Université d'Ottawa · University of Ottawa

Université d'Ottawa · University of Ottawa

FACULTÉ DES ÉTUDES SUPÉRIEURES ET POSTDOCTORALES

FACULTY OF GRADUATE AND POSTDOCTORAL STUDIES

Kang WEN

AUTEUR DE LA THÈSE - AUTHOR OF THESIS

M. A. Sc. (Mechanical Engineering)

GRADE - DEGREE

Department of Mechanical Engineering

FACULTÉ, ÉCOLE, DÉPARTEMENT - FACULTY, SCHOOL, DEPARTMENT

TITRE DE LA THÈSE - TITLE OF THE THESIS

A Model Based Impedance Control for Haptic Interface Development Using a Hardware-in-the-loop Experimental Set-up

D. Necsulescu

DIRECTEUR DE LA THÈSE - THESIS SUPERVISOR

CO-DIRECTEUR DE LA THÈSE - THESIS CO-SUPERVISOR

EXAMINATEURS DE LA THÈSE - THESIS EXAMINERS

M. Liang

J. Sasiadek

J.-M. De Koninck, Ph.D.

LE DOYEN DE LA FACULTÉ DES ÉTUDES SUPÉRIEURES ET POSTDOCTORALES

DEAN OF THE FACULTY OF GRADUATE AND POSTDOCTORAL STUDIES

MODEL BASED IMPEDANCE CONTROL FOR HAPTIC INTERFACE DEVELOPMENT USING HARDWARE-IN-THE LOOP EXPERIMENTAL SETUP

Kang Wen

A thesis submitted to the Faculty of Graduate and Postdoctoral Studies in partial fulfillment of the requirements for the degree of

MASTER OF APPLIED SCIENCE

In Mechanical Engineering

Ottawa-Carleton Institute for Mechanical and Aerospace Engineering

University of Ottawa

Ottawa, Canada

April 2004

Your file Votre référence
ISBN: 0-494-01640-X
Our file Notre référence
ISBN: 0-494-01640-X

ABSTRACT

Haptic interfacing provides the means through which human operators can interact with virtual environments. A haptic interface contains software simulated virtual environments, controllers and haptic devices. Impedance control is investigated in this thesis from the viewpoint of suitability for haptic interface development. The difference between conventional application of impedance control in robot motion control and its application in haptic interface development is also investigated. A model based impedance control methodology was developed for haptic controller design and proved feasible for defining the interaction between a human operator and a virtual environment. The factors that could have effects to the performance of a haptic interface were also investigated experimentally using parametric studies. The investigation was carried out on a HIL (Hardware-In-the-Loop) experimental setup, which combines the real hardware and mathematically simulated components. This HIL experimental set up can be used as a generic platform for the development and testing haptic interface options before prototypes are built.

Acknowledgement

I would like to express my heartfelt appreciation to all the people who helped me on my thesis work.

Particularly, I would like to thank my supervisor Dr. Dan Necsulescu for his expertise, guidance, enlightenment, encouragement, as well as the time he spent on this research. He has been showing that he is very knowledgeable, dependable and responsive throughout my study.

And I am also very grateful to my lab colleagues and friends, such as Goran and Cheng, for their supporting and sharing.

Many thanks to the technicians working in the machine shop of this department.

In particular, I would like to thank my wife, Yan, for her support, understanding and patience throughout my whole study.

Cordially yours,

Kang Wen

Table of Contents

List of Figures

List of Tables

List of Symbols

A	A matrix in robotic forward kinematics
B	coefficient of viscous friction
F_{ext}	interaction force in conventional robot motion with contact force
F_h	force feedback to human operator from virtual environment at the contact point
F_e	force generated from virtual environment
f_x	force produced in x direction of base frame from the VE for static case
f_y	force produced in y direction of base frame from the VE for static case
F_{mx}	force measured in x direction in end effecter coordinate system
F_{my}	force measured in y direction in end effector coordinate system
F_x	force measured in x direction in based frame
F_y	force measured in y direction in based frame
f_{xcal}	force measured in x direction in based frame by dSPACE after calibration
f_{ycal}	force measured in y direction in based frame by dSPACE after calibration
I	Inertia matrix
J	Jacobian matrix
J^T	Transpose of Jacobian matrix
K	elastic constant
L_1	the length of link1 of the robot arm
L_2	the length of link2 of the robot arm
m	Cartesian equivalent mass of the robot

m_1	mass of link1
m_2	mass of link2
M	Virtual mass in conventional robot impedance or the mass of virtual object of haptic interface
N	representative of the gravity item and Coriolis/centripetal item
T_3^0	the transformation matrix of forward kinematics
X	displacement along any degree of freedom
X_d	target position
X_0	position free of contact in haptic interface
X_m	motion input in x direction
Y_m	motion input in y direction
H(z)	Z transformation
y(z)	output signal in Z domain
x(z)	input signal in Z domain
a_i	coefficients of Butterworth low pass digital filter
b_i	coefficients of Butterworth low pass digital filter
τ	torque
θ	angular position

1 Introduction

As a result of fast development of technology in today's world, more and more previously imagined devices become possible to build. Haptic interface is one of the technologies that provide the link between the human operator and a virtual environment, which is, for some reason, used to emulate a real environment. Haptic interface has already been used in entertainment industry, for example for video games where the haptic interface can offer real mechanical feeling instead of only visual images. In aerospace industry, haptic interfaces have been used to train astronauts before going to work in space. In this case, the haptic interface is used to simulate the working environment in space instead of training directly in space. Haptic interfaces have also been adopted in medical surgery training for students and teleoperation. The final goal of development of haptic interfaces is to achieve a realistic dynamic interaction between human operator and the virtual environment.

1.1 Motivation for the research and research goal

As discussed above, haptic interfaces refer to a electro-mechanical system that permits human operators to interact with a virtual environment. This system includes mechanical parts, such as a robot arm; electrical parts, such as actuators, sensors and signal condition circuits; computer part, including software for the virtual environment simulation; control part, such as the control algorithm used for designing haptic controller. In fact, a haptic interface is a typical mechatronics system. For the same

reason, the concept of hardware-in-the loop applies here: "Hardware-in-the loop simulation uses a combination of components, some consisting of simulation models and others consisting of actual hardware."[29] HIL method uses a simulation model of the process and the real target hardware. The motivation behind this thesis is to use the HIL as a generic platform to carry out the experimental investigation of design options for haptic interfaces.

In practical applications, haptic interfaces play major role providing visual, tactile and force sensing. Impedance/admittance control has been proved particularly suitable to control force-position relationship for robot arms and is widely adopted also in designing of haptic interface. A HIL experimental setup is useful for the analysis of implementation issues of the control methodology adopted for haptic interfacing. The actual goal of this thesis is the development of an effective controller and the evaluation of its performance using a HIL experimental setup.

2 LITERARURE REVIEW

2.1 Haptic Interface

The word "haptic" refers to the sense of touch. "Haptic interface" refers to a kind of mechanical system that allows human operators to interact with virtual environments (simulated with application software) through those interface sensations. The technology of virtual environment and virtual reality was already applied to training pilots, surgeons etc and for entertainment using video game. For example, the surgery simulation in medical area is used to train students to practice the procedures using haptic interface (including a graphic display), which can accurately reflect the real surgical condition using the virtual reality approach. Fig.2.1 describes the haptic interface system.

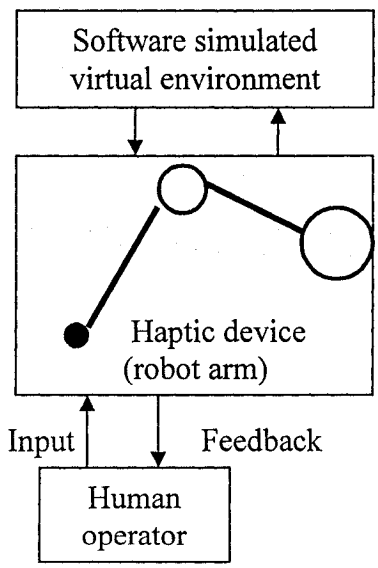


Fig. 2.1 A haptic interface system

In order to mimic the virtual environment well, haptic interface usually has several degrees of freedom. In the literature, many types of haptic devices were presented. Burdea categorized these mechanical devices as nonportable force feedback and portable force feedback [2]. The nonportable force feedback devices include those desk-grounding masters and floor and ceiling-grounded masters, such as different types of joysticks, pen based masters, sensing gloves, generalized masters and robot arms. For the portable force feedback devices, there are many types of arm exoskeletons and hand masters, for example the well-known EXOS SAFIRE masters and VIRTEX CYBERFORCE glove. But, no matter what kind of mechanical configuration it has, the haptic device is a sophisticated input/output device. Robots have been investigated for many years and it can offer many degree of freedom force/torque-motion relationships. Many researchers put their efforts in building haptic interfaces using robot arms. With the increasing of the applications in which high fidelity interaction force is needed, like the surgery training mentioned above, many researchers have focused their work on the mechanical design of the haptic system. The consensus among the researchers is that "the ideal haptic interface device is one which has low friction, inertia and backlash, is highly backdriveable, has a large force range and high mechanical bandwidth, and has a suitable working volume [3]." We can imagine that those haptic devices with fine mechanical design are specific rather than general purpose. The cost for such kind of devices is high generally. Alternatives to using robot arm were also investigated. Clover and Troy have used commercially available and general-purpose robotic manipulator as the haptic device for the application where accurate interaction between human and virtual environment is needed. [3,4]. Their approach is based on building a

controller, which compensates perfectly for mechanical limitations. In their experiments, they implemented a control methodology on the general-purpose six degree-of-freedom robot, the PUMA 560. Basically, haptic interfacing emphasizes either on advanced mechanical design or on advanced controller design. In our research, the focus is on the latter approach.

2.2 Impedance control and admittance control

There are two fundamental and dominant control methods widely used in haptic interface design. Haptic interface has two basic functions [Salisbury and Srinivasan, 1992]. In brief, the first one is to measure the input exerted by human operators and the second is to output the feedback to human operators according to the physical characteristics of virtual reality that it is simulating and let the human operators feel like they are immerged into the virtual environment. So, the two control methods are categorized by different input and output. When motion input by user is measured and force is fed back to user, impedance control is applied. Alternative method, in which forces exerted by user are measured and positions are fed back to user, is call admittance control. G.C. Burdea introduced control of haptic interfaces using a whole chapter in [2]. Both giving motion input receiving force feedback and giving force input receiving motion feedback were discussed. Although the names of impedance control and admittance control were not mentioned, he recommended the former for haptic controller design due to relatively easy implementation, which is the impedance control. Both impedance control and admittance control were proposed for haptic interfacing by

Carignan, Akin and Cleary [5,6], Clover, Luecke, Tory and McNeely [3,4]. Generally, impedance control was more often used for haptic interface than admittance control. This can be explained by the fact that force/torque sensors are more expensive than position sensors. However, in recent research work, force/torque sensors were used in impedance control for force sensing and feedback. So, this type of controller is also called impedance controller with force feedback.

Impedance control was first extensively analyzed by Hogan for the contact force control of a robot interacting with environment. In his classical three-part paper [7,8,9], two basic forms of impedances were pointed out. The lower order impedance is the relationship between output force and the input position, i.e. equivalent to the stiffness. In general the impedance is the relationship between the Laplace transforms of force and velocity. Hence, the target dynamics, which include a mass, can be described by a second order mass-damper-spring model and in Laplace domain has the form of $F(s)=Z(s)X(s) = (Ms^2+Cs+K)X(s)$. Kazerooni focused on applying impedance control with position control to robots [12] to [16]. Payandeh and Goldenberg presented a robust force controller for a manipulator performing a contact task with a rigid environment in [19]. Inoue, Sakaki, Matsumoto and Iwakane [20] introduced two approaches to impedance control, the force detection method and the acceleration method, for a manipulator with direct-drive actuators. Recently, Tafazoli, Salcudean, Hashtrudi-Zaad, and Lawrence [21] presented a novel approach for position control based on impedance control applied to the excavator cup. Albu-Schaffer and Hirzinger proposed a new controller structure based on impedance control in [22]. The proposed controller was tested on a DLR's lightweight robot. In all above conventional

application of impedance control to robots, the impedance was used to model the relationship between the motion of robot end effector (position, velocity and acceleration) and the force it exerts to the environment. For the case of a haptic interface, the application is different. In haptic interface, the physical properties of the environment are transferred (reflected) to human operator through the robot arm in the form of impedance.

2.2.1 Open-loop impedance control in haptic interfacing

Initially, open-loop impedance was used for the controller design of haptic interface. The open-loop here refers to the case that no interaction force is sensed and fed back to the controller. Open loop control is suitable to static case or the haptic device which has low mass and friction relative to the virtual environment. In this case, the actuator commands are based completely on the desired interaction force. For the case of robot arm acting as a haptic device, the torque command can be written as [4,5]:

$$\tau = J^T F_{ext} \qquad (2.1)$$

where, τ refers to the torque command sent to robot arm, J is the Jacobian matrix, F_{ext} is the interaction force between human operator and the virtual environment. The theory of open-loop impedance is straightforward and can be described as follows: when a user applies a motion input to the haptic device, sensors, such as the optical encoder mounted on the motor shaft, measure the angular motion; joint motion is then converted through forward kinematics into the Cartesian motion of the contact point, the motion of virtual object. In the next step, the model of the virtual reality generates the desired

force, according to the motion and physical characteristics of the virtual environment i.e. the impedance, the desired force is then mapped into joint torque using Jacobian matrix. As the result, a force feedback is applied to the user by the haptic interface.

2.2.2 Impedance control with force-feed back

In the open-loop impedance control algorithm, the effects from the dynamics of the haptic device are not considered. This is acceptable in static case, or the case of perfect mechanical design of haptic device i.e. with low mass and no friction. This is not, however, the case of majority robot arms. Moreover, special mechanical designs are expensive. For this reason feedback control is needed to achieve acceptable performance. The difference between force feedback (closed-loop) impedance control and open-loop impedance control is that the interaction force is sensed and fed back to the controller. Closed-loop impedance control was widely adopted [5,6]. Fig 2.2 shows block diagram for closed-loop impedance control. In this block diagram, the modified torque command is given by [5,6]:

$$\tau_c = J^T(F_d + K_f(F_d - F)) \qquad (2.2)$$

In Fig 2.2 the notations are:

X_0 is the equilibrium point of the virtual environment

Z is virtual environment impedance

K_f is force feed back proportional gain

V_h is motion command from the human operator

F is the force feedback to human operator

X is the operational space position of robot arm end effector

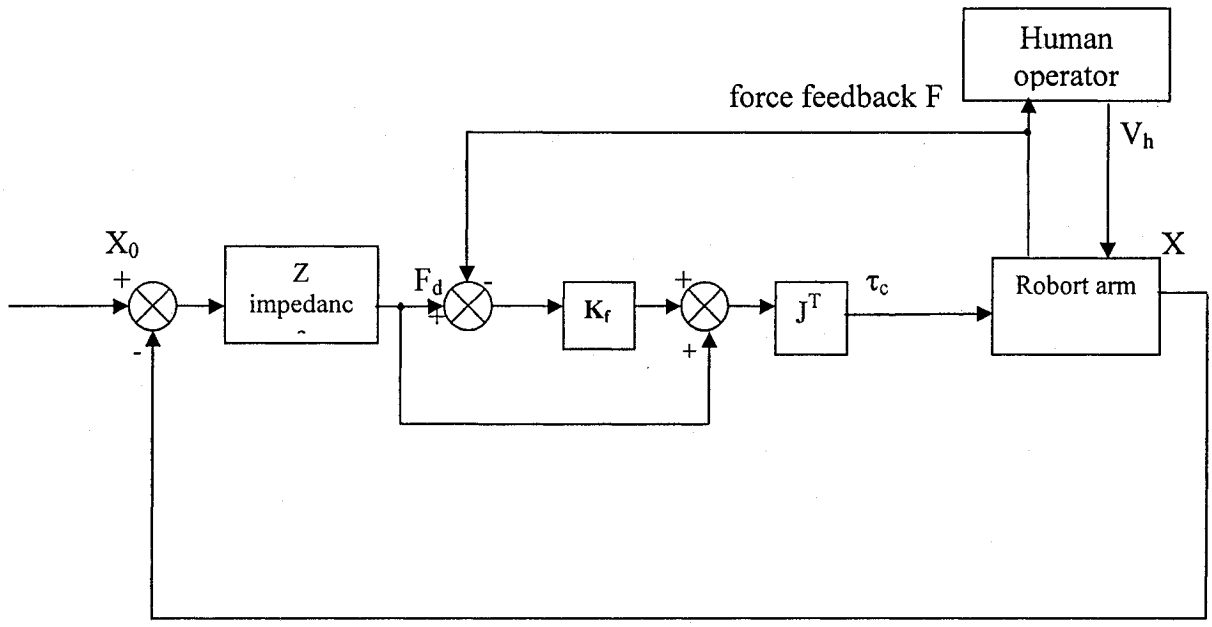


Fig 2.2 Block diagram of impedance control with force feed back

2.2.3 Admittance control for haptic interface

As mentioned above, admittance control uses the measurement of the interaction force at the contact point and derives the control signal based on the desired motion output. Clover and Troy demonstrated in simulations the application of admittance control algorithm of dynamic interaction between a user and virtual. Clover presented this algorithm mathematically in more detail in [3]. Other examples of admittance control could be found in [5,7]. In this method the interaction force is measured and the

resulting motion of virtual objects, such as position, velocity, and acceleration, are calculated. Corresponding to the time varying force input, the motion values are calculated as a resulting trajectory. Then a feedback trajectory control can be implemented. Actual motion can be measured, fed back and compared to the resulting trajectory. A PD position control could be used here.

2.3 Applications of haptic interface

Kim, Adams and Hannaford and Buttolo, Kung and Hannaford have shown that the applications of haptic interface fall in two categories [24,25,28]. Ales Bardorfer pointed out specifically that the first application of haptic interface is "virtual environment simulations, where the operator manipulates the virtual objects with software defined dynamic properties such as inertia, viscose friction and static properties such as dry friction and compliance."[23] The second application of haptic interface is "Telemanipulating systems, where the haptic interface is used as a master robot." In this case, there are two robot arms involved, one serves as the master, which is used by human operator as the input device and another serves as slave at a remote place, which will be in contact with the environment. As a result, several conventional applications of robot arms can be extended to the area of haptic interface, because, in fact, a robot arm is a type of ideal haptic device. This also justified the use in this thesis of a robot arm as a generic haptic interface development system.

3 Theoretical Framework

3.1 The concept of conventional impedance control

In the conventional robot motion control, impedance control is based on creating artificial impedance between the end-effector of the robot and the target point [42]. This mechanical impedance can be assumed as a virtual spring and a damper between the end-effector and the target point, as shown in fig 3.1. The spring generates an attraction force intended to drive the end-effector toward the target position, while the damper works like a derivative controller trying to stabilize the process. Properly selected damper will help the end-effector stop at the target point. The impedance control works as a PD controller for this case.

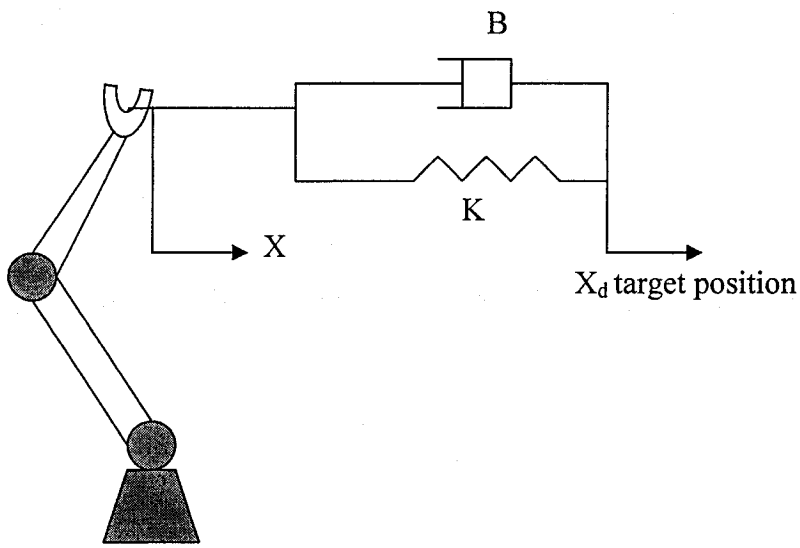


Fig 3.1 Artificial impedance between end effector and target point

Impedance control can be used for both free motion and contact motion of robot arm.

3.1.1 Impedance control for free motion of a robot arm

Free motion of a robot arm refers to the case in which no physical contact with the environment occurs. To the system described by figure 3.1, applying Newton second law we obtain following equations:

$$K(X_d - X) - B(\dot{X} - \dot{X}_d) = M\ddot{X} \qquad (3.1)$$

then, $$M\ddot{X} + B(\dot{X} - \dot{X}_d) + K(X - X_d) = 0 \qquad (3.2)$$

where, K the virtual spring coefficient

B the virtual damper coefficient

M the virtual mass

X_d the target position

Equation 3.2 gives the acceleration:

$$\ddot{X} = M^{-1}(B(\dot{X} - \dot{X}_d) + K(X - X_d)) \qquad (3.3)$$

So, if m is the Cartesian equivalent mass of the robot in Cartesian space, the force f needed to be generated in the same space by actuator at the point of end-effector is:

$$f = mM^{-1}(B(\dot{X}_d - \dot{X}) + K(X_d - X)) \qquad (3.4)$$

Then, the torque τ generated by actuator in the joint space in order to generate above force is mapped using equation 2.1,

$$\tau = J^T f$$

K and B are easy to understand here, but not M. It refers to the virtual mass, one of the parameters of the target impedance. As pointed out by Hogan, this is the apparent inertia at the end effector when we impose above target dynamics to the end effector of

the robot arm, in other words, it is the resulting lumped mass of the robot arm often applying feedback linearization to cancel all other terms of the arm dynamics [11].

3.1.2 Impedance control for robot arm in contact with environment

When the robot arm is in contact with an object and is moving it, a contact force results. If the measurement of the contact force F_{ext} is available, the dynamic process can be described according to Newton second law by following equation:

$$K(X_d - X) - B(\dot{X} - \dot{X}_d) - F_{ext} = M\dot{X} \qquad (3.5)$$

then, $$-F_{ext} = M\dot{X} + B(\dot{X} - \dot{X}_d) + K(X - X_d) \qquad (3.6)$$

Solving $\ddot{X}$ from (3.6), gives

$$\ddot{X} = M^{-1}B(\dot{X}_d - \dot{X}) + M^{-1}K(X_d - X) - M^{-1}F_{ext} \qquad (3.7)$$

If m is the Cartesian equivalent mass of the robot, the force needed to produce above dynamic behavior at the point of end-effector is:

$$f = m\ddot{X} + F_{ext} \qquad (3.8)$$

$$= m(M^{-1}B(\dot{X}_d - \dot{X}) + M^{-1}K(X_d - X) - M^{-1}F_{ext}) + F_{ext}$$

The torque commands to generate above force is again obtained using equation 2.1.

