Improvement of a

Hexapod Parallel Kinematic Machine Tool

IMPROVED KINEMATIC MODELING & MOTION CONTROL OF THE TIARA HEXAPOD PARALLEL KINEMATIC MACHINE TOOL

By

TAIYONG GUO, B.ENG. M.ENG.

A Thesis
Submitted to the School of Graduates Studies
In Partial Fulfilment of the Requirements
for the Degree
Master of Applied Science

McMaster University

MASTER OF APPLIED SCIENCE (2004)	McMaster University
(Mechanical Engineering)	Hamilton, Ontario

TITLE: Improved Kinematic Modeling & Motion Control of the Tiara Hexapod Parallel Kinematic Machine Tool

AUTHOR: Taiyong Guo
B.Eng. M.Eng. (Harbin Institute of Technology)

SUPERVISORS: Dr. G. Bone, Dr. M.A. Elbestawi
Department of Mechanical Engineering
McMaster University

NUMBEROF PAGES: xvi, 175

Abstract

A Hexapod is a novel generation machine tool based on parallel kinematic structures. Due to its potential capabilities for high dimensional accuracy, high structural rigidity and high-speed capacity, it is regarded as the most radical change in the design of machine tools in the last century. However, due to its short development history and lack of wide industrial application, it poses both theoretical and technical challenges. Solutions to overcome its obvious weakness are not yet available. Design tools, commercial motion controllers capable of dealing with non-linear dynamics and coupling effects, and maintenance topics related to measurement, calibration and accuracy evaluation are still open issues.

A hexapod, termed the Tiara hexapod, is being developed at the McMaster Manufacturing Research Institute (MMRI) at the McMaster University. The Tiara hexapod has fixed length struts and movable base joints. It is designed to be a milling machine. This thesis presents improvements of the kinematic model and servo control system, the development and implementation of the motion interpolator, as well as preliminary measurements of this machine.

Following a comprehensive literature review, the kinematic modeling process of the completely constructed machine, based on homogenous transformation matrices, the machine structures and the assembling sequences is presented. The derived kinematic model provides nominal parameters for the motion interpolator and has the flexibility for deriving the quasi-static error model for future kinematic calibration and compensation.

Next, the original servo control system of the machine was extensively tested. Problems related to the hardware and software were found and solved through reconfiguring and rewiring hardware, modifying and developing new software functions. The main motion functions of the servo control system, improved and developed both on a single axis test system and the practical machine, make the machine work safely and effectively. .

The six axis motion interpolator for the Tiara hexapod was designed based on linear blended move motion. It uses the inverse kinematics model and the identical motion time to coordinate the motions of the six parallel axes to generate the desired tool tip motion with specified feedrate. Based on two user-defined parameters, the peak jerk and the peak acceleration, the interpolator generates a smooth pure, or partial, S-curve velocity profile and cubic position profiles. After being integrated with the servo control system, the interpolator is successfully tested for the single move and blended move motions of the machine.

Finally, the performance of the improved servo control system and online interpolation motion were experimentally evaluated for the un-calibrated machine. Both single axis motions and linear interpolated motions were realized with the correct motion directions and shapes, as well as within the specified time period. The positional repeatability of axis homing motion and linear interpolated motion was measured to be less than 0.001in (25.4microns). Due to errors in the nominal parameter, the linear interpolated motion straightness accuracy was found to be very low with a maximum error of 0.108in (2.74mm).

Acknowledgements

I would like to express my most sincere gratitude to my supervisors, Dr. G. Bone and Dr. M.A. Elbestawi, for their patience, guidance and continual support throughout the course of this research. Dr. Bone has always been available, approachable, and willing to share his valuable experience and expert knowledge in the fields of mechanical engineering and control. I am very grateful for what I have been able to learn from them.

I would like to thank Dr. S. Veldhuis for his guidance and sharing his expertise in machine tool accuracy and metrology.

Gratitude is also expressed to Dan Centea and Warren, for providing the design data of the Tiara hexapod, to Jim Mclaren, for his assistance in testing, disassembling and re-assembling the machine in the Lab.

I am eternally grateful to my family and friends for their encouragement and support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

A hexapod, a type of Parallel Kinematic Machine (PKM), is a member of a novel generation of machine tools. It typically consists of a moveable spindle platform connected to a rigid base or frame through six identically jointed struts with variable length or fixed length. Due to its potential capabilities for high dimensional accuracy, high structural rigidity and high speed capacity, etc., it is regarded as the most radical change in the design of machine tools in the last century and as a potential solution to high speed machining. In addition, by the low effective inertia of the moving components, it seems to overcome the dynamic limits of conventional Serial Kinematics Machine (SKM) machining systems. Due to the repetition and simplicity of the mechanical components, and the relatively inexpensive machine frame, it might also decrease the cost to produce machine tools[1][2][3][4][5].

Since its first public presentation in 1994, more than 200 different parallel mechanisms in hexapod or tripod types, mostly as prototypes or for academic studies, have been built. However, the industry and market response was quite poor: it is observed that out of the growing number of hexapod type machines, hardly any are doing productive work in industry while almost all of them stay in university laboratories [4][6].

Suitable solutions for some of the following obvious weakness of hexapod are not yet available. These weaknesses include:

a) Unfavourable ratio of system to work space; b) Limited dexterity and small tilting angels (±15°~±30°); c) Inherent danger of strut collision and singularities in workspace; and d) Lack of standard components and calibration/compensation procedure [3].

There are even highly contrary opinions about hexapods[7]. The concept was claimed to be inferior to serial machines and practically not useful: the variable length strut machine used as a milling machine was strongly inferior to the classical CNC machine, while the fixed length strut machine seemed better because of stiffer struts, but the unfavourable transmissions caused strong variations of stiffness at the platform end. However, It has been acknowledged that with careful design, the concept of the fixed length strut machine might perform comparably to the conventional CNC in particular application.

In spite of this, there is widespread agreement that their potential for innovative, beneficial use well warrants a continuing effort to make hexapod machine tools truly "productive"[4]. Since the first parallel kinematic machine has been established in industrial application with customer benefits, it has been proven to be feasible approach to improve machine tool's performance. The future research and developments enabling a more wide practical use of hexapod include [4]:

1) Design tools; 2) New machine components such as improved joints; 3) Commercial motion controllers capable of dealing with non-linear dynamics and coupling effects; and 4) Maintenance topics, such as machine calibration and accuracy

evaluation.

In order to fully exploit the capacity of a hexapod from all the four research aspects, Tiara hexapod, a hexapod milling machine with fixed length struts and movable base joints, is being developed at McMaster Manufacture Research Institute (MMRI), McMaster University. Previous to this thesis, research work on the Tiara hexapod mainly included: conceptual mechanical design; kinematic modeling and stiffness analysis; detailed mechanical design & construction of the machine; preliminary development of an open architecture controller to the machine drive system & sensors; theoretical framework of kinematic modeling, accuracy analysis and calibration[51][52[53][54].

Following on the previous research work, the objectives of this thesis include:

1) Derive the nominal kinematic model of the completed Tiara hexapod and provide geometric parameters for the interpolator. The kinematic model should have flexibility for deriving the quasistatic error model for future kinematic calibration and compensation.

2) Improve the original servo control system of the machine.

3) Develop an online motion interpolator and integrate it with the improved servo control system.

4) Test the servo control system and interpolator functions, and finally evaluate the motion performance of the machine.

Chapter 2 reviews the current research and development state of hexapods. Emphasis is placed on the research related to kinematic and error modeling, kinematic calibration, performance & error measurement, as well as servo control and interpolators.

Chapter 3 presents the kinematic modeling process for the Tiara hexapod. Based on the previous kinematic modeling framework, the model is constructed with homogenous transformation matrixes technique, following the assembly sequences of the machine and considering the actual machine structure and components. Next, Chapter 4 details the improvement of the servo control system of the machine. The development of the online interpolator for the machine is presented in Chapter 5. The experiment testing of the machine is described in Chapter 6. This involves testing the servo control system and motion functions. Chapter 6 also details the accuracy performance measurement of the machine. Chapter 7 closes the thesis with a set of conclusions and recommendations for the future work on the Tiara hexapod.

Chapter 2

Literature Review

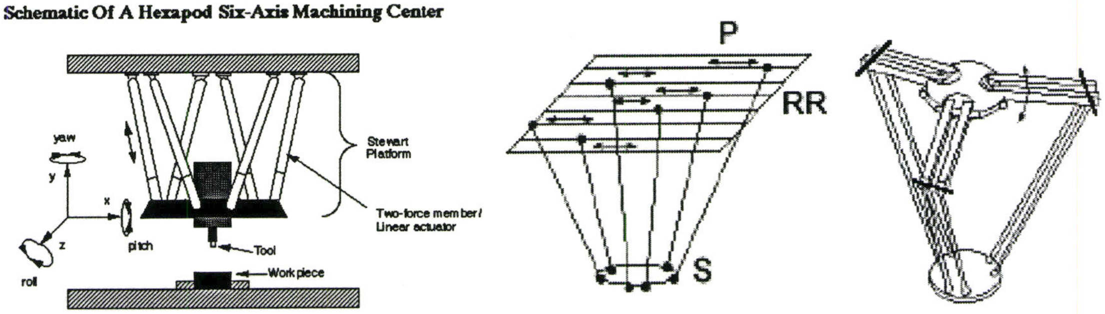
2.1 Overview

This literature review discusses the development of Parallel Kinematic Machines and more specific to this thesis, hexapods. The definition and classification of Parallel Kinematic Machines are introduced first. Then the discussion will focus on the related topics, including: kinematic and geometric error modeling, measurement, controller design and requirements, as well as motion interpolation. Finally, the chapter will close with some concluding remarks.

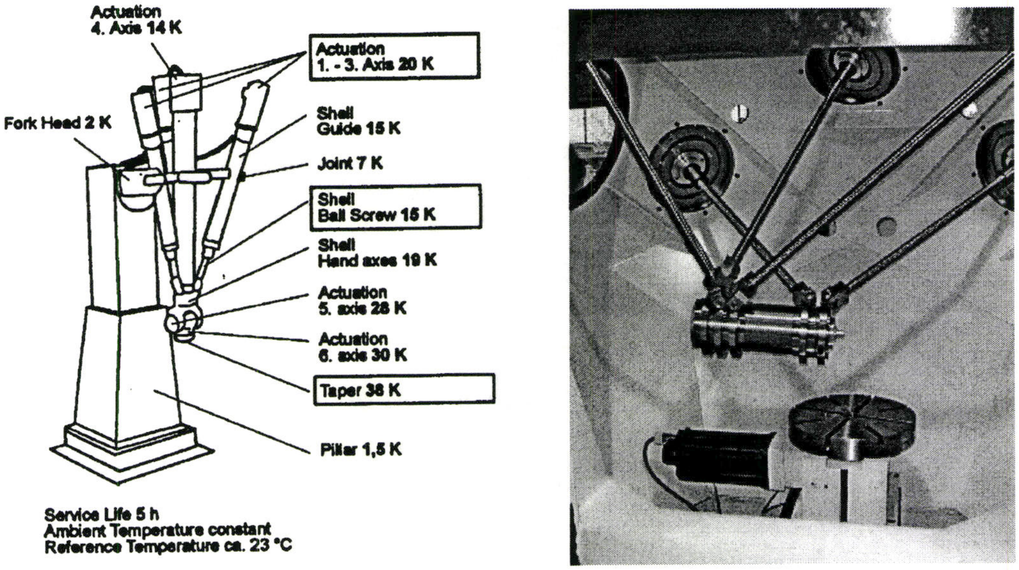
2.2 Introduction and Classification of PKM

A Parallel Kinematic Machine (PKM), typically consists of a moveable spindle platform connected to a rigid base/frame through up to six identically jointed struts with variable length or fixed length. Due to its potential capabilities of higher rigidity, higher accuracy and higher dynamic performance, as well as simple inverse kinematics, it is regarded as the most radical change in the design of machine tools in the last century and as a potential solution to high speed machining. However, PKM has inherent disadvantages such as smaller workspace volume, singularities, complex forward kinematics, difficult motion control, measurement and calibration [1][2][3][4][5][6].

According to the different configurations, PKM can be classified as: Hexapod (Figure 2.1 (a)), Hexaglide (Figure 2.1 (b)), Delta robot (Figure 2.1 (c)) ,Tripod (Figure 2.1 (d)), Others (Figure 2.1 (e)).



(a)Hexapod[4] (b) Hexaglide[4] (c) 3 DOF Delta[4]



(d) Tripod[25] (e) 5 DOF PKM[4]\

Figure 2.1 Classification of PKM

For the machining application, the three main types of PKM are compared in Table 2.1 according to load/capacity, speed, stiffness, workspace/area, structure/weight rate. It can be seen that the Hexaglide, the fixed length strut hexapod with movable base joints, is the best one for the machining application. The focus in the following sections of this chapter is on the kinematic modeling, geometric error modeling and measurement, controller design and motion interpolation of PKM for machining. Although some papers used different terms such as "Parallel manipulator/end effector", "Stewart platform", the ideas encountered are applicable to all type of PKM, including those for machining.

Table 2.1 Comparison of different PKM

Property/type	**Hexaglide**	**Hexapod**	**Delta/Flexure**
Load/capacity	Large	Large	Small
Speed	Fast	Fast	Faster
Stiffness	Better	Better	Bad
Workspace/area rate	Average	Bad	Good
Structure/weight	Big/average	Big/average	Simple/light

2.3 Kinematic and Geometric Error Modeling for PKM

The kinematic model of a machine describes the complex machine structure in a simple way that can reproduce adequately the real behaviour and performance of the machine. The geometric aspects derived from kinematic model are expressed in a mathematical model, called the geometric error model. The geometric error model describes how the relevant errors in the actual geometry of machine components, known as the parametric errors, cause errors in the position and orientation of the machine TCP (tool center point) [8][18].

Two different approaches to the kinematic modeling of PKM were proposed: the strut vectors representation [14][18] and the D-H (Denavit-Hartenberg) formation [10][13]. The former contains a faster evaluation of the kinematic performances. The latter includes the elementary components of the joints in the model. This gives the possibility to model manufacturing errors, assembly errors and kinematic errors in the actuators. The geometric error model is derived from one of these two kinematic models and expressed in different ways corresponding to the error measurement methods.

Geometric error model based on the forward measurement residual

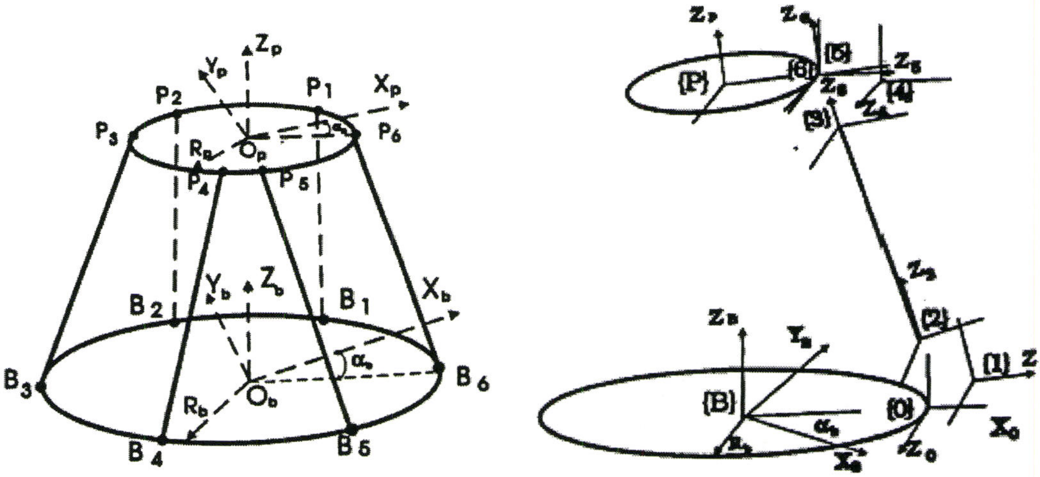
Wang and Masony [13] proposed that an error model with 132 parameters to be identified for the FAU Stewart platform hexapod. The inverse kinematics model of the platform schematically expressed in Figure 2.2 was given as

$$l_i = |b_i - Rp_i - t|, \quad \mathrm{i} = 1,2,\dots,6 \tag{2.1}$$

Where b_i, p_i are joint location on the base and platform, l_i denotes leg length and (R, t) the orientation matrix and position vector of the platform. Manufacturing tolerances of the joints, installation errors of those joints and measurement offset of the link actuators were studied and estimated. Simulation results showed that base error caused significantly large pose (platform position and orientation) errors than other error sources. The joints manufacturing tolerances had a minor effect on the pose errors. In their more recent paper [10], the authors proposed a simplified error model with only 42 parameters to be identified.

Patel and Ehmann [14] also adopted a reduced error model for a Stewart platform that omitted the manufacturing error of the joints. They defined vector chains as in Figure 2.3 (a) with the definition of Stewart platform the same as in Figure 2.2 (a).

$$\lambda_i {}^B l_i = {}^B R_p {}^P p_i + {}^B t - {}^B b_i, \quad i = 1,2,\ldots,6 \quad \text{and} \quad {}^B u_i = {}^B R_p {}^P p_i \tag{2.2}$$



(a) Stewart platform definitions (b) Joint-link train

Figure 2.2 FAU Stewart platform [13]

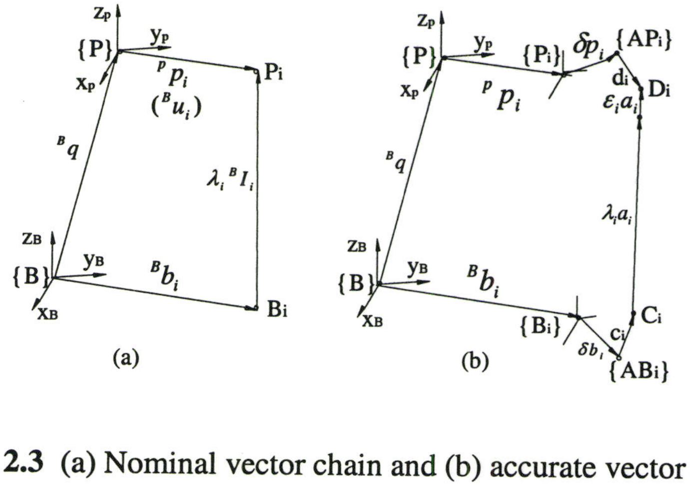


Figure 2.3 (a) Nominal vector chain and (b) accurate vector chain [14]

The nominal inverse kinematic model was expressed by taking the magnitude of the above equation as

$$\lambda_i = \left|\lambda_i {}^B I_i\right| = \left|{}^B u_i + {}^B t - {}^B b_i\right|, \quad \mathrm{i} = 1,2,\ldots,6 \tag{2.3}$$

The accurate chain model that they defined in Figure 2.3 (b) considered several error sources namely: manufacturing and assembly errors, actuators kinematic errors, elastic deformation and thermal deformation, as well as control errors. From the kinematic model in Equation 2.3, a differential error model based on the forward measurement residual was derived as:

$$\delta\Pi = J^{-1}(\delta\Lambda - N\delta A) \tag{2.4}$$

Where $\delta\Pi$ was pose error vector, $\delta\Lambda$ denoted leg length error vector, δA was joints parameters error matrix, the matrix N was a mapping from joint errors to leg length errors, and matrix J was the manipulator Jacobian. This kind of error model constructs the relationship between pose errors and parameter errors.

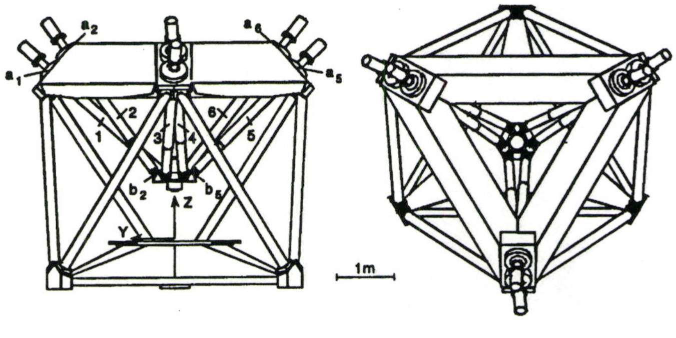


Figure 2.4 The prototype Octahedral Hexapod at NIST [17]

The error model presented by Soons [17] for the Octahedral Hexapod shown in Figure

2.4 was analyzed and found to be the same as the above one. With this kind of forward measurement residual error model, for calibration purpose the pose errors are expressed as the difference of the measured values and calculated values via forward kinematic model; and for error prediction purpose the pose errors can be computed with the modeled parameters errors.

