FORCE CONTROL FOR ROBOTIC KNEE SURGERY

by

Shi-Chi HO

A thesis submitted for the degree of Doctor of Philosophy of the Faculty of Engineering, University of London, and for the Diploma of Imperial College

June 1995

Department of Mechanical Engineering
Imperial College of Science, Technology and Medicine
University of London

ABSTRACT

The use of robots to assist in knee surgery has the potential to improve the long term outcome of prosthetic implants. The robot can make accurate cuts of complex shapes that not only improve the fit of the prosthesis, but also allow for improved design. The use of pre- programmed position control of the robot can make the cuts with good results. However, there can be a sense of lack of control by the surgeon, who loses close visual and tactile feedback. In addition, the psychological needs of both patients and surgeons must be taken into account when using an autonomous robotic manipulator. The thesis describes a new approach for the use of force control whereby the surgeon can hold a cutter on the end of the robot and move it. The surgeon can supervise and control the robot directly, and execute the cutting using his/her innate sensing, experience and judgement. The strategies of implicit force control and modified damping control, with active motion constraint, can assist the surgeon in executing the pre-planned cuts accurately and efficiently. The thesis presents the analysis of the control strategies and gives results of an experimental evaluation of a three degree-of-freedom manipulator. From preliminary experiments with animal bones, the concepts of artificial motion constraint and the controller design strategy have been shown to be feasible.

To My Parents

Acknowledgements

I wish to express my sincere gratitude to my supervisors, Dr. B.L. Davies and Dr. R.D. Hibberd, for their constant guidance and excellent advice throughout the course of this work.

Grateful acknowledgement is also made to the Ministry of Education of Taiwan and Committee of Vice-Chancellors and Principals of the Universities of the United Kingdom for their scholarship awards.

Finally, I wish to thank all the members of my family and my girlfriend, Chin-Fong Tseng, for their continual encouragement and support.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

This chapter starts with an introduction to the use of robots. The initial applications of robots in the industry and then a progressive diversification of applications of robots, which involve servicing the public at large, are described. In particular, recent applications of robots in medicine are highlighted. The development of surgery robots is examined. The levels of complexity of systems in surgery are categorized. Finally, a general methodology for the use of the surgery robot will be introduced.

1.1 The Use of Robots

The word "robot" is a derivative of Czech 'robota', which means worker. It first appeared in Karel Capek's play, Rossum's Universal Robots, in New York on October 9, 1922. However, it was not until 1965, more than thirty year after Capek's play, that a firm named Unimation was formed and made its sole business of robotics. Initially, the concept of the robot was that it would relieve the tedium of human work or even replace human labour. The public in general came to think of robots as human-like machines that had five senses and had the ability to make judgements as well as having dexterity. In fact, a more descriptive term for most industrial robots would be "mechanical arm". Judging from the use of robots in the past few decades, however, robots in reality come in between labour and hard automation.

Hard automation is the use of traditional machinery, which is usually custom-made and designed to facilitate the manufacture of a specific product. The machinery can achieve very high-speed production, high-volume production, but is usually quite expensive. This expense can become very large when a product model or design

change is introduced. Unlike hard automation, robots are flexible and programmable, which can be applied in different tasks and different environments with less customisation. They are thus better suited to small volume, high variety production. As defined by the Robot Institute of America, a robot is: "a re-programmable multi-funtional manipulator designed to move material, parts, tools, or specialized devices through variable programmed motions for the performance of a variety of tasks".

The first robotized process was implemented in the casting process in 1961 [Engelberger 74/80]. In the 1960's and 1970's, many robots of different designs emerged and were utilized widely in industrial applications typical of the factory floor, ranging from the foundry to the assembly line. Some of the prominent makes of the robots are the Unimation's PUMA series, ASEA's IRB series, Cincinnati's T^3 series, and KUKA's IR series robots etc.. Almost all of these robots fall into one the four basic geometries: Cartesian, cylindrical, spherical and revolute configurations [Asfahl 85, Cugy 84, Hartley 83]. In an industrial scene, the kinds of tasks that robots are usually applied in are:

- Loading and unloading.
- Tracking a path (e.g. in welding and spray painting).
- Selection and Inspection.
- Assembly.

As technologies advance, the applications of robots become more diversified. A general list of the areas of employment for robots today is shown in Fig. 1.1 [Engelberger 89, Dorf 88]. One trend in the use of robots is that they may come into close contact with human users. If a robot is assigned to some service task, the first expectation will be that the robot should emulate a human in getting the task done. Thus, robotic technologies will need advances in design of robot anatomies, sensory perception, artificial intelligence, expert systems and in safety systems. An initial concept of Advanced Robotics (AR) was introduced at the OECD summit in Versailles in 1982. Within the United Kingdom, a definition of AR has been agreed [Finlay 89/b]:

"The integration of enabling technologies and attributes embracing manipulator, mobility, sensors, computing (knowledge based system and artificial intelligence) and hierarchical control of autonomously complementing man's endeavours in unstructured and hostile environments."

Among these new areas of robot applications, service robots have the greatest growth potential and will be more important than manufacturing robots in the future. For instance, seventy-five percent of the American workforce is in the service sector; only twenty-five percent is involved in manufacturing. A research and development company named Transitions Research Corp. (TRC) was formed by Engelberger in 1984, and its major business is to design and build robots for service organisations. Today, TRC is a manufacturer of service robots for applications in hospitals, pharmacies and supermarkets [Pellerin].

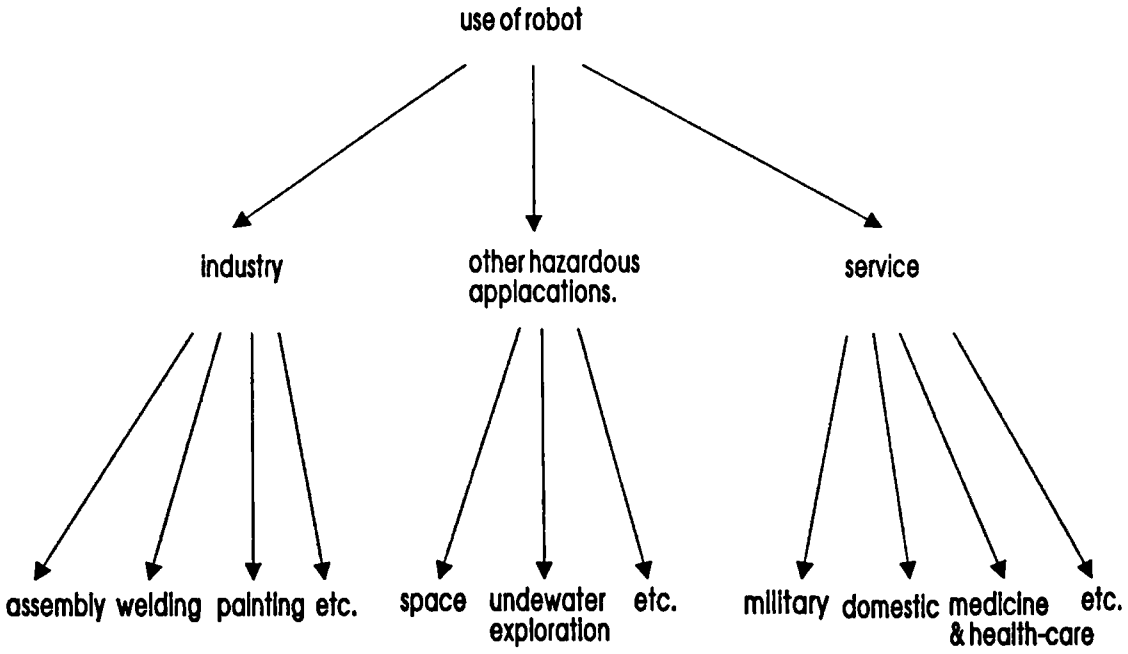


Fig. 1.1 Use of robot [after Engelberger 89]

Among the various applications of service robots, the development of medical robots has attracted much attention recently. In the UK, a feasibility study of medical robots, sponsored by the Department of Trade and Industry (DTI), was conducted by Fulmer Systems Ltd. in 1988 [Finlay 89/b]. The study indicated significant opportunities for

advanced medical robots. Some of the main reasons are (1) the age of the population is steadily increasing in most countries, while the number of young workers is decreasing. Thus, the social pressures of taking care of ageing people is becoming increasingly serious. (2) improved technologies make applying robots in health-care possible and economically justifiable. (3) both medical staff and patients have expected (and accepted) advanced automation in health-care. In general, the use of medical robots can be divided into three main categories [Finlay 89/a, Korba, Preising 91, Takatoshi 91, Takeyoshi 91]:

(1). Hospital service functions: some examples are transportation of patients and medical supplies within hospital. A project, Patient Aid to Mobility (PAM), has been developed by Dr. Kerr in the University of Salford. The project is designed to lift patients from and to beds, and to transfer patients within hospital allowing change in posture from lying to sitting when necessary [Kerr 90]. A mobile robotic materials transport system called HelpMate has been manufactured by TRC. HelpMates can perform material transport duties and have been installed in a dozen hospitals across the USA [Pellerin].

Another area is the use of robotic systems in the laboratory, such as testing of blood samples and urine analysis [Severns 84, Owens 82]. However, laboratory robots could be considered as an extension of the use of industrial robots and might not be seen as a division of medical robots.

(2). Rehabilitation functions: the main objective is to help the disabled to restore their mobility which have been lost totally or who are so weak that they are ineffective for performing tasks. The primary rehabilitation division is between workstation robots and mobile robots [Davies 84, Leifer 81, Harwin 86, Hillman 87, Jackson 87, Prior 93]. Workstation robots usually consist of a simple arm which is mounted on a table so that it can access most of the area. The robot is usually referenced to a fixed location and then moves to other fixed points to perform simple tasks such as raising food to the mouth or for inserting computer disc cartridges. The objective is to let the disabled be able to live unaided for periods ranging from one to eight hours. The

sequence of motions can often be programmed. As such, the task of driving the robot becomes much simpler and the input only consists of selecting the appropriate programme.

Mobile robots, on the other hand, are usually configured to operate in an unstructured environment. The robot can either be mounted on a free ranging mobile platform or mounted on the side of a powered wheelchair. Inputs to the robot vary from hand operated joysticks for the most able, to head operated switches for the most disabled. Because the mobile robot system has multiple inputs to control many devices, aspects of the human/computer interface have to be simplified to make the system readily controllable [Van Woerden 94].

(3). Clinical functions: this includes diagnosis, therapy and surgery.

Of the many applications of medical robots listed above, this thesis has concentrated on the feasibility of the use of force control for robotic knee surgery. Levels of complexity of systems in robotic surgery and the general methodology for a surgery robot will be described in the next section.

1.2 Robots in Surgery

The use of surgery robots offers the greatest scope for the medical use of robots. Surgical robots are used because they can provide precise and repeated motions in response to pre-programmed tasks. Usually some form of imaging system, such as Computed Tomography (CT), Magnetic Resonance Imaging (MRI), and Positron Emission Tomography (PET) etc., is used pre-operatively to define the surgical procedures. The robot is then registered with reference to the patient so that the sequence of motions can be automatically carried out.

Research into robotic surgery is a relatively new area which has been around for less than 10 years. One of the earliest robotics activities in surgery was carried out by Dr. Kwoh and colleagues at the Memorial Medical Centre, Long Beach, California. A CT guided stereotatic brain surgical procedure assisted by a PUMA 200 robot has been

developed. Stereotactic neurosurgery is a technique for guiding a probe or some delicate instruments into the brain through a small hole drilled in the skull without direct vision of the surgical site. The insertion of the probe is usually along a straight guide that is fixed to a stereotactic frame which attaches to the patient's skull. To provide vision for the surgical site, a CT-guided stereotactic head frame was developed [Kwoh 85]. The patient's skull is fixed to the frame, and three N-shaped locators are used to register the CT image with respect to the frame coordinate system. This allows any target in the image to be found in three dimensions with respect to the frame coordinate. The robot is equipped with a probe guide as its end effector and its spatial relation with the stereotactic frame is fixed. The robot can be then programmed to reach a given target point by the CT diagnosis, and submillimeter accuracy could be attained [Kwoh 88]. With slight modifications, such a stereotactic frame integrated with a robot arm can be applied for surgical procedures implemented on other parts of the body, such as the biopsy of the kidney, abscess drainage, and electrolytic ablation [Drake 91, Lavallee 89, Shao 85].

In the above cases, however, the robot was only used to hold a fixture at the appropriate position and orientation so that the surgeon can manually insert the biopsy needle into the patient. It was not until April 1991, when a special purpose robot was applied clinically for removal of prostate by the group at Imperial College, that the first truly robotic operation was implemented by using an robot to actively insert the cutting device into the patient and automatically remove quantities of tissue [Davies 89, Ng 92/93].

In general, the robot systems available may be passive, semi-active or fully active. One way of giving an overview of the use of robots in surgery is to describe the levels of complexity of systems in surgery, which includes Computer Aided Systems (CAS) and Robotic Surgery. The classification of surgery systems will be described in section 1.2.1 and the general methodology for a surgery robot will be discussed in section 1.2.2.

1.2.1 Levels of complexity of systems in surgery

It is difficult to categorize the surgery robotic system precisely. However, based on the tools used, a general level of complexity of surgery systems may be classified as in Fig. 1.3 [Davies 94/d]. Basically, these systems can be divided into CAS and Robotic Surgery depending on whether or not a powered robot is used. Basically, the CAS is a positioning system and includes three steps: pre-operative planning, datuming and then tracking motions of both tools and patients. Transmitter/Receiver remote sensors or passive linked manipulators are usually used to track the tools in the CAS.

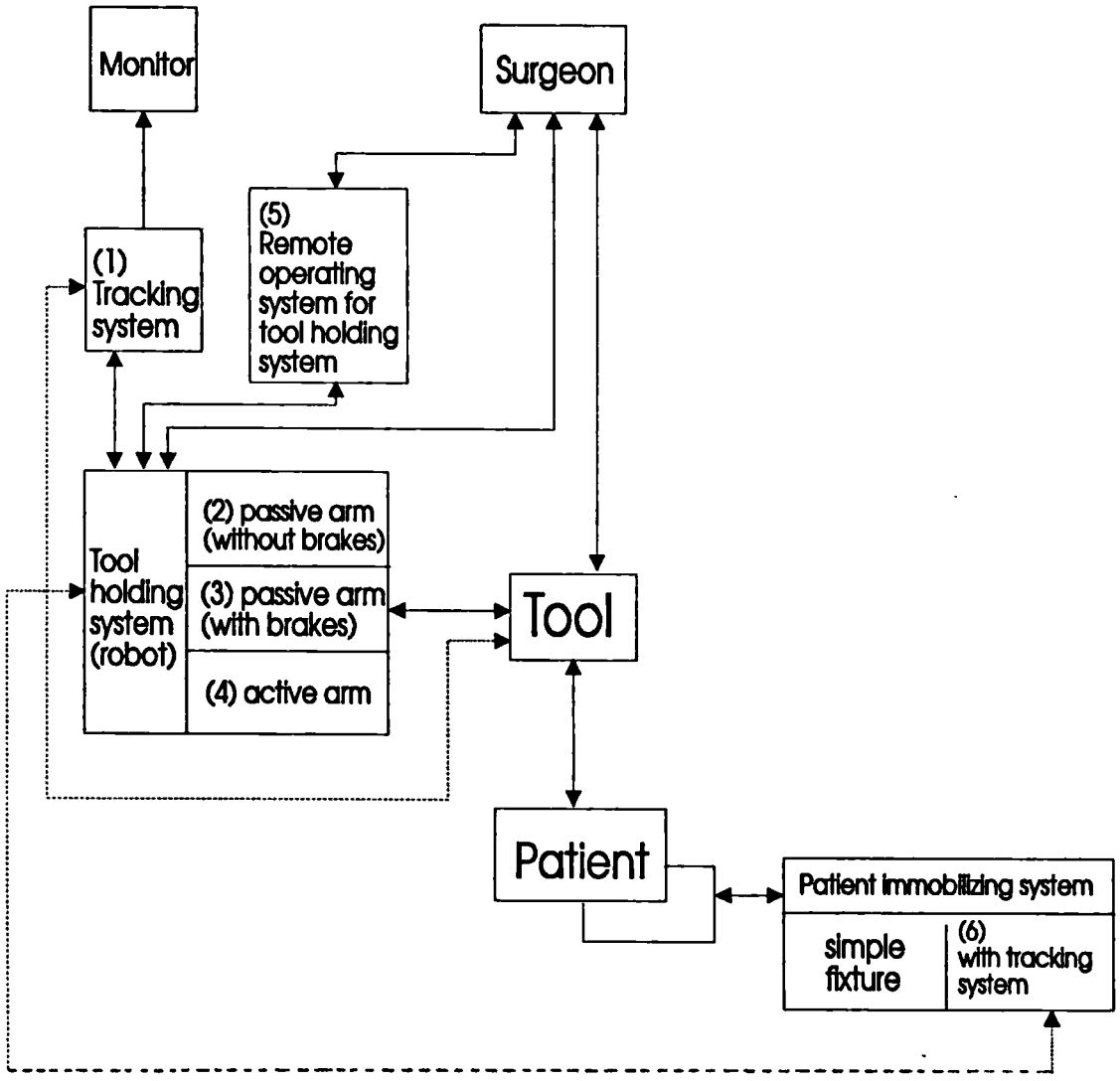


Fig. 1.3 Levels of complexity of surgical systems [after Davies 94/d]

	equipment used						description
Level	1	2	3	4	5	6	
I							hand-held tools with human innate sensing.
II	X					O	hand-held tools with location tracking system, such as cameras plus LEDS.
III	X	X				O	tools mounted to a passive arm. The robot position is sensed by joint position sensor or an independent tracking system.
IV	X		X			O	similar to III but the passive arm has powered brakes, so the tools can be locked in desired positions.
V	X			X		O	tools mounted to a powered arm, but the robot is used semi-actively. For instance, the final action is completed by the surgeon when the robot moves to the pre-defined position, or the robot is moved by the surgeon by hand under the position and force control strategies [Ho 94, 95/a/b].
VI	X			X		O	similar to V, but the robot is used to autonomously finish some subtask, such as cutting.
VII	X			X	X	O	similar to VI, but a remote operating system, such as a master-slave robot system.
note: O (optional) used to adapt the patient's movement intra-operatively.							

Fig. 1.3 continued

The difference between the CAS and Robotic Surgery is simply that the joint motions of a surgery robot are directly driven and controlled by a computer program instead of being positioned by the surgeon in CAS. The CAS has less risk of damage as a result of spontaneous undesired motions of the robotic system. However, it is not good at reaching of a pre-defined position or following a desired trajectory, nor does it prevent undesired twitches or tremors made by the surgeon. Especially, moving a tool by

using a mechanical passive arm is very difficult because the mechanism naturally promotes some directions of motion rather than others. The more complex the system, the more abilities the robot will have to assist the surgeon in implementing the operation. However, the safety problems will be more demanding when the robot system becomes more complicated [Davies 93/a,b,c]. It should be noticed that no matter what kind of robotic system is used, the robot is just an assistant tool which enables the surgeon to execute the operation better, and will not replace the function of the surgeon.

1.2.2 General Methodology for Surgery Robot

The surgical procedures for the CAS and Surgical Robot, in general, can be divided into three stages: [Cinquin 92, Cutting 92, Dohi 93, Lavallee 91]

(1) Pre-operative planning : in order to design the surgical strategy, a qualitative and quantitative 3D computer model is generated by the data from an imaging system, such as CT and MRI etc. The choice of the acquisition modality depends on which kind of tissue the surgeon is mainly interested in, or the type of applications to which they are applied. For instance, the CT scanner involves the use of X-rays; therefore, it is not desirable for use in regions near the reproductive organs because of possible effects on potency. Another example is that the MRI method is not suitable for patients who have metallic parts in their bodies, because the metallic parts accelerate in the magnetic field and cause injury.

Based on this 3D model, the surgeon then decides the surgical procedures via user interface software. These surgical procedures will be modified and completed by intra-operative data. The simulation of the intervention can also be carried out, thus the surgeon is able to pre-operatively examine the consequences of the operation.

(2) Registration : during the pre-operative planning stage, a 3D model for the patient is derived and the surgical procedures based on the model are defined. In order to

help the surgeon execute the planned procedures precisely, different assistant systems (see fig. 1.3) may be used. This implies that all the coordinates of involved systems have to be matched or registered intra-operatively. Fig. 1.4 shows the essential coordinates that are to be matched.

Usually, there are two ways of linking the image model and the patient: artificial markers and anatomical features. The artificial marker method was proposed in many projects because of its convenience in use [Taylor 89/92, Cinquin 92]. However, the artificial markers, when used, have to be put on the patient before taking the image. They are inherently invasive and subject to possible movement with respect to the anatomic objects. Registering the coordinates of the patient also implies that the patient has to be firmly fixed to the operating table, or the movement of the patient is adapted intra-operatively to the robot system. The tracking systems (spatial location system) may be a linked manipulator, magnetic systems, optical systems, or ultrasonic systems etc. [Adams 89/92].

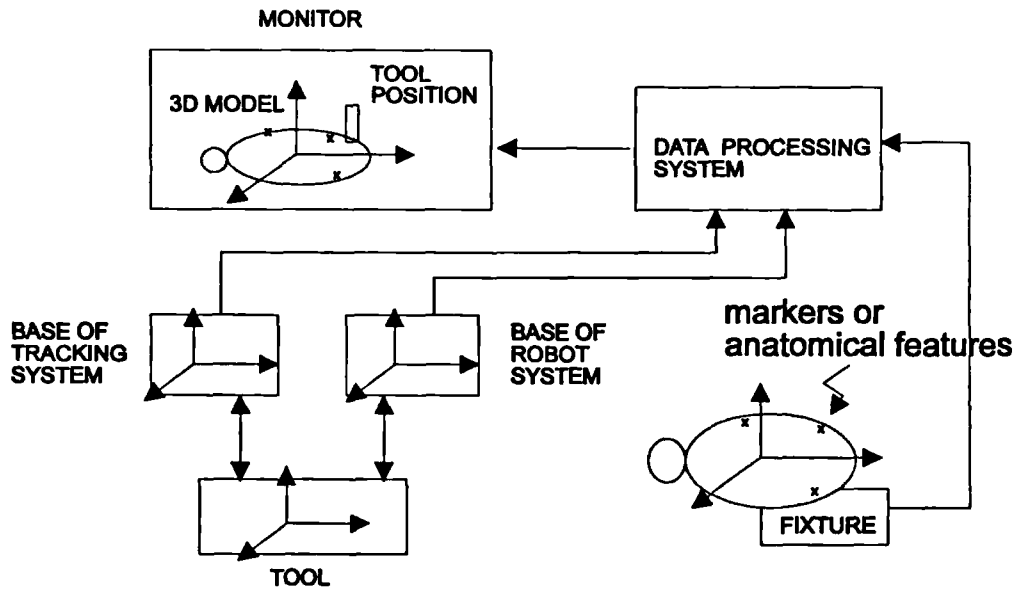


Fig. 1.4 Coordinates to be matched

(3) Execution : in a Computer Aided System the tool is connected to a tracking system (including passive arm), so the surgeon can continuously check the tool position

relative to the surgical procedures. When a powered robot is used, it can be in either semi-active or active mode, depending on the autonomy left to the surgeon.

I. Semi-active mode: the robot may hold a tool or tool-guide and move to a predefined point under position control where it is locked in position and the power removed. The final inserting action is then completed by the surgeon. Another method is that the robot holds the tool and is moved by the surgeon by hand whilst the robot is under a combination of position and force control. In this mode, the surgeon can examine the procedure and performance closely and have more direct control during the operation.

II. Active mode : the robot executes some sub-tasks autonomously under the surgeon's supervision [Taylor 92]. Trajectories of the robot can be accurately controlled. Repeated and incremental motions can also be performed without difficulty. The further potential advantage is that, once the pre-operative planning, clamping and registering activities are finished, the actual motion sequence can be performed rapidly, leading to a reduced time for the actual surgery. However, in addition to the safety problem, this kind of active system will need more psychological acceptance by the patients and surgeons.

The procedure for robotic hip replacement surgery conducted by Taylor and colleagues is given below as an example [Taylor 92]. Pre-operatively, titanium locating pins, as artificial markers, are inserted into the patient's greater trocanter and femoral condyles. A CT scan is made of the leg and a 3D model is created. An appropriate orthopaedic implant is selected, and where it is to be placed relative to the patient's femur is determined. In the theatre, the surgeon operates as normally until the femoral head is removed. At this point, the femur and robot are then fixed and referenced to the operating table. The 3D model is then input into the robot controller and the model, robot and the femur are all registered to each other. The robot controller then computes the appropriate transformation between CT and robot coordinates. It is only at this point that the robot carries out the motion sequence, during which a six degree-of-freedom force sensor is used to support redundant safety

checking. Following the procedure, the pins are removed and the surgery proceeds in a normal manner.

1.3 Conclusion

Starting from the 1980's, the use of robots has gradually branched away from the bounds of industrial tasks. Among these new areas of robot applications, service robots are believed to have the greatest growth potential and will be more important than manufacturing robots in the future. Of the various applications of service robots, the development of medical robots has attracted much attention recently. The need of robots in the medical field has been discussed in section 1.1. In general, the use of medical robots can be divided into three main categories: hospital service, rehabilitation and clinical functions. Examples and more details can also be found in section 1.1.

The development of surgery robots has been examined. Levels of complexity of systems in surgery has also been categorized. Basically, these systems can be divided into Computer Aided System (CAS) and Robotic Surgery depending on whether or not a powered robot is used. They are detailed in section 1.2.1.

The methodology for the use of the surgery robot has been introduced in section 1.2.2. The surgical procedures for the CAS and Surgical Robot, in general, can be divided into three stages: pre-operative planning, registration and execution. The procedure for robotic hip replacement surgery conducted by Taylor and colleagues has been given above as an example.

In the next chapter, the possibility of developing computer and robot assisted knee surgery will be investigated. After examining the conventional surgical procedures and reviewing relevant projects to robotic knee surgery, the motivation and objectives of this thesis will be discussed.

CHAPTER 2

MOTIVATION AND OBJECTIVES

This chapter begins with an introduction to knee surgery. Requirements of knee joint replacement are briefly described. Secondly, the conventional surgical procedure, taking Total Knee Replacement (TKR) as an example, is discussed together with its drawbacks. The possibility of developing computer and robot assisted systems in knee surgery is then investigated and relevant projects are analyzed. Afterwards, the objectives of the proposal to develop force control strategies and artificial motion constraints in assisting the execution of planned resection are elucidated.

2.1. Introduction to Knee Surgery

The knee joint is one of the most loaded and stressed joints in the human body, in which there is free motion in one major plane with significant stability. The function of the knee joint is like a hinge-joint. Stability of the knee joint is maintained by a number of special mechanisms, such as twofold or threefold expansions of bearing surfaces of the femur and the tibia, reinforcing tendons, intra-articular ligaments and collateral ligaments (Fig. 2.1 & 2.2) [Woodburne 88]. Occasionally, the knee joint suffers from arthritis caused by disease or injury, and medical treatment is needed. Many patients with pain and disability from arthritic knees may possibly be cured by physical therapy, mechanical support and analgesics. However, if these methods are not effective or adequate, surgical intervention becomes necessary. The main goals of knee surgery are to relieve pain, restore functional mobility and stability, and correct deformity.

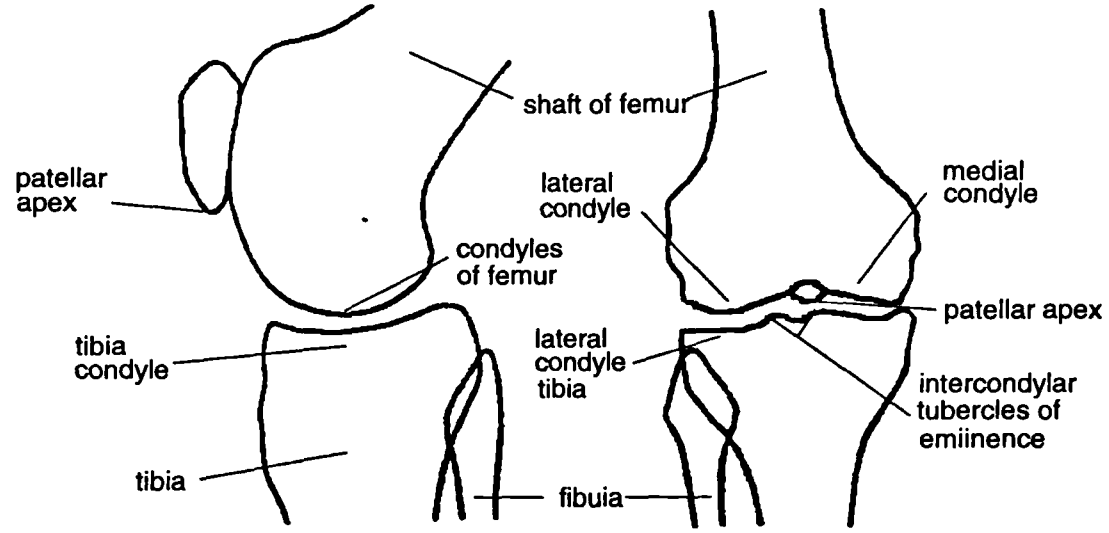


Fig. 2.1 The knee joint; posteroanterior and lateral views

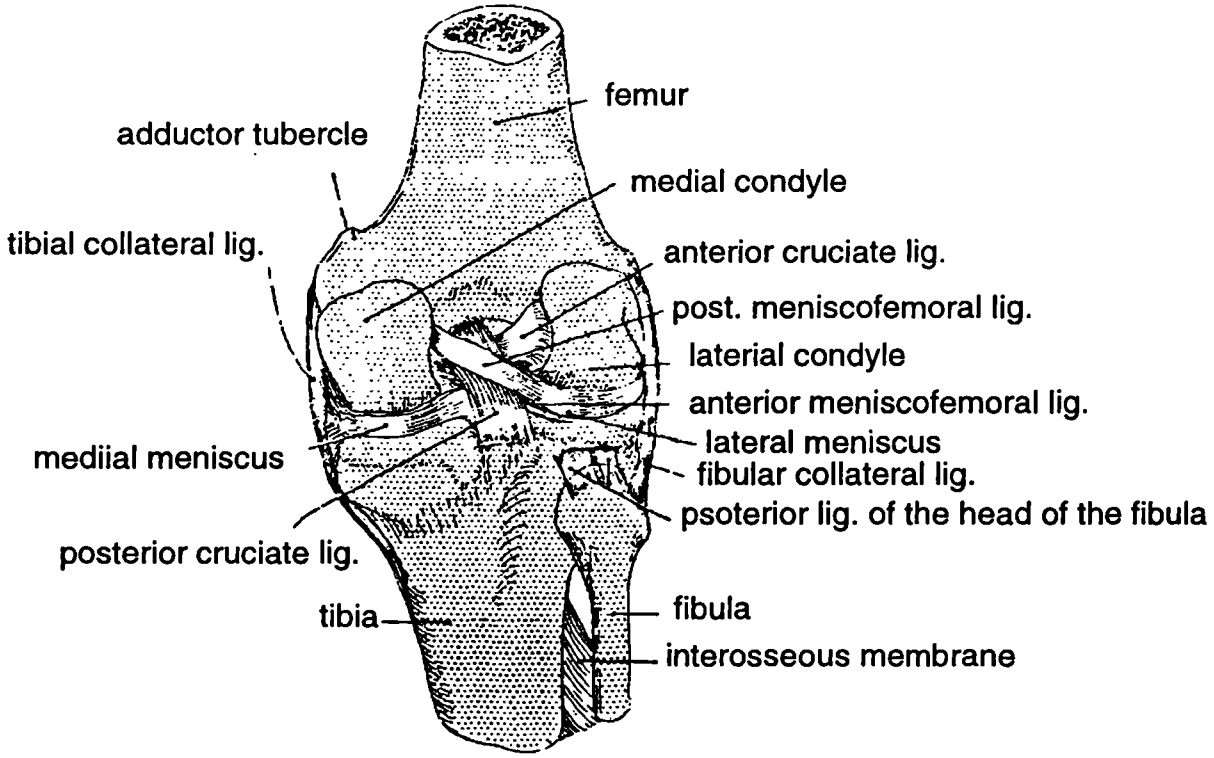


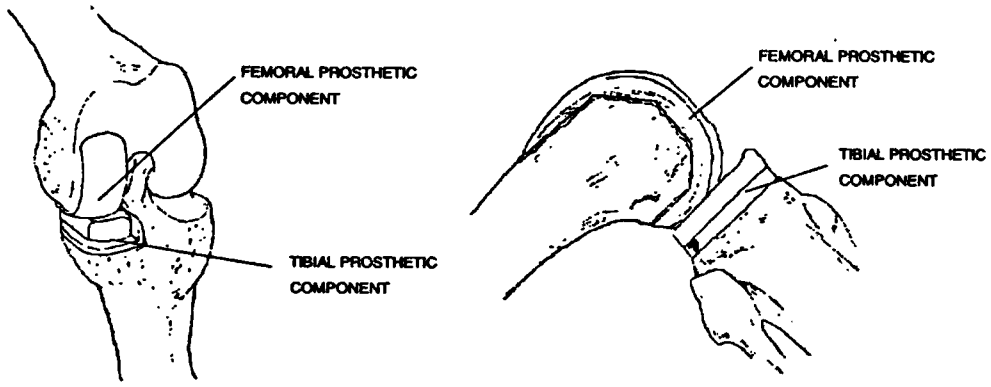
Fig. 2.2 The right knee joint opened from behind; the menisci and the cruciate ligaments (from Essentials of Human Anatomy, 1988).

Three types of surgical procedures have evolved. [Dowson 74, Kurosawa 85, Swanson 73, Walker 88]

(1). Arthrodesis: the knee joint is fixed by fusion of the femur to the tibia. By this method, pain is relieved; stability is gained and deformity is corrected. But mobility is not restored.

(2). Osteotomy: the knee is realigned by cutting femur and tibia. After realignment, the knee is splinted until healing is achieved. This technique may relieve pain, restore stability, and correct deformity, but the result is usually unpredictable. In addition, mobility is not improved.

(3). Arthoplasty: part or all of the working parts of the joint is replaced by a artificial prosthesis. Pain relief is usually achieved. In addition, stability, mobility and correction of deformity can also be attained.



(a). UNICOMPARTMENTAL KNEE REPLACEMENT (b). TOTAL KNEE REPLACEMENT

Fig. 2.3. Schematic diagram of the knee joint replacement

There are two general choices, Total Knee Replacement (TKR) and Unicompartmental Knee Replacement, both of which can be applied in arthoplasty. TKR replaces the entire knee joint by an artificial prosthesis which substitutes bearing surfaces of the distal femur and the proximal tibia (Fig. 2.3(b)), while the unicompartmental knee replacement only replaces one compartment of the joint (Fig. 2.3(a)). Up to now, TKR has been more frequently used. However, unicompartmental

knee replacement is a very important design, in which all ligaments of the joint can be retained and restored to their natural tensions. Therefore, a patient after unicompartmental knee replacement can have a more natural 'walking' sensation than after TKR.

The selection of materials for the artificial prosthesis is quite restricted. The femoral component is usually cast from a chrome-cobalt alloy, while the tibial component consists of a similar metal tray with interchangeable bearing top-insert made from ultra high molecular weight polyethylene. For a successful knee replacement, the femoral component should be placed in neutral flexion/extension in the lateral plane. In addition, there should be adequate ligamentous stability in full extension and flexion. Thus if the prostheses do not fit accurately and if the tibial and femoral components are not properly aligned, the prosthetic joint will perform inadequately and stability will decrease [Dowson 81].

Traditionally, most patients have been quite elderly. However, more knee replacement occurs as a result of sports injuries in recent times. In such cases, a long life from the prostheses is usually required, since they often occur early in the patients life. There is therefore a need for a precise fitting prosthesis which can support an active life style and be able to have ready revision when it is worn out. The latter requirement supports the need for a cementless prosthesis since removal of cement, without major damage to the bone surfaces during a revision, presents a considerable problem. Cementing needs less precise bone cuts, since the cement can be used to fill in the gap between the prostheses and the bone. However, it does add some 20-30 minutes to the operating time and flakes of acrylic cement frequently embed into the bearing surfaces, which accelerates wear amid interfering with smooth motion due to the much higher friction. Cementless prostheses, usually using porous coating, needs shorter follow-ups than cemented implants, but precision machined surfaces are required for a better fit and larger contact area with the bone than for cemented [Walker 93].

2.2 Conventional Surgical Procedures

The typical procedure for TKR is given below as an example. In TKR, conventionally, a qualitative X-ray image of the whole leg is taken prior to surgery. This X-ray image is then used to determine the angle (normally 7 degrees) of the femoral component with respect to the axis of the femoral shaft, so that the centre of the hip, knee, and ankle joint can make a straight line. Based on this determined angle, the distal femur cut is planned in a way that it is perpendicular to the ground and parallel to the proximal tibial cut. The prosthesis size is also estimated. To ensure a good alignment, a jig system is used during the operation to guide the cuts.

The surgical procedures may be different depending on the instrument system used [Technique Ref. 1]. Generally, the knee is first exposed through a longitudinal tendon-splitting incision. The soft-tissue and deformity are released and cruciate ligaments are either removed or retained. For the distal femoral resection, a hole is drilled at the femoral condyles anterior to the origin of the posterior cruciate ligament. A rod is introduced into the hole and an alignment jig is placed on the rod. The alignment jig is adjusted to the pre-planned angle and a drill guide is then slid in. Holes located by the drill guide are drilled through the bone and pins are inserted through the holes. Subsequently, a cutting block is slid over the guide pins and the femoral cuts are made by an oscillating saw. A similar procedure is used to make the proximal tibial cut. Prosthetic components of the appropriate size are inserted and tested. This process is repeated until adequate ligament tension, a range of joint motion and stability are all correct and maintained. The trial components are removed and all the bones are cleaned and dried. The prostheses are then cemented. Finally, the patella is resurfaced and a polyethylene patellar component is inserted and cemented.

The concept of the alignment and cutting guide system for the unicompartmental knee replacement is to some extent similar to that for TKR, although the instruments used are different. The surgical requirements for the unicompartmental knee replacement are more demanding than those for the TKR. Because all the ligaments of the joint are

to be retained, working space between the distal femur and proximal tibia becomes more restricted and the surgeon can only get a limited view.

There are some drawbacks in the current knee replacement jig system. Firstly, since the operation is not strongly linked to the preoperative planning, the surgeon can only execute the resection with the limited view available during the operation. Secondly, using the jig system, each cut is dependent on the quality of the previous jig location and drill. Thirdly, the cuts completed by the oscillating saw lack accuracy and flatness, and harmful heat can be generated. If the saw is slid along the jig surface, it tends to bounce off the bone. The surgeon therefore angles the blade a little: too much and the blade digs into the bone excessively, too little and it bounces off. In addition, the range of sizes required also means that many jigs have to be kept and the procedure must be highly systematic to assure an adequate quality.

The use of a robotic system and a better cutting method, such as using a rotary milling cutter, can ensure that each cut is correct with respect to all others and the tibial and femoral aspects are correctly aligned and inter related. In addition, there is the possibility of reducing the time of the operation. Furthermore, assuming that the computer and robot assisted systems are easy to use, this will in turn reduce the strain on the surgeon. The reduction in training time needed to perform knee surgery will also enable more surgeons to perform this type of surgery.

2.3. Computer and Robot Assisted Systems in Knee Surgery

Robotic assistance in orthopaedic surgery has been reported by a number of authors [Fadda 92/93, Kienzle 93, Matsen III 93, Taylor 92]. As described in the last chapter, the surgical procedures of robotic assistance can be divided into three steps: pre-operative planning, intra-operative registration, and execution.

(1). Pre-operative planning: it is required to position landmarks; image the joint; create a 3D computer model of the bones and perform pre-operative planning. The landmarks are used to link the 3D image, the patient and the robot, and the accuracy of

this process is essential to ensure a quality fit of the prosthesis. As outlined in chapter 1, artificial markers and anatomical features are two general ways used for the registration. Although the artificial marker method is the most commonly used datuming technique because of its convenience in use [Taylor 92], the technique of matching anatomical features, which can attain a better than 1 mm accuracy, is preferred because it does not need invasive pre-operative procedures [Lavallee 92].

CT scans are commonly applied for robotic orthopaedic surgery [Fadda 92], while c-arm x-ray systems, which do not give a quantitative measure, are used to check the placement of the prosthesis components intra-operatively. The use of CT scans can give good definition of bone and also show the adjacent tissue quite well. Typically a maximum of 100 slices are scanned with a spacing of 1.5 mm, mostly around the knee, but some around the ankle and hip to define alignment of the leg. After a 3D computer model is generated from CT scans, simulation of motions of the leg is performed to plan the surgical procedure. The most appropriate size of prosthesis, as well as alignment, is then chosen.

(2). Intra-operative registration: the steps are to first clamp the leg/patient [Davies 94/d, Lea 94]; fix the robot close to the knee; register the robot to the knee and the 3D computer model. In the theatre, the anaesthetised patient is first prepared and positioned on the table with her/his knee flexed and both hip and ankle of the patient are clamped with reference to the table. At this point, the surgeon carries out the preliminary incision by hand, and frees the ligament from attached soft tissue and retracts ligament and tissue from the region to be machined.

The robot is then clamped to the table and registered to the pre-operatively positioned landmarks so that the robot program and the pre-operative 3D computer model can both be matched together with the leg [Taylor 92, Wu 92].

(3) Execution: at this stage, the robot is then used to assist the surgeon in machining the knee bone. Two different approaches have been used:

(I). The robot holds specially designed cutting or drilling guides, and locates the guides at the preoperatively planned position. The surgeon then manually executes

the resection with the help of these guides. [Kienzle 93] used a Puma 560 robot with a drill guide fixed to its end. When the patient is immobilized, the registration to the robot is carried out. The robot moves the drill guide to the pre-planned positions, and the holes are drilled through the guide by the surgeon. [Matsen III 93] designed a three-dimensional distal femoral arthroplasty template which is attached to a robot arm. The robot is passively moved to the desired position with respect to the femur. The surgeon inspects the cut planes and selects the optimal positions of the articular surface of the component with respect to the distal femur, and the desired position and orientation are recorded. Subsequently, the template is exchanged for the cutting and drilling guides . The robot moves to the recorded position and orientation to guide the surgeon in making the cuts and holes. In this approach, a specially designed cutting/drilling guide system is still needed; a very substantial or stiffened robot is also required to resist the large forces caused by cutting. The cutting errors due to the use of the oscillating saw are not reduced. Furthermore, there are difficulties in how to safely drive the robot to an accurate position, so that the surgeon can precisely and easily make the preoperatively planned cuts.

(II). The robot holds the cutter and moves appropriately to resect the correct shapes. Depending on the cutting autonomy and actuation methods, the robot can be used in three modes: active, passive and semi-active.

An active robot with actuators, supervised by the surgeon, can execute the planned cutting autonomously. The example described in chapter 1, [Taylor 89/92] developed a modified industrial five-DOF SCARA robot for total hip replacement. The robot is fitted with a surgical Anspach drill and a 6-DOF force sensor, which is used to monitor the drilling force. Before surgery, titanium locating pins are inserted into the patient's greater trocanter and femoral condyles. These can be seen on the pre-operative CT scans and are used to register the CT images and 3D

computer models intra-operatively. In the theatre, the surgeon operates as normally until the femoral head is removed. The femur is then fixed and referenced to the operating table, and registered to the robot. Subsequently, the implant cavity is machined out by the robot in accordance with a pre-operative plan. After the cutting is completed, the operation is finished by the surgeon in the usual way.

When using an active (autonomous) robot, there are a number of questions and problems that need to be addressed: how to satisfy the safety requirements; how to transfer the surgeon's experience and knowledge of surgical techniques to the robot and how to implement artificial sensors which replace the surgeon's senses (force, touch, vision, and sound etc.). Also, the need for psychological acceptance, both by patients and surgeons, in using an autonomous robotic manipulator creates further difficulty.

A passive robot that has no actuators does not have the same risk as an active robot, which, if adequate safety precautions are not taken, may execute unexpected motions or cutting in the case of a malfunction. The surgeon can move the cutting tool and display its current position on a computer screen. However, a passive arm is usually not good at following a desired trajectory (for instance, cutting a groove) or reaching and maintaining a pre-computed point. [Troccaz] investigated the possibility of using a passive arm with dynamic constraint. The idea is to design a motorised joint clutching system which allows four functions: freely rotating, braking the joint, clockwise rotating only, and counter-clockwise rotating only. When the passive arm is moved, the clutching systems on the joints are controlled to constrain the robot moving direction. Separate motors and encoders are required for each direction of rotation of a joint and further very accurate encoder is required at the joint itself. Nevertheless, the technique of the clutching system and control scheme with fast response and good performance has yet to be achieved.

Another way of approaching the problem of robotic knee surgery is to design a semi-active special purpose robot, with both mechanical constraint and active motion constraint determined by a control strategy. A cutter and a 'Dead-Man'

switch [Davies 93] are fixed to the tip of the robot . The surgeon holds the cutter motor and moves it, back driving the power transmission system of the robot. Since the surgeon can supervise and control the robot directly; he is able to execute the cutting using his innate sensing, judgement and experience. From the point of view of both surgeon and patient, the robot is merely a "tool". It is evident that the surgeon performs the operation and not the robot. For these reasons, it is believed that this approach is more acceptable to both surgeon and patient. The use of additional mechanical constraints can ensure that the motions of the robot are confined to a designed safe region in the unlikely event that all other safety measures fail [Davies 92]. A force control strategy, with artificial motion constraints, can assist the surgeon to execute the pre-planned cuts accurately and efficiently.

2.4. Objectives of the Control Strategy and Artificial Motion Constraint in Assisting the Execution of Planned Resection

This thesis concentrates on the strategy for intervention and assumes that the preliminary shapes of pre-operative planning, modelling, patient immobilizing, and robot registration have been completed. Based on the 3D model., the shape and orientation of the planned resections are transferred into the robot's coordinate reference frame. The semi-active robot is then moved manually by the surgeon to execute the planned resections. In order to assist the surgeon to actually achieve these cuts, force control strategies are developed in which the motion of robot is constrained. For each cut, the surgeon can see the cutter's position on the computer screen as well as its actual position in the knee joint and can freely move the robot into the desired initial position. Once the cutter is moved inside the artificial constraint region (Fig. 2.4), which is formed on the basis of the pre-operative data, the surgeon can set the robot into force control mode. Subsequently, the surgeon can observe the trace of the cutter position, relevant to the desired cutting area, by viewing

the computer screen. Within the constraint envelope, the surgeon is able to freely move the tip of the robot which contains a rotating cutter. At the edge of the defined region, the surgeon can only move the robot along, but not beyond, the boundary. Other motions, such as that within the guard region, are prevented by the force control strategy.

Thus, in this system, the surgeon can directly control the rate of cutting by his own senses and by his experience and judgement. In addition he is able to complete the cuts more accurately and efficiently with the help of the artificial motion constraint imposed by the robot. For instance, the cuts can be divided into two stages: rough and precise cutting. Each has a separate region of motion constraint. Fig. 2.5 shows the cutting pattern of a compound shape such as the tibial plateau. Fig. 2.5a shows how a central region roughing cut can be removed at speed. Fig. 2.5b shows a slower, separately programmed, final cut which follows the profile to give an accurate shape.

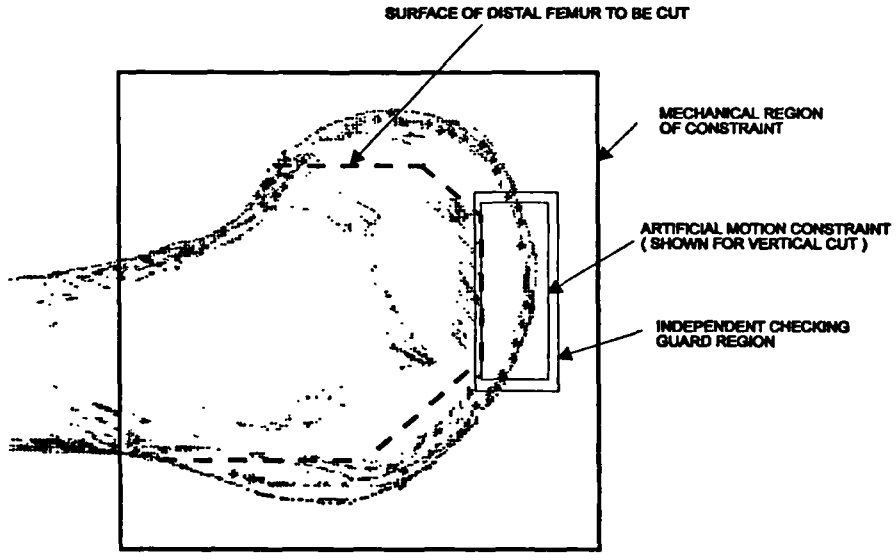


Fig. 2.4 Concepts of mechanical constraint and artificial motion constraint

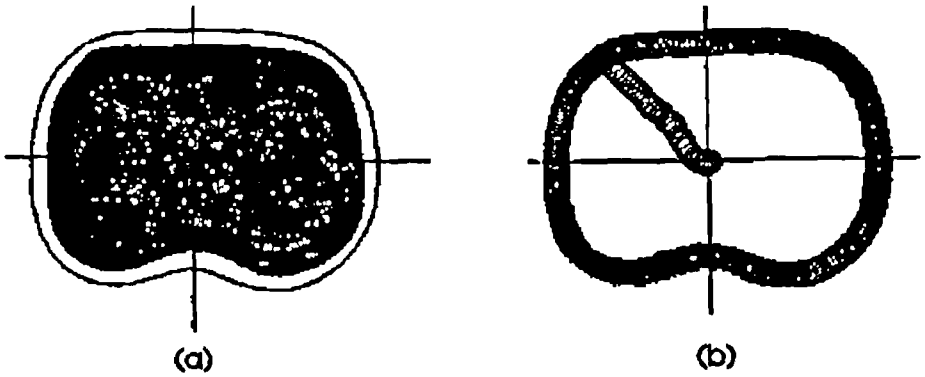


Fig. 2.5. shows a typical tibial shape (a) with a centralized roughing cut (b) with a finished profile cut

Another advantage of the control strategy is to help the surgeon to make a preliminary cut. For instance, in unicompartmental knee replacement, the working space between the distal femur and proximal tibia is quite limited because the ligaments of the joint are to be retained. The surgeon can define the artificial motion constraint by a mouse or light-pen intra-operatively, and make some 'safe' cuts to enlarge the working space without damaging ligaments or adjacent soft tissue.

The experience gained at Imperial College has shown that it is desirable to use a special purpose robot that is specifically designed for the task. In this way, the number of axes of motion can be restricted to a minimum, with force levels just adequate for the task, thus avoiding the possibility of high force robots flying off in unknown directions if a failure should occur. However, the robot is still an active manipulator and the safety issue is always the most important concern. There are some methods proposed to enhance the safety of the operation :

(1) The control strategy which forms an artificial motion constraint can assist the surgeon in achieving the cuts precisely and will also prevent the surgeon from cutting the bone or tissue outside the constraint. If a force sensor is used for the control feedback, (which will be discussed in more detail in later chapters) it can be used as an additional safety checking mechanism, where a 'safe' force level is pre-defined. When the sensed force exceeds the defined level, the power of the cutter will be shut down at once and an alarm signal is triggered.

(2) A 'Dead-Man' switch [Davies] is fixed to the end of the robot. Both the cutting action and the motions of the robot are associated with the 'Dead-Man' switch concept whereby the surgeon has only to let go of a switch to bring the system to a safe and predictable state.

(3) An independent position sensor system can also be used to monitor the movement of the robot. As shown in Fig. 2.4, another 'safe' region, which is slightly larger than the artificial motion constraint, is defined for the independent sensor system. When the robot moves out of this region, the power of the cutter and robot is shut down. A electron-magnet brake can be used to stop the robot at the same time.

(4) Finally, a robust mechanical constraint [Davies] provides a secure guarantee whereby the robot is constrained to move only in a pre-designed 'safe' volume thereby avoiding damage to adjacent regions of the patient or to other personnel, in case all other safety monitoring systems fail.

2.5. Task Specification

The motion constraint can be classified into three types:

(1). Point constraint : this task is the same as the normal position control used in an industrial robot. The strategy is that the robot is switched to position control mode only when it is moved by the surgeon to within a small distance of the defined point. The movement trajectory is decided by the surgeon intra-operatively and does not have to be designed in advance.

(2). Trajectory motion constraint: the surgeon is able to move the robot along a programmed trajectory such as a straight line or curve. Additionally, the length of the trajectory is limited. Thus, a groove of prescribed width and length can be easily and accurately cut.

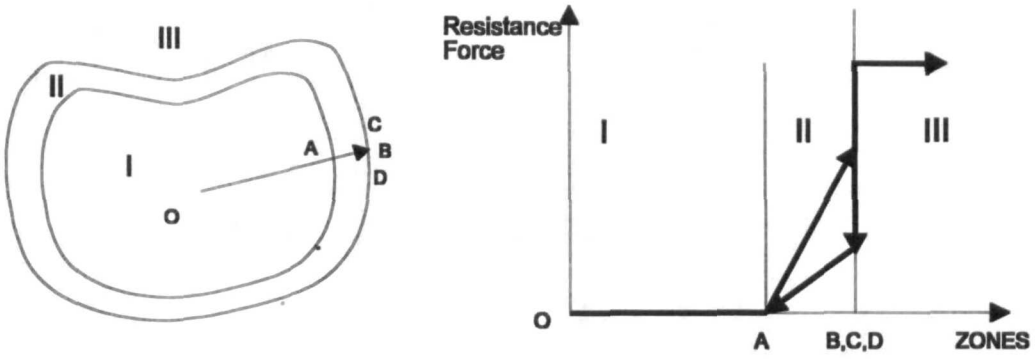


Fig. 2.6. Resistance Force in 3 different zones and different directions of motion constraint

(3). Regional motion constraint : figure 2.6 shows a schematic diagram of the resistance force in different zones and in different directions. Region I is a permitted low force zone. III is a restricted region requiring very high forces before it can be moved into, whilst II is an intermediate, transition, zone. The cutter could readily be moved from O to A, then with more resistance to point B. Movement from point B into zone III would require very high force, whilst motion along the profile to C or D would be easier. In zone I and II, the robot will stay at its current position if there is no guiding force. While the robot will be moved to the nearest point of the boundary (between zone II and III) by control strategy, if the robot is moved into zone III. With the help of the constraint, the surgeon is able to cut a defined region precisely.

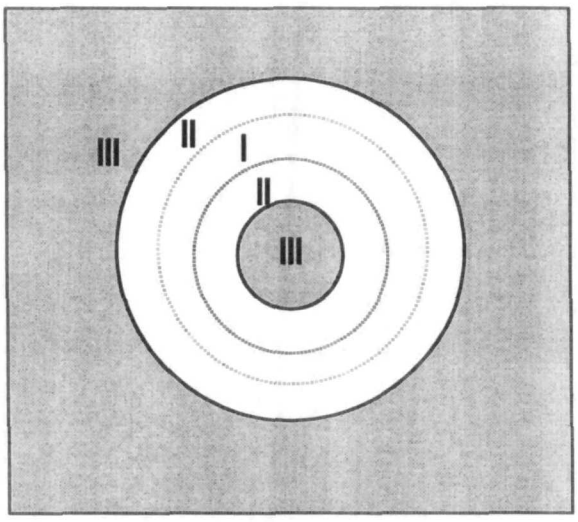


Fig. 2.7 A ring shape of motion constraint

Using a similar strategy to the regional constraint, different shapes of motion constraint can be constructed. For instance, Fig 2.7 shows a ring shape of motion constraint, where the inside and outside regions are the 'no go' areas.

2.6 Conclusion

In this chapter, the requirements of knee joint replacement have been described. The drawbacks of the conventional surgical procedures are also described. After reviewing the relevant projects of the computer and robot assisted systems in knee surgery, a semi-active special purpose robot, (with both mechanical constraint and active motion constraint formed by a control strategy), is proposed in which a robot with a cutter fixed to its end is moved by the surgeon by hand. In this way, in addition to the sensors on the robot, the surgeon can use all his/her innate sensory capabilities to directly monitor the cutter and its programs together with its performance. Since the robot is an active manipulator, some suggestions are also proposed to ensure the safety of the robot. The control strategy which constructs the artificial motion constraint is able to assist the surgeon in achieving the pre-planned cuts easily and precisely.

In the next chapter, robot position and force control algorithms are reviewed and discussed, and the following chapters describe and analyze force control strategies for the knee joint replacement task. Subsequently, both computer simulations and experiments are carried out to prove the feasibility and performance of this new approach.

CHAPTER 3

REVIEW OF ROBOT CONTROL STRATEGY

This chapter describes robot position and force control algorithms used in industrial applications. It is not intended to cover all the industrial control strategies that have been developed, but rather to highlight the aspects and concepts which will be useful for developing a control algorithm for robot assisted systems in knee surgery. Firstly, the robot position control strategy is reviewed. Joint based and Cartesian based control schemes for robot positioning are then examined, and nonlinear decoupling techniques are also described. Finally, a range of force control schemes are classified and compared.

3.1 Robot Position Control

The objectives of robot position control algorithms are primarily to control the joint actuators, so that the desired position of the robot can be achieved. In general, the torque commands (output of the controller) are derived by using the information (feedback) from joint position sensors which is compared with the desired position (input of the system) to compute the torques required. The function of the control law is to process the error signal (the difference between the input and the feedback), and then to command the robot actuators, so that the robot can move in the desired way. Depending on the error signal of the control loop which is formed in the joint space (i.e., the space of all joint vectors [after Craig 86]) or in Cartesian space (i.e. task oriented space or operational space) , the robot control scheme can be divided into two categories: joint based and Cartesian based [Craig 86]. Fig 3.1 shows a schematic diagram for the joint based control algorithm, where Xd, $\dot{X}d$, and $\ddot{X}d$ are the desired

position, velocity and acceleration vector of the robot's end-effector respectively. θ_d, $\dot{\theta}_d$, and $\ddot{\theta}_d$ are the respective desired joint position , velocity, and acceleration vector, and τ is the torque command vector for joint actuators.

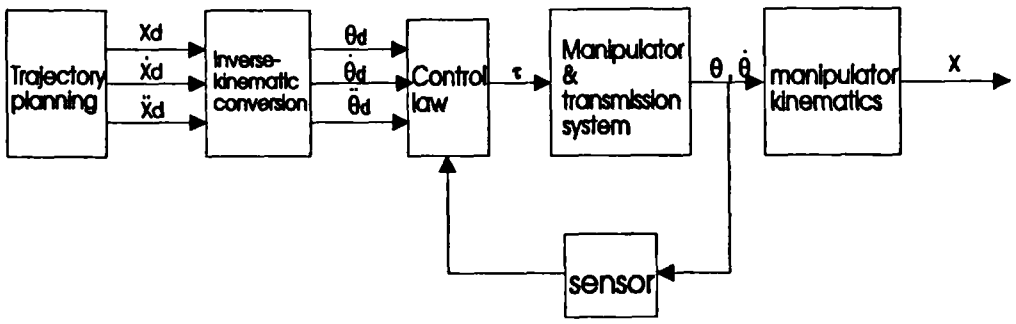


Fig. 3.1 Joint based control scheme

The inverse kinematics conversion is given by [Asada 86]

$$\theta_d = Invkin(X_d),$$
$$\dot{\theta}_d = J^{-1}(\theta)\dot{X}_d, \qquad (3.1)$$
$$\ddot{\theta}_d = \dot{J}^{-1}(\theta)\dot{X}_d + J^{-1}(\theta)\ddot{X}_d.$$

where J is the Jacobian matrix. It can be seen that equation (3.1) is computationally extensive. Thus, in practice, usually only the solution for θ_d is calculated by the inverse kinematic equation. $\dot{\theta}_d$ and $\ddot{\theta}_d$ are normally derived numerically by the first and second difference of θ_d. Figure 3.2 shows the Cartesian based control scheme, where F is the force command in Cartesian coordinates, and satisfies

$$\tau = J^T F. \qquad (3.2)$$

The kinematics and transformation computations are now included in the 'inner loop'. This may be a disadvantange of the Cartesian based control scheme. Because the resulting system will run at a lower sampling frequency compared with the joint based control method (given the same size computer) and generally degrade the stability and disturbance-rejection ability of the system.

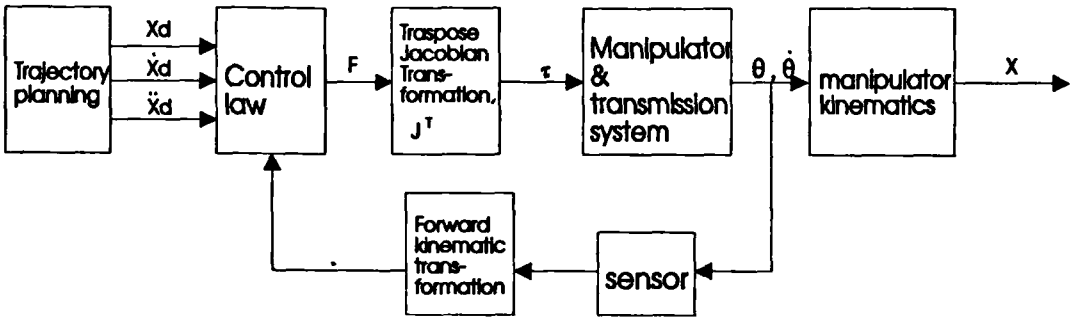


Fig 3.2 Cartesian based control scheme

The design of the control law either in joint based or Cartesian based schemes has inherently non-linear problems [Fu 87]. One of the most effective techniques is to use the nonlinear dynamic decoupling approach [Freund 75, Fu 87, Paul 81], which decouples the effect of the non-linear terms. Linear control theory is then applied to analyze and design the controller.