Control law defined by (3.4) is similar with PD control law with the difference that the K_p and K_d are related to the parameters of operational space target impedance. Control law (3.8) calculates the force command for given measurements of position, velocity and contact force. Which means that in control law (3.8), besides the feedback

terms position X and velocity $\dot{X}$, there is one more feedback term F_{ext} from the measurement of the contact force.

3.2 Impedance control in a haptic interface

A model-based impedance control for a haptic interface is used in this thesis. Similar equations, with the one from previous section, will be derived, but with conceptual difference outlined in the following section.

3.2.1 Control of haptic interface

An example will clarify first what happens in haptic interface: a human operator intends to manipulate an object or work in a particular environment, and for some reason this object is not actually available. As a replacement he/she uses a robot arm to simulate the working environment/object. **The physical properties in the environment are transferred (reflected) to human operator by the robot arm in the form of a mechanical impedance.** An ideal haptic interface design should transfer to the human operator the exact physical properties and not combine them with the effect of the robot arm dynamics. A well-designed haptic interface will let human operator feel like he/she interacts with the actual environment that is in fact simulated.

Consequently, in this case, the torque generated from the actuators for the haptic device is not intended to drive the robot and move an object. The torque generation has two tasks here, first to compensate the effects of the robot arm dynamics, and second is to transfer the physical properties of the virtual environment to human operator. This

represents the major difference between traditional robot force control and the haptic interface design. Based on above viewpoint, we can image two extreme cases:

1) **The environment is very stiff.**

   In this case, the value of K should be very large. As long as the actuators have enough torque capacity, the human operator will feel like he/she is pushing hard against a very stiff environment.

2) **The environment is very soft.**

   This situation refers to relatively small values of K and B, close to zero. In this case, the torque from the actuator will let the robot move "free in the air". Human operator can easily move it to anywhere without feeling any contact force reaction.

3.2.2 Impedance/Admittance control model of haptic interface

Based on above analysis, it can be seen that the haptic interface receives the input from a human operator, and then, generates the output according to the physical properties of the virtual environment that is simulated. Impedance control model is shown in fig 3.2, where, F_h denotes feedback force from end effector of the haptic interface to human operator while V_h refers to the motion input from human operator at the point of contact. The impedance control for haptic interface transfer (reflect) the physical properties of virtual environment to human operator in a different way from the admittance control.

In impedance control application for haptic interface, the virtual object, in this case the end effector of robot arm is in the equilibrium point X_0 initially, in the absence

of contact (i.e when the human operator is not interacting with the virtual environment). When human operator starts to be in contact with the virtual environment, an interaction force is generated and a deviation from the X_0 is also generated as the result, which has the form of impedance and is completely depends on the physical properties of the virtual environment. In this case, the motion deviation from X_0 is sensed, and force is fed back to human operator. This process is the first task of haptic controller, but, besides it the haptic device dynamic effect has to be accounted for. In real world, there are always some limitations for any mechanical system. The effects of robot inertial, joint friction, Coriolis/centripetal forces of robot can be ignored for static case, but in the dynamic case these effects may be significant and can conceal the physical properties of the virtual environment. An ideal haptic interface design should transfer to the human operator the exact physical properties of the environment and not combine with the effect of the robot dynamics. Consequently, the second task of the haptic controller is to compensate for the mechanical limitations of the haptic device. A complete impedance control model of user, haptic interface and virtual environment is described in fig 3.2.

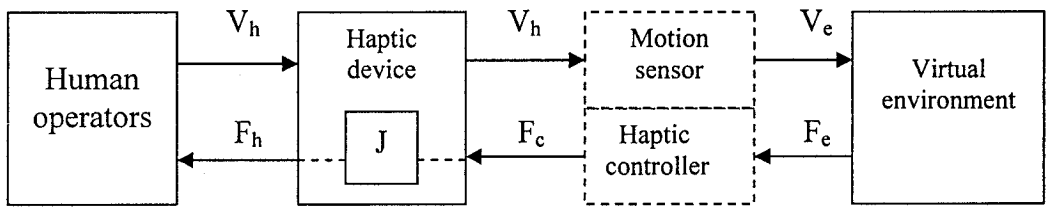


Fig 3.2 Impedance control model of user, haptic interface and VE

Remark: V_h refers to the motion input exerted by user, V_e refers to the motion of the object in virtual environment, F_e refers to the force feedback from virtual environment to user, F_c refers to the

torque/force command dent to haptic device, and F_h refers to the actual force received by user at the end effector of robot arm. Ideally, F_h should be exactly equal to F_e, and V_h should be exactly equal to V_e.

Admittance control is in fact the reversed form of the impedance control As C.L Clover, G.R.Luecke, J.J Tory and W.A.McNeely pointed out "Usually, when a robot is interacting with its environment it must act as an impedance while the environment acts as an admittance. However, in a haptic system, there are two systems interacting: a human manipulator and a mechanical manipulator. In the admittance control paradigm the impedance role is assigned to the human user and the admittance role is assigned to the mechanical haptic device [4]." In admittance model, when the human operator starts to be in contact with the virtual environment, the contact force is sensed, and for the given physical properties of virtual environment this force will result in motion deviation from the case of the absence of the applied contact force. In this case, contact force is the input and deviation of motion is the output. Their relationship depends on the physical properties of virtual environment. Haptic controller still has the two major tasks: to transfer the physical properties to the human operator and simultaneously, to compensate for the dynamic effects of haptic device. The admittance control for the user, haptic interface and virtual environment is shown in fig 3.3.

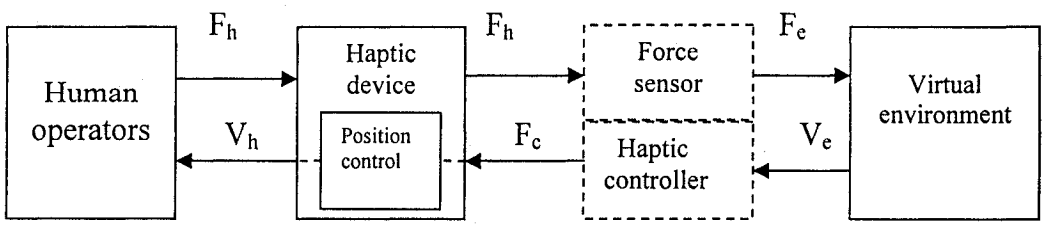


Fig 3.3 Admittance control model of user, haptic interface and VE

Remark: F_h refers to the force input exerted by user, F_e refers to the force exerted on the object in virtual environment, V_e refers to the motion of the object in virtual environment to user, F_c refers to the torque/force command sent to haptic device, and V_h refers to the actual motion of the end effector of robot arm. Ideally, F_h should be exactly equal to F_e, and V_h should be exactly equal to V_e.

3.2.3 The components in haptic interface

A haptic interface is a typical mechatronics system, which includes following subsystems:

1) Software simulated virtual environment

   This refers to the application of the haptic interface for the simulation of virtual reality. It contains the mechanical model of physical properties of the virtual environment. Usually, the software is created and compiled in host PC.

2) Haptic controller

   This is the control algorithm performing two tasks: to transfer the physical properties to the human operator and simultaneously, to compensate the effect of the haptic device. This is also achieved by the software.

3) Haptic device

   Haptic device refers to a piece of a particular mechanism with which the human operator interacts. It could be a robot arm or any device mentioned in chapter 2. Particularly, for the application in

telemanipulating systems, haptic device refers to the master robot arm, used as an input device.

4) Slave robot arm

This is the specific case of telemanipulating systems, where the robot arm serves as a slave in contact with environment.

5) Other hardware

Position sensors, force/torque sensors, communication circuits and power control circuit (drivers).

6) Human operator

Human operator exerts an input to the haptic device and receives a feedback from the same haptic device. The relationship between the feedback and the input should be completely dependent on the physical properties of the virtual environment and consequently the effect of the haptic device dynamics should be compensated by the control algorithm.

In Fig 3.4(see next page), the integration of above basic parts of a haptic interface is shown. When these components work together, they form a mechatronics system.

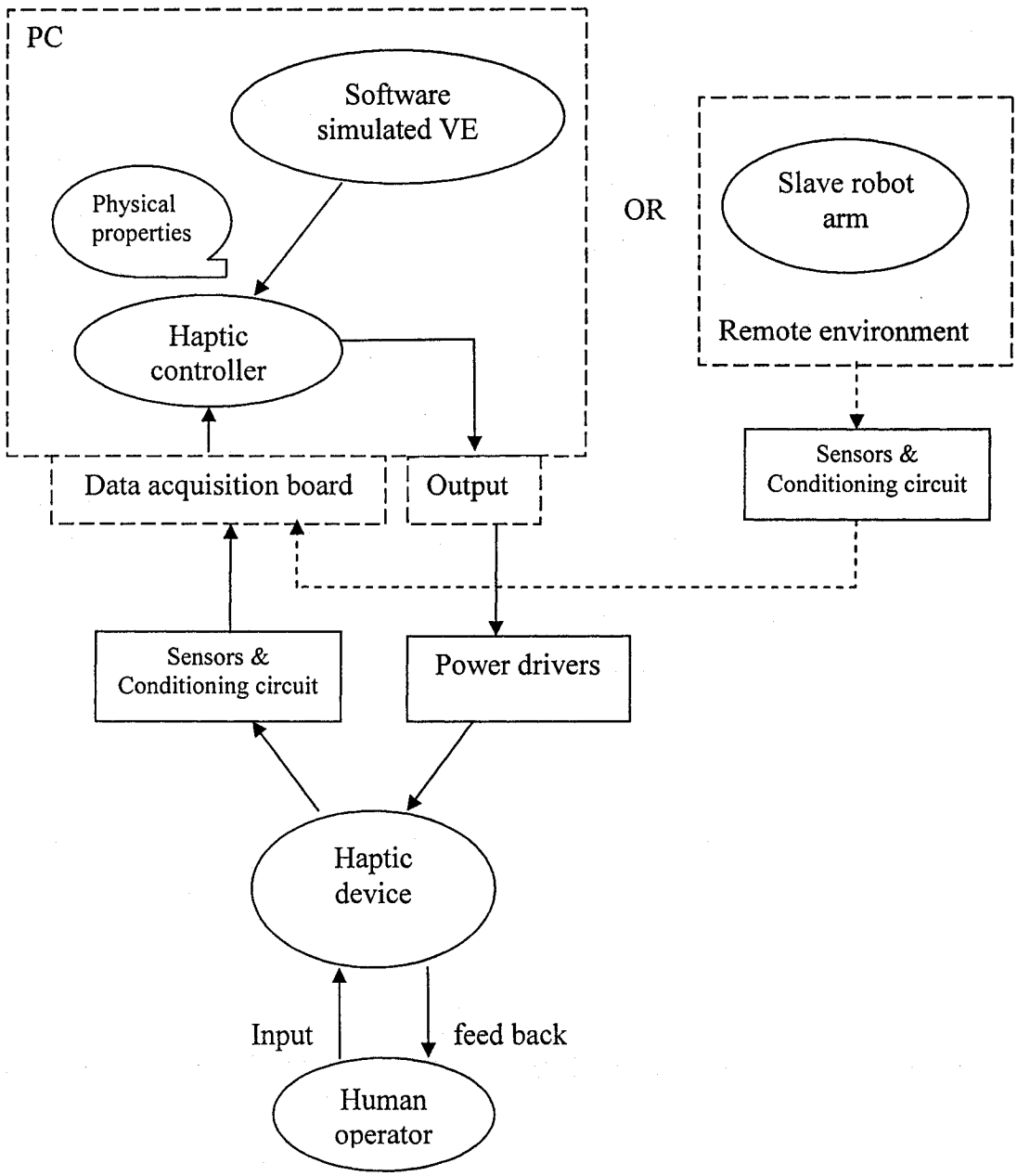


Fig 3.4 Integration of components in haptic interface

3.3 A model based impedance controller design for haptic interface

Dr. Dan Necsulescu developed a three-part impedance control scheme for conventional robot motion control in [30,31]. A similar model based impedance control is used in this thesis for the application of the haptic interface in virtual environment simulation with the difference outlined in the previous section. The theoretical foundation of this modeling is based on the two major functions of haptic controller. As mentioned in previous section, this first one is transferring physical properties of the virtual environment to human operators and the second one is the compensation of the effects of haptic device dynamics. Consequently, if robot manipulator is the haptic device it is required to apply computed torques to the joints of robot manipulator. One part of the torques has the purpose to compensate for the effects of robot arm dynamics such that there is no feel when operator pushes or pulls it. The status of the robot arm is just like "free in the air". The other part of the torques has the purpose to generate force feedback according to the input motion from the human operator. This relationship depends on the physical properties of the virtual environment. The torques applied by the actuators are the summation of the two sets of torques. This approach is required for dynamic case, but it is applicable also to static case. And this has been verified in experiments. For static case, open loop impedance control has been widely used in literature. For this reason, open loop impedance control is the first tested approach.

3.3.1 Open loop impedance control model

Open loop means here that no interaction force is sensed and fedback to the controller. Open loop control is suitable in static case or when the haptic device has low mass, and friction relative to the virtual environment. In this case, the actuator commands are based completely on the desired interaction force, and can be written as:

$$\tau = J^T F_h \qquad (3.9)$$

where, J refers to the Jacobian matrix and depends on the configuration of the robot. and F_h refers to the feedback force from virtual environment and is a function of motion input and the physical properties of virtual environment. The control diagram is:

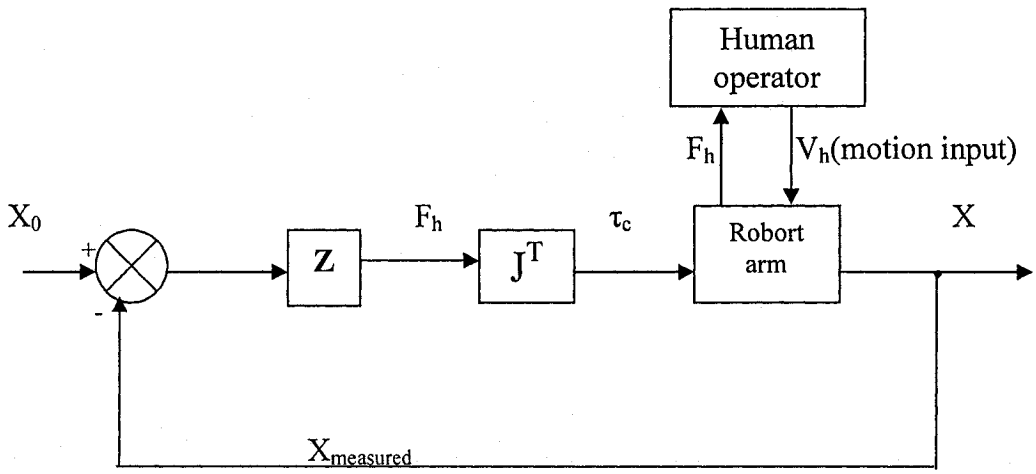


Fig 3.5 Open loop impedance control

The **Z** represents the physical properties of virtual environment called environment impedance, which gives the name of impedance control for haptic interface.

3.3.2 Model based impedance control for haptic interface

1) The virtual environment

In this thesis, the research was focused on virtual environment that can be described by the following differential equation in matrix form:

$$-F_h - B\dot{X} - K(X - X_0) = M\ddot{X} \qquad (3.10)$$

where, F_h is feedback force from the virtual environment to human operator; B is the viscosity constant; K is the equivalent elasticity constant of the virtual environment; M is the mass of the object in the virtual environment. X_0 is the initial position in absence of interaction force. The virtual environment specified here is not limited to a mass-spring-damper system, and can be represented by a complex model, reducible often to a passive M-B-K equivalent. For example, a needle penetrating tissue in medical surgery simulation can be modeled by such a M-B-K system.

2) Haptic Device

If a robot arm is chosen as a haptic device, following equation applies:

$$\tau = I(\theta)\ddot{\theta} + N(\theta) - J^T(\theta)F_h \qquad (3.11)$$

where, τ refers to the torque command sent to actuators, $I(\theta)$ refers to the inertia matrix of the robot arm, N includes the items of gravity, Coriolis/centripetal. F_h is feedback force from the virtual environment to human operator. The torque equation contains two parts. The first part includes inertia matrix $I(\theta)$ and N matrix. The compensation of this part reduces the robot dynamic such that environment dynamics becomes transparent to the operator as if there is no robot. The second part contains the F_h. This part produces a

torque with reverse direction to the torque of the first one and generates the force feedback F_h to the operator and it reflects the physical properties of virtual environment.

3) Motion Transformation between Cartesian space and joint space

$$X = KIN(\theta) \tag{3.12}$$

$$\dot{X} = J(\theta)\dot{\theta} \tag{3.13}$$

$$\ddot{X} = J(\theta)\ddot{\theta} + \dot{J}(\theta)\dot{\theta} \tag{3.14}$$

Where, KIN (θ) means forward kinematics.

Now, bringing equation (3.14) into (3.10), we get:

$$F_h = -M[J(\theta)\ddot{\theta} + \dot{J}(\theta)\dot{\theta}] - B\dot{X} - K(X - X_0) \tag{3.15}$$

solving (3.15) for $\ddot{\theta}$, we get

$$MJ(\theta)\ddot{\theta} = -F_h - MJ(\theta)\dot{\theta} - B\dot{X} - K(X - X_0) \tag{3.16}$$

then,

$$\ddot{\theta} = -J^{-1}M^{-1}F_h - J^{-1}\dot{J}\dot{\theta} - J^{-1}M^{-1}B\dot{X} - J^{-1}M^{-1}K(X - X_0) \tag{3.17}$$

bringing (3.17), and (3.13) into (3.11), we obtain:

$$\tau = M^{-1}(-IJ^{-1}F_h - IJ^{-1}BJ\dot{\theta} + IJ^{-1}K(X_0 - KIN(\theta)) - IJ^{-1}\dot{J}\dot{\theta} + N(\theta) - J^T(\theta)F_h \tag{3.18}$$

Equation (3.18) means force is measured and fedback in to the control loop. Due to the force feedback this model based impedance control is called closed loop impedance control. (see next page for block diagram Fig 3.6)

(See next page)

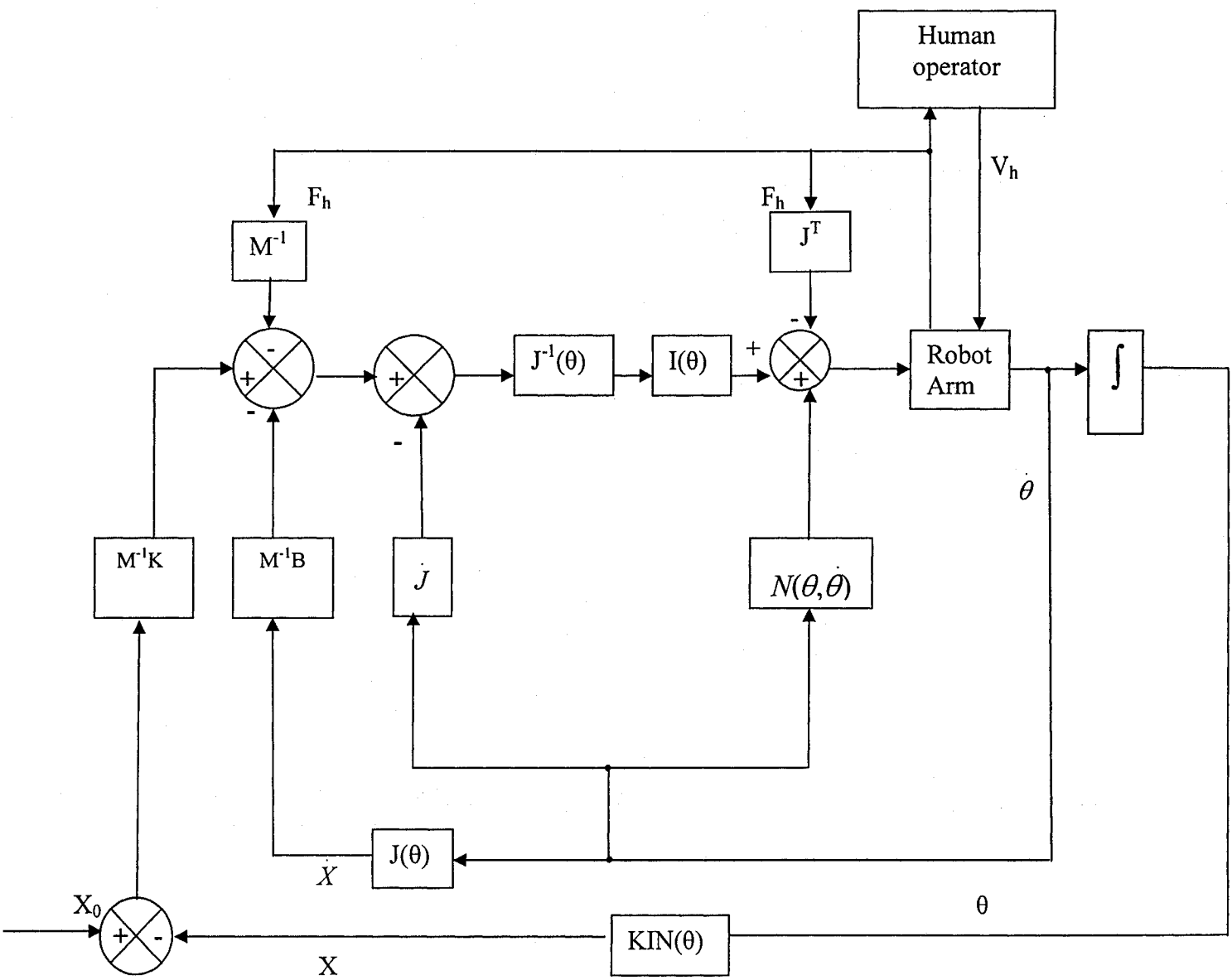


Fig 3.6 Closed loop impedance control for haptic interface

From the block diagram Fig3.6 and equation (3.18), we can see that the model based impedance is in fact the computed torque control approach, which is a special case of feed back linearization control. In this control law there are three parts:

1) Linear Cartesian control law from equation (3.10):

$$\ddot{X} = -M^{-1}F_h - M^{-1}B\dot{X} + M^{-1}K(X_0 - X) \qquad (3.19)$$

2) Cartesian decoupling scheme from equation (3.14):

$$\ddot{\theta} = J^{-1}(\theta)\ddot{X} - J^{-1}\dot{J}(\theta)\dot{\theta} \qquad (3.20)$$

3) Joint decoupling scheme, which is equation (3.11)

$$\tau = I(\theta)\ddot{\theta} + N(\theta) - J^T(\theta)F_h \qquad (3.11)$$

This feedback linearization control diagram contains an outer loop that actually is a linear feedback control law in operational space with K_p equal to K/M and K_d equal to B/M, and an inner loop for linearization of the nonlinear, coupled robot dynamics. In this thesis, this control law relates the control gains to the parameters of virtual environment, M, B and K. Generally, M, B and K are 6x6 diagonal matrices for a 6 DOF robot arm representing the physical properties of virtual environment, and is called environment impedance.