Geometric error model based on inverse measurement residual

For a hexapod machine, it is well known that the forward kinematic problem is very complex and usually computation expensive via numerical methods, while the inverse kinematic problem is easy and straightforward. The error model introduced above was constructed based on the forward measurement residual, so the complex forward kinematic problem of PKM has to be solved numerically.

To avoid the undesired forward kinematic solution, Zhuang, Masory and Yan [15] [16] proposed another error model based on minimizing the inverse kinematic residual. They adopted the same differential error method [14] to the previous error-based model proposed in [10] for the FAU Stewart platform. However, the differential error model was formulated in terms of the inverse kinematic measurement residual, which was the discrepancy between the measured leg length and the computed leg length, instead of the difference between the measured pose and computed pose. The inverse kinematic model of the Stewart platform was thus expressed as

$$(l_i + \Delta l_i)^2 = (b_i - Rp_i - t)^T (b_i - Rp_i - t), \quad \mathrm{i} = 1,2,\ldots,6 \tag{2.5}$$

Where Δl_i was the fixed offset for each legs, and the other parameters were defined the same as those in the previous section. From this inverse kinematic model, the differential error model based on minimizing inverse kinematic measurement residuals for the FAU Stewart platform as

$$\delta L = J_{id} e \tag{2.6}$$

Where δL was the discrepancy between the measured leg length l_{ij} and the computed leg length $\underline{l}_{ij}$ via the inverse kinematic solution with measured platform pose, J_{id} was called Stewart platform identification Jacobian, and e was the 42 parameter errors' matrix. This kind of error model does not require the forward kinematic solution and avoids balancing the contribution of position errors and orientation errors in the minimizing process. The identification Jacobian matrix is a block diagonal matrix, therefore one can solve the error parameters for each leg independently and efficiently. This model has a generalized form suitable to variable length struts hexapod and base joint moveable fixed length struts hexapod.

Geometric error model adopted from conventional machine tool

Song and Mou in [19] constructed the error model for a variable length strut hexapod machine. Their research about this machine on error model, measurement and calibration as well as compensation was totally adopted from the conventional serial kinematic machine model based on homogeneous transformation matrices. Twelve joints, each with three position errors and three angular errors, and six struts with the length errors were considered. The transformation matrix between the mobile platform and the fixed base frame was expressed via the Euler angle method. Thus, the mobile platform position and orientation errors were modeled implicitly as functions of strut length errors, joint position and angular errors. There were totally 78 error terms to be identified. This error model still requires numerically solving the forward kinematic problem.

Geometric error model for self-calibration

The self-calibration method requires measuring the spatial coordinates of an artefact/standard by the machine itself when the artefact/standard is placed at different positions in the machine working volume [8][20]. It differs from the direct measurement method depending on whether an external measurement device is used or not. The error model for self-calibration can be either based on forward measurement residual or inverse measurement residual.

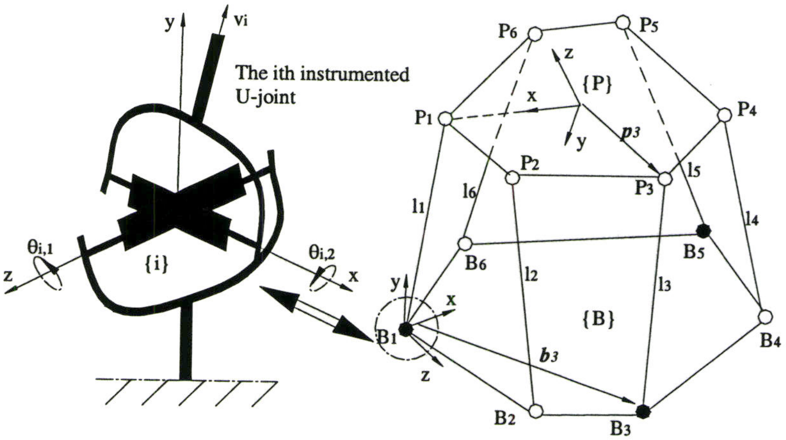


Figure 2.5 FAU Stewart platform definition for self-calibration [21]

Current PKM have not been equipped with internal sensors telling the pose of the platform. For self-calibration purpose, Zhuang in [21][24] added six redundant sensors on B1, B3 and B5 joints shown in Figure 2.5 of the previous mentioned FAU Stewart platform.

With the redundant sensors installed and the data mapping method presented by the same author in [22], three error models were formulated. One was based on forward error measurement residual; the other two were based on inverse measurement residuals for active joints and passive joints. Among them, the error model based on inverse kinematic measurement residual for the active joint was more practical. It differed from the standard one in Equation 2.6 in that the measured pose was obtained by internal active/redundant sensors and the forward kinematic solution incorporated the sensor information. Due to the additional six redundant sensors, the geometric error numbers were reduced from 42 to 36.

2.4 Geometric Error Measurement Methods for PKM

Error measurement is the key for PKM performance evaluation, kinematic calibration and accuracy compensation. Conventional metrology systems such as laser interferometers, ball bar and capacitance gages may be applied to the PKM. Other metrology devices such as the laser ball bar and those for robot metrology are also very useful, but special setup and machine motions need to be designed to obtain the desired results [18]. Besides those direct measurement methods, circular form test is applicable in practice to obtain manufacturing accuracy by commanding a series of circular motions [27].

Laser interferometry based measurement method

Although laser measurement is only an orthogonal one and each axis is investigated independently from the other axes, due to its high resolution and small measuring uncertainty it can be suitable for PKM measurement.

Wavering in [18] used an extra strut (shown in Figure 2.6) on which laser interferometer was installed as measurement device to obtain the total strut length of a hexapod machine.

Soons in [17] used a laser ball bar based on the same principle to measure the platform pose errors in the workspace. Their test results indicated that a bi-directional accuracy of about 6 microns could be achieved for the strut. However the measured poses covered only a small part of the workspace due to the limited range of the laser ball bar. The measurement strategy of the Hexel machine proposed in [26] is similar to that in [17] which utilized the triangulation measurement principle.

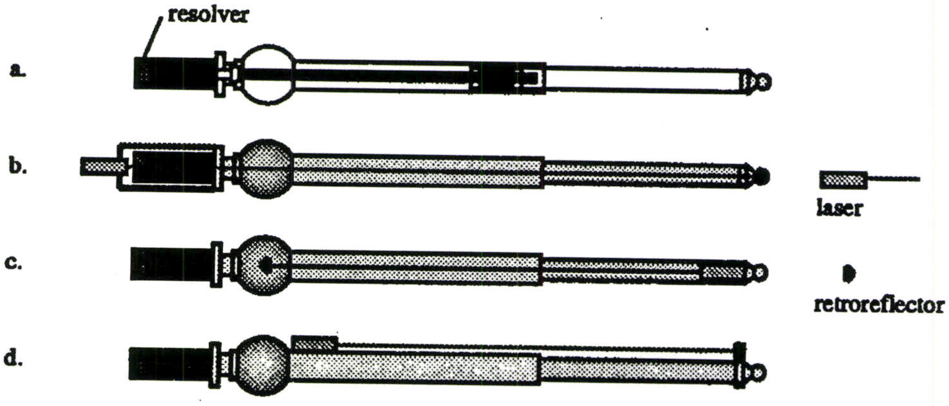
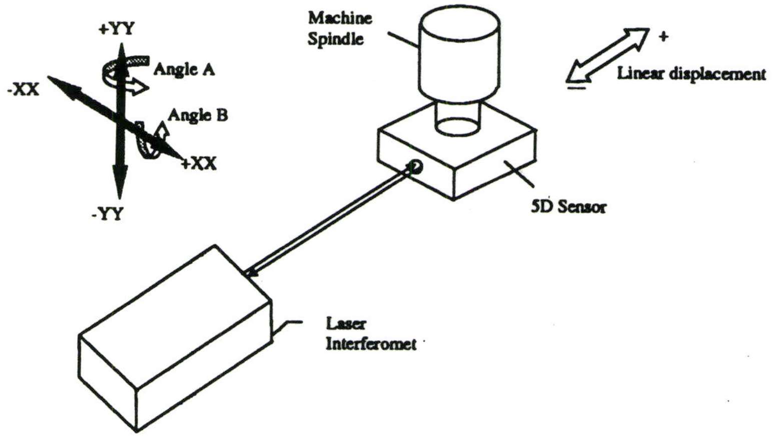
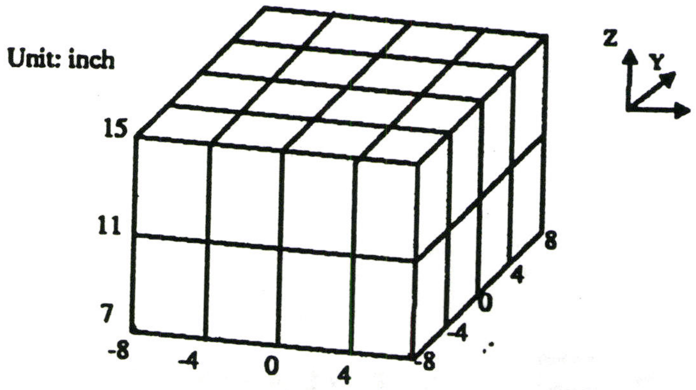


Figure 2.6 Possible strut length metrology systems [18]

Song and Mou in [19] adopted an API 5D (dimensional) laser interferometer system directly to measure the three virtual axes of a PKM as shown in Figure 2.7(a). The measured workspace was divided into a set of 4×4×4 cubes filling a $406.4 \times 406.4 \times 203.2\text{mm}^3$ space as shown in Figure 2.7 (b). The grid points where the boundary corners of these cubes overlap were the designed measurement points. Measurement was conducted along the x, y and z axis bi-directionally. The change of six actuators caused the hexapod platform's pitch, roll and yaw angular variation, and thus implemented the hexapod spindle's six degrees of freedom (DOF) movement. The 5D information was hereby obtained at the measurement points simultaneously.



(a) Laser interferometer system

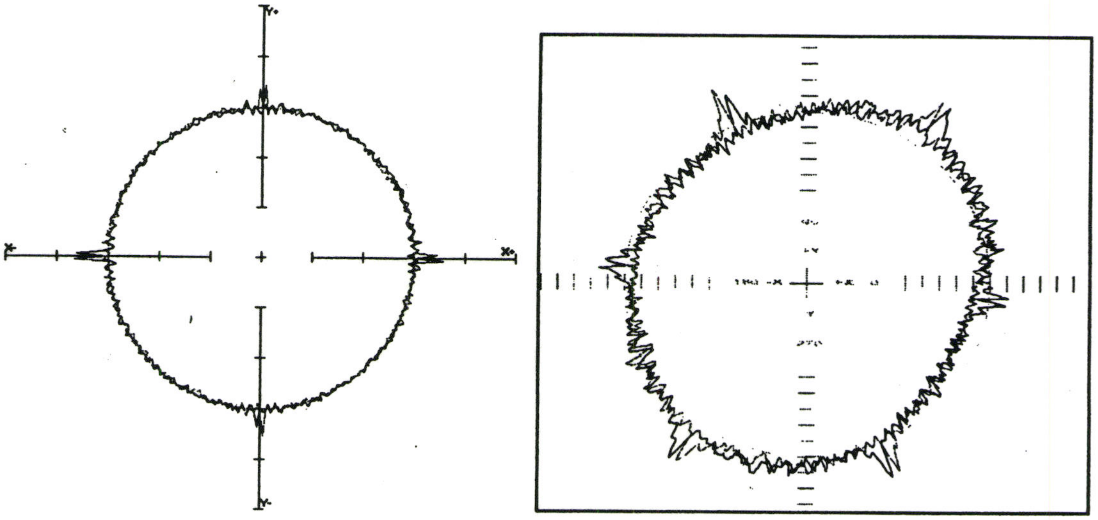


(b) Measurement layout in the workspace

Figure 2.7 Overall view of 5D Laser interferometer system measurement [19]

Ball bar probe measurement method

Powell, Whittingham and Gindy [27] performed probing repeatability test with a Renishaw Mp10 Machine tool probe to probe a block with a standard calibrated bore. The test aimed to assess the circular interpolation capability of a hexapod machine, the Variax. Ball bar data analysis plotted in Figure 2.8 shown that Variax had six reversal spikes at 60-degree interval, which were actually caused by the 12 reversals that occur with the six legs for every revolution. That result indicates that the ball bar test might show up to twelve discrete spikes for a different hexapod configuration.



(a) Conventional 3 axis machine (b) Variax

Figure 2.8 Ball bar test plot [27]

The circular form test was also performed by Heisel [25] and Hebsacker [31] on Triaglide of Mikron, a fully three DOF parallel mechanism. Test results found that, the over-swing on straight-line circular crossings that occurs with conventional SKM is hardly visible in the case of the Triaglide, and the acceleration step at the straight line to circle transition causes no significant deviation of the TCP.

These papers indicate that a ball bar is a good indicator for checking the general condition of a PKM if it is equipped with well-developed software.

Theodolite measurement method

Most of the above measurement methods directly come from the conventional CMM/CNC machine measurement and calibration. Zhuang, Masory and Yuan in [15] and [28] applied the serial robot measurement method to the PKM, in which a commercial electronic theodolite with resolution of 2 arc-second was used as measurement device. Survey theodolites are commonly used to measure large structures and adopted in the serial robot pose measurement [11]. Usually two theodolites are used together to find the spatial position of a target as shown in Figure 2.9.

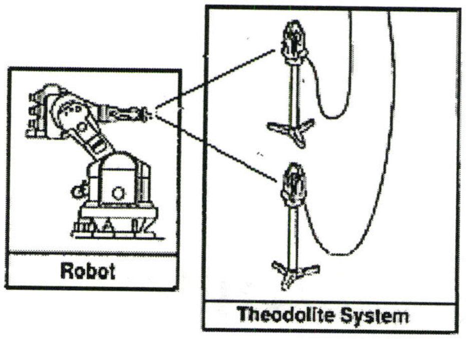


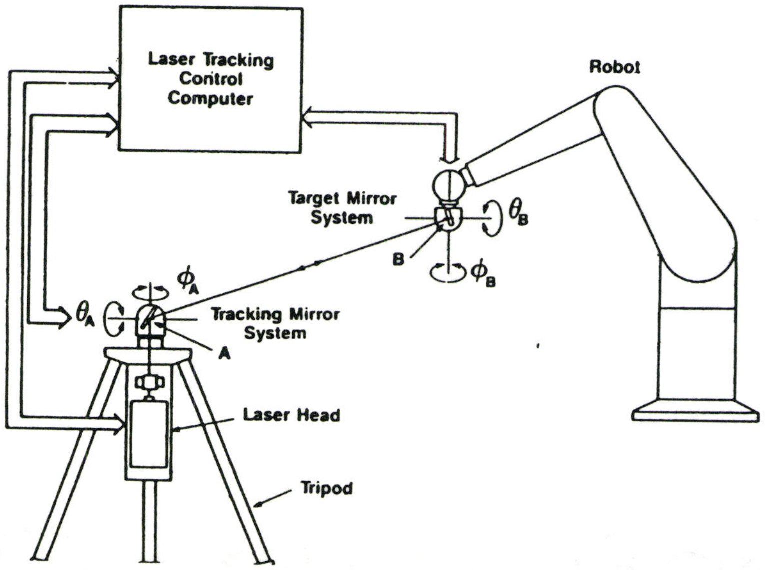
Figure 2.9 Typical layout of theodolite measurement system for serial robot [11]

Zhuang, Masory and Yuan [15][28] proposed two measurement procedures for FAU Stewart platform with one theodolite located at two stations. One procedure used two vertical tapes as assistant tools and the other used a horizontal bar. The position coordinates were calculated based on measured angles and trigonometry. Both of the procedures were verified, one on a CNC machine and the other on a CMM, to be accurate enough for their experiment. With that measurement system, the position repeatability of the FAU Stewart platform was found with the order of 0.1in(2.54mm), while the orientation repeatability is about 0.2 degree. Their one theodolite two station measurement system saves the cost for the second theodolite. Comparing with CNC machine (resolution 0.0001in(2.54microns)) for same distance measurement, the discrepancy between the CNC machine readings and the theodolite was about 0.005”(127 microns). That means the theodolite measurement system can be only used for measuring large spatial structures with relatively low accuracy.

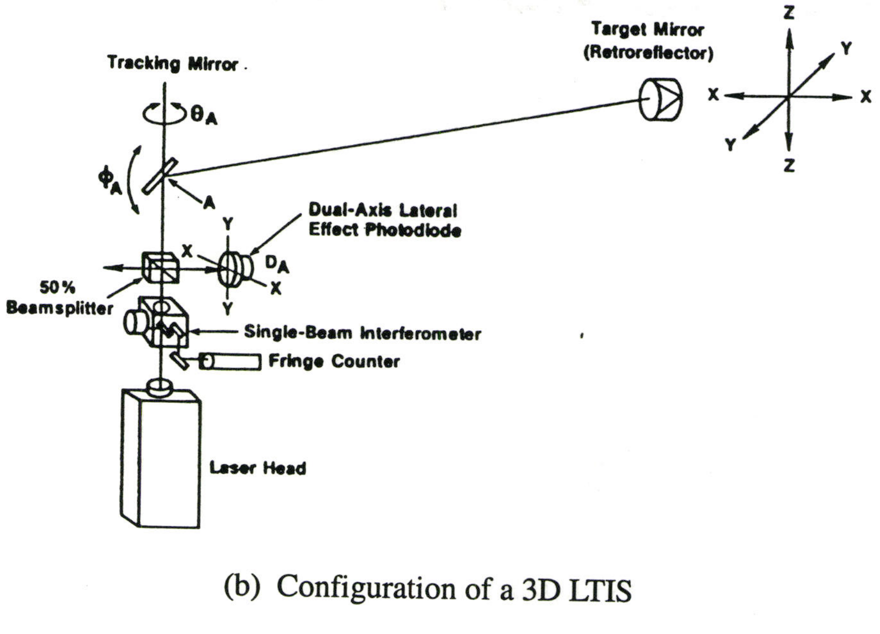
Laser tracking system measurement method

Laser tracking systems have been developed and used to measure the serial robot accuracy. The same principle has been used to measure the PKM, because laser ball bar is a kind of laser tracking system which has been used to measure leg length and pose of PKM in [17], [18], [26] and [28].

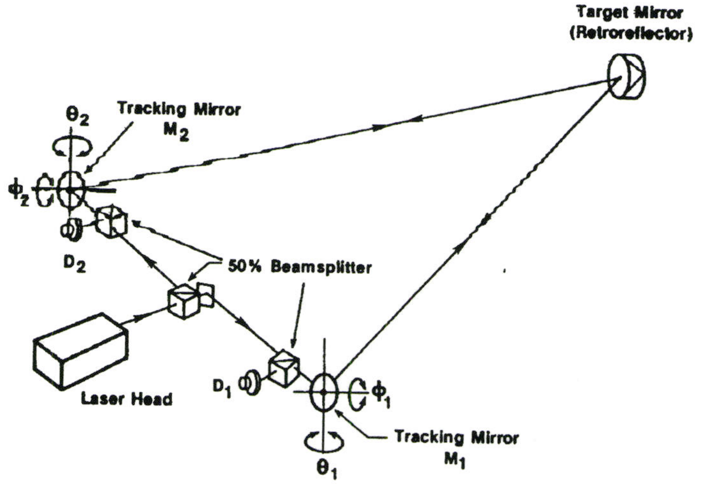
Lau, Hocken and Yuan in [29] developed two automatic single beam laser tracking interferometer systems (LTIS), one is 5D LTIS and the other is 3D LTIS as shown in Figure 2.10. The 5D LTIS is capable of measuring the pose of a moving target in five DOF except the yaw, while the 3D LTIS, a popular type in the current commercial market can only measure the X, Y and Z position of the end-effector. A 6 D LTIS was also proposed by Prenniger, Vincze and Gander [30] involving an extra vision system, but it is overly complicated and expensive. The 5D LTIS shown in Figure 2.10 (a) and the 3D LTIS in Figure 2.10(b) are compact and portable. Comparing with the two stations 3D LTIS shown in Figure 2.10 (c), they are less subjected to environmental effects such as changes in the reflective index of air, thermal gradients and vibration of the base stations. By installing three target corner reflectors on the end effector measuring their positons and analysing the data, the 3D LTIS can be used to determine the six DOF pose errors in the workspace. It is cost effective and can be used as a general-purpose laser interferometer.