3.1.1 Joint based control scheme

Generally, the dynamic equation for the manipulator in joint space can be described as [Craig 86, Fu 87]

$$\tau = M(\theta)\ddot{\theta} + h(\theta,\dot{\theta}) + c(\theta), \quad (3.3)$$

where τ = n×1 torque vector,

$\theta, \dot{\theta}, \ddot{\theta}$ = n×1 joint position, velocity and acceleration vector respectively,

$M(\theta)$ = n×n inertial matrix of the manipulator,

$h(\theta,\dot{\theta})$ = n×1 nonlinear centrifugal and Coriolis force vector,

$c(\theta)$ = n×1 gravity force vector.

It can be seen that the manipulator dynamics is highly coupled and nonlinear. In order to apply the linear control theory to design the control law, two general methods are used :

(1).Neglect the effect of the coupling terms from other links and design an independent control algorithm for each joint (independent joint control method).

(2).Take the nonlinear terms into account and design the control law by decoupling and linearizing techniques.

3.1.1.1 Independent joint control scheme [Craig 86]

This method is utilized by nearly all commercial industrial robots, because it is easy to implement and the performance is acceptable for the present industrial applications. In this method, each joint has its own control algorithm, and the influence from other joints is ignored. It is difficult to precisely analyze and predict the performance of this control strategy. However, the design concept can be described as follows. As seen in equation (3.3), the dynamic equation for joint i can be given by

$$\tau_i = M_{ii}\ddot{\theta}_i + \sum_{i \neq j} M_{ij}\ddot{\theta}_j + h_i(\theta,\dot{\theta}) + c_i(\theta), \qquad (3.4)$$

where

$\ddot{\theta}_i$ = acceleration of joint i,

M_{ij} = (i^{th} row , j^{th} column) element of M(θ) matrix,

h_i = i^{th} element of $h(\theta,\dot{\theta})$ vector,

c_i = i^{th} element of c(θ) vector.

and the equation (3.4) can be rewritten as

$$\tau_i = M_{ii}\ddot{\theta}_i + F_{dist}, \qquad (3.5)$$

where

$$F_{dist} = \sum_{i \neq j} M_{ij}\ddot{\theta}_j + h_i(\theta,\dot{\theta}) + c_i(\theta) \qquad (3.6)$$

is regarded as a disturbance force.

Note

(i). The effective inertia M_{ii} is variable. Usually, its variation due to the configuration change can be significantly reduced by multiplying the square of the gear ratio. The

maximum inertia is often less than two times the minimum inertia [Khabit, Van de vegte],

$$\frac{(M_{ii})_{max}}{(M_{ii})_{min}} \cong 2. \quad (3.7)$$

However, it is important to calculate the effective inertia, as this affects the response time of the robot and the acceleration compensation.

(ii). Concerning the dynamic disturbance F_{dist} , some factors should be considered. Firstly , similar to the variation of the inertia M_{ii}, the coupling forces from other links are also masked by the function of the gear ratio. Secondly, the magnitude of the centrifugal are proportional to the square of the velocity in each case. Thus they only come into play when the robot is moving at high speed, at which time their effects are minimal, as the position tolerance of the robot is usually large at high speed. The effects of the velocity-related coupling will not affect the system stability but simply cause position and velocity error. Finally, the main effect of the gravity force is to produce a steady state error, which can be eliminated by an integral feedback or by the use of a counter-balanced mechanism design.

If a Proportional and Derivative (PD) controller is applied and the effective inertia M_{ii} is equal to M after linearizing, the closed loop control diagram for joint i is as shown in Fig. 3.3.

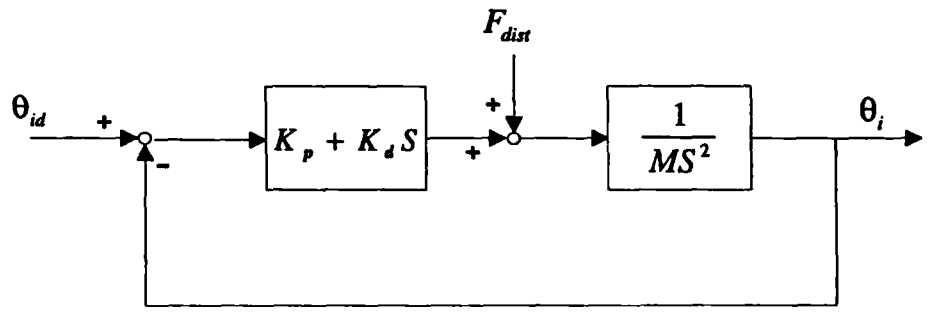


Fig. 3.3 Control diagram for one joint

Where θ_{id} is the desired joint position; θ_i is the actual position of joint i; S is the Laplace operator; and Kp and Kd are the respective proportional and velocity gains.

Difficulties in the controller design for each joint are due to the tracking and disturbance rejection problems, and "classical" or "modern" linear control theory can be used to analyze the performance of the feedback system [Brogan 85, Kuo 80/82], . The Laplace transform function of the closed loop system is given by

$$\theta = \frac{K_d S + K_p}{MS^2 + K_d S + K_p}\theta_d(S) + \frac{F_{dist}(S)}{MS^2 + K_d S + K_p} \qquad (3.8)$$

The transient response can be estimated by the characteristic equation

$$MS^2 + K_d S + K_p = M(S^2 + 2\xi\omega_n S + \omega_n^2), \qquad (3.9)$$

where ξ = the damping ratio and ω_n = the natural frequency of the system. Kp and Kd can be designed in such a way that the overall system becomes overdamped [Franklin 86]. Increasing the value of Kp can increase the system response. However, the output of the actuator is always limited. When the actuator is saturated, increasing the feedback gain can not increase the system response, but will produce a larger overshoot. It should be noted that the closed loop system will be stable for all positive values of Kp and Kd and bounded disturbance. If the system has a step input $\theta_d(S) = \theta_d/S$ and a constant disturbance $F_{dist}(S) = F_{dist}/S$, the steady state error e_{ss} can be derived by the final value theorem [Kuo 82],

$$\begin{aligned} e_{ss} &= \lim_{s\to 0} S(\theta(S) - \theta_d(S)) \\ &= -\frac{F_{dist}}{K_p} \end{aligned} \qquad (3.10)$$

One can see that, theoretically, the steady state error due to the constant disturbance can be reduced to an arbitrarily small value by increasing the proportional gain Kp. However, the maximum value of Kp is limited by hardware characteristics such as the resonance frequency of the mechanism, etc. If an integral feedback K_i/S is added to the control law, the closed loop function is then given by

$$\theta = \frac{K_d S^2 + K_p S + K_i}{MS^3 + K_d S^2 + K_p S + K_i}\theta_d(S) + \frac{SF_{dist}(S)}{MS^3 + K_d S^2 + K_p S + K_i} \qquad (3.11)$$

The overall system becomes type 2 [Kuo 82], and e_{ss} will be zero for a step input and constant disturbance, providing the system is stable. Applying the Routh-Hurwitz criterion [Kuo] to the characteristic equation, the system is stable when all feedback gains Kp, Kd and Ki are positive, and

$$K_i < \frac{K_p K_d}{M} \quad (3.12)$$

The general independent Proportional, Integrative and Derivative (PID) joint based control diagram for a multi-joint manipulator is shown as Fig. 3.4, where Kp, Kd and Ki are n×n diagonal matrices.

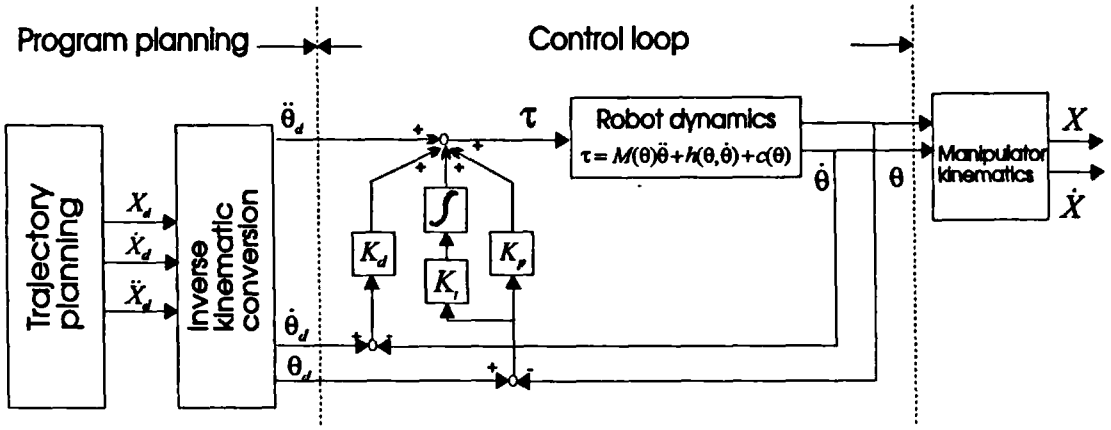


Fig. 3.4 Independent joint control scheme with PID control law

3.1.1.2 Joint based control with nonlinear decoupling (Computed torque technique)

When the dynamic interaction forces between robot joints are taken into account, the nonlinear decoupling technique is usually used and the linear control law could thus be applied [Paul 81, Fu 87, Spong 89]. Various decoupling methods may be used. A general concept is presented below.

As described in the last section, the dynamic equation of the manipulator can be represented by

$$\tau = M(\theta)\ddot{\theta} + h(\theta,\dot{\theta}) + c(\theta), \quad (3.3)$$

The control law is designed as

$$\tau = M^*(\theta)[\ddot{\theta}_d + K_d(\dot{\theta}_d - \dot{\theta}) + K_p(\theta_d - \theta)] + h^*(\dot{\theta},\theta) + c^*(\theta), \quad (3.13)$$

where Kp and Kd are n×n proportional and velocity matrices respectively. Substituting equation (3.13) into equation (3.3), the closed loop dynamic equation is given by

$$M(\theta)\ddot{\theta} + h(\theta,\dot{\theta}) + c(\theta) = M^*(\theta)[\ddot{\theta}_d + K_d(\dot{\theta}_d - \dot{\theta}) + K_p(\theta_d - \theta)] + h^*(\dot{\theta},\theta) + c^*(\theta) \quad (3.14)$$

If $M^*(\theta), h^*(\dot{\theta},\theta)$ and $c^*(\theta)$ are equal to $M(\theta), h(\dot{\theta},\theta)$ and $c(\theta)$ respectively, the closed loop equation can be reduced to

$$M(\theta)[(\ddot{\theta}_d - \ddot{\theta}) + K_d(\dot{\theta}_d - \dot{\theta}) + (\theta_d - \theta)] = 0. \quad (3.15)$$

Since M(θ) is always nonsingular, equation (3.15) can be expressed as

$$\Delta\ddot{\theta} + K_d(\Delta\dot{\theta}) + K_p(\Delta\theta) = 0, \quad (3.16)$$

where $\Delta\theta = \theta_d - \theta$. The closed loop equation is now decoupled. Kd and Kp can be chosen so that the position error vector Δθ will approach zero asymptotically. The schematic diagram for the closed loop system by using the decoupling method with a PID controller is shown in Fig. 3.5.

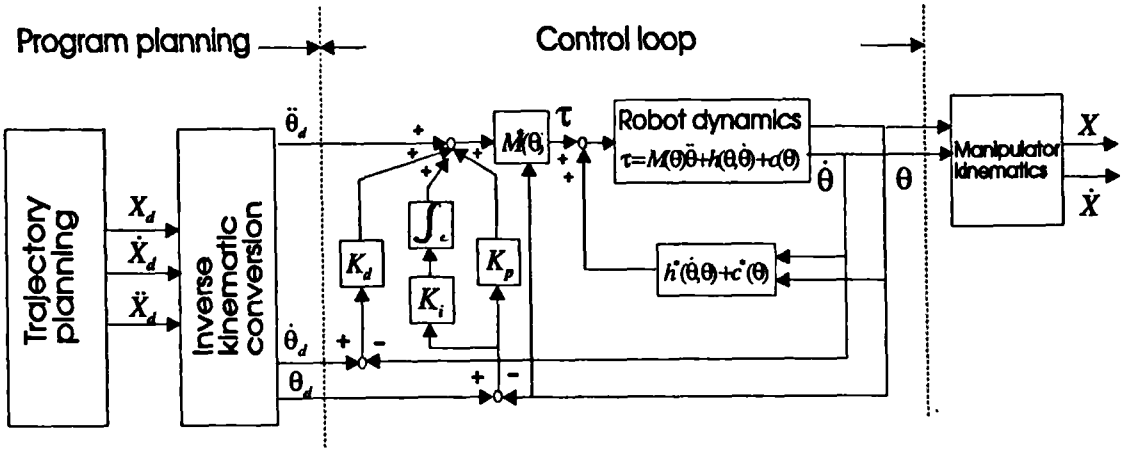


Fig. 3.5 Joint based control with decoupling scheme

Since the decoupling method takes the interaction forces between the robot links into account, the performance should be better than the independent joint control scheme. However, there exist some drawbacks to this method,

(i) The control law in equation (3.13) needs a lot of computation time. [Paul 81] suggested that when the robot is moving at a low speed, the velocity-related term $h^*(\dot{\theta},\theta)$ and the off-diagonal elements of $M^*(\theta)$ may be neglected (i.e. only the effective inertia need be considered). The control law then becomes

$$\tau = diag\{M^*(\theta)\}[\ddot{\theta}_d + K_d(\dot{\theta}_d - \dot{\theta}) + (\theta_d - \theta)] + c^*(\theta) \qquad (3.17)$$

Since $M^*(\theta)$ and $c^*(\theta)$ are functions of the robot position only, they can be calculated in advance for all the working space and stored in a huge memory if this is available [Raibert 81]. Otherwise, they can be computed by a second processor with a lower computation rate than the rate of the closed loop system [Craig 86].

(ii). If the parameters of $M^*(\theta)$, $h^*(\dot{\theta},\theta)$ and $c^*(\theta)$ are perfectly predicted, it is easy to predict the performance and stability of the system. However, modelling errors such as friction and also computation errors always exist in reality. Although the adaptive control method may be used to update the errors of the model parameters, the stability and performance analysis of the system is still very difficult to achieve [Fu 87].

3.1.2 Cartesian based control

In a joint based control algorithm, the numerical differentiation to derive the desired joint velocity and acceleration ($\dot{\theta}_d$, and $\ddot{\theta}_d$) will introduce a great deal of noise and will also introduce a lag unless the numerical differentiation can be performed with a non-causal filter [Craig 86]. In order to solve the problem, one alternative is to design the control law in Cartesian space, in which the robot end-effector position is derived by using forward kinematic equations from the sensed joint positions and comparing these with the desired position in Cartesian coordinates. As in the last section, the

decoupling method can be used in Cartesian based control. The robot dynamic equation can be expressed as [Fu 87]

$$F = M_x(\theta)\ddot{X} + h_x(\theta,\dot{\theta}) + c_x(\theta), \qquad (3.18)$$

where

F = the force vector acting on the end-effector of the robot,

X = the Cartesian position vector of the robot,

$M_x(\theta)$ = the inertia matrix in Cartesian space,

$h_x(\theta,\dot{\theta})$ = the velocity-related force vector in Cartesian space,

and $c_x(\theta)$ = the gravity force vector in Cartesian space.

The formula for computing the Cartesian dynamics from the joint dynamic equation (3.3) is summarized below [Fu 87].

$$\begin{aligned} &F = J^{-T}\tau \\ &M_x(\theta) = J^{-T}(\theta)M(\theta)J^{-1}(\theta), \\ &h_x(\theta,\dot{\theta}) = J^{-T}(\theta)[h(\theta,\dot{\theta}) - M(\theta)J^{-1}(\theta)\dot{J}(\theta)\dot{\theta}], \\ &c_x(\theta) = J^{-T}(\theta)c(\theta). \end{aligned} \qquad (3.19)$$

The control law can be designed as

$$F = M_x^*(\theta)[\ddot{X}_d + K_d(\dot{X}_d - \dot{X}) + K_p(X_d - X)] + h_x^*(\theta,\dot{\theta}) + c_x^*(\theta). \qquad (3.20)$$

Substituting equation (3.19) into equation (3.18), when the dynamic model has the exact parameters, the closed loop function is given by

$$F = M_x(\theta)[\Delta\ddot{X} + K_d(\Delta\dot{X}) + K_p(\Delta X)], \qquad (3.21)$$

where the position error ΔX is equal to (X_d-X). As described in the last section, the position error ΔX will approach zero asymptotically by choosing an adequate velocity gain matrix Kd and proportional gain matrix Kp. The required torques for the joint actuators can then be calculated by equation (3.2)

$$\tau = J^T F. \qquad (3.2)$$

Figure 3.6 shows the block diagram of the Cartesian based control with a nonlinear decoupling scheme.

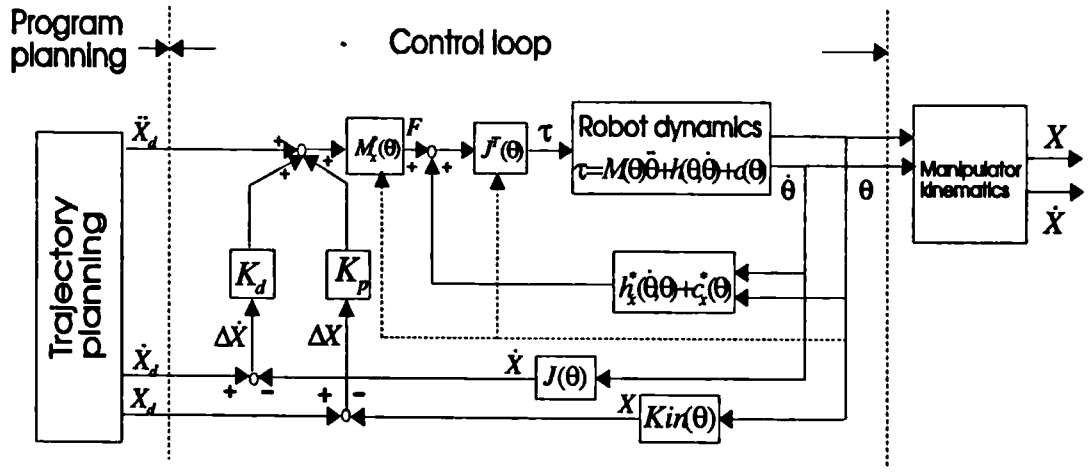


Fig. 3.6 Cartesian based control with decoupling scheme

Compared with the decoupling control method in joint space, the inverse kinematics transformation is replaced by a Jacobian matrix and forward kinematics computation, and the computation complexity is not reduced. The computations are, however, all included in the closed loop. The complexity of computation will decrease the relative sampling rate, and thus the stability, of the overall system. As in the joint based control method, one feasible implementation is to neglect the velocity-related terms in order to reduce the computation complexity. In addition, the parameters of the dynamic equation can be calculated and stored in the memory in advance or computed by a second control processor to increase the sampling rate of the closed loop.

3.2 Robot force control

In the previous section, robot position control has been described. When a robot is moving freely along a trajectory, the position control algorithm alone is able to achieve the desired objectives. However, when any contact between the robot end-effector and the environment is required, a force control approach will have to be applied to accomplish the task. A general scheme for robot force control is shown in

Fig. 3.7, where the interaction force is generated because of the contact between the robot and the environment.

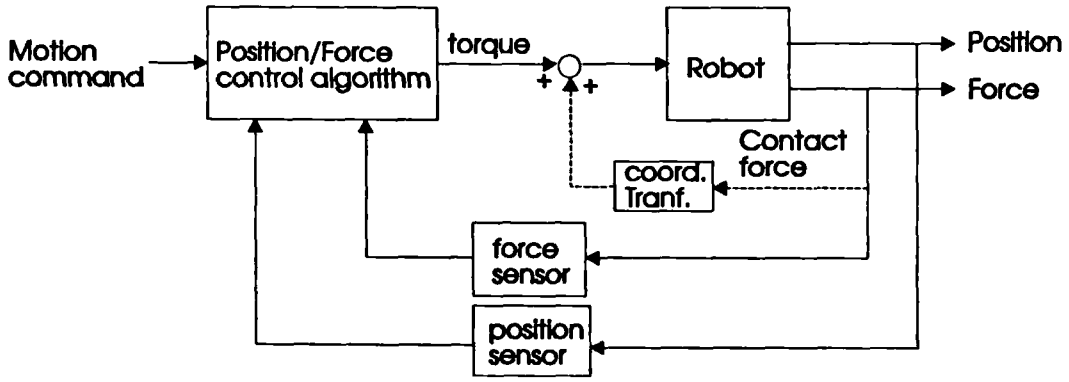


Fig. 3.7 A general force control scheme

It is seen that the interaction force will affect the robot dynamics directly. There are two generally accepted ways of simplifying the control scheme, which are: (1) the interaction force is modelled and combined with the robot dynamics [Kazerooni 89]. (2) assuming the contact force can either be neglected or is compensated by direct feedback to the joint actuators [Schutter 88/a]. Fig 3.8 shows the simplified version of the control scheme of Fig. 3.7, where the contact force is removed from the diagram. When the interaction force between the robot and the environment is sensed, the force signal can be processed in three different ways [Maples 86]: (i) becoming torque command, (ii) becoming velocity command, or (iii) becoming position command, to modify the desired motion commands.

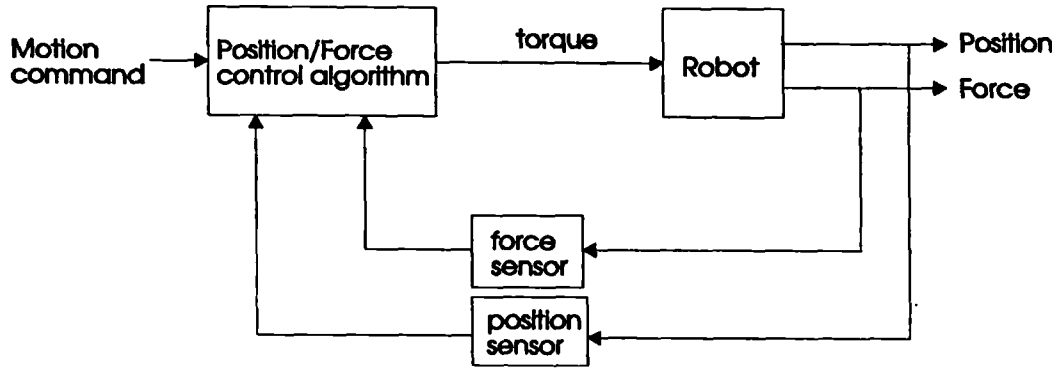


Fig. 3.8 A simplified version of Fig. 3.7

Several papers attempting to catagorize force control techniques have been published [Whitney 87, Patarinski 93, Maples 86]. Force control techniques can be classified in a similar way to those of position control techniques. They can be categorized into joint based and Cartesian based control depending on the error signal between the reference position and whether the robot position is formed in joint space or Cartesian space [Maples 86]. Another general classification of the force control algorithm, based on the work of Whitney, is shown in Fig. 3.9. The explicit force control scheme (not shown) is treated as a subclass of the hybrid position/force control method.

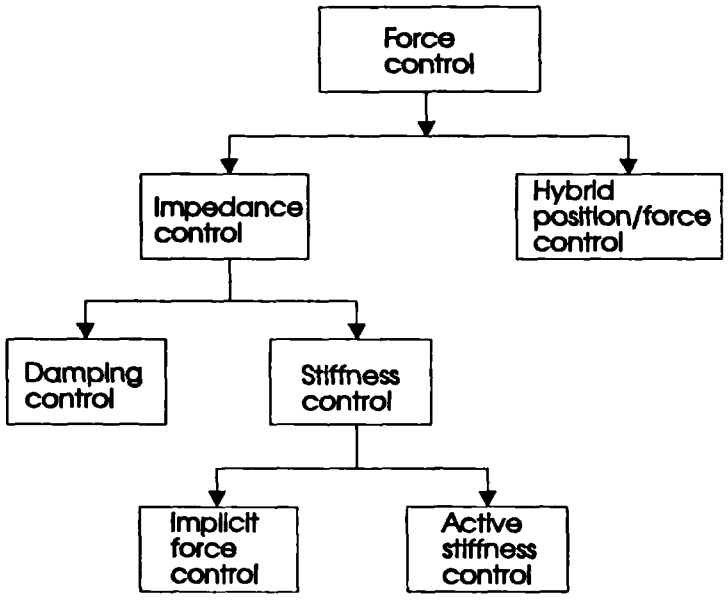


Fig. 3.9 Basic approaches of position and force control

3.2.1 Hybrid position/force control

The hybrid position/force control approach [Mason 81, Railbert 81] divides Cartesian task space into two orthogonal sets : position controlled and force controlled sub-spaces. The position control algorithm for the position controlled sub-space can be designed by the Cartesian based control methods given in the last section. The algorithm for the force controlled subspace can be treated as explicit force control and the concept is described as follows [Craig 86].

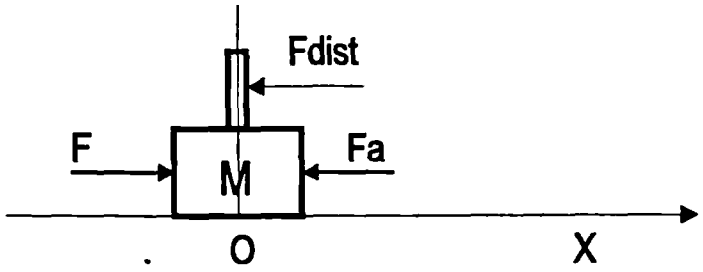


Fig. 3.10. A schematic diagram of an one DOF system

Consider a single degree of freedom system as shown in Fig. 3.10, where M is the mass of the system; X is the actual position of the system; F is the output of the actuator; F_{dist} is the disturbance force; and F_a is the interaction force on the environment. The dynamic equation satisfies

$$M\ddot{X} = F - F_{dist} - F_a . \qquad (3.22)$$

If the interaction force Fa is modelled as a spring system with a stiffness Ka, i.e. $F_a = K_a X$, equation (3.22) becomes

$$MK_a^{-1}\ddot{F}_a + F_a + F_{dist} = F . \qquad (3.23)$$

The objective of the control law is to keep the contact force Fa as the desired force Fd. In a similar way to the decoupled control technique in equation (3.13), the control law can be given by

$$F = MK_a^{-1}[\ddot{F}_d + K_d(\dot{F}_d - \dot{F}_a) + K_p(F_d - F_a)] + F_a , \qquad (3.24)$$

where Kp is the proportional gain and Kd is the derivative gain. Substituting equation (3.24) into equation (3.23), the closed loop equation is given by

$$\Delta\ddot{F} + K_d(\Delta\dot{F}) + K_p(\Delta F) = M^{-1}K_a F_{dist} , \qquad (3.25)$$

where $\Delta F = F_d - F_a$. The steady state error of the desired force is equal to

$$\Delta F = \frac{M^{-1}K_a F_{dist}}{K_p} . \qquad (3.26)$$

If the desired force F_d is a constant, $\ddot{F}_d$ and $\dot{F}_d$ then become zero. In practice, the derivative of the contact force can be derived by $\dot{F}_a = K_a \dot{X}$. Thus the control law of equation (3.24) is simplified as

$$F = -MK_d \dot{X} + MK_a^{-1} K_p (F_d - F_a) + F_a. \qquad (3.27)$$

Figure 3.11 shows the block diagram of this force control scheme for a one DOF system.

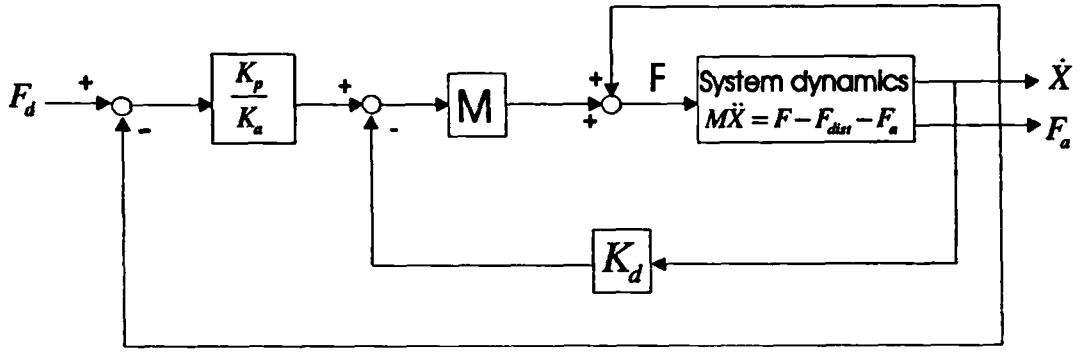


Fig. 3.11 A force control diagram for a one DOF system

By extending the force control concept to multi-joint manipulators and combining it with the position control method, the conceptual diagram of the hybrid position/force control can be shown as in Fig. 3.12. S and S' are diagonal matrices with ones and zeros. When a one is present in S, a zero is also present in S'. By selecting S and S', the axes to be position controlled or force controlled can be decided. The selection of the position controlled and force controlled axes depends on the constraints of the task [Mason], such as contour following and turning a crank. In this control approach, the task planning and control strategy can be easily separated, and this will benefit the users in specifying the tasks in programming [Schutter 88/b]. However, as with the Cartesian based position control, the inverse Jacobian matrix computation is very time-consuming. In addition, the uncertainty of the task geometry may produce instability.

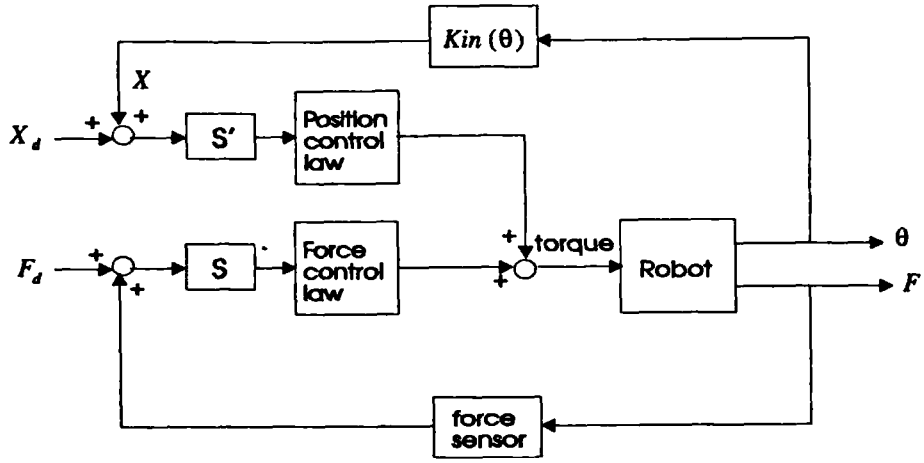


Fig 3.12 Hybrid position/force control scheme

3.2.2 Impedance control

As seen in the last section, the desired position and desired force can be expressed explicitly in the hybrid position/force control algorithm, and the control law is designed to achieve the desired goals. The impedance control theory considers the force control problem from different points of view, in which a desired 'impedance' of the robot is designed. However, the desired force can not be stated explicitly. While the robot is in contact with the environment, the end-effector of the robot should behave like a type of mechanical system, such as a spring and damper system [Hogan 85, Goldenberg 88]. Consider the conceptual diagram in Fig. 3.13, where X is the motion command and F is the resulting contact force. The robot system can then be regarded as a mechanical element with impedance Z. For instance, if X is a position command and $F = ZX$, the robot system will behave as a spring system. Thus the impedance control attempts to maintain the relationship between the contact force and the motion command error, i.e. the robot can behave as a desired impedance or admittance. The impedance control can be seen as the generalization of the stiffness control and damping control, where the robot is designed as a spring and damper system, respectively.

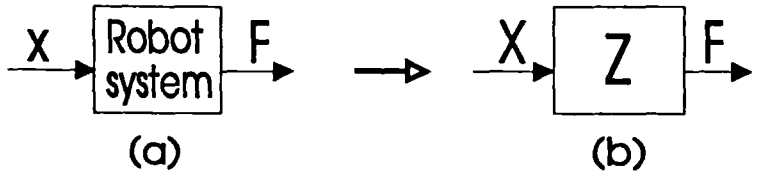


Fig. 3.13 The robot system is considered as an impedance.

3.2.2.1 Damping control

The damping control strategy can be interpreted as a means of maintaining the relationship between the desired velocity and the contact force, thus the end-effector of the robot is seen as a damper system by the environment [Whitney 87]. The control scheme can be depicted as in Fig. 3.14, where $\dot{X}_d$ is the desired velocity vector; K_{F1} is the desired admittance (impedance^{-1}) matrix with dimensions velocity/force; X_e is the location of the environment and K_E is its stiffness; and F is the contact force. It should be noted that $\Delta\dot{X}$ is seen as the error of desired impedance to be maintained instead of the error of the desired velocity. $\Delta\dot{X}$ is then converted to the desired joint motion commands, which are completed by the position feedback control.

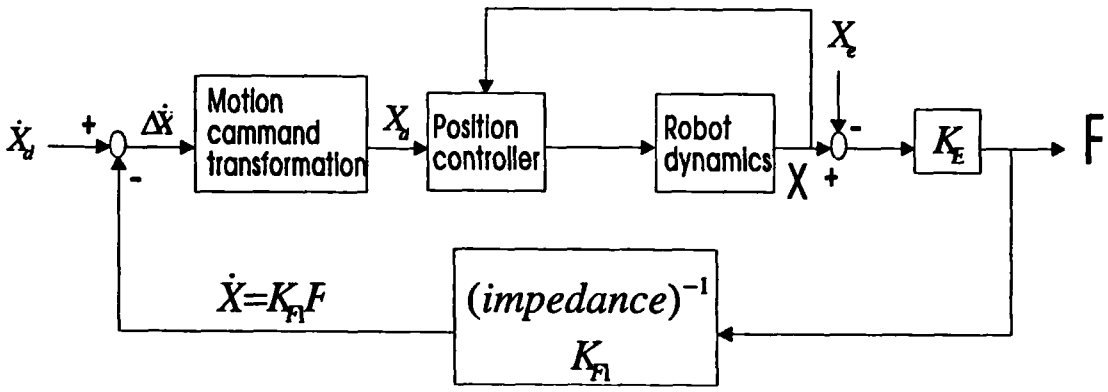


Fig. 3.14 Conceptual diagram of damping control scheme

One interesting application of the damping control methods is to guide the robot by hand. [Whitney 77] stated if $\dot{X}_d$ is equal to zero and an external force F is applied by pushing or pulling the end of the robot, $\Delta\dot{X}$ will become non zero. Thus the robot will

begin to move in the direction of pushing/pulling proportionally to the magnitude of the applied force F. However, it is seen from Fig. 3.14 that when an external force is applied to guide the robot, the location of the environment (i.e. the position of the operator's hand) X_e is a variable function, because the operator will continuously move his hand to follow the movement of the robot. Thus the control scheme, where the force signal forms a closed loop feedback, does not apply adequately. A modified damping control scheme of Fig. 3.14 is proposed and depicted in Fig. 3.15. The force signal F is multiplied by the desired admittance K_{F1} and gives the desired velocity $\dot{X}_d$ for the robot. It should be noticed that the closed loop of the force signal and the stiffness of the environment K_E are removed from the overall system. This makes the analysis of the overall system easier and more convenient. This method will be adapted for the knee surgery task and will be discussed in detail in later chapters.

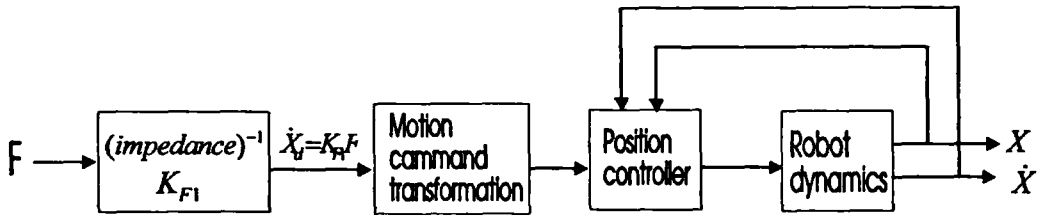


Fig. 3.15 A modified damping control scheme for guiding the robot by an external force

3.2.2.2 Stiffness control

In general, stiffness control can be divided into two categories: active stiffness control and passive stiffness control (implicit force control) depending on whether a force sensor is involved or not, respectively. Stiffness control is similar to damping control, but the contact force is converted to modify the desired position via a stiffness feedback matrix. Different kinds of control structure may be implemented. A conceptual stiffness control diagram can be shown as in Fig. 3.16, where K_{F2} is the compliance ($stiffness^{-1}$) matrix with dimensions position/force.

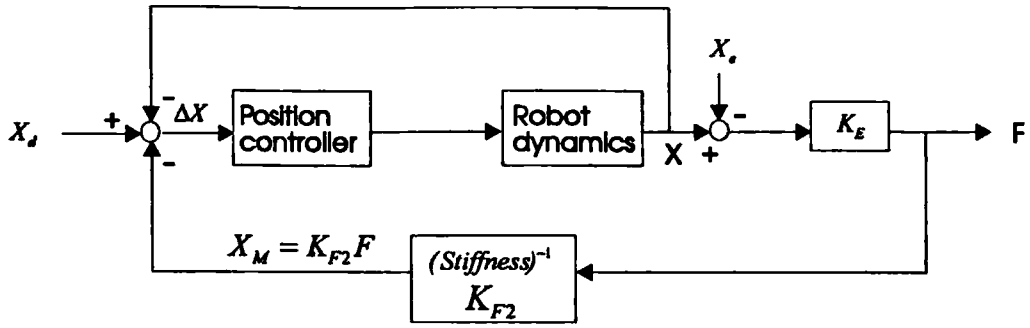


Fig. 3.16 Conceptual diagram of stiffness control scheme

3.2.2.3 Implicit force control scheme

An active force control uses a force sensor to sense the contact force and then modify the motion command. Another interesting approach is implicit force control [Whitney 87] or passive stiffness force control, where no force sensor is involved in the feedback loop. Figure 3.17 shows the implicit force control scheme implemented in joint space. The control structure of this approach is the same as the robot position control discussed in section 3.1.1. The difference is that the implicit force control is concerned with how to design the feedback gains Kp and Kd, so that the robot can behave with a desired stiffness.

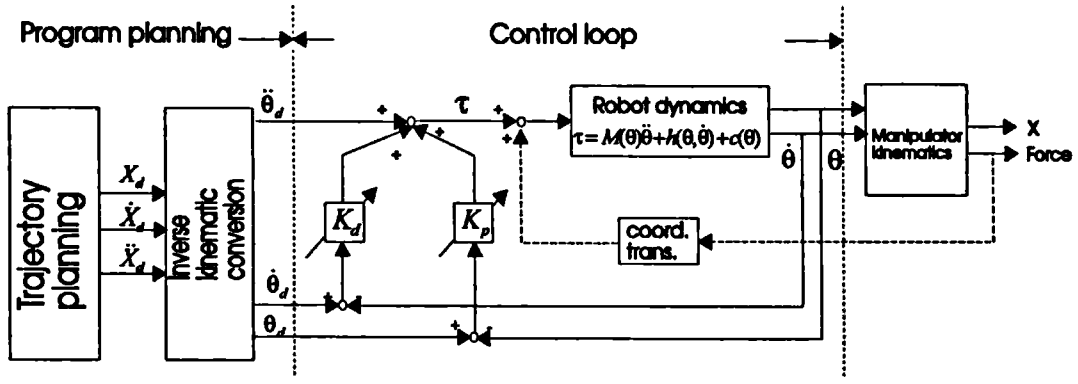


Fig. 3.17 Implicit force control scheme in joint space

One possible design concept is adopted from the active stiffness control design by [Salisbury 80]. Consider the definition of a Jacobian, which is given by

$$\delta X = J\delta\theta, \qquad (3.28)$$

where J is the Jacobian matrix; δX is the difference between the desired position and the actual position of the robot end-effector in Cartesian space; and $\delta\theta$ is the difference between the desired joint angles and the actual joint angles of the robot. An applied force F at the robot end-effector and the resultant joint torques T satisfy

$$T = J^T F. \qquad (3.29)$$

If the desired stiffness matrix in Cartesian space is equal to K_x, then the required force due to the position error δX is given by

$$F = K_x \delta X \qquad (3.30)$$

Substituting equation (3.30) and (3.28) into (3.29) yields

$$T = J^T K_x J\delta\theta. \qquad (3.31)$$

Equation (3.31) describes how the joint torques are needed for the joint angle errors $\delta\theta$, so that the robot can behave as a Cartesian spring with stiffness K_x. The term $K_\theta = J^T K_x J$ is called joint stiffness matrix by [Sailsbury 80]. However, it should be noted that equation (3.31) assumes the static and dynamic forces to the robot are directly compensated for, or small enough to be neglected.

3.3 Conclusion

Robot position control can be classified into two general categories: joint based and Cartesian based control. In general, the Cartesian based control scheme includes all the kinematics and transformation computations in the closed loop, which will decrease the sampling rate (given the same computer size) and thus degrade the stability and disturbance rejection ability of the system. The independent joint control scheme, which designs the controller for each joint and neglects the influence from other links, is utilized by nearly all commercial industrial robots. When the dynamic coupled effects are too significant to be ignored, the nonlinear dynamic decoupling

technique is usually used to decouple the effect of the non-linear terms, and linear control theory is then applied to analyze and design the controller.

Unlike the position control strategy, it is very difficult to simply divide the robot force control schemes into joint based and Cartesian based. This is because the error signal of the force feedback loop is usually calculated in Cartesian space, while the position control loop is processed in joint space. Thus, based on the design concept of the force controller and how the force error signal is processed, the force control scheme is divided into two general groups [Whitney 87]: hybrid position/force control and impedance control. The impedance control can be seen as a general scheme covering stiffness control and damping control.

In the next chapter, robot assisted systems in knee surgery and applications of industrial robots will be compared. Subsequently, control strategies with artificial motion constraint for robotic knee surgery will be presented and analyzed.

CHAPTER 4

FORCE CONTROL STRATEGY WITH AN ARTIFICIAL MOTION CONSTRAINT - ONE DOF SYSTEM

As described in chapter 2, in a new approach to knee joint replacement, the main aim of the knee surgery robot is to give various degrees of resistance/stiffness within a spatial region in order to guide the surgeon to make pre-planned cuts more easily and accurately. As in the concept of impedance control, the robot can be considered to be a variable spring-damper system. When the surgeon holds the cutter motor and moves it, he/she feels a different resistance force depending on the spring constant and the damping ratio, which varies according to the current position of the robot and the direction of motion. In extreme cases, if the cutter is moved outside the region , the spring constant and damping ratio become extremely high. From another point of view, the task is one of driving the robot actuators (motors) so that the robot simulates the function of a variable spring-damper system.

In general, there are some differences between the industrial robot applications and the tasks of this project:

(1). The motion of the robot in this task is 'passive', in the sense that the robot is used mainly to modify movement by the surgeon, although the robot is active in assisting (or constraining) the surgeon in moving along a planned trajectory or within a region. In industrial applications, however, the robot is generally commanded to achieve the desired trajectory or desired force autonomously and the external force is usually ignored or directly compensated for. The operator is not involved in the movement of the robot directly. In the approach of the I.C. project, the robot is moved manually by the surgeon. Thus the pushing/pulling force applied by the surgeon is the main contribution to the motion of the robot with a small additional component from

cutting tool forces. The main role of the robot is to modify and correct the movement made by the surgeon to follow the pre-planned trajectory or to prevent the robot from moving out of the pre-defined constraint.

(2). In industrial applications, the force control strategy is to maintain a desired contact force between the robot and the environment, whose location and geometry is usually fixed. The force concerned in this project is the 'resistance' force that the surgeon feels when he/she moves the robot. This resistance force varies in different zones of the planned motion constraint and in different directions. However, this is usually not a consideration in industrial robot applications.

(3). As seen in the previous chapter, the desired trajectory and velocity of the robot has to be designed in advance. For instance, when the robot is used to "paint" a rectangular region, the motion trajectory of the robot, (such as from up to down and from left to right,) has to be fixed and the painting speed also has to be decided in advance. But in the proposal for the robot assisted system in knee surgery, only the constraint of the region to be cut is defined. The surgeon can control the cutting procedures, (e.g. depth of cut and velocity) intra-operatively.

From the above discussion, three main points can be concluded. First, the guiding force applied by the surgeon cannot be ignored and should be taken into account in the closed loop system. Second, the robot control strategy has to be a function of the current position of the robot and the direction of motion, thus the robot can behave with variable stiffness as required. Finally, an on-line trajectory "interpreting system" for the robot will be needed to assist or constrain the movement by the surgeon in a pre-planned trajectory or region.

The next section continues with the design and analysis of a one dimensional system. The concept of an artificial motion constraint will be elucidated. Implicit force control and modified damping control strategies are also analyzed in detail. Moreover, experimental tests will be implemented to examine the feasibility and performance of the analysis.

4.1 Theoretical analysis

Fig. 4.1 shows a schematic diagram of a one degree of freedom system. Where Fs is the pulling/pushing force of the surgeon, Fr is the output force of the actuator of the robot and Fc is a milling tool cutting force. The dynamic equation of the system is given by

$$M\ddot{X} = F_r + F_s - F_c, \quad (4.1)$$

where M is the mass and X is the position of the system. The analysis of a milling cutting process is quite complicated , since the magnitude of the cutting force depends on the machining conditions such as the number of cutter teeth, the length of arc of cut, and the cutting speed etc. [Andrew 62, Koenigsberger 61, Sabberwal 62]. To simplify the model, we assume the average cutting force satisfies

$$F_c = K_c\dot{X}, \quad (4.2)$$

where Kc is the cutting force ratio with the dimension force/velocity and $\dot{X}$ is the cutting speed which is equal to the velocity of the system. Substituting equation (4.2) into equation (4.1), the system dynamic equation yields

$$M\ddot{X} + K_c\dot{X} = F_r + F_s. \quad (4.3)$$

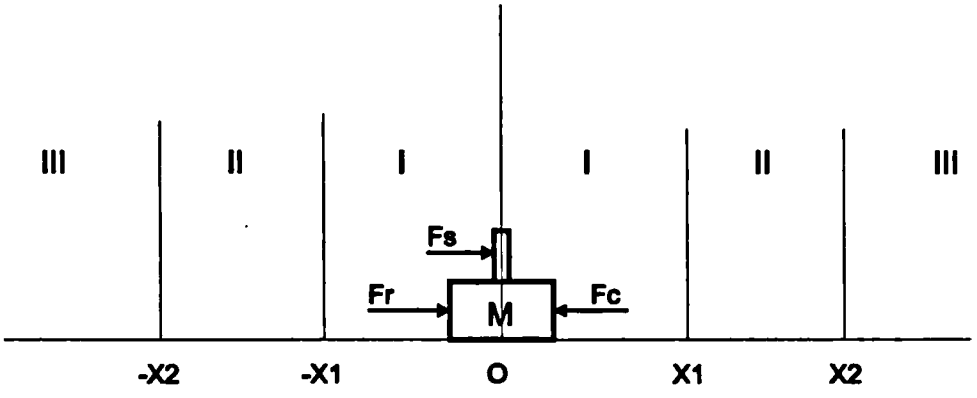


Fig. 4.1. Schematic force control system for a single degree of freedom

4.1.1 Implicit force control with an artificial motion constraint

It is seen from equation (4.3) that the function of the actuator output Fr is to regulate the external force Fs applied by the surgeon, thus the movement of the system can behave in the desired way. As described in chapter 2, the force control strategy should achieve the following objectives : (i) Inside zone I (see Fig 4.1) , the robot can be freely moved by the surgeon. (ii) While in zone II, if the robot is moved towards the outer boundary (such as from position X1 to X2), the surgeon will feel a steadily increased resistance. Thus the surgeon will be warned that he is approaching the boundary. At the outer boundary (for instance at position X2), the robot can only move back into the interior region and the resistance to go back to a central area is less than that to go forwards. (iii) Zone III is a 'no go' area.

Assuming the force sensor is not used, i.e. the external force Fs is unknown, an implicit force control (variable PD control) law is expressed as

$$F_r = K_p(X_d - X) - K_d\dot{X}, \qquad (4.4)$$

where Kp is the proportional gain, Kd is the derivative gain, Xd is the reference position and X is the system current position. In this control law, feedback gains Kp and Kd represent a spring constant and damping ratio respectively. Thus, by changing the control gains Kp and Kd, the surgeon will be able to feel a different stiffness when he back drives the robot. In order to constrain the system motion, an on-line trajectory interpreting algorithm is also needed to continuously evaluate the desired position Xd for the system.

The split rate control algorithm [Mapples 86] is used to implement the algorithm for the system, which divides the control scheme into two computation loops. The inner loop implements the position control algorithm (equation 4.4) with a faster sampling frequency, and another computation loop updates the desired position Xd and the feedback gains Kp and Kd for the inner loop control law with a slower computation interval. There are two major advantages of this technique. First, the inner position

control law for the system is separated from the trajectory decision and the update computation for the feedback gains. Therefore, it is very easy to implement the control scheme. Secondly, the computation of the inner closed loop is very simple, thus a high sampling rate of the closed loop can be attained, which will increase the stability of the system and the disturbance rejection ability. Suppose the sampling rate of the inner loop control is ten times higher than the natural frequency of the system, and the continuous time analysis is valid [Craig 86]. The control diagram for the one DOF system can be shown as Fig. 4.2, where Δt is the computation interval of the decision algorithm for the desired position and feedback gains, and S is the Laplace operator.

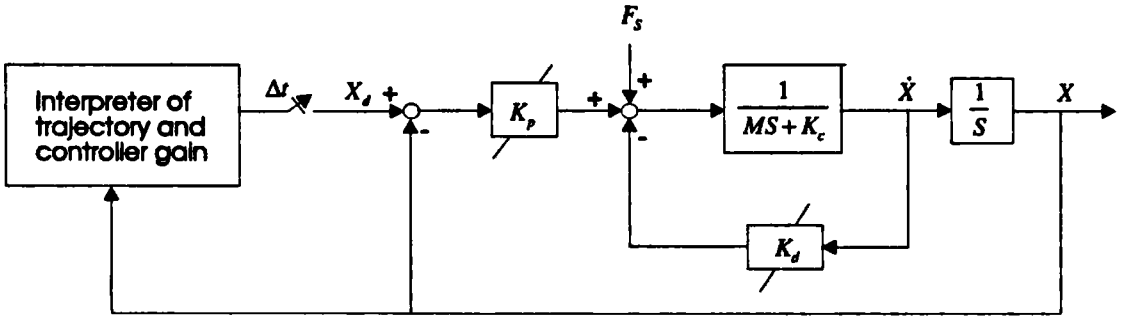


Fig. 4.2 Implicit force control diagram for one DOF system

Substituting the control law (equation (4.4)) into equation (4.3), the closed loop system within every computation interval Δt is expressed by

$$M\ddot{X} + (K_d + K_c)\dot{X} + K_p X = F_s + K_p X_d \qquad (4.5)$$

It is important that a surgeon should feel a different resistance force when he moves the robot to cut the bone. Suppose the hardness of the bone is constant, i.e. Kc is constant, when a steady force Fs is applied by the surgeon, he will feel different resistance depending on the magnitude of the final cutting velocity V_f. In other words, the overall system can be seen as a dashpot and the magnitude of Fs/V_f can be taken as the stiffness index. Therefore, an analysis will be presented of the way in which the feedback gains Kp and Kd affect the final cutting speed when an external force is applied.

Taking the Laplace transformation of equation (4.5) yields

$$M[S^2X(S)-SX_0-\dot{X}_0]+(K_d+K_c)(SX(S)-X_0)+K_pX(S)=F_s+K_pX_d, \quad (4.6)$$

where X_0 and $\dot{X}_0$ are the initial position and velocity of the system at the start of every computation cycle Δt. If equation (4.6) is rearranged, the system Laplace transfer function of X(t) is given by

$$X(S)=\frac{F_s}{MS^2+(K_c+K_d)S+K_p}+\frac{K_pX_d}{MS^2+(K_c+K_d)S+K_p} + \frac{[MS+(K_c+K_d)]X_0+\dot{X}_0}{MS^2+(K_c+K_d)S+K_p}, \quad (4.7)$$

where the first term is the system response due to the guiding force Fs; the second term is the effect due to the desired position Xd, which is used to modify the movement by Fs; and the final term is due to the initial conditions X_0 and $\dot{X}_0$. The Laplace transfer function of the velocity response $\dot{X}(t)$ equals SX(S) and is equal to

$$\dot{X}(S)=\frac{SF_s}{MS^2+(K_c+K_d)S+K_p}+\frac{SK_pX_d}{MS^2+(K_c+K_d)S+K_p} + S\frac{[MS+(K_c+K_d)]X_0+\dot{X}_0}{MS^2+(K_c+K_d)S+K_p}. \quad (4.8)$$

<u>Case I :</u> when the robot is moved within zone I and II (see Fig. 4.1).

The movement direction of the robot is dependent on the pushing/pulling force. There is no need to constrain the motion of the robot. Thus the desired position Xd is set as the current position at the end of every outer computation cycle Δt (i.e. $X_d = X(N\Delta t)$) from time $N\Delta t$ to $(N+1)\Delta t$.

Two situations will be examined. Firstly, when the external force Fs equals 0 but the initial velocity $\dot{X}_0$ is not zero at a particular moment, for example if the surgeon moves the robot and then suddenly releases his/her force, it is important to know how soon the robot can be stopped and how far the robot will go before it is stopped. Thus, the robot will not perform any undesired cut. Secondly, if the initial velocity $\dot{X}_0 = 0$

but $F_S \neq 0$, the relation between the system velocity and the feedback gains will also be analyzed. It should be noticed that the desired position and the control gains are updated after every outer sampling time Δt, thus the final-value theorem [Kuo] is not valid to analyze the closed loop system. Therefore, the change of the movement and velocity of the robot will be derived between every sampling interval Δt as can be seen below. The equilibrium state values are then estimated by integrating these changes.

(1) when $\dot{X}_0 = V_0$ and F = 0.

Using equation (4.7) and replacing X_d with X_0 yields the following expression for X(S)

$$X(S) = \frac{X_0}{S} + \frac{MV_0}{MS^2 + (K_c + K_d)S + K_p} \qquad (4.9)$$

Taking the inverse Laplace transformation of (4.9), there are three solutions depending on the value of $(K_c + K_d)^2 - 4MK_p$.

(i) when $(K_c + K_d)^2 - 4MK_p < 0$, (i.e. the proportional gain dominates the stiffness of the system)

$$X(t) = X_0 + \frac{V_0}{b}(e^{-at} \sin bt), \qquad (4.10)$$

where $a = \frac{K_c + K_d}{2M}$ and $b = \frac{[4MK_p - (K_c + K_d)^2]^{1/2}}{2M}$. Taking the derivative of X(t), the velocity $\dot{X}(t)$ is obtained by

$$\dot{X}(t) = V_0[e^{-at}(\cos bt - \frac{1}{2ab}\sin bt)]. \qquad (4.11)$$

It is seen from (4.10) and (4.11) that after every computation interval Δt, the robot moves

$$\begin{aligned} \Delta X(\Delta t) &= X(\Delta t) - X_0 \\ &= \frac{V_0}{b}(e^{-a\Delta t} \sin b\Delta t) \end{aligned} \qquad (4.12)$$

and the velocity reduction ratio is

$$\frac{\dot{X}(\Delta t)}{V_0} = e^{-a\Delta t}(\cos b\Delta t - \frac{1}{2ab}\sin b\Delta t). \qquad (4.13)$$

The extreme case is when both K_c and K_d are equal to zero. This causes a = 0 and $b = \sqrt{\frac{K_p}{M}}$. If the feedback gain remains constant, equations (4.12) and (4.13) become

$$\Delta X = \frac{V_0}{\sqrt{K_p/M}} sin(\sqrt{\frac{K_p}{M}}\Delta t), \qquad (4.14)$$

and

$$\frac{\dot{X}(\Delta t)}{V_0} = \cos(\sqrt{\frac{K_p}{M}}\Delta t). \qquad (4.15)$$

Therefore, if $0 < \sqrt{\frac{K_p}{M}}\Delta t < \frac{\pi}{2}$, the system velocity will be decreased within the time interval Δt. The motion distance before the robot stops can be obtained by

$$S = \sum_{n=1}^{n=\infty} \Delta X(n\Delta t), \qquad (4.16)$$

where $\Delta X(n\Delta t)$ is the movement between time (n-1)Δt and nΔt. Substituting (4.14) into (4.16), the movement distance S is

$$S = \sum_{n=1}^{n=\infty} \Delta X(n\Delta t) = \sum_{n=0}^{n=\infty} \frac{1}{\sqrt{K_p/M}} \sin(\sqrt{\frac{K_p}{M}}\Delta t)\dot{X}(n\Delta t) \qquad (4.17)$$

where $\dot{X}(n\Delta t)$ is the initial velocity at time nΔt. By using (4.15), equation (4.17) becomes

$$S = \sum_{n=1}^{n=\infty} \Delta X(n\Delta t)$$

$$= \frac{1}{\sqrt{K_p/M}} \sin(\sqrt{\frac{K_p}{M}}\Delta t) \sum_{n=0}^{n=\infty} \dot{X}(n\Delta t)$$

$$= \frac{1}{\sqrt{K_p/M}} \sin(\sqrt{\frac{K_p}{M}}\Delta t)\{V_0[1+\cos(\sqrt{\frac{K_p}{M}}\Delta t)+\cos^2(\sqrt{\frac{K_p}{M}}\Delta t)+.....]\}$$

$$= \frac{1}{\sqrt{K_p/M}} \sin(\sqrt{\frac{K_p}{M}}\Delta t) V_0 \frac{\lim_{n\to\infty}[1-\cos^n(\sqrt{\frac{K_p}{M}}\Delta t)]}{[1-\cos(\sqrt{\frac{K_p}{M}}\Delta t)]}$$

$$= \frac{V_0}{\sqrt{K_p/M}} \frac{\sin(\sqrt{\frac{K_p}{M}}\Delta t)}{[1-\cos(\sqrt{\frac{K_p}{M}}\Delta t)]} \qquad (4.18)$$

(ii) when $(K_c + K_d)^2 - 4MK_p > 0$

Taking the inverse Laplace transformation of (4.9) yields

$$X(t) = X_0 + \frac{V_0}{2b^*}[e^{-(a-b^*)t} - e^{-(a+b^*)t}], \qquad (4.19)$$

where $a = \frac{K_c + K_d}{2M}$ and $b^* = \frac{[(K_c + K_d)^2 - 4MK_p]^{1/2}}{2M}$. The velocity can be obtained by taking the derivative of (4.19)

$$\dot{X}(t) = \frac{V_0}{2b^*}[(a+b^*)e^{-(a+b^*)t} - (a-b^*)e^{-(a-b^*)t}]. \qquad (4.20)$$

When $K_p = 0$, both a and b^* are equal to $\frac{(K_c + K_d)}{2M}$. Equation (4.19) and (4.20), respectively, become

$$X(t) = X_0 + \frac{MV_0}{K_c + K_d}[1 - e^{-\frac{(K_c+K_d)t}{M}}] \qquad (4.21)$$

and

$$\dot{X}(t) = V_0 e^{-\frac{(K_c+K_d)t}{M}}. \qquad (4.22)$$

The motion distance before the robot is stopped can be obtained in a similar way to that of the analysis procedure in (i), i.e.:

$$
\begin{aligned}
S &= \sum_{n=1}^{n=\infty} \Delta X(n\Delta t) \\
&= \sum_{n=0}^{n=\infty} \frac{MV_0}{K_c + K_d}(1 - e^{-\frac{(K_c+K_d)}{M}\Delta t})\dot{X}(n\Delta t) \\
&= \frac{M}{K_c + K_d}(1 - e^{-\frac{(K_c+K_d)}{M}\Delta t}) \sum_{n=0}^{n=\infty} \dot{X}(n\Delta t) \\
&= \frac{M}{K_c + K_d}(1 - e^{-\frac{(K_c+K_d)}{M}\Delta t}) V_0[1 + e^{-\frac{(K_c+K_d)}{M}\Delta t} + (e^{-\frac{(K_c+K_d)}{M}\Delta t})^2 + ...] \\
&= \frac{MV_0}{K_c + K_d} \frac{(1 - e^{-\frac{(K_c+K_d)}{M}\Delta t})}{(1 - e^{-\frac{(K_c+K_d)}{M}\Delta t})} \\
&= \frac{MV_0}{K_c + K_d}.
\end{aligned}
\qquad (4.23)
$$

(iii) when $(K_c + K_d)^2 - 4MK_p = 0$

Taking the inverse Laplace transformation of (4.9) yields

$$X(t) = X_0 + V_0 t e^{-at}, \qquad (4.24)$$

where $a = \frac{K_c + K_d}{2M}$. The velocity can be obtained by taking the derivative of (4.24)

$$\dot{X}(t) = V_0(1 - at)e^{-at}. \quad (4.25)$$

The motion distance before the robot is stopped can be obtained in a similar way to that of the analysis procedure in (i), i.e.:

$$
\begin{aligned}
S &= \sum_{n=1}^{n=\infty} \Delta X(n\Delta t) \\
&= \sum_{n=0}^{n=\infty} \Delta t e^{-a\Delta t} \dot{X}(n\Delta t) \\
&= \Delta t e^{-a\Delta t} V_0 [1 + (1 - a\Delta t) e^{-a\Delta t} + [(1 - a\Delta t) e^{-a\Delta t}]^2 + \ldots..] \\
&= \frac{V_0 \Delta t e^{-a\Delta t}}{1 - e^{-a\Delta t} + a\Delta t e^{-a\Delta t}}.
\end{aligned}
\tag{4.26}
$$

From (4.18), (4.23) and (4.26), it is seen that the allowed movement of the robot, due to an initial velocity when the external force is zero, can be restricted by changing the values of the proportional gain Kp or the derivative gain Kd. An increase in value of the derivative gain Kd will increase the system damping ratio. At first glance, the proportional feedback Kp is like a spring which stores the kinematic energy. However, the desired position after every computation cycle Δt is updated as the system current position, which releases the stored energy. Thus the robot can also behave as a dashpot as a result of the proportional feedback.