Critically damped situation is expected to give good performance of the haptic controller. The experiments presented in chapter5 are for the evaluation of the model based haptic controller of M, B and K. Experimental setup is presented in chapter 4 and the results in chapter 5.

4 DESCRIPTION OF EXPERIMENTAL SETUP

Haptic interface development refers to both haptic device and interfacing software development. Model based impedance control proposed in chapter 3 is based on a generic multi degree of freedom robot arm capable of both motion and force/torque control and feedback. Experimental testing of this control methodology was carried out on a two degree of freedom planar robot, which satisfies haptic interface device requirements: low friction and backlash, backdriveable, large force range and high mechanical bandwidth as well as suitable working volume [3,4]. The only exception is inertia, which is relatively high. This system can be utilized to evaluate the performance of the controller, in case the effects of inertia are supposed compensated by the computed torque. The experimental setup contains all the components described in section 3.2.3, for virtual environment simulation, and is presented in the section that follow.

4.1 Manipulator hardware

4.1.1 Hardware configuration

One arm of a dual-arm robot facility in the lab was used as the haptic device. This simple two-degree freedom planar robot arm is satisfactory to prove the concept of the controller design proposed in chapter 3. The hardware configuration of this robot arm is shown in fig 4.1.

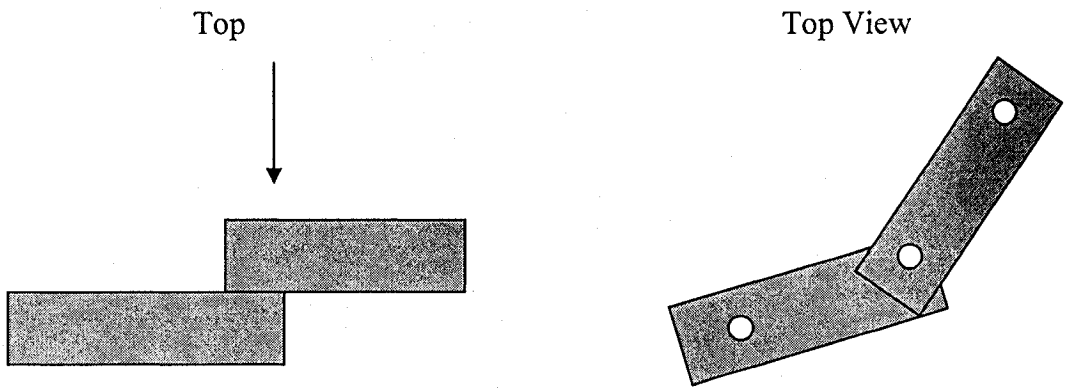


Fig 4.1 Two degree of freedom robot arm

The mechanical design is very simple and makes the calculation of kinematics and dynamics easier. Each link, made of aluminum, has been designed for high rigidity. This robot arm can simulate the motion in a plane of a virtual object in its working volume because it is equipped with NSK direct drive motors at its two link joints, it can provide low friction, low backlash and backdriveability, which is not available in normal geared robot manipulators. The detail link parameters can be found in appendix A.

4.1.2 Kinematic and dynamic model

The kinematic and dynamic model for this kind of mechanical configuration can be found in many robot books. This work used Denavit-Harteberg conventions to develop all kinematic models and Lagrange's equation to derive dynamic models. The

detail procedures for deriving these models is shown in appendix B. The main results are:

1) The Transformation matrix T_2^0:

$$T_2^0 = \begin{vmatrix} \cos(\theta_1+\theta_2) & -\sin(\theta_1+\theta_2) & 0 & L_2\cos(\theta_1+\theta_2)+L_1\cos\theta_1 \\ \sin(\theta_1+\theta_2) & \cos(\theta_1+\theta_2) & 0 & L_2\sin(\theta_1+\theta_2)+L_1\sin\theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} \quad (4.1)$$

Forward kinemics is :

$$X=L_1*\cos\theta_1 + L_2*\cos(\theta_1+\theta_2) \quad (4.2)$$

$$Y=L_1*\sin\theta_1 + L_2*\sin(\theta_1+\theta_2) \quad (4.3)$$

2) The Jacobian matrix J:

$$J = \begin{vmatrix} -L_1\sin\theta_1 - L_2\sin(\theta_1+\theta_2) & -L_2\sin(\theta_1+\theta_2) \\ L_1\cos\theta_1 + L_2\cos(\theta_1+\theta_2) & L_2\cos(\theta_1+\theta_2) \end{vmatrix} \quad (4.4)$$

3) The dynamic model

The dynamic model is described by equation (3.11)

$$\tau = I(\theta)\ddot{\theta} + N(\theta) - J^T(\theta)F_h$$

Given the configuration of the robot arm, each part in above equation is given by:

Inertial Matrix I:

$$I(\theta)=\begin{vmatrix} \frac{1}{3}m_1L_1^2+\frac{1}{3}m_2L_2^2+m_2L_1^2+m_2L_1L_2\cos\theta_2 & \frac{1}{3}m_2L_2^2+\frac{1}{2}m_2L_1L_2\cos\theta_2 \\ \frac{1}{3}m_2L_2^2+\frac{1}{2}m_2L_1L_2\cos\theta & \frac{1}{3}m_2L_2^2 \end{vmatrix} \tag{4.5}$$

Centrifugal and Coriolis Matrix:

$$N=\begin{vmatrix} -\frac{1}{2}m_2L_1L_2\sin\theta_2\dot{\theta}_2^{\,2}-m_2L_1L_2\sin\theta_2\dot{\theta}_1\dot{\theta}_2 \\ \frac{1}{2}m_2L_1L_2\sin\theta_2\dot{\theta}_1^{\,2} \end{vmatrix} \tag{4.6}$$

The gravity and friction terms are ignored for the particular planar mechanical design of this robot arm.

Applied torque and force:

$$\tau=\begin{vmatrix}\tau_1 & \tau_2\end{vmatrix}^T \tag{4.7}$$

F_h is the interaction force at the end effector of the robot arm and is measured by force sensor.

$$F_h=\begin{vmatrix}F_x & F_y\end{vmatrix}^T \tag{4.8}$$

Equation (4.9) gives:

$$J^T = \begin{vmatrix} -L_1 \sin\theta_1 - L_2 \sin(\theta_1 + \theta_2) & L_1 \cos\theta_1 + L_2 \cos(\theta_1 + \theta_2) \\ -L_2 \sin(\theta_1 + \theta_2) & L_2 \cos(\theta_1 + \theta_2) \end{vmatrix} \quad (4.9)$$

4.2 Actuator, Driver and Sensors

4.2.1 NSK Megatorque Motor

This two-degree of freedom robot arm is driven by two direct drive motors, high torque brushless actuators. These two motors are capable of providing very high torque, which is high enough for the applications in this thesis. The larger one, which drives link1, is able to generate torque up to 88.2 Nm, while the smaller one, which drives link2, is able to provide torque up to 9.8 Nm. Consequently, the maximum force that can be generated at the end effector for the given link configuration can reach about several hundred newton. The motors can work in position, velocity and torque mode. The function of torque mode offers a convenient interface through which we can send the computed torque commands to the driver unit, and the processor in the driver unit will control the driving current accordingly.

4.2.2 Driver unit

The driver unit contains a driver for each motor. Each driver has a RS232 terminal and input/output interface. The input/output interface receives and sends

signals to computer, while the RS232 terminal is used to send commands from host PC to the driver unit to set these motors to torque mode. Besides these interface circuits, the driver unit contains three major parts: digital signal processor, power amplifier and resolver interface. The digital signal processor is a 16-bit microprocessor receiving commands from control PC, comparing the command variables with measured values of the controlled variables, and then making correction continuously so that the motor can follow the command. The power amplifier is the one actually driving the motor. The resolver interface takes the position signal from the built-in resolver and sends it to the digital signal processor.

4.2.3 Position sensor, the built-in resolver

The direct drive motor has a built-in high-resolution resolver. The working principle of resolver can be found easily in literature and gives resolutions compared to high-resolution optical encoders. The difference is that instead of an optical device, the optical encoder, the resolver is a mechanical device. The signal coming from the resolver interface is a digital signal and can be directly transmitted to digital input of the data acquisition board. In our experiment, it was connected to the incremental encoder board of the dSPACE computer. The larger motor has a resolver with a resolution of 614400 pulses per revolution, while the smaller motor has a resolver with a resolution of 409600 pulses per revolution [32]. This means a much more accurate measurement than generally available from optical encoders.

4.2.4 Force sensor

The availability of a force/torque sensor is essential for building a haptic interface. Without the force sensor, the performance of the haptic interface cannot be quantified. The sensor has to be accurate and small enough to be installed at the end-effector of the robot arm. Commercially available force/torque sensors are very expensive and for these reason, a 2D force sensor was developed particularly for the research work of this thesis.

The development of this force sensor is based on using strain gages. The direction of sensitivity of the strain gage is the direction along which the strain or force can be measured (fig 4.2).

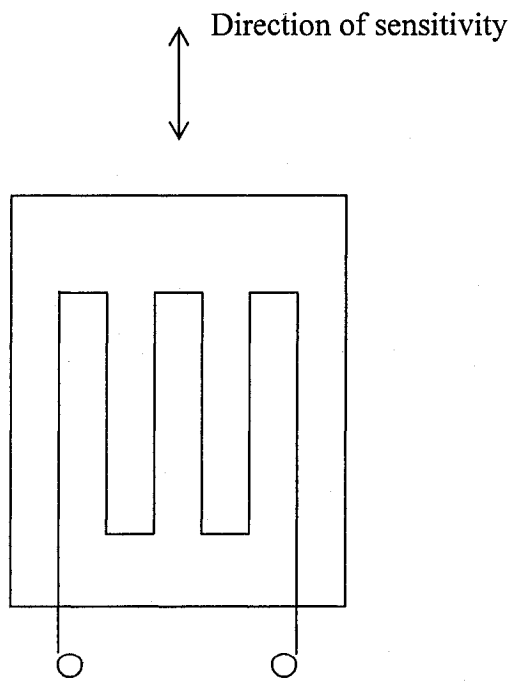


Fig 4.2 Sensitivity direction of strain gage

Based on this property of strain gage, a 2D force sensor was built by mounting two pieces of strain gage on the two sides of a piece of metal specimen respectively (Fig 4.3).

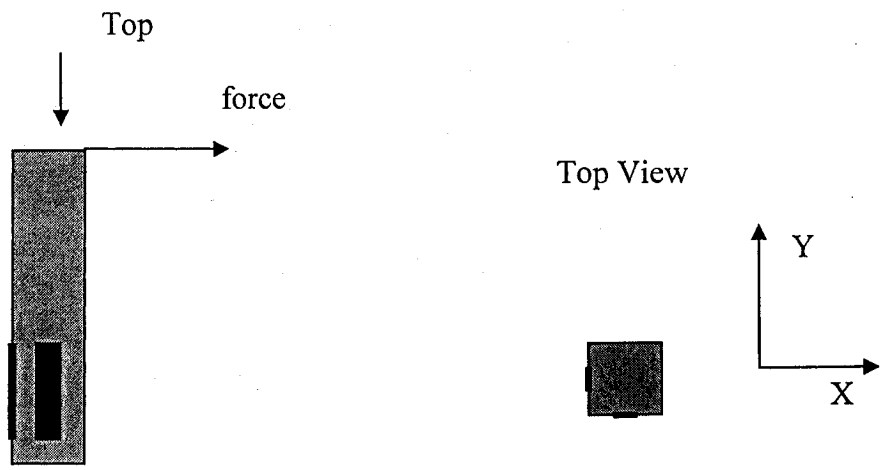


Fig 4.3 2D strain gage

For a given force, in elastic working range as illustrated in fig 4.3, the produced elastic deformation should be proportional to the applied force. So, any force vector in a particular plane can be measured by its two components in X and Y direction. Once we establish the linear relation, this sensor is ready for measurement. The calibration results of the force sensor are presented in chapter 5. The strain gage used in this experiment is EA-06-250BH-120, with nominal resistance 120 ohm. The detailed information about this strain gage can be found in [36].

4.3 Signal conditioning circuit for force sensor

The signal from resolver, through the driver unit, can be directly connected to the incremental encoder board of the dSPACE computer. The signal conditioning is already built in. For the force sensor, the change of resistance of the strain gage caused by external force need to be converted to an electrical signal, such as voltages, and has to meet the working range of the analog input channel of data acquisition board in the dSPACE computer. A circuit was built for this thesis work according to similar signal conditioning circuit for strain gage based measurement in [1]. It contains a Wheatstone bridge, which converts the resistance change of the strain gage to output voltage change and an inverting operational amplifier circuit, to make the output voltage meet the –5v /+5v working range of the analog input channel of the dSPACE computer. A 10k adjustable resistor was used to set the output voltage to zero when free of external force. R_2, R_3 and R_4 were chosen equal to the nominal resistance of strain gage. 10V DC power supply is needed to produce the working current for stain gage, and two 9v batteries are needed to provide power to the LM741 operational amplifier. The circuit is shown in Fig 4.4.

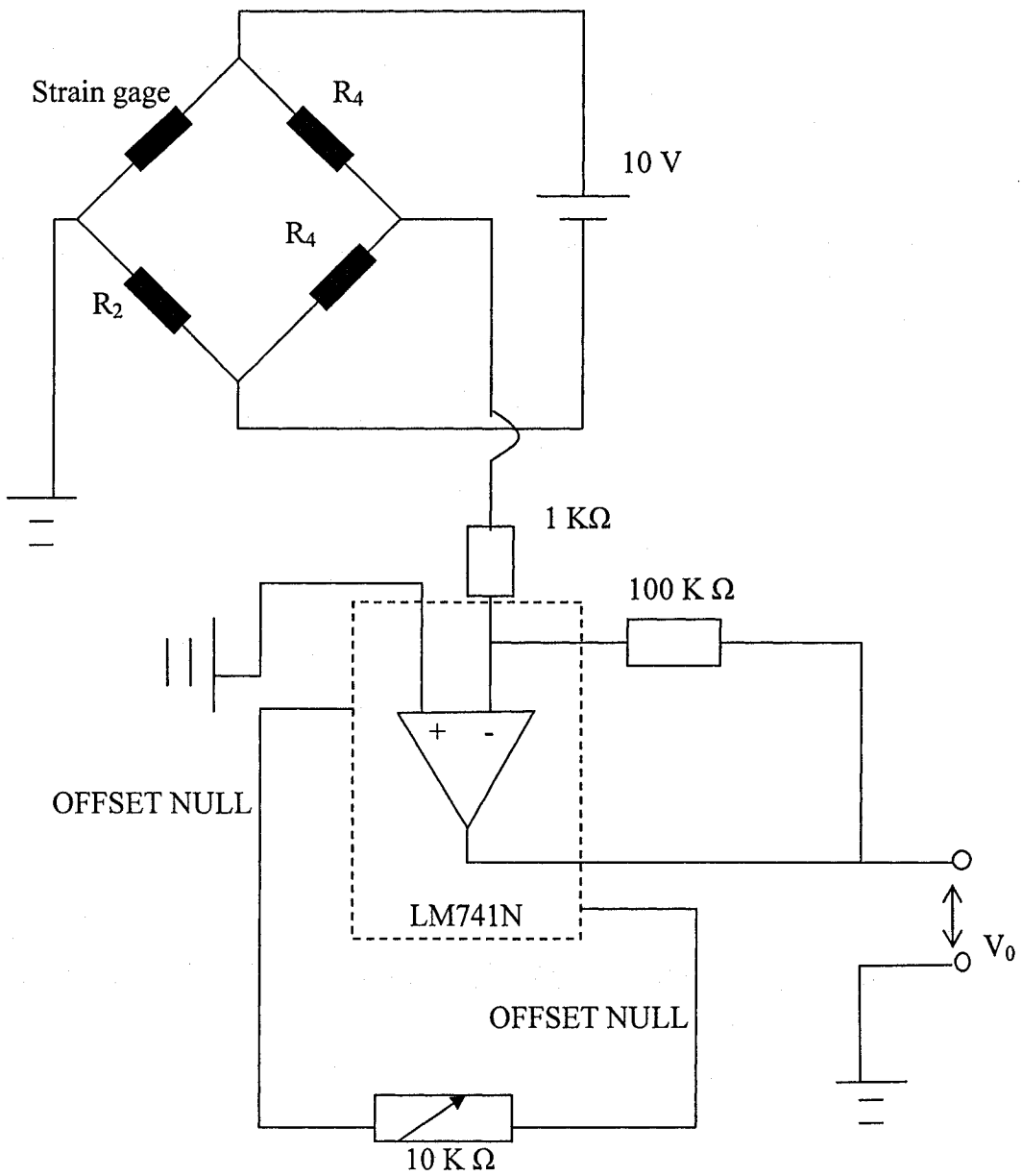


Fig 4.4 Signal conditioning circuit of force sensor

4.4 Control & Monitoring System

4.4.1 Hardware

The hardware of control and monitoring system contains a host desktop PC and a dSPACE computer. The dSPACE system includes: Processor board, A/D converter board, D/A converter board and incremental encoder board. The dSPACE computer has no direct user access and has to be connected to a host PC through a PC expansion card. The host PC is used to build software application using C language, compile it using a vendor developed compiler to machine executable language and load the compiled program to dSPACE computer through the PHS-bus. dSPACE computer performs real time control and monitoring. The manufacturer developed supporting software allows user access the traced signal from host PC. Following are the boards installed in dSPACE computer.

DS1002

This board contains a Texas Instruments TMS320C30 digital signal processor. This board is also connected to the host PC through a PHS-bus. The processor speed is 60 nano seconds for executing a single instruction. Detail information about this processor can be found in [33].

DS2002

There is one DS2002 A/D board installed in dSPACE computer. The board offers two separate 16 bit A/D converter with 16 multiplexed inputs each [33]. The conversion time for 16 bit is 5 μs. Two channels in the A/D board were used to receive analog signals from the 2D force sensor for thesis work.

DS2101

There are two DS2101 D/A boards installed in dSPACE computer. These board offer 5 fully parallel 12 bit voltages output D/A converters and 3 μs settling time to 0.01% [33]. Two outputs have been connected to the driver units of two motors respectively to send torque commands.

DS3001

Two DS3001 incremental encoder boards can be found in dSPACE computer. Each of them has 5 fully parallel 24 bit encoder interface channels. The resolver signal coming from the driver unit interface can be directly connected to the board.

Host PC

The host PC is a IBM 486/33 desktop having vendor (dSPACE company) developed supporting software installed and communicating with dSPACE computer through a PHS-bus. The vendor developed supporting software and complier provide the interface for users to trace and visualize the control signal and develop their own application software using C language

4.4.2 Supporting software

The supporting software developed by the vendor came with the hardware. Its function is to setup the connection between subsystems as well as between user and dSPACE computer. Four major software components are listed here:

SED30

This software is used to configure hardware, such as to choose which board is going to be used for a particular application, select address for boards, set working range for analog input, and so on. After all hardware is configured, a setup file will be generated.

MON30

This software is used to run the program in dSPACE. It also provides the options of uploading user developed application program, resetting boards, clearing channels as well as quitting the software but keeping the program running, which can let the user trace wanted variables.

Trace30

This software permits user to trace signals during the process, either display the signals in screen or print charts into disk. It also permits the user to save traced history data into a disk, which is very important for data analysis. Users specify which signals should be traced by creating a file with suffix TRC using editor. A .MAP file links these signals to corresponding input/output ports.

Compiler

The vendor developed a complier that can compile user developed C programs to machine executable language, particular to Texas Instruments TMS320C30 digital signal processor.

4.5 System integration – HIL experimental setup diagram

Above described components in haptic interface form a mechatronics system as well as a HIL experimental setup. Fig 4.5 shows the experimental setup diagram.

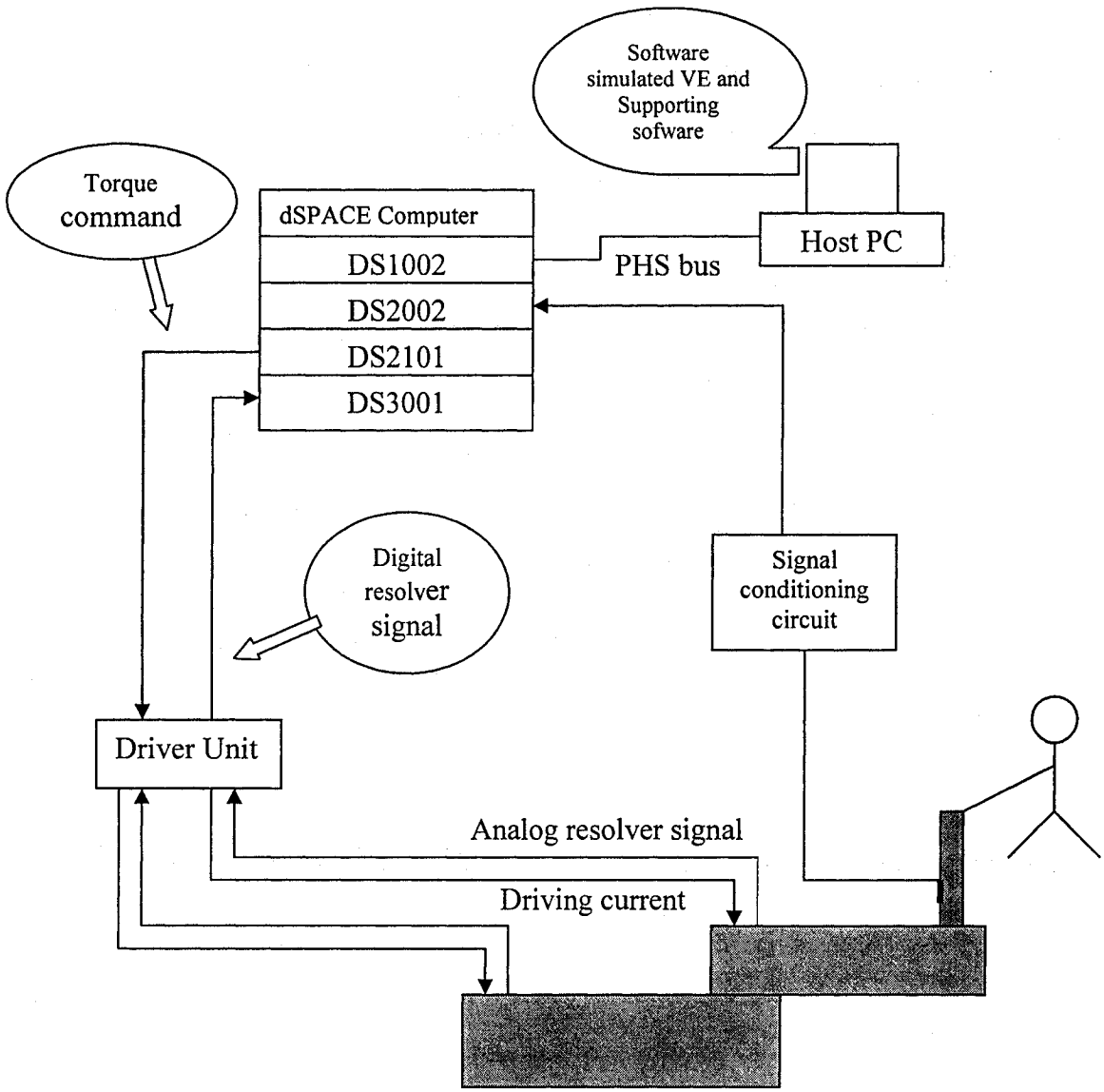


Fig 4.5 HIL experimental setup diagram

4.6 Digital filter

The necessity of a digital filter was not perceived at the beginning of the work until problem arised during experiments. Although we have accurate enough sensors installed, especially the resolvers, which can generate more that 400k pulses per revolution, no sensors are perfect. Computation induced noise has significant effects to acceleration estimation. For the analog input, this noise also has effect to the accuracy of measurement. Without the implementation of the digital filter, further experiments could not be carried out and the haptic interface could not be built. In order to eliminate the effect, a low pass digital filter was implemented. A Butterworth low pass filter has following form in Z domain [35]:

$$H(z)=\frac{y(z)}{x(z)}=\frac{\sum b_i z^{-i}}{1+\sum a_j z^{-j}} \qquad (4.10)$$

The equivalent discrete-time equation of the filter is given by:

$$y(nT)=\sum b_i x(nT-iT)-\sum a_j y(nT-jT) \qquad (4.11)$$

For a second order Butterworth filter, the following equation was used for programming:

$$y(n)=b(1)x(n)+b(2)x(n-1)+b(3)x(n-2)-a(2)y(n-1)-a(3)y(n-2) \qquad (4.12)$$

The coefficients a and b can be obtained from a Matlab function.