(a) 5D LTIS



(b) Configuration of a 3D LTIS



(c) Two-station 3D LTIS

Figure 2.10 Configurations of 5D LTIS and 3DLTIS [29]

2.5 Servo Control Systems for Hexapods

PKM, especially Hexapod, are in general programmed in Cartesian space similar to conventional CNC machine tools. The transformation from Cartesian space to joint space is computed in the controller at least each interpolation cycle. Commercial control systems were generally used in most recently developed PKM. The specific transformation functions were either implemented on external processor boards or directly within the NC via original equipment manufacturer (OEM) interfaces [4].

Weck, Giesler, Meylahn and Saimer [1] pointed out that the implementation of PKM for machining application needed powerful servo controller based on Open Architecture principle. The control and programming should be similar to conventional CNC machine but necessarily include functions such as: manual operation in Cartesian space, the security of the machine for monitoring the workspace borders, checking collisions and jogging operation outside collision area. For the Octahedral hexapod, they implemented those functions on a Sinumerik 840D controller using the OEM-interface, the Dyna-M was controlled by the OSACA based WZL-NC.

Wavering introduced the controller of the experimental Octahedral Hexapod at NIST [18]. The controller consisted of an Ingersoll Intelligent Front End (IFE) computer and a conventional Computer Numerical Control controller. The IFE interpreted part program files (reading G and M codes), interpolated motion segments, performed inverse kinematic computation and checked workspace limits. It also provided jogging and manual data input (MDI) capabilities. The IFE sent strut position commands to the CNC that performed low level servo control and added compensations for effects such as ball screw lead errors and backlash.

Prior research has shown that an open architecture controller (OAC) is needed for that hexapod that can provide internal openness to allow the implementation and testing of experimental control algorithms, provide external openness to allow integration with higher level factory control systems and provide an additional test bed for the validation of open architecture interface standards being developed for machine tools and allow remote access of controller and sensor information.

Neugebauer, Wieland, Schwaar and Hochmuth [2] described the OAC requirement for a hexapod prototype: sufficient calculating capacity of control; open control concepts for hardware and software extensions; operating and NC programming similar to conventional ones with entire functionality maintained. They implemented this OAC with the commercial control Andronic 400 and a DSP card.

Pierrot and Shibukawac [46] introduced a PC based OAC system as shown in Figure 2.11 for the Hexam, a base joint movable hexapod. The PC based OAC consisted of a personal computer and Delta Tau's PMAC (Programmable Multi_Axis Controller) board [45]. PMAC, provided by the Delta Tau Data System, Inc, is a family of high performance servo motion controllers capable of commanding up to eight axes of motion simultaneously with a high level of sophistication. The motion control process was executed in real time operation. It was established by exchanging various data such as slide positions and external signals via dual port RAM on the PMAC board. The PC wrote slide position data, emergency stop commands or received external signals, etc. using the dual port RAM. The slide position data stored in the dual port RAM was converted into the positioning commands for servomotors by the CPU in the PMAC board and then interpolated into segments to be output every sampling period. Besides this, the PMAC board monitored external signals from operation switches, hydraulic and air pressure, as well as internal parameters.

Again using Delta-Tau's PMAC OAC board, El-Khasawneh and Ferreira [47] developed a control system schematically shown in Figure 2.12 for the Tetrahedral Tripod mechanism. This mechanism was actuated by hydraulic drives. In their controller, the inverse kinematics acted as an electronic cam relating the output of the interpolator (command positions) to the joint positions or the leg lengths. Using the PMAC facilities of PID control with velocity and acceleration feed forward, the following error was reported to be about 10 microns for contouring speeds of about 10ipm.

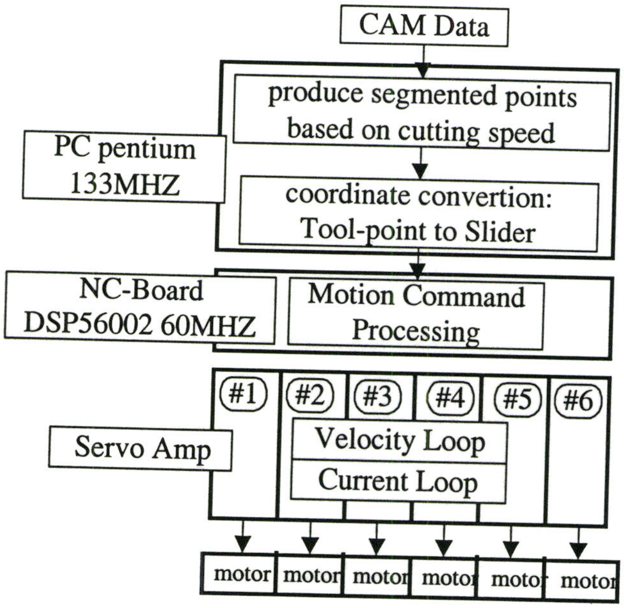


Figure 2.11 PC based OAC system of HexaM [46]

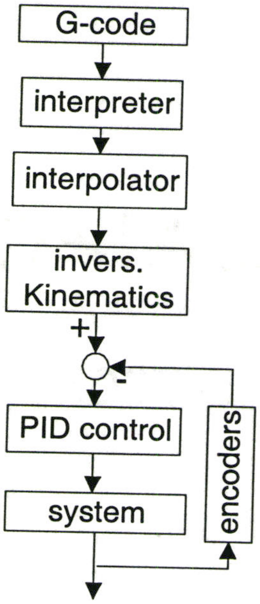


Figure 2. 12 Schematic of the OAC for Tetrahedral[47]

Weck and Saimer [4] summarized that the current drive control of PKM was usually realized in joint space by independent linear axis controllers designed similar to conventional CNC machines as cascaded P/PI controllers. With this kind of controllers, at higher velocities the tracking errors increased drastically thus the usable velocity had to be limited in favour of path accuracy. These limitations were induced by the non-linear behaviour inherent with PKM such as variable inertia as well as dynamic coupling effects of the individual drive axis. Non-linear control strategies, which incorporate the non-linear behaviour of the parallel structure into the control laws, require high computational efforts for the calculation of the dynamic properties on the one hand and fast interfaces to the drive controller on the other hand. Thus, their implementation into commercial machine tool controller is still an unresolved issue.

2.6 Interpolation Methods for PKM

PKM should be able to utilize the motion interpolator similar to that for conventional CNC machine, except a specific inverse kinematic model is applied to transform the interpolated tool path from Cartesian space to joint space. There are two main categories interpolators for CNC machine tools. One is hardware based Reference Pulse Interpolators, the other is Reference Word or Sampled-Data Interpolators [37] [38][39]

Interpolators based on Reference Pulse technique were limited by the computer hardware. They use a computer to produce a sequence of reference pulses for each axis of motion. Each pulse generates a motion of one basic length unit (BLU) of axis travel. The number of such pulses represents the position, and the frequency of the pulses is proportional to the desired velocity. The reference pulse interpolators are based on an iterative technique controlled by an interrupt clock. At each interrupt, a single iteration of the interpolation routine is executed. The feedrate of the system is proportional to the interrupt rate and therefore the maximum feedrate or axis velocity in BLU is limited by the computer hardware [37].

Another scheme, the reference word technique interpolator [38][39], is more complex and required substantially more memory, but it is not limited to a maximum velocity determined by the computer like the former one. The control loop is closed by software through the computer itself [39]. The interpolator generates the binary words in real time that is proportional to the desired velocity of the axis motion. The control loop compares a reference binary word with the feedback signal at constant sampling period to determine the position error. The error is finally used to form the motion command feeding to a DAC, which in turn supplies a voltage proportional to the required axis velocity. Therefore, the computer does not limit the maximum velocity with reference word technique.

Current computer technology has eliminated the need for Reference Pulse techniques. More recent research efforts are focused on interpolating general curves of any order and methods of interaction between CAD and CNC controllers. Koren, Lo and Sphpitalni [40] analyzed linear interpolation and its drawbacks in detail. They pointed out that approximation of general spatial curves with linear segments involved two competing criteria. The first was the requirement of accuracy or minimum tolerances. With an increase in curvature, an increasing number of linear segments were required. As a consequence, a large number of linear segments required greater transmission, storage and computational capacity, which would reduce the feed-rate, deteriorate the surface finish and increase the machining time. It was therefore concluded that even if computer hardware limitations were non-existent, the linear interpolation performed poorly with respect to general spatial curves.

Many research efforts have been done to improve linear interpolators and to provide solutions for CNC machine tool interpolators. The most significant advancement in generalized interpolator research is based on the machine tool model proposed by Chou and Yang [41]. Lin and Koren [42] considered the cutter orientation and machine tools kinematics of the five-axis surface interpolator. They analyzed the disadvantages of the conventional five-axis machining motion command generation architecture. Five axis machining requires mapping the segmented tool path points into the joint space via the inverse kinematics model, and then the linear interpolation of the resultant axes commands. They presented an alternative five-axis machining motion command generation architecture, which used a parametric interpolator of the tool path to accurately and smoothly command the axes.

For hexapod machine tools, Merlet, Perng and Daney [43] presented a method that described the tool path as 5-DOF tool path. In their research, the machine model was based on 6-DOF Stewart platform with orientation defined by means of the Euler angles ψ,θ,ϕ. The proposed algorithm aimed to optimize the unneeded sixth DOF, the rotation around the platform's normal and corresponding to the spindle's axis. This sixth DOF was parameterized by the angle ϕ. Given the desired 5-DOF tool path, their algorithm ensured the resulting 6-DOF trajectory lies inside the machine workspace, and avoids singularities.

However, this optimization algorithm was computational expensive, and therefore it is not well suited for a real time implementation.

Nguyen, Antrazi and Park [44] proposed three trajectory planning algorithms for a Stewart platform based end-effector with passive compliance for part assembly. The algorithms applied either to straight-line or arbitrary moves by generating trapezoidal link velocity profiles for the actuating stepper motors. They used an analytical solution for the parallel mechanism's inverse kinematics to transform the tool path in Cartesian space to the joint space. The trapezoidal velocity profiles were typically used to generate motion at the axis level, as they were simple to integrate and differentiate. However, to generate the linear varying velocity, the acceleration has to be piecewise constant. As a consequence, the rate of change of the acceleration, also termed the jerk, is infinity. Unfortunately, the bigger the jerk, the worse the dynamic property of the machine and the surface finish of the machined part.

The outstanding characteristics of the previously introduced PMAC board[45] are the power and flexibility of its trajectory generation algorithms: it uses piece wise constant jerk to generate triangular or trapezoidal acceleration profiles, which in turn produce full or partial second order velocity profiles and cubic position profiles for each motion axis. For this reason, PMACs have been widely used in OAC controllers for CNC machine tools [48][49] and robots, e.g. [50], as well as for PKM [46][47].

2.7 Conclusion

This literature review introduces a number of topics on PKM, including the kinematic modeling, geometric error modeling and measurement, servo controller and motion interpolation. Some conclusions are made below:

1) The inverse measurement residual error model is the most suitable error model for PKM.

2) The laser tracking technique (either 3D or 5D) is the best choice for pose and slide errors measurement. Survey theodolite is another alternative measurement device.

3) The servo controller should be designed using OAC concept and should include advanced non-linear control strategies, functions to ensure safe operation and accuracy related functions.

Chapter 3

Improved Nominal Kinematic Model of the Tiara Hexapod

3.1 Introduction

This chapter will continue the previous research work on kinematic modeling of Tiara hexapod. After introducing the actual structure of frame, joints legs and platform of Tiara hexapod, a detail derivation of the nominal kinematic model is carried out according to the assembling sequence of the machine. Parameters at each coordinate transformation and the final ones required by the motion interpolator are provided. Following the short description of the inverse kinematic model of the machine, the nominal workspace is calculated and compared with the previous research result. Note that all the positions are in millimetres and all angles are in degrees

3.2 Introduction to the Tiara Hexapod

As explained in chapter 2, for the machining application, the fixed length strut hexapod with movable joint base is the best choice. The Tiara hexapod, the focus of this research, belongs to this type of hexapod. It is shown in Figure 3.1 without worktable and pneumatic compensator installed. Its 3D CAD model is shown in Figure 3.2.

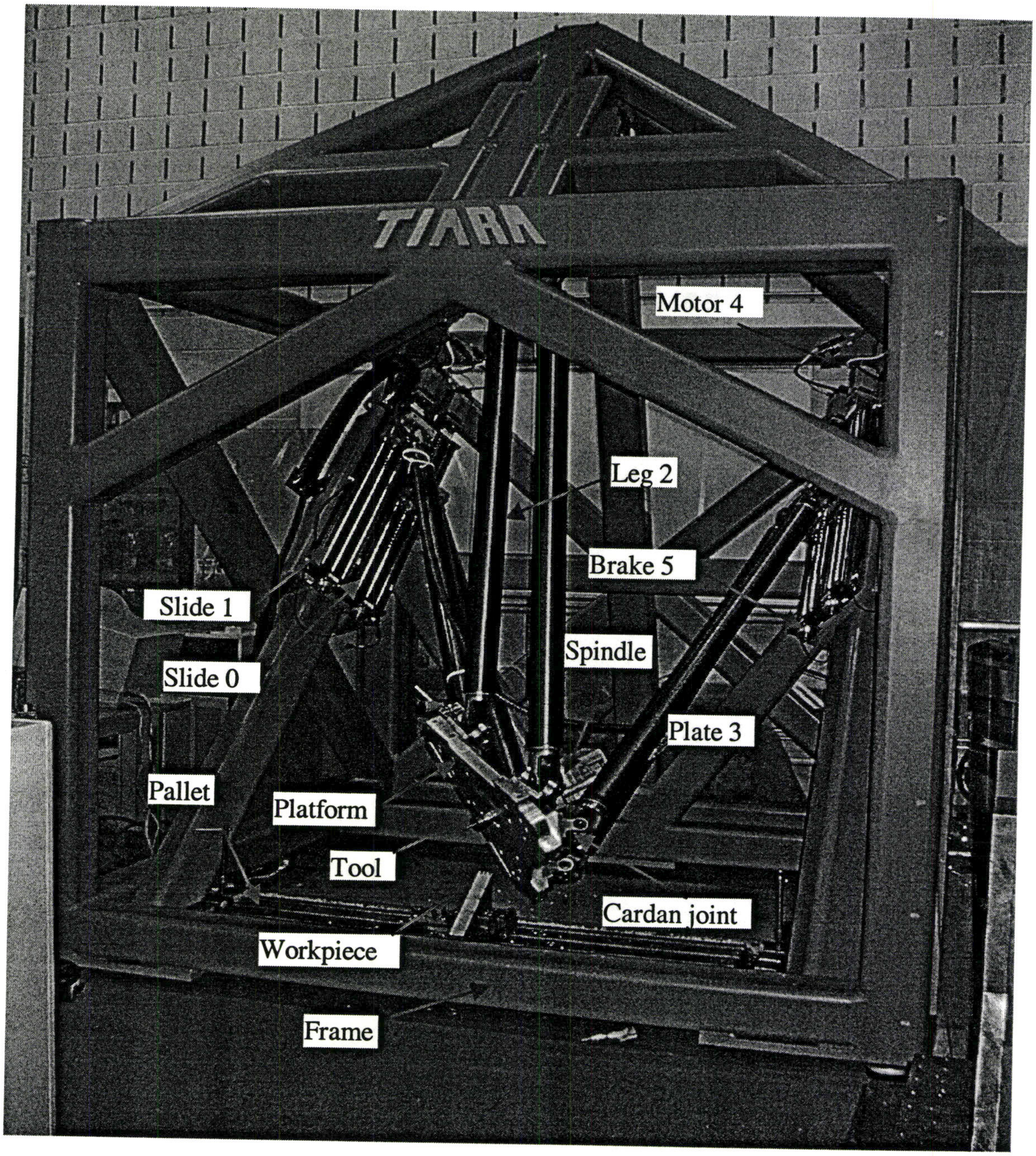


Figure 3.1 Picture of the actual Tiara hexapod

Its rigid frame is a welded frame structure of square steel tubes. Three plates were welded on three inclined beams on the frame with certain orientation angles. Two ball screw slides are then bolted on each plate. So there are totally six ball screws driven by six motors. Six special designed Cardan joints are mounted on the slide carriages with six special designed wedges. Another six of the same type Cardan joints are fixed on the movable spindle platform with six other type of wedges. The platform and frame are connected with 6 carbon fibre tubes that are permanently glued on the joints

The Tiara hexapod was designed to machine aluminium dies and moulds in automobile industry and to be inserted in the existing production line. This is why the Tiara hexapod has a unique layout with a 45-degree angle between the platform at home position and the shop floor.

To date, research work done on Tiara Hexapod includes [51][52][53][54]: conceptual mechanical design; kinematic modeling and stiffness analysis; detailed mechanical design & construction of the machine; preliminary development of OAC to the machine drive system & sensors; theoretical framework of accuracy analysis and calibration based on kinematics model. The complete machine shown in Figure 3.2 was constructed at the beginning of 2002, so most of the previous research work on the Tiara hexapod was not based on the actual joint-leg-platform system.

In the nominal model developed by Meyer in [53], the platform plate diameter R_{plat} was 178.9811mm, the diameter of the circle on the frame to which the slide mounting plate is tangent was 1879.2mm, leg length was 1280mm, Cardan joint height was 46.3mm. The corresponding values for the constructed machine are 270mm, 1930mm, 1426.246mm and 85 mm. Moreover, the joint dimensions, joint mounting wedge dimensions and their installation form on the actual machine have changed significantly from the previous work [53].

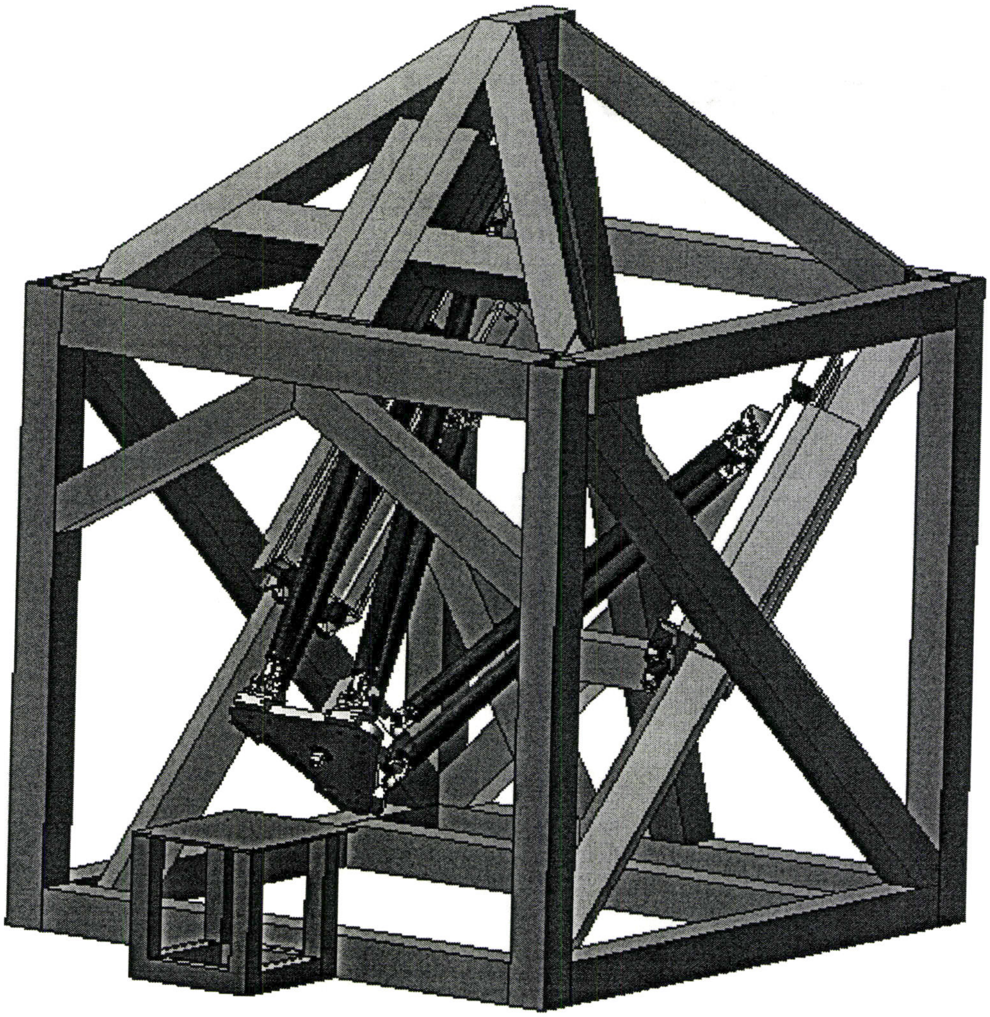


Figure 3.2 3D CAD model of the constructed Tiara Hexapod

Furthermore, in the original motion control system developed by Lacheray in [54] and [57], the nominal parameters for slide joint and platform joint centres were only 50% of the parameters given in [53].