(2). when $\dot{X}_0 = 0$ and F_s = constant (i.e. $F_s(S) = F_s / S$)

The way in which the feedback affects the steady state velocity when an external force is applied will be analyzed. Using equation (4.7), X(S) is given as

$$
X(S) = \frac{X_0}{S} + \frac{F_s / S}{MS^2 + (K_c + K_d)S + K_p} \tag{4.27}
$$

Taking the inverse Laplace transformation of (4.27), there are three solutions depending on whether $[(K_c + K_d)^2 - 4MK_p.]$ is positive, negative or zero.

(i) when $(K_c + K_d)^2 - 4MK_p < 0$

$$
X(t) = X_0 + \frac{F_s}{K_p}[1 - e^{-at}(\cos bt + \frac{a}{b}\sin bt)], \tag{4.28}
$$

where $a = \frac{K_c + K_d}{2M}$ and $b = \frac{[4MK_p - (K_c + K_d)^2]^{1/2}}{2M}$. The velocity can be obtained by taking the derivative of X(t)

$$\dot{X}(t) = \frac{dX(t)}{dt}$$

$$= \frac{F_s}{K_p}[ae^{-at}(\cos bt + \frac{a}{b}\sin bt) - e^{-at}(-b\sin bt + a\cos bt)]$$

$$= \frac{F_s}{K_p}e^{-at}\sin bt[\frac{a^2+b^2}{b}]$$

$$= \frac{F_s}{K_p}e^{-at}\sin bt[\frac{K_p}{Mb}]$$

$$= \frac{F_s}{Mb}e^{-at}\sin bt. \qquad (4.29)$$

So, within Δt interval, the velocity due to the force F_S increases

$$(\Delta\dot{X})_F = \dot{X}(\Delta t)$$

$$= \frac{F_s}{Mb}e^{-a\Delta t}\sin b\Delta t. \qquad (4.30)$$

In order to examine the effect of the proportional gain Kp, it is assumed that $K_c = 0$ and $K_d = 0$ (i.e. a = 0 and $b = \sqrt{\frac{K_p}{M}}$), the equation (4.30) becomes

$$(\Delta\dot{X})_F = \frac{F_s}{\sqrt{K_p/M}}\sin(\sqrt{\frac{K_p}{M}}\Delta t). \qquad (4.31)$$

The velocity $\dot{X}(n\Delta t)$ at time nΔt can be derived by adding the transient response $[\dot{X}(n\Delta t)]_i$ (which is equal to $\dot{X}[(n-1)\Delta t)]\cos(\sqrt{\frac{K_p}{M}}\Delta t)$ from equation (4.15)) to the force response $(\Delta\dot{X})_F$,

$$\dot{X}(n\Delta t) = (\Delta\dot{X})_F + [\dot{X}(n\Delta t)]_i$$

$$= (\Delta\dot{X})_F + \dot{X}[(n-1)\Delta t)]\cos(\sqrt{\frac{K_p}{M}}\Delta t)$$

$$= (\Delta\dot{X})_F + \cos(\sqrt{\frac{K_p}{M}}\Delta t)\{(\Delta\dot{X})_F + \dot{X}[(n-2)\Delta t)]\cos(\sqrt{\frac{K_p}{M}}\Delta t)\}$$

$$= (\Delta\dot{X})_F[1 + \cos(\sqrt{\frac{K_p}{M}}\Delta t) + \cos^2(\sqrt{\frac{K_p}{M}}\Delta t) + ... + \cos^{n-1}(\sqrt{\frac{K_p}{M}}\Delta t)]$$

$$= (\Delta\dot{X})_F[\frac{1-\cos^n(\sqrt{\frac{K_p}{M}}\Delta t)}{(1-\cos(\sqrt{\frac{K_p}{M}}\Delta t))}]. \qquad (4.32)$$

Combining (4.31) and (4.32), the steady state velocity V_f is given as

$$V_f = \lim_{n\to\infty}\dot{X}(n\Delta t) = \frac{(\Delta\dot{X})_F}{[1-\cos(\sqrt{\frac{K_p}{M}}\Delta t)]}$$

$$= \frac{F_s}{\sqrt{K_p/M}}\frac{\sin(\sqrt{\frac{K_p}{M}}\Delta t)}{[1-\cos(\sqrt{\frac{K_p}{M}}\Delta t)]}. \qquad (4.33)$$

When Comparing (4.33) with (4.18), it can be seen that both equations have a similar format, although the steady state velocity in (4.33) is due to external forces, whereas the movement of the robot in (4.18) is caused by the initial velocity. When the proportional gain Kp is increased, the final velocity decreases. Thus, the stiffness of the system can be designed by varying the value of Kp.

(ii) when $(K_c + K_d)^2 - 4MK_p > 0$

Taking the inverse Laplace transformation of (4.27) yields

$$X(t) = X_0 + \frac{F_s}{M}[\frac{M}{K_p} - \frac{1}{2b^*(a-b^*)}e^{-(a-b^*)t} + \frac{1}{2b^*(a+b^*)}e^{-(a+b^*)t}], \quad (4.34)$$

where $a = \frac{K_c + K_d}{2M}$ and $b^* = \frac{[(K_c + K_d)^2 - 4MK_p]^{1/2}}{2M}$. By taking the derivative of X(t), the velocity is

$$\dot{X}(t)=\frac{F_s}{2Mb^*}[e^{-(a-b^*)t}-e^{-(a+b^*)t}]. \qquad (4.35)$$

When K_p is equal to zero (i.e. $a=b^*=\frac{(K_c+K_d)}{2M}$) within Δt interval, the velocity due to the force F_S increases by

$$\begin{aligned}(\Delta\dot{X})_F &= \dot{X}(\Delta t)\\ &=\frac{F_s}{K_c+K_d}[1-e^{-\frac{K_c+K_d}{M}\Delta t}],\end{aligned} \qquad (4.36)$$

When the initial velocity is not zero, the velocity after every Δt increment is

$$\dot{X}(\Delta t)=(\Delta\dot{X})_F+[\dot{X}(\Delta t)]_i, \quad (4.37)$$

where $[\dot{X}(\Delta t)]_i$ is the transient response due to the initial velocity. From (4.22), $[\dot{X}(\Delta t)]_i$is equal to

$$[\dot{X}(\Delta t)]_i=\dot{X}_0e^{-\frac{(K_c+K_d)}{M}\Delta t}, \quad (4.38)$$

The velocity at time nΔt is then given by

$$\begin{aligned}\dot{X}(n\Delta t)&=(\Delta\dot{X})_F+\dot{X}[(n-1)\Delta t)]e^{-\frac{(K_c+K_d)}{M}\Delta t}\\ &=(\Delta\dot{X})_F+e^{-\frac{(K_c+K_d)}{M}\Delta t}\{(\Delta\dot{X})_F+\dot{X}[(n-2)\Delta t)]e^{-\frac{(K_c+K_d)}{M}\Delta t}\}\\ &=(\Delta\dot{X})_F[1+e^{-\frac{(K_c+K_d)}{M}\Delta t}+(e^{-\frac{(K_c+K_d)}{M}\Delta t})^2+.....+(e^{-\frac{(K_c+K_d)}{M}\Delta t})^{n-1}]\\ &=(\Delta\dot{X})_F[\frac{1-(e^{-\frac{(K_c+K_d)}{M}\Delta t})^n}{(1-e^{-\frac{(K_c+K_d)}{M}\Delta t})}].\end{aligned} \qquad (4.39)$$

The steady state velocity V_f is given as

$$\begin{aligned}V_f&=\lim_{n\to\infty}\dot{X}(n\Delta t)=\frac{(\Delta\dot{X})_F}{(1-e^{-\frac{K_c+K_d}{M}\Delta t})}\\ &=\frac{F_s}{K_c+K_d},\end{aligned} \qquad (4.40)$$

(iii) when $(K_c + K_d)^2 - 4MK_p = 0$

Taking the inverse Laplace transformation of (4.27) yields

$$X(t) = X_0 + \frac{F_s}{Ma^2}[1-(1+at)e^{-at}], \quad (4.41)$$

where $a = \frac{K_c + K_d}{2M}$. By taking the derivative of X(t), the velocity is

$$\dot{X}(t) = \frac{F_s}{M} te^{-at}. \quad (4.42)$$

The steady state velocity V_f can be obtained in a similar way to that of the analysis procedure in (i) and (ii), and is given as

$$\begin{aligned} V_f &= \lim_{n\to\infty} \dot{X}(n\Delta t) \\ &= \frac{F_s}{M} \frac{\Delta t e^{-a\Delta t}}{(1-e^{-a\Delta t} + a\Delta t e^{-a\Delta t})}. \end{aligned} \quad (4.43)$$

<u>Case II</u> : when the robot is moved toward zone III.

Assume the initial position of the robot X_0 is at the boundary of zone II, for instance $X_0 = X_2$ (see Fig. 4.1). When the robot is moved toward zone III, the function of the controller is then to constrain the robot to stay in zone II. Thus the desired position X_d is always set as X_2. If the initial velocity equals to zero and the external force is constant, the system transfer function is found to be

$$X(S) = \frac{X_2}{S} + \frac{F_s / S}{MS^2 + (K_c + K_d)S + K_p} \quad (4.44)$$

Since the desired position and the control gains are not changed when the robot is in zone III, the steady state position X_s can be derived by the final-value theorem

$$\begin{aligned} X_s &= \lim_{S\to 0} SX(S) \\ &= X_2 + \frac{F_s}{K_p}. \end{aligned} \quad (4.45)$$

Therefore, the maximum position error (distance to zone II) due to the external force Fs is

$$X_{error} = X_s - X_2$$
$$= \frac{F_s}{K_p}. \qquad (4.46)$$

Note that the position error in zone III can be eliminated by adding an integral feedback, however, this may cause severe transient oscillations due to the integrator windup (or reset windup) problem [Franklin 86]. One possible solution is to clamp the integrator output in order to limit values to a within the selected areas.

Recall that the transient response can be analyzed by the characteristic equation

$$MS^2 + (K_c + K_d)S + K_p = 0,$$

or $$S^2 + 2\zeta\omega_n S + \omega_n^2 = 0, \qquad (4.47)$$

where ζ = system damping ratio and ω_n = natural frequency, and the parameters are in relation to one another as follows:

$$2\zeta\omega_n = \frac{K_c + K_d}{M} \qquad (4.48)$$

$$\omega_n^2 = \frac{K_p}{M} \qquad (4.49)$$

To have a critically damped or an overdamped system, the damping ratio has to be greater than or equal to unity, i.e.

$$\zeta = \frac{K_c + K_d}{2\sqrt{MK_p}} \geq 1,$$

or $$K_d \geq 2\sqrt{MK_p} - K_c \qquad (4.50)$$

4.1.1.1 Discussion

(1). In zones I and II, the movement direction of the robot is subject to the surgeon's decision. Thus the desired position for the control law is updated as to the current position of the robot after every computation cycle. The function of the robot is to provide variable stiffness according to its position and direction of motion. This function can be reached by varying either the proportional gain Kp or the derivative feedback gain Kd. Using the proportional feedback can have a similar damping effect to the derivative feedback, but it should be noticed that the response is related to the updating rate of the desired position (see equation (4.33)).

(2). In zone III, the control is switched to high gain position control. Integral feedback can be added to eliminate position error. When the robot is pushed back to zone II by the robot actuator, the potential energy of the robot caused by the proportional gain is absorbed. It will therefore not oscillate, because the desired position is switched to the current position of the robot. Hence the control does not suffer from the conventional force control instability problems. Seen differently, the robot system is like a spring and damper system in zone III, while in zone I and II, it behaves as a variable damper system. The feedback gains can be designed in different ways to achieve the specific tasks and performance can be predicted from the above analysis. Fig. 4.3 shows a general design scheme for the control gains. The value of the derivative gain Kd increases steadily from zone I to zone II. The forward value is, however, higher than the backward value. Thus, the robot can be moved back to zone I more easily. In zones I and II, the proportional gain is set as a small constant rather than at zero, and this reduces the effect of the signal noise, such as the noise from the computation for the velocity.

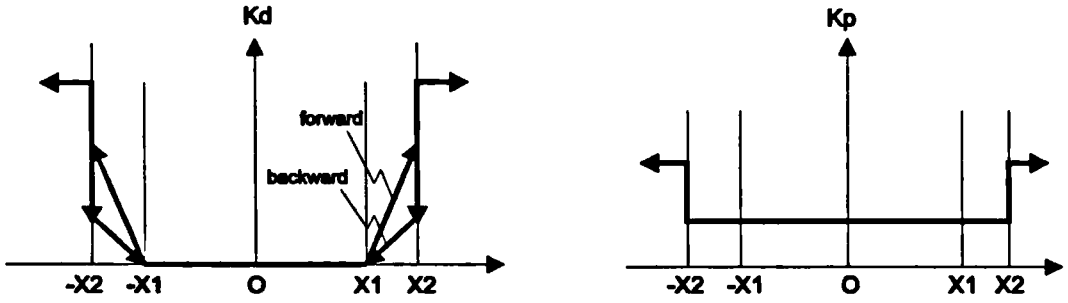


Fig. 4.3. Design of the control gains for one DOF system

In order to obtain an artificial motion constraint, the decision algorithm for the desired position after every computation cycle Δt is given by

$X_d(N) = X(N)$ if X(N) is in zone I or II

$X_d(N) = X_2$ or $-X_2$ if X(N) is in zone III,

where X(N) is the current position of the robot at time $N\Delta t$.

(3).It is desirable to increase the maximum feedback gain in zone III as high as possible, so the inevitable disturbances can be quickly suppressed or the steady state error can be reduced. However, the maximum value is limited by other considerations. For example, the robot mechanism is not absolutely rigid. Thus it is recommended that the natural frequency of the overall system should be less than half the smallest resonance frequency ω_r of the mechanism to avoid stimulating resonance, i.e. $\omega_n \leq 0.5\omega_r$. For instance, the value of Kp by in a one DOF system is limited by

$$K_p \leq \frac{1}{4} M\omega_r^2. \qquad (4.51)$$

Furthermore, the maximum output of the actuator has to be taken into account in order to avoid saturation.

(4). The inner control law is very easy to implement. Using the split control rate algorithm will give the inner closed loop a high sampling frequency, which will increase the stability of the system and bandwidth limitation of the overall system. The updating rate of the desired position and the control gains is not crucial to the performance or the stability of the system. However, it will affect the 'smoothness' of

operation. To the extent that the updating rate is slow, the change in resistance force will have a 'jump' which will cause the movement to become jerky in operation.

4.1.2 Modified damping control with an artificial motion constraint

In the previous section, the implicit force control method which does not involve a force sensor was analyzed. However, when the robot is very difficult to move manually because of the type of mechanism (e.g. a ball bearing screw drive system) used, it becomes necessary to use force sensors to sense the guiding force and program the robot to then follow the force command (other advantages of using a force sensor will be discussed in a detail in the next chapter). If the guiding force of the operator is equal to F_S, the motion command can be modelled as

$$\dot{X}_d = K_f F_s, \qquad (4.52)$$

where $\dot{X}_d$ is the desired velocity and K_f is the desired admittance. It should be noticed that in industrial application, the contact force between the robot and the environment is usually modelled as a function of the robot position,

$$F = K_E(X - X_E), \qquad (4.53)$$

where F is the contact force; K_E is the stiffness of the end-effector and environment; X is the robot current position and X_E is the location of the environment. However, in our application, the guiding force cannot be simply modelled as (4.53), since the position of the surgeon's hand X_E is not predictable. Therefore, the guiding force will be generally assumed to be independent (this will be discussed in a detail in chapter 5).

Equation (4.52) is similar to the damping control method [Whitney 87] in that the external force is modelled as the velocity command. Thus, the proposed control algorithm will be called modified damping control. The control law is now given by

$$F_r = K_p(X_d - X) + K_d(\dot{X}_d - \dot{X}) \qquad (4.54)$$

Figure 4.4 shows the block diagram of the modified damping control. Compared with Fig. 4.2 of the implicit force control, the desired velocity $\dot{X}_d$ which is decided by the guiding force F_S and the force feedback gain K_f is now included in the control loop.

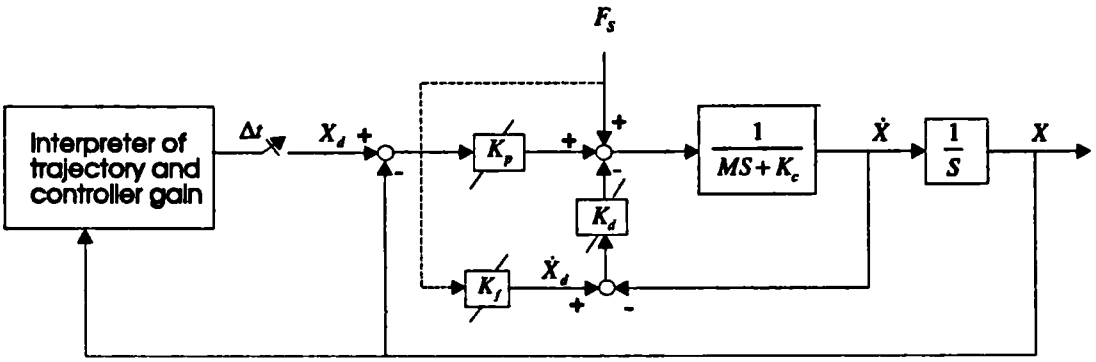


Fig. 4.4 Modified damping control with an artificial motion constraint scheme for an one DOF system

The closed loop transfer function between every computation cycle Δt can be expressed as

$$X(S)=\frac{F_s}{MS^2+(K_c+K_d)S+K_p}+\frac{K_dK_fF_s}{MS^2+(K_c+K_d)S+K_p}$$
$$+\frac{K_pX_d}{MS^2+(K_c+K_d)S+K_p}+\frac{[MS+(K_c+K_d)]X_0+\dot{X}_0}{MS^2+(K_c+K_d)S+K_p}$$
$$=\frac{(1+K_dK_f)F_s}{MS^2+(K_c+K_d)S+K_p}+\frac{K_pX_d}{MS^2+(K_c+K_d)S+K_p}+\frac{[MS+(K_c+K_d)]X_0+\dot{X}_0}{MS^2+(K_c+K_d)S+K_p}. \quad (4.55)$$

Compared with the system transfer function (4.7) in the implicit force control scheme, the only difference is that the response $\frac{K_dK_fF_s}{MS^2+(K_c+K_d)S+K_p}$ due to the motion command is added into the equation (4.55). The $K_dK_fF_s$ term can be seen as the amplifying factor of the guiding force F_s. This command to the actuators will help the operator to guide the robot. Suppose the task for the modified damping control algorithm is the same as that for the implicit force controller (see Fig. 4.1) , i.e. the motion constraint region is divided into three zones, (1) zone I : free moving area , (2)

zone II : variable resistance force area, and (3) zone III : 'no go' area. In order to perform this task successfully, both the desired position and the feedback gains (Kp, Kd and K_f) have to be designed for the control law. The decision algorithm for the desired position is dependent on the definition of the artificial motion constraint, and is the same as the analysis in the last section. Therefore, the main problem is how to design the feedback gains to have desired system stiffness. Assume the system initial velocity is zero and the guiding force F_s is constant,

Case I : when the robot is moved in zone I and II. Similar to the analysis in the previous section, the steady state velocity V_f is given by

(i) when $(K_c + K_d)^2 - 4MK_p < 0$

$$V_f = \frac{(1 + K_d K_f) F_s e^{-a\Delta t} \sin b\Delta t}{Mb[1 - e^{-\frac{a}{2}\Delta t}(\cos b\Delta t - \frac{a}{2b}\sin b\Delta t)]}, \quad (4.56)$$

where $a = \frac{K_c + K_d}{2M}$, $b = \frac{[4MK_p - (K_c + K_d)^2]^{1/2}}{2M}$ and Δt is the updating interval of the desired position and velocity. When $K_c = 0$ and $K_d = 0$, it yields

$$V_f = \frac{F_s}{\sqrt{K_p / M}} \frac{\sin(\sqrt{\frac{K_p}{M}}\Delta t)}{[1 - \cos(\sqrt{\frac{K_p}{M}}\Delta t)]} \quad (4.57)$$

(ii) when $(K_c + K_d)^2 - 4MK_p > 0$

$$V_f = \frac{(1 + K_d K_f) F_s [e^{-(a-b^*)\Delta t} - e^{-(a+b^*)\Delta t}]}{2Mb^*[1 - \frac{1}{2b^*}[(a+b^*)e^{-(a+b^*)\Delta t} - (a-b^*)e^{-(a-b^*)\Delta t}]]}, \quad (4.58)$$

where $b^* = \frac{[(K_c + K_d)^2 - 4MK_p]^{1/2}}{2M}$. When Kp = 0 , it gets

$$V_f = \frac{(1 + K_d K_f) F_s}{K_c + K_d}, \quad (4.59)$$

(ii) when $(K_c + K_d)^2 - 4MK_p = 0$

$$V_f = \frac{(1 + K_d K_f) F_s \Delta t e^{-a\Delta t}}{M(1 - e^{-a\Delta t} + a\Delta t e^{-a\Delta t})}., \quad (4.60)$$

where $a = \frac{K_c + K_d}{2M}$.

In general, as the feedback gain Kd or Kp increases, the final velocity due to a constant force is decreased. In order to get a different stiffness, one method is to set the control gains Kp and Kd as constant and change the value of K_f. However, the higher the value of $K_d K_f$ (see equation (4.55)), the more sensitive the system to the noise of the force signal and the less stable the system will become. Therefore, one design method is to keep the value of $K_d K_f$ a constant, which is used to compensate the friction force or overcome the moving difficulty because of the mechanism design, and vary the proportional gain Kp (as described in the previous section) to have a different stiffness depending on the robot position and the moving direction. If the value of $K_d K_f$ is chosen so that

$$K_d K_f = c, \quad (4.61)$$

where c is a constant and has to be decided empirically. Then substituting (4.61) into equation (4.52), the desired velocity in zone I & II is given by

$$\dot{X}_d = \frac{c}{K_d} F_s, \quad (4.62)$$

<u>Case II</u> : when the robot is moved toward zone III.

Assume the initial position of the robot X_0 is at the boundary of zone II, for instance $X_0 = X_2$ (see Fig. 4.1). As analyzed in the last section, the desired position X_d is always set as X_2. If the initial velocity is equal to zero and the external force is constant, the system transfer function is found

$$X(S)=\frac{X_2}{S}+\frac{1}{S}\frac{(1+K_dK_f)F_s}{MS^2+(K_c+K_d)S+K_p} \qquad (4.63)$$

Using the final-value theorem, the steady state position X_s is

$$\begin{aligned} X_s &= \lim_{S\to 0} SX(S) \\ &= X_2+\frac{(1+K_dK_f)F_s}{K_p}, \end{aligned} \qquad (4.64)$$

i.e., the maximum position error (distance to zone II) due to the external force Fs is

$$\begin{aligned} X_{error} &= X_s - X_2 \\ &= \frac{(1+K_dK_f)F_s}{K_p}. \end{aligned} \qquad (4.65)$$

Therefore, if the term $1+K_dK_f$ is set as to zero (i.e. $K_f=-\frac{1}{K_d}$), the position error can be reduced to zero. When the initial position of the robot is just at the edge of zone III, a checking algorithm has to be given

$$K_f=-\frac{1}{K_d} \quad \text{when } F_sX\geq 0$$

and $$K_f=\frac{c}{K_d} \quad \text{when } F_sX<0, \qquad (4.65)$$

where X is the robot current position and c is the constant value from (4.62). Thus, the robot can be moved back into zone II easily but not beyond zone III.

4.1.2.1 Discussion

The modified damping control method, which involves a force sensor, can compensate for the friction force or overcome the moving difficulty because of the mechanism design. In general, the decision algorithm of the desired position is only dependent on the definition of the artificial motion constraint and is given by

$X_d(N)=X(N)$ if X(N) is in zone I or II

$X_d(N)=X_2$ or $-X_2$ if X(N) is in zone III,

where X(N) is the robot current position at time NΔt. The desired velocity is decided by the algorithm $\dot{X}_d = K_f F_s$, and

(i) $K_f = \frac{c}{K_d}$, if X(N) is in zone I or II,

where c is a constant and will be decided empirically.

(ii) $K_f = -\frac{1}{K_d}$, when $F_s X \geq 0$,

and $K_f = \frac{c}{K_d}$, when $F_s X < 0$, if X(N) is in zone III.

The feedback gain design for the modified damping control method can be shown as in Fig. 4.5. The value of the proportional gain Kp increases steadily from zone I to zone II. The forward value is, however, higher than the backward value. Thus, the robot can be moved back to zone I more easily. The derivative gain Kd is chosen so that the an overdamped system can be attained.

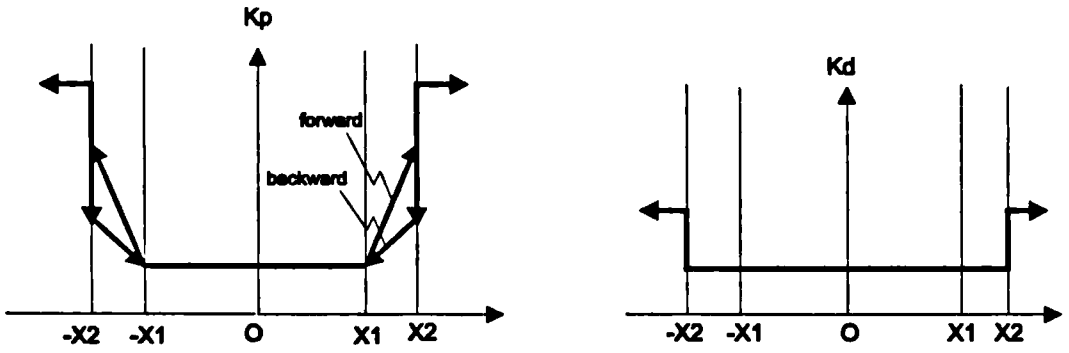


Fig. 4.5 Design of the control gains for one DOF system

It should be noticed that the control law of the inner closed loop

$$F_r = K_p(X_d - X) + K_d(\dot{X}_d - \dot{X}) \qquad (4.54)$$

does not include the force feedback gain K_f, and the system characteristic function is the same as that in the implicit force control algorithm. Therefore, it will not have the

kind of stability problems as in industrial robot force control. The stability problem will be discussed in detail in the next chapter.

4.2 Experimental tests

The theoretical analysis in the last section will now be examined by experimental tests. A one DOF system is set up as in Fig. 4.6, where the link is 120 mm long. A mass M is hung from the end of the one-link manipulator to simulate a constant guiding force F.

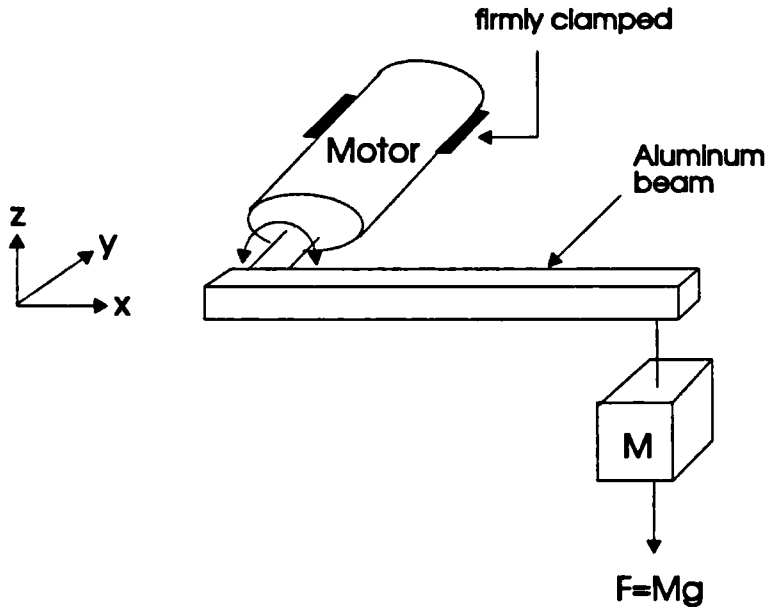


Fig. 4.6 The schematic diagram of the one-link manipulator driven by a constant force.

4.2.1 Identification of the One DOF system

The mathematical model of the one link-manipulator system will be first identified and then used for the analysis and design of the force control gains (the same model of the motor will be used for computer simulations and for the controller design of a multi-DOF manipulator in chapter 6 and 7). In general, the experimental approach for

determining system parameters of a mathematical model includes four steps [Landau 90]:

1. Input/output data acquisition under an experimentation protocol.
2. Choose the model structure.
3. Estimate the model parameters.
4. Examine the validation of the identified model.

However, based on laws of physics, a fairly complete model (knowledge type model) can be first described, and experiments are then designed to identify the unknown parameters.

4.2.1.1 The mathematical model and experimental identification

Fig. 4.7 shows the schematic diagram of a one DOF system, where the actuator load is assumed to consist of an inertia J, a damper with damping constant B, and a static friction τ_s, and a Coulomb friction τ_f. The actuator is an armature-controlled DC motor; R_a and L_a are the armature resistance and inductance, respectively; E_a is the control voltage of the digital-to-analogue converter (DAC), and i_a is the output of the current amplifier.

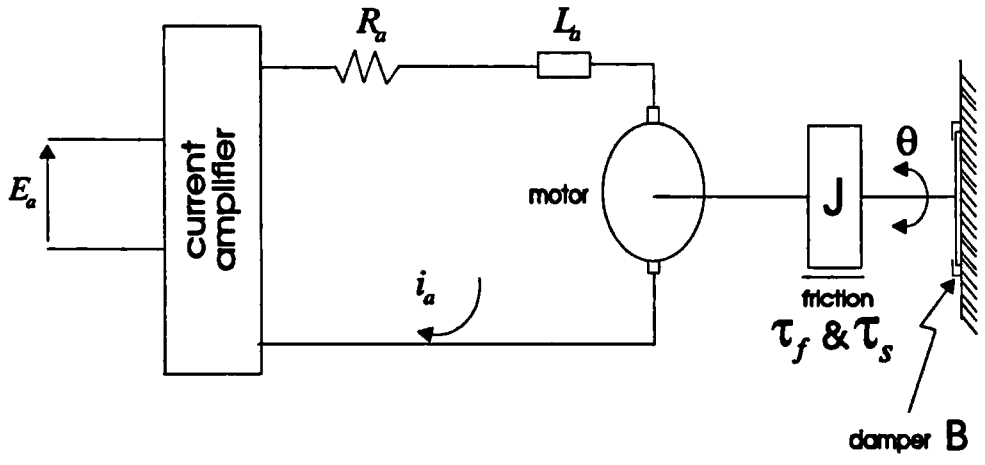


Fig. 4.7 The schematic diagram of a one DOF system

The dynamic equation of the system can be described by

$$T(t) = J\ddot{\theta}(t) + B\dot{\theta}(t) + \tau_f sgn(\dot{\theta}), \qquad (4.67)$$

and

$$T(t) = K_t i_a(t), \qquad (4.68)$$

where $sgn(\dot{\theta})$ is equal to $\frac{\dot{\theta}}{|\dot{\theta}|}$; T and θ are the torque output and the position of the motor, respectively; and K_t is the motor torque constant. It should be noticed that the static friction force $(\tau_s)_{\dot{\theta}=0}$ is not included in the dynamic equation (4.67). The static friction exists only when the motor is stationary but is on the point of moving, and its sign depends on the direction of motion or the initial direction of velocity. Once motion of the motor begins, $(\tau_s)_{\dot{\theta}=0}$ vanishes and Coulomb friction τ_f takes over.

Assuming the bandwidth of the current amplifier is much higher than the mechanical system, the amplifier can be represented as a constant gain K_a,

$$i_a = K_a E_a. \qquad (4.69)$$

The DAC is described by

$$E_a = K_s V, \qquad (4.70)$$

where V is the input of the DAC from a computer and is derived from a control algorithm. Combining (4.68), (4.69) and (4.70) yields

$$T(t) = KV, \qquad (4.71)$$

where the torque gain K is equal to $K_s K_a K_t$. The block diagram of the open loop system is shown as in Fig. 4.8. The parameters of the torque gain K, friction force $\tau_f sgn(\dot{\theta})$, system inertia J and damping constant B will be identified by experiments.

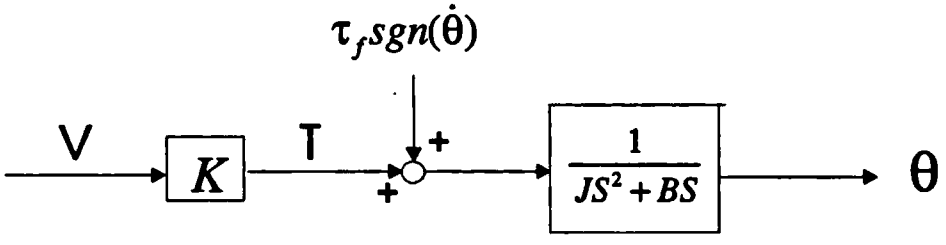


Fig. 4.8 The block diagram of an one DOF system

(I) Torque gain K and static friction $(\tau_s)_{\dot{\theta}=0}$

The value of the torque gain K can be estimated by sending a control signal (range from -100 to 100) to the DAC from the motor controller (which will be described in a detail in chapter 7) and measuring the output torque of the motor. This can be represented by the following equation,

(Torque measured) = K * (Input value of the DAC) - static friction,

Fig. 4.9 shows the experimental results, in which output torque of the motor was measured by a strain gauge based force sensor (see chapter 7). Curve fitting was applied to the experimental data by a least-square algorithm, and the value of the torque gain K was found as 0.135 Nm/(input to DAC) and the static friction was about 0.41 Nm.

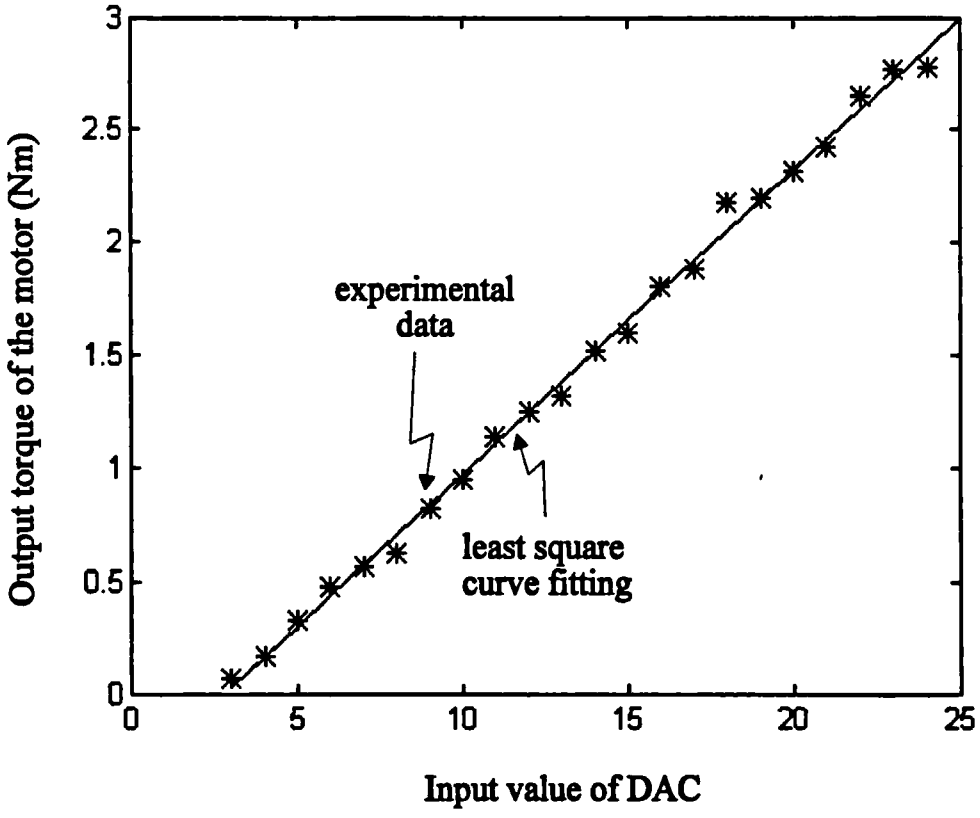


Fig. 4.9 The experimental data of the controlling signal to DAC and the motor output torque.

(II) Inertia J, damping constant B and Coulomb friction $\tau_f\ sgn(\dot{\theta})$

There are three different types of classical identification methods used for obtaining parametric models: impulse, step and sine-wave testing. In general, the parameters of the model are estimated by measuring and analyzing the response to the simple wave-forms. The step response method will be adapted here, because a step function is the easiest of all inputs to produce and has less risk of saturation within the system [Norton 86]. In order to estimate these system parameters, the experiments of a closed loop control system are designed. Fig. 4.10 shows the schematic diagram, where K_p is the proportional feedback gain and θ_d is the reference position. The dynamic equation of the closed loop system in Fig. 4.10 can thus be given as

$$J\ddot{\theta} + B\dot{\theta} + KK_p\theta = KK_p\theta_d - \tau_f, \text{ when } \dot{\theta} > 0, \qquad (4.72)$$

and

$$J\ddot{\theta} + B\dot{\theta} + KK_p\theta = KK_p\theta_d + \tau_f, \text{ when } \dot{\theta} < 0. \quad (4.73)$$

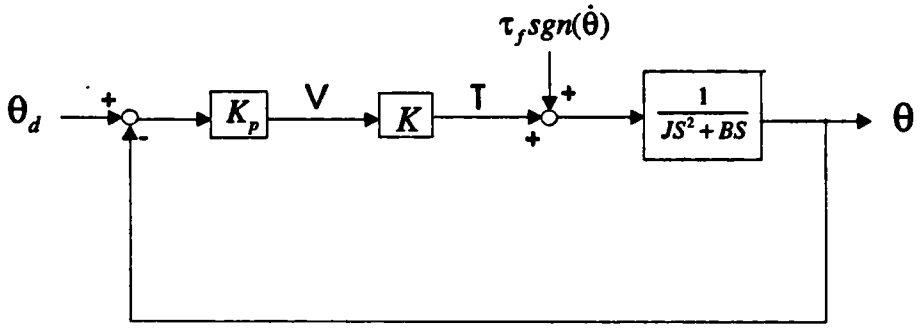


Fig. 4.10 The closed control scheme for an one DOF system with a constant proportional gain.

Notice that as described for (4.67) the static force is not included in the dynamic equations. Equation (4.72) and (4.73) can be rewritten as

$$\ddot{\theta} + 2\xi\omega_n\dot{\theta} + \omega_n^2\theta = \frac{KK_p}{J}\theta_d - \frac{\tau_f}{J}, \qquad \text{when } \dot{\theta} > 0, \qquad (4.74)$$

and

$$\ddot{\theta}+2\xi\omega_n\dot{\theta}+\omega_n^2\theta=\frac{KK_p}{J}\theta_d+\frac{\tau_f}{J}, \qquad \text{when } \dot{\theta}<0, \tag{4.75}$$

where the natural frequency $\omega_n=\sqrt{\frac{KK_p}{J}}$ and the damping ratio $\xi=\frac{B}{2\sqrt{JKK_p}}$. The system vibrates at a frequency

$$\begin{aligned}\omega_d &= \sqrt{1-\xi^2}\omega_n \\ &= \sqrt{\frac{4JKK_p-B^2}{4J^2}},\end{aligned} \tag{4.76}$$

which is defined as the frequency of damped vibration [Rao 86]. Assume the system starts with zero position and velocity, i.e. the initial conditions are

$$\begin{aligned}\theta(t=0)&=0\\ \dot{\theta}(t=0)&=0.\end{aligned} \tag{4.77}$$

When a step function input θ_d is given , the system response within first half cycle when $\dot{\theta}>0$ (from time zero to time $\frac{\pi}{\omega_d}$) is given by

$$\theta(t)=(\theta_d-\frac{\tau_f}{KK_p})-\frac{(\theta_d-\frac{\tau_f}{KK_p})}{\sqrt{1-\xi^2}}exp(-\xi\omega_n t)sin(\omega_d+\phi), \tag{4.78}$$

where $\phi=tan^{-1}(\frac{\xi}{\sqrt{1-\xi^2}})$. The overshoot happens at time $\frac{\pi}{\omega_d}$ and is equal to

$$\theta(t=\frac{\pi}{\omega_d})-\theta_d=(\theta_d-\frac{\tau_f}{KK_p})exp(-\frac{\xi\pi}{\sqrt{1-\xi^2}})-\frac{\tau_f}{KK_p}. \tag{4.79}$$

It should be noted that because of the effect of the Coulomb friction, the overshoot in (4.79) is equal to zero when $[(\theta_d-\frac{\tau_f}{KK_p})\exp(-\frac{\xi\pi}{\sqrt{1-\xi^2}})-\frac{\tau_f}{KK_p}]$ is less or equal to zero even though the system is underdamped. In brief, the frequency of damped vibration (4.76) is only the function of the inertia J and the damping constant B. Therefore, it can be used to estimate J and B. While the equation of overshoot (4.79) can be used to identify the Coulomb friction.

Fig. 4.11 shows the experimental results of different values of the proportional gain with the reference position θ_d = 0.01 radian. When K_p is equal to 4000 and 5000, the

frequency of damped vibration ω_d is about 143 Hz and 151 Hz, respectively. Through equation (4.76), the inertia J and damping constant B are derived as

$J = 0.023\ Kgm^2$

B = 1.86 Nm/(rad./sec).

The Coulomb friction can be attained in a similar manner by measuring the overshoot and using equation (4.75), which is equal to 0.32 Nm. The mathematical model of the one DOF system can then be represented by the expression

$$T(t) = 0.023\ddot{\theta}(t) + 1.86\dot{\theta}(t) + 0.32sgn(\dot{\theta}). \qquad (4.80)$$

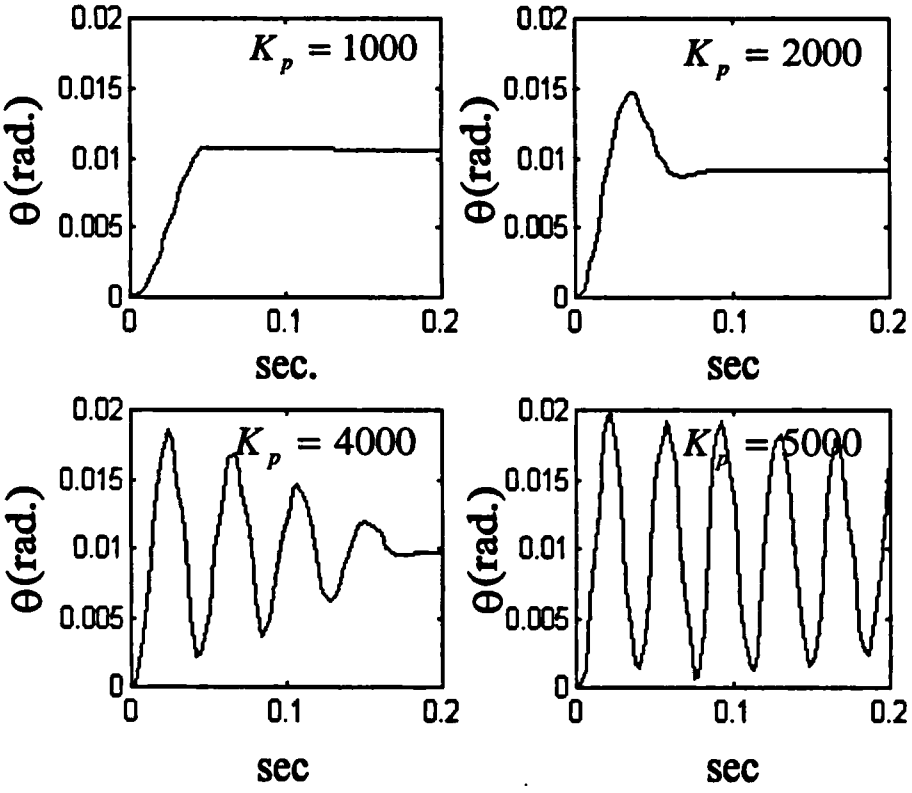


Fig. 4.11 The experimental results of the position control scheme in Fig. 4.10 with θ_d =.01 and variable proportional feedback gain K_p.

Notes

(1). The natural frequency ω_n and damped vibration frequency ω_d of the system are not changed by the Coulomb friction.

(2). The Coulomb friction τ_f causes the steady state position error, because the system motion ceases when the position error is less than $\frac{\tau_f}{KK_p}$ (in which the output torque of the motor to correct the position error is less than τ_f). But Coulomb friction can increase the stability in a control system [Kuo 80].

(3). Increasing the proportional feedback gain K_p decreases the damping ratio ξ, while ω_n and ω_d are increased, thus it increases the vibration frequency.

4.2.1.2 Validation of the identified model

The mathematical model of the one DOF system has been identified by step response testing in the last section. Validation of the model will be examined here. Derivative feedback gain is added to the control system. Based on the identified model, effects of the control gains are analyzed and then verified by experimental tests.

When a derivative feedback is used, the PD position control diagram is shown as in Fig. 4.12, where K_p and K_d are the proportional and derivative feedback gains, respectively.

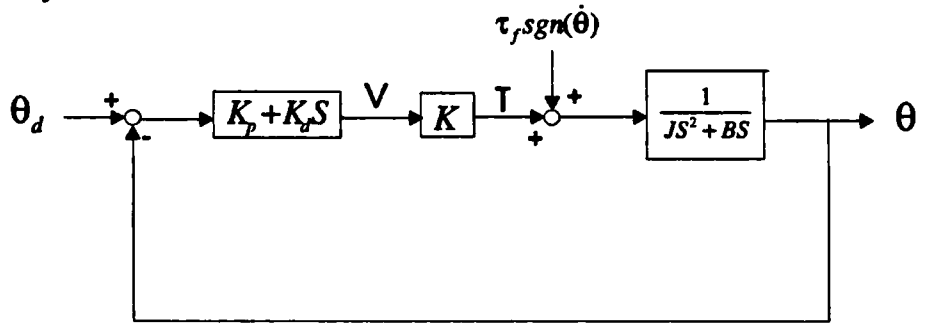


Fig. 4.12 The PD position control scheme for an one DOF system.

The dynamic equation of the closed loop system can be given by

$$J\ddot{\theta}+(B+KK_d)\dot{\theta}+KK_p\theta=KK_p\theta_d+\tau_f sgn(\dot{\theta}). \qquad (4.81)$$

The natural frequency ω_n, damped frequency ω_d and damping ratio ξ are equal to

$$\omega_n=\sqrt{\frac{KK_p}{J}}, \qquad (4.82)$$

$$\omega_d = \omega_n\sqrt{1-\xi^2}$$

$$= \sqrt{\frac{4JKK_p - (KK_d + B)^2}{4J^2}}, \qquad (4.83)$$

and

$$\xi = \frac{(KK_d + B)}{2\sqrt{JKK_p}}. \qquad (4.84)$$

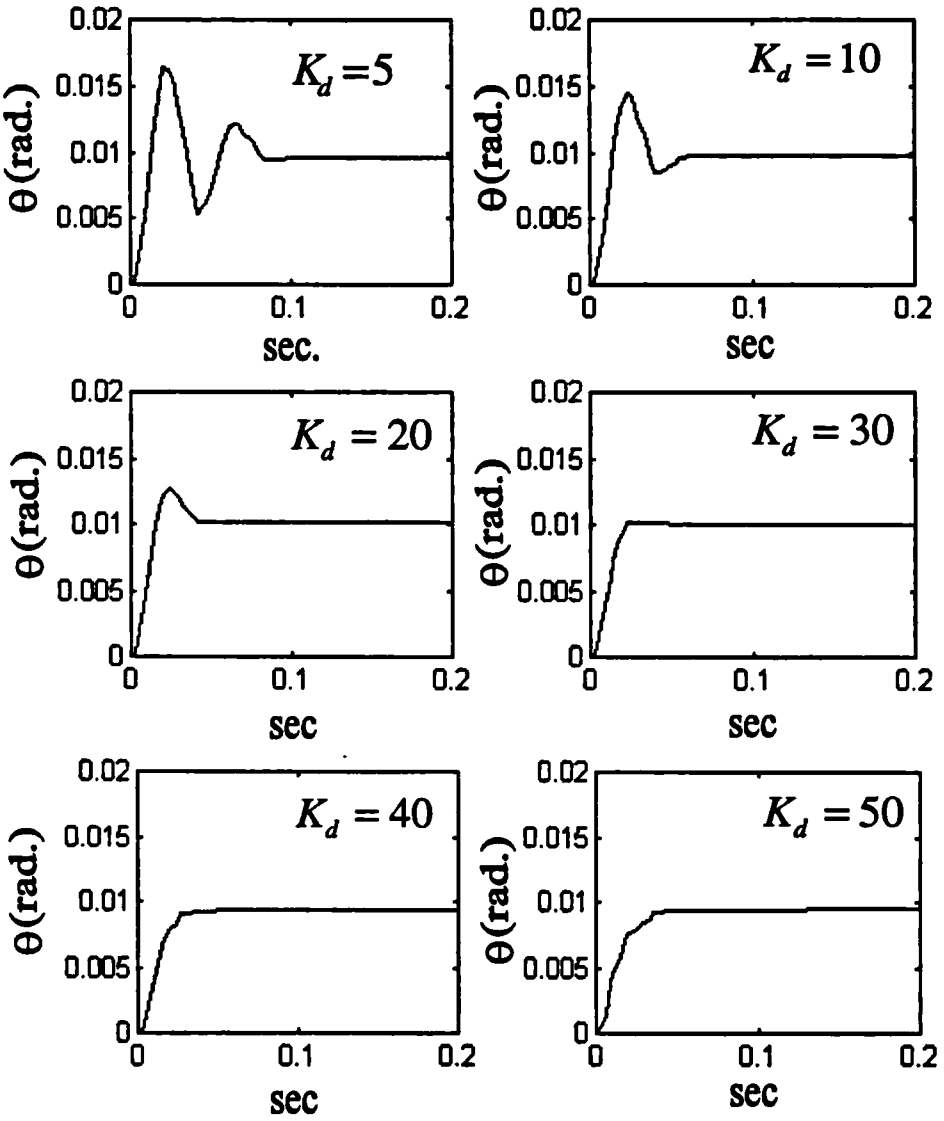


Fig. 4.13 The experimental step input results of K_d in PD position control with θ_d =.01, K_p = 4000 and variable derivative feedback gain K_d.

Recall that the overshoot of the system response is zero when

$$\theta(t=\frac{\pi}{\omega_d})-\theta_d=(\theta_d-\frac{\tau_f}{KK_p})exp(-\frac{\xi\pi}{\sqrt{1-\xi^2}})-\frac{\tau_f}{KK_p}\leq 0. \quad (4.79)$$

Increasing the derivative gain will increase the damping ratio and thus reduce the overshoot. If the input θ_d and the proportional gain K_p are given the values of 0.01 and 4000 respectively, the overshoot will be equal to zero (by deriving for (4.79)) when ξ is equal or greater than 0.67 (in which $K_d \geq 23$). Fig. 4.13 shows the experimental results of position with $\theta_d = 0.01$ and $K_p = 4000$ and the derivative feedback gain K_d varying from 5 to 50. As predicted from equation (4.79), the overshoot of the system decreases as K_d increases and equals zero when K_d is near 30 (which is slightly more than the estimated value). Therefore, when designing an overdamped system by using the identified model in the next chapters, the derivative gain applied in practice will be a little bigger than the analyzed value.

4.2.2 Implicit force control

This section continues with an analysis of the implicit force control strategy with an artificial motion constraint as described in the section 4.1.1. The controller will be designed on the basis of the identified mathematical model (4.80). Subsequently, experimental tests will be implemented to verify the analysis.

The one DOF system was set up as in Fig. 4.6, and the control diagram was as shown in Fig. 4.2. The motion constraint (see Fig. 4.1) was defined as

$\theta = 0 \sim 0.2$ radian, zone I

$\theta = 0.2 \sim 0.3$ radian, zone II

and $\theta \geq 0.3$ radian, zone III.

A one kilogram mass was fixed at the end of the one-link manipulator to simulate a guiding force. At time zero, the weight was suddenly released as the link was parallel to the horizontal plane. When the manipulator moves, its current position and the resistance force from the actuator were recorded. The sampling times of the inner and

outer loop were 2.0ms and 0.21ms, respectively. It was noted that the maximum steady state error e_{ss}(defined as the distance from the final position to the boundary of zone III) without using integral feedback was given by

$$e_{ss}(t=\infty) \leq \frac{(\tau_f + FL)}{K_p}, \qquad (4.85)$$

The relative effects of different types of control gains on the system were examined by the experimental tests. Fig. 4.14 to Fig. 4.16 show the experimental results, where the maximum values of the proportional gain K_p and the derivative feedback gain K_d are 4000 and 50, respectively. Fig. 4.14(a) shows the result in which the control gains are given as zero in zone I & II, and the control law becomes a high gain PD position control in zone III. The only resistance force is due to Coulomb friction within zone I & II. The system is constrained from moving out of zone III after the transient response time but an overshoot occurs in zone III. Fig. 4.14(b) is the result when the derivative control gain Kd remains as a constant in zone II. It is seen that the manipulator provides a constant resistance after a short transient time when it is moved into the zone II, and there is no overshoot in zone III. Fig. 4.15(a) shows that the resistance force is continuously increased when the derivative gain is proportionally increased in zone III (from A to B). Fig. 4.15(b) & 4.16 show the results of other different types of control gains. It is concluded that the desired variable stiffness in different positions and different directions can be designed by changing either the proportional gain or the derivative gain, and the system can be constrained from moving out of zone III. The design of control gains in which the system is overdamped in zone II & III is recommended. The maximum values of the control gains depend on the desired stiffness but are limited by the characteristics of the system hardware (see the conclusion in the section 4.1.1).

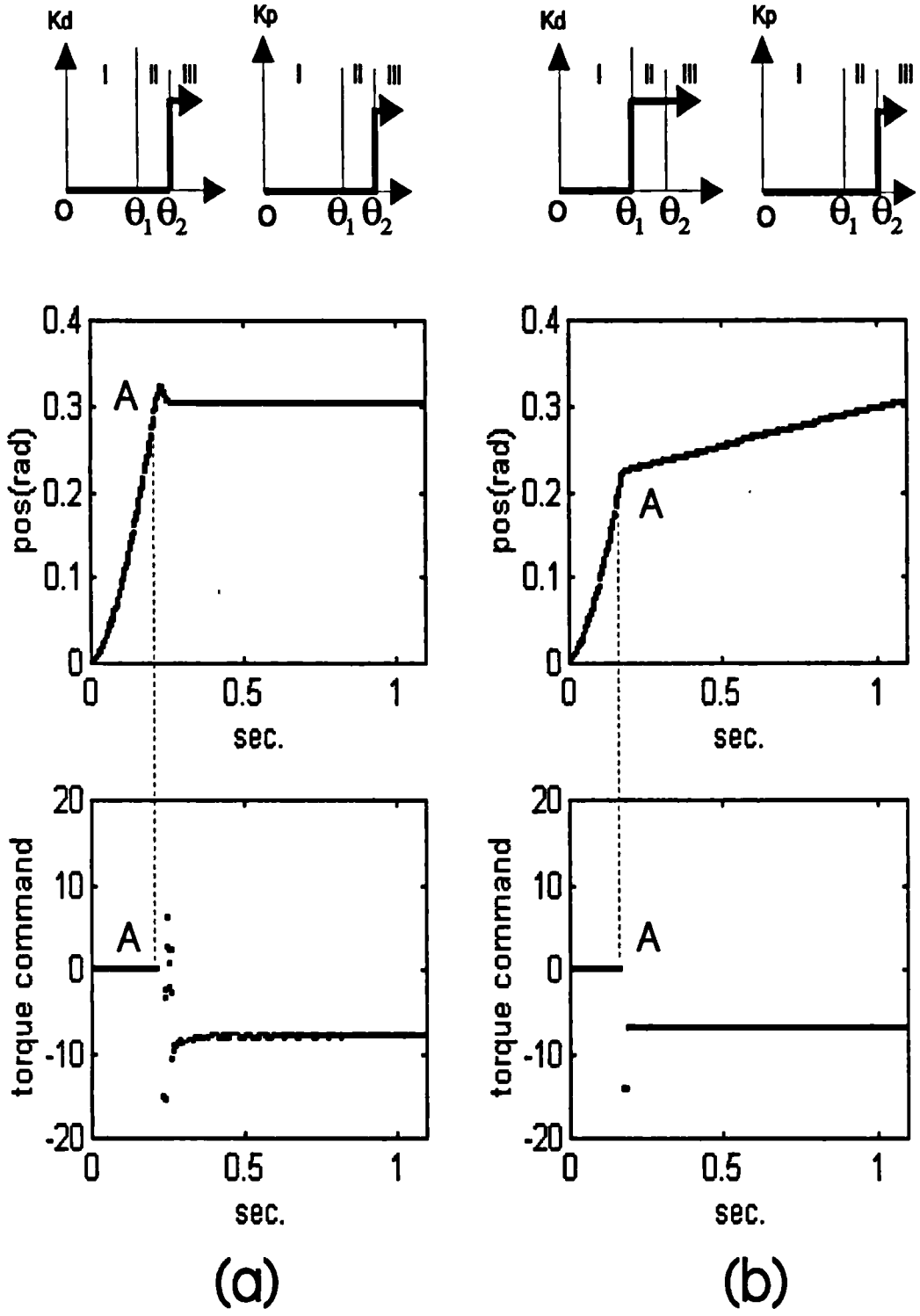


Fig. 4.14 'Resistance' force from the actuator and displacement characteristics against time, when the one-link manipulator is driven by a constant force.

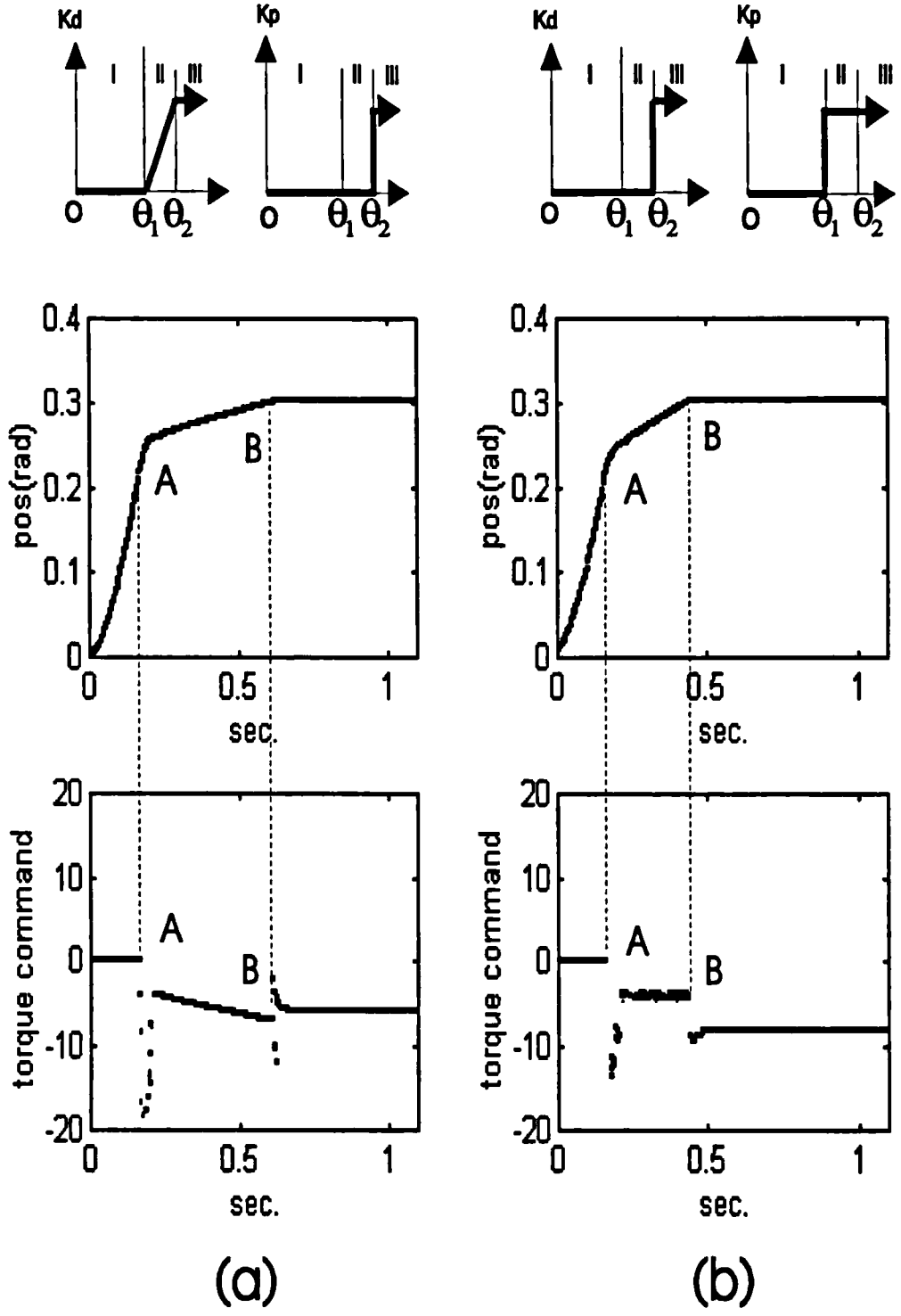


Fig. 4.15 'Resistance' force and displacement characteristics as for Fig. 4.14, but with different types of control gains.