The coefficients, for cut-off frequency of 20Hz and 10 Hz at samples rate of 1k samples/s, are listed in the table 4.1. 20 Hz cut-off frequency was the one used in the experiments. The experiments at different sampling rate were also conducted.

Table 4.1 Coefficients of Butterworth low pass digital filter at 1k sampling rate

Coefficient	10 Hz	20 Hz
b(1)	0.0009	0.00362
b(2)	0.0018	0.0072
b(3)	0.0009	0.00362
a(2)	-1.9111	-1.8226
a(3)	0.9149	0.83718

4.7 Safety issues

As we mentioned before, the NSK Megatorque Motor can generate very large torques. These torques are large enough to pose safety dangers. Effects from noise or any computation error may cause the torque command reach its upper limit. To avoid this situation, a safety check function was added into program. Torque commands were checked before they were sent to driver unit from DS2101 board. Only torque command within spec limit can be applied.

Another issue that may cause safety issues is the output torque command signal that still retained after aborting the program, as long as dSPACE computer is not turned off. Next time when program starts again, the retained signal will be sent to driver unit first. If the torque command is big enough, it may hurt the user, and consequently D/A channels have to be cleared every time before running the program.

5 Experimental results

5.1 Experimental design and preliminary analysis

The experiments presented in this chapter were designed for evaluating the performance of the haptic interface and at the same time for proving that this type of haptic interface, based on the theoretical framework described in chapter 3, can be implemented using the hardware-in-the-loop experimental setup described in chapter 4. Before any experiment can be carried out, the force sensor has to be calibrated. The availability of a force sensor is essential to the control method that uses force feedback for the haptic interface, while for open loop control, this is not necessary. In this research work, however, for evaluating the performance of the haptic interface, a force sensor is needed also in open loop impedance control for measuring the interaction force at the interaction point, even if not used for feedback and it has no contribution to the actual performance of the haptic interface.

First, the experiments were conducted for static case, using open loop impedance control methodology. Forces were measured for both individual static positions and slow motion. The performance of the haptic interface was evaluated by comparing the measured force and the computed value. The error should be within the sensing resolution of human being. The virtual environment used to test the haptic interface was a mass block attached to two identical springs, as shown in section 5.2.1. The second part of the experiments was implemented based on the model based close loop impedance control methodology. The following aspects were considered to have

effects on the performance of haptic interface: first, according to the model based control, the control gains are related to the virtual environment properties, such that different combinations of virtual environment parameters may result in different performance levels of the haptic interface. Further, it is reasonable to assume that critically damped case may have better results than other combinations of parameters. Second, the effect of sampling rate was evaluated. The higher the sampling rate the faster the system can react to the input, such that better performance is expected for higher sampling rate. Sampling rate of 1k S/s(Samples/second), 800 S/s, 500 S/s and 200 S/s were tested in the experiments. Third, the effect from sensor noise and acceleration calculation noise on the performance of haptic interface was investigated. This was not obvious in beginning, but during the experiments we found that without solving this problem first, other experiments for dynamic simulation cannot be carried out. A Butterworth low pass digital filter with cut off frequency of 20 Hz was implemented to reduce the effect of the noise. Results are illustrated for both before and after the implementation of the digital filter.

5.2 System calibration

The purpose of the calibration experiment is to find the exact relationship between the force exerted on a metal specimen and the voltage output from the strain gage mounted on it. In elastic working range this relationship should be linear, and the experiment should give the constant K and, at the same time, to validate the 2D-force sensor built with strain gage. Fig 5.1 shows the experimental setup for force sensor calibration.

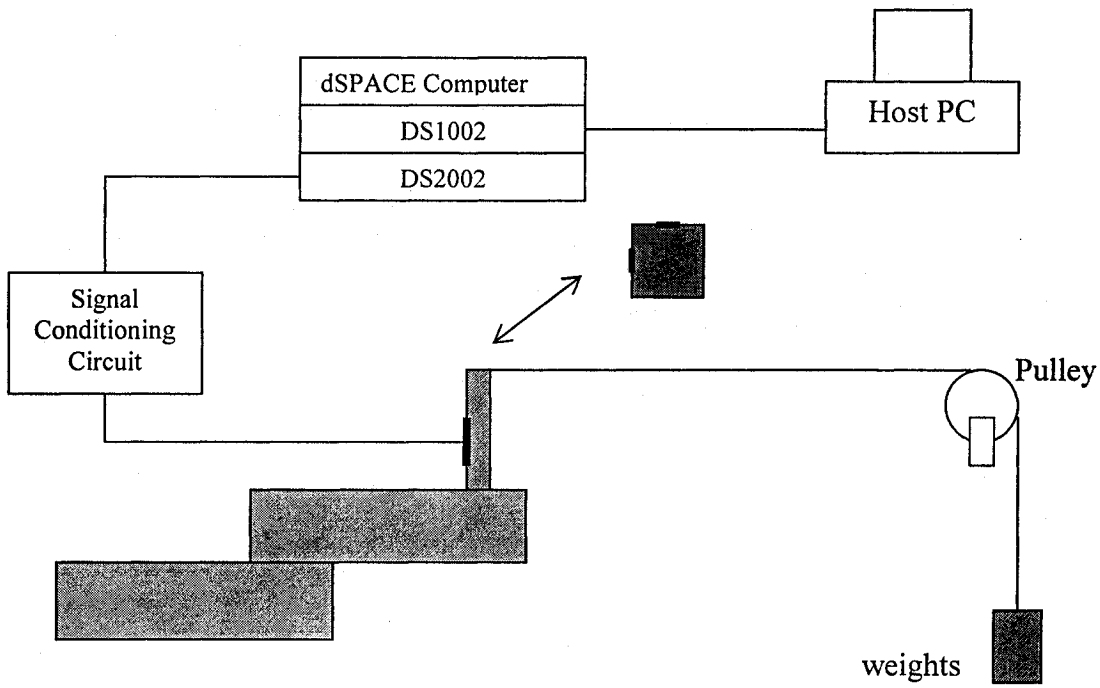


Fig 5.1 Force Sensor Calibration Setup

Weights ranging from 0 – 7 lbs were used to calibrate the force sensor. Table 5.1 presents the average value for 500 readings from the sensor for each weight applied. Two channels of DS2002 are connected to the two strain gages of the sensor through signal conditioning circuit respectively. Fig 5.2 and Fig 5.3 show the linearity of the voltage reading verses the exerted weights.

Force (lb)	Average Voltage (V)
0	0.0903
1	0.1034
2	0.1204
3	0.1365
4	0.1542
5	0.1732
6	0.1892
7	0.2064

(a)

Force	Average Voltage (v)
0	0.0902
1	0.1065
2	0.1238
3	0.1412
4	0.1583
5	0.1765
6	0.1922
7	0.2186

(b)

Table 5.1 (a) Calibration data of analog input channel 1

(b) Calibration data of analog input channel 2

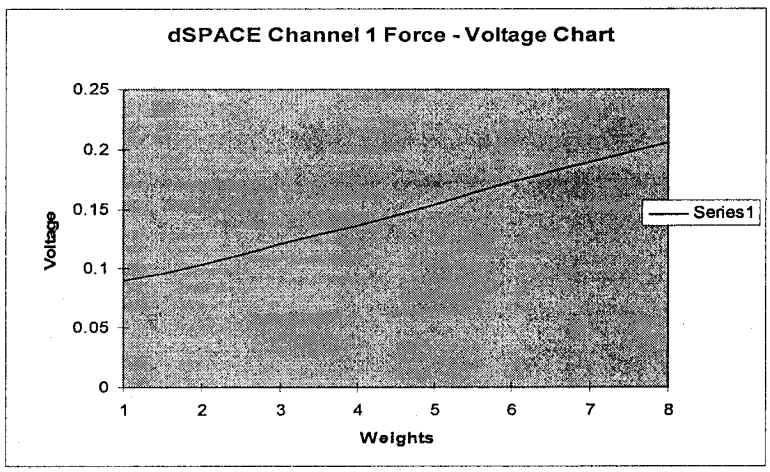


Fig 5.2 Calibration result for Channel 1

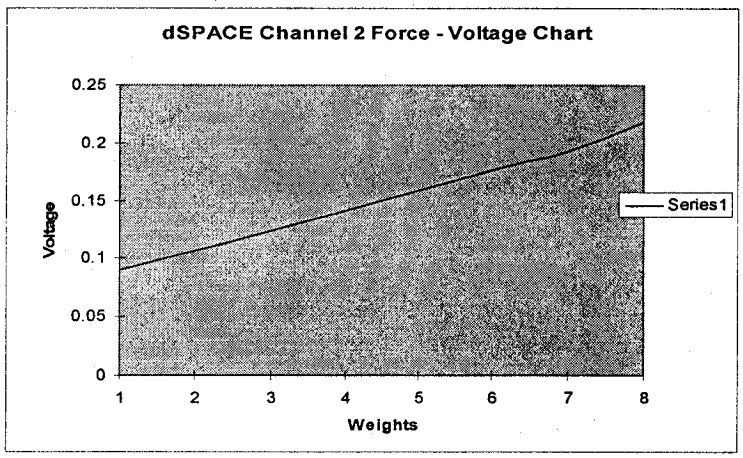


Fig 5.3 Calibration result for Channel 2

Summary of the Calibration

The result shows that the 2-D sensor provides us a linear working range. The two strain gages of the sensor are connected to the two analog input channels of dSPACE through two signal conditioning circuits, respectively. The calculation of the K value for channel 1 gave 0.0165v/lb, and for channel 2 gave 0.0183v/lb. After calibration, the dynamic behavior of the real object and of the haptic device can be compared. Hence, the performance of the haptic interface can be evaluated.

5.3 Experimental results for open loop impedance control

5.3.1 Virtual environment to be simulated for static case

Open loop impedance control was applied for static case based on the model described in section3.3.1. The virtual environment simulated to test the haptic interface consists of a mass block attached to two identical springs with K=458N/m, shown in fig

5.4, where, fx_1,fy_1,fx_2 and fy_2 are the forces in x and y direction of the base frame produced by the spring1 and spring2 respectively. In the static situation, the effect of the mass is not present. ΔX and ΔY are the displacements of the mass block with respect to the starting point, which is $\theta = 60^0$, and $L_1=L_2=10$cm. Equation 5.1 to equation 5.6 describe the relationship between the displacement in Cartesian space and the total force generated by the springs in X and Y directions when the mass block is pulled to any fixed position, for example the position shown in Fig 5.4 and were derived based on the geometry relationship between the new position and the original position (starting point).

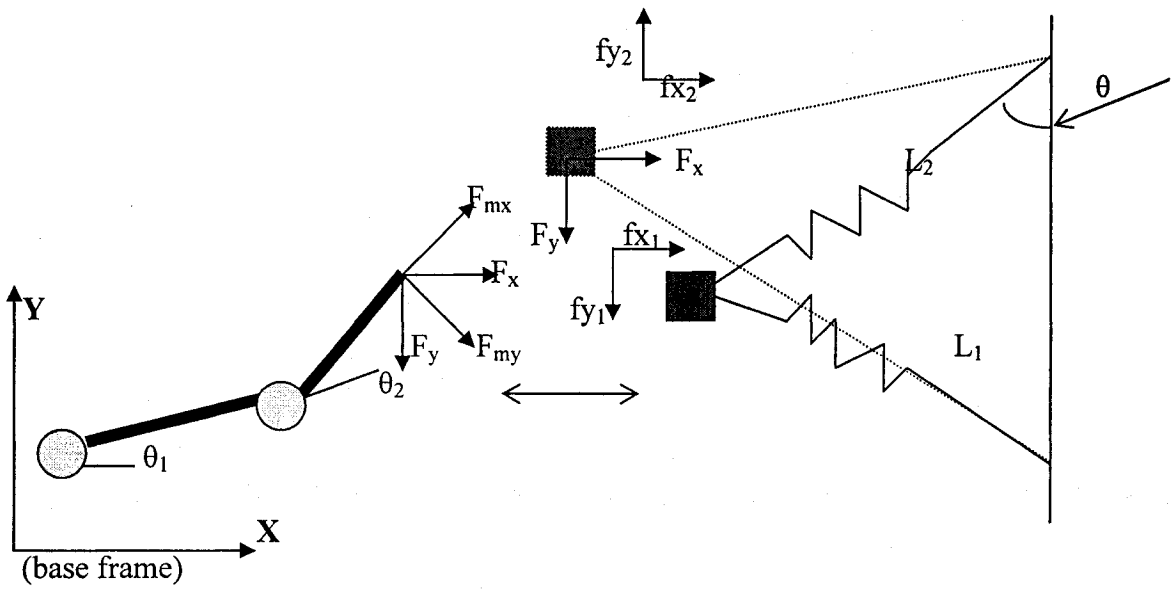


Fig 5.4 Virtual environment for static case simulation

$$fx1= \Delta L1*\cos(\text{arctg}\ ((L*\sin(90^0- \theta) + \Delta Y)/(L*\cos(90^0- \theta)+\Delta X))*K \quad (5.1)$$

$$fy1= \Delta L1*\sin(\text{arctg}\ ((L*\sin(90^0- \theta)+ \Delta Y)/(L*\cos(90^0- \theta)+\Delta X))*K \quad (5.2)$$

$fx2 = \Delta L2 * \cos(\text{arctg}((\Delta Y - L * \cos\theta)/(L * \cos(90^0 - \theta) + \Delta X)) * K$ (5.3)

$fx2 = \Delta L2 * \sin(\text{arctg}((\Delta Y - L * \cos\theta)/(L * \cos(90^0 - \theta) + \Delta X)) * K$ (5.4)

$Fx = fx1 + fx2$ (5.5)

$Fy = fy1 + fy2$ (5.6)

where, Fx is the total force produced by the two springs in X direction.

Fy is the total force produced by the two springs in Y direction.

$\Delta L1$ is the change of the length of spring1.

$\Delta L2$ is the change of the length of spring

The force vector can be written in the form of $F = \begin{vmatrix} Fx \\ Fy \end{vmatrix}$ and, substituted into equation (2.1), gives the torque command required to generate the equivalent force. The force is measured with respect to the end effecter coordinates frame and has to be converted to the base frame using following equations.

$$Fx = F_{mx}\cos(\theta_1 + \theta_2) - F_{my}\sin(\theta_1 + \theta_2) \quad (5.7)$$

$$F_Y = F_{my}\sin(\theta_1 + \theta_2) + F_{my}\cos(\theta_1 + \theta_2) \quad (5.8)$$

5.3.2 Experimental results

Experiments for both fixed position and slow motion were carried out to test the open loop impedance control method.

Test results for various fixed positions are shown in Fig 5.5 and 5.6.

(See next page)

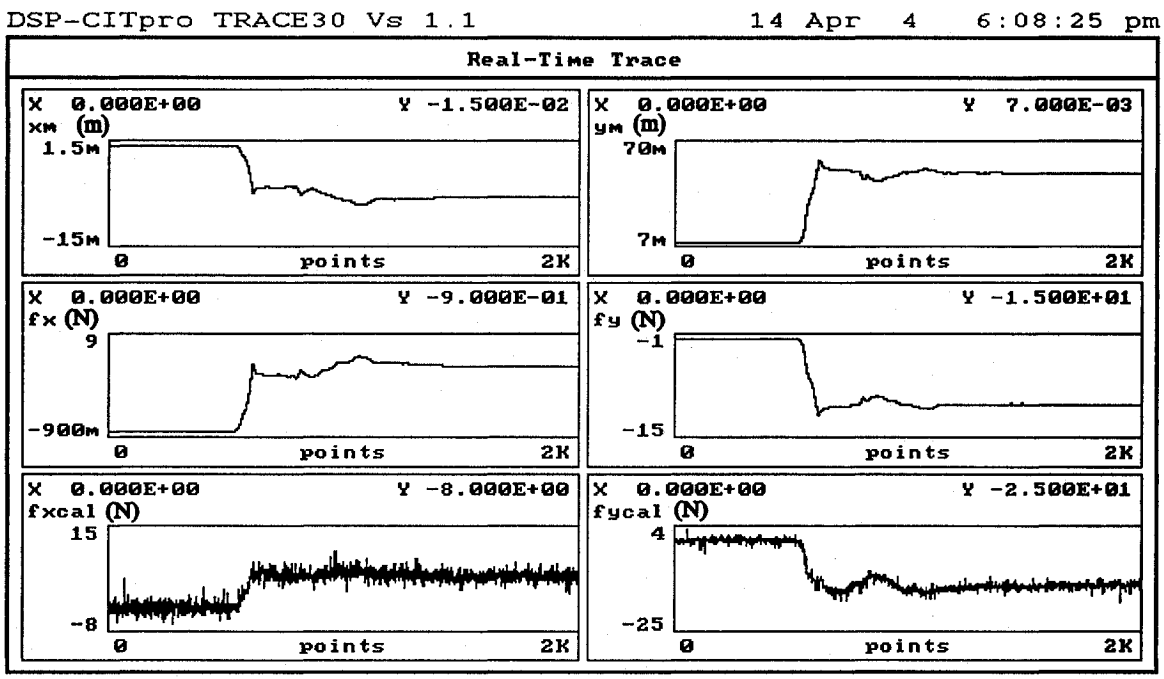


Fig 5.5 Static force measured for position (0.3727m, 0.0508m)

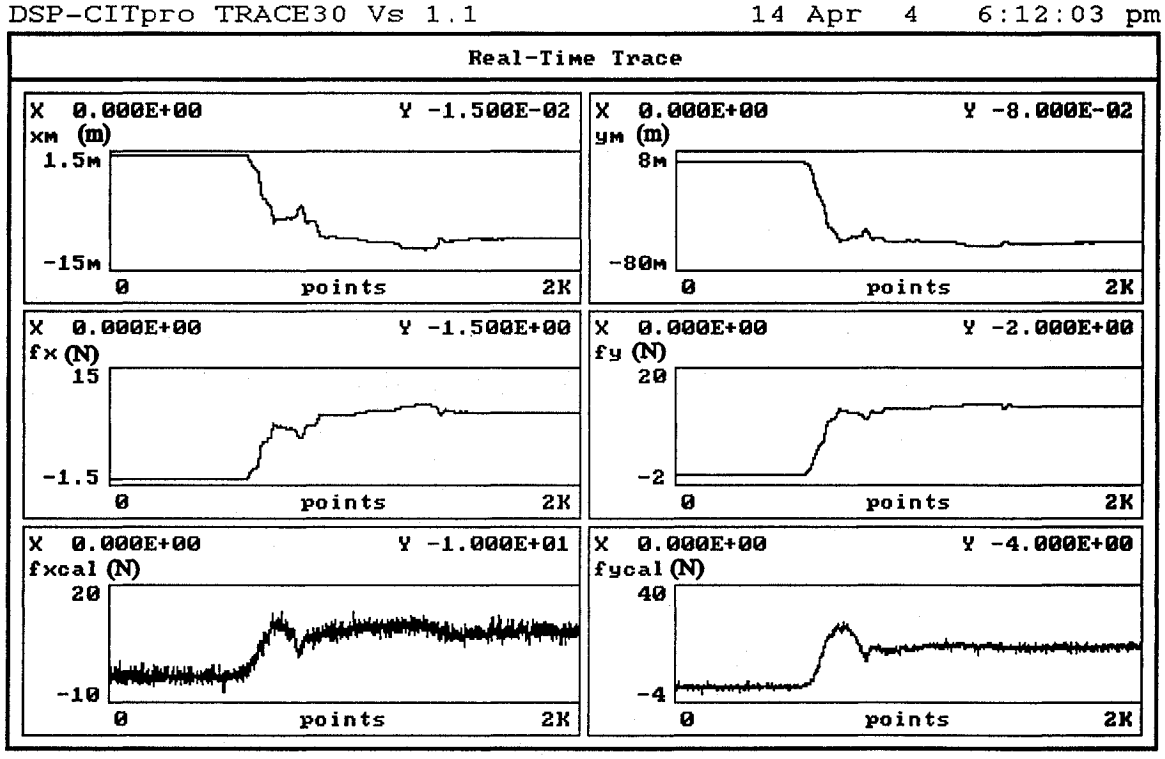


Fig 5.6 Static force measured for position (0.369m, -0.0594m)

Fig 5.5 and Fig 5.6 show the static force measurement for two positions. X_m and Y_m (first row of the chart) and refer to haptic device motion input by human operator in X direction and Y direction with respect to base frame and sensed by the motion sensor resolver. The second row (fx and fy) is the computed theoretical value of the force for motion input with respect to the original position (free of exerted force). The third row (fxcal and fycal) is the corresponding measured force. For all the experiments of this thesis, fx and fy always refer to the calculated theoretical value of the force in X and Y directions and fxcal and fycal always refer to the measured value of the force after calibration in X and Y directions. In total, more than 10 individual positions were tested, and some of the results are presented in table 5.2.