Obviously, the nominal model and parameters provided by [53] and used in [54] and [57] are too inaccurate to be used for the machine motion control and performance evaluation.

In the following sections, based on the 3D CAD model of the Tiara hexapod and its components, and actual measurements for some components; and utilising some of the formulations and software developed in [51] and [53] for hexapod kinematic modeling, the nominal model of the actual Tiara hexapod is derived. This new model will provide more accurate and detailed nominal parameters for the motion interpolator.

The modeling procedure follows the machine assembling sequences and keeps the same parameters nomenclature as in [51][53].

3.3 World and Machine Coordinates Systems of the Tiara Hexapod

The world coordinate system ($O-X_O Y_O Z_O$) of the Tiara hexapod is located on the floor or the bottom plane of its frame as shown in Figure 3.3. In this simplified 3D model of the frame, the three plates with thickness 25.4mm to which the slides are attached are welded on the frame. The three centre lines of the plates are parallel to the three frame beams. These three lines are distributed evenly with 120 degrees interval on a cylindrical surface with 1930mm diameter. The machine coordinate system ($B-X_B Y_B Z_B$) is defined by the plate upper end surfaces as shown.

In the $B-X_B Y_B Z_B$ system, the Y_B axis is perpendicular to plate 3 and bisects it. The X_B axis is parallel to the external surface of plate 3 with positive direction pointing from plate 2 to plate 1. Thus the Z_B axis points to the floor with 45-degree inclined angle. Relationship between world coordinate system and machine coordinate system is constructed as follows.

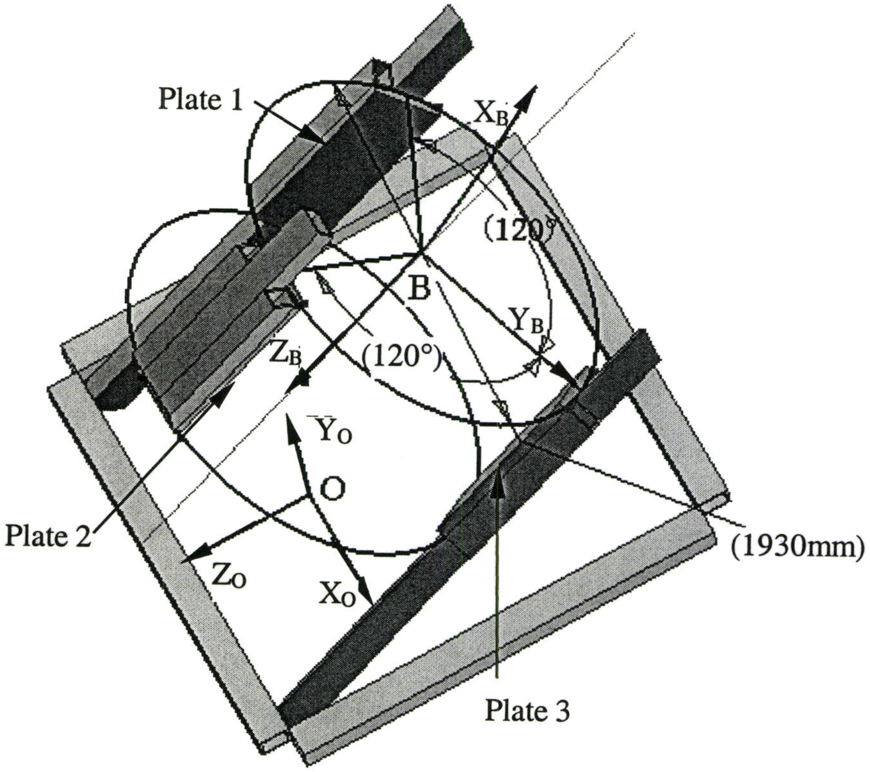


Figure 3.3 World and machine coordinate systems definition

First, the $O-X_OY_OZ_O$ system rotates 45 degrees around the X_O axis, the homogenous rotation transformation matrix is

$$ {}_{o}^{B}Rotate\_x = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.1} $$

Then, it rotates -120 degrees around the new Z_O axis

$$ {}_{o}^{B}Rotate\_z = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 & 0 \\ \sin\gamma & \cos\gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.2} $$

Finally, it translates by (0, 2100, –800) to obtain the coordinate system $\mathrm{B}\text{-}\mathrm{X_B Y_B Z_B}$

$$ {}_{o}^{B}Translate = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -2100 \\ 0 & 0 & 1 & 800 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.3} $$

The homogeneous transformation matrix representing the above transformations is expressed as

$$ {}_{o}^{B}T = ({}_{o}^{B}Rotate\_x)({}_{o}^{B}Rotate\_z)({}_{o}^{B}Translate) \tag{3.4} $$

With this transformation, the centre point (0,2100, -800) of upper circle in $\mathrm{O}\text{-}\mathrm{X_O Y_O Z_O}$ system in Figure 3.3 becomes the origin of $\mathrm{B}\text{-}\mathrm{X_B Y_B Z_B}$. For example, the point (835.7145, 1758.821, -1141.79) on the upper circle in $\mathrm{O}\text{-}\mathrm{X_O Y_O Z_O}$, which plate 3 is tangent to the frame at, becomes (0, 965,0) in $\mathrm{B}\text{-}\mathrm{X_B Y_B Z_B}$ system.

The parameter 965mm is named as R_{low} to follow the nomenclature used in the previous research [52] and [53]. The 25.4mm thickness of the plate is termed as *actplth.*

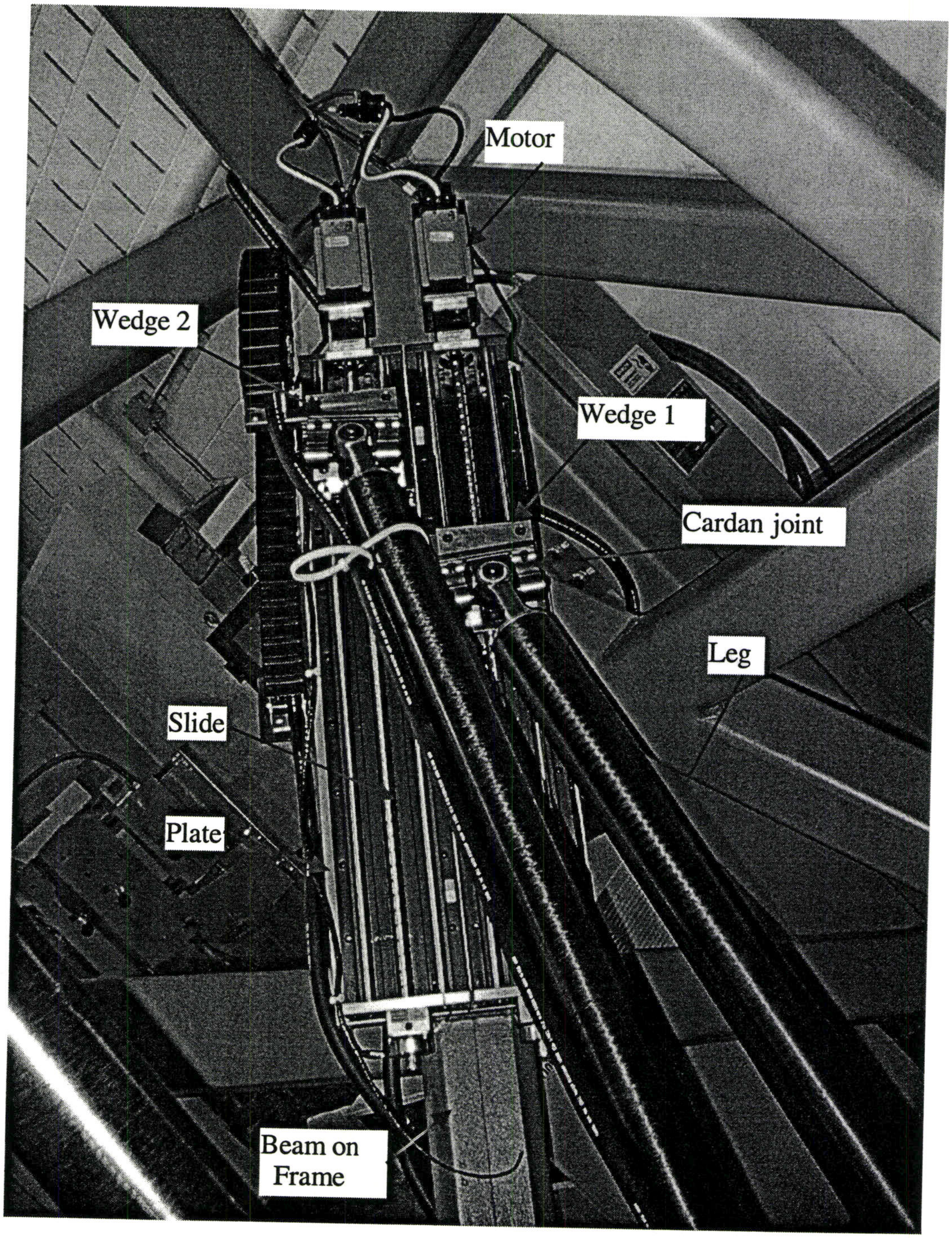


Figure 3.4 Wedges, joints and legs on the slides

3.4 Local Coordinate System and Nominal Parameters on the Slides

3.4.1 Plate Coordinate and Slide Coordinate System

As shown in Figure 3.4 and introduced in section 3.2, one pair of ball screw slides is attached to each of the three plates. A joint and leg is in turn installed on the wedge that is bolted on the slide. The local coordinate systems, Plate coordinate system, $Pl_j\text{-}X_{Pl}Y_{Pl}Z_{Pl}$, j=1,2,3; Slide coordinate system $Sl_i\text{-}X_{Sl}Y_{Sl}Z_{Sl}$, i=1,2,...,6, are defined in Figure 3.5.

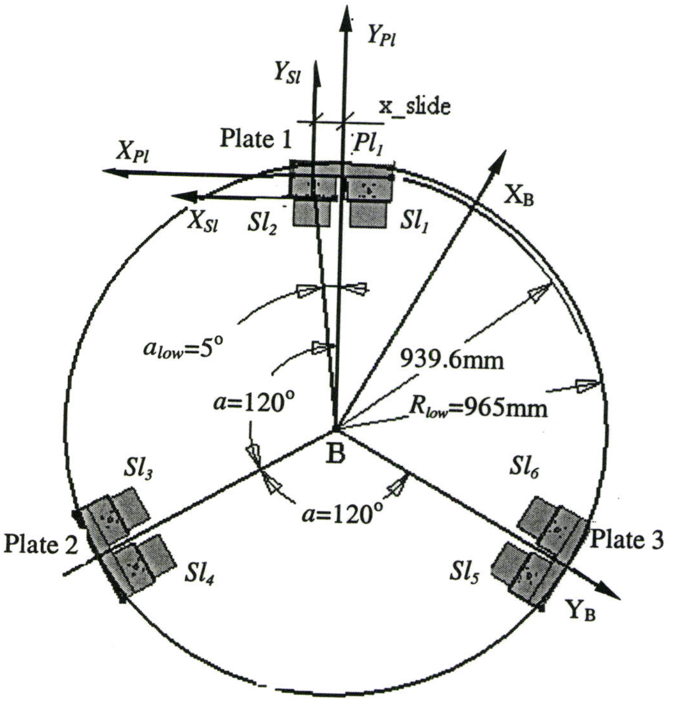


Figure 3.5 Machine, plate and slide coordinate systems definition

The plate system $Pl_j\text{-}X_{Pl}Y_{Pl}Z_{Pl}$ locates its origin at the centre point of the plate upper surface, with the Y_{Pl} axis along its thickness direction and pointing out of the circle with radius R_{low}. The Z_{Pl} axis is parallel to the $\mathrm{Z_B}$ axis of $\mathrm{B - X_B Y_B Z_B}$.

The machine coordinate system $\mathrm{B - X_B Y_B Z_B}$ rotated 120, 240 and 360 degrees around its $\mathrm{Z_B}$ axis and translated by (0, R_{low}-*actplth*, 0) gives the local plate system Pl_j-$X_{Pl}Y_{Pl}Z_{Pl}$ at plate *j*, *j*=1 to 3, via Equation (3.5).

$$ {}^{Pl}_{B}T_j = \begin{bmatrix} \cos\gamma_j & -\sin\gamma_j & 0 & 0 \\ \sin\gamma_j & \cos\gamma_j & 0 & -(R_{low} - actplth) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \gamma_j = -j*120^\circ, j = 1,2,3 \tag{3.5} $$

The origin of the *i*th slide coordinate system, Sl_i-$X_{Sl}Y_{Sl}Z_{Sl}$, is named $F_l\_slide\_center_i$ as shown in Figure 3.6. Its three coordinate axes are parallel to those of the plate coordinate system as shown in Figure 3.5 and 3.6. The relationship between the plate coordinate system Pl_j-$X_{Pl}Y_{Pl}Z_{Pl}$ and slide coordinate system Sl_i-$X_{Sl}Y_{Sl}Z_{Sl}$ is shown in Figure 3.6. It is a simple translation transformation expressed by Equation (3.6)

$$ {}^{Sl}_{Pl}T_{j,i} = \begin{bmatrix} 1 & 0 & 0 & (-1)^{i-1}x\_slide \\ 0 & 1 & 0 & acth \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ j = 1, i = 1,2; j = 2, i = 3,4; j = 3, i = 5,6 \tag{3.6} $$

Where $x\_slide = (R_{low} - actplth)\cdot\tan(a_{low}) = 88.9\text{mm}$ is defined in Figure 3.5, and *acth* (48mm) is the height of slide from its bottom to the top surface of carriage.

$F_l\_slide\_center_i$, the original slide carriage positions in machine system

$\mathrm{B} - \mathrm{X_B Y_B Z_B}$ can be obtained using

$$\begin{aligned} &{}^{B}F_{l}\_slide\_center_{i} = ({}^{B}_{Pl}T_{j})({}^{Pl}_{Sl}T_{i})({}^{Sl}Fl\_slide\_center_{i}) = ({}^{Pl}_{B}T_{j})^{-1}({}^{Sl}_{Pl}T_{i})^{-1}[0\,0\,0\,1]^{\mathrm{T}} \\ &j=1, i=1,2;\ j=2, i=3,4;\ j=3, i=5,6 \end{aligned} \tag{3.7}$$

3.4.2 Slide Wedge Coordinate System

There are two types of joint installation wedges installed on each pair of slides. The wedge2 as shown in Figure 3.4.and Figure 3.6 is installed on the even numbered slide, while wedge1 is installed on the odd numbered slide. Wedge 1 and wedge 2 are symmetric about the $Y_{Pl}Z_{Pl}$ plane of $Pl_j\text{-}X_{Pl}Y_{Pl}Z_{Pl}$ system, so we only need to study one of them. The wedge coordinates system $W\text{-}X_W Y_W Z_W$ is shown in Figure 3.6 and Figure 3.7. Its three coordinate axes are parallel to those of slide coordinate system. It can be obtained by translating the slide coordinate system to $(0,0,-(L_3 + H\_L_i + W\_off))$ with

$${}^{W}_{Sl}T_i = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & L_3 + H\_L_i + W\_off \\ 0 & 0 & 0 & 1 \end{bmatrix}, i = 1,2,\ldots,6 \tag{3.8}$$

Where L_3 (12.5mm) is the nominal thickness of slide block given by the slide manufacture, and W_off (40mm) is the wedge installation offset value on the carriage. H_L_i is the distance between the carriage and slide block when the slide carriage is located in the home position. This value is obtained through experimental measurement in Chapter 6. So we temporally assign the ${}^{W}_{Sl}T_i$ as a unit matrix, and set another vector

${}^{B}F_{l}\_z_{i}$ to include the Z_{B} direction translation terms:

$$ {}^{B}F_{l}\_z_{i}=[0\ \ 0\ \ L_{3}+H\_L_{i}+W\_off]^{T},i=1,2,...,6 \tag{3.9} $$

This vector will be calculated and added to slide joint centre at home position when the measured H_L_{i} is available.

