


Figure 10: Actuator length trajectories (fixed-gain controller)

Adaptive Gain Leg Length vs Time

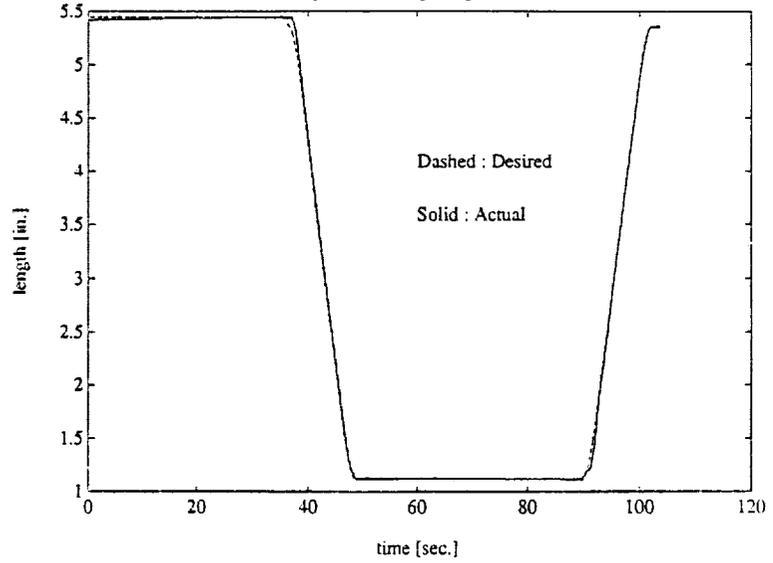


Figure 11: Actuator length trajectories (adaptive controller)

Fixed Gain Leg Length Error vs Time

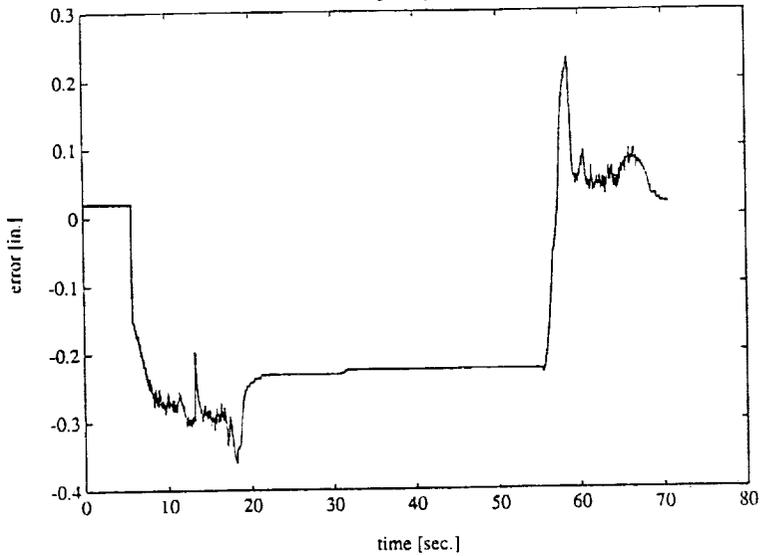


Figure 12: Actuator length error trajectories (fixed-gain controller)