Position X (m)	Position Y (m)	Theoretic force magnitude (N)	Measured force (N) magnitude	Error %
0.3624	0.07919	23.1033	22.096	4.36
0.353	0.09599	32.6553	28.829	11.72
0.343	0.1046	41.087	36.744	10.57
0.3443	0.1167	44.043	40.748	7.48
0.3379	0.1186	48.587	46.2579	4.79
0.3504	0.09359	33.506	27.995	16.44
0.3608	0.1089	32.082	29.323	8.60
0.3528	0.1286	43.032	39.528	8.14

Table 5.2 Force measurements for various positions (X,Y)

In Fig 5.7 we can see the distribution of the theoretical value and several measured value in the working envelop of the robot arm.

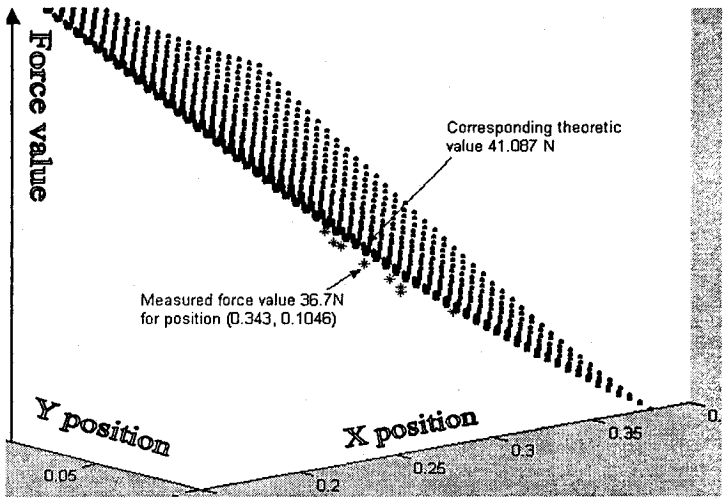


Fig 5.7 Measured and theoretical values of the interaction force versus position (X,Y)

Experimental results from Table 5.2 and Fig. 5.7 confirm that the measured force outputs values of the proposed HIL setup follow closely the force commands given an arbitrary motion applied by the human operator to the haptic device. The measured force values are close to the corresponding theoretical values, confirming the ability of the proposed HIL experimental setup to emulate a haptic interface, within 4-16% errors. Given the force sensing resolution of human being discussed in [2], this error is small enough.

Further, the open loop impedance control was tested for slow motion, and the result is shown in Fig 5.8.

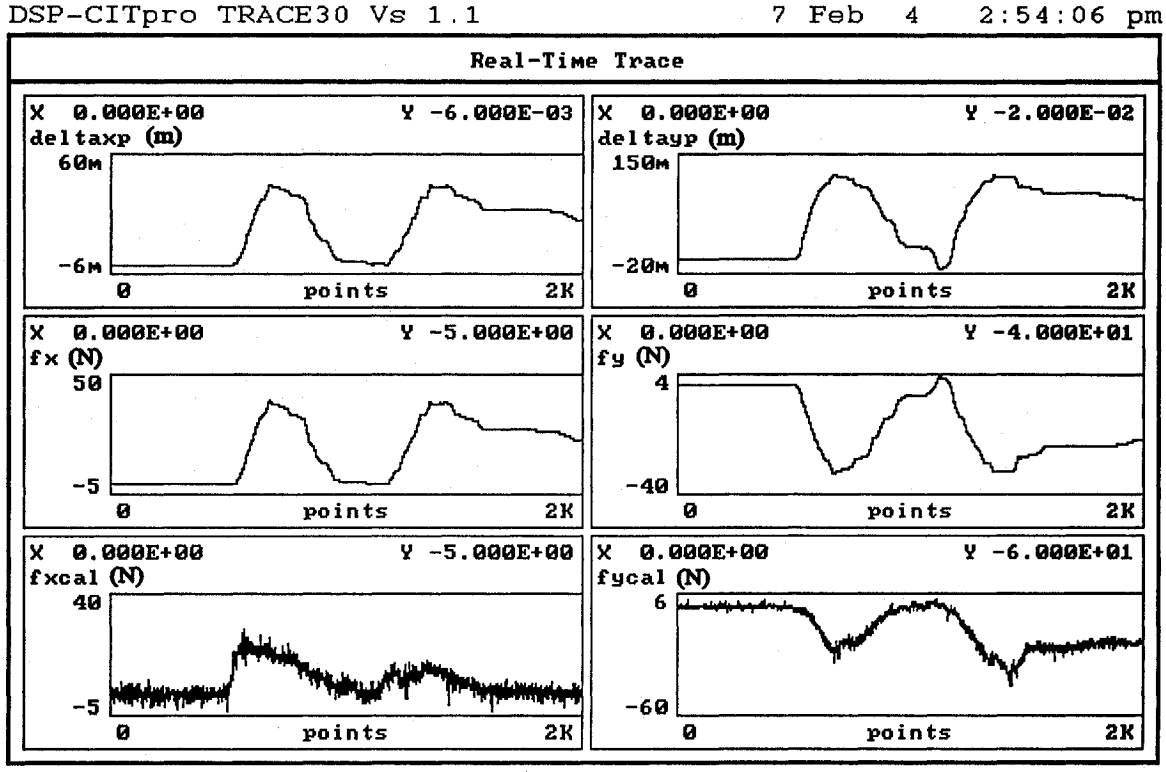


Fig 5.8 Force measurement for slow motion

In Fig. 5.8, similar notations were used except deltaxp and deltayp. Mathematically, the deltayp is equal toY_m of Fig 5.5 and 5.6, while deltaxp is equal to negative of X_m. This is because experiment of Fig 5.8 was conducted earlier and the necessity of notation change was realized and done recently. This result shows that for slow motion, the measured fore and the computed theoretical value are also well matched.

5.4 Experimental results for model based closed-loop impedance control

The experiments for model based close loop impedance control include effects from: sensor and acceleration estimation, virtual environment parameters, sampling rate and the applicability of the model based control for static case.

5.4.1 Virtual environment to be simulated

The virtual environment to be simulated has been described in section 3.3.2. In brief, along any degree of freedom, its dynamic behavior can be described by the differential equation:

$$-F_h - B\dot{X} - K(X - X_0) = M\ddot{X} \qquad (3.10)$$

where, F_h is the force feedback from virtual environment to the human operator, and M, B, K are constant. If there is only position change and insignificant acceleration and velocity or the object is moving very slow, this equation applies to a static case: $F_h=K(X_0-X)$. In the absence of F_h in static case, $X=X_0$, i.e. the equilibrium position.

5.4.2 Experiment 1: the effects of noise

The negative effects of noise had to be solved before other experiments could be implemented. In order to solve this problem a Butterworth low pass digital filter with cut-off frequency equal to 20 Hz, described in section 4, was applied to both analog input signals and to the acceleration estimation. In Fig 5.9, the first row is the analog input signal before the implementation of the digital filter; the second row is the signal after applying the digital filter. These signals are voltage signals directly read from

analog channels and haven't been converted to force yet. We can clearly see the difference between filtered and non-filtered signals. Fig 5.10 shows the noise from the numerical computation of the acceleration. Fig 5.12 shows the digitally filtered acceleration estimation. This noise has significant influence on the results, because its instantaneous values of the noisy acceleration are so large that calculated forces could be ten times greater than measured one. We notice this by comparing the ranges of value for acceleration in Fig 5.10(approx ±200) and Fig 5.12(approx ±20). Noise from analog channel also causes problems sometimes. For relative larger force exerted, the torques to be applied to joints, which are calculated from measured force, could be out of range (beyond the capability of motor) as shown in Fig 5.11. For example, tc1 & tc2 are calculated torques and t1 and t2 refer to the actual value send to the two motors. Fig 5.13 shows that after implementing digital filter, the calculated torques fall in a reasonable range. In particular, we can observe that after noise filtering the torque commands t1 and t2 are practically the same value as the calculated torques tc1 and tc2 after the safety check of the program.

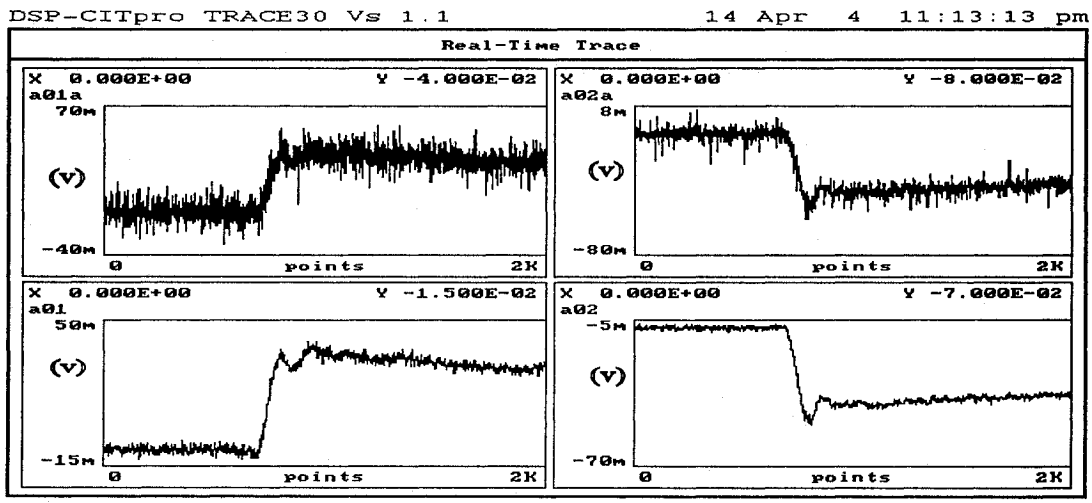


Fig 5.9 Analog voltage signal input before and after applying the digital filter

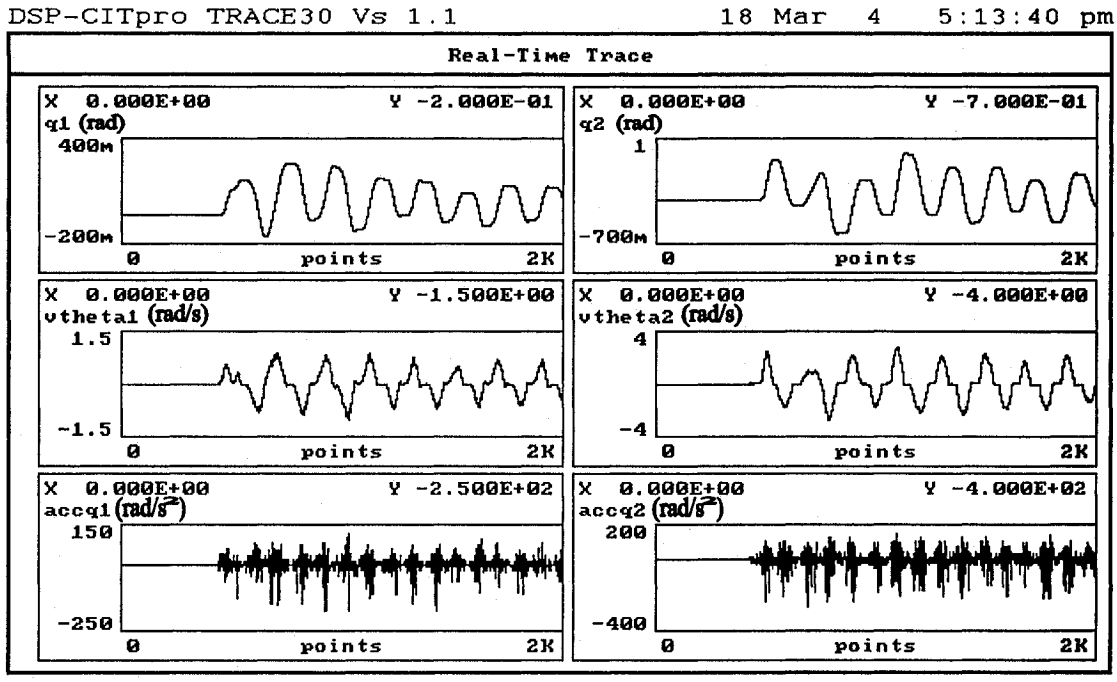


Fig 5.10 Noise from acceleration estimation

In this figure, q1 and q2 refer to angular position, vthetal and vtheta2 refer to angular velocity, and accq1 and accq2 are the estimated accelerations.

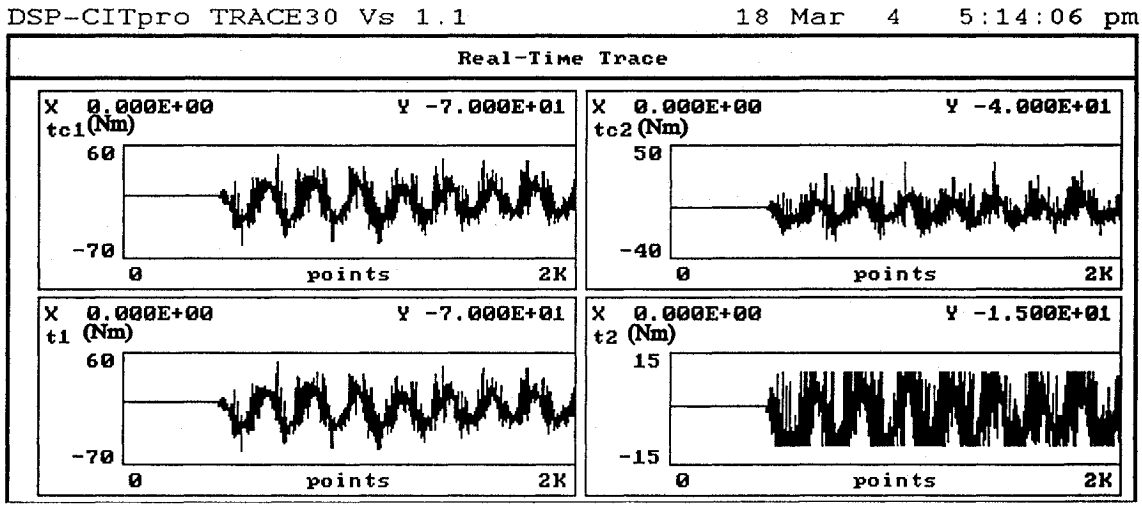


Fig 5.11 Calculated torques tc1 and tc2 through measured forces and torque commands t1 and t2

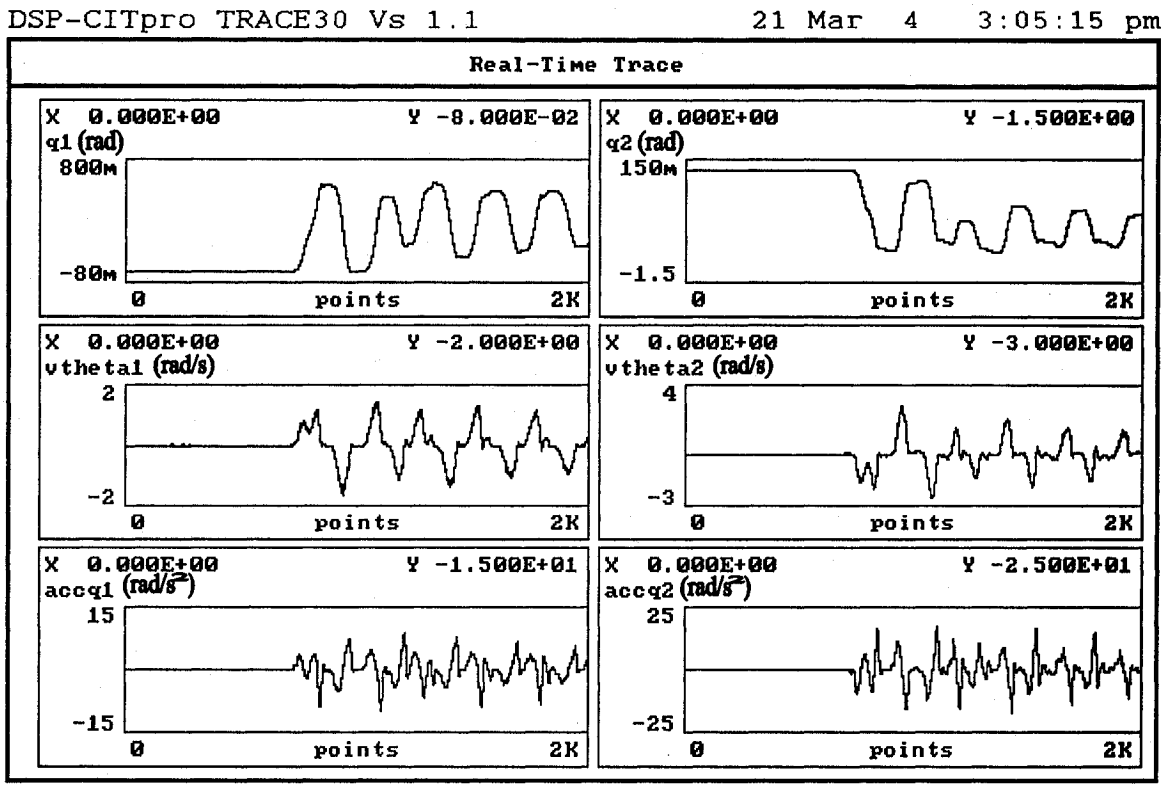


Fig 5.12 Acceleration estimation after digital filtering

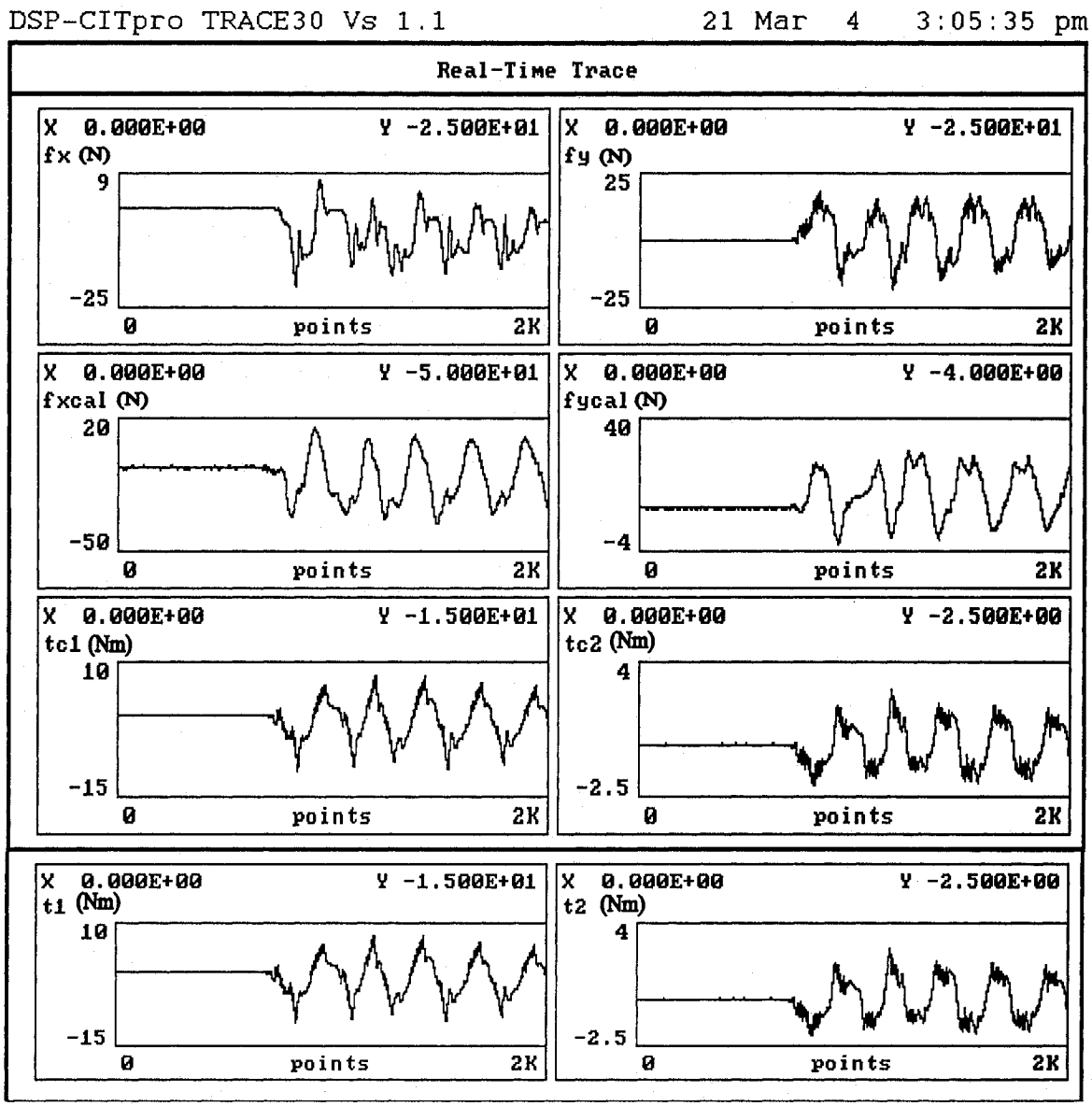


Fig 5.13 Forces and torques measured and calculated after digital filtering

5.4.3 Experiment 2: the effects of virtual environment parameters

The virtual environment is generally described by :

$$-F_h - B\dot{X} - K(X - X_0) = M\ddot{X} \qquad (3.10)$$

From previous control diagram shown in fig 3.7 (linear control law part), it is reasonable to assume that the changes of M, K, B values may have different contributions to the performance of the haptic interface. Further, in critically damped case, gives better results than other parameter combination.

Experimental parameters and conditions:

* sampling rate 1k S/s
* Cut-off frequency 20HZ for Butterworth filter
* K=200 N/m
* M=4kg, 2kg
* Adjust the value of B to achieve the case of underdamping and criticaldamping.

The performance was judged based on the error between the measured force and its computed value. Fig 5.14 shows the result of critically damped case for M=2kg, B=30Ns/m. Fig 5.15 shows the result of critically damped case for M=4kg, B=44Ns/m. Fig 5.16 shows the result of a little damped case with M=4kg, B=18Ns/m, and Fig 5.17 shows another very little damped case with M=2kg, B=8Ns/m. Other combination of M and B were also tested. The results are listed in table 5.3 and the trend chart illustrating the change of error verses the change of constant B is shown in Fig 5.18.