3.4.3. Slide Joint Coordinate System

The Cardan joint designed and made in the MMRI (see Figure 3.4) will be bolted on the inclined surface of the wedge as shown in Figure 3.7. The joint coordinate system $J\text{-}X_{J}Y_{J}Z_{J}$ is hence constructed on the inclined surface of the wedge. J, the origin of the $J\text{-}X_{J}Y_{J}Z_{J}$ coordinate system, is defined as the center of the joint base. In the wedge system, J is ${}^{W}J=[x_{J}\ y_{J}\ z_{J}1]^{T}=[0\ \text{-}56.55\ \ 51.5\ \ 1]^{\mathrm{T}}$, while in machine system $\mathrm{B}\text{-}\mathrm{X}_{\mathrm{B}}\mathrm{Y}_{\mathrm{B}}\mathrm{Z}_{\mathrm{B}}$, it can be calculated with Equation (3.10):

$$ \begin{aligned} {}^{B}F_{l}\_joint\_base\_center_{i} &=({}^{B}_{Pl}T_{j})({}^{Pl}_{Sl}T_{i})({}^{Sl}_{W}T_{i})({}^{W}J) \\ &=({}^{Pl}_{B}T_{j})^{-1}({}^{Sl}_{Pl}T_{i})^{-1}({}^{W}_{Sl}T_{i})^{-1}[0\,\text{-}\,56.55\ \ 51.51\ 1]^{\mathrm{T}} \\ & j=1,i=1,2;\ j=2,i=3,4;\ j=3,i=5,6 \end{aligned} \tag{3.10} $$

Equations (3.7) and (3.10) produce the slide joint base vector:

$$ \begin{gathered} {}^{B}Slide\_joint\_base\_vector_{i}={}^{B}F_{l}\_joint\_base\_center_{i}-{}^{B}F_{l}\_slide\_center_{i}, \\ i=1,2,...,6 \end{gathered} \tag{3.11} $$

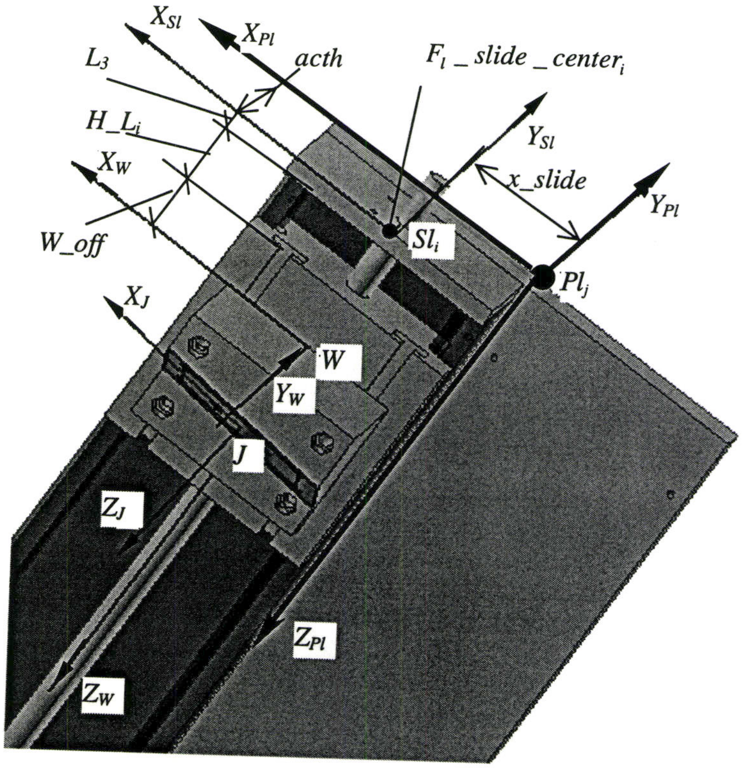


Figure 3.6 Relationship between plate, slide and wedge coordinate systems

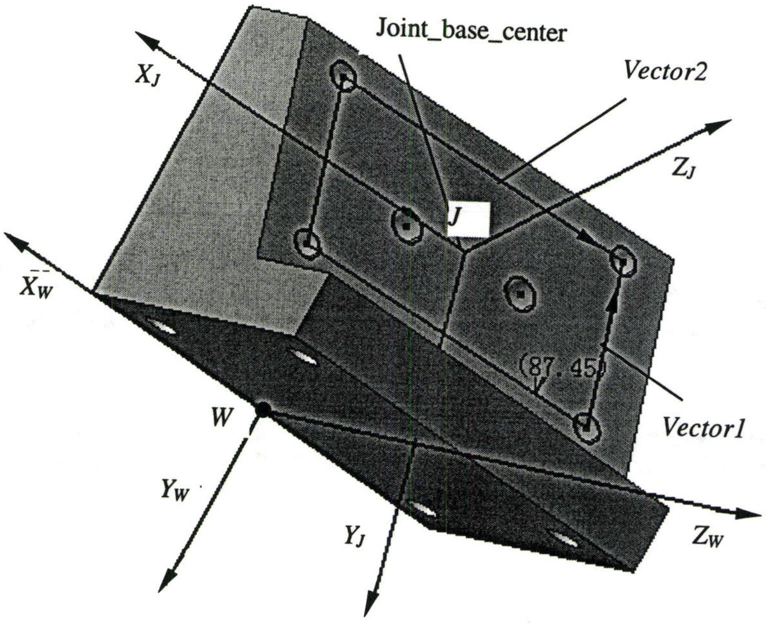


Figure 3.7 Wedge and joint coordinate system

In order to construct the relationship between $W\text{-}X_W Y_W Z_W$ and $J\text{-}X_J Y_J Z_J$, two vectors on the joint base surface are measured from the 3D model shown in Figure 3.7. Those two vectors, ${}^{W}Vector1 = [0\ 49.05\ 24.994]^{\mathrm{T}}$, ${}^{W}Vector2 = [-99.5\ 0\ -9.82]^{\mathrm{T}}$ are then used to construct the Z_J axis of $J\text{-}X_J Y_J Z_J$ in wedge coordinate system

$$ {}^{W}Z_J = \frac{{}^{W}Vector1 \times {}^{W}Vector2}{\left\|{}^{W}Vector1\right\| \left\|{}^{W}Vector2\right\|} = \left[z_x\ z_y\ z_z\right]^T \tag{3.12} $$

where $\times$ indicates the cross product operation and $\| \ \|$ indicates the length of a vector. From Figure 3.8, it can be found that vector ${}^{W}Z_J = \left[z_x\ z_y\ z_z\right]^T$ is collinear with the local Z_J axis of coordinate system $J\text{-}X_J Y_J Z_J$, so the rotation and translation transformation matrixes from $J\text{-}X_J Y_J Z_J$ to $W\text{-}X_W Y_W Z_W$ can be expressed as

$$ {}^{W}_{J}R_{rotation} = \begin{bmatrix} \frac{z_y^2}{1+z_z} + z_z & \frac{-z_x z_y}{1+z_z} & z_x & 0 \\ \frac{-z_x z_y}{1+z_z} & \frac{z_x^2}{1+z_z} + z_z & z_y & 0 \\ -z_x & -z_y & z_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.13} $$

and

$$ {}^{W}_{J}T_{transl} = \begin{bmatrix} 1 & 0 & 0 & x_J \\ 0 & 1 & 0 & y_J \\ 0 & 0 & 1 & z_J \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.14} $$

The Cardan joint is installed on the $J\text{-}X_J Y_J$ plane shown in Figure 3.8. In the joint coordinate system $J\text{-}X_J Y_J Z_J$, the joint centre is the intersection of the two axes of the joint. Its joint center position vector is designed as ${}^{J}Joint\_center_i = [0\ \ 0\ \ 55\ \ 1]^T, i = 1,2,\ldots,6$. So at slide home position, slide joint centre position vector in machine system $\mathrm{B}\text{-}\mathrm{X_B Y_B Z_B}$ can be calculated with Equation (3.15) and assigned to a temporary vector

$$\begin{aligned}{}^{B}F_l\_temp_i &= ({}^{B}_{Pl}T_j)({}^{Pl}_{Sl}T_i)({}^{Sl}_{W}T_i)({}^{W}_{J}T_{transl}\,{}^{W}_{J}R_{rotation})\,{}^{J}Joint\_center_i \\ &= ({}^{Pl}_{B}T_j)^{-1}({}^{Sl}_{Pl}T_i)^{-1}({}^{W}_{Sl}T_i)^{-1}({}^{W}_{J}T_{trans}\,{}^{W}_{J}R)[0\ \ 0\ \ 55\ \ 1]^T, i = 1,2,\ldots,6\end{aligned} \tag{3.15}$$

3.4.4 Slide Joint Parameters for the Interpolator

For a given pose (platform position and orientation), the interpolator of the hexapod uses the inverse kinematic model to verify if the machine can reach this pose. This model requires the original positions of the slide joint centres, the joint orientation vectors and the slide orientation vectors.

The first parameter is now given by

$${}^{B}F_{l_i} = {}^{B}F_l\_temp_i + {}^{B}F_l\_z_i, i = 1,2,\ldots,6 \tag{3.16}$$

This parameter is the actual slide joint centre home position, since it includes the measured homing distance H_L_i as mentioned in discussion of Equation (3.9).

The second parameter, the joint orientation vector, is defined as the slide joint centre vector in the machine coordinate system, *slide_joint_vector*. It can be found using Equation (3.17).

$$\begin{gathered}{}^{B}Slide\_joint\_vector_i = {}^{B}F_l\_temp_i - {}^{B}F_l\_slide\_joint\_base\_center_i, \\ i = 1,2,\ldots,6\end{gathered} \tag{3.17}$$

The *slide_joint_vector* will be used to calculate the joint pivot angles. In the nominal model, the slide is regarded as a perfect one. That means the joint on carriage will move along a straight line. Hence, the third parameter, the slide orientation vector is simply defined as a unit vector $U_i = [0\,0\,1]^T$ in the machine coordinate system.

Another parameter needed for the motion control is the maximum movement distance of each slide, $Lact_i$.

$$Lact_i = H\_frame - 2L_3 - T1 - H\_L_i - H\_U_i \tag{3.18}$$

Where *H_frame*, 1000mm, is the slide length; *T1*, 160mm, is the length of the carriage; and H_U_i is the distance between carriage and end block of the slide when the carriage reaches its maximum limits switch. The value of H_U_i is measured through the homing movement experiments presented in Chapter 6. The maximum position of the slide joint centre is denoted as ${}^{B}F_{ui}$ and expressed in Equation (3.19)

$${}^{B}F_{ui} = {}^{B}F_{li} + Lact_i * U_i, i = 1,2,...,6 \tag{3.19}$$

The computed values of these parameters are all listed in Appendix A. These values will be used with the inverse kinematic model and the online interpolator presented in Chapter 5.

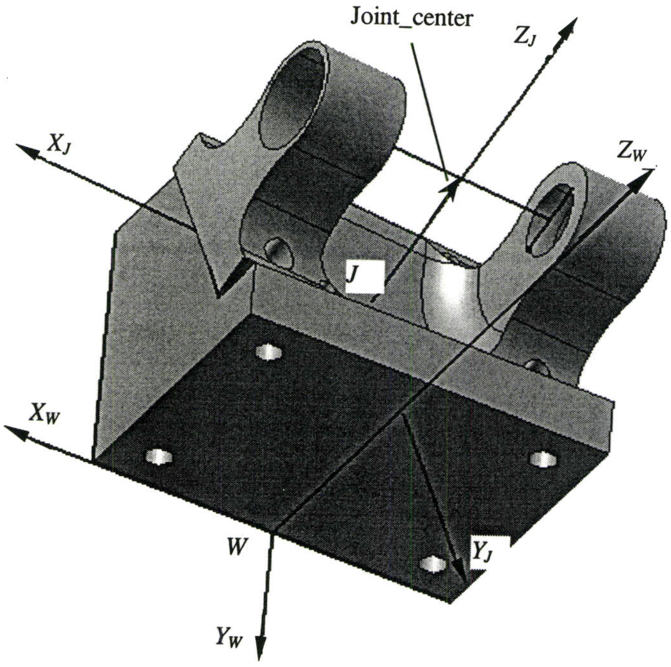


Figure 3.8 Slide wedge, joint housing and slide joint centre vector

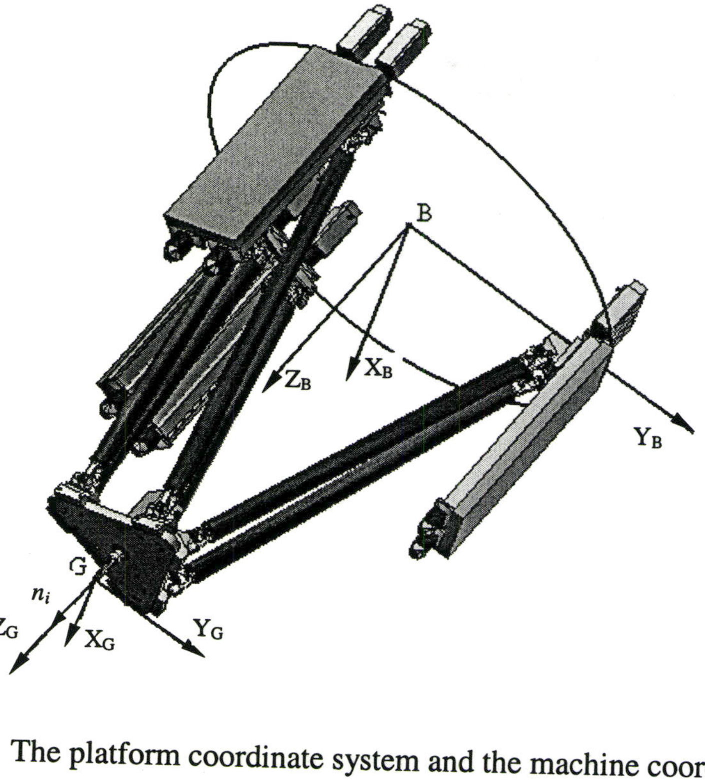
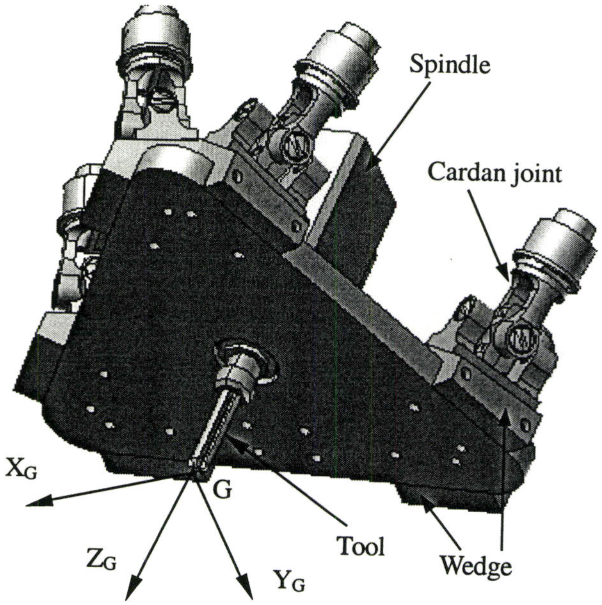
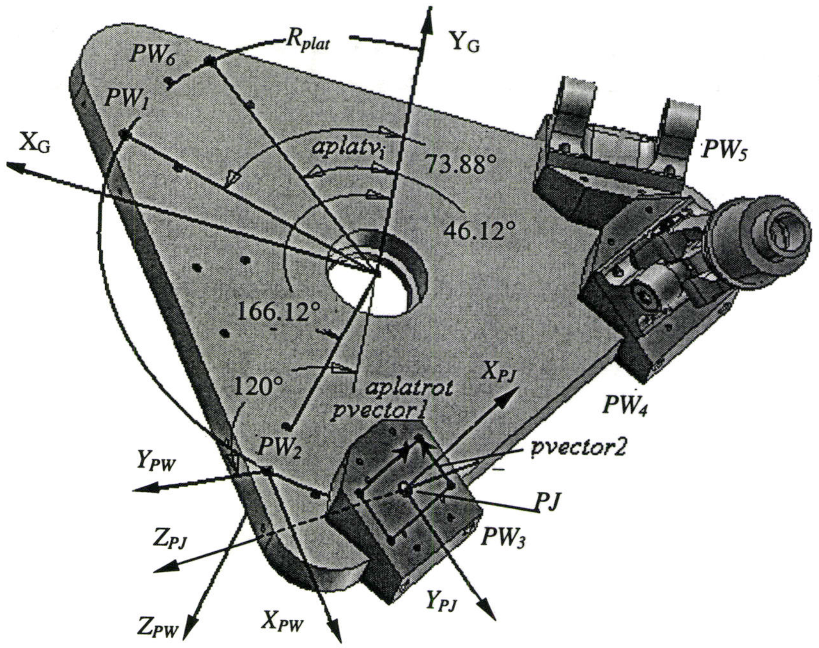


Figure 3.9 The platform coordinate system and the machine coordinate system



(a) The platform coordinate system



(b) The wedge coordinate system and the joint coordinate system

Figure 3.10 Platform related coordinate systems

3.5 Platform Related Coordinate Systems and Parameters

3.5.1 Platform Coordinate System

The platform structure, spindle, wedges and platform joints are shown in Figures 3.9 and 3.10. An Electrical spindle is bolted on the platform through the centre bore of the platform. Six wedges are installed on the platform, and six dowel pins are used to set their radial distance from the center bore (Figure 3.10(b)).

The dowel pin holes, labelled as $PW_i, i = 1,2,...,6$ in Figure 3.10(b), are distributed on the platform along a circle with angular interval $aplatv_i$ defined by Equation (3.21). The circle radius is defined as *Rplat* and equals to 270mm. The platform joints are mounted on the inclined surface of each wedge and are centred by the pinhole on that surface.

The 6 DOF pose of the platform consists of position and orientation terms in the machine coordinate system $B - X_B Y_B Z_B$. In Figure 3.9, G = $[Gx, Gy, Gz]^T$, the tool tip position in the machine coordinates of $B - X_B Y_B Z_B$, represents the platform position. While $\vec{n} = [n_x \; n_y \; n_z]^T$, the unit vector of the spindle in machine coordinate system, defines the orientation of the platform, since the spindle is perpendicular to the platform. The origin of the platform coordinate system $G - X_G Y_G Z_G$ is located at tool tip G.(see Figure 3.9 and 3.10(a)). The distance from the surface of the platform to tool tip is defined as the length of spindle *Lspi*. The Y_G axis of the platform coordinate system is perpendicular to the straight line connecting the centres of dowel pin holes 5 and pin 6 (see Figure 3.10(b)). The Z_G axis is perpendicular to the surface of the platform and points to the tool tip, so X_G axis is parallel to the connection line of pin holes 5 and 6 with positive direction pointing to pin hole 5. With the origin position G and platform orientation vector n known, the transformation from the platform coordinate system $G - X_G Y_G Z_G$ to the machine coordinate system $B - X_B Y_B Z_B$ can be constructed in the same way as that for slide joint and wedge system transformation in section 3.4.3. The formulation is also can be found in [51], [52] and [53] as

$$ {}^B_G T = {}^B_G T_{trans} \, {}^B_G R \tag{3.20}$$

Where $${}^B_G R = \begin{bmatrix} \frac{n_y^2}{1+n_z} + n_z & \frac{-n_x n_y}{1+n_z} & n_x & 0 \\ \frac{-n_x n_y}{1+n_z} & \frac{n_x^2}{1+n_z} + n_z & n_y & 0 \\ -n_x & -n_y & n_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$ and $${}^B_G T_{trans} = \begin{bmatrix} 1 & 0 & 0 & G_x \\ 0 & 1 & 0 & G_y \\ 0 & 0 & 1 & G_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

3.5.2 Platform Wedge Coordinate System

The six specially designed wedges are mounted on the platform as shown in Figure 3.10. A dowel pin locates each wedge's radial position from the centre of the platform. The wedge coordinate system $PW\text{-}X_{PW} Y_{PW} Z_{PW}$ therefore has its origin located at the corresponding pin hole centre on the platform upper surface. The pin hole centre position in the platform system is defined as $(x_p, y_p, -Lspi)$. Moving the platform coordinate system to $(x_p, y_p, -Lspi)$ first, and then rotating it around its Z_G axis with an angle *aplatrot*, gives the local wedge coordinate system $PW\text{-}X_{PW} Y_{PW} Z_{PW}$. Its Y_{PW} axis is perpendicular to the straight edge of the platform and points out of the platform, while its Z_{PW} axis parallel to the $\mathrm{Z_G}$ axis of $\mathrm{G - X_G Y_G Z_G}$ Therefore, the transformation matrix from the platform coordinate system $\mathrm{G - X_G Y_G Z_G}$ to the wedge coordinate system $PW\text{-}X_{PW} Y_{PW} Z_{PW}$ is same as those in [51] and [53]

$$ {}^{PW}_{G}T_i = \begin{bmatrix} \cos(aplatrot_i) & -\sin(aplatrot_i) & 0 & -x_{pi} \\ \sin(aplatrot_i) & \cos(aplatrot_i) & 0 & -y_{pi} \\ 0 & 0 & 1 & L_{spi} \\ 0 & 0 & 0 & 1 \end{bmatrix}, i = 1,2,\ldots,6, (aplatrot_i) \quad (3.21) $$

Where $aplatrot_5 = aplatrot_6 = 0^\circ$ $aplatrot_1 = aplatrot_2 = 120^\circ$ $aplatrot_3 = aplatrot_4 = -120^\circ$ are same as in those in [51], [52] and [53], <u>but the parameters given below are different</u>:

$$ \begin{aligned} & x_{P_i} = Rplat * \sin(aplatv_i), aplatv_6 = 46.12^\circ, aplatv_1 = 73.88^\circ\, aplatv_2 = 166.12^\circ \\ & y_{P_i} = Rplat * \cos(aplatv_i), aplatv_5 = -46.12^\circ, aplatv_4 = -73.88^\circ\, aplatv_3 = -166.12^\circ \\ & i = 1,2,\ldots,6, \qquad Rplat = 270mm \end{aligned} $$

This transformation matrix (3.21) is used to derive the pin hole centre location in the platform coordinate system $\mathrm{G - X_G Y_G Z_G}$ as

$$ {}^{G}platform\_pin\_hole\_location = {}^{G}_{PW}T\, {}^{PW}Pin = ({}^{PW}_{G}T)^{-1}[0\,0\,0\,1]^{\mathrm{T}} \quad (3.22) $$