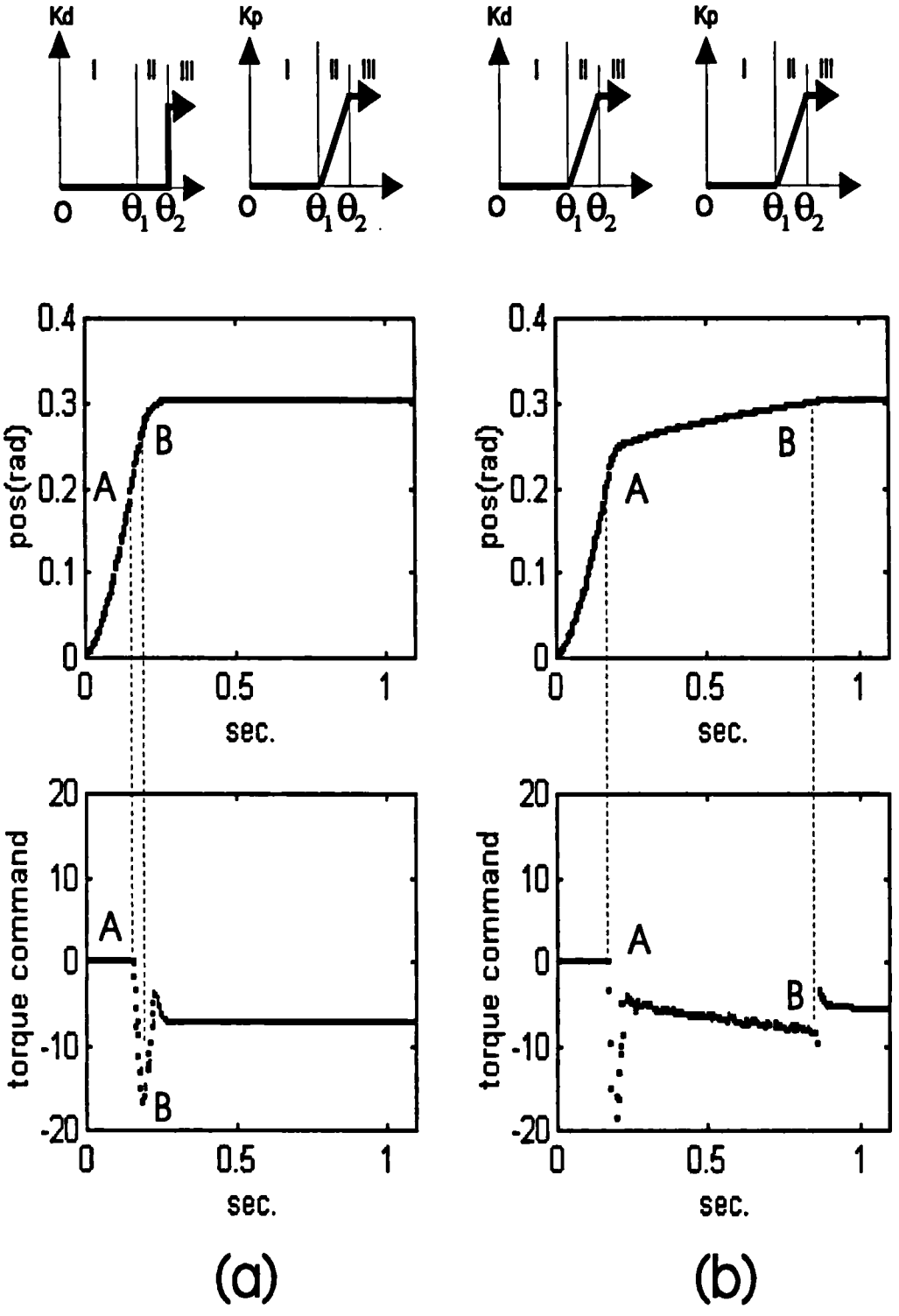


Fig. 4.16 'Resistance' force and displacement characteristics as for Fig. 4.14, but with different types of control gains.

4.2.3 Modified damping force control

In the previous section, implicit force control for a one DOF system was implemented, and effects of the feedback gains were examined. In this section, experiments by using the modified damping control method, which involves a force sensor, will be carried out. Its performance will be compared to that using implicit force control. The set-up of the one DOF system and the definition of motion constraint are as for the implicit force control. The control diagram is shown as in Fig. 4.17. This is similar to Fig. 4.4 except a dead-band filter is added to reduce the noise of the force sensor.

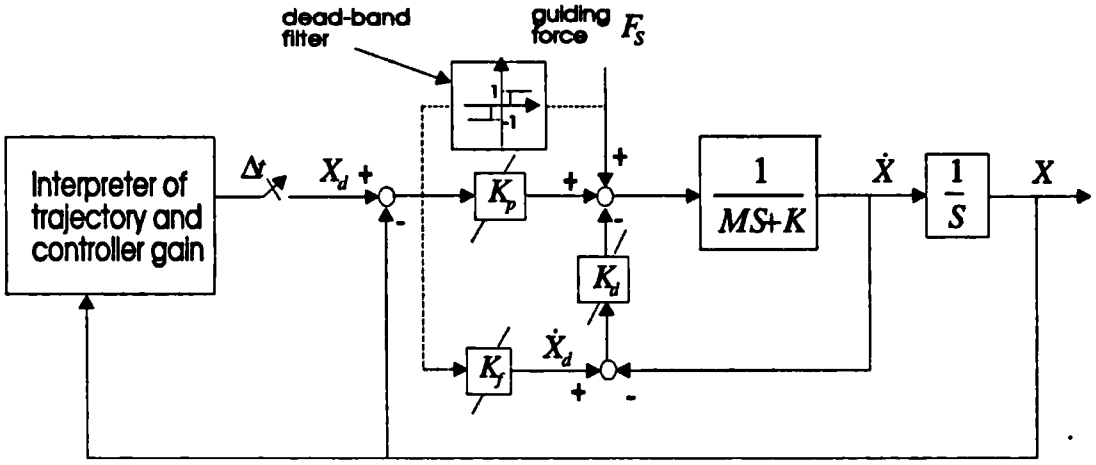


Fig. 4.17 Modified damping control for an one DOF system

In order to compare the performance of implicit force control and modified damping control, the manipulator was moved by hand using these two control algorithms separately and the guiding force and displacement were recorded. Fig. 4.18 shows the results of the implicit force control method. The manipulator was moved from $\theta = 0$ to $\theta = 0.3$ rad. by hand as an approximate ramp input and the maximum values of the control gains K_d and K_p were 50 and 4000, respectively. It can be seen that the manipulator did not move before the guiding force overcame the friction force (at point A) which was about 3.5 Newton (N). In zone II (from B to C), the resistance force of the manipulator was continuously increased. When the manipulator was moved towards the boundary of zone III, the manipulator was restricted from moving

outside the zone. The position error was about 0.0045 radian (0.54 mm) when the external force was equal to 20 N. A similar experiment by using the modified damping control method is implemented. The desired velocity as analyzed in the section 4.1.2 was given by

$$\dot{X}_d = \frac{c}{K_d} F_s, \qquad (4.62)$$

where F_s is the guiding force. The constant c is designed as

c = 0.15 in zone I,

c = 0.05 in zone II,

and in zone III

c = 0.15 when $F_s\theta < 0$,

c = -.005 when $F_s\theta > 0$.

Fig. 4.19 shows the experimental result of the modified damping control method. The manipulator starts moving at point A when the guiding force is just over 0.5 N. In zone III, the position error is about 0.001 radian (0.12mm) when the guiding force is equal to 20 N. Compared with the result of the implicit force control method, the manipulator can be more easily moved within the free motion region (zone I) and yields a smaller position error.

note: Position error may be reduced by adding an integral feedback in zone III. Its advantages and disadvantage will be discussed in chapter 5.

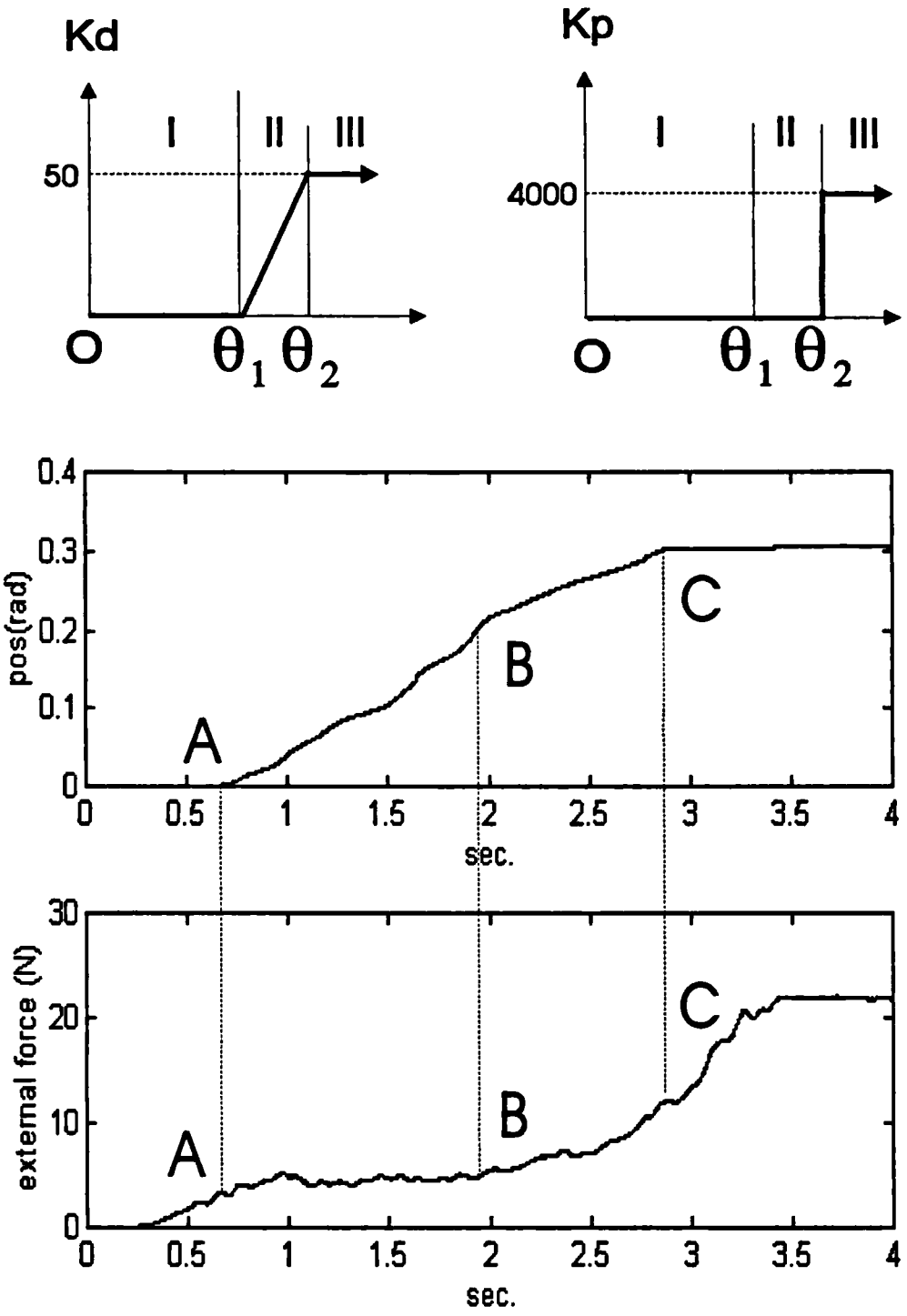


Fig. 4.18 Force and displacement characteristics against time, when the manipulator is moved by hand from $\theta = 0$ to $\theta = 0.3$ rad under the implicit force control.

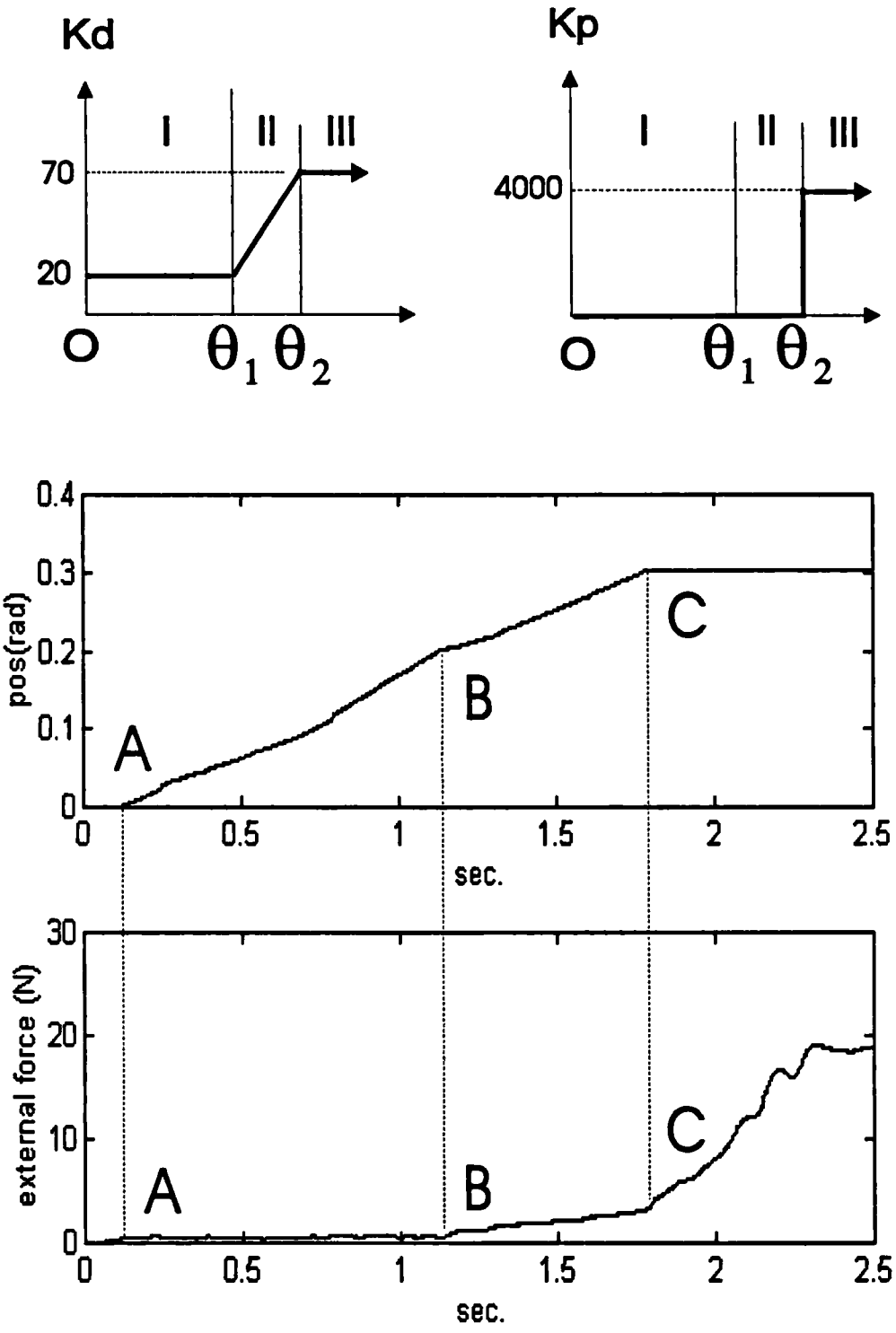


Fig. 4.19 Force and displacement characteristics as for Fig. 4.18, but with modified damping control method.

4.3 Conclusion

In this chapter, force control strategies, which includes implicit force control and modified damping control, with an artificial motion constraint for a one DOF system have been analyzed. The overall control algorithm is split into two computational loops with different rates. The outer loop ('interpreter' program) updates the desired position and control gains at the end of every computational cycle with a slower rate. The inner loop program implements the control law at a faster sampling rate, which can increase the bandwidth and thus the stability of the overall system. Through the design of the desired position and control gains, the robot can be constrained to move only in the planned zones and have variable stiffness in different positions and different directions of motion.

A one DOF manipulator has been set up, and its mathematical model was identified. Based on the model, experimental tests have been implemented to examine the performance of the control algorithms. From experimental results, the analysis has been confirmed.

For simplicity sake, in the analysis of this chapter, the external forces have only been considered as the surgeon's guiding force and the cutting force. The guiding force is assumed to be independent of the robot position and the average cutting force is modelled as proportional to the cutting speed. More complicated cases will be discussed in section 5.4, and will also be simulated in chapter 6.

CHAPTER 5

FORCE CONTROL STRATEGY FOR MULTI-DOF ROBOT - THEORETICAL ANALYSIS

The concept of a force control strategy with an artificial motion constraint for a one DOF system has been analyzed and confirmed by experimental tests in the previous chapter. This design concept will be extended to a multi-DOF system in this chapter. Difficulties of applying this idea into the multi-DOF robot are:

(i). How to specify and construct an artificial motion constraint in three dimensional space.

(ii). How to design the desired position and velocity of the robot based on the robot current position, the guiding force and the motion constraint.

(iii). How to design the control gains and then transform the desired stiffness from Cartesian space to joint space.

5.1 Representation of the motion constraint

In most robot applications, robots are normally commanded to move to a desired position or along a desired trajectory. Therefore, it is necessary to describe the desired motion of the manipulator, i.e. the time history of the robot position, velocity and acceleration. In general, the problem can be divided into three parts:

(i) the user interface by which an operator can specify the desired path,

(ii) the representation of the trajectory in the computer, and

(iii) computing and generating the actual trajectory from the internal representation on line.

For our application, there are two more considerations. Firstly, the artificial motion constraint is not just a trajectory, but could also be a volume. Secondly, the relative distance of the robot current position and the motion constraint has to be calculated on-line in order to ensure the desired stiffness of the robot. Therefore, it is important to find an efficient, fast way to represent and store the motion constraint in the computer memory. Thus, the control algorithm can access these data and easily derive the relative position of the robot within the motion constraint.

Assume that the motion constraint is always specified by points, which can be either attained from a pre-operative image processing system or defined by using a mouse or light-pen on the computer screen intra-operatively. As described in the chapter two, the motion constraints are classified into three types: point, trajectory and closed volume constraints.

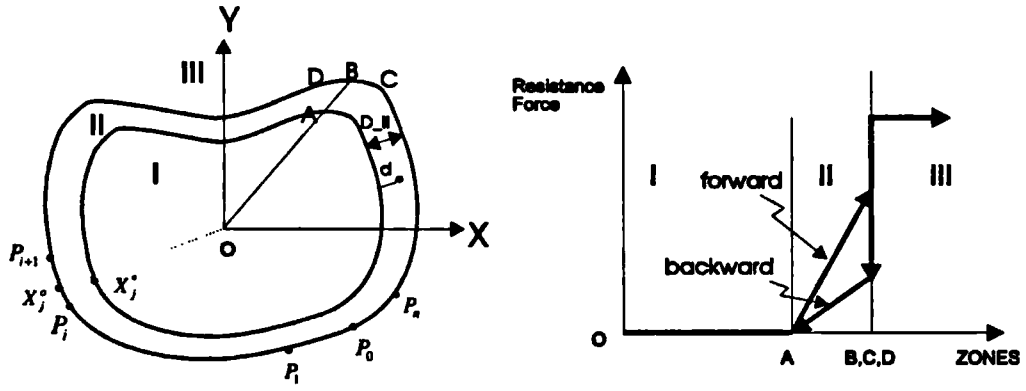


Fig. 5.1. Resistance Force in 3 different zones and different directions of motion constraint.

5.1.1 Closed volume constraint

For a 3D volume constraint, the constraint can be sliced into pieces along the z axis. Figure 5.1 shows the regional constraint in each x-y plane. Inside zone I, the robot can be freely moved by the surgeon. While in zone II, the resistance force is steadily increased from zone II to the boundary of zone III. At the outer boundary, the robot

can only move along the boundary or back into the interior region. Zone III is a 'no go' area.

To represent the two-dimensional constraint in the computer, one simple and efficient method is to make a look-up table for the regional constraint in polar coordinates. Suppose the motion constraint at $z = z_i$ plane is defined by the points

$$P_0(x_0, y_0), P_1(x_1, y_1), \cdots, P_n(x_n, y_n),$$

and the centre of the coordinate is enclosed in the constraint. The procedures of building the look-up table are as follows :

(i). By using the B-spline curve-fitting method [Ammeraal], two sets of points can be derived from the defined points $P_i(x_i, y_i)$ (see Fig. 5.1),

$$\begin{array}{ll} X_0^o(x_0^o, y_0^o) & X_0^*(x_0^*, y_0^*) \\ X_1^o(x_1^o, y_1^o) & X_1^*(x_1^*, y_1^*) \\ \vdots \quad \text{and} & \vdots \\ X_N^o(x_N^o, y_N^o) & X_N^*(x_N^*, y_N^*), \end{array}$$

where $X_j^o(x_j^o, y_j^o)$ are $X_j^*(x_j^*, y_j^*)$ are the points in the outer and inner loop, respectively. If m points are added between every two successive points $P_i(x_i, y_i)$ and $P_{i+1}(x_{i+1}, y_{i+1})$, $X_j^o(x_j^o, y_j^o)$ is derived by

$$\begin{aligned} x_j^o &= a_3 t^3 + a_2 t^2 + a_1 t + a_0 \\ y_j^o &= b_3 t^3 + b_2 t^2 + b_1 t + b_0, \end{aligned} \qquad (5.1)$$

where $j = (m+2)i + k$ and k is an integer varied from 0 to m+1; t = k/(m+2). The coefficients of these equations are

$$\begin{aligned} a_3 &= (-x_{i-1} + 3x_i - 3x_{i+1} + x_{i+2})/6 \\ a_2 &= (x_{i-1} - 2x_{i+1} + x_{i+2})/2 \\ a_1 &= (-x_{i-1} + x_{i+1})/2 \\ a_0 &= (x_{i-1} + 4x_i + x_{i+1})/6, \end{aligned} \qquad (5.2)$$

the coefficients (b_3, b_2, b_1, b_0) are derived from $(y_{i-1}, y_i, y_{i+1}, y_{i+2})$ in a similar manner. While $X_j^*(x_j^*, y_j^*)$ are the points in the inner loop and satisfy

$$x_j^* = x_j^o + D\sin\theta_j'$$
$$y_j^* = y_j^o + D\cos\theta_j', \qquad (5.3)$$

where D is the width of zone II and θ_j' is the tangential slope in the outer loop at position $X_j^o(x_j^o, y_j^o)$ and is given by

$$\theta_j' = \tan^{-1}\left(\frac{dy_j^o}{dx_j^o}\right)$$
$$= \frac{dy_j^o/dt}{dx_j^o/dt} \qquad (5.4)$$
$$= \frac{a_3t^2 + a_2t + a_1}{b_3t^2 + b_2t + b_1}$$

(ii). Converting $X_j^o(x_j^o, y_j^o)$ and $X_j^*(x_j^*, y_j^*)$ from Cartesian coordinate to polar coordinate respectively.

$$X_j^o(x_j^o, y_j^o) = X_j^o(r_j^o, \theta_j^o)$$
$$X_j^*(x_j^*, y_j^*) = X_j^*(r_j^*, \theta_j^*), \qquad (5.5)$$

where $r_j^o = \sqrt{(x_j^o)^2 + (y_j^o)^2}$ and $\theta_j^o = \tan^{-1}\left(\frac{y_j^o}{x_j^o}\right)$, and the parameters r_j^* and θ_j^* are derived from x_j^* and y_j^* in a similar manner.

(iii). Assume the points in (5.5) have been sorted ascending in angle θ_j^o and θ_j^* respectively, a lookup table can then be made easily. In order to save memory, the lookup table will just have 360 rows with the format

$$Table(z_i, \theta_j) = [R_j^* \quad R_j^o \quad \theta_j'], \qquad (5.6)$$

where z_i is the z = z_i plane; $\theta_j = j * \frac{\pi}{180}$, j = from 0 to 359; R_j^* , R_j^o and θ_j' are the inner loop radius, outer loop radius and the tangential slope of the outer loop at angle θ_j, respectively. The values of the radii and tangential slope of other angles can be derived by an interpolation method on-line.

note :

(1). The look-up table described above is valid when the motion constraint has a one-to-one mapping in polar coordinates, i.e. there is only one intersection point at every angle θ. If this condition is not sufficient, the format of the lookup table in (5.6) can be modified as

$$Table(z_i, \theta_j) = [n \quad R^*_{j1} \quad R^o_{j1} \quad \theta'_{j1} \cdots \quad R^*_{jn} \quad R^o_{jn} \quad \theta'_{jn}],$$

where n is the number of the intersection points between the motion constraint and the straight line $\theta = \theta_j$.

(2). In the above analysis, it is assumed that the robot current position is directly mapped to a planar constraint look-up table at $z = z_i$. However, in order to save the memory required, the look-up table in (5.6) will be built along the z axis with equal distance, i.e. $z_{i+1} - z_i = z_i - z_{i-1}$. Assume the robot current position is at (x, y, z), and $z_i < z < z_{i+1}$, the relative information of the motion constraint can be interpolated by the two successive look-up tables,

$$Table(z_i, \theta_j) = [R^*_j \quad R^o_j \quad \theta'_j]$$

and $$Table(z_{i+1}, \theta_j) = [R^*_j \quad R^o_j \quad \theta'_j].$$

(3). When the coordinate of the robot is not exactly aligned to that of the motion constraint, it is necessary to transform the points which specify the constraint into the robot coordinate before deriving the look-up table.

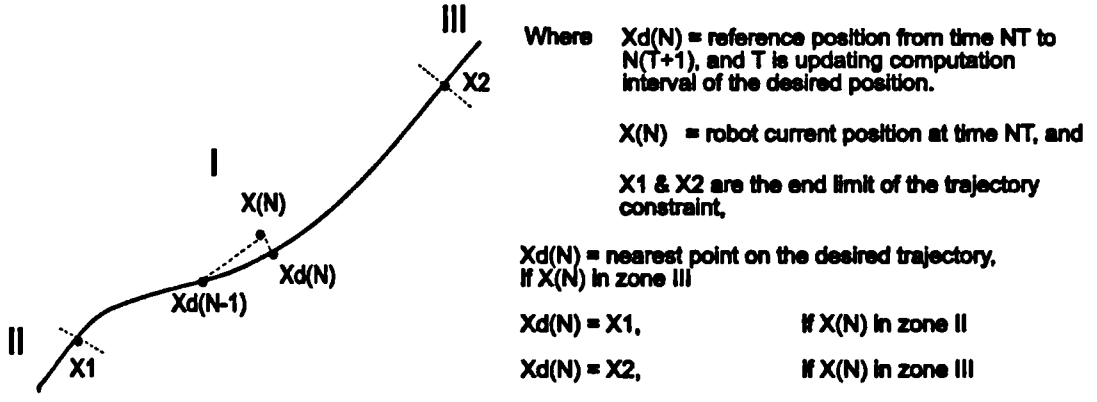


Fig.5.2 Definition and determination of reference position for trajectory constraint

5.1.2 Trajectory constraint

A trajectory constraint can be constructed as shown in Fig. 5.2, where zone II and III are 'no go' area. The trajectory can be represented and stored in the computer by the defined points. However, a pre-process of B-spline curve-fitting as described above may be necessary to attain more intermediate points, which are used later to calculate the desired position by interpolation.

5.2 Design of the desired position

The control system is split into two separate and parallel computation loops. The inner loop implements the control law and controls the output force of the robot with a faster computing cycle. The outer loop decides the desired position and velocity (if a force sensor is used) of the robot and the gains of the motor control loop at a slower computation cycle (detailed in a later section). Basically the change of the control gains can achieve the desired robot stiffness, while the design of the desired position can restrict the robot from being moved out of the region of artificial constraint. Therefore the first problem for the control system is to calculate (i) the relative distance between the robot current position and the boundary of zone II & III, which is used to decide the desired stiffness. and (ii) the desired position in real time.

5.2.1 Closed volume constraint

Assume the robot current position at time NT is X(N) = $[x \quad y \quad z]=[r \quad \theta \quad z]$, where $r=\sqrt{x^2+y^2}$ and $\theta=tan^{-1}(y/x)$, and the mapped look-up table is $Table(z,\theta_i)=[R_i^* \quad R_i^o \quad \theta_i']$, where j is the integer index varied from 0 to 360. In order to examine which zone the robot current position is, the radii from the coordinate centre to the inner loop and outer loop at angle θ are first found by an interpolation method. If $\theta_j < \theta < \theta_{j+1}$, the radii are given by

$$R^* = \frac{(\theta - \theta_j)R_{j+1}^* + (\theta_{j+1} - \theta)R_j^*}{(\theta_{j+1} - \theta_j)} \qquad (5.7)$$

and $$R^o = \frac{(\theta - \theta_j)R^o_{j+1} + (\theta_{j+1} - \theta)R^o_j}{(\theta_{j+1} - \theta_j)}. \quad (5.8)$$

The tangent angle is given in a similar manner,

$$\theta^* = \frac{(\theta - \theta_j)\theta'_{j+1} + (\theta_{j+1} - \theta)\theta'_j}{(\theta_{j+1} - \theta_j)}. \quad (5.9)$$

(i). If $r \le R^o$, the robot is in zone I or II, the relative distance between the robot and the boundary is equal to $(R^o - r)$. If $(R^o - r) \le \Delta r$ where Δr is a defined small distance, i.e. the robot is very close to the outer boundary, the desired position is set as the nearest point on the boundary, whose approximate solution is equal to $[R^o \quad \theta \quad z]$. This will reduce the 'disturbing' movement along the boundary. Otherwise, the desired position $X_d(N)$ is updated as the robot current position X(N) after every computation cycle, thus the robot will stay at its present position if the operator's guiding force is removed.

(ii). If $R^o < r$, the robot is in zone III. The desired position $X_d(N)$ has to be set as the nearest point at the boundary, so the controller can force the robot back into the constraint. One approximate solution for the desired position can be obtained efficiently as follows. As shown in Fig. 5.3, X is the robot current position; point C is the intersection between line OX and the boundary; line EF is the tangent at C and θ^* is the tangent angle; line XA and AD is orthogonal to line EF and OX respectively; and point B is the approximate desired position to be derived.

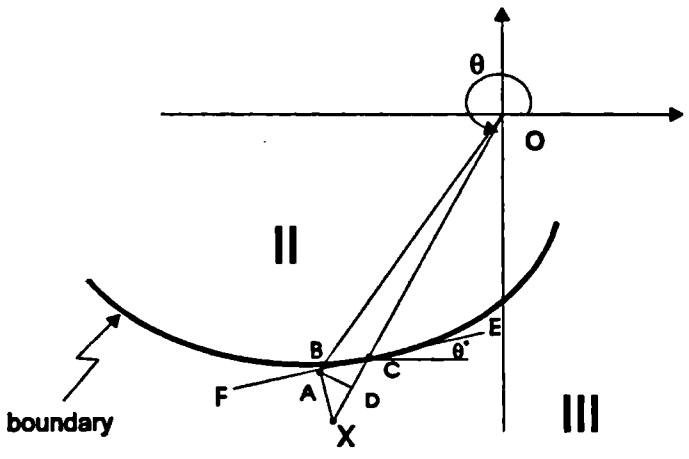


Fig. 5.3 Approximate solution of the desired position

If $\overline{OX}=r, \overline{OC}=R^o, \overline{AD}=a, \overline{AX}=b,$ and $\overline{DX}=c,$ then the geometrical relationship has

$$\phi=\frac{3}{2}\pi-\theta+\theta^*, \qquad (5.10)$$

$$b=(r-R^o)\cos\phi, \qquad (5.11)$$

$$\begin{aligned} a &= b\sin\phi \\ &= (r-R^o)\cos\phi\sin\phi \qquad (5.12) \\ &= \frac{(r-R^o)}{2}\sin 2\phi, \end{aligned}$$

and

$$\begin{aligned} c &= b\cos\phi \\ &= (r-R^o)\cos^2\phi \qquad (5.13) \\ &= \frac{(r-R^o)}{2}(1+\cos 2\phi). \end{aligned}$$

In order to find out the position of point B, the angle ψ is calculated first,

$$\begin{aligned} \tan\psi &= \frac{a}{r-c} \\ &= \frac{\frac{1}{2}(r-R^o)\sin 2\phi}{r-\frac{1}{2}(r-R^o)(1+\cos 2\phi)} \\ &= \frac{(r-R^o)\sin 2\phi}{(r+R^o)-(r-R^o)\cos 2\phi} \\ &= \frac{(r-R^o)\sin[2(\frac{3}{2}\pi-\theta+\theta^*)]}{(r+R^o)-(r-R^o)\cos[2(\frac{3}{2}\pi-\theta+\theta^*)]} \\ &= -\frac{\sin[2(\theta-\theta^*)]}{\frac{(r+R^o)}{(r-R^o)}+\cos[2(\theta-\theta^*)]}, \qquad (5.14) \end{aligned}$$

i.e.

$$\psi=\tan^{-1}\{-\frac{\sin[2(\theta-\theta^*)]}{\frac{(r+R^o)}{(r-R^o)}+\cos[2(\theta-\theta^*)]}\}. \qquad (5.15)$$

In equation (5.15), it is noticed that r and θ are given by the robot current position, while R^o and θ^* are calculated from the look-up table as expressed in (5.8) and (5.9). According to the value of angle (θ - ψ), the radius r_d from the coordinate centre to the desired position B can be obtained from the look-up table. Subsequently, the desired position is given by

$$\begin{aligned} x_d &= r_d \cos(\theta - \psi) \\ y_d &= r_d \sin(\theta - \psi). \end{aligned} \qquad (5.16)$$

and $z_d = z_i$.

The desired position for each joint can then be calculated by inverse kinematic transformation.

5.2.2 Trajectory constraint

As seen in Fig. 5.2, when the robot current position X(N) is in zone II or III, the desired position is given as the limit position X1 or X2, respectively. While the robot is in zone I, the desired position $X_d(N)$ is designed as the nearest point on the trajectory to X(N). Assume X(N) is between point $X_1(x_1, y_1, z_1)$ and $X_2(x_2, y_2, z_2)$ in the constraint, $X_d(N)$ can be attained by an interpolation method,

$$\begin{aligned} X_d &= [x_d \quad y_d \quad z_d] \\ &= X_1 + \frac{a(x_r - x_1) + b(y_r - y_1) + c(z_r - z_1)}{a^2 + b^2 + c^2} \cdot [a \quad b \quad c] \end{aligned} \qquad (5.17)$$

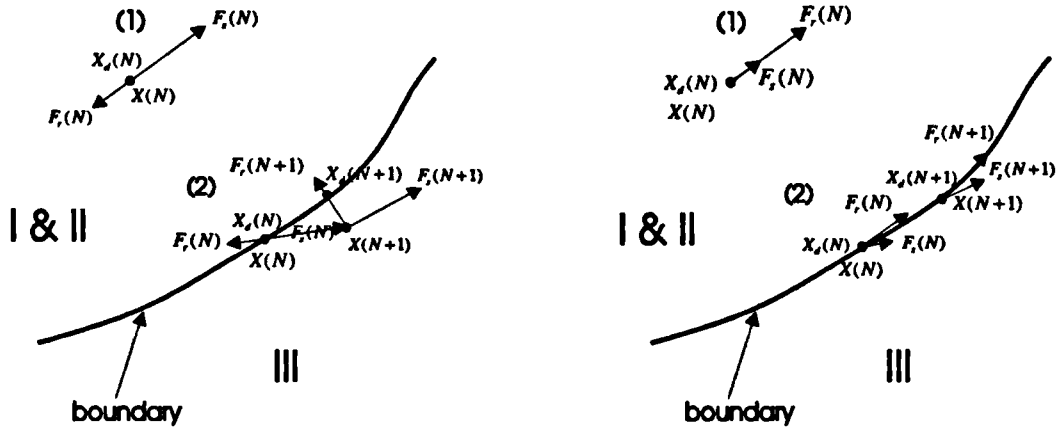
or

$$\begin{aligned} x_d &= x_1 + ka \\ y_d &= y_1 + kb \\ z_d &= z_1 + kc, \end{aligned} \qquad (5.18)$$

where $a = x_2 - x_1, b = y_2 - y_1, c = z_2 - z_1$, and $k = \dfrac{a(x_r - x_1) + b(y_r - y_1) + c(z_r - z_1)}{a^2 + b^2 + c^2}$.

5.3 Force control strategy

When the surgeon moves the cutter motor, he back-drives the various joints under the force control strategy in which the robot is constrained with a variable stiffness. The principle of the force control strategy is as shown in Fig. 5.4, where X(N) is the robot current position at time NT (T is the sampling interval of the outer computation loop). $X_d(N)$ and $F_r(N)$ are the desired position (designed as above) and the force output of the robot, respectively, while $F_s(N)$ is the surgeon's guiding force from time NT to (N+1)T.



(a) No force sensor used (Implicit force control)

(b) Force sensor involved (Modified damping control)

Fig. 5.4 The principle of the force control strategy

Two main functions of the force control strategy of the robot are: providing desired stiffness in zone I and II, and assisting (or restricting) the surgeon in moving along the boundary. Fig. 5.4(a) shows the concept of the robot force F_r with respect to the guiding force F_s in different zones, when a force sensor is not used. In zone I and II (case (1)), the direction of F_r whose magnitude depends on the desired stiffness is opposite to F_s. In case (2), suppose the initial position of the robot is at the boundary and moved by F_s, F_r will provide a designed stiffness from time NT to (N+1)T. At time (N+1)T, the 'interpreter' program will update the desired position and thus the

control algorithm will force the robot back to the boundary. If a force sensor is involved, the guiding force can become a force command to the robot. Fig. 5.4(b) shows the sketch of the forces. The robot is now in an 'active' mode, which follows F_s to move along the desired direction and its velocity depends on the desired stiffness.

As discussed in the chapter 3, in general, the control algorithms can be divided into joint based control and cartesian based control. However, the former method is preferred because it can attain a higher bandwidth of the closed loop system and thus increase the stability and disturbance rejection ability. Figure 5.5 shows the block diagram of the force control system in joint space. The system is divided into two separate and parallel computation cycles. The inner loop implements the control law with a faster sampling rate, and the outer loop updates the desired positions and the gains of the control law at the end of every sampling cycle. The design method of the desired position has been discussed in the last section. In the following, different force control strategies which implement the design concept in Fig. 5.4 will be investigated.

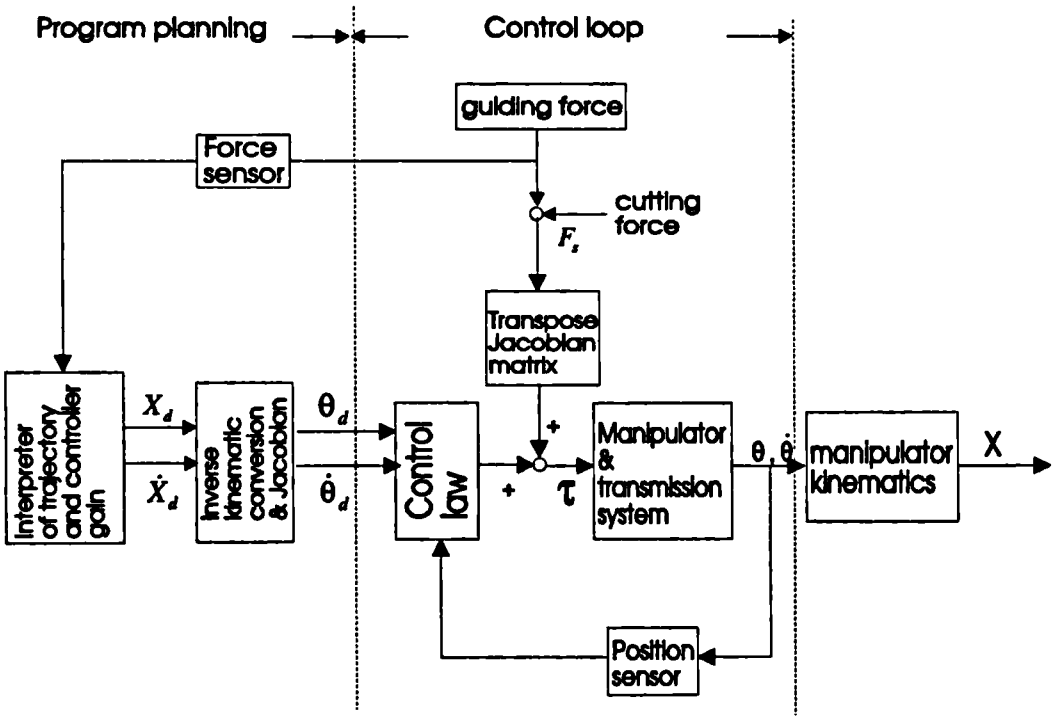


Fig 5.5 Joint based force control scheme

The force control strategies in this thesis are classified as in figure 5.6. The difference between the implicit force control and the modified damping control strategy is that the latter method uses a force sensor to measure the surgeon's guiding force and is used to design the desired velocity for the robot.

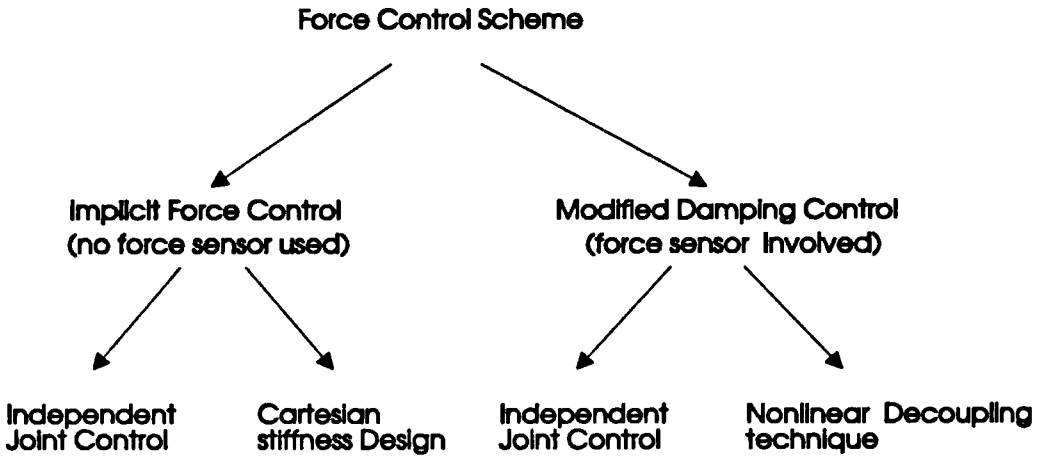


Fig. 5.6 Range of possible force control methods for robot assisted surgery

5.3.1 Implicit force control

When a force sensor is not used, the motion of the robot is 'passive' in a sense that the robot is mainly used to modify the movement by the surgeon. The robot also simulates the function of a variable spring-damper system which is changed on the basis of the robot current position and direction of motion. In order to achieve this, one efficient way is to use proportional and derivative feedback, (which are like a spring constant and damping ratio,) and adjust the control gains.

5.3.1.1 Independent joint control

The dynamic equation of the robot in joint space can be written in the general form,

$$M(\theta)\ddot{\theta}+h(\theta,\dot{\theta})+f(\theta,\dot{\theta})+c(\theta)=\tau+J^{T}F_{s}, \quad (5.19)$$

where τ = n×1 motor output torque vector,

J = system Jacobian matrix which is defined as $\frac{\partial X}{\partial \theta}$ where X = Cartesian position ,

F_s = operator's guiding force,

$\theta, \dot{\theta}, \ddot{\theta}$ = n×1 joint position, velocity and acceleration vector respectively,

$M(\theta)$ = n×n inertial matrix of the manipulator,

$h(\theta, \dot{\theta})$ = n×1 non-linear centrifugal and Corilois force vector,

$f(\theta, \dot{\theta})$ = n×1 cutting force and friction force vector,

$c(\theta)$ = n×1 gravity force vector.

If the friction is modelled as Coulomb plus joint viscous forces, it can be represented by

$$\tau_{friction} = c_f \, \mathrm{sgn}(\dot{\theta}) + v_1 \dot{\theta} \quad , \qquad (5.20)$$

where c_f = Coulomb friction constant; $\mathrm{sgn}(\dot{\theta})$= sign of the velocity $\dot{\theta}$ and v_1 = viscous force constant. Assume the cutting force F_c is proportional to the robot Cartesian velocity $\dot{X}$, resultant torques τ_c in each joint due to F_c can be expressed in joint space by

$$\begin{aligned} \tau_c &= J^T F_c \\ &= J^T (v_2 \dot{X}) \\ &= J^T (v_2 J \dot{\theta}) \\ &= v_2 J^T J \dot{\theta}, \end{aligned} \qquad (5.21)$$

where v_2 is the cutting force factor. Therefore, $f(\theta, \dot{\theta})$ is equal to

$$\begin{aligned} f(\theta, \dot{\theta}) &= \tau_{friction} + \tau_c \\ &= c_f \, \mathrm{sgn}(\dot{\theta}) + v_1 \dot{\theta} + v_2 J^T J \dot{\theta}. \end{aligned} \qquad (5.22)$$

The implicit force control law can be designed as,

$$\tau = K_p (\theta_d - \theta) - K_d \dot{\theta}, \qquad (5.23)$$

where K_p is the diagonal proportional feedback gain; K_d is the diagonal derivative feedback gain; and θ_d is the desired position. Substituting (5.23) into (5.19) yields,

$$M(\theta)\ddot{\theta} + h(\theta, \dot{\theta}) + f(\theta, \dot{\theta}) + c(\theta) = K_p (\theta_d - \theta) - K_d \dot{\theta} + J^T F_s \qquad (5.24)$$

which can be expressed as,

$$M(\theta)\ddot{e} + (K_d + v_1 I + v_2 J^T J)\dot{e} + K_p e = J^T F_s + T_{int}, \qquad (5.25)$$

where $e=\theta-\theta_d$, I = identity matrix and $T_{int}=-(h(\theta,\dot{\theta})+c(\theta)+c_f sgn(\dot{\theta}))$. The desired position vector θ_d is designed on the basis of the robot current position and the motion constraint as described in section 5.2. Through designing the control gains K_p and K_d, the desired variable stiffness can be obtained. For instance, if the constraint is defined as in Fig. 5.1, the control gains for each joint can be designed as in Fig. 5.7, where D_II and d are defined as in Fig. 5.1.

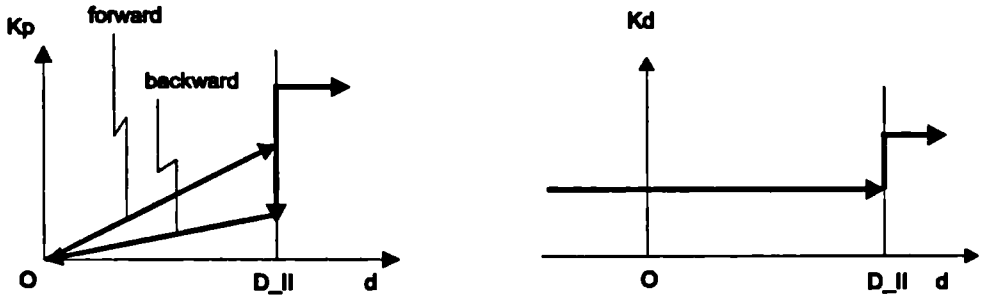


Fig. 5.7. Design of control gains for region constraint

The advantage of independent joint control is that it is the easiest method to implement. However, the drawbacks are

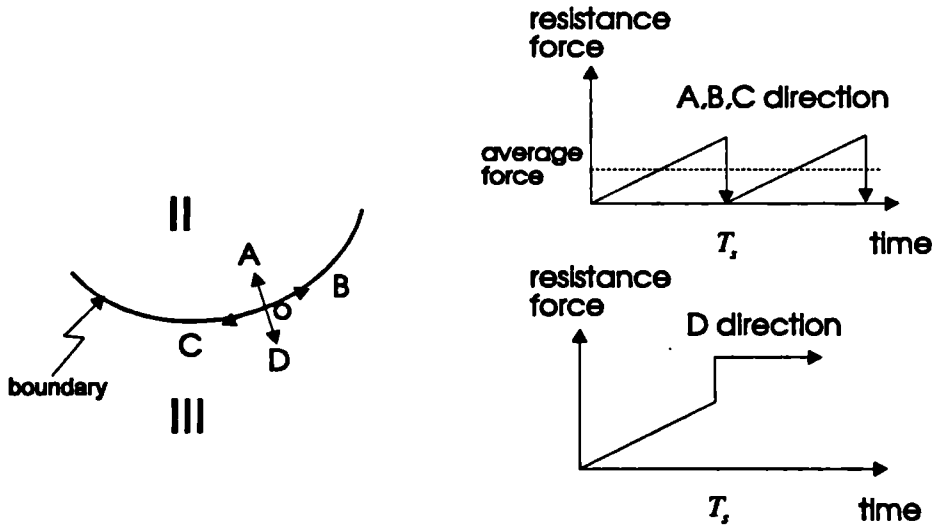


Fig. 5.8. The resistance force in different directions along the boundary by using an independent joint method of implicit force control

(i), it is difficult to design the desired stiffness in Cartesian space, especially at the boundary of the motion constraint. For instance, figure 5.8 shows the resistance

force when the robot is moving in different directions along the boundary, where T_s is the computation time for the outer trajectory interpreter loop. At the boundary, in order to have a quick response and also reduce the position error (defined as the distance over the zone II&III boundary), the control gain will be set high. However, this will also increase the resistance force and make it difficult to move the robot along the boundary.

(ii), the dynamic interaction forces are not taken into account, which can be seen as a disturbance that will produce position errors especially when the robot is moving at high speed.

5.3.1.2 Cartesian stiffness design

If the robot dynamic equation is expressed in Cartesian space, which can be represented by

$$M_x(\theta)\ddot{X} + h_x(\theta,\dot{\theta}) + f_x(\theta,\dot{\theta}) + c_x(\theta) = F_r + F_s, \qquad (5.26)$$

where F_s = operator's guiding force,

F_r = robot output force,

$\theta,\dot{\theta}$, = n×1 joint position and velocity vector respectively,

$\ddot{X}$ = n×1 robot Cartesian acceleration,

$M_X(\theta)$ = n×n Cartesian inertial matrix ,

$h_X(\theta,\dot{\theta})$ = n×1 non-linear velocity-related force vector (including Coulomb friction force),

$f_X(\theta,\dot{\theta})$ = viscous force and cutting force,

$c_X(\theta)$ = n×1 gravity force vector.

Comparing (5.26) to (5.19), the parameters of both equations have the following relationships

$$
\begin{aligned}
&M_x(\theta) = J^{-T} M(\theta) J^{-1}, \\
&h_x(\theta,\dot{\theta}) = J[h(\theta,\dot{\theta}) + c\,\mathrm{sgn}(\dot{\theta}) - M(\theta)J^{-1}\dot{J}\dot{\theta}], \\
&f_x(\theta,\dot{\theta}) = v_1 J^{-T} J^{-1}\dot{X} + v_2\dot{X}, \\
&c_x(\theta) = J^{-T} c(\theta), \\
&F_r = J^{-T}\tau,
\end{aligned}
\qquad (5.27)
$$

The control law can be given by

$$
F_r = K_{pc}(X_d - X) - K_{dc}\dot{X} + M_x^*(\theta)\ddot{X}_d + h_x^*(\theta,\dot{\theta}) + c_x^*(\theta), \qquad (5.28)
$$

where K_{pc} is the diagonal proportional feedback gain; K_{dc} is the diagonal derivative feedback gain; X_d is the reference position and X is the robot current position. Substituting (5.28) into (5.26), if $M_x^*(\theta) = M_x(\theta)$, $h_x^*(\theta,\dot{\theta}) = h_x(\theta,\dot{\theta})$ and $c_x^*(\theta) = c_x(\theta)$, it yields,

$$
M_x(\theta)\Delta\ddot{X} + (K_{dc} + v_1 J^{-T} J^{-1} + v_2)\Delta\dot{X} + K_{pc}\Delta X = F_s, \qquad (5.29)
$$

Through designing the control gains K_{pc} and K_{dc}, the desired Cartesian stiffness can be achieved. For instance, if $K_{pc} = \begin{bmatrix} K_{px} & 0 \\ 0 & K_{py} \end{bmatrix}$ are the control gains in a two-dimensional motion constraint (see Fig. 5.1) then: In zone I, K_{px} and K_{py} can be both set as 0; In zone II, K_{px} is also equal to K_{py}, but the value can be varied in a similar way to the design of gains in independent joint control (see Fig. 5.7). At the boundary, if the X-axis is parallel to the boundary, the stiffness along the X-axis can be reduced by giving a small K_{px} while giving a high K_{py}. Therefore, the robot can be more easily moved along the boundary than in other directions. In order to increase the sampling rate of the inner closed loop, the control law (5.28) can be implemented in joint space. Multiplying (5.28) by the Jacobian matrix J^T, the joint output torque is given by

$$
\begin{aligned}
\tau &= J^T F_r \\
&= J^T[K_{pc}(X_d - X) - K_{dc}\dot{X}] + J^T(M_x^*(\theta)\ddot{X}_d + h_x^*(\theta,\dot{\theta}) + c_x^*(\theta))
\end{aligned}
\qquad (5.30)
$$

In equation (5.30), if the desired position Xd is very close to the robot current position, $(X_d - X)$ is equal to $J(\theta_d - \theta)$. In addition, $\dot{X}$ is equal to $J\dot{\theta}$ according to the definition of Jacobian. Rearranging equation (5.30) yields,

$$\begin{aligned}\tau &= J^T K_{pc} J(\theta_d - \theta) - J^T K_{dc} J\dot{\theta} + T_{int} \\ &= K_{p\theta}(\theta_d - \theta) - K_{d\theta}\dot{\theta} + T_{int}\end{aligned}, \qquad (5.31)$$

where $T_{int} = J^T(M_x^*(\theta)\ddot{X}_d + h_x^*(\theta,\dot{\theta}) + c_x^*(\theta))$ is the feedforward decoupling term, which can be updated by the outer computation loop at a slower sampling rate. Usually, the desired acceleration $\ddot{X}_d$ is not available and is thus removed from T_{int}. If the robot velocity is slow and the gravity force $c_x(\theta)$ is counter-balanced, T_{int} can be neglected. $K_{p\theta} = J^T K_{pc} J$ is called the joint stiffness matrix [Sailsbury 80], and $K_{d\theta} = J^T K_{dc} J$ is defined here as a joint damping matrix. It should be noticed that the joint stiffness and damping matrix are not diagonal; therefore, the position error and velocity in each joint will affect the torque commands in all other joints. However, it is this coupling effect which transforms the Cartesian stiffness and damping into joint space.

<u>note :</u>

When the robot is in zone III (see Fig. 5.1), integral feedback can be added into the control law to reduce the position error. However, the robot is usually slightly over zone II and inside zone III because of the orthogonal component of the guiding force, when it is moved along the boundary. The effect of integral feedback will increase the resistance force and could produce a disturbance motion because of integral windup. One solution is to let the integral gain be zero, except when the robot is over the boundary a defined finite distance, and also limit the upper and lower values of the output of the integral feedback.

In summary, the implicit force control strategy can be shown as in Fig. 5.9. Although the external guiding force is unknown, through designing the desired position and the control gains on-line, the robot can behave with the function of variable stiffness in different positions and in different directions of motion. In addition, an 'active' artificial motion constraint can be achieved. The independent joint control is the easiest method to implement and the computation time for the inner control law and the interpreter program is the shortest compared with other algorithms. However, it is

difficult to design the desired stiffness in Cartesian space. The Cartesian stiffness design can easily specify the desired stiffness and damping in Cartesian space and transform it into joint space. The disadvantage is that the position error and velocity in each joint will affect the output commands in all other joints, which reduces the disturbance rejecting ability of each joint.

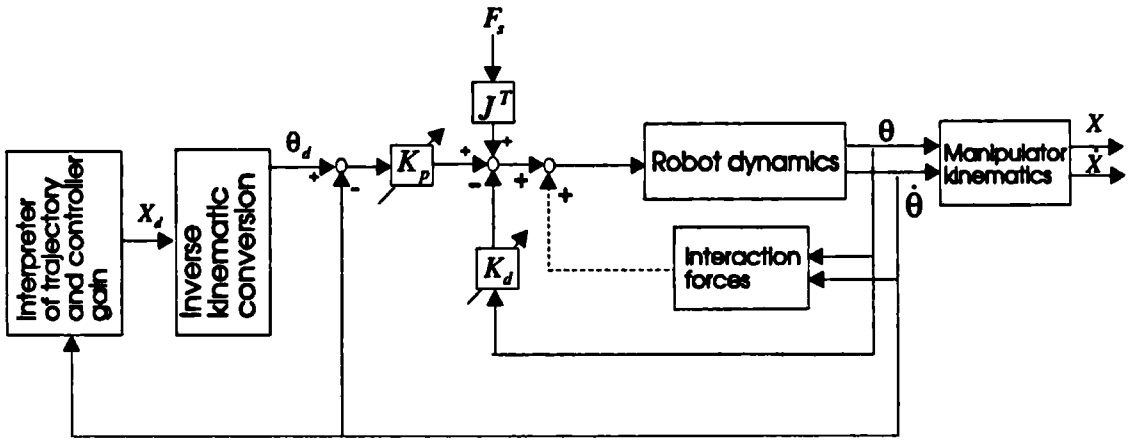


Fig. 5.9 Implicit force control strategy with an artificial motion constraint.

5.4 Modified Damping Control

There are three main advantages of using a force sensor to measure the surgeon's pushing/pulling force. Firstly, by sensing the guiding force, the control law can command the robot to follow the operator's desired movement, especially when the robot is difficult to move by hand because of high friction or mechanism design. Secondly, the Cartesian stiffness design of the implicit force control algorithm can produce desired variable stiffness and damping along the X-Y-Z axes. However, it will be impossible to have a different stiffness such as in -X and +X directions by using the implicit force control technique. Finally, the force sensor can be used as a redundant safety checking mechanism, where a 'safe' force level is pre-defined. When the external force exceeds the defined level, the power for the cutter will be shut down at once and an alarm signal is triggered. Comparing the modified control law to the implicit force control, the major difference is that the desired velocity for the robot is

designed in the first method. The design of the desired velocity is based on the relative position of the robot and motion constraint and the direction and magnitude of the guiding force. Recalling the robot dynamic equation in joint space,

$$M(\theta)\ddot{\theta}+h(\theta,\dot{\theta})+f(\theta,\dot{\theta})+c(\theta)=\tau+J^{T}F_{s}, \qquad (5.19)$$

The modified force control law can be designed as,

$$\tau=K_{p}(\theta_{d}-\theta)+K_{d}(\dot{\theta}_{d}-\dot{\theta})+M^{*}(\theta)\ddot{\theta}_{d}+h^{*}(\theta,\dot{\theta})+f^{*}(\theta,\dot{\theta})+c^{*}(\theta), \qquad (5.32)$$

where K_p is the diagonal proportional feedback gain; K_d is the diagonal derivative feedback gain; and θ_d and $\dot{\theta}_d$ are the desired position and velocity, respectively. Substituting (5.32) into (5.19) yields, (if $M^{*}(\theta)=M(\theta)$, $h^{*}(\theta,\dot{\theta})=h(\theta,\dot{\theta})$, $f^{*}(\theta,\dot{\theta})=f(\theta,\dot{\theta})$ and $c^{*}(\theta)=c(\theta)$)

$$M(\theta)\ddot{e}+K_{d}\dot{e}+K_{p}e=J^{T}F_{s}, \qquad (5.33)$$

where $e=\theta-\theta_{d}$. if the desired velocity $\dot{\theta}_d$ is designed, the motion of the robot is dominated by the control algorithm. The guiding force becomes mainly a command input to the control algorithm instead of a direct driving force as in the implicit force control algorithm. The design of the desired position is the same as for implicit force control, while the design method of the desired velocity is described below.