Adaptive Gain Leg Length Error vs Time

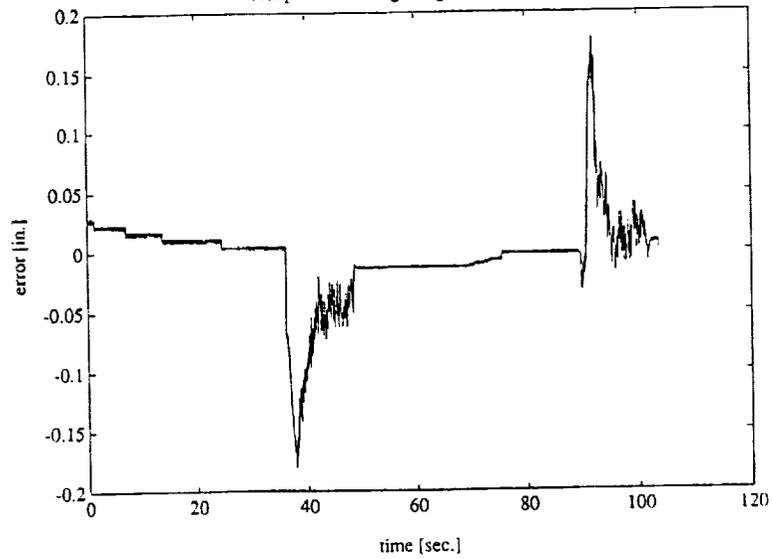
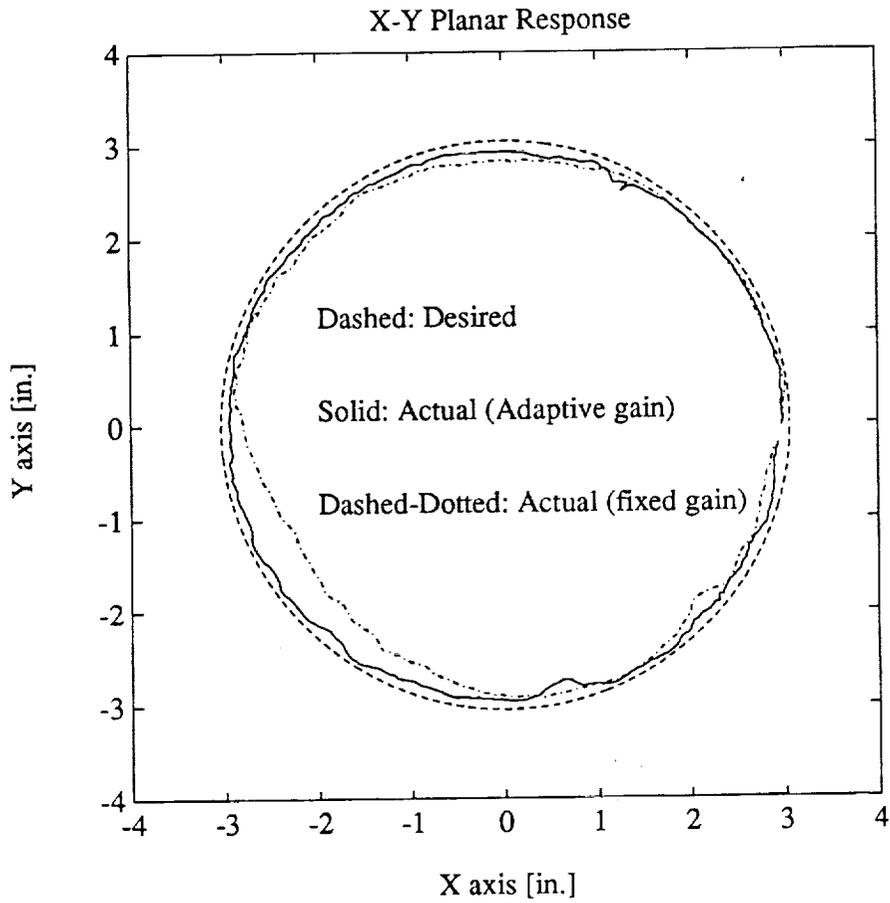
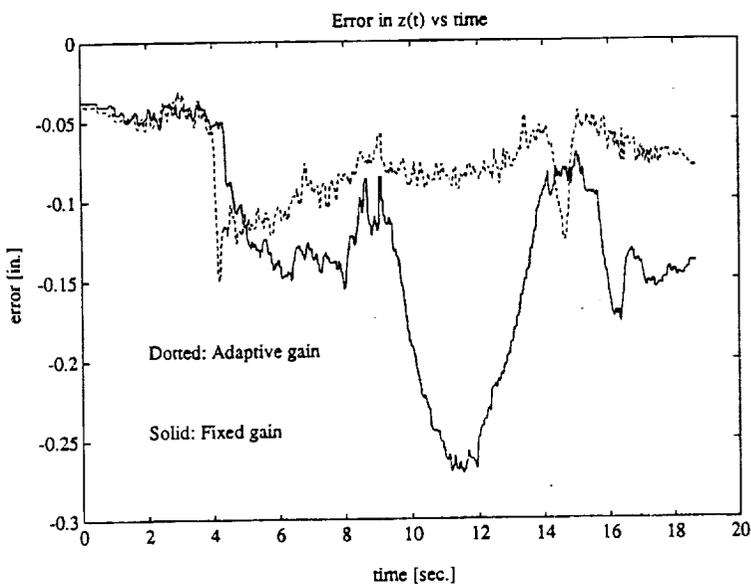


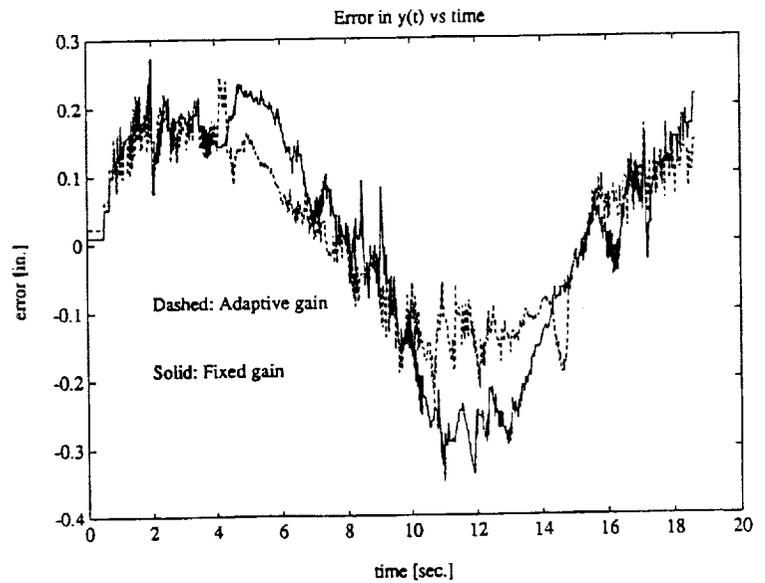
Figure 13: Actuator length error trajectories (adaptive controller)