3.5.3 Platform Joint System and Parameters for the Interpolator

As with the slide joint installation, the platform joint is installed on the surface of the wedge. The joint coordinate system $PJ\text{-}X_{PJ} Y_{PJ} Z_{PJ}$ is located at dowel pin hole centre on the wedge surface as shown in Figure 3.10. Its $X_{PJ}Y_{PJ}$ plane coincides with the wedge surface on which the joint will be mounted. Its X_{PJ} axis is parallel to the platform and points in the declining direction of the wedge surface, while its Y_{PJ} axis points out of the platform. The origin PJ of $PJ\text{-}X_{PJ}Y_{PJ}Z_{PJ}$ system is named as platform joint base centre. In the wedge coordinate system, its position vector can be measured from the 3D CAD model of a wedge as ${}^{PW}PJ = [x_{PJ}\ y_{PJ}\ z_{PJ}\ 1]^T = [0\ 0\ -31.071\ 1]^T$. In the platform coordinate system $\mathrm{G - X_G Y_G Z_G}$, this joint base centre is easy to be obtained with

$$\begin{aligned} {}^{G}platform\_joint\_base\_center &= ({}^{G}_{PW}T)({}^{PW}PJ) = ({}^{PW}_{G}T)^{-1}({}^{PW}PJ) \\ &= {}^{G}platform\_pin\_hole\_location + {}^{PW}PJ \end{aligned} \tag{3.23}$$

In order to construct the transformation matrix between the platform joint coordinate system $PJ\text{-}X_{PJ}Y_{PJ}Z_{PJ}$ and platform wedge coordinate system $PW\text{-}X_{PW}Y_{PW}Z_{PW}$, two vectors on the wedge surface, the $X_{PJ}Y_{PJ}$ plane of platform joint system, are measured in the wedge coordinate system as shown in Figure 3.10. They are ${}^{PW}pvector1 = [49.89 - 26.251\ \ - 9.0651]^T$, ${}^{PW}pvector2 = [-49.849 - 22.7151 - 15.979]^T$. The cross product of these two vectors is the orientation vector of the wedge surface, or the unit vector of Z_{PJ} axis of $PJ\text{-}X_{PJ}Y_{PJ}Z_{PJ}$ but expressed in wedge system $PW\text{-}X_{PW}Y_{PW}Z_{PW}$.

$${}^{PW}Z_{pJ} = \frac{{}^{PW}pvector1 \times {}^{PW}pvector2}{\left\| {}^{PW}pvector1 \right\| \left\| {}^{PW}pvector2 \right\|} = \left[pz_x\ pz_y\ pz_z \right]^T \tag{3.24}$$

The homogenous rotation and translation transformation matrixes from $PJ\text{-}X_{PJ}Y_{PJ}Z_{PJ}$ to $PW\text{-}X_{PW}Y_{PW}Z_{PW}$ can be in the form same as Equation (3.13) & (3.14) as follows

$$ {}^{PW}_{PJ}R = \begin{bmatrix} \dfrac{pz_y^2}{1+pz_z}+pz_z & \dfrac{-pz_x pz_y}{1+z_z} & pz_x & 0 \\ \dfrac{-pz_x pz_y}{1+pz_z} & \dfrac{pz_x^2}{1+pz_z}+z_z & pz_y & 0 \\ -pz_x & -pz_y & pz_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.25} $$

$$ {}^{PW}_{PJ}T_{trans} = \begin{bmatrix} 1 & 0 & 0 & x_{PJ} \\ 0 & 1 & 0 & y_{PJ} \\ 0 & 0 & 1 & z_{PJ} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.26} $$

Because the slide joint and the platform joint have the same design, in the joint coordinate system $PJ\text{-}X_{PJ}Y_{PJ}Z_{PJ}$, the joint centre, the intersection of the two axes of the joint is ${}^{PJ}Joint\_center_i = [0 \quad 0 \quad 55 \quad 1]^T, i = 1,2,\ldots,6$.

Through two stages of coordinates transformations, from the platform joint coordinate system $PJ\text{-}X_{PJ}Y_{PJ}Z_{PJ}$ to the platform wedge coordinate system $PW\text{-}X_{PW}Y_{PW}Z_{PW}$, then to the platform coordinate system $\mathrm{G\text{-}X_G Y_G Z_G}$, the joint centre in the platform coordinate system becomes

$$ \begin{aligned} {}^{G}platform\_joint\_center_i = {}^{G}P_i &= ({}_{PW}^{G}T_i)({}^{PW}_{PJ}T_{trans}\,{}^{PW}_{PJ}R)({}^{PJ}Joint\_center_i) \\ &= ({}^{PW}_{G}T_i)^{-1}({}^{PW}_{PJ}T_{trans}\,{}^{PW}_{PJ}R)[0 \quad 0 \quad 55 \quad 1]^T, i = 1,2,\ldots,6 \end{aligned} \tag{3.26} $$

The platform joint vector measured in $\mathrm{G\text{-}X_G Y_G Z_G}$ is simply the difference of joint centre and joint base centre, which will be used in inverse kinematic solution.

$$\begin{aligned}{}^{G}platform\_joint\_vector_i = {}^{G}&platform\_joint\_center_i \\ &-{}^{G}platform\_joint\_base\_center_i, \\ i = 1,2,..,6&\end{aligned} \tag{3.27}$$

All the parameters given above will be constant in the platform coordinate system, while in the machine coordinate system they will change with the platform pose variation. Because the inverse kinematic solution and machine motion interpolation are carried out in the machine coordinate system $\mathrm{B - X_B Y_B Z_B}$, those platform joint parameters need to be transformed to $\mathrm{B - X_B Y_B Z_B}$ system with Equation (3.20), as follows

$$\begin{aligned}{}^{B}P_i = {}^{B}platform\_joint\_center_i = {}^{B}_{G}T * {}^{G}platform\_joint\_center_i \\ = {}^{B}_{G}T * {}^{G}P_i, i = 1,2,..,6\end{aligned} \tag{3.28}$$

Besides the parameters of slide joint and platform joint derived above for the kinematic model and interpolator, the vector for Cardan joint axis1 (as shown in Figure 3.13), the slide joint installation angle between slide joint axis1 and $\mathrm{Y_B}$ axis of the machine coordinate system, the platform joint installation angles between platform joint axis1 and $\mathrm{Y_G}$ axis of the platform coordinate system, are also calculated and listed in Appendix A.

Another indispensable parameter needed for kinematic modeling and motion interpolation is the leg length, which is explained in the next section.

3.6 Leg length Parameters

The six legs physically connect the platform and the rigid frame of the Tiara hexapod. One complete leg assembly is shown in Figure 3.11. One rotary bearing is installed on the platform joint and inside one end of the carbon fibre tube. Each leg is assembled by first cooling the steel joints to make them contract, fitting them into the tube and forming the permanently connection with glue.

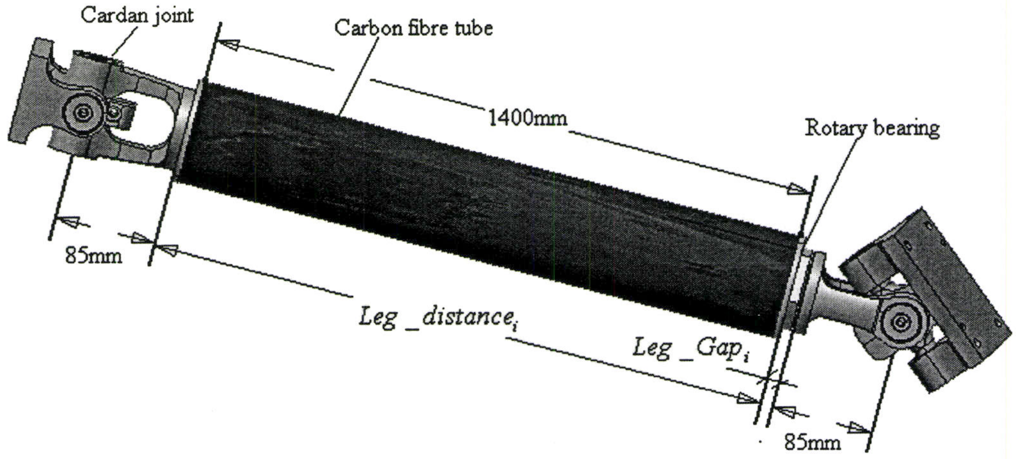


Figure 3.11 Leg length definitions

The leg length is defined as the distance between centres of the two joints. It consists of the three parts given in Equation 3.29:

$$L_i = Leg\_distance_i + Leg\_Gap_i + 2 * joint\_offset \tag{3.29}$$

Where $joint\_offset$ is designed to be 85mm, $Leg\_distance$, listed in Appendix A, Table A17, is the distance between the external surfaces of the two steel inserts on the joints. It equals to the tube length of 1400mm plus the flanges' thickness of the assembled inserts, which was measured with a calliper. Leg_Gap is the distance between insert's flange surface and the end surface of the yoke part of the platform joints. This distance was also measured with the calliper and is listed in Appendix A, Table A18. The leg length measurement showed that it varies with different legs. As we will see in Chapter 6, this leg length variation affects the accuracy of the platform pose.

3.7 Inverse kinematic model of the Tiara Hexapod

Based on the parameters of slide joint centre at home position F_{li}, slide orientation vector U_i, platform joint centre P_i, leg length L_i and joint vector given in the previous sections, the kinematic model of the Tiara hexapod is shown in Figure 3.12. For a given feasible pose, the machine actuators will drive the six slides moving from their home positions F_{li} to their desired position S_i. Each slide moves distance λ_i along the direction U_i in $\mathrm{B}\text{-}\mathrm{X_B Y_B Z_B}$ system, satisfying the equation

$$\overrightarrow{BS}_i = \overrightarrow{BF_{li}} + \overrightarrow{F_{li}S_i} = \overrightarrow{BF_{li}} + \lambda_i \cdot \overrightarrow{U_i} \tag{3.30}$$

Leg L_i connects joint centre S_i on the slide and joint centre P_i on the platform with the vector relationship:

$$\overrightarrow{S_iP_i} = \overrightarrow{S_iF_{li}} + \overrightarrow{F_{li}P_i} = -\lambda_i \cdot \overrightarrow{U_i} + \overrightarrow{F_{li}P_i} \tag{3.31}$$

Its length can then be expressed as:

$$L_i^2 = \left\|\overrightarrow{S_iP_i}\right\|^2 = \lambda_i^2 - 2\lambda_i \cdot \overrightarrow{U_i} \cdot \overrightarrow{F_{li}P_i} + \left\|\overrightarrow{F_{li}P_i}\right\|^2 \tag{3.32}$$

Rearrange equation (3.32) produces

$$\lambda_i^2 - 2\cdot\overrightarrow{U_i}\cdot\overrightarrow{F_{li}P_i}\cdot\lambda_i + \left\|\overrightarrow{F_{li}P_i}\right\|^2 - L_i^2 = 0 \tag{3.33}$$

This second-order polynomial function of the slide joint centre displacement λ_i is the inverse kinematic model of the Tiara hexapod. A similar formulation can be found in [51][52][53][54]. The smaller one of the two solutions for λ_i that is within the motion range for the slide is the one that is used in the motion control.

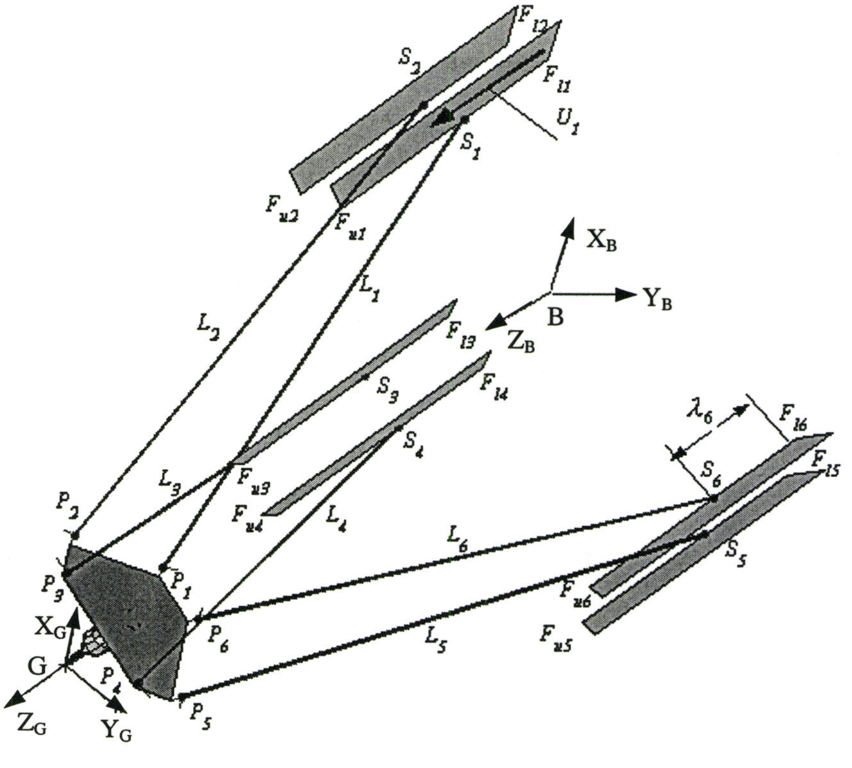


Figure 3.12 Kinematic model of the Tiara hexapod

As mentioned in Chapter 2, for a hexapod machine, the inverse kinematic problem is simple and straightforward. It is easy to solve Equation (3.33) to get the closed-form solution of the inverse kinematics problem, but some constraints must be considered during the solution process. For the nominal inverse kinematic problem, the constraint is the joint pivot angles interference. Because of the special layout of the Tiara hexapod, the Cardan joint pivot angle interference can only occur on the platform joints (i.e. not on the slide joints). The platform joint pivot angle interference has been studied in the previous work of [51] and [52], but some of the design assumptions have been changed with the actual Tiara hexapod. Some definitions are therefore redefined below.

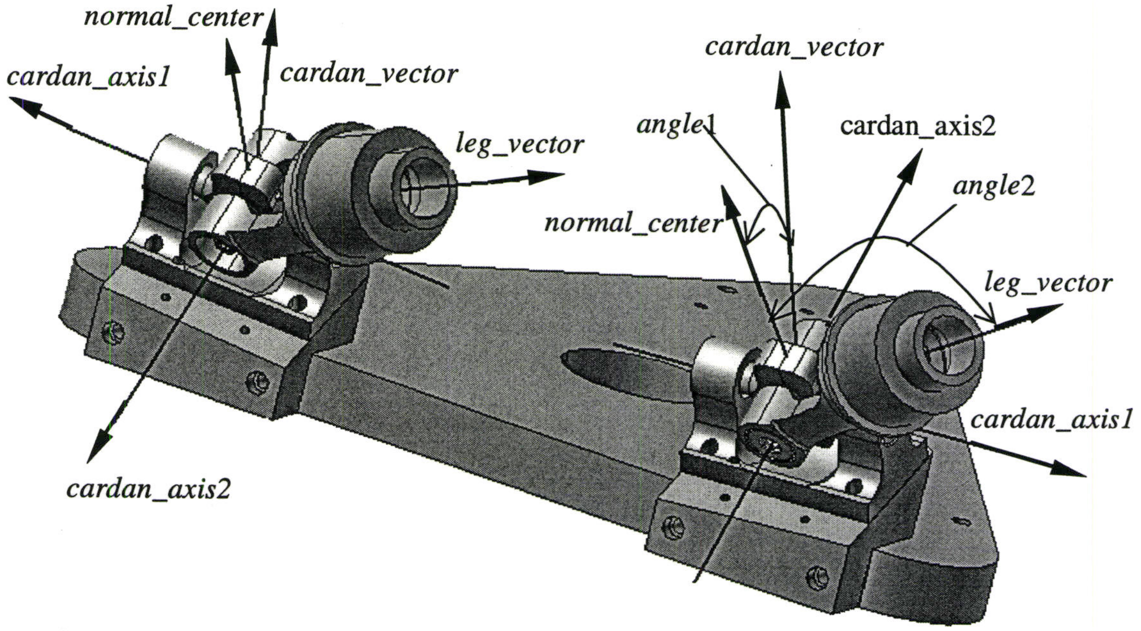


Figure 3.13 Cardan joint pivot angles definition

Two platform joints are shown in Figure 3.13. The joint axis 1 vector, $platform\_joint\_vector_i$ given in section 3.6 is now denoted by *cardan_axis1* with values listed in Appendix A. The leg vector $\overrightarrow{S_iP_i}$ was defined in Equation 3.31. The vector of Cardan joint axis 2 is obtained via Equation 3.34

$$cardan\_axis2_i = \overrightarrow{S_iP_i} \times cardan\_axis1_i \tag{3.34}$$

The normal vector of the joint center block is easily obtained as

$$normal\_center_i = cardan\_axis1_i \times cardan\_axis2_i \tag{3.35}$$

The pivot angle inside the joint shown in Fig. 3.13 is defined as

$$angle1_i = \arccos\left(\frac{\text{Platform\_joint\_center\_vector}_i}{\left\|\text{Platform\_joint\_center\_vector}_i\right\|} \bullet \frac{normal\_center_i}{\left\|normal\_center_i\right\|}\right) \tag{3.36}$$

and

$$angle2_i = \arccos\left(\frac{\overline{S_iP_i}}{\left\|\overline{S_iP_i}\right\|} \bullet \frac{normal\_center_i}{\left\|normal\_center_i\right\|}\right) \tag{3.37}$$

where • indicates the dot product operation. If those two angles satisfy the conditions and the pivot angle diagram given in [51], there will be an interference free solution for the inverse kinematic problem Equation (3.33). For a given pose, if the Equation 3.33 has a solution, this pose is inside of the Tiara hexapod workspace. All those feasible poses form the workspace of the machine.

3.8 Nominal Workspace of the Tiara Hexapod

The workspace of the Tiara hexapod is the collection of the spatial points reachable by the tool tip G on the moving platform [51]. It is determined by the layout of the machine structure, component parameters, spatial geometrical constraints created by the machine structure, and the operational constraints of machine application.

The workspace analysis software developed in [51] and [53] for the Tiara Hexapod, was modified to account for the various leg lengths, and changed joint vectors. The modified software was used to calculate the workspace of the actual Tiara hexapod with the nominal parameters derived above. This workspace is shown in Figures 3.14 and 3.15.

There are some changes in the actual nominal workspace compared with the one presented in the previous study [53]. In Figure 3.14, the XY plane view of the workspace, the maximum X range is about (-800mm, 800mm) which has an increase of 180mm compared with that in [53]. Its maximum Y range is about (-800mm, 900mm), while in [53], this was (–640mm, 700mm). These changes in the current workspace are due to the enlarged radii R_{low} of the frame and R_{plat} of the platform, which result in position changes of the slide joint and platform joint base centres. The nominal workspace in the Z direction (see Figure 3.15) now begins at about 1959mm and ends at about 2590mm. Comparing with the workspace given in [53], the workspace has a 410mm Z increment at the beginning. This is caused by the increased leg length, increased homing distance H_L_i and bigger sizes of the joints. The current workspace has an 80 mm Z decrement at the top end. This is caused by the reduced maximum slide motion distance. This reduction in motion is due to inclusion of the homing distance H_L_i and of the maximum limit switch related distance H_U_i.