When the robot is in zone I or II (see Fig. 5.1), the robot will follow the surgeon's guiding force. Thus, the desired velocity can be designed as

$$\dot{X}_{d}=K_{f}F_{s}, \qquad (5.34)$$

where F_s is the surgeon's guiding force, K_f is a scalar force feedback gain and $\dot{X}_d$ is the desired velocity. The desired Cartesian velocity $\dot{X}_d$ can be transformed into joint space by a resolved motion rate control technique [Fu 87, Whitney 91]

$$\begin{aligned}\dot{\theta}_{d}&=J^{-1}\dot{X}_{d}\\&=J^{-1}K_{f}F_{s},\end{aligned} \qquad (5.35)$$

where J is the system Jacobian matrix and $\dot{\theta}_d$ is the desired velocity vector in joint space. The feedback gain K_f will determine the sensitivity of the external force command (or stiffness) and can be decided empirically. When the robot is moved along the boundary between zone II and III, the control law will be designed to help the surgeon to move along the boundary or back to the zone II easily. In addition, the robot will be prevented from moving into zone III by the control algorithm.

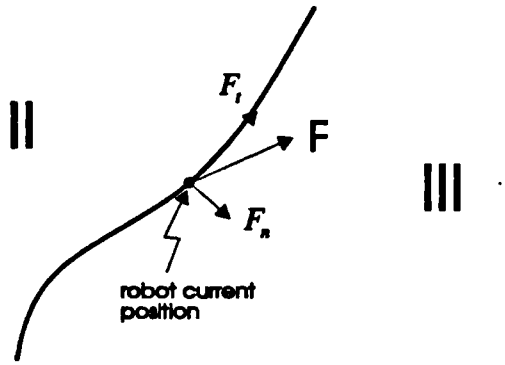


Fig. 5.10 The external force diagram when the robot is at the boundary.

As shown in Fig. 5.10, the surgeon's guiding force $\vec{F}_s$ is divided into the tangential force $\vec{F}_t$ and the normal force $\vec{F}_n$. Assume the tangential slope at the robot current position is $\vec{t}$, (which can be found from the look-up table described in section 5.1) the tangential force $\vec{F}_t$ and the normal force $\vec{F}_n$ are given by

$$\begin{aligned} \vec{F}_t &= \vec{F}_s \cdot \vec{t} \\ \vec{F}_n &= \vec{F}_s - \vec{F}_t. \end{aligned} \qquad (5.36)$$

When the robot is being moved back into zone II or along the boundary, i.e. the determinant of $\|\vec{F}_n \times \vec{t}\|$ satisfies

$$\|\vec{F}_n \times \vec{t}\| \le 0, \qquad (5.37)$$

the desired velocity can be given the same as equation (5.35). Otherwise, the desired velocity design can be designed in two parts. The first component is to follow the tangential force $\vec{F}_t$, (which is designed in a similar manner as (5.35)) and yields

$$\dot{\theta}_{d1} = J^{-1} K_f \vec{F}_t, \qquad (5.38)$$

where $\dot{\theta}_{d1}$ is the desired velocity vector in joint space according to the tangential force. In order to prevent the robot being moved into zone III, the robot will provide the equivalent force to eliminate the normal force $\bar{F}_n$. By using the Jacobian matrix, the required torque is

$$\tau = -J^T \bar{F}_n \,. \qquad (5.39)$$

From the control law of (5.31), the required torque to compensated $\bar{F}_n$ is equal to

$$\tau = K_d \dot{\theta}_{d2}, \qquad (5.40)$$

where $\dot{\theta}_{d2}$ is the desired velocity. Combining (5.39) with (5.35), the desired velocity in order to eliminate the normal force $\bar{F}_n$ is equal to

$$\dot{\theta}_{d2} = -K_d^{-1} J^T \bar{F}_n. \qquad (5.41)$$

Adding (5.41) into (5.38), the total desired velocity in joint space is

$$\begin{aligned} \dot{\theta}_d &= \dot{\theta}_{d1} + \dot{\theta}_{d2} \\ &= J^{-1} K_f \bar{F}_t - K_d^{-1} J^T \bar{F}_n. \end{aligned} \qquad (5.42)$$

However, it should be noted that the design of $\dot{\theta}_{d2}$, in order to eliminate the normal force $\bar{F}_n$, may cause a shaking movement because of the error of the force sensor and the error of mathematical model. One alternative method is to remove $\dot{\theta}_{d2}$ from (5.42) and compensate $\bar{F}_n$ through position control design as in implicit force control.

The general diagram of the modified damping control strategy is shown in Fig. 5.11. In a similar way to the implicit force control, the interaction forces are updated by the outer computation loop at a slower sampling rate, which can be neglected if the robot velocity is slow and the gravity force is counter-balanced. K_p and K_d are designed in such a way that the system will be overdamped. Designing the desired velocity through adjusting the force feedback gain K_f, the robot can readily have the desired stiffness.

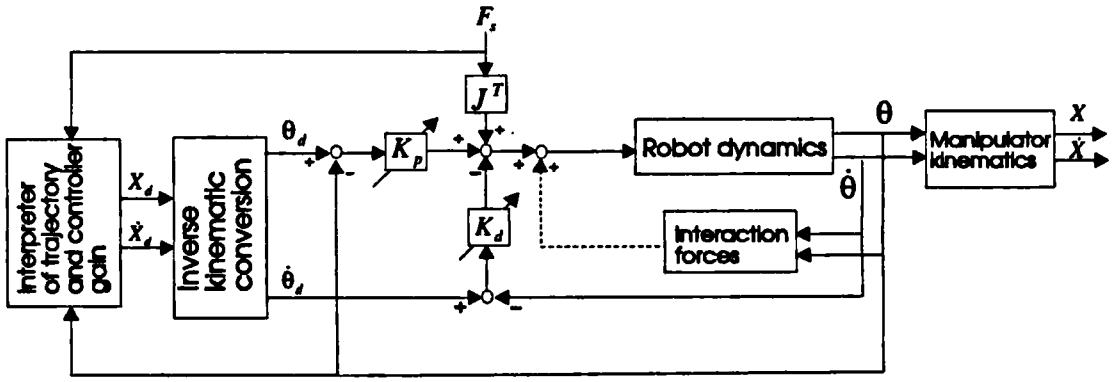


Fig. 5.11 Modified damping control strategy with an artificial motion constraint.

5.4 Stability analysis

Stability of the force control system is the most critical issue in the overall operation. Therefore, following the above analyses of force control strategies, the stability of these algorithms will be examined. In chapter 3, different force control algorithms have been reviewed. It will be recalled that the contact force between the environment and the robot is normally modelled as (see Fig. 3.14 and 3.15)

$$F_c = K_E(X - X_E), \qquad (5.43)$$

where F_c is the contact force, X is the robot actual position, X_E is the environment position and K_E is the combined stiffness of the robot and the environment. This contact force can be considered as a very stiff spring force. Therefore, if the contact force is feedback to the control system, from the stability point of view, force feedback acts similarly to a very high gain position feedback, which underdamps the system. The overall system is still stable from the analysis of a linear and perfect model, because poles of the system are still in the left half plane. However, the model of a real system is always non-linear and imperfect. As a result, the system may be unstable due to unmodelled high frequency dynamics [An 87a/b]. Possible sources of instability include:

- link flexibility;
- bandwidth limits of actuators,

- insufficient sampling rate,
- impact contact force,
- stiction and Coulomb friction,
- inappropriate control algorithm,
- inverse kinematic transformation (kinematic instability), and
- dynamics of force sensors etc.

However, these possible causes of instability are not equally important in different robotic systems [Patarinski 93]. Several papers which analyzed the stability problems have been published [Hogan 85a/b, Qian 92, Vukobratovic, Waibel 91, Wen 91]. Although most analyses are based on a linear one DOF model, or on a linearized model of the robot along a desired trajectory, the results still provide an insight into the problems. Some interesting conclusions are described below.

(1). The force feedback gain must decrease as the stiffness of the environment increases [Kazerooni 88, Maples 86, Whitney 91, Waibel 91]. In other words, a fast force feedback response needs a more compliant environment. [Whitney 91] analyzed a simple model of a discrete-time controlled system with linear force feedback. Simplifying the analysis by removing the derivative feedback, he derived the following stability condition

$$0 < T_s K_E K_F < 1, \quad (5.44)$$

where T_s is the sampling interval of the closed loop system, K_E is the stiffness of the environment and sensor and K_F is the force feedback gain. For fixed T_s, equation (5.44) indicates that the higher force feedback gain can be used if the environment is more compliant. Equation (5.44) also implies that reducing T_s can increase the upper limit of K_F. However, the response is also limited by bandwidth of the hardware. Thus , there is a T_s such that smaller T_s will not increase the system response.

(2). Stiction friction can cause limit cycle oscillations in a force control system, while Coulomb friction can extend the system stability bounds but may lead to an input independent stability [Townsend 87].

(3). The instability due to the inverse kinematic transformation is very dependent on the geometry of a manipulator, and different control algorithms will have different levels of problems [An 87a/b]. For instance, a hybrid force/position control method when implemented on a revolute manipulator will produce instability. However, the same controller results in a stable system when performed on a polar manipulator. On the other hand, the stiffness control algorithm has no kinematic instability problem because it uses the Jacobian transpose for coordinate transformations.

(4). Damping plays a key role in system stability. If damping exists in a system, the instability due to insufficient sampling rate in a discrete-time force control system may be avoided by using a low pass filter with a cut-off frequency sufficiently below the system open loop resonance frequency [Qian 92].

In the previous analyses of our control algorithms in section 5.3, for the sake of simplicity, two assumptions have been made:

(i). the cutting force is linearly proportional and only to the robot Cartesian velocity, and

(ii). the surgeon's guiding force is independent of the robot position and velocity.

Consequently, both the cutting force and guiding force do not explicitly come into the closed loop algorithm. In other words, the system stability is not affected by the external forces no matter how stiff the environment is. In the analysis below, the cutting force will be modified as dependent on both velocity and position, which is the case when an impact force upon cutting occurs. In addition, the case when the guiding force is related to the robot position and the environment will be discussed.

To simplify the analysis, a single-axis model is examined again. Fig 5.12 shows the schematic diagram of the system. The dynamic equation for this system can represented by the expression

$$M\ddot{X} = F_s + F_r + F_c, \qquad (5.45)$$

where M is the mass, F_s is the surgeon's guiding force, F_r is the output force of the robot and F_c is the cutting force.

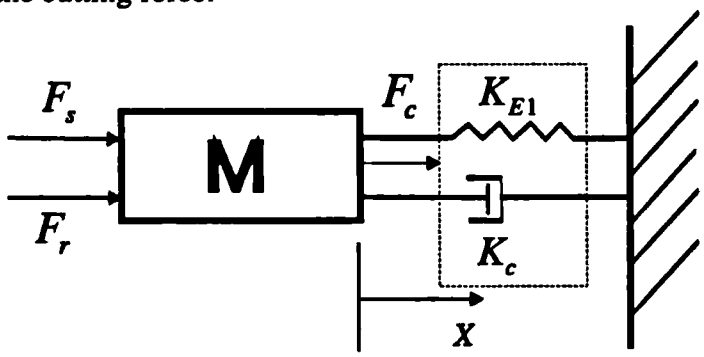


Fig. 5.12 Model of a single-axis system and the environment

The cutting force F_c as shown in Fig. 5.12 is modelled as

$$F_c = -K_c\dot{X} - K_{E1}(X - X_{E1}), \qquad (5.46)$$

where K_c is the cutting force constant, $\dot{X}$ is the robot velocity, X is the robot position, X_{E1} is the environment position and K_{E1} is the stiffness between the cutter and the environment. The control law as analyzed in chapter four is give by

$$F_r = K_p(X_d - X) + K_d(\dot{X}_d - \dot{X}). \qquad (4.48)$$

It should be noted that $\dot{X}_d$ is equal to zero if an implicit force control algorithm is applied. Substituting (4.48) and (5.46) into (5.45) yields

$$\begin{aligned} &M\ddot{X} + (K_d + K_c)\dot{X} + (K_p + K_{E1})X \\ &= F_s + K_pX_d + K_d\dot{X}_d + K_{E1}X_{E1}. \end{aligned} \qquad (5.47)$$

It can be seen that the environment stiffness K_{E1} will increase the natural frequency and thus reduce the damping of the system.

In a modified damping control system, $\dot{X}_d$ is designed on the basis of F_s and the robot current position with respect to the motion constraint (see section 5.3.2). In general, $\dot{X}_d$ is given by

$$\dot{X}_d = K_fF_s, \qquad (5.48)$$

where K_f is the force feedback. Assuming F_s is a function of the robot position (for instance, the robot is against a very hard environment and the operator keeps pushing toward that environment), and can be represented by

$$F_s = -K_{E2}(X - X_{E2}), \qquad (5.49)$$

where X_{E2} is the position of the operator's hand which holds the robot and K_{E2} is the overall stiffness of the cutter, the force sensor and the operator's hand. Combing (5.49) with (5.48) attains

$$\dot{X}_d = -K_f K_{E2}(X - X_{E2}). \qquad (5.50)$$

Substituting (5.50) into (5.47) yields

$$\begin{aligned} &M\ddot{X} + (K_d + K_c)\dot{X} + (K_p + K_{E1} + K_d K_f K_{E2})X \\ &= F_s + K_p X_d + K_{E1} X_{E1} + K_d K_f K_{E2} X_{E2}. \end{aligned} \qquad (5.51)$$

From equation (5.51), the following conclusions can be drawn:

(1). In implicit force control, the force feedback K_f is zero. As a result, the guiding force does not affect the characteristic equation of the system. While the stiffness between the cutter and the environment decrease the system damping.

(2). In modified damping control, K_{E2} is multiplied by K_d and K_f if the guiding force is dependent on the robot position. This will result in very high gain 'position' control, and will under damp the overall system. Increasing K_f will increase the sensitivity of the guiding force, in other words, which makes the robot more compliant in following the force command. However, on the other hand, this will decrease the system damping and thus reduce the ability to reject disturbance forces or the impact force upon cutting.

(3). From experimental experience, limiting the upper value of the force input F_s to the control law can increase the stability of the system. From another point of view, setting the upper limit of F_s means that the stiffness K_{E2} in (5.51) is artificially reduced in cases of a high external force.

5.5 Conclusion

In this chapter, the force control strategy with an artificial motion constraint for multi-DOF manipulators has been analyzed. The problems of the overall control system can be divided into three parts:

(i). representation of the motion constraint,

(ii). the design algorithm of the desired position, and

(iii). the force control strategy.

In section 5.1, the method and procedures of constructing a look-table, which represents the motion constraint in a computer, has been described. Base on the look-up table of the motion constraint, an effective on-line algorithm of designing the desired position of the robot has then been proposed in section 5.2.

In section 5.3.1, the implicit force control and modified damping control algorithms have been analyzed. The design of implicit force control in which a force sensor is not involved can be divided into independent joint control method and Cartesian stiffness design. The independent joint control is the easiest and also the least computational time consuming algorithm when implemented. However, it is difficult to design the desired stiffness in Cartesian space. On the other hand, Cartesian control can easily specify the desired stiffness in Cartesian space. Nevertheless, the position error and velocity in each joint will affect the output commands in all other joints, which reduces the disturbance rejecting ability of each joint.

In section 5.3.2, modified damping control, which involves a force sensor in the system, is investigated. There are three main advantages of using a force sensor to measure the surgeon's pushing/pulling force. Firstly, by sensing the guiding force, the control law can command the robot to follow the operator's desired movement, especially when the robot is difficult to move by hand because of high friction or the mechanism design. Secondly, the Cartesian stiffness design of the implicit force control algorithm can produce desired variable stiffness and damping along the X-Y-Z axes. However, it will be impossible to have a different stiffness such as in -X and +X

directions by using the implicit force control technique. Finally, the force sensor can be used as a redundant safety checking mechanism, where a 'safe' force level is pre-defined. There are also two design methods in the modified damping control algorithm: independent joint control and non-linear decoupling technique. The first method is similar to the independent joint control method in implicit force control, except that a desired velocity for the robot can be designed. If the robot is moving at high speed, the non-linear decoupling technique may have to be used to reduce the position errors caused by the decoupled forces among the links.

Stability of the force control system is the most critical issue in the overall operation. In section 5.4, stability of our proposed force control algorithms has been analyzed. It has been concluded that (1) The guiding force does not affect the characteristic equation of the system in an implicit force control, while the stiffness between the cutter and the environment decrease the system damping. (2). Increasing K_f will increase the sensitivity of the guiding force in modified damping control, in other words, which makes the robot more compliant in following the force command. However, on the other hand, this will decrease the system damping and thus reduce the ability to reject disturbance forces or the impact force upon cutting.

In the next chapter, computer simulations will be carried out to examine the performance of the analysis in this chapter. In chapter 7, a three-DOF robot will be built up and then experimental tests will be implemented to attain further verification of our force control strategy.

CHAPTER 6

COMPUTER SIMULATION

Following the analysis in the previous chapter, computer simulations will be performed to verify the validity of the force control strategy that has been proposed. A two DOF manipulator is taken as the example. The kinematics and dynamics of the manipulator are firstly derived. Subsequently, computer simulations of both implicit force control and modified damping control with a circular motion constraint are implemented. The performance of these two force control algorithms will be examined. The effects of sampling rates, change of inertia and cutting force to the overall system will also be discussed.

6.1 An example of a two-link planar manipulator

The two-link planar manipulator is taken as an example because its configuration is the same as that of the XY plane mechanism of our three-DOF manipulator (see chapter 7). Fig. 6.1 shows the schematic diagram of the manipulator, where the gravitational force is perpendicular to XY plane. For simplicity, it is assumed that the mass distribution of the links is homogeneous and the centroids locate at the middle of the links. Their masses are equal to m_1 and m_2, respectively. Masses of the joint motor and cutter are represented as a point mass at the distal end of each link. They are equal to M_1 and M_2 respectively.

6.1.1 Kinematics, inverse-kinematics and Jacobian of the manipulator

It is normally desirable to describe the motion of the robot in Cartesian space. Therefore, the position of the robot has to be transformed from joint space to Cartesian coordinate. As seen in Fig. 6.1, the Cartesian position of the manipulator (X_2, Y_2) and the joint positions θ_1 and θ_2 have the following relations:

$$X_2 = L_1 \cos\theta_1 + L_2 \cos(\theta_1 + \theta_2)$$
$$Y_2 = L \sin\theta_1 + L_2 \sin(\theta_1 + \theta_2) \qquad (6.1)$$

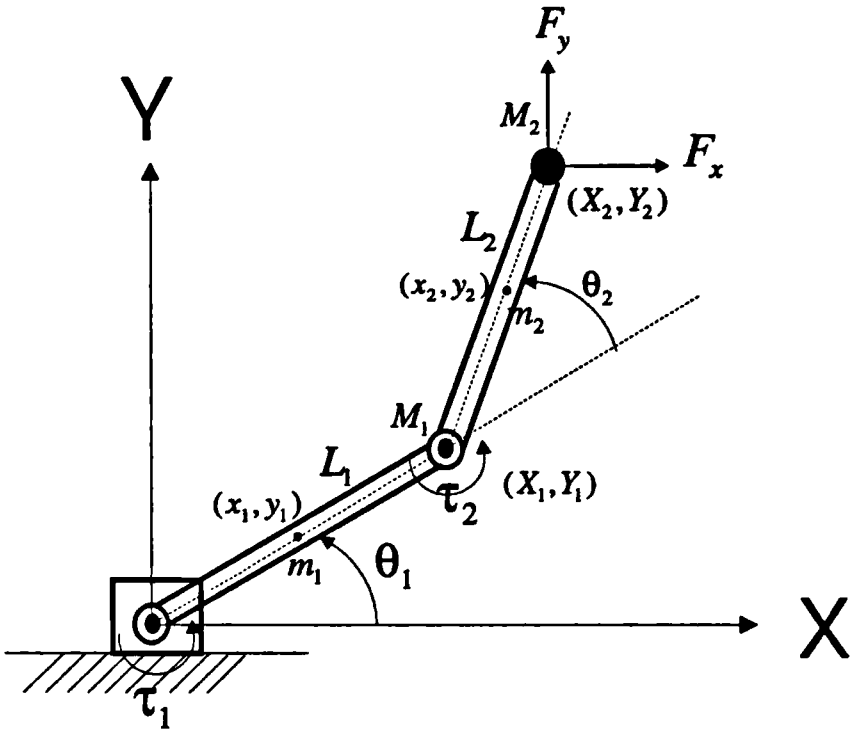


Fig. 6.1 Plane geometry of a two-link manipulator

where L_i are the length of the links of the manipulator, respectively.

When the desired Cartesian position $X_d = [x_d \quad y_d]$ and velocity $\dot{X}_d = [\dot{x}_d \quad \dot{y}_d]$ of the robot are determined by the control algorithm, they have to be transformed into desired joint angles θ_{id} and joint velocities $\dot{\theta}_{id}$. Through geometric analysis [Craig 86, Fu 87], the desired joint angles θ_{id} can be given as

$$\theta_{1d} = tan^{-1}(\frac{y_d}{x_d}) - cos^{-1}(\frac{x_d^2 + y_d^2 + L_1^2 - L_2^2}{2L_1\sqrt{x_d^2 + y_d^2}})$$
$$\theta_{2d} = cos^{-1}(\frac{x_d^2 + y_d^2 - L_1^2 - L_2^2}{2L_1L_2}). \qquad (6.2)$$

The desired joint velocities $\dot{\theta}_{id}$ can be derived by resolved motion rate control [Whitney 91] and are equal to

$$\begin{bmatrix}\dot{\theta}_{1d}\\ \dot{\theta}_{2d}\end{bmatrix} = J^{-1}(\theta)\begin{bmatrix}\dot{x}_d\\ \dot{y}_d\end{bmatrix}, \qquad (6.3)$$

where $J^{-1}(\theta)$ is the inverse of the Jacobian matrix. The Jacobian matrix is defined as $\frac{\partial X}{\partial \theta}$, where X is the Cartesian position vector and θ is the joint position vector.

Taking the partial differential of equation (6.1), the Jacobian matrix equals

$$J(\theta) = \begin{bmatrix}-L_1\sin\theta_1 - L_2\sin(\theta_1+\theta_2) & -L_2\sin(\theta_1+\theta_2)\\ L_1\cos\theta_1 + L_2\cos(\theta_1+\theta_2) & L_2\cos(\theta_1+\theta_2)\end{bmatrix}. \qquad (6.4)$$

The inverse of the Jacobian matrix is then equal to

$$J^{-1}(\theta) = \frac{1}{L_1 L_2\sin\theta_2}\begin{bmatrix}L_2\cos(\theta_1+\theta_2) & L_2\sin(\theta_1+\theta_2)\\ -L_1\cos\theta_1 - L_2\cos(\theta_1+\theta_2) & -L_1\sin\theta_1 - L_2\sin(\theta_1+\theta_2)\end{bmatrix}. \qquad (6.5)$$

6.1.2 Dynamics of the manipulator

The dynamic equations of a manipulator, generally, can be obtained by iteration of numerical computational algorithm or represented by a closed form mathematic model [Craig 86]. Use of the equations as a numerical computational algorithm is attractive because the equations can be applied directly to calculate the joint torques corresponding to any motion in the computer. However, the closed form dynamics can provide better insight to the structure of the equations. For instance, how does the change of inertia affect the performance of the system? Therefore, the dynamic equations of the manipulator will be written in closed form here.

Dynamics of the manipulator can be derived by Lagrange's equations [Goldstein],

$$\frac{d}{dt}\left[\frac{\partial(T-V)}{\partial\dot{\theta}_i}\right] - \frac{\partial(T-V)}{\partial\theta_i} = \tau_i + J^T F_s, \qquad (6.6)$$

where T is the kinetic energy; V is the potential energy; τ_i are joint torques; J^T is the transpose of the Jacobian matrix and $F_s = \begin{bmatrix}F_x & F_y\end{bmatrix}'$ is the guiding force. Procedures of

deriving the dynamic equations are described as below. The kinematic energy of the manipulator is given as

$$T=\frac{1}{2}\{M_1(\dot{X}_1^2+\dot{Y}_1^2)+M_2(\dot{X}_2^2+\dot{Y}_2^2)+I_1\dot{\theta}_1{}^2+ I_2(\dot{\theta}_1{}^2+\dot{\theta}_2{}^2)+m_1(\dot{x}_1^2+\dot{y}_1^2)+m_2(\dot{x}_2^2+\dot{y}_2^2)\}, \tag{6.7}$$

where the derivatives are time differentials. I_1 and I_2 are mass moments of the first and second links and equal to $\frac{1}{12}m_1L_1^2$ and $\frac{1}{12}m_1L_2^2$, respectively. From Fig. 6.1, the following geometrical relationships can be obtained

$$\begin{bmatrix}x_1\\y_1\end{bmatrix}=\begin{bmatrix}\frac{1}{2}L_1c_1\\\frac{1}{2}L_1s_1\end{bmatrix},$$

$$\begin{bmatrix}X_1\\Y_1\end{bmatrix}=\begin{bmatrix}L_1c_1\\L_1s_1\end{bmatrix},$$

$$\begin{bmatrix}x_2\\y_2\end{bmatrix}=\begin{bmatrix}L_1c_1+\frac{1}{2}L_2c_{12}\\L_1s_1+\frac{1}{2}L_2s_{12}\end{bmatrix} \text{ and}$$

$$\begin{bmatrix}X_2\\Y_2\end{bmatrix}=\begin{bmatrix}L_1c_1+L_2c_{12}\\L_1s_1+L_2s_{12}\end{bmatrix}, \tag{6.8}$$

where $s_i=\sin\theta_i$, $c_i=\cos\theta_i$, $s_{ij}=\sin(\theta_i+\theta_j)$ and $c_{ij}=\cos(\theta_i+\theta_j)$. When taking the time differentiation of equation (6.8) yields,

$$\begin{bmatrix}\dot{x}_1\\\dot{y}_1\end{bmatrix}=\begin{bmatrix}-\frac{1}{2}L_1s_1\dot{\theta}_1\\\frac{1}{2}L_1c_1\dot{\theta}_1\end{bmatrix},$$

$$\begin{bmatrix}\dot{X}_1\\\dot{Y}_1\end{bmatrix}=\begin{bmatrix}-L_1s_1\dot{\theta}_1\\L_1c_1\dot{\theta}_1\end{bmatrix},$$

$$\begin{bmatrix}\dot{x}_2\\\dot{y}_2\end{bmatrix}=\begin{bmatrix}-L_1s_1\dot{\theta}_1-\frac{1}{2}L_2s_{12}(\dot{\theta}_1+\dot{\theta}_2)\\L_1c_1\dot{\theta}_1+\frac{1}{2}L_2c_{12}(\dot{\theta}_1+\dot{\theta}_2)\end{bmatrix} \text{ and}$$

$$\begin{bmatrix}\dot{X}_2\\\dot{Y}_2\end{bmatrix}=\begin{bmatrix}-L_1s_1\dot{\theta}_1-L_2s_{12}(\dot{\theta}_1+\dot{\theta}_2)\\L_1c_1\dot{\theta}_1+L_2c_{12}(\dot{\theta}_1+\dot{\theta}_2)\end{bmatrix}. \tag{6.9}$$

Substituting (6.9) into (6.7), the kinetic energy of the system is equal to

$$T=[(\frac{1}{2}M_1+\frac{1}{2}M_2+\frac{1}{6}m_1+\frac{1}{2}m_2)L_1^2+(\frac{1}{2}M_2+\frac{1}{6}m_2)L_2^2+(M_2+\frac{1}{2}m_2)L_1L_2c_2]\dot{\theta}_1^{\,2}$$
$$+(\frac{1}{2}M_2+\frac{1}{6}m_2)L_2^2\dot{\theta}_2^{\,2}+[(M_2+\frac{1}{2}m_2)L_2^2+(M_2+\frac{1}{2}m_2)L_1L_2c_2]\dot{\theta}_1\dot{\theta}_2. \qquad (6.10)$$

The potential energy of the manipulator is equal to zero because the manipulator only moves in the horizontal plane. Therefore, the Lagrange's equations (6.6) can be rewritten as

$$\frac{d}{dt}(\frac{\partial T}{\partial \theta_i})-\frac{\partial T}{\partial \theta_i}=J^TF_s+\tau_i. \qquad (6.11)$$

Substituting (6.10) into (6.11), the dynamic equations of the manipulator is given by

$$\begin{bmatrix}(M_1+M_2+\frac{1}{3}m_1+m_2)L_1^2+(M_2+\frac{1}{3}m_2)L_2^2+(2M_2+m_2)L_1L_2c_2 & (M_2+\frac{1}{3}m_2)L_2^2+(M_2+\frac{1}{2}m_2)L_1L_2c_2\\ (M_2+\frac{1}{3}m_2)L_2^2+(M_2+\frac{1}{2}m_2)L_1L_2c_2 & (M_2+\frac{1}{3}m_2)L_2^2\end{bmatrix}\begin{bmatrix}\ddot{\theta}_1\\ \ddot{\theta}_2\end{bmatrix}$$
$$+\begin{bmatrix}-(2M_2+m_2)(\dot{\theta}_1\dot{\theta}_2+\frac{1}{2}\dot{\theta}_2^2)L_1L_2s_2\\ (M_2+\frac{1}{2}m_2)L_1L_2s_2\dot{\theta}_1^2\end{bmatrix}=\begin{bmatrix}-(L_1s_1+L_2s_{12}) & L_1c_1+L_2c_{12})\\ -L_2s_{12} & L_2c_{12}\end{bmatrix}\begin{bmatrix}F_x\\ F_y\end{bmatrix}+\begin{bmatrix}\tau_1\\ \tau_2\end{bmatrix},$$

or

$$M(\theta)\ddot{\theta}+V(\theta)=J^TF_s+\tau. \qquad (6.12)$$

$M(\theta)$ is the inertia matrix and $V(\theta)$ is the centrifugal and Corilois force vector. In equation (6.12), friction force and cutting resistance are not included. If the friction force τ_{fric} is modelled as Coulomb friction, friction τ_{fric} can then be represented as described in the chapter 5 by

$$\tau_{fric}=\begin{bmatrix}c_1sgn(\dot{\theta}_1)\\ c_2sgn(\dot{\theta}_2)\end{bmatrix}, \qquad (6.13)$$

where c_i is the Coulomb force constant; $sgn(\dot{\theta}_i)$ is equal to +1 or -1 depending on whether $\dot{\theta}_i$ is positive or negative.

If the cutting resistance force is assumed to be proportional to the Cartesian velocity of the robot, joint torques corresponding to the cutting resistance can be expressed as

$$\tau_c=v_cJ^TJ\begin{bmatrix}\dot{\theta}_1\\ \dot{\theta}_2\end{bmatrix}, \qquad (6.14)$$

where v_c is the cutting force constant.

When adding (6.13) and (6.14) into (6.12), the dynamic equation of the manipulator can then be written as

$$M(\theta)\ddot{\theta}+V(\theta)+F(\theta,\dot{\theta})=J^T F_s+\tau, \qquad (6.15)$$

where $F(\theta,\dot{\theta})=\tau_{fric}+\tau_c$.

6.2 Computer simulation

Based on the model of equation (6.15), computer simulations will be examined in this section. The general block diagram of the control system is shown in Fig 6.2. The desired position θ_d , desired velocity $\dot{\theta}_d$ and the control gains K_p and K_d are updated at a slower sampling rate T_{s1}, while the inner control law is performed at a shorter sampling interval T_{s2}. The feedforward decoupling force τ_{int} may be neglected depending on the force control strategy used (see the analysis in chapter 5). If τ_{int} is involved, it is updated after every T_{s1} second. It should be noted that time delay exists in reality because of numerical computation. Although the time delay is not shown in Fig. 6.2, it will be taken into account in the simulation program. K is the gain of the motor amplifiers and is given as 0.135. In addition to Coulomb friction (see (6.13)), stiction friction is also added into each joint in the simulation, which is equal to 0.45 Nm.

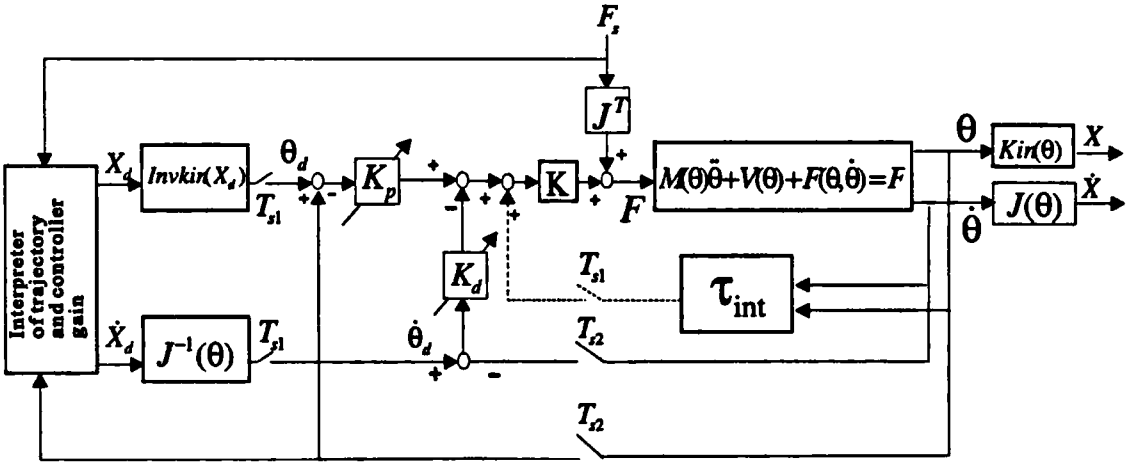


Fig. 6.2 A general block diagram of the force control strategy

The parameters of the model (6.15) are given the values similar to that found in the experimental mechanism shown in Fig. 6.1. They are listed in table 6.1, where c_i is the Coulomb force constant.

L_1	L_2	m_1	m_2	M_1	M_2	c_1	c_2
0.13	0.157	0.072	0.072	0.51	0.51	0.38	0.38

Table 6.1 Parameters of the manipulator for computer simulation

The units of length and mass are meter and kilogram, respectively. According to the values in table 6.1, the parameters of equation (6.15) are given as

$$M(\theta)=\begin{bmatrix} 0.032+0.0223c_2 & 0.0132+0.0111c_2 \\ 0.0132+0.0111c_2 & 0.0132 \end{bmatrix},$$

$$V(\theta)=\begin{bmatrix} -0.0223s_2(\dot{\theta}_1\dot{\theta}_2+\frac{1}{2}\dot{\theta}_2^{\,2}) \\ 0.0111\dot{\theta}_1^{\,2} \end{bmatrix} \text{ and}$$

$$J^T(\theta)=\begin{bmatrix} -0.13s_1-0.157s_{12} & 0.13c_1+0.157c_2 \\ -0.157s_{12} & 0.157c_{12} \end{bmatrix}. \qquad (6.16)$$

The motion constraint is given as a circle and is defined as

zone I: radius < 5mm

zone II: 5mm ≤ radius < 10mm, and

zone III: radius ≥ 10mm.

In order to simulate the motion of the manipulator which is from the centre of the constraint to the boundary and then along the boundary, the guiding force F_s is given as described below. In zones I and II, the direction of F_s is toward zone III. When the manipulator is at the boundary or in zone III, the direction of F_s becomes parallel to the boundary, but a "noise" force orthogonal to F_s is added to simulate the variation of the external force. If the current position of the manipulator equals $[x \quad y]$ and its distance to the centre of the constraint is equal to r, F_s can be described as

IF (r ≥ zone III)

$$F_s = \text{FORCE} \times \left[-\frac{y}{r} \quad \frac{x}{r}\right]' + \text{NOISE} \times \text{RAND} \times \left[\frac{x}{r} \quad \frac{y}{r}\right]'$$

ELSE

$$F_s = \text{FORCE} \times \left[\frac{x}{r} \quad \frac{y}{r}\right]', \qquad (6.17)$$

where both FORCE and NOISE are equal to 10 Newton and RAND is a random number uniformly distributed in the interval (0,1).

The simulation program is written in MATLAB, which is used because of its capability of vector and matrix calculations. The overall control system can be considered as a second-order differential equation. The numerical computational algorithm of solving the equation is base on Runge-Kutta-Nystrom method, which is more accurate compared to the Taylor expansion method [Kreyszig].

6.2.1 Implicit force control

In implicit force control, the guiding force is not measured. Therefore, the desired velocities $\dot{X}_d$ and $\dot{\theta}_d$ are equal to zero. As described in chapter 5, implicit force control includes independent joint control and Cartesian stiffness design. Their performance will be examined below.

6.2.1.1 Independent joint control

Fig. 6.4 shows the simulation result of using the independent joint control algorithm for the case where K_p and K_d of the control law for each joint are given as in Fig. 6.3; T_{s1} and T_{s2} are equal to 0.005 sec. and 0.001 sec., respectively and the cutting force equals zero. The initial configuration of the manipulator is at $\theta_1 = 0$ and $\theta_2 = 90^o$. The maximum position errors occur at positions A, B and C. At position A, the overshoot, which is about 0.6mm, is caused by kinetic energy in addition of the external force. At positions B and C, the position error, which is about 0.5mm, is bigger because the effective torques corresponding to the guiding force given as in (6.17) have maximum values at those configurations.

In order to examine the effects of the change of inertia, M_2 is increased to 1 Kg. This increases the inertia matrix of the manipulator about two times when $\theta_2 = 90^o$. Fig. 6.5 shows the computer simulation as for Fig. 6.4, but M_2 equals one kilogram. It can be seen that the rising time of the manipulator is a little slower compared to that of Fig. 6.4. However, the overshoot is slightly decreased.

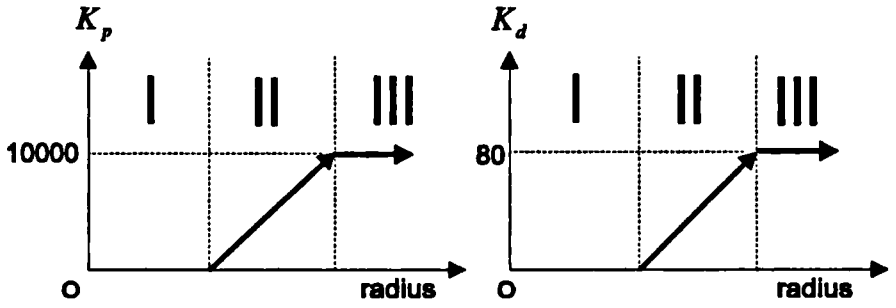


Fig. 6.3 Design of the control gains for independent joint control

As for Fig. 6.4, but a disturbing cutting force is added into the simulation program, which is given as

$$\text{cutting force} = B \times RAND \times \dot{X}, \qquad (6.18)$$

where B = 10 N/m/sec. RAND is a random number uniformly distributed in the interval (0,1) as in (6.17) and $\dot{X}$ is the Cartesian velocity of the manipulator. Fig. 6.6 shows the simulation result. It can be seen that the position error at point A is reduced. This is because the cutting force provides additional resistance and thus reduces the kinetic energy.

It has been analyzed in section 4.1 that a slow sampling frequency of the interpreter of the desired trajectory and control gains, T_{s1}, will cause the movement to become 'jerky' in operation. Theoretically, the sampling frequency of the interpreter must be at least twice the bandwidth of the external force so that the inner control law can effectively respond and attain the desired stiffness. Normally, the maximum frequency of human hand is around 3~5 Hz. However, the cutting frequency should also be taken into account. For instance, for a cutter with 6000 rpm speed, the frequency of the cutting force will be equal to 100 Hz.

Fig. 6.7 and 6.8 show simulations of T_{s1}= 0.01 and T_{s1}= 0.02, respectively. It can be seen that the movement of the manipulator becomes less smooth along the boundary. In addition, the position error becomes higher at position A. This is because that there is also a time delay of T_{s1} of the interpreter program, which decreases the response of the manipulator to its current status. The velocity of the robot corresponding to the same guiding force is also slower, if the sampling interval T_{s1} is longer. The reason is that the 'potential' energy of the manipulator due to the proportional feedback is released after every T_{s1} second. If T_{s1} is longer, then the resistance force from the controller will become higher during every sampling interval T_{s1} (see the analysis in section 5.3.1.1).

As discussed in chapter 5, integral feedback can reduce the position error in zone III. However, if the updating rate of the control gains is slow, adding of integral gain to the control law will not improve the performance. Fig. 6.9 shows the result, which is as for Fig. 6.8 but a integral feedback gain equal to 0.5 is added to the control algorithm. The results shows that the position error at point A is not reduced. The velocity of the manipulator, however, becomes even slower, because the integral gain increases the resistance force during the operation.

The sampling rate of the inner control law is more critical to the stability of the overall system. Some factors that could provide a lower limit to the acceptable sampling rate are: tracking effectiveness of reference inputs, disturbance rejection, sensitivity to plant-parameter variations and structural resonances [Craig 86, Franklin 86]. The fundamental lower bound on the sampling rate is twice the system bandwidth in order for the closed loop system to track a reference input. Fig. 6.10 and 6.11 show simulations of T_{s2} = 0.002 and T_{s2} = 0.005, respectively. It can be seen that in the extreme case when T_{s2}= 0.005, which is equal to T_{s1}, the system becomes unstable.

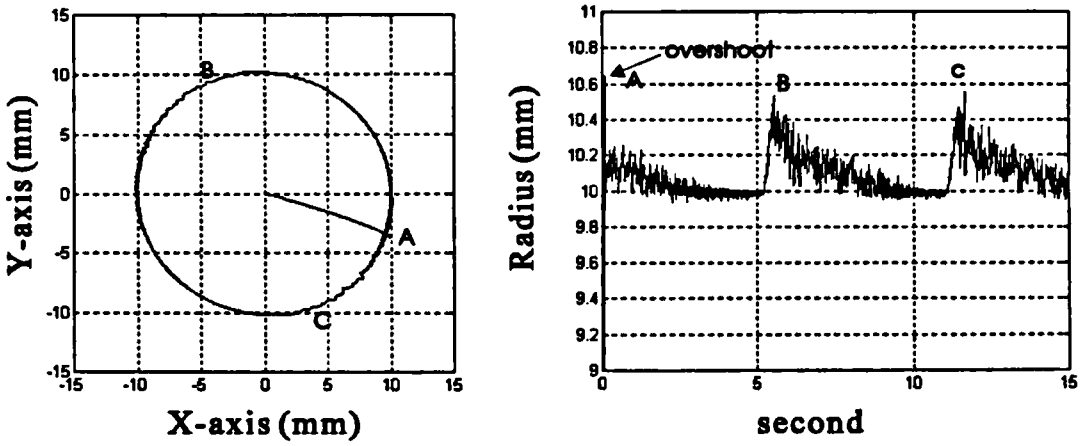


Fig. 6.4 Simulation of independent joint control when $T_{s1} = 0.005$ sec. and $T_{s2} = 0.001$ sec.(also see text)

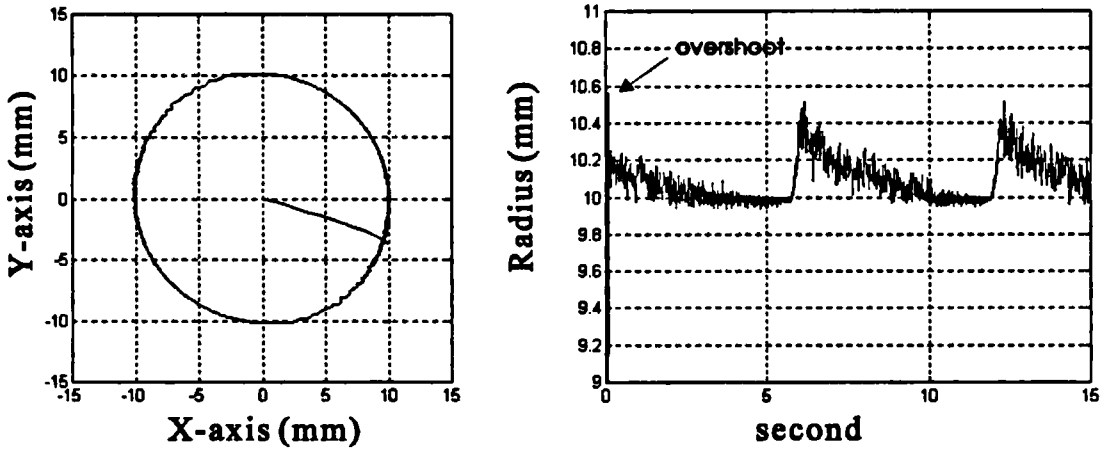


Fig. 6.5 As for Fig. 6.4, but inertia of the manipulator is increased

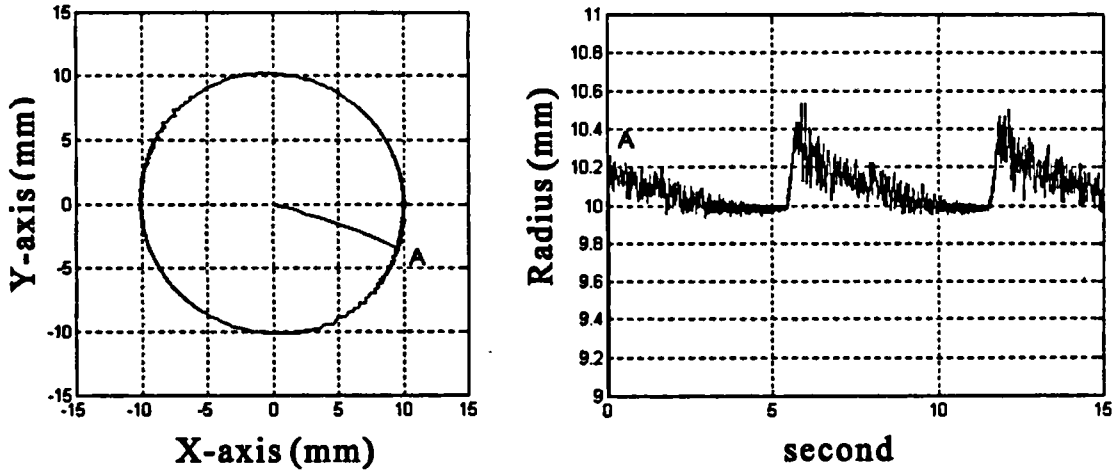


Fig. 6.6 As for Fig. 6.4 but cutting force is added to the model

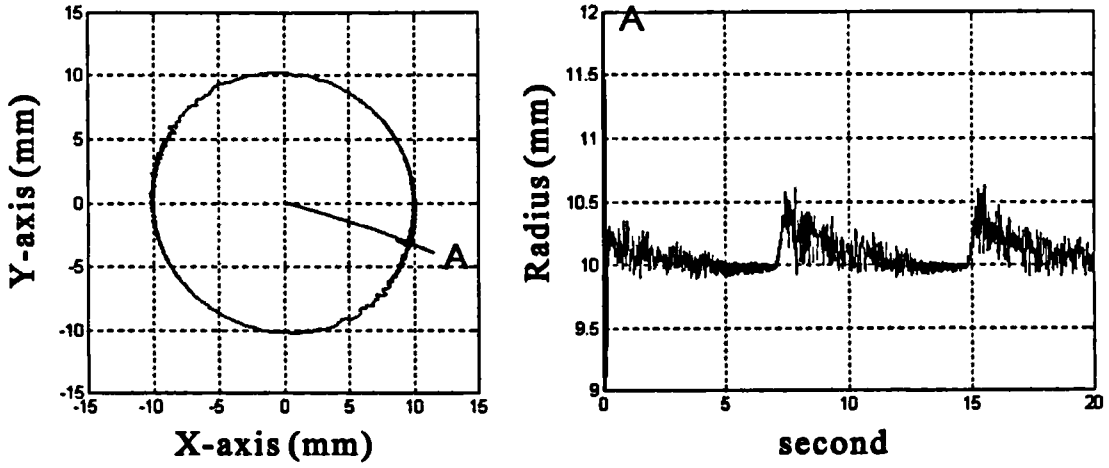


Fig. 6.7 As for Fig 6.6, but $T_{s1} = 0.01$

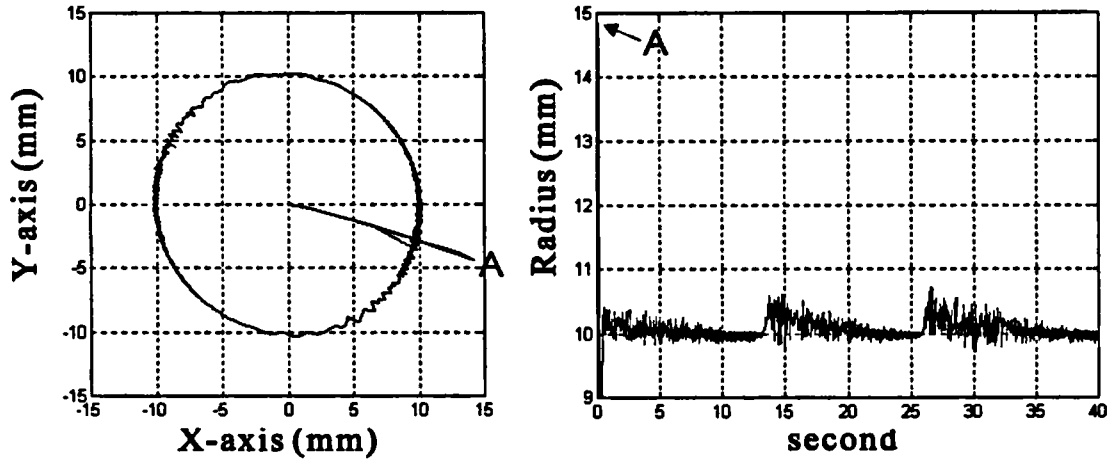


Fig. 6.8 As for Fig 6.6, but $T_{s1} = 0.02$

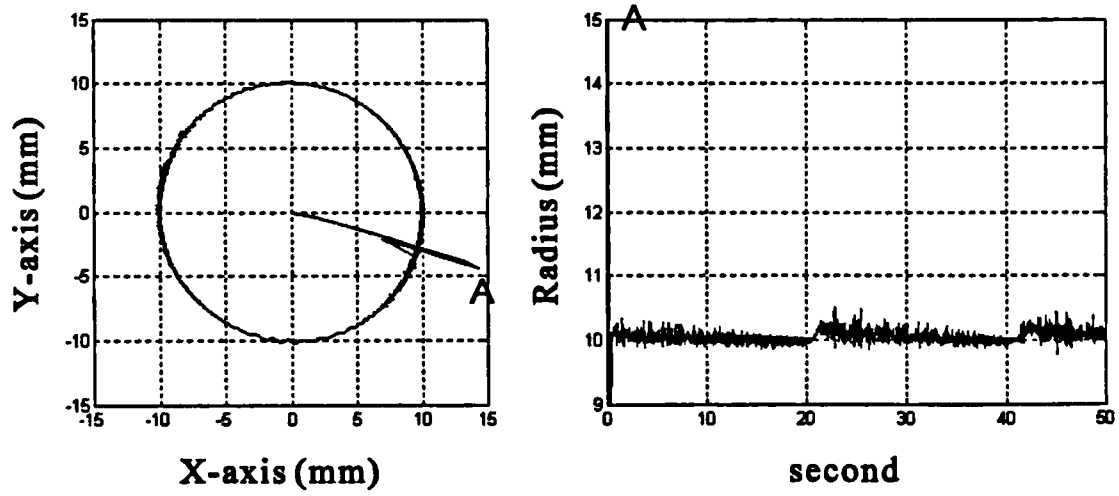


Fig. 6.9 As for Fig. 6.8, but integral feedback is added into the control law.

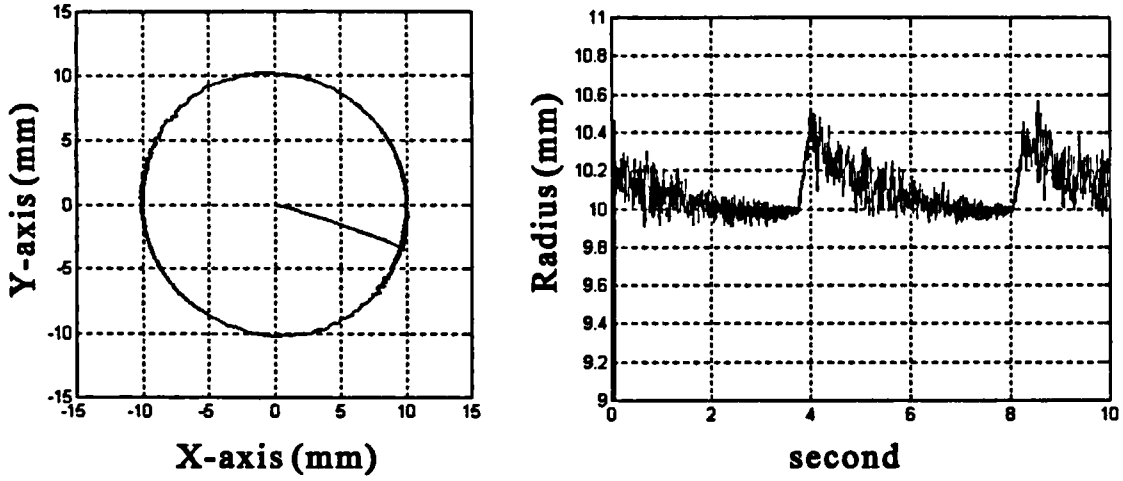


Fig. 6.10 As for Fig. 6.6, but $T_{s2} = 0.002$.

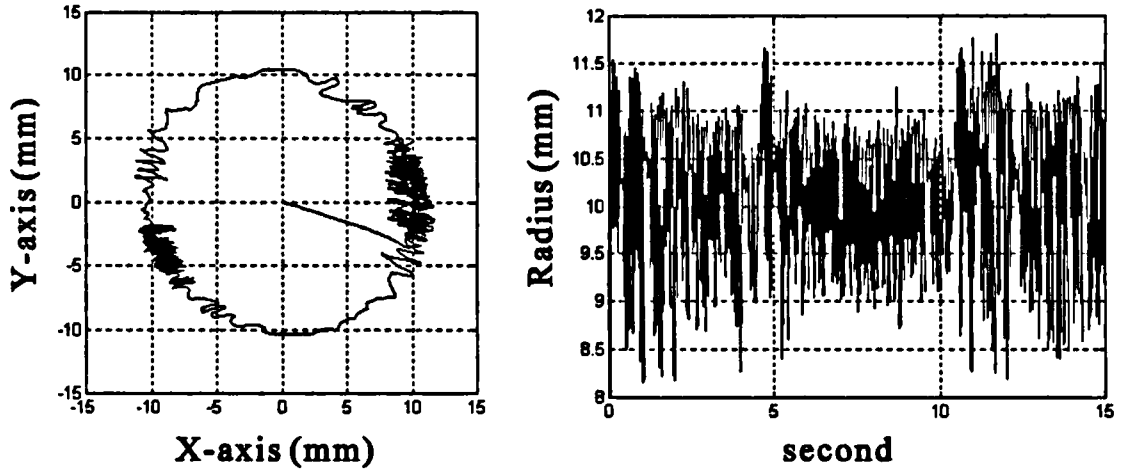


Fig. 6.11 As for Fig. 6.6, but $T_{s2} = 0.005$

6.2.1.2 Cartesian stiffness design

The independent joint control algorithm is the easiest method to implement and the computation time for the inner control law and the interpreter program is also the shortest compared to other control algorithms. However, it is difficult to design the desired stiffness in Cartesian coordinates (see section 5.3.1).

The Cartesian stiffness design, on the other hand, can easily specify the desired stiffness and damping in Cartesian space. The disadvantage is that the position error and velocity in each joint will affect the torque command in all other joints.

Fig. 6.12 show the simulation result for the case where the Cartesian proportional gain matrix K_{pc} is equal to $\begin{bmatrix} 2\times10^6 & 0 \\ 0 & 2\times10^6 \end{bmatrix}$ and $\begin{bmatrix} 2\times10^6 & 0 \\ 0 & 1 \end{bmatrix}$ in zones II and III, respectively (see the analysis in section 5.3.1.2). The derivative gain K_d is given as for Fig. 6.4. It can be seen that position error in Fig. 6.12 is less than that in Fig. 6.4, while velocities in both cases are similar.

The dynamic equations are as for Fig. 6.12, but the derivative gain is designed in Cartesian space and is given as $\begin{bmatrix} 8000 & 0 \\ 0 & 8000 \end{bmatrix}$ and $\begin{bmatrix} 8000 & 0 \\ 0 & 5000 \end{bmatrix}$ in zones II and III, respectively. The result is shown in Fig. 6.13. Its performance is better in terms of smoothness of movement, if compared to that of Fig. 6.12. In addition, the manipulator is more easily moved along the boundary.

Fig. 6.14 is the simulation as for Fig. 6.12, but forward decoupling is added into the control law. The movement of the manipulator is faster compared to that of Fig. 6.12. This is because the velocity related force such as the resistance of the cutting force is compensated by the decoupling algorithm.

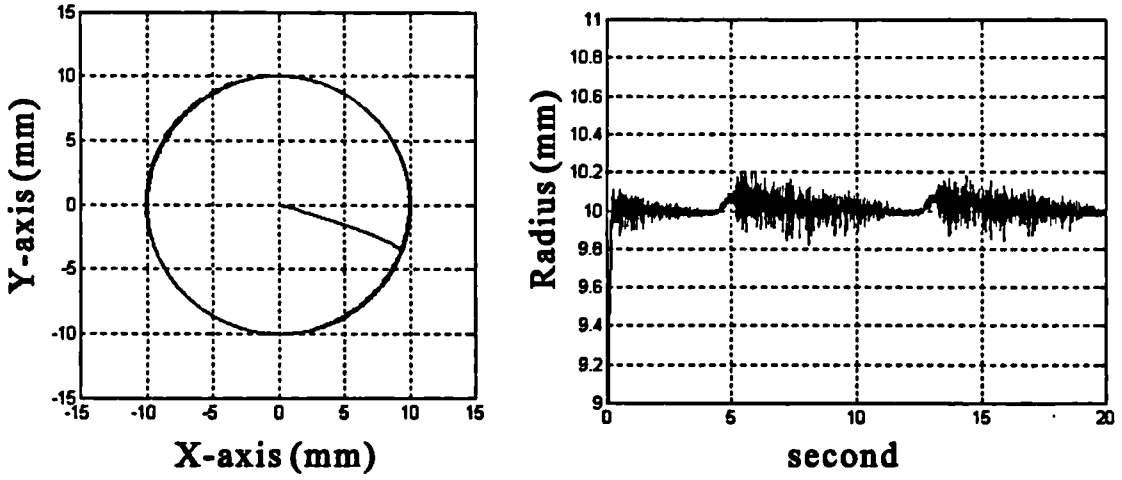


Fig. 6.12 Simulation of Cartesian stiffness design

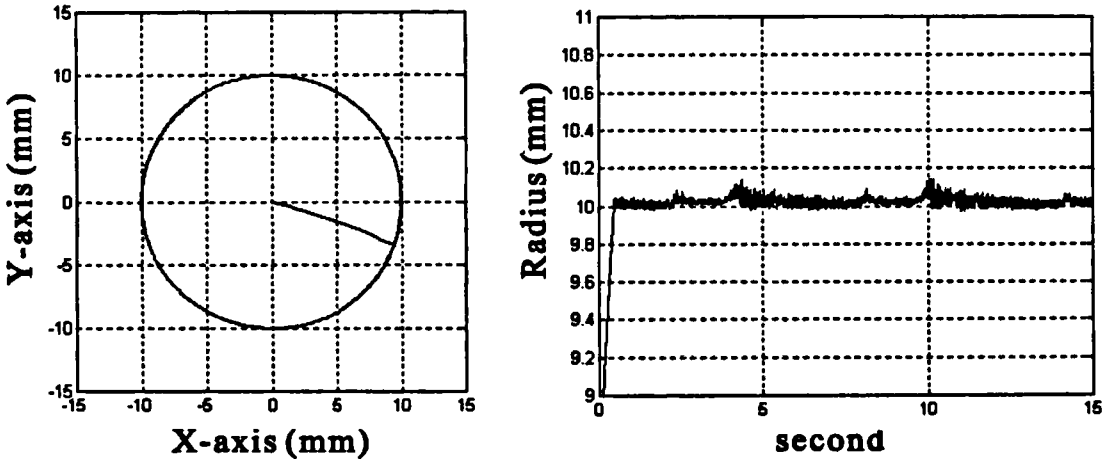


Fig. 6.13 As for Fig. 6.12, but derivative gains are also designed in Cartesian space

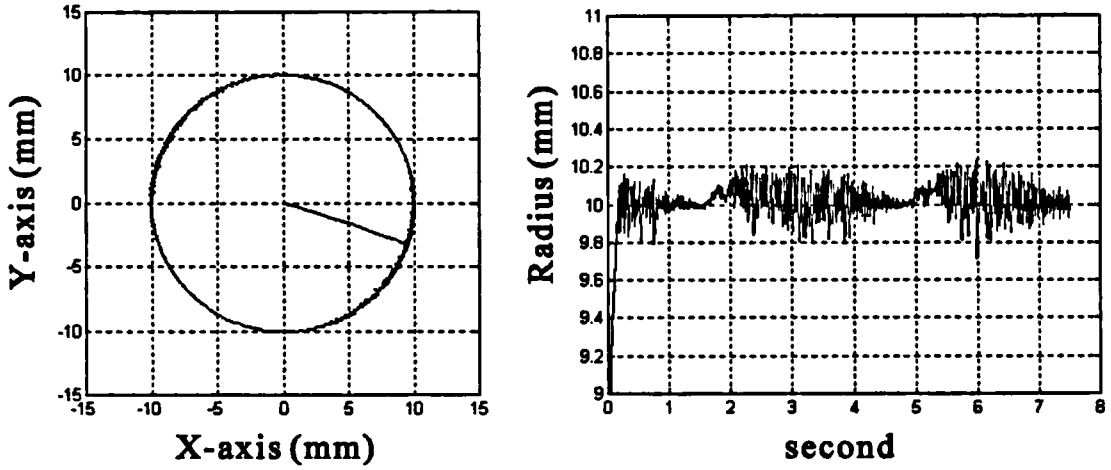


Fig. 6.14 As for Fig. 6.12, but with feedforward decoupling

6.2.2 Modified damping control

In modified damping control, a force sensor is used to measure the guiding force. The desired velocity of the manipulator is then determined according to the magnitude and direction of the force and the relative position of the current position of the robot to the motion constraint. The advantages of using modified damping control can be seen in the analysis of section 5.4. There are two ways of implementing a modified damping control algorithm: independent joint control and nonlinear decoupling technique. Their performance will be examined by the following simulations.