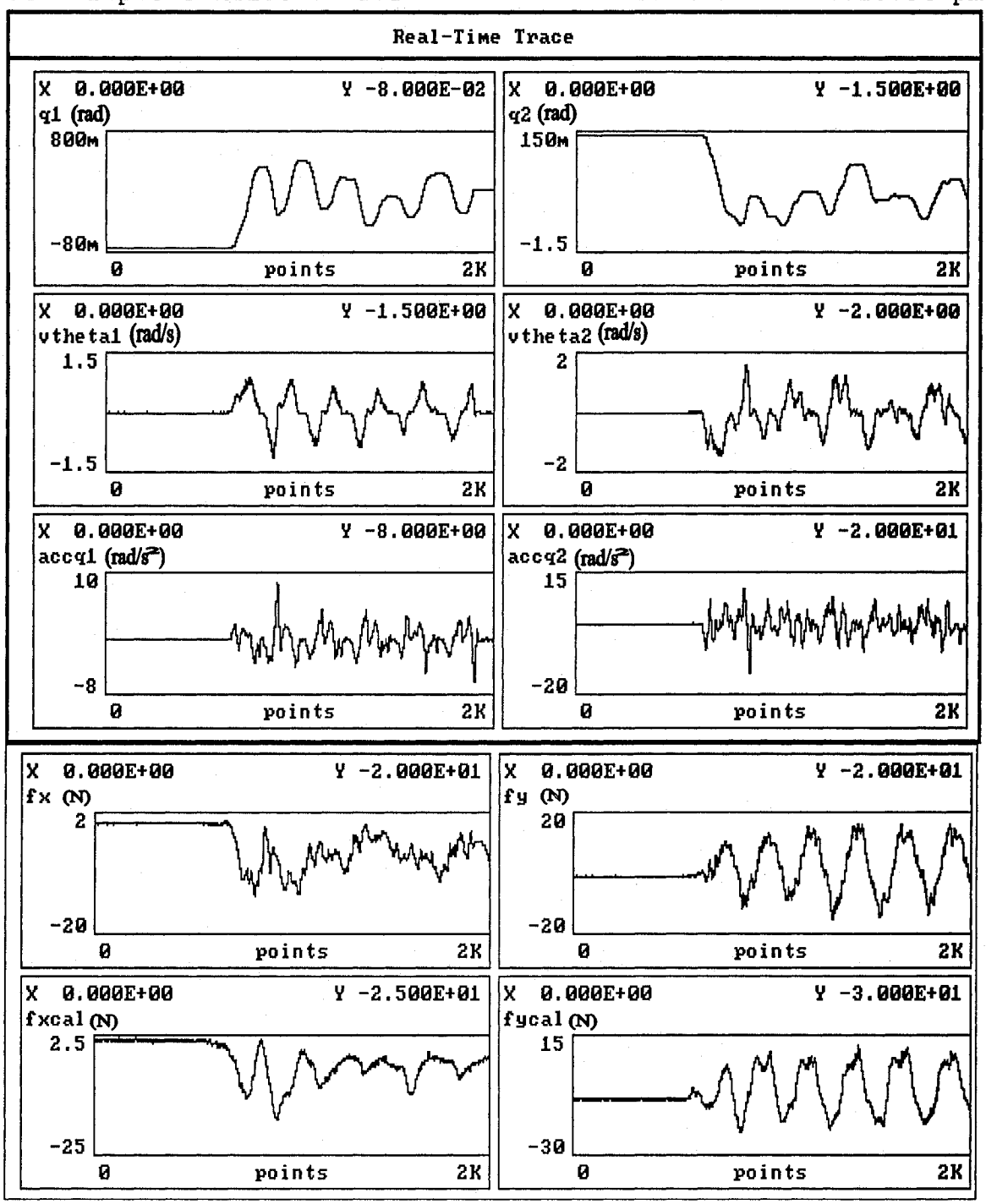


Fig 5.14 Results for critically damped case (M=2kg, B=30)

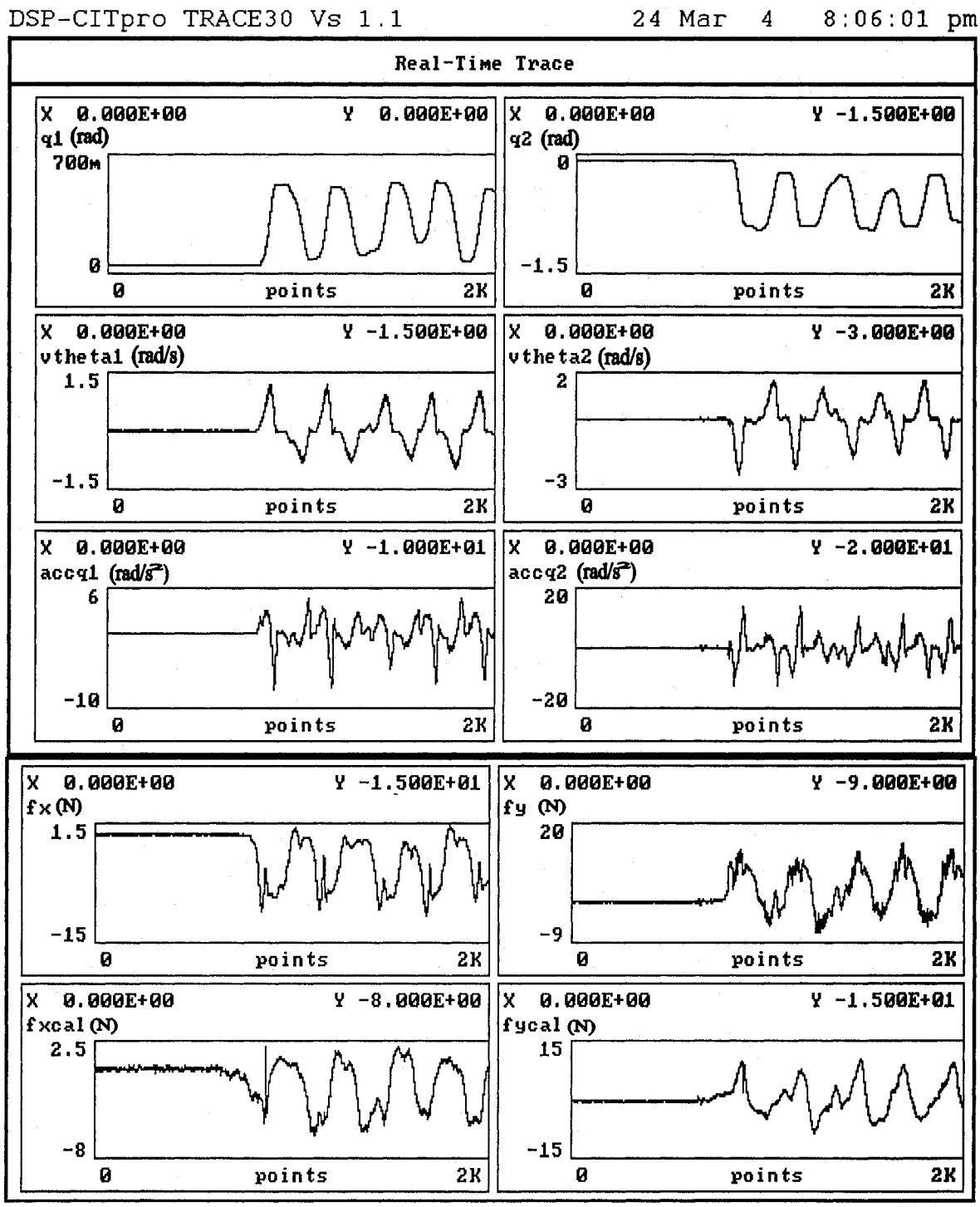


Fig 5.15 Results for critically damped case (M=4kg, B=44)

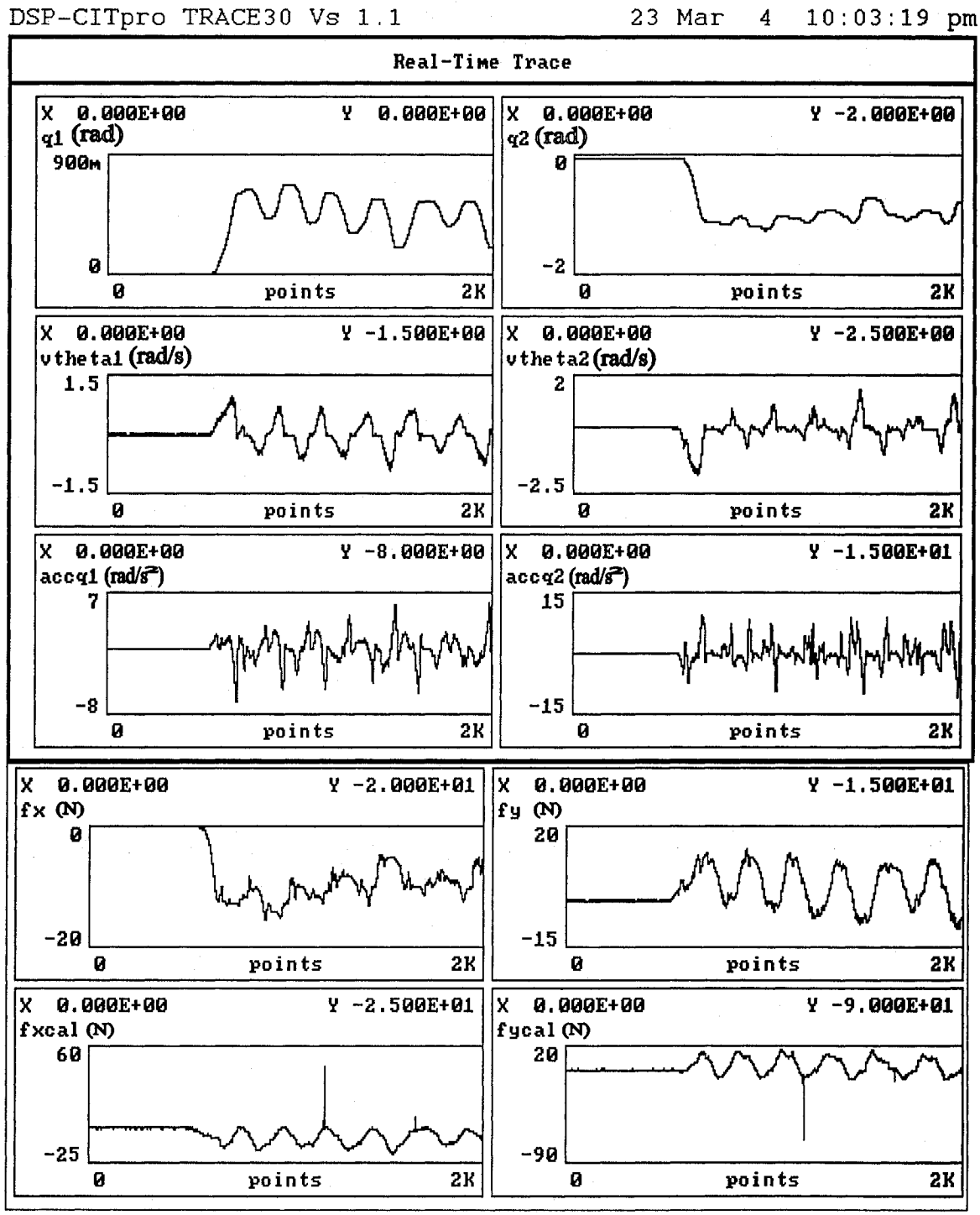


Fig5.16 Result for underdamped case (M=4, B=18)

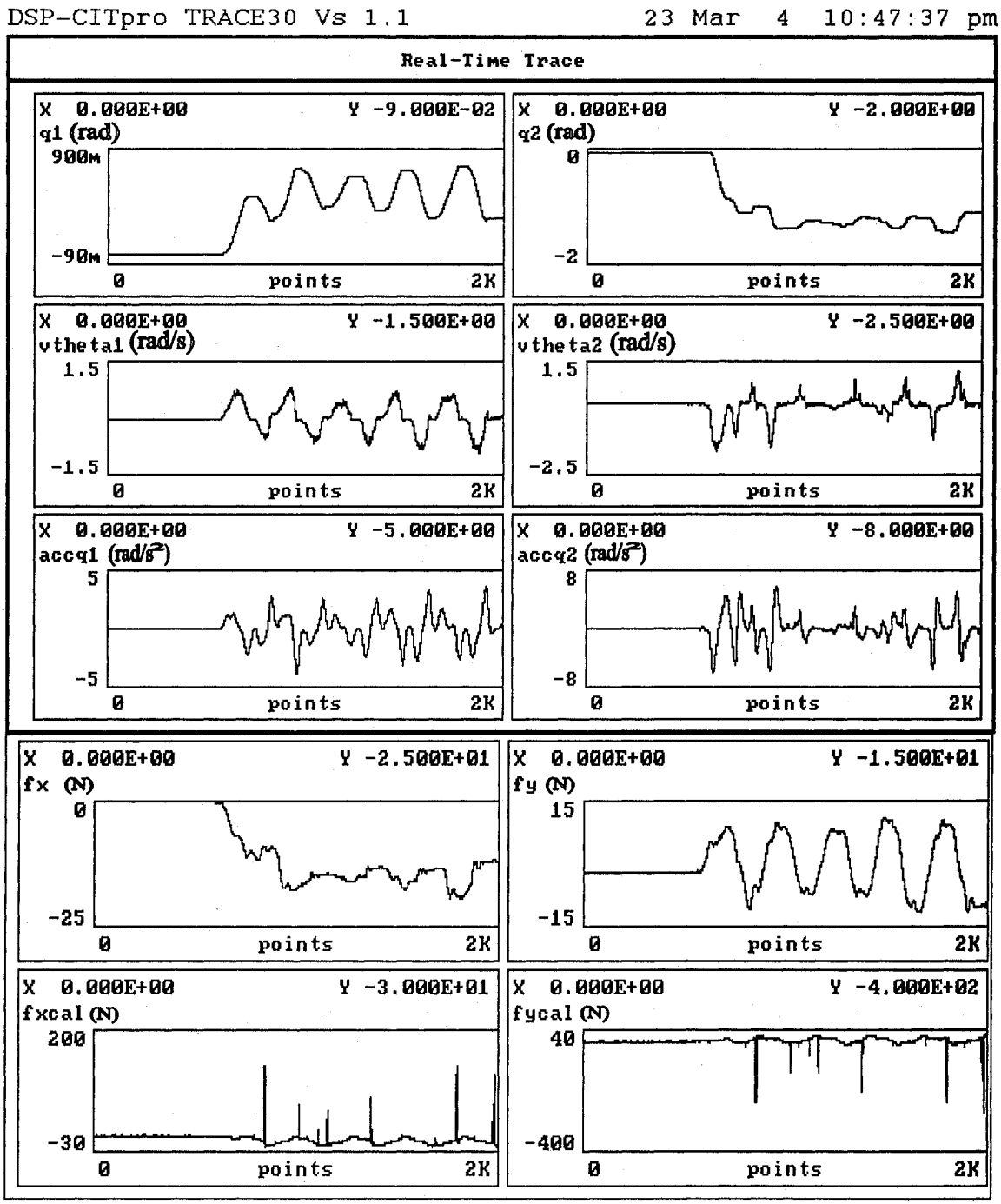


Fig 5.17 Result for underdamped case (M=2, B=8)

M \ B	16	18	20	24	30	32	40	60
2 kg	31.2%		22.9%	17.2%	14.8%	13.9%		
4 kg		27.8%		21.2%	15.9%	16.8%	12.8%	14.0%

Table 5.3 Errors of different mass and viscosity combinations

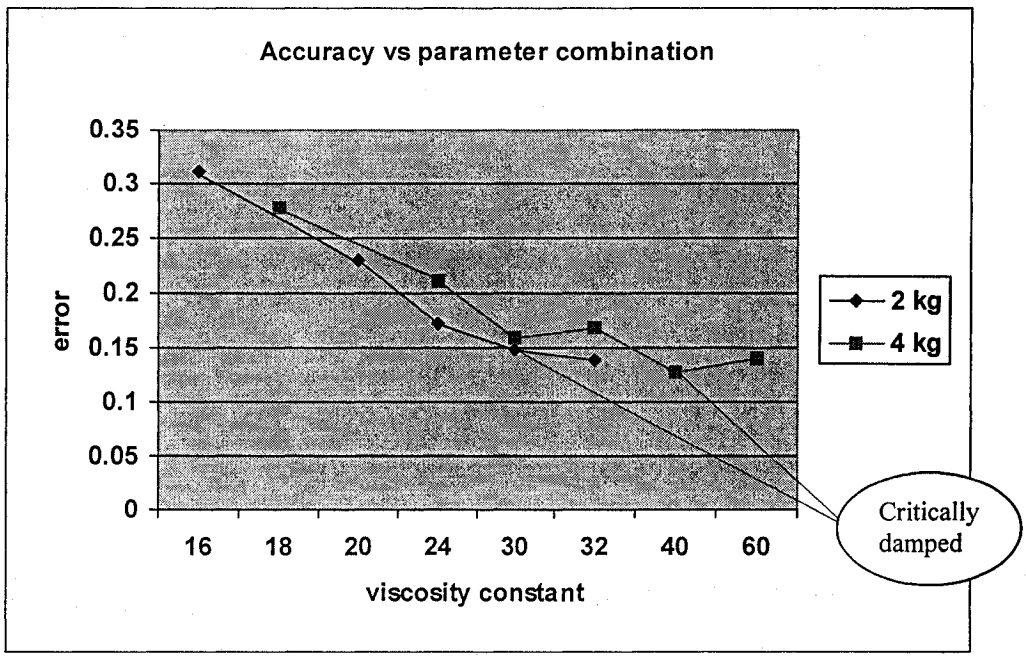


Fig 5.18 Trend chart – Error vs Change of B

This confirms that critically damped case results in better performance.

5.4.4 Experiment 3: The effects of sampling rate

This experiment was designed to observe the influence of sampling rate on the performance of haptic interface. Previous experiment was conducted at the sampling rate 1k, which is high enough to our application. In this experiment, sampling rate of 200, 500 and 800 samples/sec were implemented. Results are shown in the following figure. The cut-off frequency of 20Hz requires different coefficients in the equation for different sampling rates. They are listed in following table 5.4. From fig 5.19 to fig 5.22, results of sampling rate at 1 k, 800, 500 and 200 S/s [Samples/sec] are illustrated accordingly. Fig 5.23 shows the trend of the error with respect to sampling rate.

Sampling rate	b(1)	b(2)	b(3)	a(2)	a(3)
1k	0.00362168	0.00724336	0.00362168	-1.82269495	0.83718165
800	0.0055427	0.0110854	0.0055427	-1.7786317	0.8008026
500	0.01335920	0.02671840	0.01335920	-1.64745998	0.70089678
200	0.06745527	0.13491054	0.06745527	-1.14298050	0.41280159