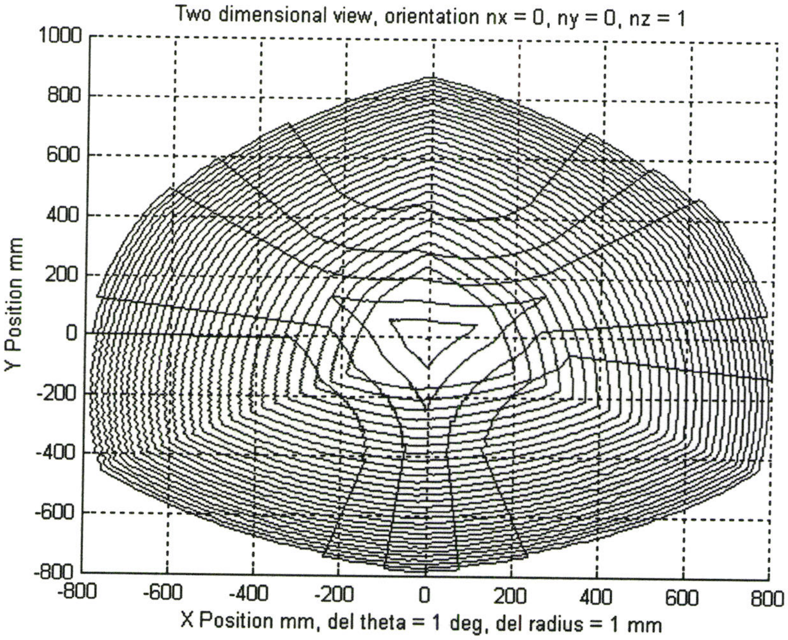


Figure 3.14 Two-D view of the workspace (XY plane)

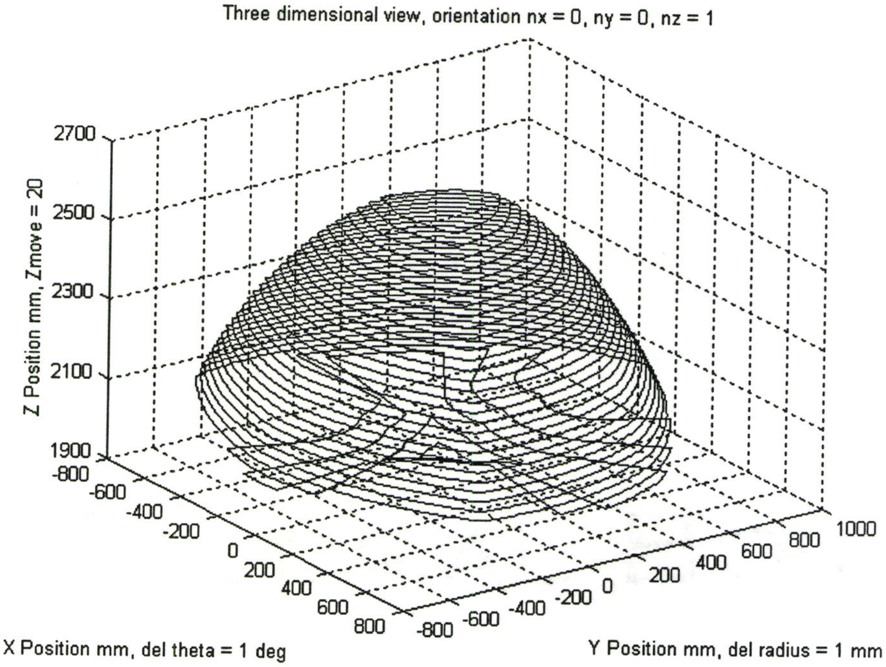


Figure 3.15 Three-D view of the workspace

3.9 Conclusion

To begin this chapter, the actual Tiara hexapod constructed at MMRI was introduced with detailed 3D CAD models for its components. Next, by following the assembly sequences for this machine, its nominal kinematic model is developed. Local coordinate systems are set up on each main component of the machine. The homogenous transformation matrix method is used to create the relationship between those local coordinates systems, and to construct the kinematic model. The nominal parameters required by the machine interpolator, and all the intermediate parameters resulting from each coordinate transformation stage are computed and listed in the Appendix A. The inverse kinematic model of the machine was presented next. The workspace computed with the nominal parameters of the actual Tiara hexapod has changed in all three dimensions, compared with the previously reported results.

Chapter 4

Improvement of the Servo Control System

4.1 Introduction

The original servo control system of the Tiara Hexapod was developed before the final construction of the Tiara hexapod, so most of the functions in the software had not been properly tested on the actual machine. This chapter details the testing and improvement of the servo control system. After introducing the control system background, a description of the servo control system hardware is presented, followed by an explanation of the structure and functions of the software. Emphasis is placed on the solutions to the problems found during the control system testing and improvement procedure. Finally, the main functions of the improved servo control software are summarized.

4.2 Control System Background

Mr. Lacheray originally developed the servo control system of the Tiara hexapod in his Master's thesis [54] and research work [57] at MMRI before March 2002. It was stated in [57] that, by February 2002, the Tiara hexapod control computer had been properly configured to run all the parts of the developed servo control software; and that the PID controllers needed tuning online first once the machine was completely constructed.

However, the test of servo control software integrated with the interpolator developed in Chapter 5 discovered that most functions of the servo control software were not tested on the machine because of time limitations. Phenomena observed and problems found during testing included:

- Platform falling downwards when initializing and stopping the machine.
- Oscillating and false limit switch signals.
- Jogging and homing functions were not developed and tested for the six axes.
- Not enough safety protection functions for amplifier faults, signal generation or interpolation failure and mistaken selection of motion mode, etc.
- Motion functions for the machine that are unsafe: open loop homing H_OL and homing from limit switch to center H_TO_CENTER.

Obviously, only after those problems are fixed, the main functions are developed and well tested, can the online interpolation and performance measurement continue. The remainder of this chapter describes the coorections and improvements that were performed.

4.3 Hardware of the Servo Control System

The physical structure of one drive axis of the Tiara hexapod is shown in Figure 4.1. It includes: ball screw slide with pitch 20mm and repeatability +/-5micron; one

Kollmorgen Silverline brushless series DC motor directly coupled to the ball screw shaft with an aluminum flexible coupling; a 2048-line-per revolution incremental rotary encoder mounted on the back shaft inside the DC motor; and two GXL-N12F micro proximity sensors mounted at both ends of slide; one spring set electric brake installed on the ball screw shaft. Quadrature counting is used to incease the effective encoder resolution by a factor of 4 to 8192 counts per revolution. With this ball screw and encoder the resolution based on the manufacturer's specifications is 1 encoder count=2.4μm.

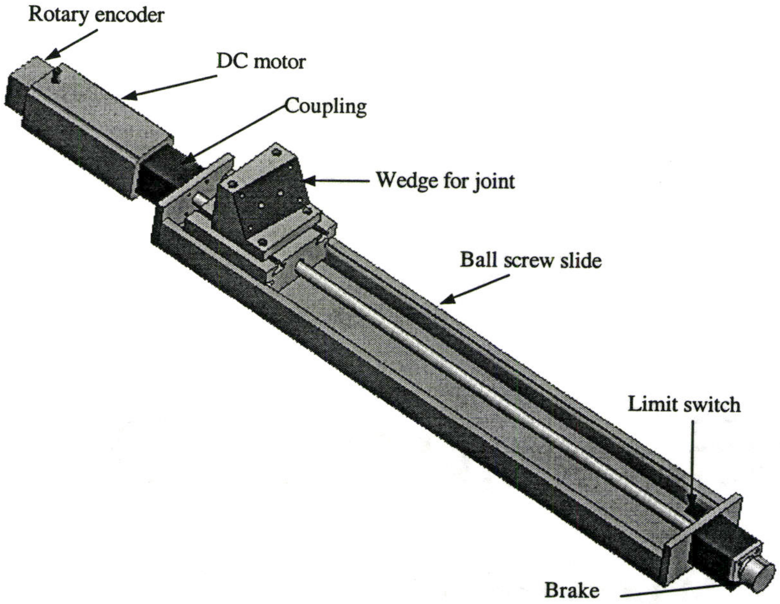


Figure 4.1 Hardware layout of one drive axis of the Tiara hexapod

There are totally six brakes controlled by a TBC-7 brake controller, so that the six ball screws can be clamped or released at the same time. By default, the brakes are enabled and holding the ball screw shafts. When a 5VDC output signal is sent to the brake controller, the brake controller will power on the brakes, the brakes are disabled and the ball screws can rotate freely. The DC motor is driven by B30A40ACF type brushless servo amplifier made by Advanced Motion Controls.

An ISA Bus servo I/O card, STGII-8, manufactured by Servo To Go, Inc. [58] forms the heart of the six axis servo control system. It includes: 8 channels of encoder input, with 24 bit counters; 8 channels of analog output with + 10 V to - 10 V span and 13 bit resolution; 32 bits Digital Input and Output (DIO) configurable in various input and output combinations and Opto-22 compatible; 8 channels of analog input with +/-10V or +/-5V span and 13 bit resolution. The 32 Digital I/O lines are grouped in 8 bit units, and referred to as ports A, B, C and D.

In the original servo control software, ports A, B and the upper four bits of port D were configured as digital inputs. 18 bits out of these 20 bits (i.e. except B7 and D7 bits) were assigned to the maximum limit switch signal, the minimum limit switch signal and amplifier fault signal for the six axes. Port C and the lower four bits of port D were configured as digital outputs with bits C2 to C7 setting the amplifier enable signls and bit D0 enabling/disabling the six brakes. Six out of the eight DAC channels were used to generate and send the control signals to the six amplifiers.

The Servo To Go card and control software are hosted in a PC. The PC is operated under the Real Time Linux operating system to provide the hard real time capabilities that enable the servo control algorithms to execute at at each sampling period of 0.3ms. The power supply and electrical circuit are installed in two control panels. In addition, a portable emergency stop switch is wired to allow the power to the machine to be shut off under emergency situation.

4.4 Servo Control Software of the Tiara Hexapod

4.4.1 Design Philosophy of the Servo Control Software

The servo control software of the Tiara hexapod is based on the OAC philosophy, especially the machine tool OAC designed and implemented by Teltz [48][49] at MMRI. The MMRI OAC is designed as a prototype for an industrial machine tool controller, capable of being applied to any machine tool. It only requires making a relatively small amount of modification to adapt the OAC from a Leblond Turret Lathe controller to a multiple axis robot controller [50].

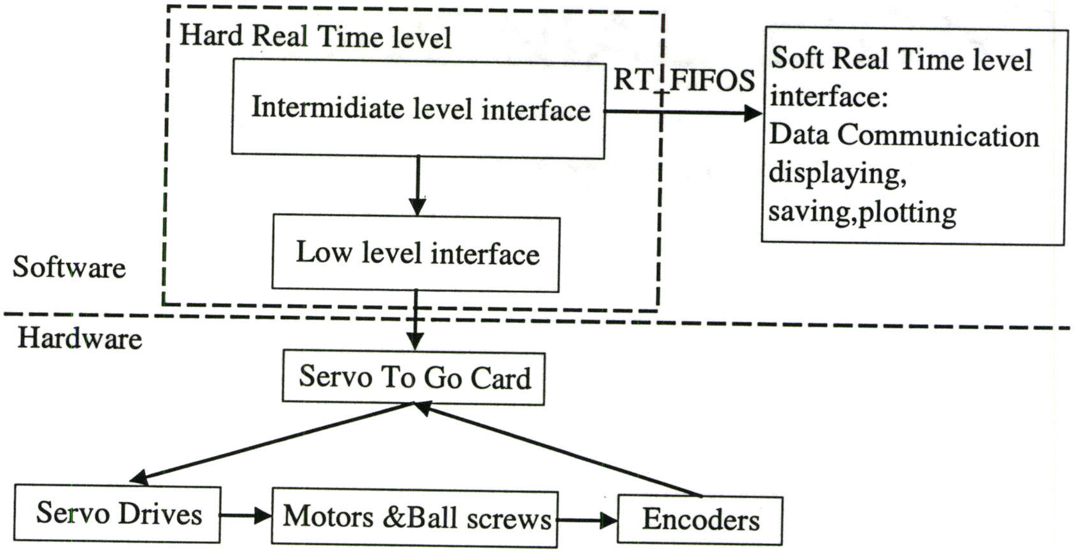
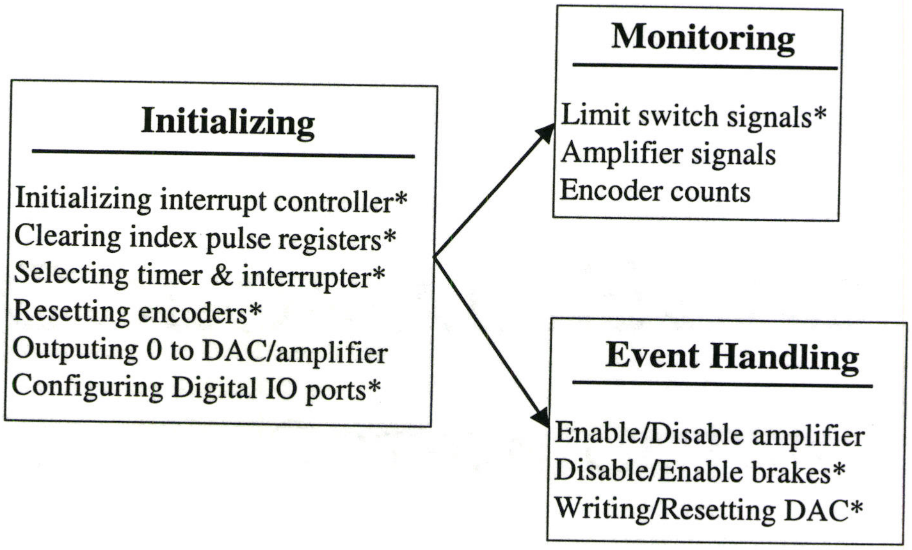


Figure 4.2 Hardware and software structure

With the variable definition, data structure, function prototype, software hierarchy, as well as machine operation modes compliant with MMRI OAC server, the servo control software of the Tiara hexapod mainly addressed the low-level information: machine motion and I/O data of Tiara hexapod [54]. It was written in ANSI C code in modular structure and can be divided into two level as shown in Figure 4.2: Hard Real Time level consisting of low level interface and intermediate level interface, and Soft Real Time level interface for data communication and monitoring.



* indicates the functions developed or improved by the author of this thesis

Figure 4.3 Low level interface functions

4.4.2 The Low Level Interface

The low level interface is shown in Figure 4.3. It provides access to the Servo To Go card. All the functions regarding initialization, status monitoring, and event handling are encapsulated at this level. It represents the interface layer between the intermediate level interface and the hardware. Its functions will now be described.

Initialization functions of the low level interface are encapsulated in the

STG_Initialization module. They were not used in the original servo control software and are modified and used in this thesis research. The problems related to them and solutions to the problems are discussed in section 4.6

Status monitoring functions are realized via the Read_ All_From_STG module. The status monitored by the servo control software is intended to reflect the state of the hexapod machine and its components at each sampling period. The status signals include limit switch signals, amplifier fault signals and axis positions.

Unlike the conventional CNC machine tools and CMM, in the normal operation of the Tiara hexapod except for the homing mode, when the input bit connected to a limit switch becomes low, the intermediate level control software will issue a command for the low level interface to execute the corresponding event handling function: stop all axes immediately. This will casue all axes to be stopped and not only the axis which activates the limit switch.

Amplifier fault signals are produced when the current drawn by the servo amplifiers exceeds a preset limit or when power to the amplifier is disconnected. The relative position of each axis is determined by the incremental encoder counts.

Event handling functions in the low level interface are inside the Write All_to_STG and STG_Inhibit modules. The Write_All_to_STG module works normally in the following sequence: enabling the amplifiers, disabling the brakes and writing the set-point commands to DACs to obtain the voltage signal for the amplifiers. The STG_Inhibit module writes 0 to DAC, disables the amplifiers, enables the brakes and terminates the servo control software, under one of the following normal or abnormal situations:

1) Completing the normal operations the closed loop homing or interpolation motions.

2) There are amplifier fault signals or limit switch signals except in homing mode.

3) One or more amplifiers are disabled due to an axis position outside its safety limits.

4) Mistaken selection of machine CNC operation modes and modes' commands.

5) Failure of signal generation or online interpolation.

The event handling functions related to safety operations did not prevous exist and were implemented in this research. Detail on these functions are presented in section 4.8.

4.4.3 The Intermediate Level Interface

The intermediate level interface communicates with the low level interface functions in order to execute motion commands on the Tiara hexapod. It also communiacates with the other software layers above or parallel to it such as soft real time interface for data communicating, monitoring, saving, plotting, etc. The flowchart of the intermediate level functions including online interpolation and low level interface functions is shown in Figure 4.4. The supervisory thread (M2 in Figure .4.4) loads tool path data for the online interpolation described in the next chapter.

Initialization and Configuration of the Machine and Motion

Following the Servo To Go card initialization (M3), the software configures the machine and motion conditions (M4). Specifically, the software configures the closed loop controllers and motion safety limitations for the individual axes of the machine.

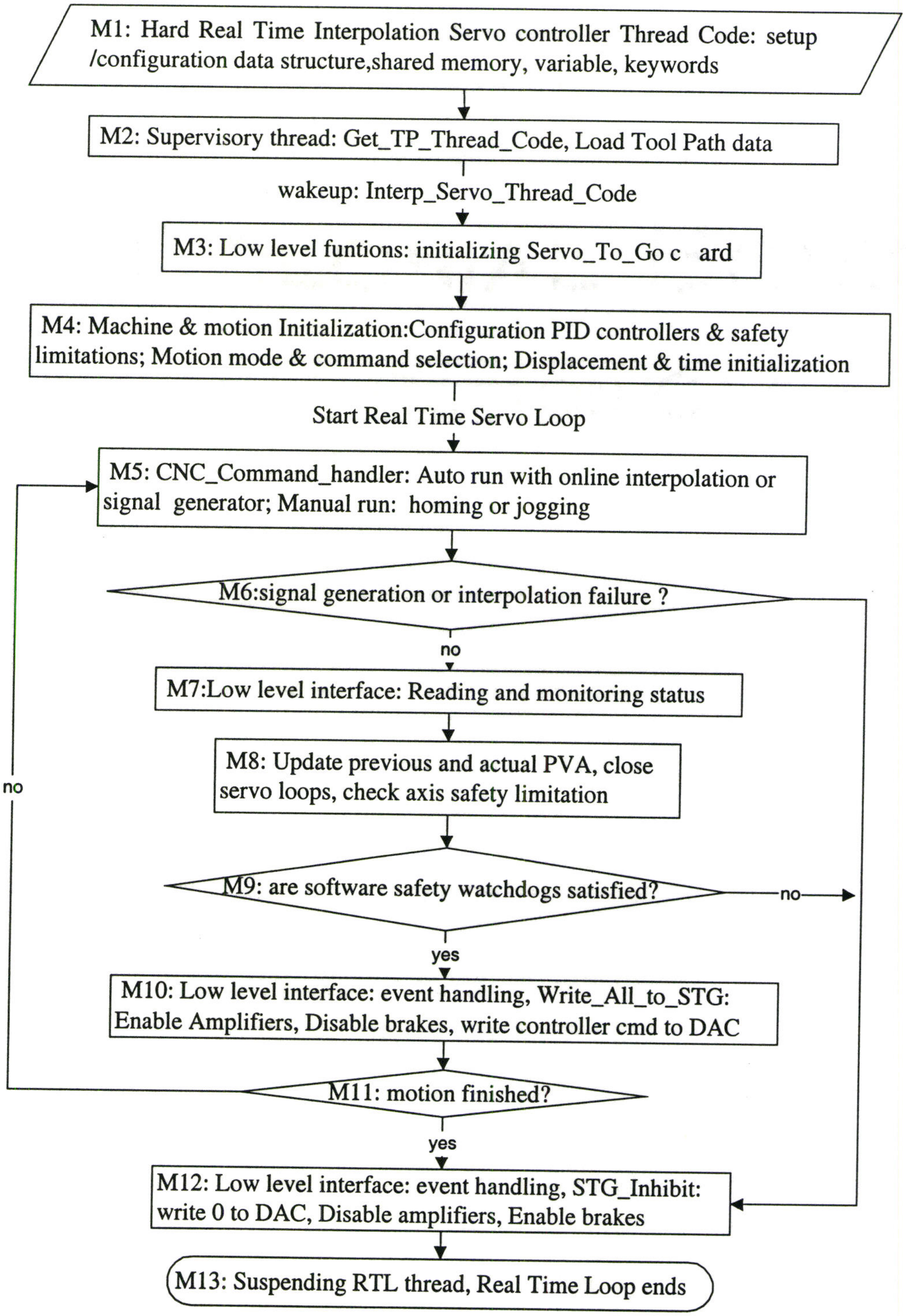


Figure 4.4 Flowchart of the servo control software of the Tiara hexapod

Motion configuration (M4) consists of setting CNC mode and the corresponding mode commands that are illustrated in Figure 4.5. The signal parameters generated by the existing signal generator functions, such as signal type, signal delay time and signal amplitude are also initialized here. Displacement initialization is designed for interpolated motion. It defines how many length blocks inside the tool path are to be moved and the starting number of the blocks. Timing initialization sets up the sampling period for servo loop and records start time of the servo loop.