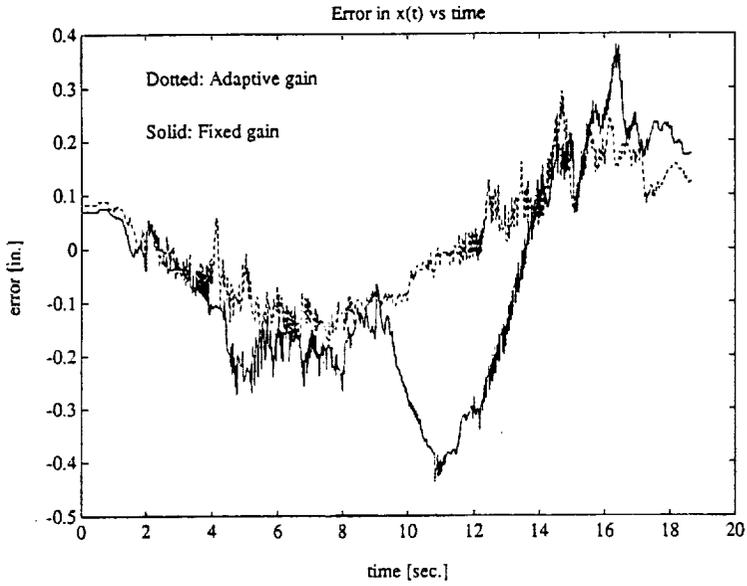
**Figure 14:** x-y planar response of tracking the circular motion



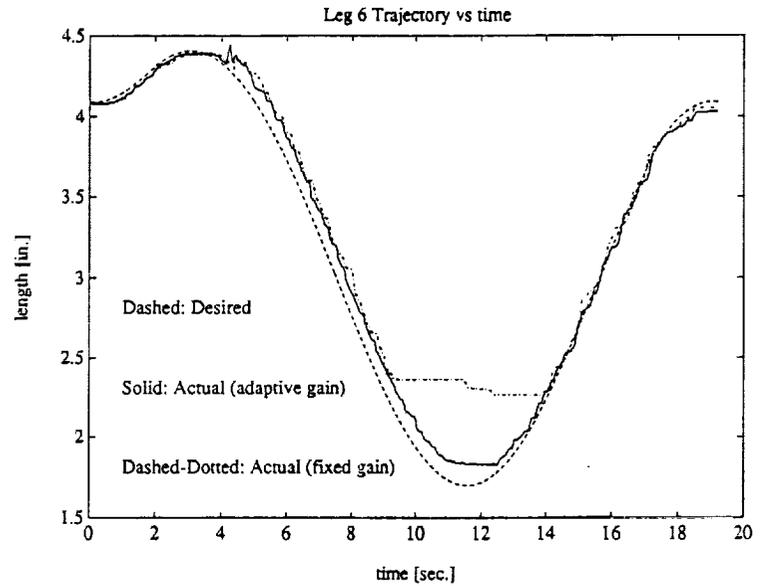
**Figure 15:** Trajectories of position errors in z(t)



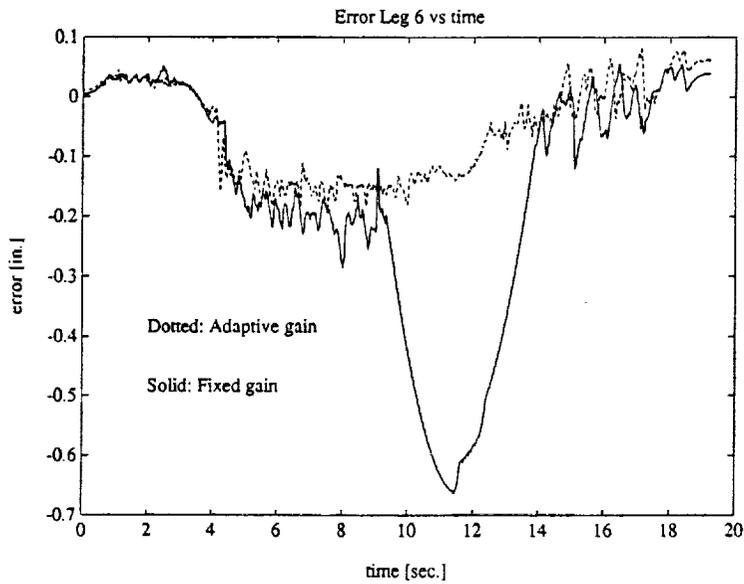
**Figure 16:** Trajectories of position errors in y(t)



**Figure 17:** Trajectories of position errors in  $x(t)$



**Figure 18:** Trajectories of the sixth actuator length



**Figure 19:** Trajectories of the sixth actuator length errors