Table 5.4 Digital filter parameters at different sampling rates

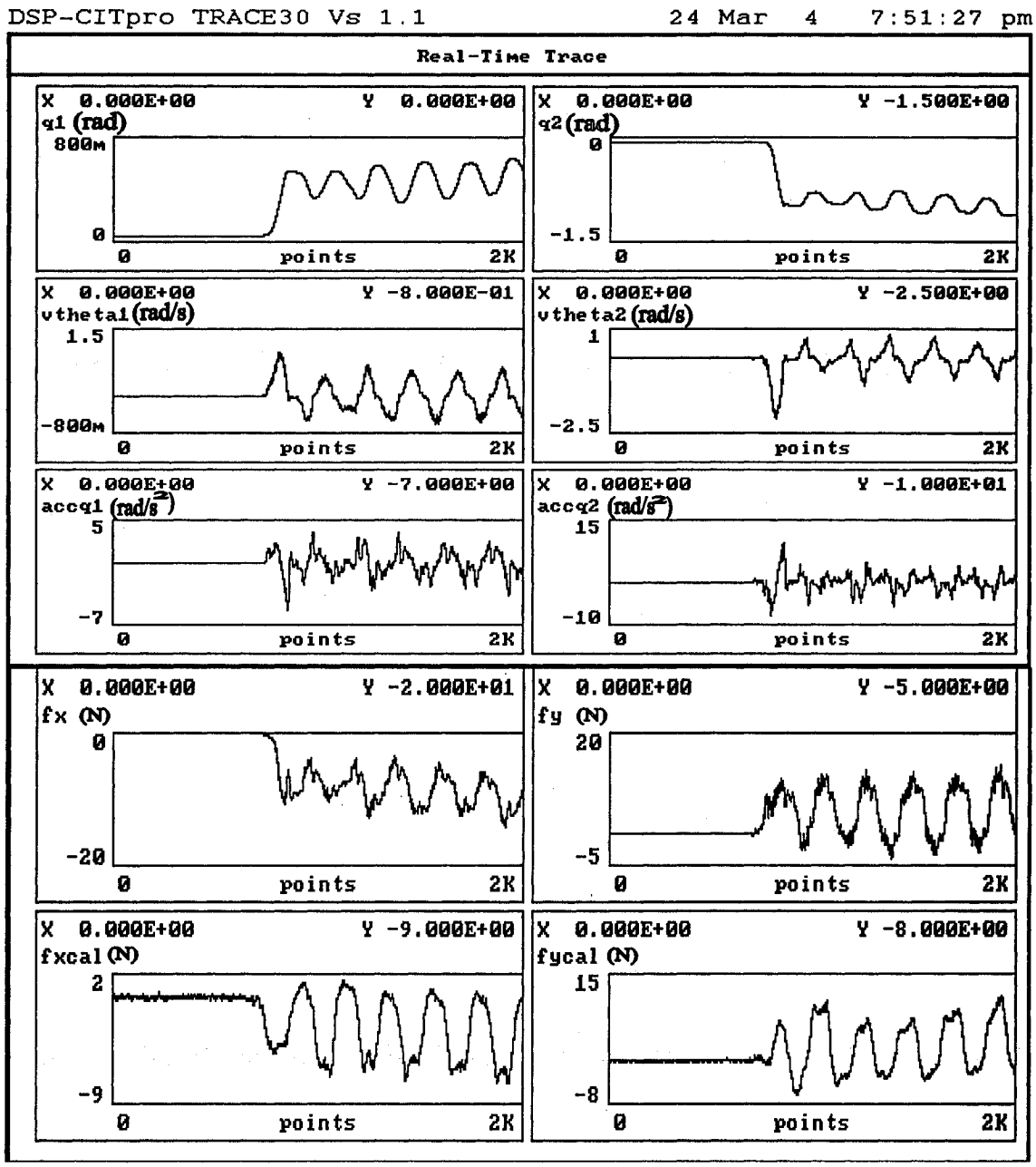


Fig 5.19 Results for sampling rate 1k S/s

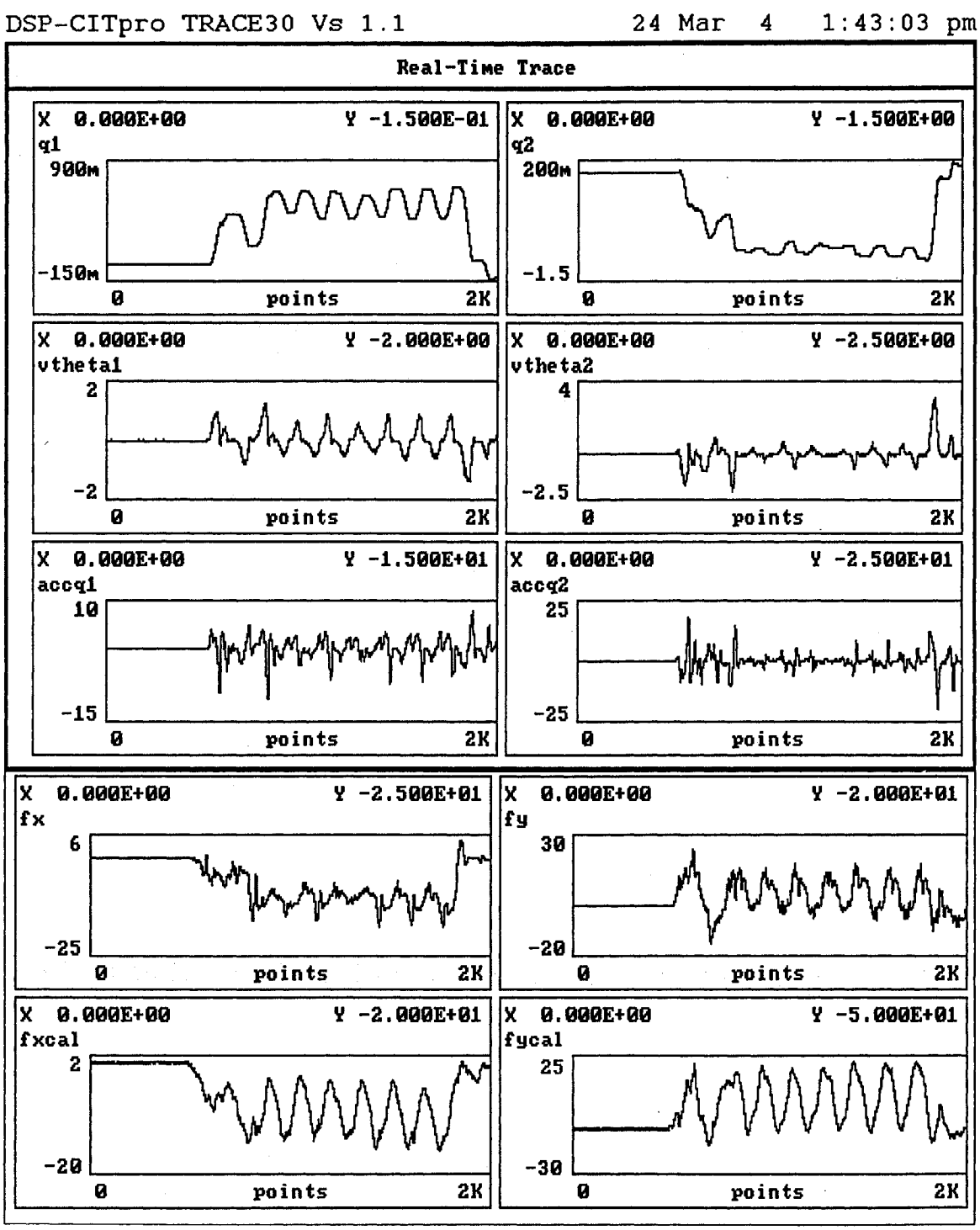


Fig 5.20 Results for sampling rate 800 S/s

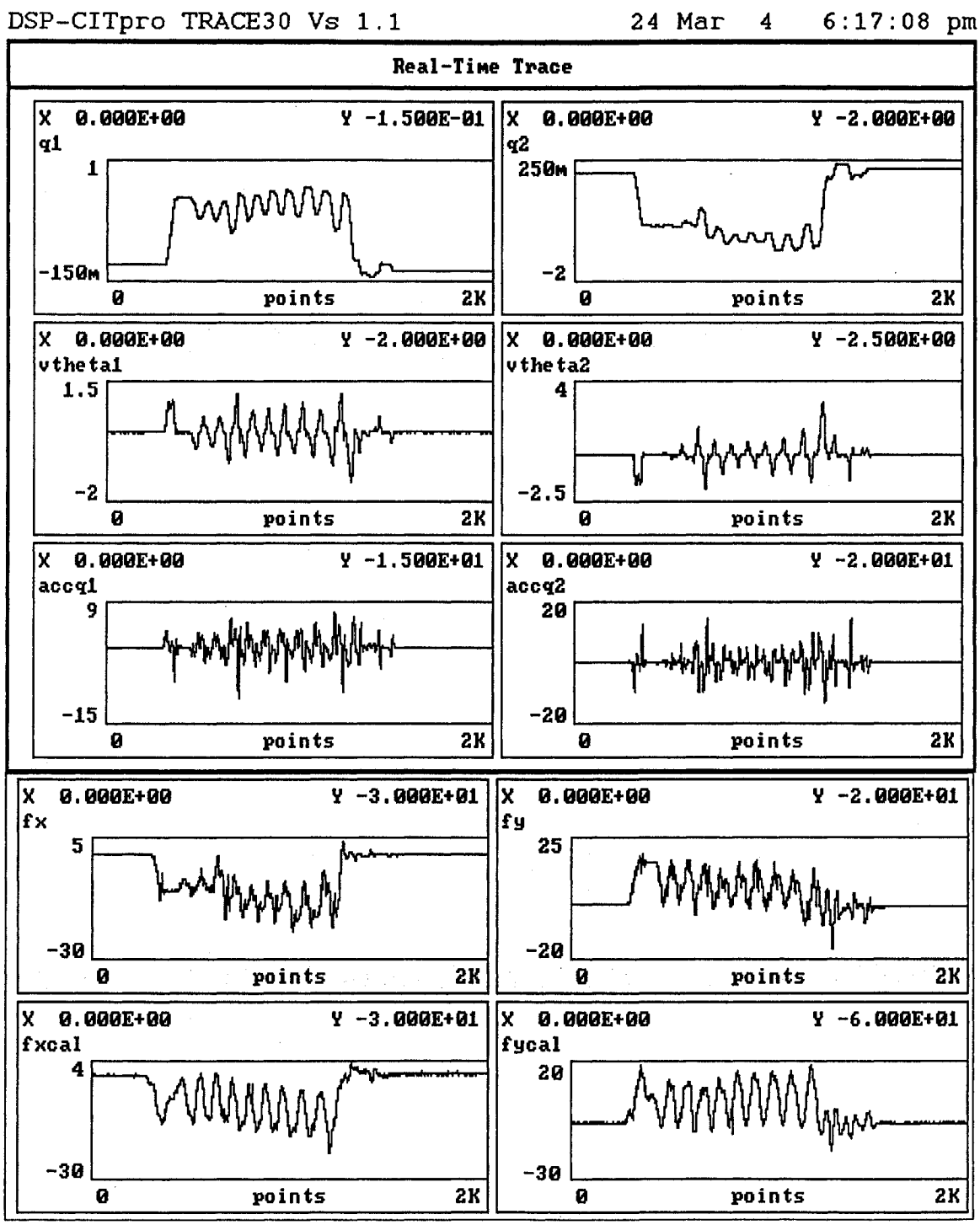


Fig 5.21 Results for sampling rate 500 S/s

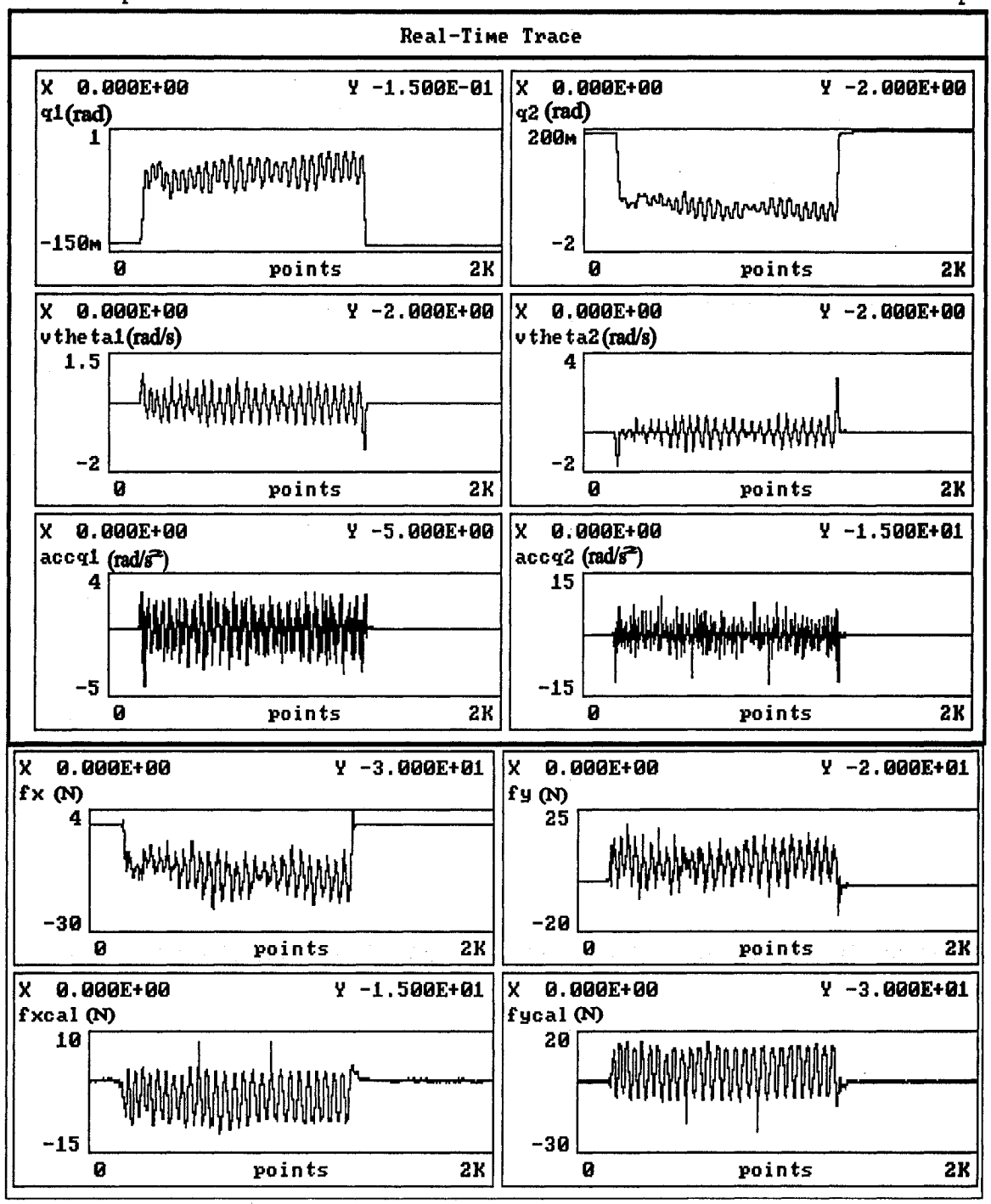


Fig 5.22 Results for sampling rate 200 S/s

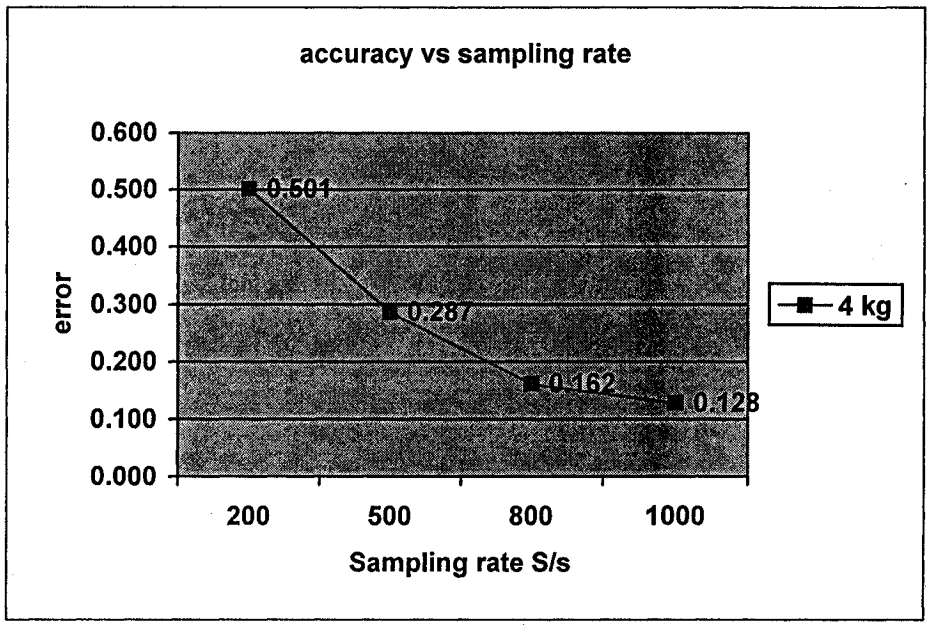


Fig 5.23 Trend chart – error vs sampling rate

This confirms that with the decrease of sampling rate, errors are increasing.

5.4.5 Experiment 4: Closed loop impedance control for static case

In this experiment, the model based impedance control was also implemented for static case. Fig 5.24 shows the results of a position test using the model based impedance control. Several points were tested, and the results show that for static case the performance of the two control approaches is very close. The average error of the model based control for static case is around 12%.

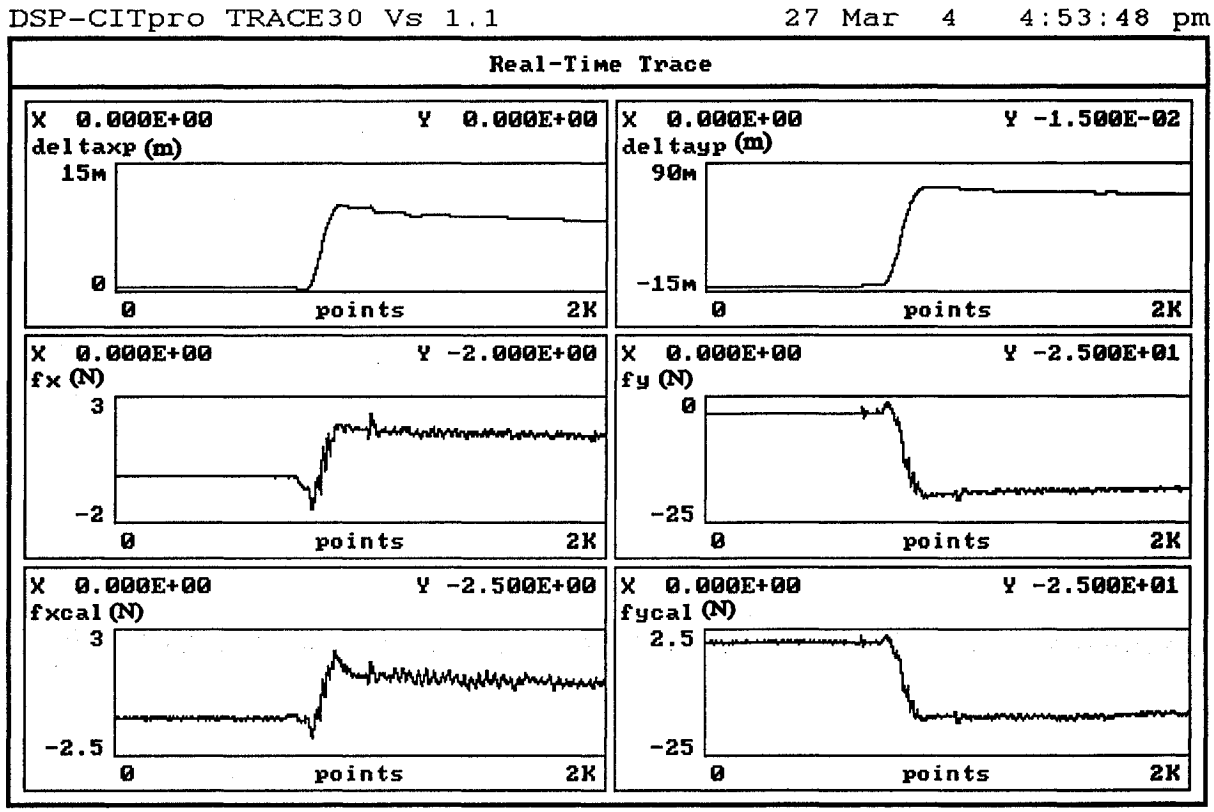


Fig 5.24 Result for static case using the model based closed loop control

The experiment verified that the model based impedance control developed in this thesis can be also used in static case.

5.4.6 Experiment 5: simulation for very stiff and very soft virtual environments

The model based impedance control was also tested for extreme cases, a very stiff virtual environment and a very soft virtual environment. Fig 5.25 shows the result for a very stiff virtual environment with K= 10000. We can see from the result that for a motion around 0.023m, the feedback force reached around 220N, and one motor reached its upper limit of torque capacity and started to vibrate. Greater value of K was tried, but the motors reached unstable working conditions.

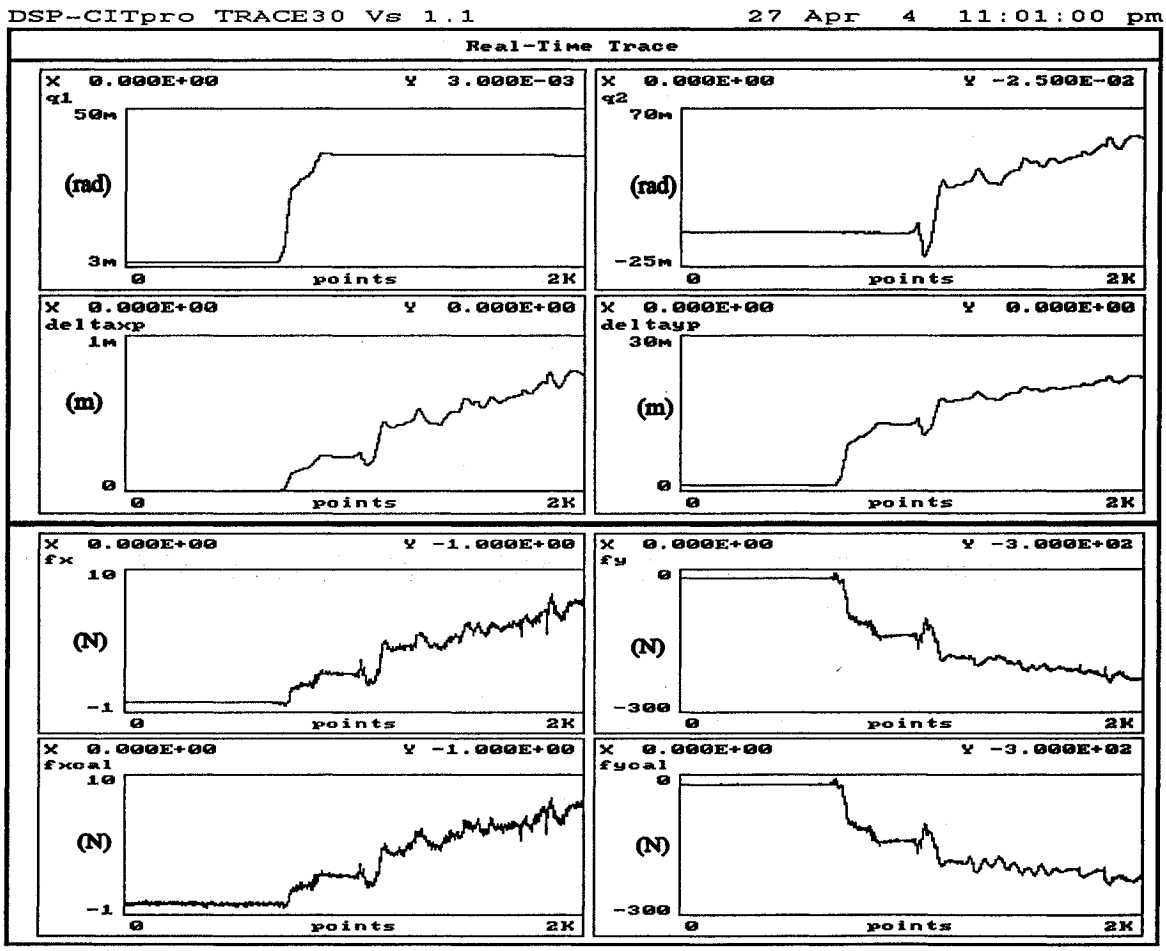
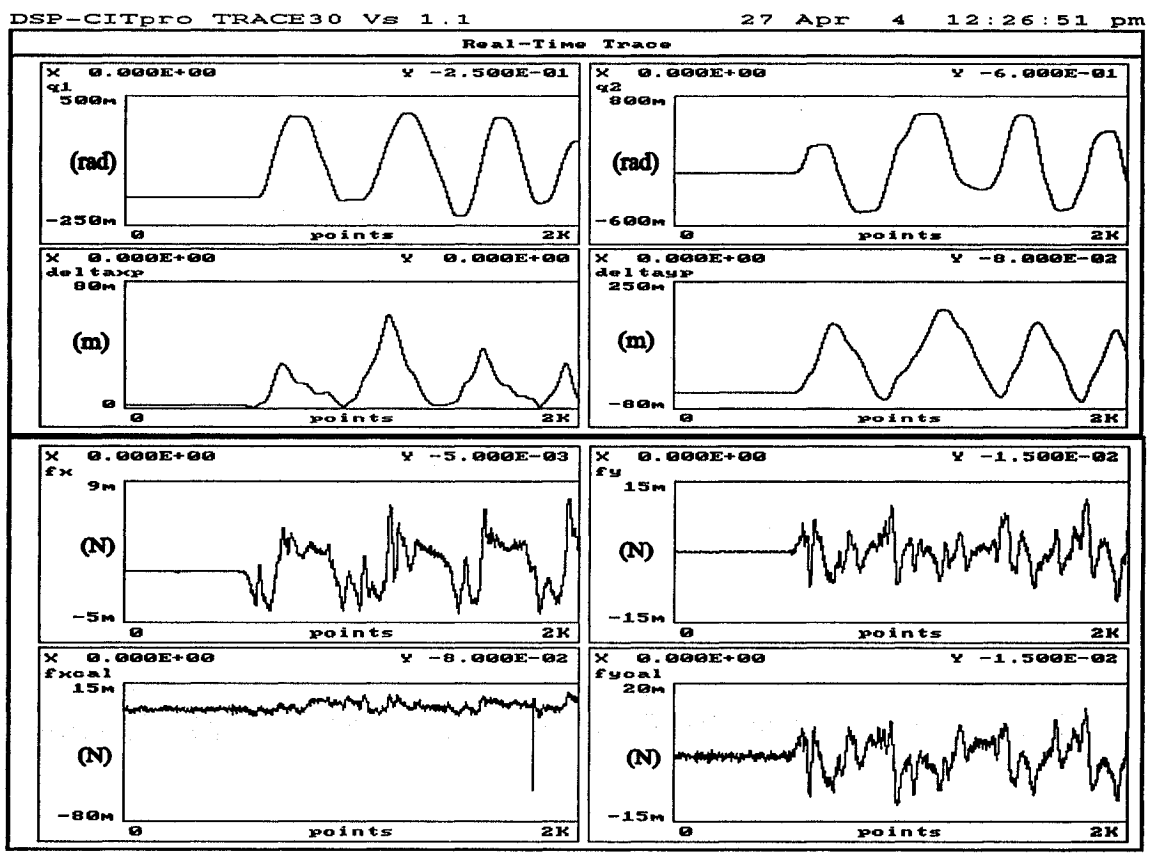


Fig 5.25 Simulation for a very stiff virtual environment (K=10000)

Fig 5.26 shows the simulation results for a very soft virtual environment with K=0.01, M=0.1, and B=0.01. We can notice that for input motion around 0.1m, and the measured feedback force is about 0.01N which almost cannot be sensed by human operator.