Fig. 6.15 shows a simulation using the independent joint control algorithm. The sampling rates, initial position of the manipulator and model of cutting force are given as for Fig. 6.4. The control gains K_p and K_d are given as in Fig. 6.3, but the maximum value of K_p is equal to 20000 and K_d equals 10 in zone I. The force feedback gain K_f (see section 5.4) equals 0.02 in zone I and is given as 0.01 in zones II and III. Compared to that using implicit force control, the manipulator is easier to operate and the position error is also smaller by performing the modified damping control algorithm.

Fig. 6.16 shows a simulation as for Fig 6.15, but integral feedback is added into the control law. It can be seen that position error is reduced by the integral gain.

Fig. 6.17 show a simulation using the nonlinear decoupling technique. In order to increase the effect of coupling forces, M_2 is given as 2 kg. Compared to that utilising independent joint control, the movement of the manipulator is more smooth and the position error is also smaller by using the decoupling method.

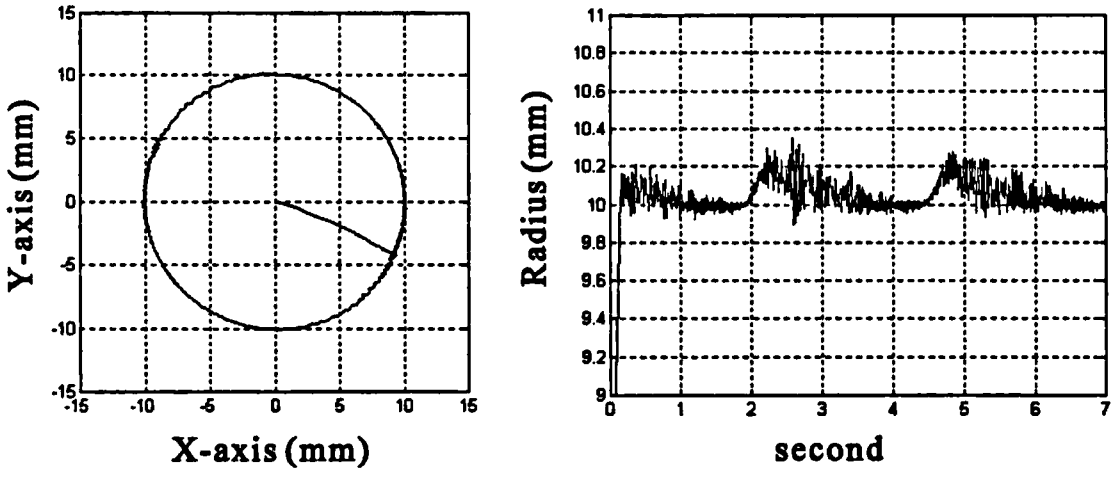


Fig. 6.15 Simulation of modified damping control (also see text).

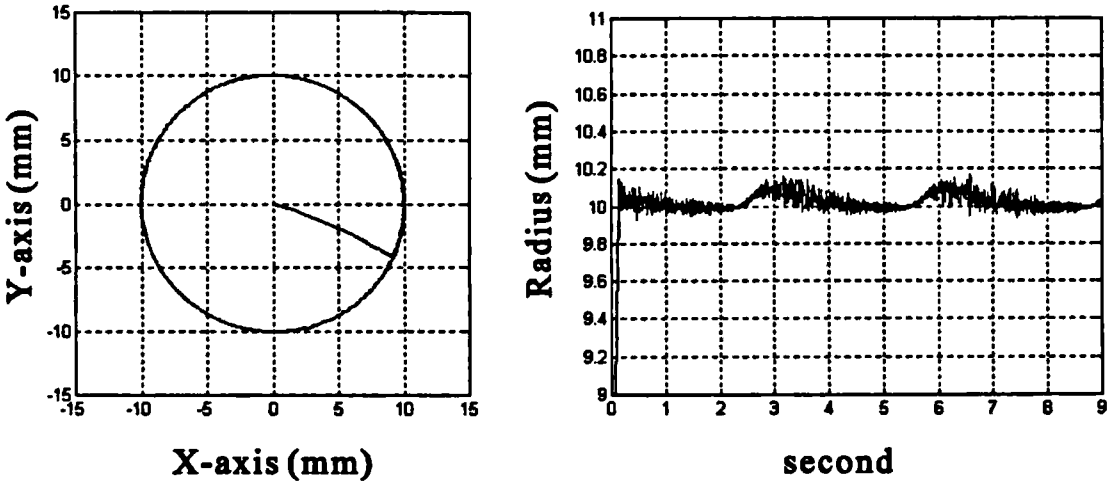


Fig. 6.16 As for Fig. 6.15, but integral feedback is added to the control law.

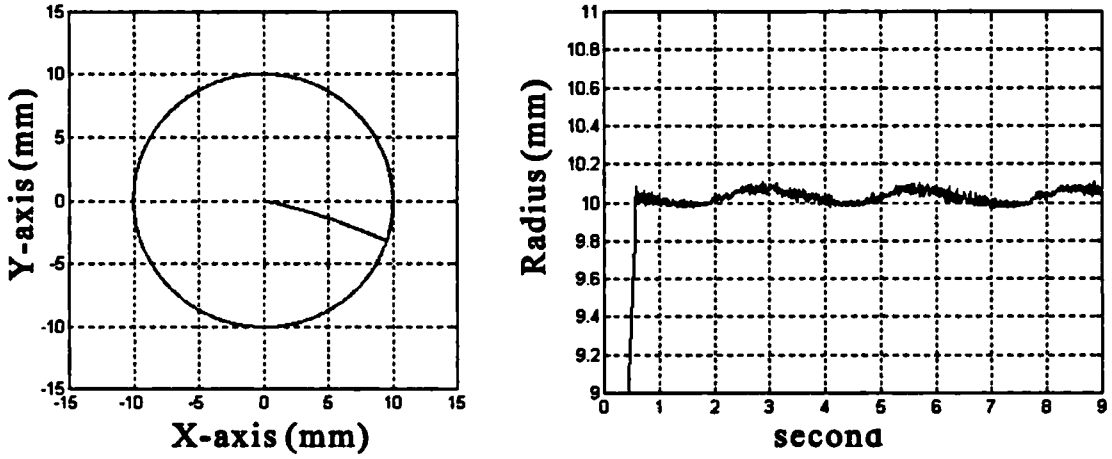


Fig 6.17 As for Fig. 6.12, but by decoupling method and $M_2 = 2$ kg.

6.3. Conclusion

Simulations of force control strategy with an active motion constraint have been carried out in the chapter. From simulation results, some points can be concluded:

(1). The sampling interval of the interpreter of the desired trajectory and control gains, T_{s1}, is not critical to the stability of the system. However, a slow T_{s1} will cause the movement to become 'jerky' in operation. The effect is further worsened by a time delay of T_{s1} in the interpreter program.

The 'potential' energy of the manipulator due to the proportional feedback is released after every T_{s1} interval. If T_{s1} is longer, the resistance force from the

controller will become higher during every sampling interval T_{s1}. As a result, it will become difficult to operate the manipulator dextrously.

(2). The sampling rate of the inner control law, T_{s2}, is more critical to the stability of the overall system. In the extreme case when T_{s2} is as slow as T_{s1}, the system becomes unstable.

(3). Integral feedback can reduce position error, however, which will increase the resistance force. If the updating rate of the control gains is too slow, adding the integral gain to the control law will not improve the performance because of the time delay.

(4). Change of inertia will influence the system response and the inertia force. If inertia forces would cause large position error, the problem can be solved by using decoupling feedback.

(5). When implementing implicit force control, the independent joint control algorithm is the easiest to implement. However, it is difficult to design the desired stiffness in Cartesian coordinates. The Cartesian stiffness design method, on the other hand, can easily specify the desired stiffness and damping in Cartesian space. From simulation results, it has been shown that by using Cartesian stiffness design, the manipulator is easier to be moved along the boundary and the position error is smaller compared to that of independent joint control. However, it should be noted that position error and velocity in each joint will affect the torque command in all other joints in the Cartesian stiffness design algorithm. This will decrease the disturbance rejecting ability of the manipulator in reality.

(6). There are two ways of implementing a modified damping control algorithm: independent joint control and nonlinear decoupling technique. The latter method should be chosen if the effect of inertia forces is significant. From a performance point of view, modified damping control is better than implicit force control. The disadvantage is a force sensor is needed to measure the guiding force. This will increase the cost and difficulty of mechanism design of the manipulator.

The concepts of artificial motion constraint and the controller design strategy have been shown to be feasible from the computer simulations. In the next chapter, a three-DOF manipulator and force sensors will be described. Subsequently, the performance of these force control strategies will be further verified by tests using the experimental system.

CHAPTER 7

EXPERIMENTAL RESULTS

This chapter starts with the experimental set up. The mechanism design of a three-DOF manipulator will be described, and the characteristics of each component will also be discussed. Subsequently, the structure of the computer software which implements the force control algorithm is presented. Finally, the experimental results by implementing implicit force control and modified damping control on the robot will be examined.

7.1 Experimental hardware

The block diagram of the overall system is shown in Fig. 7.1. The system can be divided into five parts:

- an IBM compatible personal computer 486 (PC/486),
- a digital signal processor (DSP) based motion controller,
- motor amplifiers,
- a three-DOF manipulator with an end-mill cutter fixed to its tip, and
- a two-axis XY plane force sensor plus a z-axis force sensor.

The PC/486 reads positions of the robot and guiding forces from the DSP, then computes and determines the values of reference positions, reference velocities and feedback gains based on the control strategy discussed in chapter 5. The sampling interval T_{s1} varies from 3.3ms to 5ms depending on the control algorithm used. At the end of every cycle T_{s1}, those values of desired positions, desired velocities and feedback gains are transferred to the motion controller via the dual ported memory (DPM). The motion controller implements the control law and independently controls the motors through the motor amplifier at a sampling rate T_{s2} whose value is about 0.63ms. Details of the system will be illustrated in the following sections.

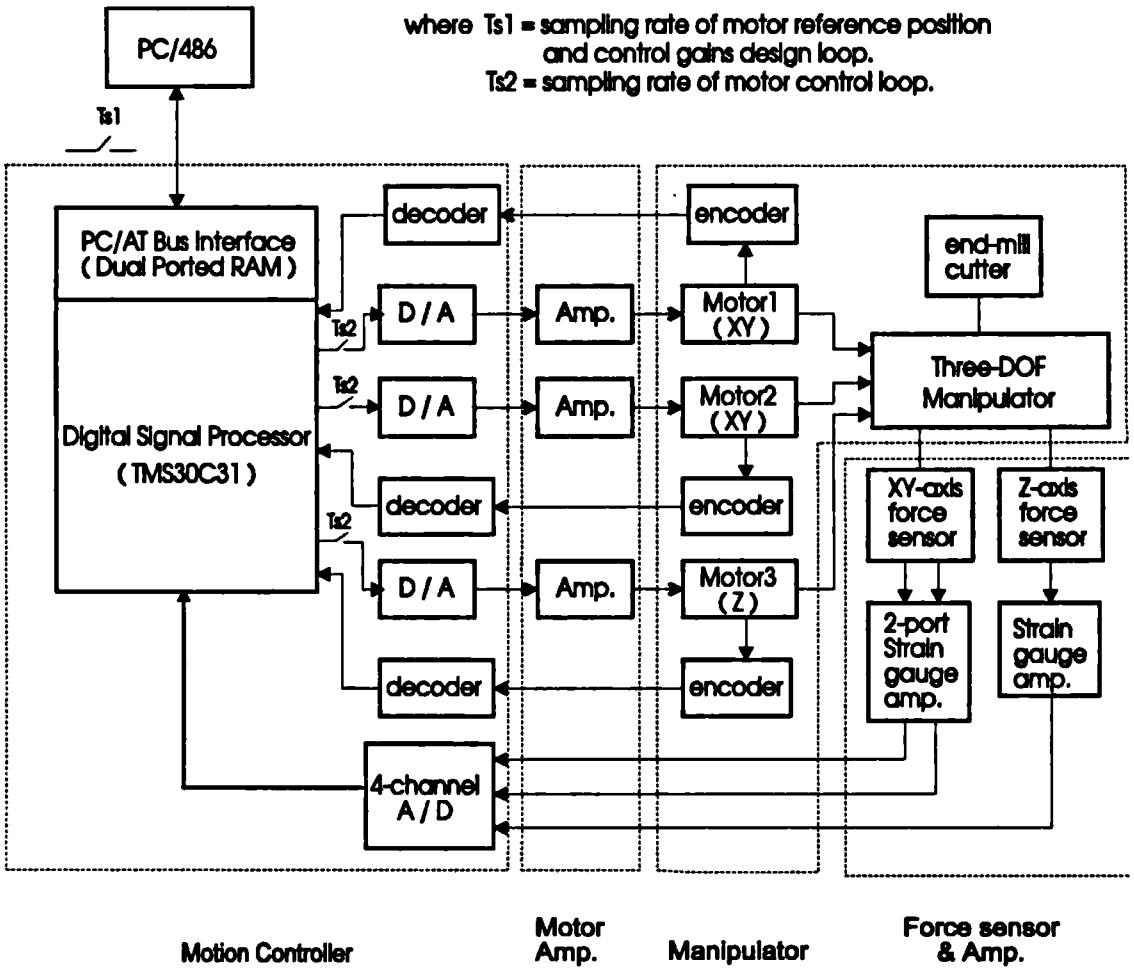


Fig. 7.1. Block diagram of the hardware

7.1.1 Mechanism of the three-DOF robot

The configuration of the manipulator is shown in Fig. 7.2. An end-mill cutter is fixed to the end of the robot. The first and second jointed links define coordinates of the cutter in the horizontal (XY) plane (see Fig. 7.3), whereas the ball screw drive system, which moves the jointed links up and down, defines the position of the cutter in Z-axis (see Fig. 7.4). A two-axis force sensor is mounted near the cutter, which can measure the force in the XY plane. An Z-axis force sensor, which senses the force in the vertical direction, is also constructed. Combining these two force sensors, a three dimensional force can then be measured. The design of the force sensors will be discussed in detail in the next section. The working space, transmission mechanism and actuators of the robot are described below.

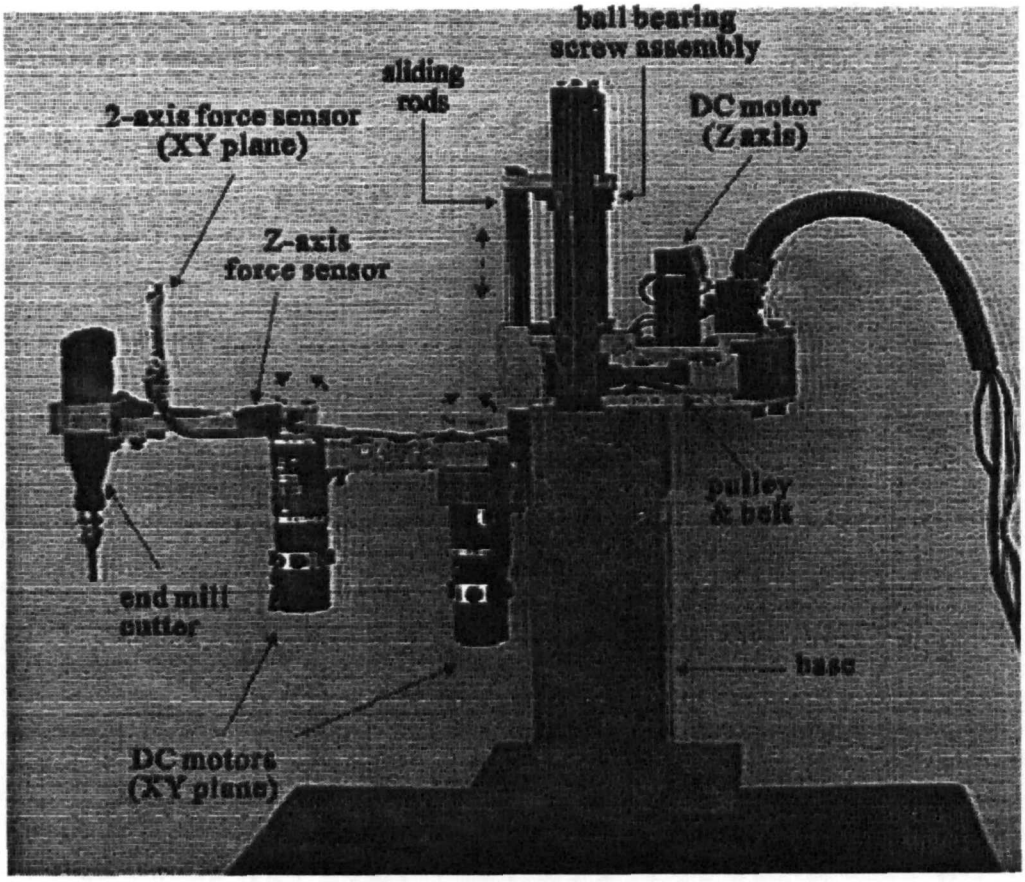


Fig. 7.2 Picture of the manipulator

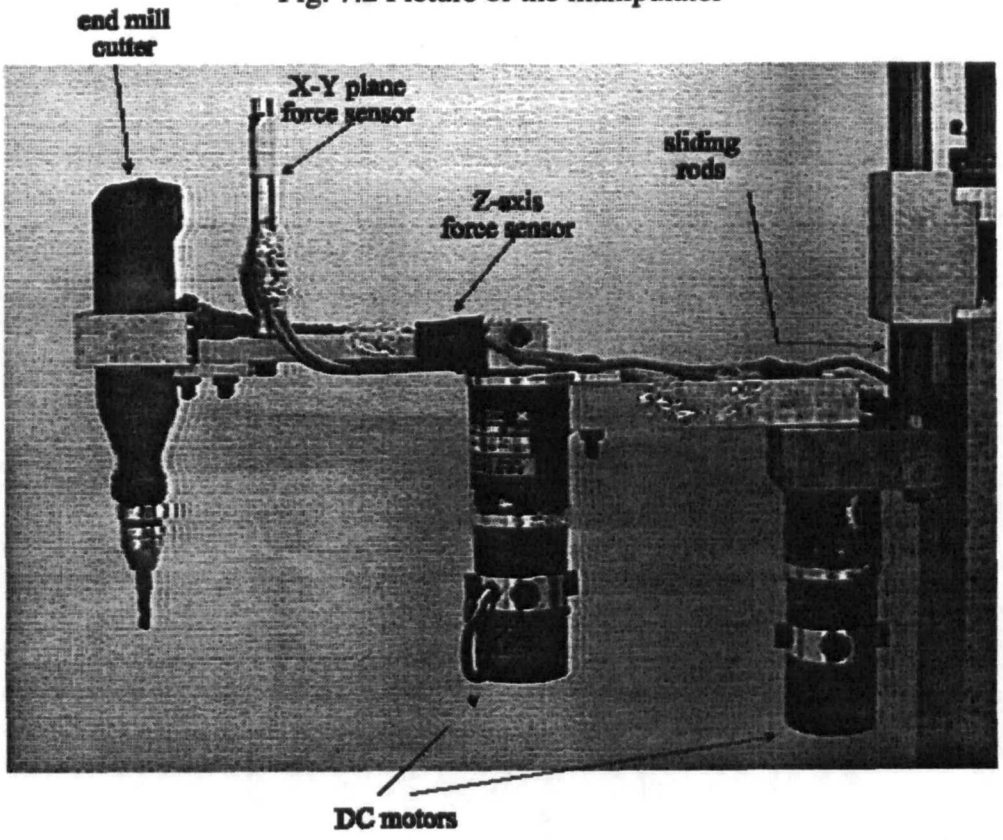


Fig. 7.3 Detail of the jointed links

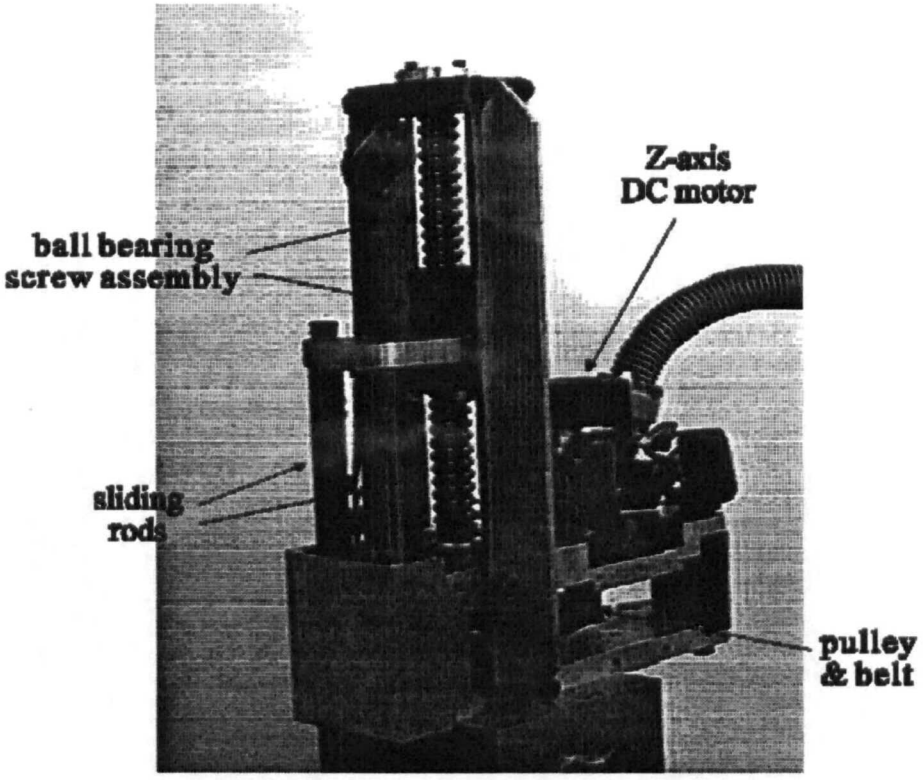


Fig. 7.4 Detail of the drive system for Z-axis

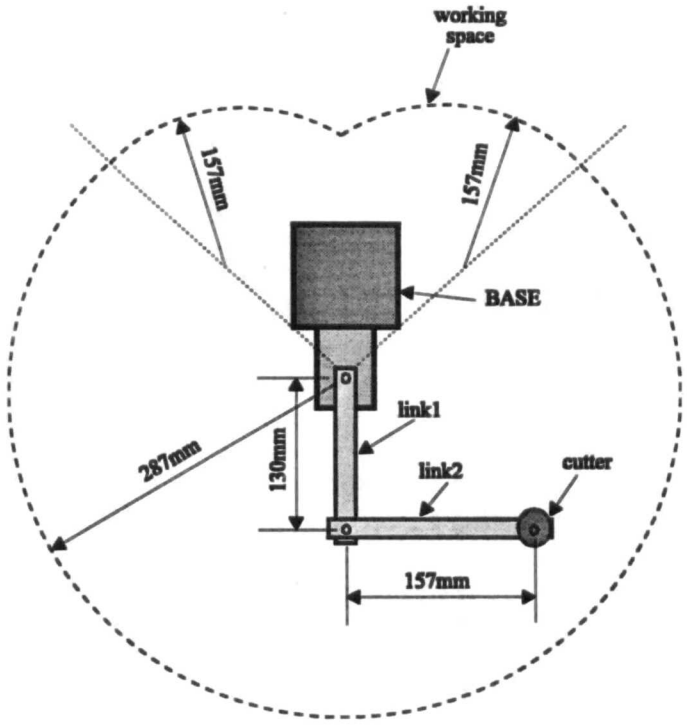


Fig. 7.5 Working space of the jointed arms of the manipulator in the horizontal plane.

Working space

The first link and second link of the manipulator are 130mm and 157mm long, respectively. The working space of the jointed arms of the manipulator in the XY plane is shown in Fig. 7.5. The maximum horizontal reach of the robot is 287mm, whereas the range of movement in the Z-axis is 120mm.

Transmission system and actuators

There are two main kinds of motor being used for industrial robots: stepping motors and direct current (DC) motors. The stepping motor is synchronous, so there is a correlation between the input command and the resulting position. However, the stepping motors have some disadvantages: (1). the acceleration and deceleration of the motor are discontinuous, (2). the performance standards are limited and (3). the torque varies with the position of the rotor and is very difficult to be controlled by a feedback loop [Lhote 84]. Because force control is critical to our system, DC motors are, therefore, used for this project rather than stepping motors.

Both DC servo motors used for the jointed arms in XY plane are from Harmonic Drive Company (type: RH-11-6001). Each motor includes a harmonic drive gearbox and an optical rotary encoder. The gear ratio of the harmonic drive is 50:1, and the resolution of the rotary encoder is 1000 pulse per revolution. The resolution of the encoder is multiplied by a factor of four if a quadrature encoder is used [Tech. Ref. 4]. The resolution at the output shaft of the motor is calculated by the following equation

Resolution at the motor output shaft

= gear ratio × resolution of encoder × encoder multiplier.

Therefore, the resolution of the motors used is equal to 200,000. Identified mathematic models for these two motors can be seen in chapter 4.

The Z-axis mechanism of the robot (see Fig. 7.4) consists of a DC motor, a belt drive, and a ball screw assembly. The DC motor for the Z-axis is from Maxon Motors (type: 2332-968, 15 Watt), which includes an encoder with resolution of 500 counts per turn.

The belt drive is composed of a toothed belt and two toothed pulleys, by which the output speed of the motor is reduced to one fourth. The ball screw assembly consists of a bearing with an internal thread located on a screwed rod. The bearing is connected to the jointed links via two sliding shafts. The upper end of the screwed rod is mounted to a ball bearing which is fixed to the base of the robot, whereas the lower end is inserted into the bigger toothed pulley of the belt drive. When the motor rotates, the belt drive will rotate the screw rod of the ball screw assembly. Consequently, the jointed-link mechanism will be moved up/down. The pitch of the screw rod is 5mm. Thus, if the motor rotates one turn, the end-mill cutter will move up or down by 1.25mm.

7.1.2 Motion controller & motor amplifiers

As described in chapter 5, computation of the overall force control system is divided into two parallel loops. In order to increase the computation speed, in addition to the PC/486, an independent processor is desirable. Traditionally, two processors can be linked via a serial communication such as RS 232 or RS 422. The data communication can be collected by an interrupt method, thus the processors can work independently and do not have to wait for the data transmitting and receiving. But the drawback of the serial communication is its slow speed. Therefore, a motion controller ,which has near zero communication time with the PC, has been chosen for our application. The motion controller is from Optimised Control Ltd (type: Nextmove), consisting of:

- A floating point 32 bit Digital Signal Processor (DSP) TMS320C31 from Texas Instruments Inc.,
- 16 bit ISA bus interface via 2K byte dual ported memory (DPM) offering near zero wait state access to information over the ISA bus,
- Four channels of incremental encoder input,
- Four channels of 12 bits analogue output (digital-to-analogue converter (D/A)), and

• Four 12 bit differential analogue inputs (analogue-to-digital converter (A/D)) etc.

The motion controller is plugged into the PC interface slot and communicates with the PC via DPM (see Fig. 7.1). In addition to implementing the force control law, functions of the motion controller also include: sending the control signals to the motor amplifiers, reading position data of the motors from the encoders, reading force signals from the force sensors and writing the position and force information into the DPM.

The two motor amplifier cards are also from Optimised Control Limited (type: EuroAmp/2). Each card possesses two DC voltage outputs, whose type is pulse-width modulation (PWM). The amplifier is configured as a current amplifier, which is the same as a transconductance amplifier or torque amplifier. The gain of the amplifier can be adjusted such that a particular input demand (within the range ±10V) corresponds to a particular level of current, up to a maximum of ±3.5A.

7.1.3 Force sensor

For modified damping control, it is necessary to measure the surgeon's guiding force. Depending on the location of the sensor on the robot, there are three types of force sensors generally used for industrial applications:

(1). Wrist force sensor.

(2). Joint torque sensor.

(3). Tactile or hand sensor [Nicholls 89].

The tactile sensor is normally used to measure the contact force between the robot end-effector and the grasped object. The joint torque sensor is located at each joint, whose disadvantage is that the gravity and Coriolis force are correlated to the sensed force signal. The wrist-mounted force sensor is more difficult in mechanical design. However, they are more sensitive and easier to use than joint sensors [Van Brussel 85].

In our experimental system, the three-dimensional force is measured by a two-axis force sensor together with an independent one-dimensional force sensor (see Fig. 7.3). The two-axis force sensor, which is similar to a wrist-mounted type, is used to measure the force in XY plane, whereas the one-dimensional force mounted near the root of the second link can measures the force in the Z direction.

7.1.3.1 Configuration of the force sensors

A. The two-axis force sensor

The two-axis force sensor is designed in a way such that it can be easily manufactured and can also be easily mounted onto the robot. The shape of the sensor is a circular shaft made of aluminium alloy. The interior of the shaft is hollowed out in order to increase the sensitivity. There are eight strain gauges firmly glued near to the root of the aluminium shaft. Each four strain gauges are connected into bridge circuits, so the force signal can be amplified and the temperature effect can be compensated (see Fig. 7.6). When the force sensor is blended with the external force, the resistance of the strain gauge will be changed. Through the use of Wheatstone-bridge circuits, the variation in resistance is converted into an electrical voltage signal which can then be read by the computer via an analogue-to-digital converter. The two strain gauge bridges are attached to the sensor body in an orthogonal arrangement. Thus, sensed forces can be easily decoupled into X-Y signals.

B. The z-axis force sensor

The Z-axis force sensor utilises four strain gauges attached near to the root of the second link of the manipulator (see Fig. 7.3 and 7.7). The strain gauges are connected into the bridge circuit in a similar manner to that of the two-axis force sensor.

The principle of the bridge circuit and the force sensors are described in appendix A, and the calibration results of the sensors are presented in appendix B.

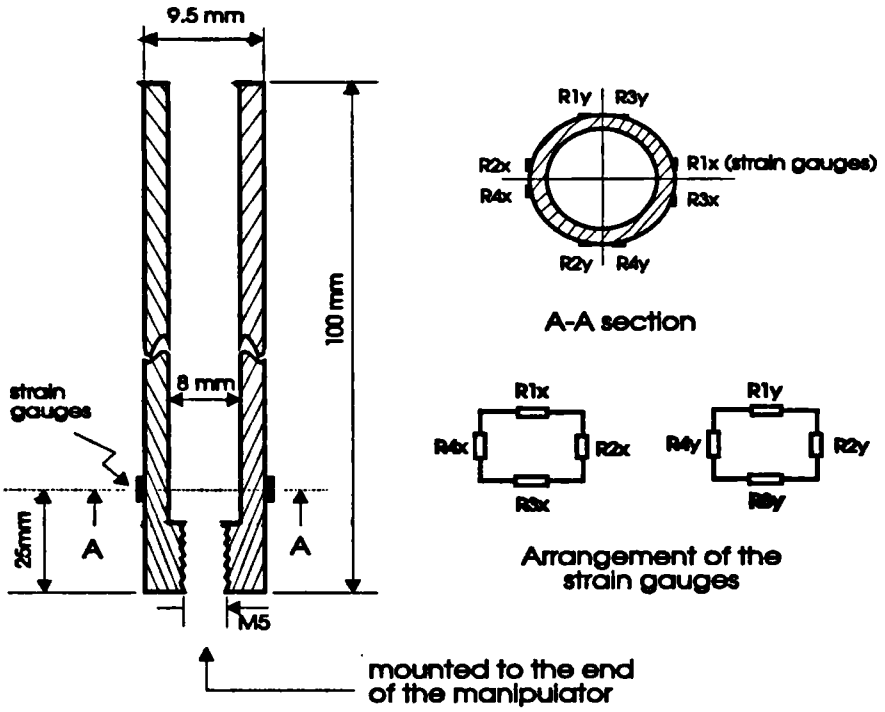


Fig. 7.6 Structure of the 2-axis force sensor and the arrangement and connection of the strain gauges

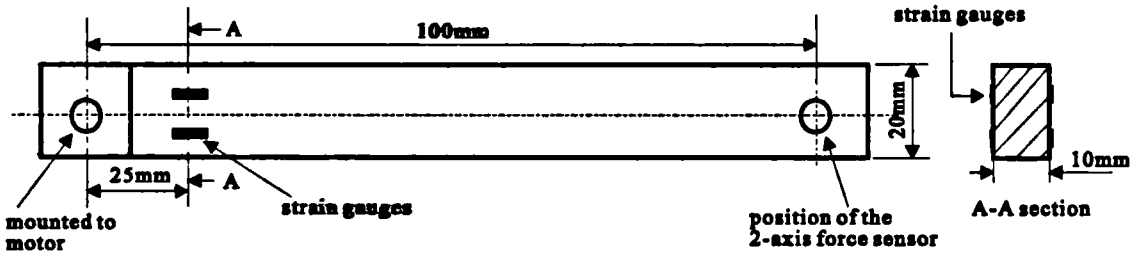


Fig 7.7 Configuration of the Z-axis force sensor.

7.1.3.2 Force signal processing (Coordinate transformation)

When using the force sensors, force signals are normally required to be transformed between different coordinates. For instance, it is desirable to transform the force signal from the coordinates of the force sensor into the coordinates of robot base, if the desired trajectory is defined in the base coordinate. Coordinate transformation of the force signal is dependent on the configuration of the manipulator in addition to the positions of the force sensors on the manipulator.

Fig. 7. 8(a) shows the schematic diagram of the mechanism of the manipulator as described in section 7.1.1. $F=\begin{bmatrix}F_x & F_y & F_z\end{bmatrix}$ is the external force; XYZ is the coordinate of the base of the robot; L_1 is the distance between the two-axis force sensor and the z-axis force sensor and L_2 is the distance from the position of the external force to the end of the force sensor. Fig. 7.8(b) shows the diagram of the external force and the jointed links in XY plane, in which X_1Y_1Z is the coordinate of the two-axis force sensor.

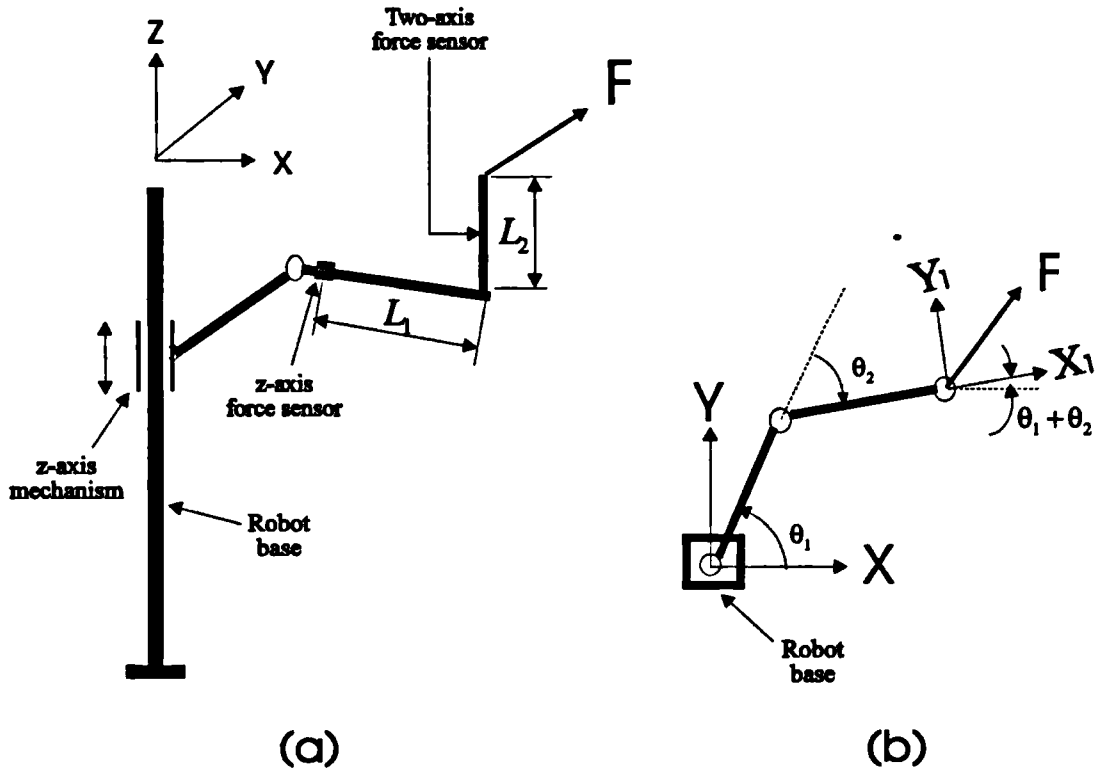


Fig. 7. 8 Schematic diagram of the manipulator, force sensors and external force

If $F^1=\begin{bmatrix}F_x^1 & F_y^1 & F_z\end{bmatrix}$ is the external force in X_1Y_1Z coordinate, the following geometrical relationship can be attained (see Fig. 7.8)

$$\begin{bmatrix}F_x\\F_y\end{bmatrix}=\begin{bmatrix}\cos(\theta_1+\theta_2) & -\sin(\theta_1+\theta_2)\\ \sin(\theta_1+\theta_2) & \cos(\theta_1+\theta_2)\end{bmatrix}\begin{bmatrix}F_x^1\\F_y^1\end{bmatrix}. \qquad (7.1)$$

In equation (7.1), F_x^1 and F_y^1 can be obtained from the A/D converter. Substituting the mathematical model of the two-axis force sensor (see appendix B) into (7.1) yields

$$\begin{bmatrix} F_x \\ F_y \end{bmatrix} = \begin{bmatrix} \cos(\theta_1+\theta_2) & -\sin(\theta_1+\theta_2) \\ \sin(\theta_1+\theta_2) & \cos(\theta_1+\theta_2) \end{bmatrix} \left(\begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix} \begin{bmatrix} V_x \\ V_y \end{bmatrix} + \begin{bmatrix} v_x \\ v_y \end{bmatrix} \right). \quad (7.2)$$

Therefore, when force signals are read in from A/D, the equivalent forces F_x and F_y in base coordinate can be derived from equation (7.2).

As for the force signal in the Z axis, it can be seen from Fig. 7. 8 that in addition to F_z, F_x^1 also influences the value read from the z-axis force sensor because of the configuration of the mechanism. By modifying equation (B.5) in appendix B, the relationship between the output of A/D, V_z, and the external force can be given as

$$F_z + \frac{L_2}{L_1} F_x^1 = K_z V_z + v_z. \quad (7.3)$$

From (B.1), F_x^1 is equal to

$$F_x^1 = k_{11} V_x + k_{12} V_y + v_x. \quad (7.4)$$

By combining (7.4) with (7.3), F_z is given as

$$F_z = K_z V_z + v_z - \frac{L_2}{L_1}(k_{11} V_x + k_{12} V_y + v_x). \quad (7.5)$$

Equation (7.5) describes the relationship between the force in z-axis and the output of the A/D converters. It can be seen that F_z is coupled with the signal of the two-axis force sensor because of both the configurations of both the manipulator and the force sensors.

7.2 Software structure

The flow chart of the software is shown as in Fig. 7.9. The control software of the overall system consists of two programs. The first one is written in Borland C and running in a PC/486, which implements the interpreter program of the desired trajectory and controller gains. The second program is written in TMS320 floating-

point DSP optimizing C. The program is compiled and then linked as a Common Object File Format (COFF) code. The COFF code runs in the DSP, which executes the inner control law and outputs the control signal to drive the robot motors.

The program in the PC/486 first receives the point data of a motion constraint, which can either be from a data file or defined on the computer screen by a mouse on line. Subsequently, a look-up table of the motion constraint is constructed in a way as described in chapter 5.2, and stored in the computer memory. The PC program then down loads the COFF code into the DSP of the motion controller, and triggers the COFF code to start running.

The program in the PC reads the position and force data from the DSP. Subsequently, by referencing the look-up table, values of the reference position, velocity and feedback gains based on the force control strategies discussed in chapter 5.3 are determined. At the end of every computation cycle, those values are transferred to the motion controller via the DPM. When the PC program receives a command to quit from the user, it will set the output of the control law at zero.

The COFF code on the DSP reads the current position of the robot and the external force from the A/D converters, and subsequently updates these data in the DPM. The COFF code also reads the desired position, desired velocity and feedback gains from the PC program via DPM. It then implements the force control algorithm independently.

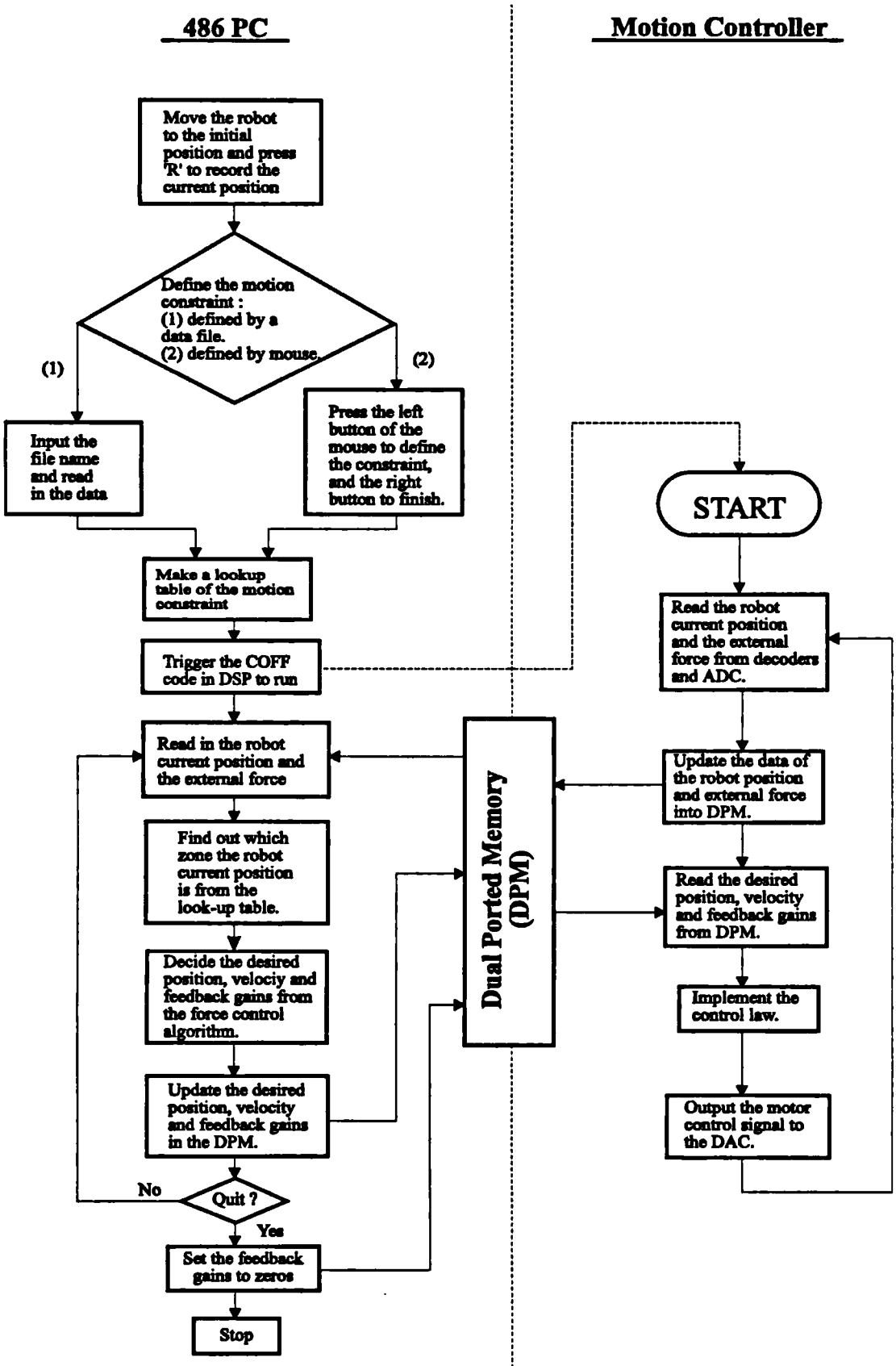


Fig. 7.9 Flow chart of the control software

7.3 Implicit force control

The analysis of implicit force control in section 5.3.1 has been examined by computer simulations in section 6.2.1. Further investigation of the control algorithm will be carried out by experiments here. The manipulator is moved by hand and its position is traced on the computer screen. The hand's guiding force is recorded by the force sensors (see Fig. 7.2), and the position of the cutter is calculated from the joint encoders. For the sake of simplicity, the motion constraints here will only be defined in the horizontal plane. Thus, the experimental results can be clearly presented and the performance of the force control algorithm can be easily examined. Applications of three-dimensional motion constraint will be implemented in section 7.5.

The coordinate is defined as in Fig. 7.10. The Z axis is positive in downward direction, XY is the world coordinate, and X_cY_c, which rotates with the jointed links, is the cutter coordinate. The motion constraint is defined in XY coordinates, and its centre is given at the initial position of the cutter.

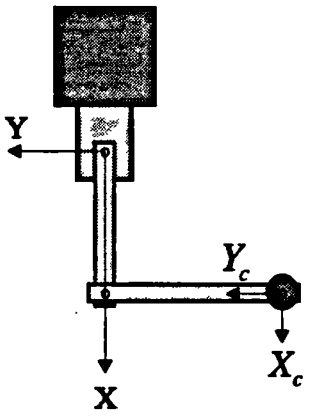


Fig. 7.10 Coordinate of the manipulator

7.3.1 Independent joint control

Fig. 7.11 shows the force and displacement characteristics in the X and Y directions against time with a 20mm horizontal motion constraint. The initial position of the manipulator is at $\theta_1 = 0$ and $\theta_1 = 90^o$. Control gains K_p and K_d for each joint are equal to 10000 and 80, respectively. The result shows that when the robot approaches

the end limit position B, the resistance force suddenly increases. Position error in Y axis is less than 0.08mm when the external force is about 20N and 4N along X and Y axes, respectively.

Fig. 7.12 shows the result of a circular constraint with a radius of 20mm. The motion constraint is defined as

zone I: radius < 10mm

zone II: 10mm ≤ radius < 20mm, and

zone III: radius ≥ 20mm.

Control gains K_p and K_d for each joint are given the same as for the previous computer simulation in Fig. 6.3. The radius error is about 0.4mm when the external force is 20N. It can be found that both the simulation (see Fig. 6.4) and experimental results approach a similar performance. The analysis of the force control strategy is further confirmed here.

Fig. 7.13 shows the experiment as for Fig 7.12, but with a rectangular constraint. The robot is moved from the centre position o to the boundary A, and then along the boundary (A→B→C→D→E→A). Control gains are given the same as for Fig. 7.12. The graph shows that the resistance force steadily increases as the robot approaches boundary A, and stays almost constant as the robot moves along the boundary at constant speed. The maximum position error in zone III is about 0.5mm when the external force is 20N.

Motion constraints defined for the above experiments are all regular shapes, which can be represented by explicit mathematical functions. For a general shape of motion constraint, it can be constructed and represented by a look-up table as described in section 5.1. Fig. 7.14 shows a contour of motion fitted by a B-spline method, which is then represented by a look-up table in the computer. Fig. 7.15 shows the experimental results of this motion constraint by using independent joint design of the implicit force control algorithm.

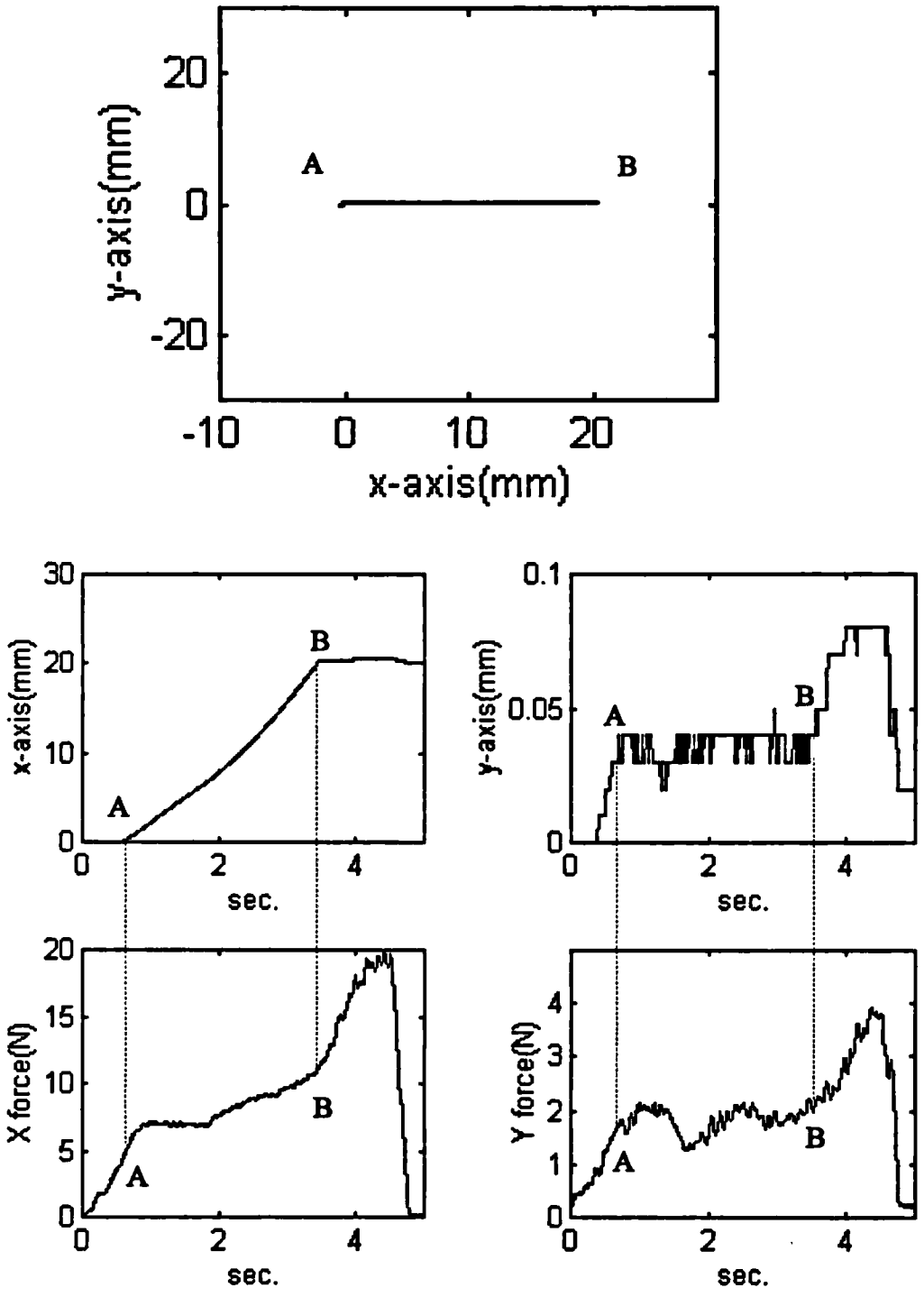


Fig. 7.11 Force and displacement characteristics in the X & Y directions against time, when a 20mm horizontal motion is operated by hand to the 3-DOF manipulator (also see text)

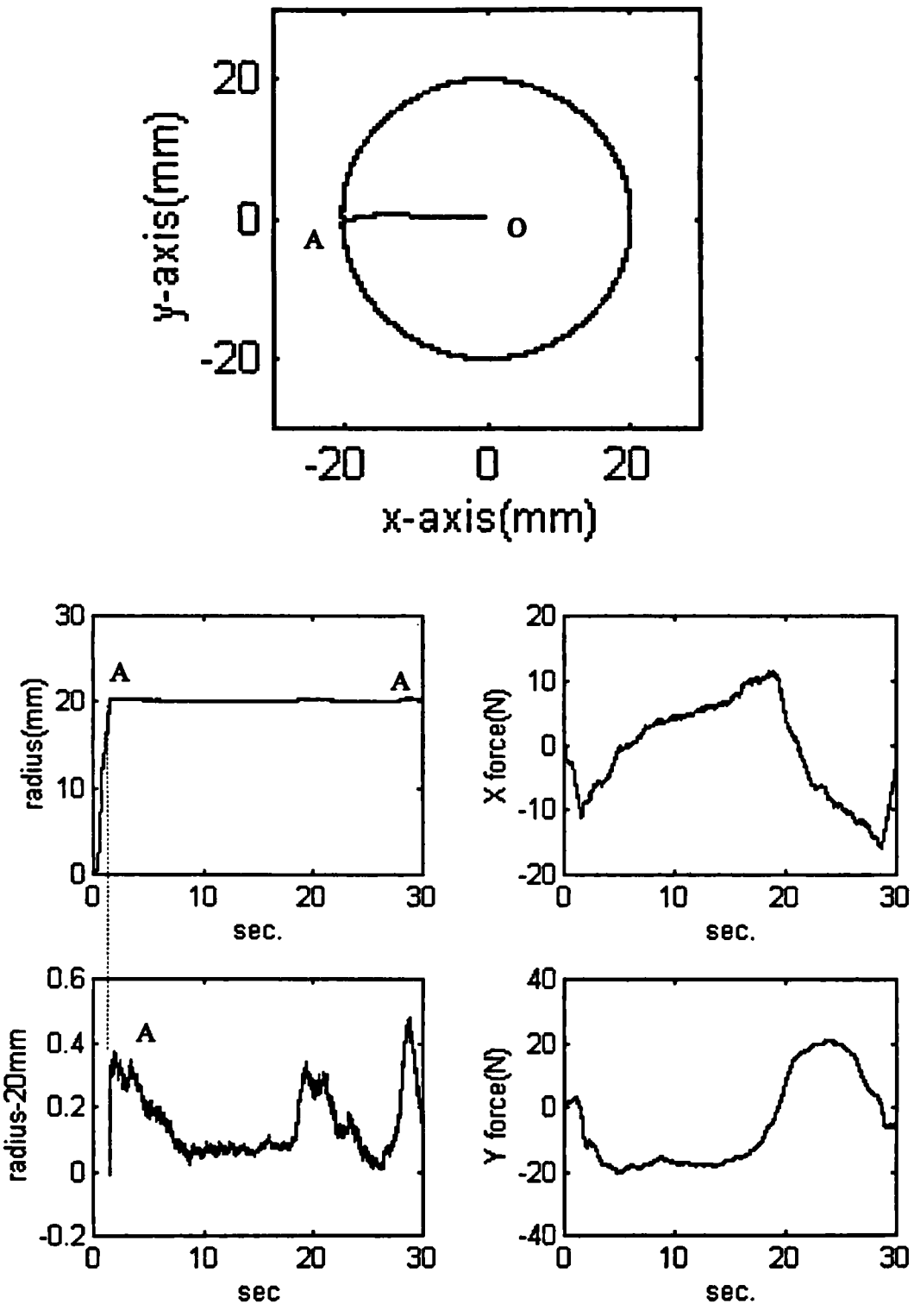


Fig.7.12 Force and displacement characteristics as for Fig. 7.11, but with a circular motion constraint with a radius of 20mm.

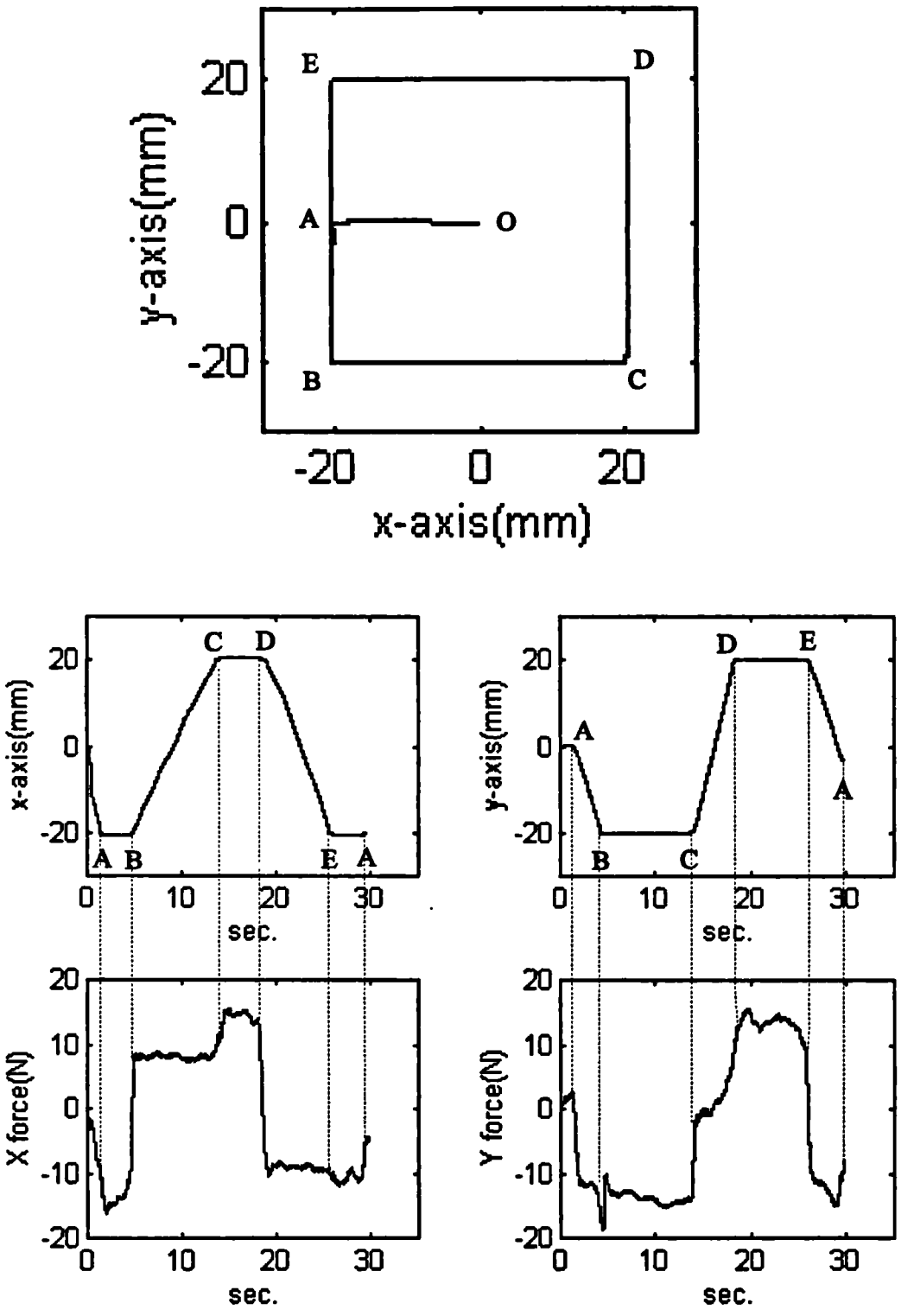


Fig. 7.13 Force and displacement characteristics as for Fig. 7.11, but with a rectangular motion constraint with 40mm length

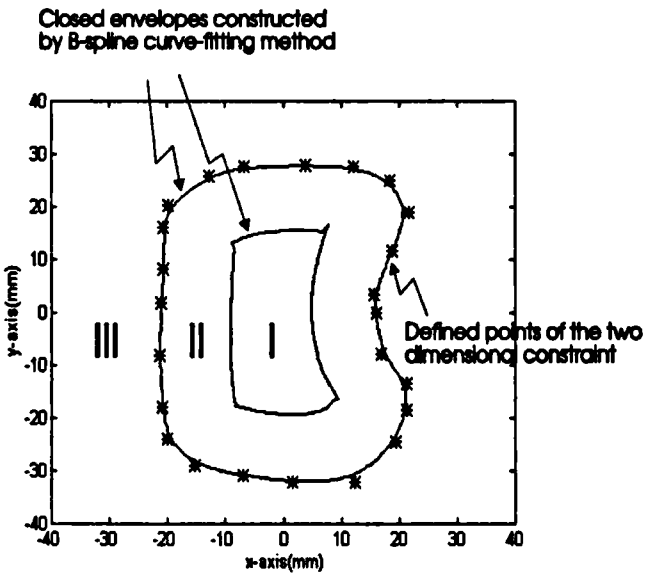


Fig. 7.14 The result of the motion constraint by using B-spline method to fit the defined points

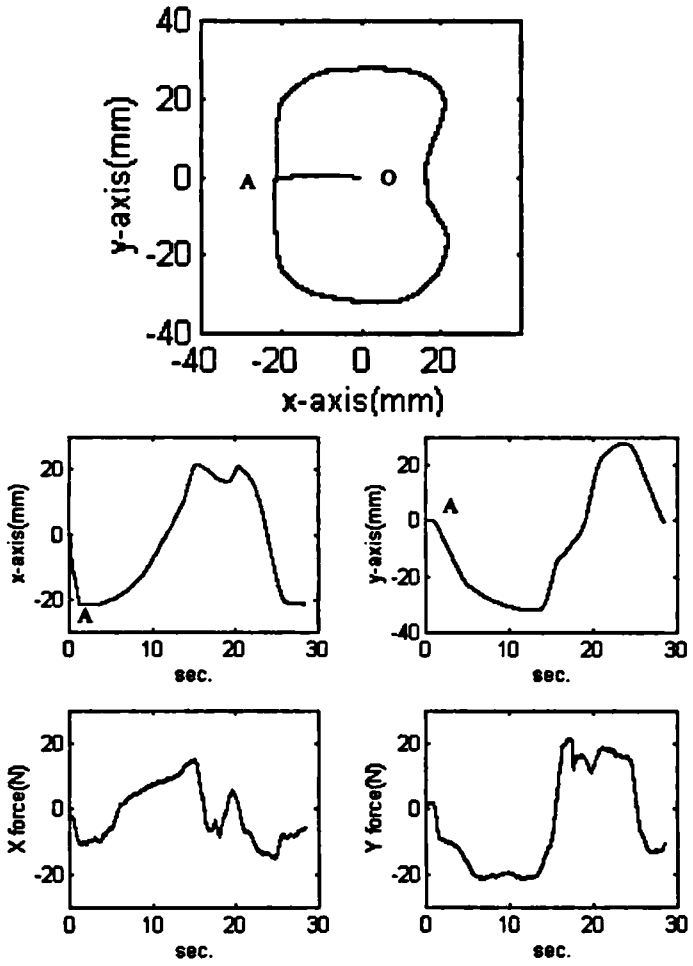


Fig. 7.15 Force and displacement characteristics as for Fig. 7.11, but with a constraint defined as in Fig. 7.14

7.3.2 Cartesian stiffness design

Fig. 7.16 shows the experimental result of a circular constraint with a radius of 20mm controlled by the Cartesian stiffness design method. The Cartesian proportional gain matrix K_{pc} is equal to $\begin{bmatrix} 1.2\times10^5 & 0 \\ 0 & 10^2 \end{bmatrix}$ in zone III. In zone II, K_{pc} is given as

$$\begin{bmatrix} 1.2\times10^5 & 0 \\ 0 & 1.2\times10^5 \end{bmatrix} \times \frac{d}{D},$$

where d is the distance from the current position of the robot to zone III and D is the width of zone II. The Cartesian derivative gain matrix K_{dc} is given in a similar way to that of K_{pc}, but with values of $\begin{bmatrix} 5000 & 0 \\ 0 & 5000 \end{bmatrix}$ and $\begin{bmatrix} 5000 & 0 \\ 0 & 10 \end{bmatrix}$ in zones II and III, respectively. It should be noted that the Y component of the Cartesian gains K_{pc} and K_{dc} in zone III is parallel to the tangent of the boundary of the motion constraint. When the Y component of the gains have smaller values compared to that of X component, the stiffness along the boundary is smaller than those in other directions. Therefore, the robot can be moved more easily along the boundary. This design concept has to be confirmed by experimental tests. However, it is found that the configuration of the manipulator affects the stiffness in addition to the design of the control gains. For instance, when the joint control gains at position B in Fig. 7.16 are calculated from Cartesian gains K_{pc} and K_{dc} (see section 5.3.1.2), they are equal to $\begin{bmatrix} 2140 & 82 \\ 82 & 5 \end{bmatrix}$ and $\begin{bmatrix} 9.0 & 0.7 \\ 0.7 & 0.5 \end{bmatrix}$ for proportional and derivative gain matrices, respectively. Both the proportional and derivative control gains for the second joint motor have very small values. As a result, when the robot is moved along the boundary, a resistance drop and thus a 'slippery' movement occurs around position B. This is the reason why there is a position error 'jump' at positions B and C. In order to avoid this problem, one solution is not to give very small values to both proportional and derivative gains for a joint motor at the same time. Fig. 7.17 shows the result as for Fig. 7.16, but K_{pc} is equal to $\begin{bmatrix} 1.2\times10^5 & 0 \\ 0 & 1.2\times10^5 \end{bmatrix}$ in zone III. The result shows

that a drop in stiffness still happens at positions C and E. Fig. 7.18 is the result for the case where K_{pc} is the same as that for Fig. 7.16 but K_{dc} is given as $\begin{bmatrix} 5000 & 0 \\ 0 & 5000 \end{bmatrix}$ in zone III. It has been found that the stiffness along the boundary remains almost the same. Therefore, it is recommended that values of components of the Cartesian damping matrix K_{dc} should not have too big a difference in order to avoid a drop in stiffness along the boundary. This has been confirmed by further experiments. One example is shown in Fig. 7.19, where K_{dc} is given as $\begin{bmatrix} 5000 & 0 \\ 0 & 2500 \end{bmatrix}$ in zone III and K_{pc} is the same as for Fig. 7.16.

Effects of coupling forces can be estimated from the dynamic equations 6.11 and 6.15. For instance, at position $\theta_1 = 0$ and $\theta_2 = 90^\circ$, velocity-related forces are equal to 0.033Nm and 0.011Nm for joints one and two, respectively, when both $\dot{\theta}_1$ and $\dot{\theta}_2$ are equal to 1 rad/sec. In this case, the velocity of the manipulator is equal to 0.44 m/sec. Compared to the external force, the effect of the coupling forces can be neglected for our system. This has been confirmed by experimental tests. However, for other cases when the effect of the coupling forces is significant, nonlinear decoupling terms (as analyzed in chapter 5) should be added to the control algorithm to compensate for the coupling forces.

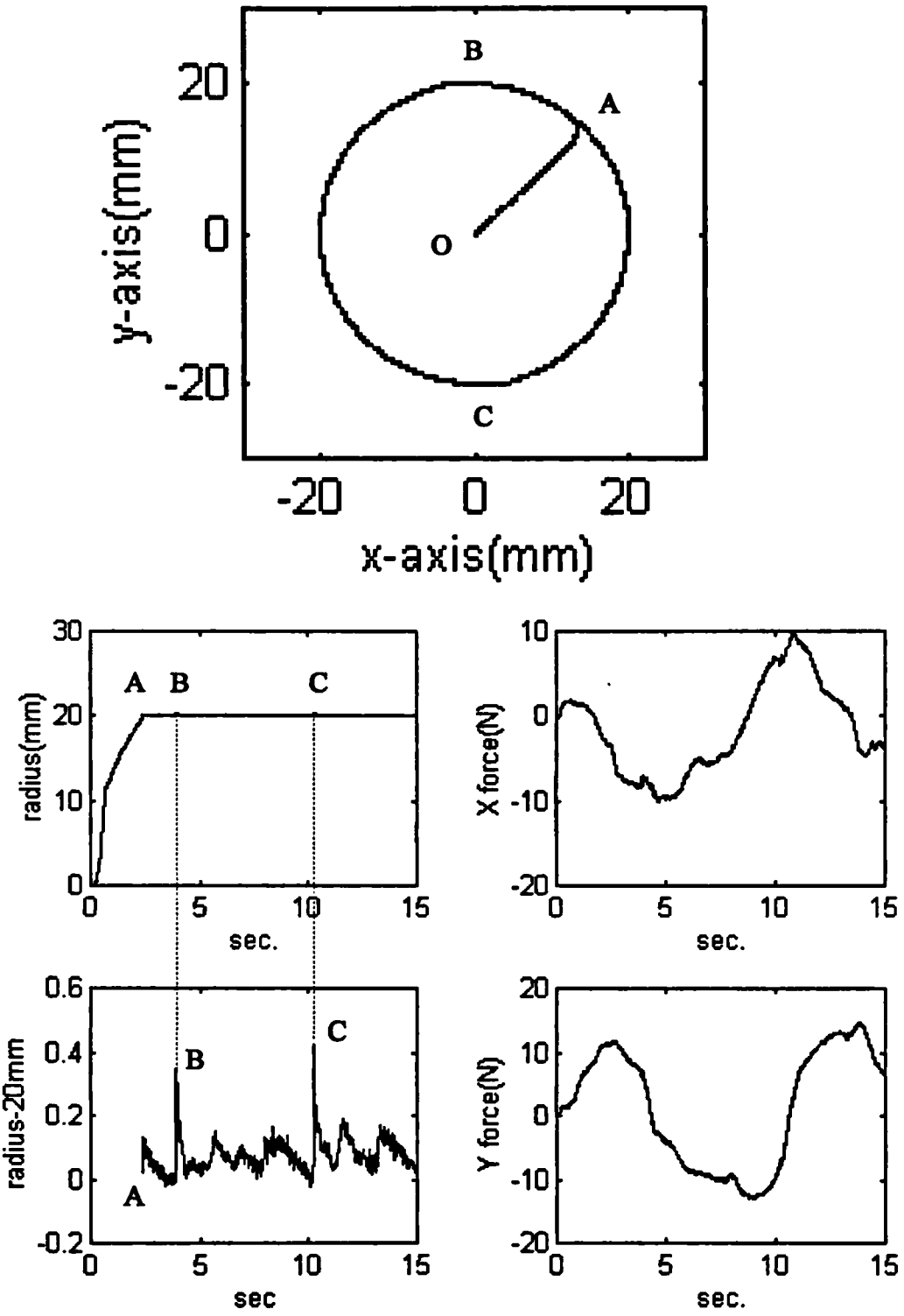


Fig. 7.16 Force and displacement characteristics as for Fig. 7.12, but with Cartesian stiffness design (also see text).

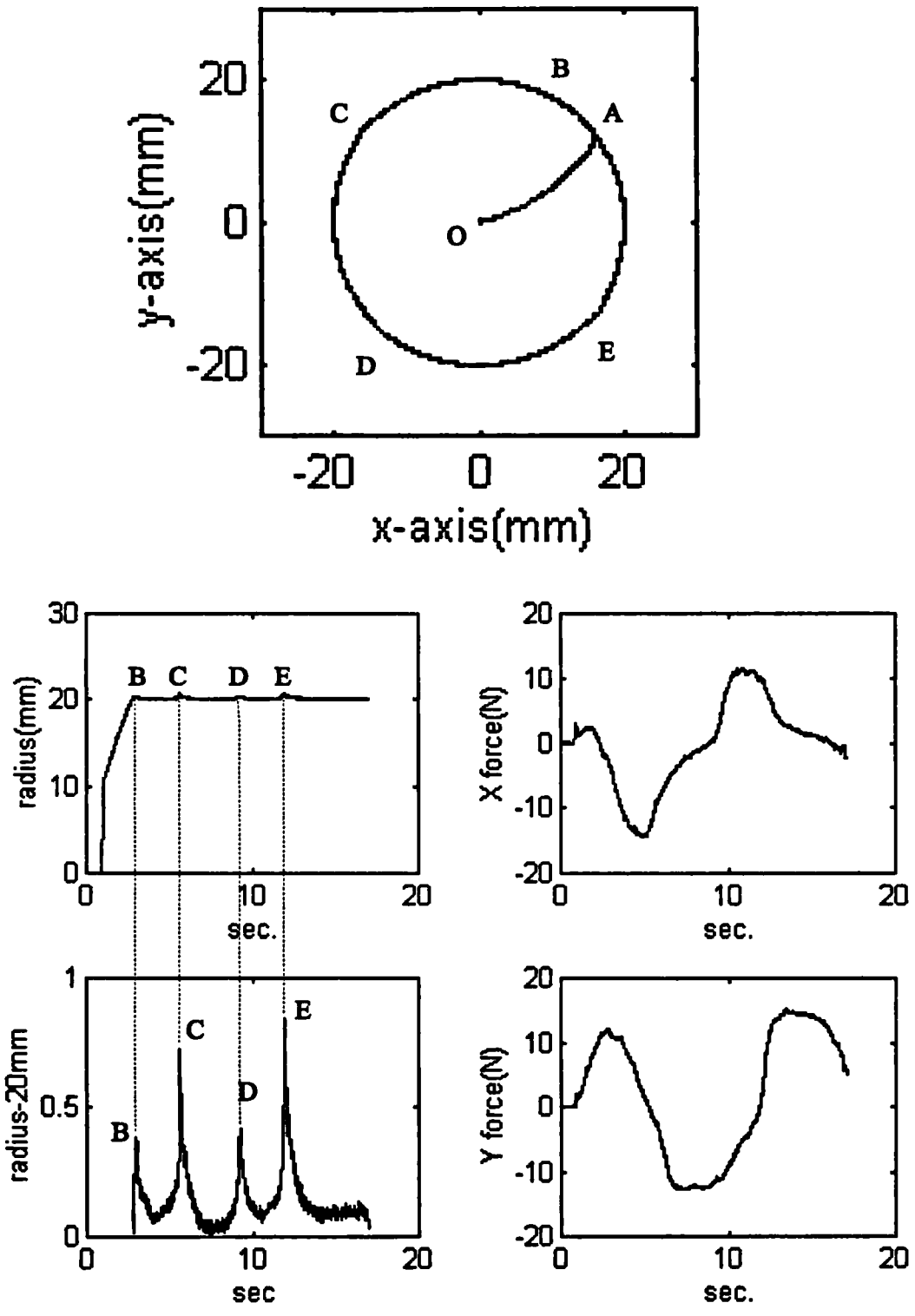


Fig. 7.17 As for Fig. 7.16, but with different Cartesian stiffness.

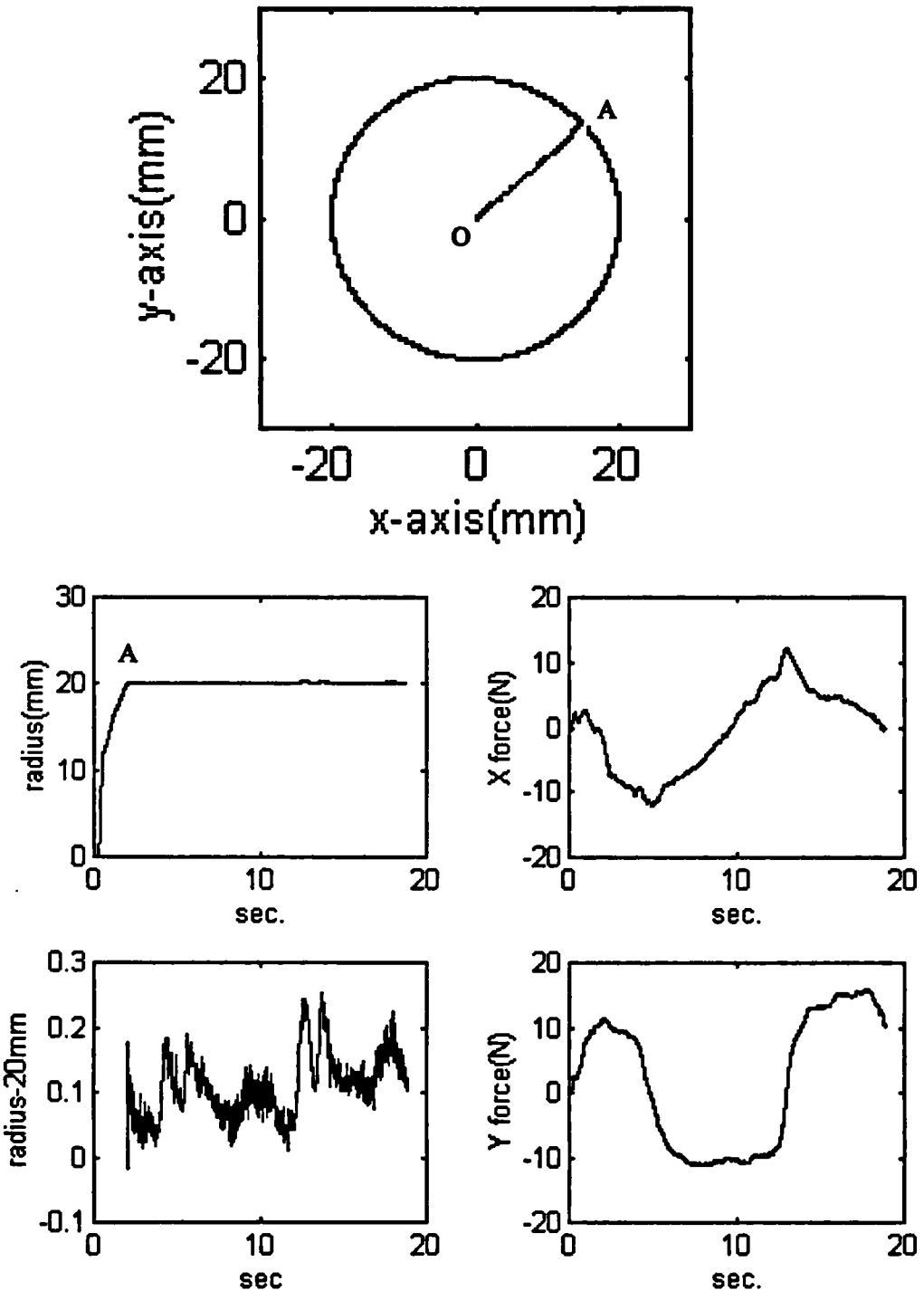


Fig. 7.18 As for Fig. 7.16, but with different Cartesian damping.

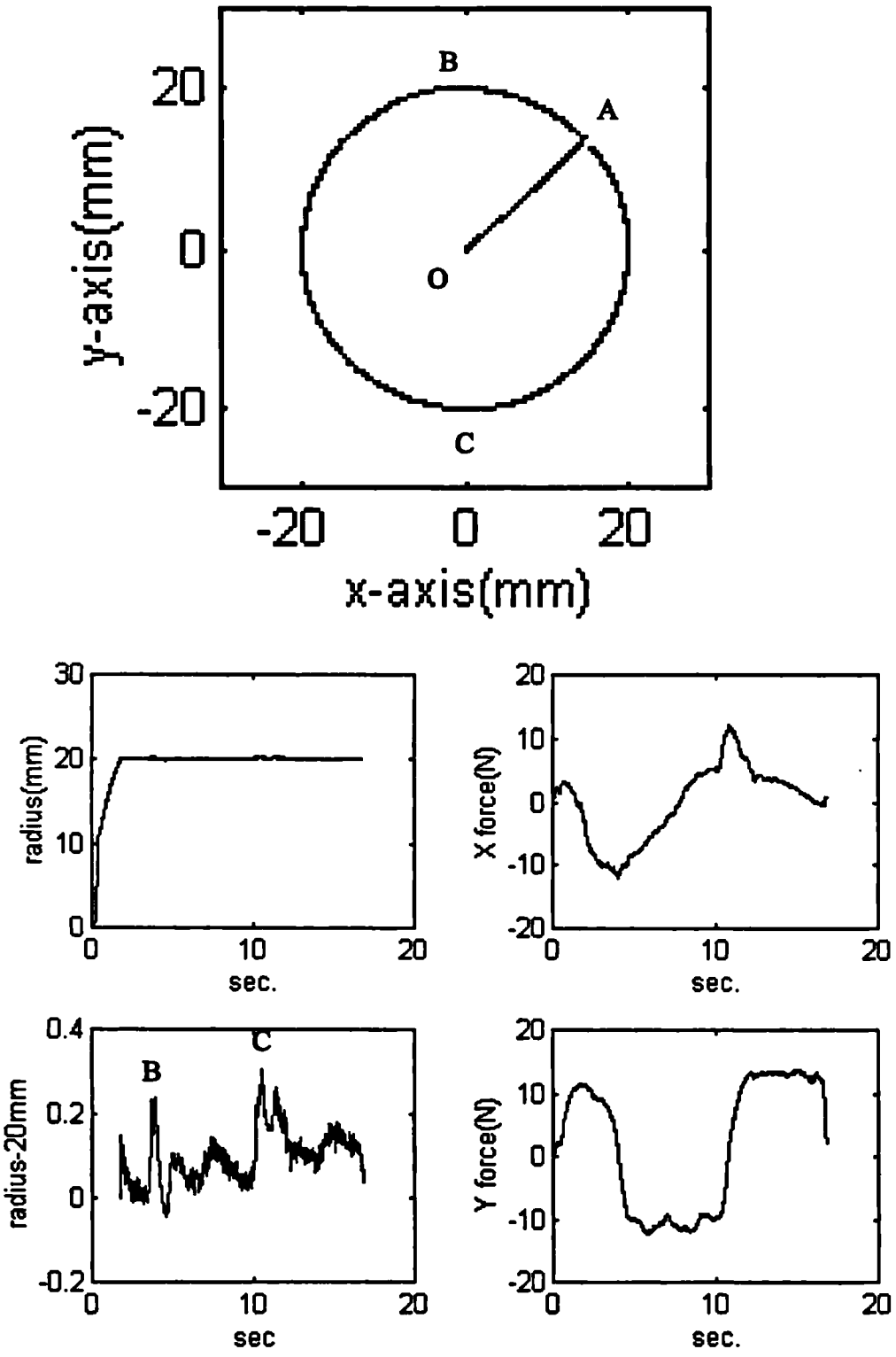


Fig. 7.19 As for Fig. 7.16, but with different Cartesian damping.

7.4 Modified damping control

Fig. 7.21 shows the result of a circular constraint defined as that of Fig. 7.16, but with modified damping control. The control gains K_p, K_d and K_f are given as those for simulation of Fig. 6.15. Force signal forces are processed through a saturation with a dead zone filter as shown in Fig. 7.20. The lower limit f_1 is used to reduce the effect of the noise, and the upper limit f_2 is to limit the maximum force command of the control algorithm. Here, f_1 and f_2 are given as 30 and 1500, respectively. The experimental result shows that by modified damping control the manipulator can be operated more easily and also have smaller position error compared to that by implicit force control. Results of Fig. 7.22 and Fig. 7.23 have drawn the same conclusion.

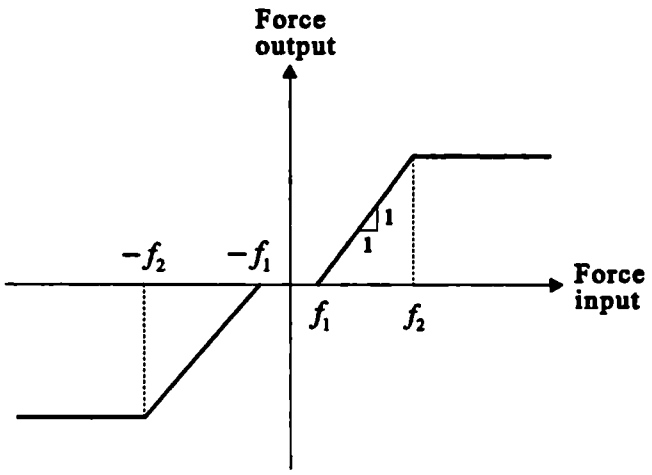


Fig. 7.20 Saturation with a dead zone filter

In the stability analysis (section 5.4), it has been concluded that increasing K_f will increase the sensitivity of the guiding force. As a result, the robot will be more compliant of following the external force command. However, this will decrease the system damping and reduce the ability of rejecting the impact force upon cutting. In order to verify this analysis, experiments in which the cutter is firmly held by hand or pushed against steel deliberately have been tested. From experimental result, it has been found that when K_f is given over 0.94 and the upper limit of the force signal f_2 equals 1500, vibration occurs for the above situations. However, if f_2 is given as 1000, K_f can be increased to 0.057 before vibration happens. Disadvantage of

lowering f_2 is that the control algorithm will lose the sensitivity for external force over than f_2. In practice, depending on the cutting conditions and design of the mechanism of the manipulator, these gains should be determined empirically.

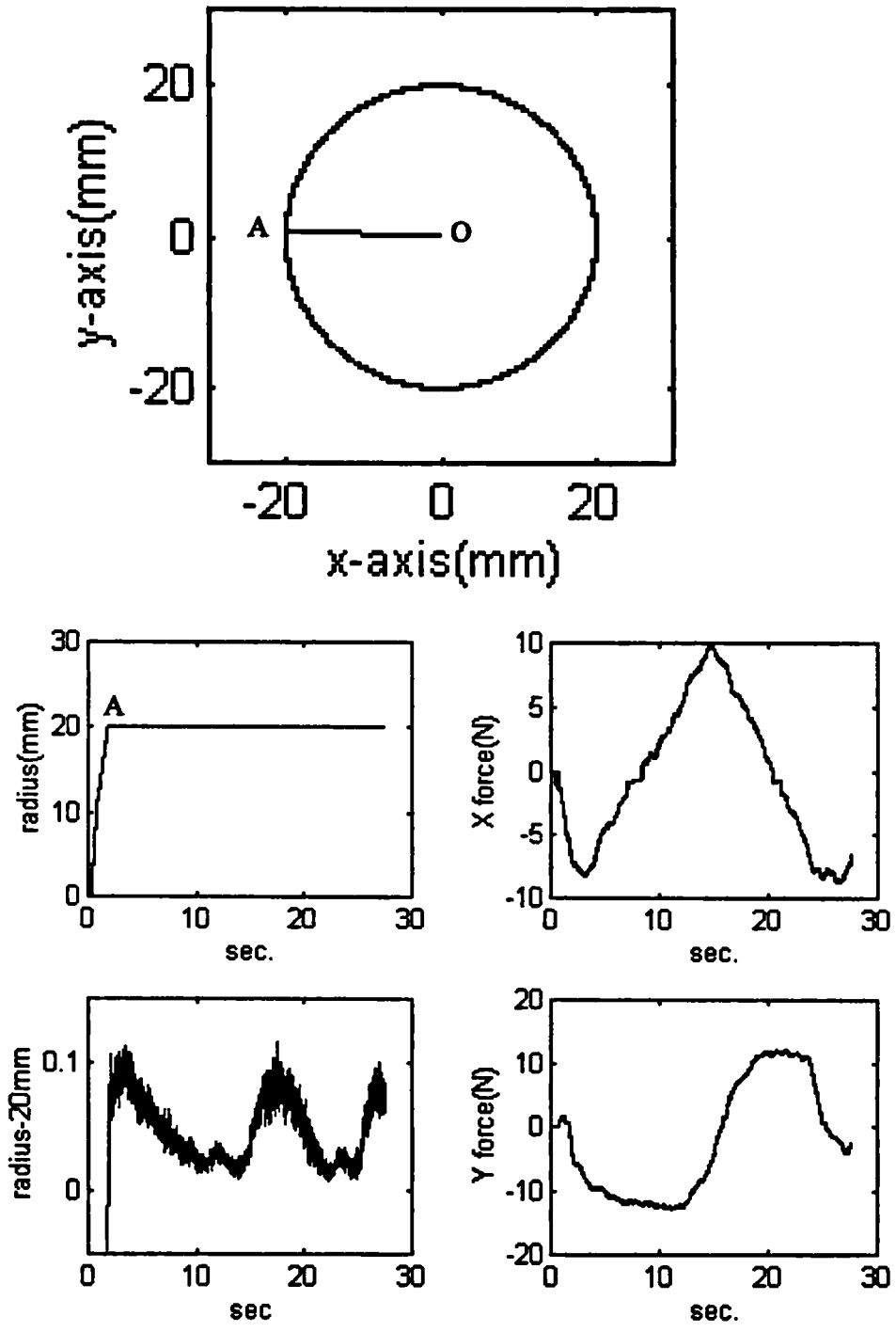


Fig. 7.21 Force and displacement characteristics as for Fig. 7.16, but with modified damping control

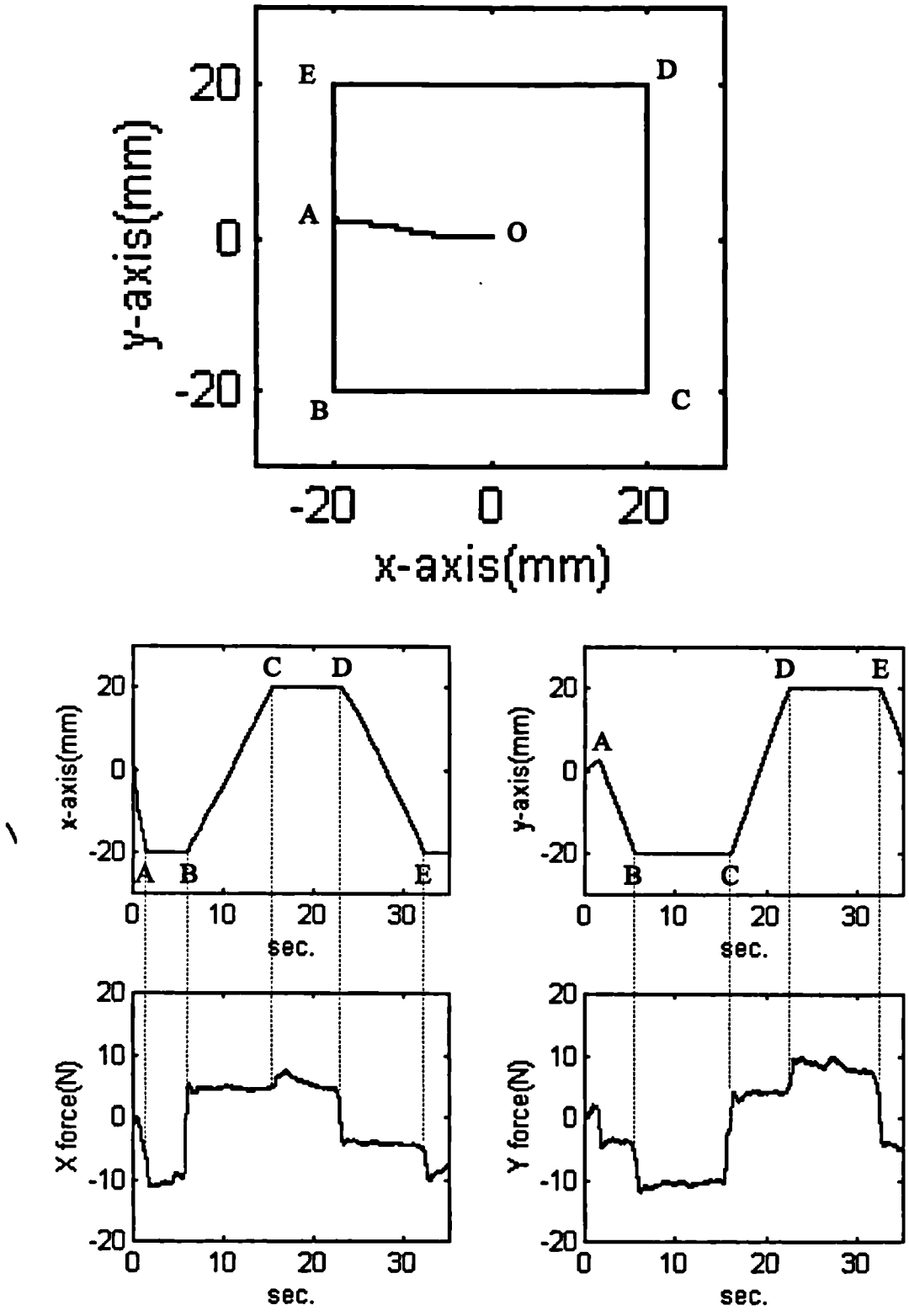


Fig. 7.22 Force and displacement characteristics as for Fig. 7.13, but with modified damping control

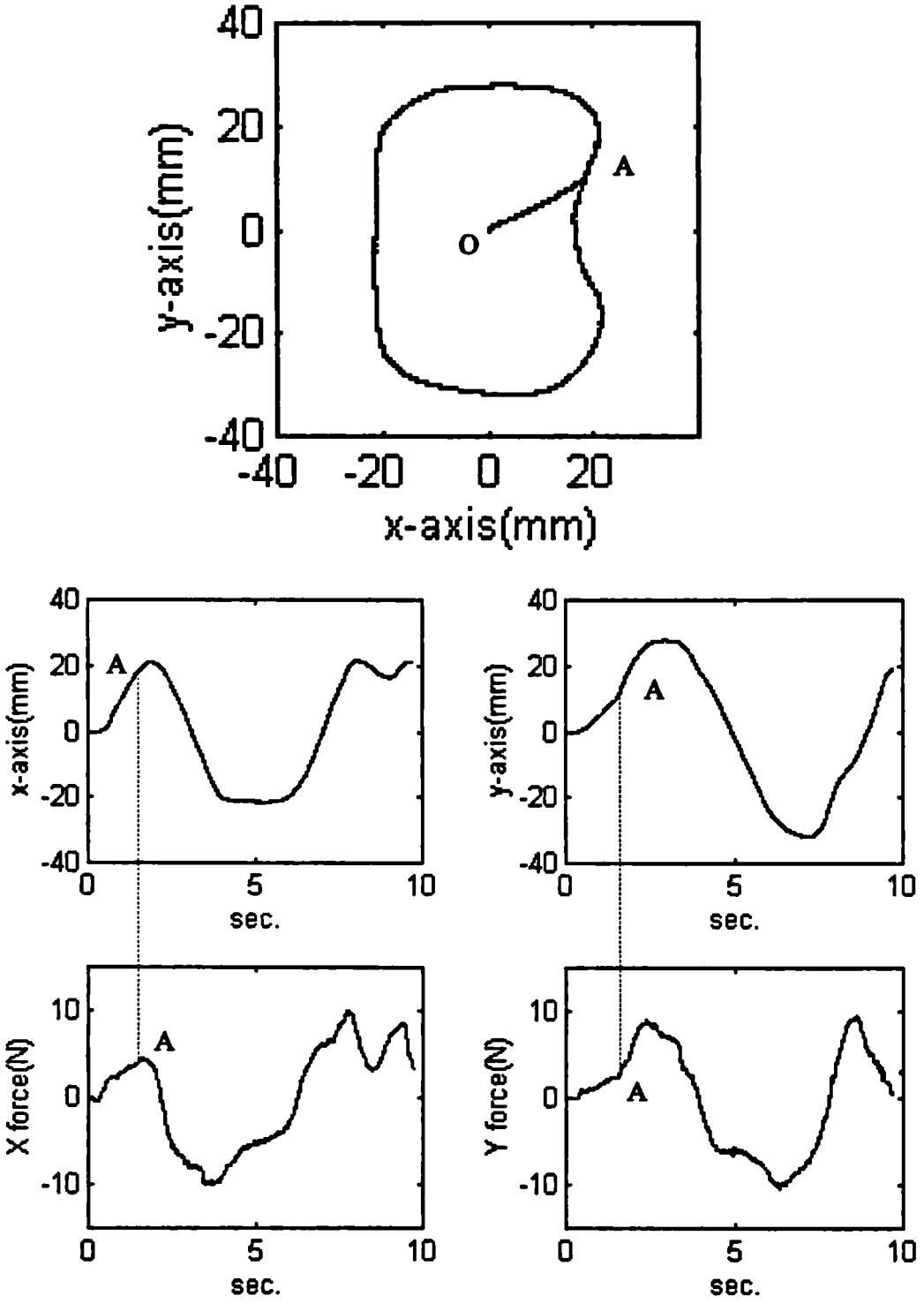


Fig. 7.23 Force and displacement characteristics as for Fig. 7.14, but with modified damping control

7.5 Cutting examples

The cutter (see Fig. 7.2) used is modified by a 25W high speed drill from RS Components Ltd. The drill weighs 190 grams and its maximum speed without load can reach 16,000 rpm. The conventional drill has been replaced by an end-milling cutter with a diameter of 5 mm.

7.5.1 Cutting of polystyrene foam

Material of high density foamed polystyrene was first used for cutting experiments. When the speed of the manipulator was 6 mm/sec., the down cutting resistance was about 1.5N. Cutting error was about 0.5 mm caused by vibration of the milling cutter. Fig. 7.24 is an cutting example of 40 mm straight line constraint. The actual desired length of cut was 45 mm, which is equal to the length of constraint plus the diameter of the cutter. Fig. 7.25 shows the result of a rectangular motion constraint with length of 40 mm, and Fig. 7.26 is the result of a circular constraint with a radius of 20 mm. Fig. 7.27 is the cut defined as in Fig. 7.14. The experimental results have shown the total cutting error is less than 1 mm either by implicit force control or by modified damping control. However, by using the modified damping control algorithm, the desired stiffness can be designed more easily and can also attain a smaller position error.

The above cut examples are all column shapes of motion constraint, i.e. the constraint along Z axis remains the same. Fig. 7.28 shows an example of motion constraint with a cone shape. Fig. 7.28(a) is the top view of the cut and Fig. 7.28(b) is a sectional view of the cone.

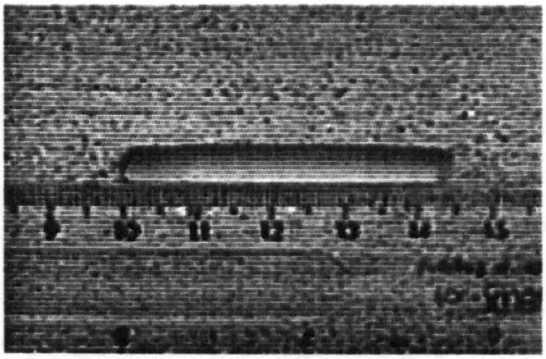


Fig. 7.24 Cut of a straight groove

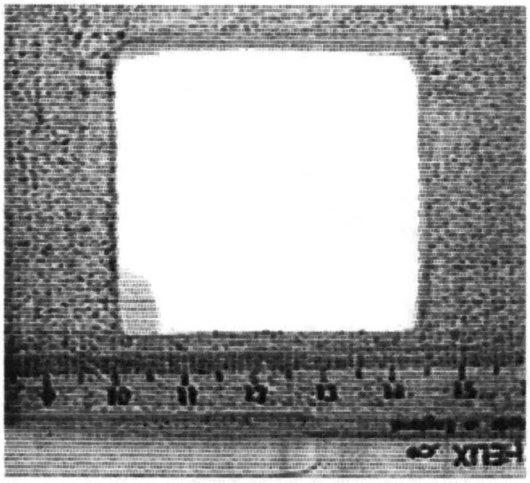


Fig. 7.25 Cut of a rectangular shape

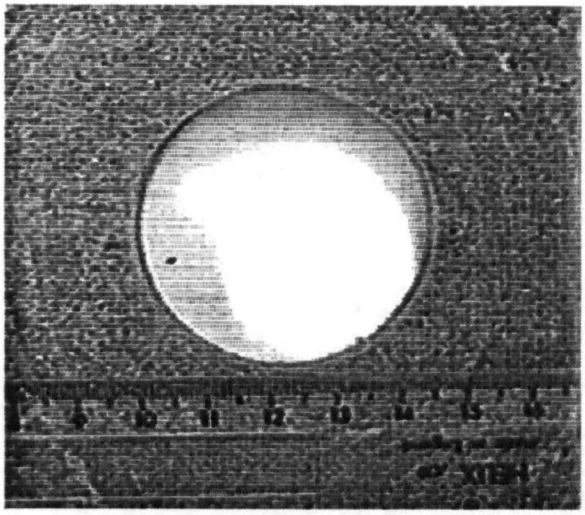


Fig. 7.26 Cut of a circular shape

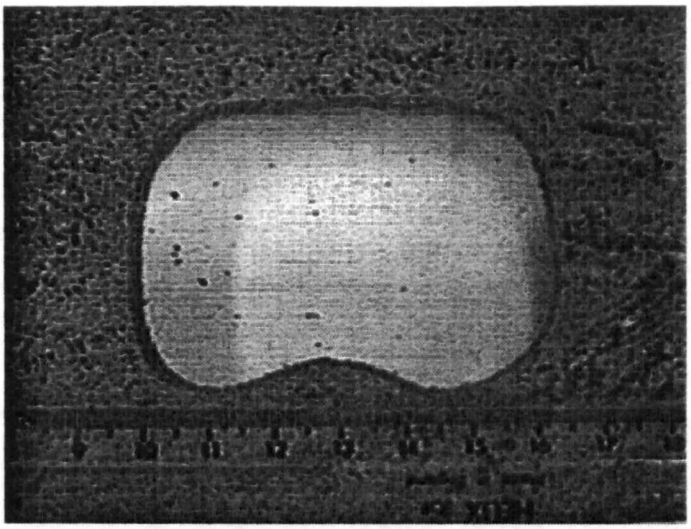
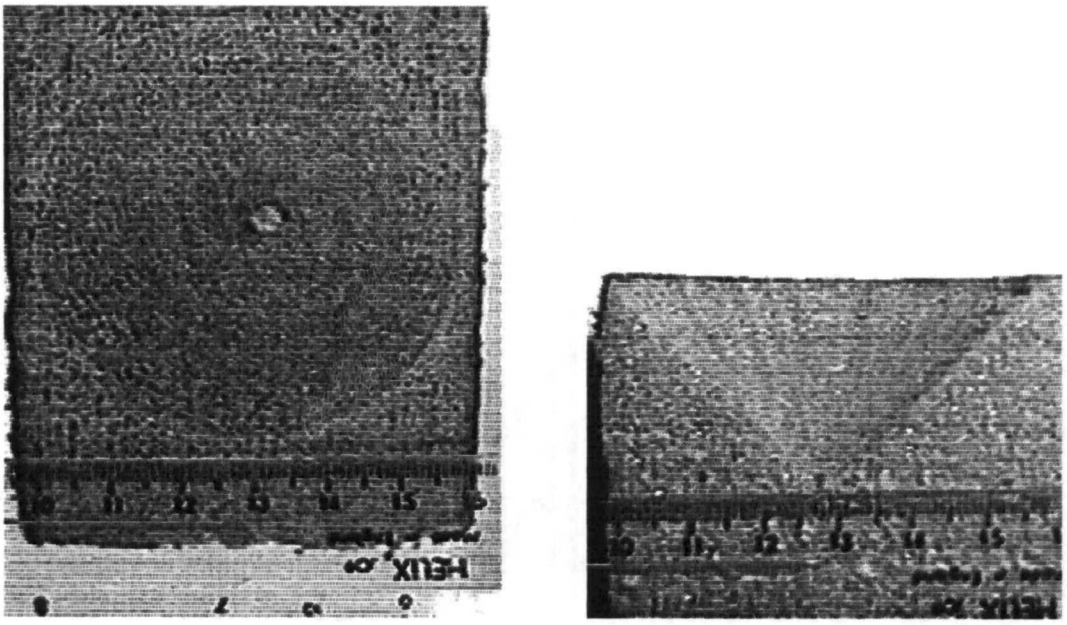


Fig. 7.27 Cut of a tibial shape



(a) (b)

Fig. 7.28 Cut of a cone shape

Another application is to attain a complex shape by combining simple shapes of the motion constraint. Taking the constraint defined in Fig. 7.29 as an example, it consists of a circular column and a half sphere constraints. By cutting off the shaded area, a desired dome can be obtained. The result is shown in Fig. 7.30.

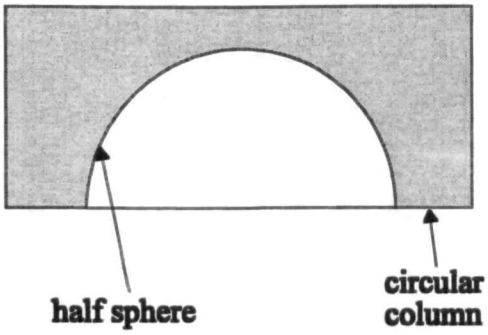


Fig. 7.29 Combination of different shapes of motion constraint

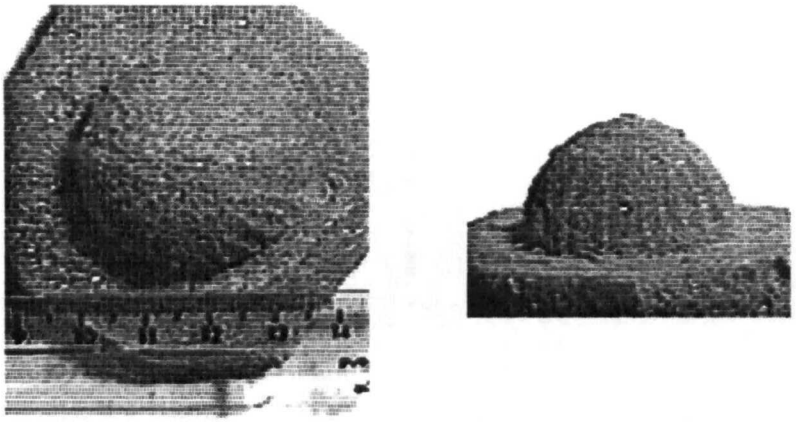
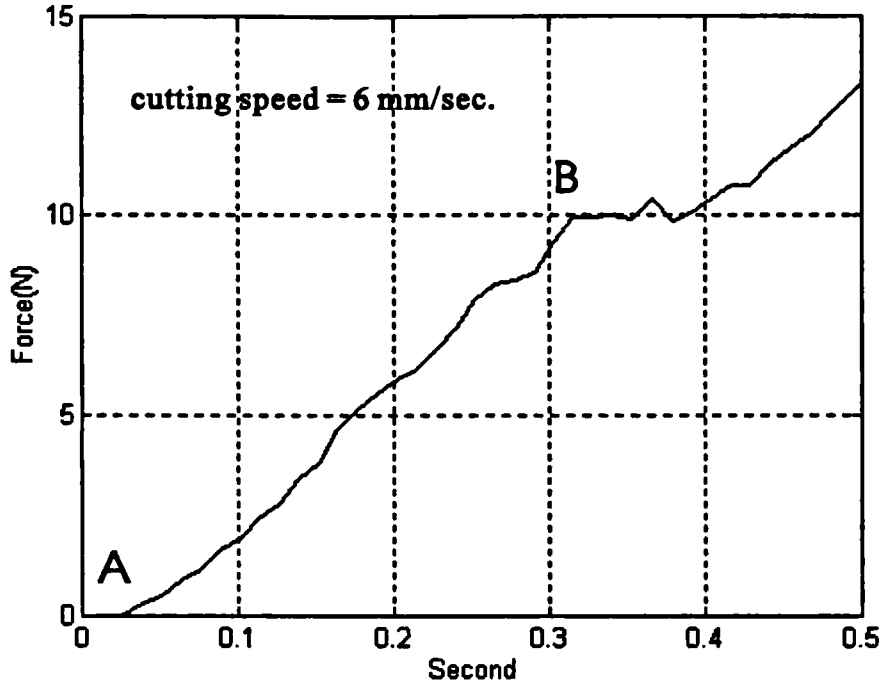


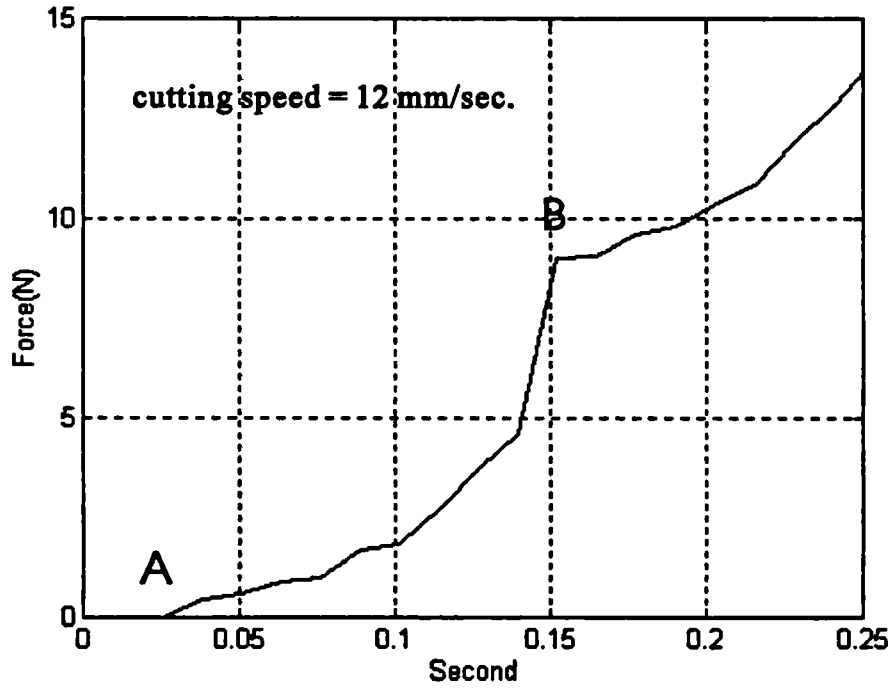
Fig. 7.30 Cut of a dome shape

7.5.2 Cutting of animal bones

Cutting experiments on beef and pork bones have been carried out. The end-milling cutter used is the same as that for the tests in the previous section. Cutting forces for animal bones are much higher than those of polystyrene foams. Fig. 7.31 (a) and (b) show experimental results of the down cutting resistance of beef bones when the translation speed of the cutter remains 6 mm/sec. and 12 mm/sec. respectively. The cutter starts cutting the bone at point A, and cuts through the outer skin at point B. The cutting resistance steadily increases as the contact surface between the cutter and bones becomes larger. However, it can be seen that the cutting resistance drops a little bit when the cutter is cutting through the outer layer of the bone.



(a)



(b)

Fig. 7.31 Down cutting resistance of beef bones

Photos of the robot and a cutting sample of beef bone are shown in Fig. 7.32 and Fig. 7.33, where the bone is firmly fixed to the table by a clamp. Fig. 7.34 shows an

example of the current position of the robot relative to the constraint on a computer screen when cutting is in progress. In the left half of the screen, the constraint of a typical tibial shape is profiled. The red region shows the area been cut and the white circle represents the current position of the robot, whose coordinates are shown in the upper corner. The red vertical line is the depth constraint, and the dark blue shows the deepest being travelled. The current depth of the robot is represented by the white line. Fig. 7.35 shows cutting examples of beef bones. The left hand side is a tibial shape cut in a beef hip. A V-groove and a hole are also cut in the middle of the bone. The right hand side of Fig. 7.35 is the cutting result of a dome shape constraint on a knee joint of a beef bone, which proves the capability of the force control strategy for the unicompartmental knee replacement. Experimental results show that the overall cutting error of animal bones by the rig is less than 1 mm when the final results are compared to the computer design.

7.6 Conclusion

In this chapter, both implicit force control and modified damping control have been verified by experimental evaluation. Position error can be effectively controlled within 0.5 mm when the external force is equal to 20N. By using a modified damping control algorithm, the robot can be operated more easily and a smaller position error can be attained. Cutting tests on high density polystyrene foams and animal bones have also been performed. When cutting animals bones, the impact cutting force is quite large which normally produces instability problems for an autonomous robot. Using this approach, the surgeon holds the robot and moves it, and then he/she can directly control the cutting process. Upon cutting a hard bone, the surgeon can approach the bone slowly and cut it through little by little. Therefore, the robot will not bounce back upon cutting. From experimental results, the feasibility of using these force control algorithms for robotic knee surgery has been demonstrated.

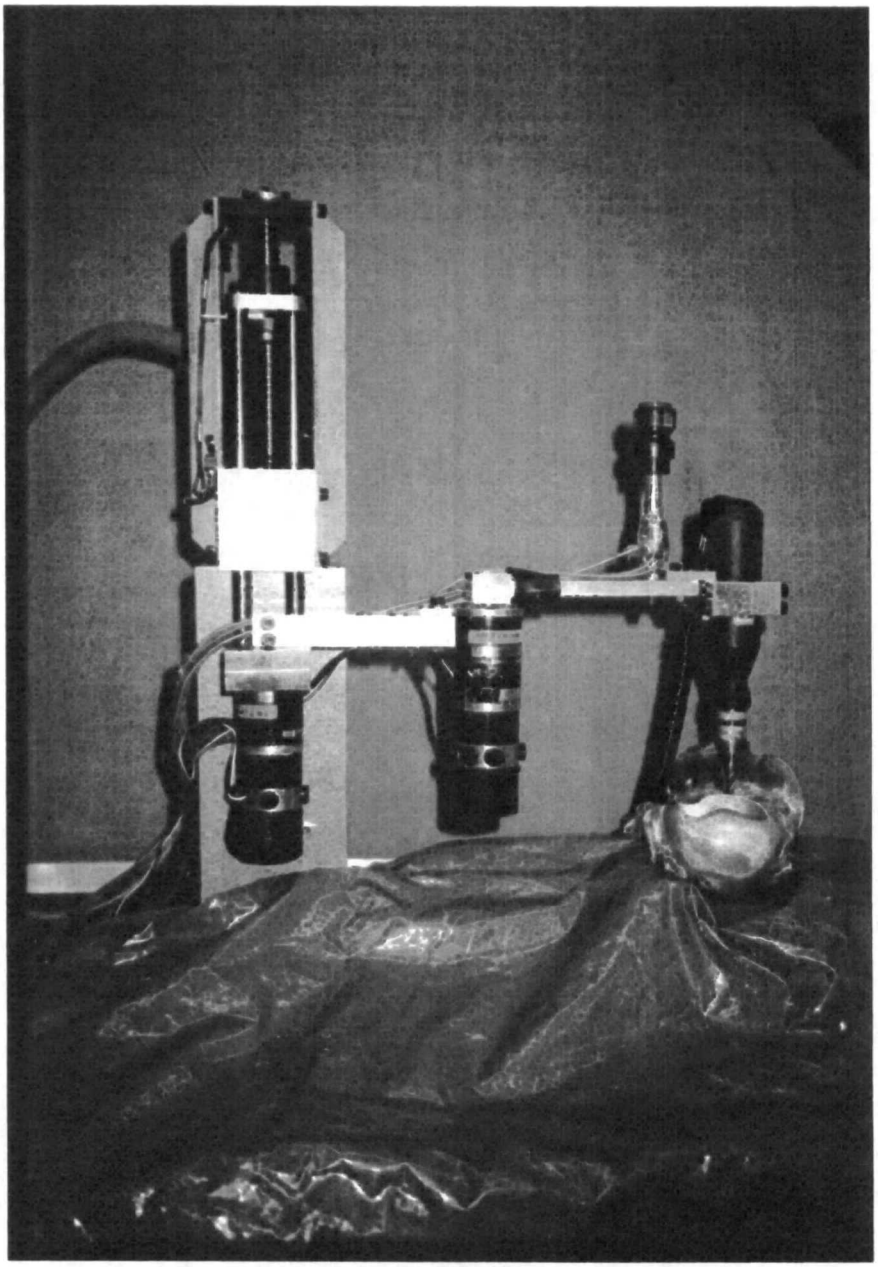


Fig. 7.32 Cutting set-up of the robot and a beef bone

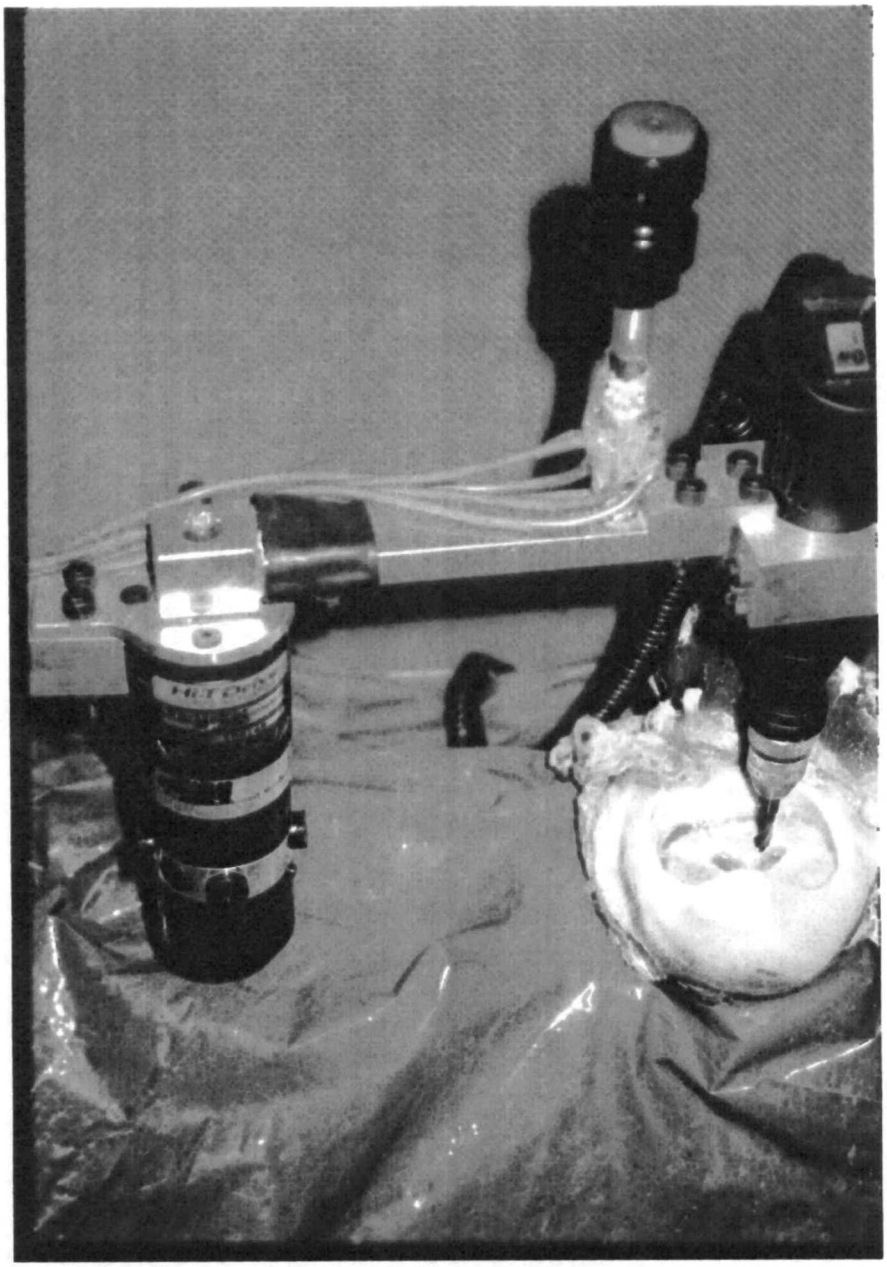


Fig. 3.33 Close up view of Fig. 3.32

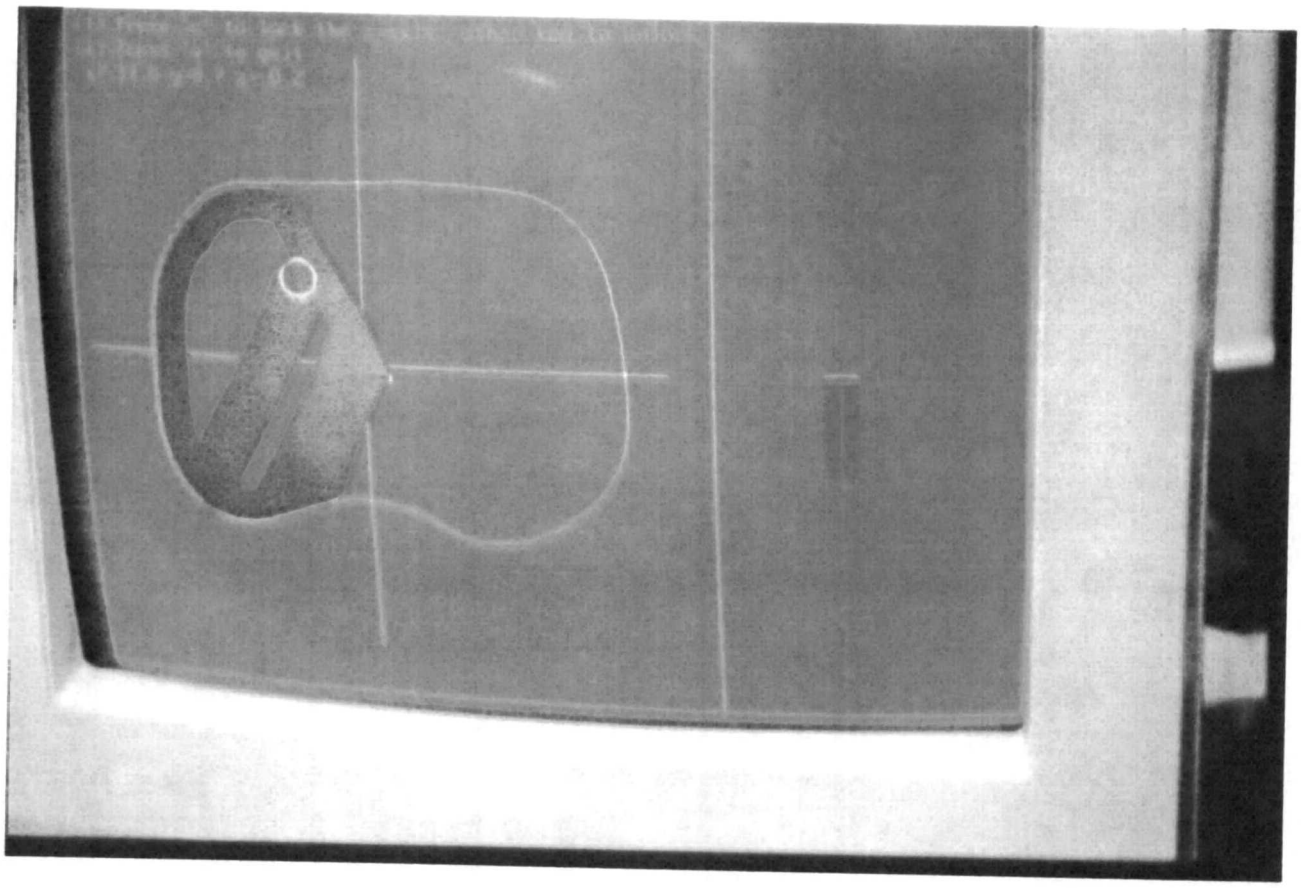


Fig. 3.34 Trace of the robot and a 3D column constraint with a tibial shape in XY plane shown on the computer screen when cutting is in progress.