5.26 Simulation for a very soft virtual environment (K=0.01)

6 Conclusions and Future works

6.1 Conclusions

The following conclusions can be drawn from the research:

1. HIL experimental approach, that combines the real hardware and mathematically simulated components, provides an ideal platform for testing conceptual designs for haptic interfaces, before prototypes are available. The proposed HIL experimental setup in this work proved to be a feasible solution for developing haptic interfaces.
2. The difference between conventional applications of impedance control for robot motion control and its application in developing haptic interfaces was analyzed. In haptic interfaces, the dynamic properties of virtual environment are transformed to human operator through a haptic device in the form of mechanical impedance.
3. For the open loop impedance control methodology, implemented for simulation of static case, the errors between HIL measured force and theoretical values were about 10%. This confirms that, a general-purpose robot under open loop impedance control, can be build as a haptic device for static case.
4. A model based impedance control methodology, introduced and implemented using the HIL experimental setup, proved to be a feasible approach for building haptic interface using a general-purpose robot arm for both dynamic and static case.

5. For a haptic interface, noise, sampling rate and virtual environment parameters could have significant effects on the performance of the haptic interface. These effects discussed and analyzed experimentally in the thesis show important influence on the errors between HIL measured values and theoretical values.

6.2 Future works

Further research could be continued in two directions:

1) Model based approach could be used for admittance control in which trajectory following simulation of a virtual object can be simulated. A robot arm with more degrees of freedom will have to be used as a haptic device development system. At the same time, a force/torque sensor with more degrees of freedom is also needed.
2) Haptic interface used for telemanipulating could also be built with two robot arms. In this case, the virtual environment will be replaced by a slave robot arm working on a remote environment, while a master robot arm will act as the haptic device. Human operators would control the remote slave robot arm by manipulating the master robot arm.

7 References

[1] Necsulescu, D.S., Mechatronics, Prentice Hall, 2001.

[2] Burdea, G.C., Force and Touch Feedback for Virtual Reality, NewYork:Wiley,1996.

[3] Clover, C.L., A control-system architecture for robots used to simulate dynamic force and moment interaction between humans and virtual objects, IEEE Trans. Syst., Man,Cybernet., *C*, Vol.29, No.4, Nov 1999, pp.481-493.

[4] Clover, C.L., Luecke, G.R., Troy, J.J., and McNeely, W.A., Dynamic simulation of virtual mechanisms with haptic feedback using industrial robotics equipment, Proc. IEEE Conf. Robotics and Automation, Albuquerque, NM, Apr 1997, pp.724-730.

[5] Carignan, C.R. and Akin, David, Using Robots for Astronaut Training, IEEE Control Systems Magazine, April 2003.

[6] Carignan, C.R. and Cleary, K.R., Closed-loop force control for haptic simulation of virtual environments, Electron. J. Haptics Res., Vol.2, no.2, Feb.2000 [Online]. Available: http://www.haptics-e.org

[7] Hogan, N., Impedance control: an approach to manipulation (part 1-theory), The Journal of Dynamic Systems, Measurement, and Control, Vol.107, March 1985, pp.1-7.

[8] Hogan, N., Impedance control: an approach to manipulation (part II-implementation), The Journal of Dynamic Systems, Measurement, and Control, Vol.107, March 1985, pp.8-16.

[9] Hogan, N., Impedance control: an approach to manipulation (part III-Applications), The Journal of Dynamic Systems, Measurement, and Control, Vol.107, pp.17-24, March1985.

[10] Hogan, N., Controlling impedance at the man/machine Interface, in Proc. IEEE Int.Conf. Robot. Automat., Scottsdale, AZ, 1989, pp.1626-1631.

[11] Hogan, N., Stable Execution Of Contact Tasks Using Impedance Control, The Proceedings of the IEEE International Cofference on Robotics and Automation, 1987, pp.1047-1054.

[12] Kazerooni, H., Human/Robot Interaction via the Transfer of Power and Information Signals Part I: Dynamics and Control Analysis, IEEE Trans.Syst.Man Cyber., Vol.20, No 2. 1990

[13] Kazerooni, H., Sheridan, T.B. and Houpt, P.K., Robust Compliant Motion for Manipulators, Part I: The Fundamental Concepts of Compliant Motion, IEEE Journal Of Robotics And Automation, Vol.RA-2, No.2, June 1986, pp. 83-92.

[14] Kazerooni, H., Houpt, P.K. and Sheridan, T.B., Robust Compliant Motion for Manipulators, Part II: Design Method, Vol.RA-2, IEEE Journal Of Robotics And Automation, No.2, June 1986, pp.93-105.

[15] Kazerooni, H., Robust, Non-liner Impedance Control for Robot Manipulators, IEEE International Conference, Vol 4, Mar 1987, pp.741-750.

[16] Kazerooni, H. and Her, Ming-Guo, The Dynamics and Control of a Haptic Interface Device, IEEE Transactions on Robotics and Automation, Vol.10, No.4, August 1994, pp. 453-464.

[17] Hannaford, Blake and Ryu, Jee-Hwan, Time-Domai Passivity Control of Haptic

Interfaces, IEEE Transactions on Robotics and Automation,Vol.18, No.1, February 2002, pp.1-10.

[18] Volpe, R.A., Real and Artificial Forces in the Control of Manipulators: Theory and Experiments, PhD thesis, 1990, http://robotics.jpl.nasa.gov/people/volpe/papers, pp58.

[19] Payandeh, S. and Goldenberg, A.A., A Robust Force Controller: Theory and Experiments, Proceedings of the IEEE International Conference On Robotics and Automation, Sacramento, California, Apr 1991, pp36-41

[20] Inoue, Y., Sakaki, T., Matsumoto, T. and Iwakane, T., Impedance Control of a Multi-D.O.F. Direct-Drive Manipulator, Vol.2-162

[21] Tafazoli, S., Salcudean, S.E., Hashtrudi-Zaad, K. and Lawrence, P.D., Impedance Control of a Teleoperated Excavator, IEEE Transactions On Control Systems Technology, Vol.10, No.3, May 2002, pp.335-367

[22] Albu-Schäffer, A. and Hirzinger, G., Cartesian Impedance Control Techniques for Torque Controlled Light-Weight Robots, Proceedings of the 2002 IEEE International Conference on Robotics & Automation, Washington, DC, May 2002, pp.657-663

[23] Bardorfer, A. and Munih, M., Connecting Haptic Interface with a Robot, Melecon 2000- 10th Mediteranean Electrotechnical Conference, May. 2000, Cyprus, pp.1-4

[24] Kim, W.S., Developments of a New Force Reflecting Control Schemes and an Application to a Teleoperation Training Simulator, Proc. of the 1992 IEEE Int. Conf. On Robotics and Automation, Nice. France, May 1991.pp1412-19.

[25] Adams, R.J. and Hannaford, B., Stable Haptic Interaction With Virtual

Environments, IEEE Transactions on Robotics and Automation, Vol.15, June 1999, pp.465-474.

[26] Adams, R.J., Moreyra, M.R. and Hannaford, B., Stability and performance of haptic displays: Theory and experiments, in Proc. ASME Dynamics Systems and Control Division DSC-Vol.64, 1998, pp.227-234.

[27] Adams, R.J. and Hannaford, B., Control law design for haptic interfaces to virtual reality, IEEE Trans. Contr. Syst. Technol., Vol.10, No.1, pp.3-13, 2002.

[28] Buttolo, P., Kung, D. and Hannaford, B., Manipulation in Real, Virtual and Remote Environments, Proc. of the IEEE Conf. On System, Man and Cybernetics, Vol. 5, Vancouver, BC, October 1995. pp.4456-61.

[29] Grega, W., Hardware-in-the-loop simulation and its application in control education, 29th ASEE/IEEE Frontiers in Education Confference,1999, pp.7-12.

[30] Necsulescu, D.S., Artificial Impedance Approach of The Trajectory Generation And Collision Avoidance for Single and Dual Arm Robots, International Workshop in adaptive and Nonlinear Control: Issues in Robotics, Grenoble (France) November 21-23 1990.

[31] Necsulescu, D.S. and Jassemi-Zargani, R.J., Impedance control for robotic manipulation, Second workshop on military robitic applications, RMC, Kingston, Ontario August 1989, pp.2-10.

[32] Megatorque Motor System, user's Manual, Motion and Control Technology NSK NIPPON SEIKO K.K., Japan, 1989.

[33] DSP-CITpro, Hardware Manual, v.1. dSPACE digital signal processing and control engineering GmbH, West Germany, 1990.

[34] DSP-CITpro, Trace Manual, v.1. dSPACE digital signal processing and control engineering GmbH, West Germany, 1990.

[35] Matlab/Simulink, The Mathworks Inc., www.mathwork.com.

[36] CEA-series strain gages engineering data sheet, Measurements group, INC 2004.

[37] Pelletier, Michel and Doyon, Michel, On the Implementation and Performance of Impedance control on Position Controlled Robots, IEEE International Conference, Vol.2, 1994, pp.1228-1233.

[38] Lawrence, D.A., Impedance Control Stability Properties in Common Implementations, IEEE International Conference, Vol.2, 1988, pp.1185-1190.

[39] Necsulescu, D.S. and Basic, G., Analysis of Implementation Issues for Haptic Interfaces using HIL Concept, HAVE 2002, IEEE International Workshop on Haptic Virtual Environments and their Applications, Ottawa, Canada, November 17-18, 2002.

[40] Khatib, Oussama, A Unified Approach for Motion and Force Control of Robot manipulators: The Operational Space Formulation, IEEE Journal Of Robotics and Automation, Vol. RA-3, No.1, February 1987, pp.43-53.

[41] Kim, B., Free Motion and Obstacle Avoidance Control for Mobile Robots, Master thesis, Department of Mechanical Engineering University of Ottawa, 1992.

[42] Jassemi-Zargani, Rahim, Impedance Control of a Dual-Arm Robot, PhD Thesis, Department of Mechanical Engineering University of Ottawa, 1998.

[43] Lewis, F.L., Jagannathan, S. and Yesildirek, A., Neural Network Control of Robot Manipulators and Nonlinear Systems, Taylor & Francis, 1999.

[44] Lewis, F. L., Abdallah, C. T. and Dawson, D. M., Control of Robot Manipulators,

Macmillan, 1993.

[45] Murray, R.M., Li, Zexiang and Sastry, S. S., A Mathematica Introduction to Robotic Manipulation, CRC Press, 1994.

[46] Crandall, S.H., Karnopp, D.C., Kurtz, E.F. and Pridmore-Brown, D.C., Dynamics oF Mechanical and Electro-Mechanical Systems, McGraw-Hill, 1968.

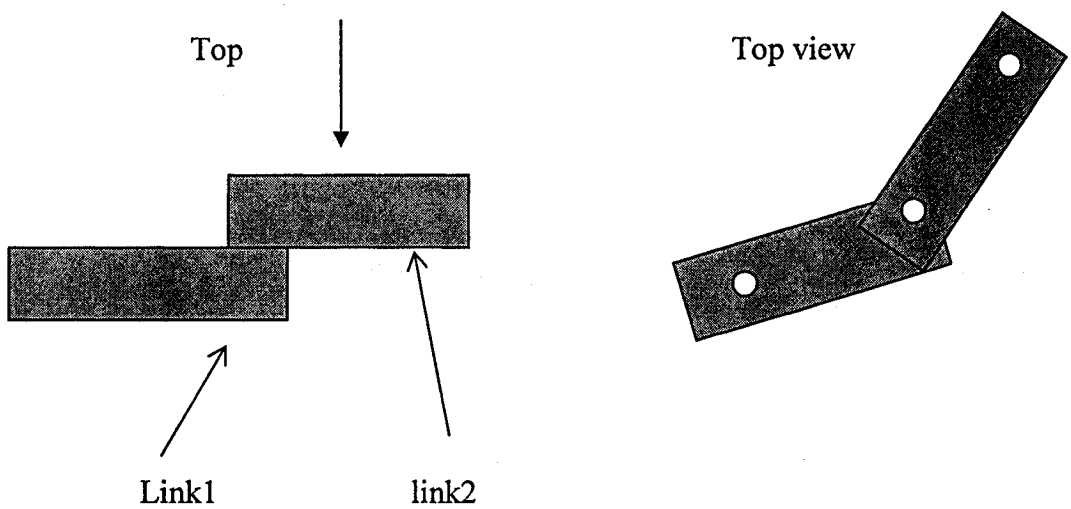
[47] Vukobratović, Miomir and Stokić, Dragan, Applied Control of Manipulation Robots, Berlin, New York: Springer-Verlag, 1989.

[48] Spong, M.W. and Vidyasagar, M., Robot Dynamics and Control, John Wiley &Sons, 1989.

APPENDIX A

Robot arm link parameters

Robot arm link parameters



L_1 =0.2207m

L_2 =0.1603m

M_1 =12.88kg

M_2 =2.5kg

Material: Aluminum

μ =2700kg/m^3

f_1 =270HZ

f_2 =876HZ

f_1 and f_2 are very high, which means they can be considered as rigid bodies.

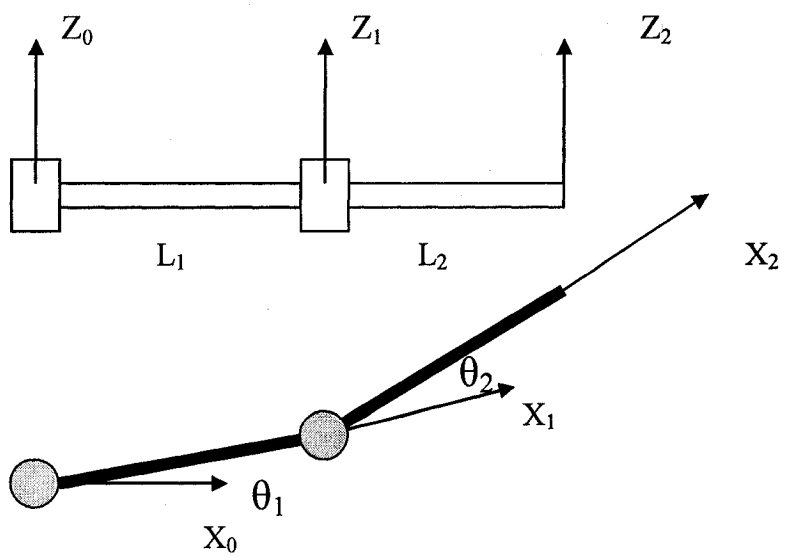
APPENDIX B

Derivation for Kinematic equations

1.0 Forward kinematics of Robot arm

Denavit-Hartenberg convention for kinematic derivation:

#	θ	d	a	α
1	θ_1	0	L_1	0
2	θ_2	0	L_2	0



$$A1=\begin{vmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & L_1\cos\theta_1 \\ \sin\theta_1 & \cos\theta_1 & 0 & L_1\sin\theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$A_2=\begin{vmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & L_2\cos\theta_2 \\ \sin\theta_2 & \cos\theta_2 & 0 & L_2\sin\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$^{R}T_H=A_1*A_2=\begin{vmatrix} \cos(\theta_1+\theta_2) & -\sin(\theta_1+\theta_2) & 0 & L_2\cos(\theta_1+\theta_2)+L_1\cos\theta_1 \\ \sin(\theta_1+\theta_2) & \cos(\theta_1+\theta_2) & 0 & L_2\sin(\theta_1+\theta_2)+L_1\sin\theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

So, position:

$$X=L_1*\cos\theta_1 + L_2*\cos(\theta_1+\theta_2)$$

$$Y= L_1*\sin\theta_1 + L_2*\sin(\theta_1+\theta_2)$$

2.0 Jacobian

$$\begin{vmatrix} d_x \\ d_y \end{vmatrix} = \begin{vmatrix} J \end{vmatrix} \begin{vmatrix} d\theta_1 \\ d\theta_2 \end{vmatrix}$$

$dx=\partial x/\partial\theta_1*d\theta_1+\partial x/\partial\theta_2*d\theta_2$

$\partial x/\partial\theta_1= -L_1*\sin\theta_1 - L_2*\sin(\theta_1+\theta_2)$

$\partial x/\partial\theta_2= -L_2*\sin(\theta_1+\theta_2)$

$dy=\partial y/\partial\theta_1*d\theta_1+\partial y/\partial\theta_2*d\theta_2$

$\partial y/\partial\theta_1= L_1*\cos\theta_1 + L_2*\cos(\theta_1+\theta_2)$

$\partial y/\partial\theta_2= L_2*\cos(\theta_1+\theta_2)$

$$J = \begin{vmatrix} \frac{\partial x}{\partial \theta_1} & \frac{\partial x}{\partial \theta_2} \\ \frac{\partial y}{\partial \theta_1} & \frac{\partial y}{\partial \theta_2} \end{vmatrix}$$

$$J = \begin{vmatrix} -L_1 \sin\theta_1 - L_2 \sin(\theta_1 + \theta_2) & -L_2 \sin(\theta_1 + \theta_2) \\ L_1 \cos\theta_1 + L_2 \cos(\theta_1 + \theta_2) & L_2 \cos(\theta_1 + \theta_2) \end{vmatrix}$$

Because total virtual work at the joints must be the same as the total work at the hand frame, we get:

$$\begin{vmatrix} F_x & F_y \end{vmatrix} * \begin{vmatrix} dx \\ dy \end{vmatrix} = \begin{vmatrix} \tau_1 & \tau_2 \end{vmatrix} \begin{vmatrix} d\theta_1 \\ d\theta_2 \end{vmatrix}$$

$$\begin{vmatrix} dx \\ dy \end{vmatrix} = \begin{vmatrix} & J & \end{vmatrix} \begin{vmatrix} d\theta_1 \\ d\theta_2 \end{vmatrix}$$

So, $$\begin{vmatrix} F_x & F_y \end{vmatrix} * \begin{vmatrix} & J & \end{vmatrix} \begin{vmatrix} d\theta_1 \\ d\theta_2 \end{vmatrix} = \begin{vmatrix} \tau_1 & \tau_2 \end{vmatrix} \begin{vmatrix} d\theta_1 \\ d\theta_2 \end{vmatrix}$$

$F^T * J = \tau^T$

$\tau = J^T * F$

APPENDIX C

Dynamic model derivation

Lagrangian mechanics will be used for solving dynamics of this robot.

For rotational motion: $T_i = \frac{\partial}{\partial t}(\frac{\partial L}{\partial \dot{\theta}_i}) - \frac{\partial L}{\partial \dot{\theta}_i})$

Here, L is the Lagrangian and is defined as: L=K-P

K is the kinetic energy of the system, and P is the potential energy of the system.

X_i and θ_i denote the variables of linear and rotate joints respectively.

For given robot arm configuration, K is derived as:

$$K = \dot{\theta}_1^{\,2}(\frac{1}{6}m_1L_1^{\,2} + \frac{1}{6}m_2L_2^{\,2} + \frac{1}{2}m_2L_1^{\,2} + \frac{1}{2}m_2L_1L_2\cos\theta_2) + \dot{\theta}_2^{\,2}\frac{1}{6}m_2L_2^{\,2} +$$

$$\dot{\theta}_1\dot{\theta}_2(\frac{1}{3}m_2L_1^{\,2} + \frac{1}{2}m_2L_1L_2\cos\theta_2)$$

$$P = 0$$

$$L = K - P = K$$

$$T_1 = \frac{\partial}{\partial t}(\frac{\partial L}{\partial \dot{\theta}_1}) - \frac{\partial L}{\partial \theta_1}$$

$$= (\frac{1}{3}m_1L_1^{\,2} + \frac{1}{3}m_2L_2^{\,2} + m_2L_1^{\,2} + m_2L_1L_2\cos\theta_2)\ddot{\theta}_1 + (\frac{1}{3}m_2L_2^{\,2} + \frac{1}{2}m_2L_1L_2\cos\theta_2)\ddot{\theta}_2 -$$

$$m_2L_1L_2\sin\theta_2\dot{\theta}_1\dot{\theta}_2 - \frac{1}{2}m_2L_1L_2\sin\theta_2\dot{\theta}_2^{\,2}$$

$$T_2 = \frac{\partial}{\partial t}(\frac{\partial L}{\partial \dot{\theta}_2}) - \frac{\partial L}{\partial \theta_2}$$

$$= (\frac{1}{3}m_2L_1^{\,2} + \frac{1}{2}m_2L_1L_2\cos\theta_2)\ddot{\theta}_1 + \frac{1}{3}m_2L_2^{\,2}\ddot{\theta}_2 + \frac{1}{2}m_2L_1L_2\sin\theta_2\dot{\theta}_1^{\,2}$$

So,

Inertial Matrix I:

$$I(\theta) = \begin{vmatrix} \frac{1}{3}m_1L_1^2 + \frac{1}{3}m_2L_2^2 + m_2L_1^2 + m_2L_1L_2\cos\theta_2 & \frac{1}{3}m_2L_2^2 + \frac{1}{2}m_2L_1L_2\cos\theta_2 \\ \frac{1}{3}m_2L_2^2 + \frac{1}{2}m_2L_1L_2\cos\theta & \frac{1}{3}m_2L_2^2 \end{vmatrix}$$

Centrifugal and Coriolis Matrix:

$$N = \begin{vmatrix} -\frac{1}{2}m_2L_1L_2\sin\theta_2\,\dot{\theta}_2^{\,2} - m_2L_1L_2\sin\theta_2\,\dot{\theta}_1\dot{\theta}_2 \\ \frac{1}{2}m_2L_1L_2\sin\theta_2\,\dot{\theta}_1^{\,2} \end{vmatrix}$$

Appendix D

Procedures for using the experimental setup

1) Set the working mode of the direct drive motors as torque mode every time when driver unit is turned off according to the procedure addressed in [45].
2) Connect the batteries of the signal conditioning circuit for force sensor. The system was calibrated for 9v battery. If batteries get exhausted, replacement is necessary.
3) Check the reading from the force sensor, when no force applied the reading should be zero. Small drift can be corrected by program. If large variance is observed check the circuit.
4) If any change is done to user-developed program or any new program is created, compile it before start to run software MON30.
5) Run MON30 software and load the compiled program to DSP in dSPACE.
6) Clear the D/A, incremental encoder channels before starting run the program in DSP.
7) If any variables need to be traced, quit the screen without stop the DSP.
8) Start TRACE30 to trace signals.
9) When tracing is done, save traced data to disk or print traced chart in disk, or both.
10) To stop the running of the DSP, type MON30 again, then a command line will be prompted to user asking if need to stop, type Y to stop the running of DSP.