Fig. 3.35 Cutting results for beef bones.

CHAPTER 8

CONCLUSIONS

Starting from the 1980's, the use of robots has gradually branched away from the bounds of industrial tasks. Among these new areas of robot applications, service robots are believed to have the greatest growth potential and will be more important than manufacturing robots in the future. Of the various applications of service robots, the development of medical robots has attracted much attention recently. Some of the main reasons are (1) the age of the population is steadily increasing in most countries, while the number of young workers is decreasing. Thus, the social pressures of taking care of ageing people is becoming increasingly serious. (2) improved technologies make applying robots in health-care possible and economically justifiable. (3) both medical staff and patients have expected (and accepted) advanced automation in health-care.

The use of surgery robots offers the greatest scope for the medical use of robots. Surgical robots are used because they can provide precise and repeated motions in response to pre-programmed tasks. Usually some form of imaging system, such as Computed Tomogrphy (CT), Magnetic Resonance Imaging (MRI), and Positron Emission Tomography (PET) etc., is used pre-operatively to define the surgical procedures. The robot is then registered with reference to the patient so that the sequence of motions can be automatically carried out.

Of the many applications of medical robots, this thesis has concentrated on the feasibility of the use of force control for robotic knee surgery. The conventional surgical procedures have some drawbacks. Firstly, the operation is not strongly linked to the preoperative planning, the surgeon can only execute the resection with the limited view available during the operation. Secondly, using a jig system, each cut is dependent on the quality of the previous jig location and drill. Thirdly, the cuts completed by the oscillating saw lack accuracy and flatness, and harmful heat can be

generated. If the saw is slid along the jig surface, it tends to bounce off the bone. The surgeon therefore angles the blade a little: too much and the blade digs into the bone excessively, too little and it bounces off. In addition, the range of sizes required also means that many jigs have to be kept and the procedure must be highly systematic to assure an adequate quality. The use of a robotic system and a better cutting method, such as using a rotary milling cutter, can ensure that each cut is correct with respect to all others and the tibial and femoral aspects are correctly aligned and inter related.

When using an active (autonomous) robot, there are a number of questions and problems that need to be addressed: how to satisfy the safety requirements; how to transfer the surgeon's experience and knowledge of surgical techniques to the robot and how to implement artificial sensors which replace the surgeon's senses (force, touch, vision, and sound etc.). Also, the need for psychological acceptance, both by patients and surgeons, in using an autonomous robotic manipulator creates further difficulty. A passive robot that has no actuators does not have the same risk as an active robot, which, if adequate safety precautions are not taken, may execute unexpected motions or cutting in the case of a malfunction. The surgeon can move the cutting tool and display its current position on a computer screen. However, a passive arm is usually not good at following a desired trajectory (for instance, cutting a groove) or reaching and maintaining a pre-computed point.

This thesis has presented a new approach for assisting in the execution of resection in knee surgery for prosthetic implants without the use of a complex jig system. A semi-active robot contains a rotary cutter which is moved by the surgeon by hand, hence the surgeon can execute the operation fully under his control using his innate sensing, judgement and experience. From the point of view of both the surgeon and the patient, the robot is merely a "tool". It is evident that the surgeons perform the operation and not the robot. For these reasons, it is believed that this approach is more acceptable to both surgeon and patient.

In order to assist the surgeon in executing the pre-planned cuts easily, a new concept of artificial motion constraint formed by a force control strategy has been investigated.

Details of objectives of the motion constraint can be found in chapter two. With the help of the artificial motion constraint, the surgeon can efficiently and accurately complete the resections. In addition, cooperation between the surgeon and the robot is very simple and easy.

There are some differences between the industrial robot applications and this task. First, the guiding force applied by the surgeon cannot be ignored and should be taken into account in the closed loop system. Second, the robot control strategy has to be a function of the current position of the robot and the direction of motion, thus the robot can behave with variable stiffness as required. Finally, an on-line trajectory "interpreting system" for the robot will be needed to assist or constrain the movement by the surgeon in a pre-planned trajectory or region.

In general, difficulties of implementing the concept of the motion constraint are:

(i). How to specify and construct an artificial motion constraint in three dimensional space.

(ii). How to design the desired position and velocity of the robot based on the robot current position, the guiding force and the motion constraint.

(iii). How to design the control gains and then transform the desired stiffness from Cartesian space to joint space.

The method of defining a three-dimensional motion constraint and representing it as a lookup table in the computer has been developed (see chapter 5). The lookup table of the motion constraint can be easily accessed by the force control algorithm and the desired trajectory and gains for the robot can be calculated in real time. Subsequently, an effective on-line algorithm for designing the desired trajectory of the manipulator has also been proposed.

The force control strategy includes implicit force control and modified damping control. The design of implicit force control, in which a force sensor is not involved, can be divided into independent joint control method and Cartesian stiffness design. The independent joint control is the easiest and also has the least computational time

consuming algorithm when implemented. However, it is difficult to design the desired stiffness in Cartesian space. The Cartesian stiffness design can most easily specify the desired stiffness in Cartesian space. Nevertheless, the position error and velocity in each joint will affect the output commands in all other joints, which in turn reduces the disturbance rejecting ability of each joint.

The main difference between modified damping control and implicit force control is that the former method involves a force sensor in the system. There are three main advantages of using a force sensor to measure the surgeon's pushing/pulling force. Firstly, by sensing the guiding force, the control law can command the robot to follow the operator's desired movement, especially when the robot is difficult to move by hand because of high friction or the configuration of the mechanism of the manipulator. Secondly, the Cartesian stiffness design of the implicit force control algorithm can produce the desired variable stiffness and damping along the X-Y-Z axes. However, it will be impossible to have a different stiffness such as in the -X and +X directions by using the implicit force control technique. Finally, the force sensor can be used as a redundant safety checking mechanism, where a 'safe' force level is pre-defined. There are also two design methods in the modified damping control algorithm: independent joint control and the non-linear decoupling technique. The first method is similar to the independent joint control method in implicit force control except that a desired velocity for the robot can be designed. It is easy to implement and proves to be stable. However, if the robot is moving at high speed, the non-linear decoupling technique may have to be used to reduce position errors caused by the decoupled forces. The stability of the proposed force control algorithms has also been analyzed in section 5.4., and the following conclusions were drawn:

(1). In implicit force control, the force feedback K_f is zero. As a result, the guiding force does not affect the characteristic equation of the system, while the stiffness between the cutter and the environment decreases the system damping.

(2). In modified damping control, the stiffness resulting from the cutter, the force sensor and the operator's hand K_{E2} is multiplied by the derivative feedback gain

K_d and the force feedback K_f, if the guiding force is dependent on the robot position. This will result in very high gain 'position' control, and under damp the overall system. In other words, increasing K_f will increase the sensitivity of the guiding force, which makes the robot more compliant in following the force command. However, on the other hand, this will decrease the system damping and thus reduce the ability to reject disturbance forces or the impact force upon cutting.

(3). From experimental experience, limiting the upper value of the force input F_s in the control law can increase the stability of the system. From another point of view, setting the upper limit of F_s means that the stiffness K_{E2} is artificially reduced in cases where the external force is high.

In chapter 6, computer simulations have been carried out to examine the analysis of the force control strategy. Performances of different algorithms have been compared, and some points have been concluded:

(1). The sampling interval of the interpreter of the desired trajectory and control gains, T_{s1}, is not critical to the stability of the system. However, a slow T_{s1} will cause the movement to become 'jerky' in operation. The effect is further worsened by a time delay of T_{s1} of the interpreter program.

The 'potential' energy of the manipulator, due to the proportional feedback, is released after every T_{s1} interval. If T_{s1} is longer, the resistance force from the controller will become higher during every sampling interval T_{s1}. As a result, it will become difficult to operate the manipulator dextrously.

(2).The sampling rate of the inner control law is more critical to the stability of the overall system. Some factors that could provide a lower limit to the acceptable sampling rate are: tracking effectiveness of reference inputs, disturbance rejection, sensitivity to plant-parameter variations and structural resonances. The fundamental lower bound on the sampling rate is twice the system bandwidth in order for the closed loop system to track a reference input.

(3). Integral feedback can reduce the position error, however, which will increase the resistance force. If the updating rate of the control gains is too slow, adding integral gain to the control law will not improve the performance because of the time delay.

(4). When implementing implicit force control, the independent joint control algorithm is the easiest to implement. However, it is difficult to design the desired stiffness in Cartesian coordinates. The Cartesian stiffness design method, on the other hand, can easily specify the desired stiffness and damping in Cartesian space. From simulation results, it has been shown that by using Cartesian stiffness design, the manipulator is easier to be moved along the boundary and the position error is smaller compared to that of independent joint control. However, it should be noted that the position error and velocity in each joint will affect the torque command in all other joints in the Cartesian stiffness design algorithm. This will decrease the disturbance rejecting ability of the manipulator in reality.

(5). There are two ways of implementing a modified damping control algorithm: independent joint control and nonlinear decoupling technique. The latter method will be chosen if the effect of inertia forces is significant. From a performance point of view, modified damping control is better than implicit force control. The disadvantage is that a force sensor is needed to measure the guiding force. This will increase the cost and difficulty of mechanism design for the manipulator.

A three-DOF manipulator with force sensors has been built up for experimental tests. Details of the hardware have been described in chapter 7. The performance of the force control strategies have been further examined in experiments. Three-dimensional examples of cutting have also been carried out in a range of materials by using an end-mill cutter. From the experimental results, the feasibility of the concept of artificial motion constraint and the controller design strategy have been again confirmed.

When applying this new method of artificial motion constraint to robotic knee surgery, the force level of the robot has to be considered in order to attain an optimal design. Human sensing of force cannot be as accurate as an artificial sensor.

Normally, people can only estimate external load or weight roughly. For instance, loads under 0.9 kgw (9N) are 'light' to most people, while loads over 2.5 kgw (25N) become 'heavy' to human sensing. Between those two limits, the force level is not so obvious to most people [Paulat 92]. Therefore, it is recommended that the maximum output of the robot is given as 25N when designing the manipulator. Thus, the force level can remain low but is still adequate for 'restricting' the surgeon from moving the cutter out of the defined constraint.

The surgeon can trace the position of the manipulator from the computer screen and use this to perceive the information of the relative distance between the current position of the manipulator and the boundary of motion constraint. However, it is still desirable to have an obvious difference of force levels between the cutting resistance and the output resistance of the manipulator. Thus, the surgeon can distinguish the resistance force in different zones easily. For instance, when the surgeon approaches the boundary of the motion constraint, he/she can acknowledge a clear 'warning' signal because of the increase of resistance. The magnitude of cutting resistance depends on the machining conditions such as the shape of the cutter, the length of arc of cut, and the cutting speed in addition to types of tissue to be cut. When designing the machining conditions, it is recommended that the applied force be designed to be below 9N (light load to human) when cutting the hardest tissue (bone).

In this thesis, it has been demonstrated that a preoperative plan can be executed accurately and precisely with the help of an artificial motion constraint formed by a force control strategy. The surgeon's innate sensing, judgement and experience are always preserved, and cooperation between the surgeon and the robot is very simple and easy. Because the surgeon can sense the resistance to cutting directly, he/she can slow down the rate of cutting or take a lighter cut. However, further work is required to demonstrate that the concept of this force control for robotic knee surgery can be applied in an operating room. A special purpose robot will have to be designed and placed on a base that is capable of positioning, orienting and datuming with reference to the knee location. Following laboratory studies, it will be necessary to conduct

cadaver investigations to ensure a good understanding of the clinical requirements before conducting clinical trials on patients. Only at this stage will it be possible to evaluate the real benefits of robotic knee surgery against the disadvantages of the imaging, modelling, clamping and datuming that are necessary for effective robot surgery. Although active motion constraint has been applied in this thesis to knee surgery, the capability of accurate bone resection could also be readily adapted to assist in a range of other orthopaedic surgery tasks. The judgement of the medical community on this novel form of force control with an artificial motion constraint for robotic orthopaedic surgery is awaited with interest.

REFERENCES

Papers & Books :

[Adams 89] Adams J., Gilsbach J., Krybus K., Ebrecht D., Mosges R., "CAS - A Navigation Support for Surgery", IEEE Eng. in Med. & Biology Soc. 11th Int. Conf., 1989.

[Adams 92] Adams L., Knepper A., Krybus W., Meyer-Ebrecht D., Pfeifer G., Ruger R., "Orientation Aid for Head and Neck Surgeons," Innovation and Technology in Biology and Medicine, pp 409-425, Vol. 13, No. 4, 1992.

[Ammeraal] Ammeraal L., "Programming Principles in Computer Graphics," John Wiley & Sons.

[An 87/a] An C.H., Hollerbach J.M., "Dynamic Stability Issues in Force Control of Manipulators," pp. 890-896, Proc. IEEE Conf. of Robotics and Automation, 1987.

[An 87/b] An C.H., Hollerbach J.M., "Kinematic Stability Issues in Force Control of Manipulators," pp. 897-903, Proc. IEEE Conf. of Robotics and Automation, 1987.

[Andrew 62] Andrew C., Tobias S.A., " Vibration in Horizontal Milling," pp 369-378, Vol. 2, Int. J. Mach. Tool Des. Res., 1962.

[Asada 86] Asada H., Slotine J.J.E., "Robot Analysis and Control", John Wiley and Sons, 1986.

[Asfahl 85] Asfahl C.R., "Robots and Manufacturing Automation", John Wiley & Sons Inc., 1985.

[Beltrame 91] Beltrame F., et al., "A Laboratory for Computer-Assisted Orthopaedic Surgery", pp882-887, Proc. of 5th Int'l Conf. on Advanced Robotics, June 1991.

[Brogan 85] Brogan W.L., "Modern Control Theory", Prentice-Hall Inc, 1985.

[Cinquin 92] Cinquin P., Demongeot J., Troccaz J., Lavallee S., Champleboux G., Brunie L., Leitner F., Sautot P., Perez B., Djaid M., Fortin T., Chenin M., and Chapel A., "IGOR : Image Guided Operating Robot. Methodology, Medical Applications, Results.," Innovation and Technology in Biology and Medicine, pp 373-393, Vol. 13, No. 4, 1992.

[Craig 86] Craig J.J., " Introduction to Robotics : Mechanics & Control," Addison-Wesley 1986.

[Cugy 84] Cugy A., "Industrial Robot Specifications", Kogan Page Ltd, 1984.

[Cutkosky 86] Cutkosky M.R., Wright P.K., " Active Control of a Compliant Wrist in Manufactruing Tasks," pp. 36-43, Vol. 108, Tans. of ASME, Feb. 1986.

[Cutting 92] Cutting C., et al., "Computer Aided Planning and Execution of Craniofacial Surgical Procedures", pp. 1069-1071, Proc. of 14th Int. Conf. of IEEE EMBS, Paris, France, Oct. 1992.

[Davies 84] Davies B.L., "The Use of Robots to Aid the Severely Disabled", IEE Electronics and Power, pp 211-214, 1984.

[Davies 89] Davies B.L., Hibberd R.D., Coptcoat M.J., Wickham J.E.A., " A Surgeon Robot Prostatectomy - A Laboratory Evaluation, " pp. 273-277, Vol 13, No. 6, Journal of Medical Engineering & Technology, 1989.

[Davies 92] Davies B.L., Hibberd R.D., Ng W.S., Timoney A.G. and Wickham J.E.A., "Mechanical Contraints - The Answer to Safe Robotic Surgery ?," Innovation and Technology in Biology and Medicine, pp 426-436, Vol. 13, No. 4, 1992.

[Davies 93/a] Daveis B.L., "Safety of Medical Robots", pp 311-317, Proc. of 6th Int. Conf. on Advanced Roibotics, Tokyo, Nov. 1993.

[Davies 93/b] Davies B.L., "Safety of Medical Robots," pp. 311-317, Proceedings of the Int. Conf. of Advanced Robotics, 1993.

[Davies 93/c] Davies B.L., "Safety of Medical Robots", Safety Critical Systems, Book Chapman Hall, Part 4, Ch. 15, pp. 193-201, Spring 1993.

[Davies 94/a] Davies B.L., Hibberd R.D., "Opportunities for Robotic Technologies in Knee Arthroplasty," report of Centre for Robotics and Automated Systems, Imperial College of Science, Technology and Medicine, April 1994.

[Davies 94/b] Davies B.L., "Safety Critical Problems in Medical Systems", Proc. of Second Safety Critical Systems Conf., Birmingham, UK, Feb. 1994.

[Davies 94/c] Davies B.L., "Soft Tissue Robotic Surgery", Proc. first European Robots in Medicine Conf., Barcelona, Spain, June 1994.

[Davies 94/d] Davies B.L., "Medical Robotics ", PhD thesis, Imperial College, Univ. of London, 1994.

[Dohi 93] Dohi T., et al., "Robotics in Computer Aided Surgery", Proc. 6th Int. Conf. on Advanced Robotics, pp 379-383, Tokyo, Japan, Nov. 1993.

[Dorf 88] Dorf R.C., "International Encyclopedia of Robotics: Applications and Automation", John Wiley & Sons, 1988.

[Dowson 74] Dowson D., Swanson S.A.V., "Engineering Considerations in Total Knee Replacement," Instn Mech Engrs, 1974

[Dowson 81] Dowson D., Wright V., "Introduction to the Biomechanics of Joints and Joint Replacement," Mechanical Eng. Publications Ltd, 1981

[Drake 91] Drake J., Joy M., Goldengerg A., Kreindler D., "Computer and Roibotic Assisted Resection of Brain Tumours", pp 888-892, 5th Int. Conf. on Advanced Robotics, Pisa, Italy, June 1991.

[Edwards 90] Edwards R., "Robots in Medicine: Safety Aspects", Robotics in Medicine Seminar, IMechE, 1990.

[Engelberger 74] Engeelberger J.F., "Three Million Hours of Robot Field Exeperience", pp 164-169, vol. 1, The Industrial Robot, 1974.

[Engelberger 80] Engeelberger J.F., "Robotics in Practice", AMACOM, 1980.

[Engelberger 89] Engeelberger J.F., "Robotics in Service", Kogan Page Ltd, 1989.

[Eppinger 92] Eppinger S.D., Seering W.P., "Three Dynamic Problems in Robot Force Control," pp. 751-758, Vol. 8, No. 6, IEEE Trans. on Robotics and Automation, Dec. 1992.

[Fadda 92] Fadda M., Martelli S., Dario P., Marcacci M., Zaffagnini S., Visani A., "First Steps towards Robot-Assisted Discectomy and Arthroplasty," Innovation and Technology in Biology and Medicine, pp 394-409, Vol. 13, No. 4, 1992.

[Fadda 93] Fadda M., et. at., "Execution of Resections in Computer Assisted Orthroplasty", Proc. 6th Int. Conf. on Advanced Robotics, pp337-340, Tokyo, Japan, Nov. 1993.

[Finlay 89/a] Finlay P.A., "Medical Robotics - Why , What and When," pp37-39, Industrial Robot, 1989.

[Finlay 89/b] Finlay P.A., "The United Kingdom Advanced Medical Robotics Initiative," 11th Int. Conf. of IEEE Engineering in Medicine and Biology, 1989.

[Finlay 90] Finlay P., et at., "Report of a Project Definition Study Completed in Mar 1990", Commercial in Confidence, 1990.

[Fowler 91] Fowler H.C., Eppinger S.D., "Bandwidth Performance of a Direct Drive Manipulator Under Joint Torque and Endpoing Force Control," pp. 230-237, Proc. Int. Conf. on Robotics and Automation, California, April 1991.

[Franklin 86] Franklin G.F., Powell J.D., Emami-Naeini A., "Feedback Control of Dyanmic Systems," Addison Wesley, 1986.

[Freund 75] Freund E., "The Structure of Decoupled Nonlinear Systems," pp. 443-450, vol. 21, no. 3, Int. J. Contr. 1975.

[Fu 87] Fu K.S, Gonalez R.C., Lee C.S.G., "Robotics : Contorl, Sensing, Vision, and Intelligence," McGraw-Hill, 1987

[Funda 94] Funda J., Taylor R.H., et al., "An Experimental User Interface for an Interactive Surgical Robot", pp 196-203, Prod. First Syjp. on Robotic Surgery, Pittsburghy, USA, Sept. 1994.

[Gere 84] Gere J.M., Timoshenko S.P., "Mechanics of Materials," Wadsworth Inc. 1984.

[Goldenberg 88] Goldenberg A.A., "Implementation of Force and Impedance Control in Robot Manipulators," Proc. IEEE Conf. on Robotics and Automation, 1988.

[Goldstein] Goldstein H., "Classical Mechanics", Addision-Wesley.

[Hartley 83] Hartely J., "Robots at Work", IFS Ltd., 1983.

[Harwin 86] Harwin W.S., "Critical Appraisal of World Research on Robotic Aids for the Physically Disabled", Cambridge University, Engineering Department Report, pp 1-9, Cambridge, 1986.

[Hillamn 87/a] Hillamn M.R., "Robotics for the Handicapped in the US", Bath Inst. of Medical Engineering Reprot, pp 1-30, Bath, 1987.

[Hillman 87/b] Hillman M.R., "A Feasibility Study of a Robot Manipulator for the Disabled", Jl. Medical Eng. and Technology. pp 160-165, 1987.

[Ho 94] Ho S.C., Davies B.L., Hibberd R.D., Cobb J.P., "Robot Knee Surgery - Implicit Force Control Strategy with Active Motion Constraint", vol. 3, pp. 1235-1248, Proc. of European Robotics and Intelligent Systems Conference, Malaga, Spain, August 1994.

[Ho 95/a] Ho S.C., Hibberd R.D., Davies B.L., "A Force Control Strategy with Active Motion Constraint for Robot Assisted Knee Surgery", Journal of IEEE Engineering in Medicine & Biology Society, May, 1995.

[Ho 95/b] Ho S.C., Davies B.L., Hibberd R.D., Cobb J.P., "Force Control for Robotic Surgery", Proc. Int. Conf. on Advanced Robotics, Sept., 1995 (to be published).

[Hogan 85/a] Hogan N., " Impedance Control : An Approach to Manipulation : Part I, II, & III, " pp. 1-24, Vol. 107, Journal of Dynamic Systems, Measurement, and Control, March 1985.

[Hogan 85/b] Hogan N., "Impedance Control : An Approach to Manipulator," Journal of Dynamic Systems, Measurement, and Control, pp 1-24, Vol. 107, March, 1985.

[Hogan 87] Hogan N., "Stable Execution of Contact Tasks Using Impedance Control," pp. 1047-1053, pp. 897-903, Proc. IEEE Conf. of Robotics and Automation, 1987.

[Hunt 83] Hunt V.D., "Industrial Robotics Handbook", Industrial Press Inc., 1983.

[Jones 88] Jones T., "Out of the Lab and into the Home", Proc. of Joint Coordinating Forum for Int. Advanced Roibotics Programme, National Research Council of Canada, pp 8.1-8.2, Ottawa, Ontario, 1988.

[Kazanzides 92] Kazanzides P., Zuhars J., Mittelstadt B., Taylor R.H., "Force Sensing and Control for a Surgical Robot," pp 612-617, Proc. of IEEE Conf. on Robotics and Automation, Nice, France, May 1992.

[Kazerooni 88] Kazerooni H. and Tsai T., "Stability Criteria for Compliant Maneuvers," pp. 1166-1172, Proc. IEEE Conf. of Robotics and Automation, 1988.

[Kazerooni 89] Kazerooni H., "On the Robot Compliant Motion Control," pp. 416-425, Vol. 111, Trans. ASME, September, 1989.

[Kerr 90] Kerr D.R., "Patient Aid to Mobility (PAM)", IMechE seminar: Robotics in Medicine, 14 June 1990.

[Khatib 85] Khatib O., "The Operational Space Formulation in Robot Manipulator Control," pp 11-13, 15th ISIR, Tokyo, September, 1985

[Kienzle 93] Kienzle T.C., Stulberg S.D., Peshkin M., Quaid A., Wu C.H., "An Integrated CAD-Robotics System for Total Knee Replacement Surgery," Proceedings of the 1993 IEEE Int. Conf. on Robotics and Automation, pp. 889-894, Vol. 1, Atlanta, GA

[Koenigsberger 61] Koenigsberger F., Sabberwal A.J.P., "An Investigation into the Cutting Force Pulsations During Milling Operations," pp. 15-33, Vol. 1, Int. J. Mach. Tool Des. Res., 1961.

[Kohn 91] Kohn A.F., Furuie S.S., "Safety in Medical Signal Analysis", IEEE EMBS Mag., pp 56-62, Dec. 1991.

[Korba] Korba L.W., "International Inventory of Robotics Projects in the Healthcare Field", National Research Council of Canada Report, NRCC, No 30379.

[Koren 85] Koren Yoram, "Robotics for Engineers", McGraw-Hill Book Company, 1985.

[Kreyszig] Kreyszig E., "Advanced Engineering Mathematics," John Wiley & Sons.

[kuo 80] Kuo B.C., "Digital Control Systems, " Holt, Rinehart and Winston, Inc., 1980

[Kuo 82] Kuo B.C., "Automatic Control Systems, " Prentice-Hall Inc. , 1982

[Kurosawa 85] Kurosawa H., Walker P.S., et. al., "Geometry and Motion of the Knee for Implant and Orthotic Design", Jl. Biomechanics, 1985.

[Kwoh 85] Kwoh Y.S., et al., "A new Computerized Tomographic Aided Robotic Stereotaxis System", Robotic Age, June 1985.

[Kwoh 88] Kwoh Y.S., Hou J., Jonckheere E.A., Hayati S., "A Robot with Improved Absolute Positioning Accuracy for CT Guided Stereotactic Brain Surgery," pp.153-160, Vol. 35, No. 2, IEEE Trans. on Biomedical Engineering, Feb. 1988.

[Landau 90] Landau I.D., "System Identification and Control Design : Using P.I.M + Software, " Prentice Hall Inc., 1990.

[Lavallee 89] Lavallee S., "A New System for Computer Assisted Neuro-Surgery", IEEE Eng. in Med. & Biology Soc., 11th Int. conf. pp 926-927, 1989.

[Lavallee 91] Lavallee S., Brunie L., Mazier B., Cinquin P., "Matching of Medical Images for Computer and Robot Assisted Surgery", Proc. of IEEE EMBS Conf., Orlando, Florida, Nov. 1991.

[Lavallee 92] Lavallee, S., et. al., "Geometrical Methods of Multi-Modality Images Registration for Computer Integrated Surgery", Proc. first Int. Med. Mechatronics Workshop, pp. 127-138, Malaga, Spain, Oct. 1992.

[Lea 94] Lea J.T., Mills A., Watkins D., Peshkin M., Kienzle T.C., Stulberg S.D., "Registration and Immobilization for Robot Assisted Orthopaedic Surgery", pp 63-68, vol. 2, Proc. First Symp. on Robotic Surgery, Pittsburgh, USA, Sept. 1994.

[Leifer 78] Leifer L., "Robotic Aids for the Severely Disabled: Feasibility Assessment", Advances in External Control of Human Extremities Dubrovnik, Yugoslavia (ETAN), August 1978.

[Leifer 81] Leifer L., "Rehabilitative Robotics", Robotics Age, pp 4-15, 1981.

[Lhote 84] Lhote F., Kauffmann J., Andre' P., Taillard J., "Robot Components and Systems," Anchor Brendon Ltd. 1984

[Maples 86] Maples J.A., Becker J.J., " Experiments in Force Control of Robotic Manipulators," pp. 1381-1386, Proc. IEEE Conf. on Robotics and Automation, 1986.

[Mason 81] Mason M., "Compliance and Force Control for Computer Controlled manipulators," IEEE Transactions on Systems, Man and Cybernetics, pp418-432, 1981.

[Matsen III 93] Matsen III F.A., Garbini J.L., Sidles J.A., Pratt B., Baumgarten D., Kaiura R., "Robotic Assistance in Orthopaedic Surgery - A Proof of Principle Using Distal Femoral Arthroplasty," pp. 178-186, No. 296, Clinical Orthopaedics and Related Research, 1993.

[McCloy] McCloy D., Harris M., "Robotics: An Introduction," Open University Press, 1986.

[McEwen 84] McEwen J.A., "Medical and Surgical Robotics", pp 11-12, Proc. of 10th Canadian Edical and Biological Engineering Conf of the Future of Health, 1984.

[McEwen 89] McEwen J.A., et al., "Development and Initial Clinical Evalustion of Pre-robotic and Robotic Surgical Retraction Systems", pp 881-882, IEEE EMBS 11th Annual Int'l Conf., 1989.

[Minns 94] Minns R.J., "Methods of Cutting in Orthopaedic Surgery - Development and Pitfalls," meeting reports, vol. 9, no. 2, Theoretical Surgery, 1994

[Mittelstadt 93] Mittelstadt B., Kazanides P., Zuhers J., Cain P., Williamson B., "Robotic Surgery: Achieving Predictable Results in an Unpredictable Environment", pp. 367-372, Proc. of 6th Conf. on Advanced Robotics, Tokyo, Nov. 1993.

[Ng 92] Ng W.S., "Development of a Surgeon Assistant Robot for Prostatectomy", PhD thesis, Imperial College, Univ. of London, Nov. 1992.

[Ng 93] Ng W.S., Davies B.L., Hibberd R.D., Timoney A.G., "Robotic Surgery : A First-Hand Experience in Transurethral Resection of the Prostate, " pp. 120-125, IEEE Engineering in Medicine and Biology, March 1993.

[Nicholls 89] Nicholls H.R., Lee M.H., "A survey of Robot Tactile Sensing Technology," pp. 3-30, Vol. 8, No. 3, The International Journal of Robotics Research, June, 1989

[Norton 86] Norton J.P., "An Introduction to Identification," Academic Press Inc., 1986.

[Owens 82] Owens G.D., Ekstein R.J., "Robotic Sample Preparation Station", Anal. Chem. 54, pp 2347-2451, 1982.

[Patarinski 93] Patarinski S.P., Botev R.G., "Robot Force Control : A Review," Mechatronics Vol. 3, No. 4, pp 377-398, 1993.

[Paul 81] Paul R.P., "Robot Manipulators : Mathematics, Programming, and Control, " MIT press, 1981

[Paulat 92] Paulat B.M, "Fundamentals of Industrial Ergonomics," Prentice Hall, 1992

[Payandeh 91] Payandeh S., Goldengerg A.A., " A Robust Force Controller : Theory and Experiments," pp. 36-41, Proc. IEEE Int. Conf. on Robotics and Automation, California, 1991.

[Pellerin 93] Pellerin C., "The Market for Robotics Services," pp. 12-14, Vol. 20, No. 2, Industrial Robot, 1993.

[Preising 91] Preising B., Hsia T.C., Mittelstadt B., "A literature Review : Robots in Medicine," pp. 13-22, IEEE Engineering in Medicine and Biology, June 1991.

[Qian 92] Qian H.P., Schutter J.De., "The Role of Damping and Low Pass Filtering in the Stability of Discrete Time Implemented Robot Force Control," pp.1368-1373, Proc. IEEE Conf. of Robotics and Automation, France, May 1992

[Quittel 85] Quittel F., "Future Computer Technology in Healthcare", Computers in Healthcare, pp 24-28, Nov. 1985.

[Raibert 81] Raibert M.H., Craig J.J., "Hybrid Postion/Force Control of Manipulators," pp. 126-133, Vol. 102, Tans. of ASME, June 1981.

[Rao 86] Rao S.S., " Mechanical Vibration", Addison Wesley Inc., 1986

[Roberts 85] Roberts R.K., Paul R.P., Hillberry B.M., "The Effect of Wrist Force Sensor Stiffness on the Control of Robot Manipulators," pp. 269-274, Proc. IEEE Conf. on Robotics and Automation, 1985.

[Robot 85] "Handbook of Industrial Robotics", John Wiley & Sons, 1985.

[Sabberwal 62] Sabberwal A.J.P., "Cutting Forces in Down Milling," pp27-41, Vol. 2, Int. J. Mach. Tool Des. Res., 1962.

[Sailsbury 80] Salisbury J.K., "Active Stiffness Control of a Manipulator in Cartesian Coordinates," 19th IEEE conference on Decision and Control, Dec. , 1980.

[Salculdean 94] Salculdean S.E., Yan J., "Design and Control of a Motion Scaling System for Microsurgery Experience", Proc. First Symp. on Robotic Surgery, pp 211-217, vol. 1, Pittsburgh, USA, Sept. 1994.

[Sandeman 92] Sandeman D., Marshall C., Bret P., "Medical Robotics in Neurosurgery: The Potential of 3D Image Guidance Systems of Intelligent Micro Manipulators", Proc. first Int. Workshop on Mechatronics in Med. and Surgery. pp. 153-157, Costa Del Sol, Spain, Oct. 1992.

[Sautot 92] Sautot P., Cinquin P., Lavallee S., Troccaz J., "Computer Assisted Spine Surgery: A first Step Towards Clinical Application in Orthopaedics", pp 1071-1072, Proc. of IEEE EMBS Conf., Paris, France, Oct. 1992.

[Schneider 81] Schneider W., Schmeisser G., Seamone W., "A Micro-processor Controlled Robotic Arm/Worktable System for the High-Level Guadriplegic," pp. 41-47, IEEE Computer, January, 1981.

[Schuler 86] Schuler C.A., McNamee W.L., "Industrial Electronics and Robotics," McGraw-Hill Inc, 1986.

[Schutter 88/a] Schutter J., Brussel H., "Compliant Robot Motion. II. A Control Appraoch Based on External Control Loops," Vol. 7, No. 4, International Journal of Robotics Research, August 1988.

[Schutter 88/b] Schutter J., Brussel H., "Compliant Robot Motion. I. A Formalism for Specifying Compliant Motion Tasks," Vol. 7, No. 4, International Journal of Robotics Research, August 1988.

[Seippel 83] Seippel R.G., "Transducer, Sensors, and Detectors", Prentice-Hall Company, 1983.

[Severns 84] Severns M.L., Hawk G.L., "Medical Laboratory Automation Using Robotics", pp 633-643, Robotics and Artificial Intelligence, Springer-Verlag, Berlin, 1984.

[Shao 85] Shao H.M. et. al., "A New CT-Aided Stereotactic Neurosurgery Technique", IEEE Transc. on Biomedical Engineering, pp 540-544, Jul. 1985.

[Snyder 85] Snyder W.E., "Industrial Robots: Computer Interfacing and Control", Prentice Hall International, New Jersey, 1985.

[Spong 89] Spong M.W., Vidyasagar M., "Robot Dyanmics and Control," John Wiley & Sons, 1989.

[Swanson 73] Swanson S.A.V., Freeman M.A.R., Heath J.C., "Laboratory Tests on Total Joint Replacement Prostheses", Jouranl of Bone and Joint Surgery, 55B, pp 759-773, 1973.

[Takatoshi] Takatoshi Ide, Siddigi N.A., Noriya A., "Expectations for Medical and Health Care Robotics," AR, Feb. 1991.

[Takeyoshi 91] Takeyoshi Dohi, "Robot technology in medicine", AR, March 1991.

[Taylor 89] Taylor R.H., et al., "Robotic Hip Replacement Surgery in Dogs", pp 887-889, IEEE EMBS Int'l Conf., 1989.

[Taylor 91/a] Taylor R.H., Paul H.A., Kazanzides P., Mittelstadt B.D., Hanson W., Zuhars J. Williamson B., Musits B., Glassman E., Bargar W.L., "Taming the Bull: Safety in a Precise Surgical Robot," pp. 865-870, Proc. of 5th Int. Conf. on Advanced Robotics, Pisa 1991.

[Taylor 91/b] Taylor R.H., Funda J., LaRose D., "A Telerobotic System for Augmentation of Endoscopic Surgery", Proc. of 5th Int. Conf. on Advanced Robotics, Pisa, Italy, June 1991.

[Taylor 92] Taylor R.H., Paul H.A., Cutting C.B., Mittelstadt B., Hanson W., Kazanzides P., Musits B., Kim Y.Y., Kalvin A., Haddad B., Khoramabadi D., and Larose D., "Augmentation of Human Precision in Computer-Integrated Surgery," Innovation and Technology in Biology and Medicine, pp 450-468, Vol. 13, No. 4, 1992.

[TIDE 93] TIDE Project No. 128, "A General Purpose Multiple Master Multiple Slave Intelligent Interface for the Rehabilitation Environment", ECE DG XIII/C3, Brussell 1993.

[Townsend 87] Townsend W.T., Salisbury J.K., "The Effect of Coulomb Friction and Stiction on Force Control," pp. 883-889, Proc. IEEE Conf. of Robotics and Automation, 1987.

[Troccaz 93] Troccaz J., Lavallee S., Hellion E., "PADYC : A Passive Arm with Dynamic Constraints, " pp. 361-366, ICAR, 1993.

[Tuchi 85] Tuchi S., Tanie K., Komoriya k., Abe M., "Electrocutaneous Communication in a Guide DOG ROBOT (MELDOG)," pp. 461-469, Vol. 132, No. 7, IEEE Transactions on Biomedical Engineering, July, 1985.

[Uchiyama 91] Uchiyama M., Nakamura Y., Hakomori K., "Evaluation of the Robot Force Sensor Structure Using Singular Value Decomposition," pp.39-52, Vol. 5, No. 1, Advanced Robotics, 1991.

[Van Brussel 85] Van Brussel H., Belien H., Thielemans H., "Force Sensing for Advanced Robot Control," Proc. 5th Int. Conf. on Robot Vision and Sensory Control, Amsterdam, pp 59-68, 1985

[Van Woerden 94] Van Woerden J.A., Jackson R.D., Davies B.L., Hibberd R.D., "M3S - A Standard Communication Architecture for Rehabilitation Applications", Computing and Control Eng. Jl., Vol 5, No. 5, pp 213-218, IEE Press, Oct. 1994.

[Vidyasagar 78] Vidyasagar M., "Nonlinear Systems Analysis", Prentice-Hall Inc, 1978.

[Vukobratovic] Vukobratovic M., Surdilovic D., Anton S., "Contact Stability Issues in Impedance Control: Theory and Experimental Results," pp. 635-642, 24th ISIR

[Waibel 91] Waibel B.J., Kazerooni H., "Theory and Experiments on the Stability of Robot Compliance Control," pp. 95-104, Vol. 7, No. !, IEEE Transactions on Robotics and Automation, Feb. 1991.

[Walker 88] Walker P.S., Rovick J.S., Robertson D.D., "The Effects of Knee Brace Hinge Design and Placement on Joint Mechanics", J. Biomechnaics, Vol. 21, No. 11, pp 965-974, 1988.

[Walker 93] Walker P.S., "The Scope for Robotics in Total Knee Surgery," Seminar for Robotics in Keyhole Surgery, ImechE, London, 5 May 1993

[Wen 91] Wen J.T., Murphy S., "Stability Analysis of Position and Force Control for Robot Arms," pp. 365-370, Vol. 36, No. 3, IEEE Trans. on Automatic Control, March 1991.

[Whitney 77] Whitney D.E., "Force Feedback Control of Manipulator Fine Motions," pp. 91-97, J. of Dyanmic Systems, Measurement, and Control, June 1977.

[Whitney 87] Whitney D.E., "Historical Perspective and State of the Art in Robot Force Control," The international Journal of Robotics Research, Vol. 6, No. 1, Spring 1987.

[Woodburne 88] Woodburne R.T., Burkel W.E., "Essentials of Human Anatomy," Oxford University Press, 1988

[Wu 85] Wu C.H., "Compliance Control of a Robot Manipulator Based on Joint Torque Servo," pp. 55-71, Vol. 4, No. 3, Int. J. of Robotics Research, Fall 1985.

[Wu 92] Wu C.H., Papaiannou J., Kienzle T.C., Stulberg S.D., "A CAD-Based Human Interface for Pre-Operative Planning of Total Knee Replacement Surgery", Proc. of IEEE Int. Conf. on Systems Man & Cybernetics, Chicago USA, Oct. 1992.

[Zamorano 93] Zamorano L., Nolte L., Jiang C., Kadi M., "Image-Guided Neurosurgery : Frame-Based and Frameless Approaches," pp 403-422, Vol 3, No. 6, Neurosurgical Operative Atlas, 1993.

Technique Reference:

1. "Unicompartmental Replacement Principles and Surgical Technique," Biomet Ltd.

2. "P.F.C Modular Knee System with Specialist Instruments - Surgical Technique," Johnson & Johnson Orthopaedic.

3. "Kinemax Modular Total Knee System - Instrument Surgical Technique," Howmedica Inc.

4. "Optoelectronics: Designer's Catalog," Hewlett Packard Co.

APPENDIX A

Wheatstone-bridge Circuit and Principle of the Force Sensor

Fig. A.1 shows the diagram of the Wheatstone-bridge circuit. V_o is the bridge power supply. R1, R2, R3, and R4 are the resistance of the strain gauges and ΔV is the voltage output due to the external force. The voltage output of the bridge circuit in Fig.A.1 can be expressed by the representation

$$V = V_o\left(\frac{R_4}{R_1 + R_4} - \frac{R_3}{R_2 + R_3}\right). \qquad \text{(A.1)}$$

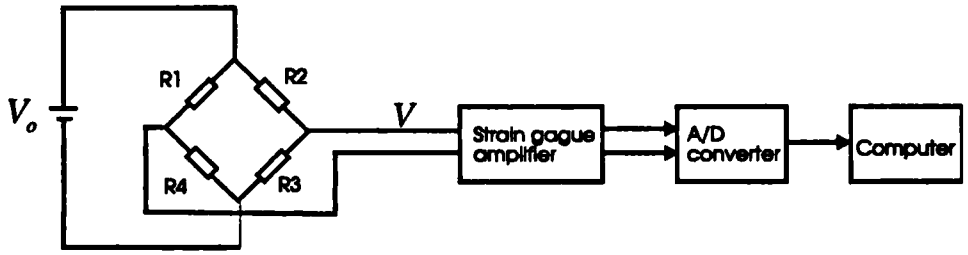


Fig. A.1 The Wheatstone-bridge circuit

If the resistance of all the strain gauges is the same, V should be equal to zero. When the resistance of the strain gauges is slightly changed because of the force applied, the variation of the voltage output ΔV can be derived by taking the differentiation of equation (A.1) and equals

$$\Delta V = V_o\left[\frac{R_4\Delta R_1 - R_1\Delta R_4}{(R_1 + R_4)^2} - \frac{R_3\Delta R_2 - R_2\Delta R_3}{(R_2 + R_3)^2}\right], \qquad \text{(A.2)}$$

where ΔR_i are variations of resistance of the strain gauges. If the strain gauges are geometrically arranged as shown in Fig. 7.6 or Fig. 7.7, the change of the resistance will have the relationship

$$\Delta R = \Delta R_1 = \Delta R_3 = -\Delta R_2 = -\Delta R_4, \qquad \text{(A.3)}$$

where ΔR is the change of the nominal resistance of the strain gauges. Substituting (A.3) into (A.2) yields

$$\begin{aligned}\Delta V &= V_o[\frac{2R(\Delta R)}{4R^2}+\frac{2R(\Delta R)}{4R^2}] \\ &= V_o\frac{\Delta R}{R}.\end{aligned} \qquad \text{(A.4)}$$

If the nominal length of the strain gauge is equal to L, the resistance change can be expressed by

$$\begin{aligned}\frac{\Delta R}{R} &= f\frac{\Delta L}{L} \\ &= f\varepsilon,\end{aligned} \qquad \text{(A.5)}$$

where f is the stain gauge factor, ΔL is the length change of the strain gauge and ε is the strain due to the external force. Combining (A.4) and (A.5) yields

$$\begin{aligned}\Delta V &= V_o\frac{\Delta R}{R} \\ &= V_o f\frac{\Delta L}{L} \\ &= V_o f\varepsilon.\end{aligned} \qquad \text{(A.6)}$$

If the force sensor is fixed to the tip of the robot, the sensor is like a cantilever beam (see Fig. A.2). The relationship between the normal external force and the strain can be represented by [Gere 84]

$$\varepsilon = \frac{My}{EI}, \qquad \text{(A.7)}$$

where M is the bending moment due to the external force; y is the distance from the neutral surface to the outer beam surface; E is the beam's Young's modulus and I is the moment of inertia of the cross-sectional area with respect to the neutral axis.

In the case of the two-axis force sensor, the beam has a hollow circular cross section. Therefore, the moment of inertia is given by

$$I = \frac{\pi(d_o^4 - d_i^4)}{64}, \qquad \text{(A.8)}$$

where d_o and d_i are the outer and inner diameters of the hollow circular beam, respectively.

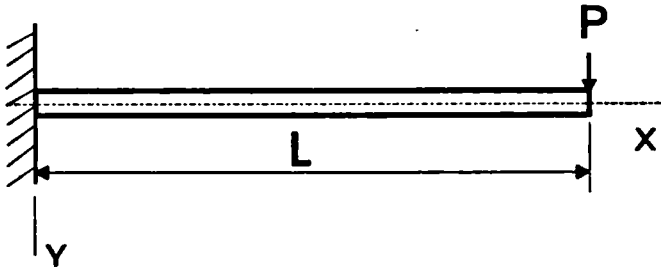


Fig. A.2 Bending of a cantilever beam

Substituting (A.8) into (A.7), the relationship between the strain and external force is given as

$$\begin{aligned} \varepsilon &= \frac{My}{EI} \\ &= \frac{64PLy}{E\pi(d_o^4 - d_i^4)} \\ &= \frac{32PLd_o}{E\pi(d_o^4 - d_i^4)}, \end{aligned} \qquad \text{(A.9)}$$

where P is the normal external force and L is distance from the external force to the position of the strain gauges. Combining (A.9) with (A.6), the voltage output of the Wheatstone bridge circuit with respect to the external force is given by

$$\begin{aligned} \Delta V &= V_o f \varepsilon \\ &= \frac{32V_o fLd_o}{E\pi(d_o^4 - d_i^4)} P, \end{aligned} \qquad \text{(A.10)}$$

As shown in Fig. 7.6, the two strain gauge bridges are bonded to the aluminium shaft in an orthogonal arrangement. Therefore, forces measured by the two bridge circuits will be orthogonal to each other. As a result, force signals can be easily decoupled into forces in X and Y axes.

For the one-dimensional force sensor, the beam has a rectangular cross section. Thus, the moment of inertia is equal to

$$I = \frac{wh^3}{12}, \qquad \text{(A.11)}$$

where w and h are the width and height of the cross section respectively. The voltage output of the bridge circuit with respect to an external force in z-axis is therefore equal to

$$\begin{aligned} \Delta V &= V_o f \varepsilon \\ &= V_o f \frac{12PLy}{Ewh^3} \qquad \text{(A.12)} \\ &= \frac{6V_o fL}{Ewh^2} P, \end{aligned}$$

In summary, depending on the range of input and desired sensitivity, the required mechanism of the force sensors can be determined by equation (A.10) and (A.12).

APPENDIX B

Calibration of the Force Sensors

(I). The two-axis force sensor

Based on equation (A.10), the output of the force sensor can be represented by a linear model

$$\begin{bmatrix} F_x \\ F_y \end{bmatrix} = \begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix} \begin{bmatrix} V_x \\ V_y \end{bmatrix} + \begin{bmatrix} v_x \\ v_y \end{bmatrix}. \qquad \text{(B.1)}$$

F_x and F_y are the respective external force in X and Y axes, and V_x and V_y are the voltage signals read from the A/D converter; k_{ij} are constant parameters of the force sensor and v_x and v_y are offset errors. These parameters k_{ij}, v_x and v_y need to be calibrated. Equation (B.1) can be written as

$$\begin{aligned} \begin{bmatrix} V_x \\ V_y \end{bmatrix} &= \begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix}^{-1} \begin{bmatrix} F_x \\ F_y \end{bmatrix} - \begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix}^{-1} \begin{bmatrix} v_x \\ v_y \end{bmatrix} \\ &= \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} F_x \\ F_y \end{bmatrix} - \begin{bmatrix} v_x' \\ v_y' \end{bmatrix}. \end{aligned} \qquad \text{(B.2)}$$

When F_y is equal to zero, (B. 2) becomes

$$\begin{bmatrix} V_x \\ V_y \end{bmatrix} = \begin{bmatrix} a_{11} \\ a_{21} \end{bmatrix} F_x - \begin{bmatrix} v_x' \\ v_y' \end{bmatrix}. \qquad \text{(B.3)}$$

By giving F_x and recording the corresponding outputs V_x and V_y, the parameters a_{11}, a_{21}, v_x and v_y can then be derived from the experimental results by using least-square analysis. The parameters a_{12} and a_{22} can be found in a similar way to those for a_{11} and a_{21}.

The foil strain gauges used are 2 mm long, whose nominal resistance is 120Ω and gauge factor is 2. The voltage supply of the strain gauge bridges is 12V. Output of the bridge circuit is processed through a low-pass filter to reduce the noise and then boosted 1000 times by an amplifier. The output of the strain gauge is read into the computer via a 12-bit analogue-to-digital converter, whose input limit ranges from -

5V to +5V. Table B.1 and Fig. B.1 show the calibration results of the two-axis force sensor. By using the least-square approximation, the parameters of the force sensor are given by

$$\begin{bmatrix} F_x \\ F_y \end{bmatrix} = \begin{bmatrix} 1.781 & -0.046 \\ -0.089 & 1.750 \end{bmatrix} \begin{bmatrix} V_x \\ V_y \end{bmatrix} + \begin{bmatrix} -0.56 \\ 0.67 \end{bmatrix} \qquad \text{(B.4)}$$

(II). The z-axis force sensor

The mathematical model for the z-axis force sensor can be represented by

$$F_z = K_z V_z + v_z, \qquad \text{(B.5)}$$

where F_z is the external force; V_z is the output of A/D; K_z is the constant factor and v_z is the offset error. Table B.2 and Fig. B.2 show the calibration results. Analyzing the experimental results by using the least-square method, the model of the z-axis force sensor is given as

$$F_z = 0.904 V_z - 3.014. \qquad \text{(B.6)}$$

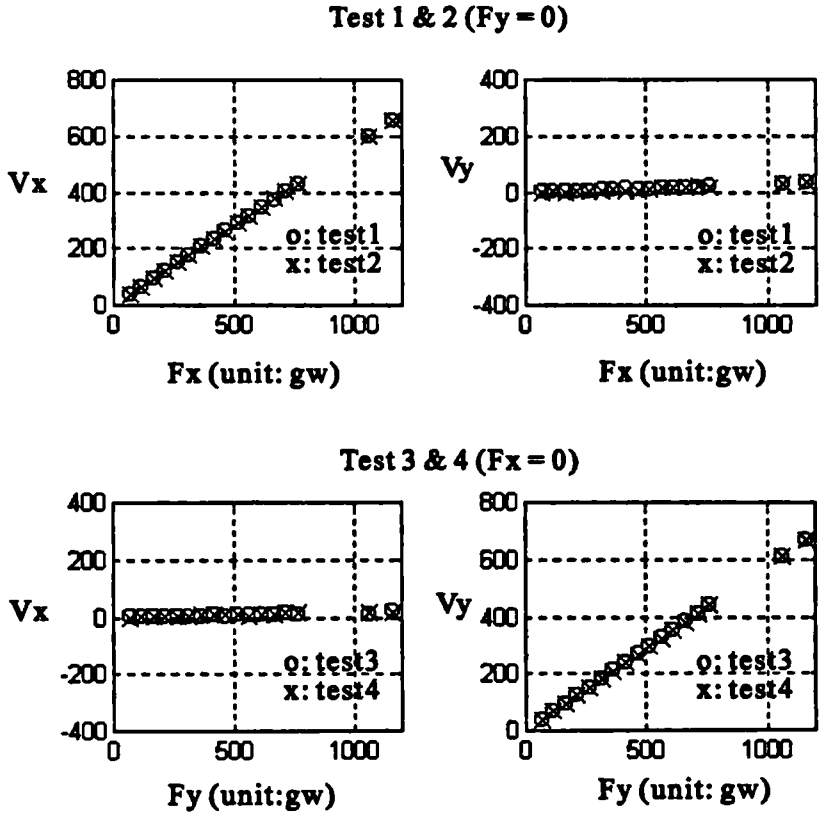


Fig. B.1 Calibration results of the two-axis force sensor

Calibrated Forces (unit : gw)		Data from ADC (Test 1)		Data from ADC (Test 2)	
F_x	F_y	V_x	V_y	V_x	V_y
65	0	38	1	37	1
115	0	66	3	65	1
165	0	94	4	94	0
215	0	122	6	121	4
265	0	149	4	149	5
315	0	178	8	177	4
365	0	206	9	206	9
415	0	234	16	234	4
465	0	262	10	261	6
515	0	290	12	290	11
565	0	318	14	318	13
615	0	346	14	346	14
665	0	374	17	375	15
715	0	402	16	403	19
765	0	431	20	431	16
1065	0	598	29	600	26
1165	0	655	35	658	32

Calibrated Forces (unit : gw)		Data from ADC (Test 3)		Data from ADC (Test 4)	
F_x	F_y	V_x	V_y	V_x	V_y
0	65	1	39	0	39
0	115	2	68	2	67
0	165	2	97	2	95
0	215	4	126	1	123
0	265	2	154	2	152
0	315	3	183	4	180
0	365	6	212	8	210
0	415	7	241	9	237
0	465	4	269	10	266
0	515	7	297	10	294
0	565	8	326	3	322
0	615	10	355	12	352
0	665	8	384	10	381
0	715	13	412	14	410
0	765	9	441	17	439
0	1065	9	610	14	609
0	1165	19	669	17	666

Table B.1. Calibration data of the two-axis force sensor

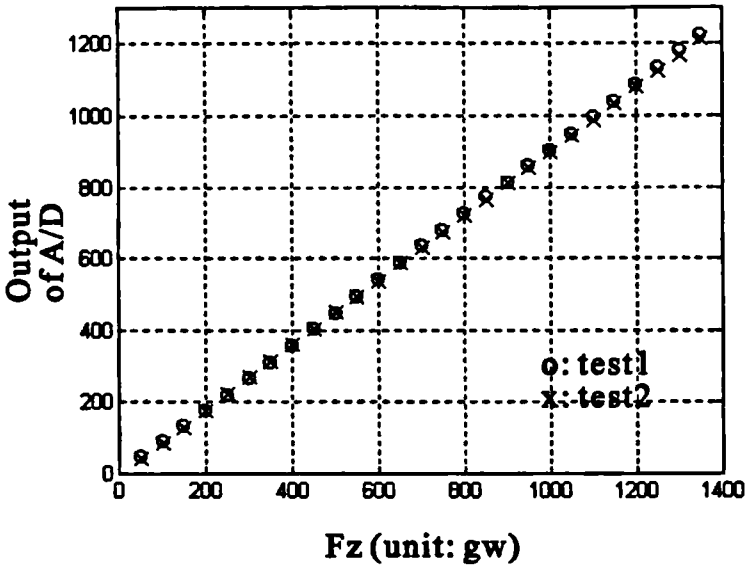


Fig. B.2 Calibration results of the Z-axis force sensor

Force of F_z (unit : gw)	Output of A/D (Test 1)	Output of /D (Test 2)
50	46	44
100	88	86
150	132	130
200	175	178
250	219	225
300	266	269
350	310	315
400	357	362
450	402	403
500	448	449
550	494	495
600	540	538
650	586	588
700	634	630
750	678	674
800	724	720
850	773	765
900	813	809
950	858	854
1000	903	895
1050	950	942
1100	994	988
1150	1037	1033
1200	1087	1080
1250	1132	1122
1300	178	1167
1350	1221	1213

Table B.2 Calibration data of the z-axis force sensor

APPENDIX C : AUTHOR'S PUBLICATIONS (in date order)

1. Davies B.L., Hibberd R.D., Ho S.C., Cobb J.P.: A Novel Robotic System for Knee Surgery. Seminar: Future Technologies and Surgery. B.E.S Durham. April, 1994.

2. Ho S.C., Davies B.L., Hibberd R.D., Cobb J.P., "Robot Knee Surgery - Implicit Force Control Strategy with Active Motion Constraint", vol. 3, pp. 1235-1248, Proc. of European Robotics and Intelligent Systems Conference, Malaga, Spain, August 1994.

3. Davies B.L., Hibberd R.D., Ho S.C., "The Use of Force Control in Robot Assisted Knee Surgery", vol. 1., pp. 258-263, Proc. of First International Symposium on Medical Robotics and Computer Assisted Surgery, Pittsburgh, Pennsylvania, September, 1994.

4. Ho S.C., "Force Control Strategy for Robotic Surgery", Proc. of First Symposium of the Chinese Institute of Engineers in UK, April, 1995.

5. Ho S.C., Hibberd R.D., Davies B.L., "A Force Control Strategy with Active Motion Constraint for Robot Assisted Knee Surgery", Vol. 14, No. 3, pp. 292-300, IEEE Engineering in Medicine & Biology Magazine, May, 1995.

6. Ho S.C., Davies B.L., Hibberd R.D., Cobb J.P., "Force Control for Robotic Surgery", Proc. Int. Conf. on Advanced Robotics, Spain, Sept., 1995 (to be published).

7. Ho S.C., Davies B.L., Hibberd R.D., Cobb J.P., "Active Force Control as a Dynamic Constraint in Robotic Knee Surgery", Proc. of the Second Int. Workshop: Mechatronics in Medicine & Surgery, Bristol, September 1995 (to be published).

Appendix D Nomenclature

M_{ij} (i^{th} row , j^{th} column) element of M(θ) matrix

L_a armature inductance

R_a armature resistance

K_{F2} compliance ($stiffness^{-1}$) matrix with dimensions position/force

Δt computation cycle

E_a control voltage of the digital-to-analogue converter

τ_f Coulomb friction

c_f Coulomb friction constant

v_2 cutting force factor

K_{F1} desired admittance

$\ddot{X}d$ desired Cartesian acceleration of the robot's end-effector

Xd desired Cartesian position vector of the robot's end-effector

$\dot{X}d$ desired Cartesian velocity vector of the robot's end-effector

F_d desired force

$\ddot{\theta}_d$ desired joint acceleration vector

θ_d desired joint position vector

$\dot{\theta}_d$ desired joint velocity vector

$c_x(\theta)$ gravity force vector in Cartesian space

$M_x(\theta)$ inertia matrix in Cartesian space

$J^{-1}(\theta)$ inverse of the Jacobian matrix

K_θ joint stiffness matrix

L_i length of the links of the manipulator

X_e location of the environment

K_t motor torque constant

ω_n natural frequency

i_a output of the current amplifier

ΔX position error vector

ω_r resonance frequency

T_s sampling interval

T_{s2} sampling interval of an inner control law, which is shorter than T_{s1}

T_{s1} sampling interval of the 'interpreter' program

$sgn(\dot{\theta})$ sign of the velocity $\dot{\theta}$

τ_s static friction

K_E stiffness of the environment

J^T transpose of the Jacobian matrix

$h_x(\theta,\dot{\theta})$ velocity-related force vector in Cartesian space

v_i viscous force constant

$K_{d\theta}$ joint damping matrix

$\ddot{\theta}_i$ acceleration of joint i

ζ system damping ratio

τ torque command vector for joint actuators

ADC (A/D) analogue-to-digital converter

AR Advanced Robotics

B damping constant

$c(\theta)$ n×1 gravity force vector

CAS Computer Aided Systems

c_i i^{th} element of $c(\theta)$ vector

COFF Common Object File Format

CT Computed Tomography

DAC (D/A) digital-to-analogue converter

DC direct current

DOF degree of freedom

DPM dual ported memory

DSP digital signal processor

DTI Department of Trade and Industry

F force vector acting on the end-effector of the robot

F_a interaction force on the environment

Fc milling tool cutting force

F_{dist} disturbance force and is equal to $(\sum_{i \neq j} M_{ij}\ddot{\theta}_j + h_i(\theta,\dot{\theta}) + c_i(\theta))$

Fr output force of the actuator of the robot

Fs pulling/pushing force of the surgeon

$h(\theta,\dot{\theta})$ n×1 nonlinear centrifugal and Coriolis force vector

h_i i^{th} element of $h(\theta,\dot{\theta})$ vector

I.C. Imperial College

J inertia

J Jacobian matrix

Kc cutting force ratio with the dimension force/velocity

Kd velocity feedback gain matrix

Kf force feedback gain

Ki integral feedback gain matrix

Kp proportional feedback gain matrix

M mass

$M(\theta)$	n×n inertial matrix of the manipulator
MRI	Magnetic Resonance Imaging
N	newton
PAM	Patient Aid to Mobility
PD	proportional and derivative feedback controller
PET	Positron Emission Tomography
PID	proportional, integral and derivative feedback controller
PWM	pulse-width modulation
RAND	random number
T	torque output of the motor, respectively
TKR	Total Knee Replacement
TRC	Transitions Research Corp.
X	Cartesian position vector of the robot
Z	impedance