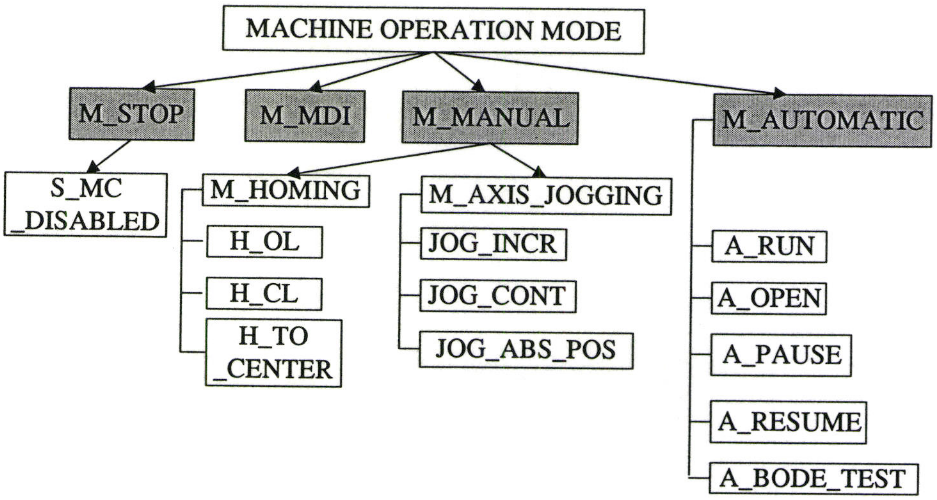


Figure 4.5 Motion modes and commands for Tiara hexapod control system

(Motion modes are shaded in grey)

Real Time Servo Loop - CNC Command Handling

The real time servo loop begins with the CNC command handler (M5). It realizes the motions of the specified CNC mode and its commands. M_MDI, the manual data input mode, is not available currently, so the software has been modified to redirect it to the M_STOP mode, which will inhibit the Servo To Go card and stop the machine. M_AXIS_JOGGING command under M_MANUAL mode was only designed for axis 0 in the original servo control software. Through online testing and software modification, this command was made functional for all of the six axes. The homing command has three types homing motions: open loop homing (H_OL) without PID controller, closed loop homing (H_CL) with PID controller, and homing all axes from limit switch to axis center (H_TO_CENTER). These motions are not realized for the six axes in the original servo control software and their implementation will be discussed in section 4.7. M_AUTOMATIC mode has five commands as shown in Figure 4.5. Currently, the first three commands are not available. In this thesis research, they are redirected to M_STOP mode. The A_BODE_TEST function was designed for controller performance characterization. A_RUN was modified and widely tested in this research. It can run with RAMP, SINE_WAVE, SQUARE_WAVE or DWELL signals generated by the existed signal generator functions for axis position adjustment, demonstration and controller tuning purposes. It also can run with the interpolated setpoint command for machine performance measurement and for actual machining in the future.

Real Time Servo Loop - Close Axis Loop & Safety Checking

The remaining functions (M6 to M10) in Figure 4.4 of the real time servo loop are straightforward. Either an online interpolator or a signal generator is used to produce the setpoint position, velocity and acceleration (PVA) for each axis. Following a series of safety criteria checking, the axis's previous PVA and the actual PVA are updated and computed using the monitored actual axis position. The difference between axis setpoint position and the actual axis position forms the following error, which is in turn used by the PID controllers to compute the control command in DAC bits. After the Write_All_to STG module talks with the Servo To Go card, DC motors will then drive ball screws to realize the desired motion in one sampling period.

Real Time Servo Loop - Termination

The real time servo loop can be terminated in two ways, by a software emergency_stop command or by the real time servo control software itself. The emergency_stop command is written in Real Time Linux syntax. By removing the specified real time threads from the Real Time Linux kernel, it can terminate the real time servo loop and stop the machine at any time.

For the closed loop homing motion function and the interpolation motion function developed in this thesis, the intermediate level interface software can automatically detect the end of the motion, and then terminate the real time servo loop as well as stop the machine. The intermediate level interface modified in this thesis can now terminate the real time servo loop and stop the machine though the low level interface, whenever the safety functions presented in section 4.8 are not satisfied.

4.5 Tuning the PID Controllers of the Tiara Hexapod

As Figure 4.1 shows, each axis employs an incremental encoder for position feedback. A standard feedback control algorithm, Proportional plus Integral plus Derivative (PID) control, is used with each axis servo control system. This is due to its simple implementation and tuning procedure, widespread application and sufficient performance in the control processes in industry [59]. The discrete PID algorithm performed for each axis during each sample period, is approximately as follows:

$$DAC_{output} = K_p E + K_i (IE) + K_d (AV) \tag{4.1}$$

Where K_p is the proportional gain that is used to increase the stiffness. It scales the following error E in the PID algorithm. K_i is the integral gain that helps to reduce the steady state error. It scales the sum of all the error, IE. If the sum of the error IE is greater than the integration limit (eI), then the sum is set to the integration limit. K_d is the derivative gain that is used to increase system damping. It scales the difference in the velocity error (i.e. the current error minus the error from the last sample period) in the PID algorithm.

PID controller tuning is a trial and error procedure, to determine what magnitude of the gains would be required to get the desired response out of each axis and ultimately out of the machine. This manual tuning is difficult and time consuming. Proper tuning is critical for providing precise and smooth motion of the machine. The tuning procedure was implemented jointly by Lacheray, other team members and the author of this thesis. The tuning procedure for the Tiara hexapod can only be used to tune two axes simultaneously with the servo control software shown in Figure 4.4, and a special designed graphical user interface Gain_Interface_Handler. Before tuning can begin, machine and motion configuration must follow the sequence below.

A. Preparation

1) Rotate the six axes ball screws manually or utilizing M_JOGGING command to locate the six carriages approximately at the same position with about 400mm distance to the top end of each slide. This guarantees enough motion space for the two axes to be tuned while the other four axes stay at rest.
2) Disconnect the other four brakes physically in the control panel to ensure they are enabled.

B. Updating related control code files using any text editor

1) In **'../control/servo/servo_test.c'** and in **'Servo_Controller_Thread_Code()'**, 'ENABLE' the amplifiers of the two axes to be tuned and 'DISABLE' the other four axes.
2) In **'../control/cnc/cnc_command.c'** and in **'CNC_Default_Configuration()'**, set motion as M_AUTMATIC mode with A_RUN command. Assign **'SQUARE_WAVE'** to the parameter **'machine->axis[i].signal.type'** for the two axes to be tuned and **'DWELL'**for the other four axes.
3) In **'../control/servo/signal_generator.c'** and in **'Signal_Parameters _Aquisition()'** , initialize the parameter specifications for the SQUARE_WAVE.
4) In **'../control/cnc/hexa_config.c'** and in **'Axis_Safety_Limits_Configuration()'**, set the safety limits in encoder counts for the two axes to be tuned.
5) In **'../control/servo/display_servo.c'** and in **'main()'**, determine which axis' parameters will be displayed by activating the proper part in the code.
6) In **'../control/interface/interface.c'** and in **'Gain_Interface_Handler()'**, select

which two axes will be tuned through the graphical user interface by selecting the parameter **I** and **J** to represent the axes' numbers.

7) In **'../control/cnc/hexa_config.c'** and in **'Axis_Controller_Configuration()'**, select the controller type for each axis by setting the follows: **Controller->type = C_PID; and Controller->PID.type = DISCRETE_PID.** Set the initial values of controller parameters in the same file.
8) Open a **Xterm** window and change the directory to **'../control/servo/'**, compile the updated code files by typing **mymake.**

C. Tuning up procedure

1) Open two **Xterm** windows. In one Bold window, run the tuning up procedure by typing **load_servo_test**. In the other Bold window, type **emergency_stop** command for software emergency stop when needed.
2) Controller parameters can be tuned in the **tune up window** that pops up after running the test while the performance specifications are displayed in the same window.
3) To stop the tuning procedure at any time, close the **tune up window** and press **Enter** key in the **Xterm** window in which **emergency_stop** command is loaded.
4) Change the the gain values for the PID controller for the two axes being tuned in the **tune up window** in the following sequence, and send the entered data by hitting the button **Send** in that window.
   - Set all the gains but the proportional gain K_p to zero. Gradually increase

K_p until the rise time T_{rise}, is minimized with a minimal amount of steady sate error, E_{ss}, and overshoot in an acceptable range.

- Once the rise time is minimized, increase the derivative gain K_d to reduce the overshoot and settling time without over damping the system.
- Gradually increase the integral gain K_i to eliminate the steady state error.

The other four axes are tuned in pairs in the same manner. The tuned PID controller gains for six axes are listed in Table 4.1 along with the response specifications.

Table 4.1 Tuned gains' values and system responses

Axis No	K_p	K_d	K_i	T_{rise} (sec.)	E_{ss} (encoder counts)	Overshoot (encoder counts)
0	0.0016	0.000049	0.0000004	0.162	0	8
1	0.0033	0.000075	0.000003	0.125	0	9
2	0.002	0.000065	0.000003	0.167	0	6
3	0.002	0.00005	0.000003	0.16	0	5
4	0.00128	0.0000378	0.00001	0.153	0	6
5	0.0006	0.000015	0.00001	0.15	0	3

4.6 Improvement of the Servo Control Software

After tuning the servo controller and developing the online interpolator, the original servo control system was tested on the actual machine. However, the online test failed at the very beginning. The problems listed in section 4.2 were found in both the intermediate level interface and low level interface functions of the servo control software. These problems directly affect the safety and normal operation of the Tiara hexapod. They are caused either by the mechanical design or the servo control system of the machine. The problems and solutions related to the servo control system are described below.

4.6.1 Axis Moving Fast in the Wrong Direction

When loading the online interpolation motion control thread into the Real Time Linux kernel and running the machine, occassionaly one or more of the six axes moved very fast with undesired distance and direction, and the machine had to be stopped with the emergency stop switch. Analysis and monitoring of the status data indicated that the encoder counts were incorrect. Further study of the original intermediate level interface of the servo control software determined that the Servo To Go card was not initialized. By adding the low level interface initialization functions to the servo control software, this problem was fixed but another serious problem happens: the platform of the machine falls downwards before it begins to move.

4.6.2 Platform Falling Downwards Before Moving

The problem of platform falling downwards is not only dangerous to the safe operation of the machine and machine user, it also influences the performance and measurement of the interpolation motion. The problem resulted from the low level interface with Servo To Go card, during or after its initialization process.

It is clear that the gravity force drives the platform's downwards motion, while the brakes do not or cannot block the falling movement. As mentioned earlier in section 4.3, the six shaft brakes are controlled by one digital output bit D0. If D0 is set to a low level

(0VDC) the brakes are powered on and their brake functions are disabled, the ball screws can rotate freely. In the initializing functions shown in Figure 4.3, the digital IO are only configured for input and output without any port writing operation. Figure 4.4 (M10) shows that, writing to port D to disable the brakes happens after enabling the amplifier. The platform might fall down at the very moment when the brakes are disabled but motors do not output enough torque to resist the downwards motion. Several tests were carried out to find the solution.

Solution to Platform Falling Downwards

First, the events handling sequence was changed to: enabling amplifiers, writing DAC and disabling brakes. Second, the events handing function Write_All_To_STG was removed from the motion control loop in Figure 4.4. In both cases, the problem was observed again. That means before performing events handling functions, the brakes had been disabled, and D0 bit was low. Monitoring the digital IO bits' values confirmed this conclusion.

The solution procedure had to focus on the Servo To Go card itself and its initialization functions. The Servo To Go card has two 82C55s integrated circuits on board. One is used for digital IO ports A, B and C. The other is used for port D, the CNTRL0 register for interrupt control and the BRDTST register for board test. BRDST is an input and is the power on default for the 82C55. As an output, CNTRL0 defines the interrupt request number. It must be initialized first and works only after port D has been set as either input or output by the direction register D_DIR. The direction of port D can be reset later, but CNTRL0 will be cleared. Generally, after initializing the interrupt controller CNTRL0, the direction registers, ABC_DIR for ports A, B and C, D_DIR for port D, have to be set by the initialization functions to select which IO lines are inputs and which are outputs. All the 32 bits signals of ports A, B, C and D come from one 82C55 and are pulled up by a 10K resistor to +5 volts; so inputs will default to high if a line is broken or not connected. For any port that is configured as an output port, reading will return the current state of the output—whatever you last wrote to it. If D0 is high, after configuring it as output bit and without writing zero to it, it should be high. However, the platform falling downwards indicates that after configuring port D and with no writing operation, D0 becomes low level and enables the brakes. So the problem resulted from the initialization of interrupt controller that uses D_DIR register for the first time.

In the original interrupt controller initialization function, the eight bits of port D were first set as output for configuring CNTRL0 register. Later on, in the following D port initialization function, port D was reset with upper four bits as input and lower four bits as output. Unfortunately, after resetting port D, CNTRL0 that has been cleared by resetting port D was not restored! The solution to the problem then became clear: first set port D as input in the interrupt controller configuration, save the CNTRL0 setting, then reset D with upper four bits as input, lower four bits as output in port D configuration, and finally restore CNTRL0. After implementing this software modification, running the servo control software with setpoint commands provided by the online interpolator or by the signal generator, the platform never fell downwards again before it was commanded to move.

4.6.3 Oscillatory and False Limit Switch Signals

During the procedure for solving the preceding problems, the status of the digital IO was monitored. The signal level of input bits A0, A1 of A port and B1 of B port were found to be always flipping between high (1) and low (0). The A0 and A1 bits are connected with the maximum and minimum/home limit switches of axis 0, and B1 with the maximum limit switch of axis 3. When software detected the flipping of A0, A1 and B1, neither did the carriage of the two axes or external iron/steel objects approach the limit switches' sensing zone, nor were the two motors being inhibited. (The limit switch should send 0VDC to the Servo To Go card and inhibit the amplifier, if it were activated).

Input signals from the limit switches affect the safety operation of the machine and the implementation of the homing functions. The primary reason for the faulty input signals was likely due to one of the following hardware problems: the limit switches, the Servo To Go card, the electrical wiring, signal isolation or electromagnetic disturbance. Two experimental procedures were performed to solve the problem.

1) Manually Activating the Limit Switches

The six legs and the platform were first dismounted from the slide joints for safety. The machine was run with the A_RUN command and a small slope RAMP signal. The limit switch to be tested was manually activated with a metal bar. The activation of each limit switch was indicated by its red LED. The corresponding amplifier was also successfully inhibited with the motor in the dwell condition. The recorded monitoring data displayed the corresponding input bits as 0 except for the A0, A1 and B1. This result indicates the limit switches, their signal feedback wires to the amplifiers and to the Servo To GO card except for the A0, A1 and B1 bits work normally.

2) Testing on a Single Axis System

In order to measure the digital signals related to bits A0, A1 and B1, the 0th axis drive without the brake controller was removed from the machine to set up a single axis experimental setup. First, the above test was repeated on this single axis system. Multimeter measurements indicated the wires connecting A0, A1 and B1 had sent correct signals to the servo card. Second, with this single axis control system running slowly with RAMP signal, using multimeter to measure the A0, A1 and B1bits, the readings were flipping between 5VDC and 0VDC the same as the status information monitored by the software, even though there was no limit switch activated. Finally, to avoid the electromagnetic disturbance influence, the Servo To Go card was tested at midnight when there was neither heavy machine operated near the Tiara hexapod in the lab, nor power supply to the amplifier, DC motor and limit switch of that single axis system. The host PC was the only one to be powered on to support the card, to run the software for monitoring the status of the concerned digital input bits A0, A1, and B1. Multimeter measurement reading of those three bits and the software monitoring status confirmed that these bits workwere faulty, the direct reason is out the scope of this thesis. The solution to the faulty of A0, A1and B1 bits was to reconfigure the input and output ports of the Servo To Go card as: ports B, C and upper four bits of D are input signals, A and lower four bits of D are output signals, skipping C1, B1, A0 and A1 bits. With the single axis system setup, all the reconfigured digital input and output signals were extensively tested and measured. They all worked properly. Based on the new configuration for the input and output ports, the corresponding wires in the control panel of the Tiara hexapod were reconnected. The new digital IO ports' configuration and wiring mapping are listed in Appendix B

4.7 Development of the Homing Function for the Tiara Hexapod

After fixing the above problems, especially obtaining the correct limit switch signals, the homing function in the original servo control software was ready to be tested. Homing allows the machine coordinate system reference to be set. This utility function is very important for hexapod machines, for they have no physical X, Y, Z axes. Any movement of the tool tip depends on the six axes' relative positions to the homing positions.

For the conventional multiple axis CNC or CMM, homing function is achieved by homing each axis individually, usually beginning with Z, followed by Y and finally the X [39]. The original hexapod homing function was designed based on this principle. "The homing procedure of Tiara Hexapod is designed to home in on each axis individually at the first pulse after the micro-proximity sensor located on top of the hexapod structure"[54].

However, the homing function in the original software was designed for axis 0 without the FOR or WHILE loop for other five axes. If simply adding a FOR loop for all the six axes, the homing procedure becomes homing the six axes simultaneously, which is different from the statement in [54]. Moreover, the successful homing function mainly depends on the homing limit switch signals, which have been found incorrect above for axis 0. This means the homing function had not been properly implemented on the actual machine. Therefore, the homing function of the Tiara hexapod had to be tested, modified or redesigned from the beginning based on the single axis system setup.

4.7.1 Homing Function Tests on the Single Axis System

As mentioned in previous sections, there are three types of homing motions: closed loop homing H_CL, open loop homing H_OL and homing all axes from limit switches to axes' center H_TO_CENTER. In the single axis system, the closed loop and open loop homing functions were initially modified for homing six axes simultaneously. One of the six axes is tested individually with the other five axes are not used. The homing procedure consists of the four steps:

1) Move the axis towards the home position (negative direction) until the limit switch is activated.

2) Change movement direction to positive and move away from the limit switch until it is deactivated.

3) Move axis in the positive direction until the encoder index pulse (or marker) is detected.

4) Set this position as the home position. For closed loop homing of all six axes, set the current setpoint position to be the home position and hold this position until all the

home positions are established. For open loop homing, this position cannot be held, so set the encoder counter to 0 and disable the corresponding amplifier.

For the parallel kinematic machine, especially the Tiara hexapod, the open loop homing function is useless and could be regarded as dangerous. For the Tiara Hexapod, all the six brakes are controlled by one output signal. If one or more axes find their home position and disable their amplifiers while other axes keep moving, the brakes for the moving axes have to be disabled for the moving axes, that is, all the brakes are disabled. As a result, the platform has one or more degrees of freedom unconstrained by the homed axes. Under this situation, the homed axes are free to move and will not stay at their home positions due to the force of gravity. The free moving of the homed axes will finally result in the homing procedure failure for the other axes, and possibly move the machine into a singularity, even damaging the machine. Therefore, in the improved servo control software, to prevent unsafe operation, this open loop homing has been redirected to M_STOP mode. Running open loop homing will stop the machine.

The H_TO_CENTER type motion is actually an A_RUN motion with RAMP signal starting from axes' home position or maximum position. It is specifically designed to re-enable the amplifiers and move all the axes to the center positions of their motion strokes, after all the home positions have been reached via the open loop homing function. However, moving all six axes to their center positions can be realized with M_JOGGING or A_RUN motion, so this H_TO_CENTER type motion in the improved servo control software, is also redirected to M_STOP mode.

Based on the single axis experiment results and discussion above, the only homing function potentially useful to the Tiara hexapod is the closed loop homing function. This functon required further development to become useful in practice.

4.7.2 Closed Loop Homing Function Development

With the same homing procedure, the closed loop homing function was tested again on the six axes of the machine simltaneously. For safety, and for convenient measurement of the homing position H_L_i, the legs and platform were not assembled during these tests.

Unfortunately, the closed loop homing function that worked well with the single axis system failed for the six axes system. Fake home limit switch signals were detected in the homing process. When some axes found their home position and dwelled at these positions, at least two of the other axes being homed detected home limit switch signals before they got into the sensing zones of their limit switches. Those axes dwelled at positions that were far way from their real home positions. Since all the digital ports and limit switch have been physically tested individually, they are proven working normally. Two potential causes for the fake home limit switch signals are: poor signal isolation and electromagnetic disturbance inside the control panel.

Electromagnetic disturbance was found inside the control panel that influences the spindle controller. When no power is supplied to the spindle and its controller, the spindle controller's screen is always displaying rotation speed values of the spindle. There is also no optical isolation between the input signal wires and the connectors of the Servo To Go card inside the control panel. The connection between the four 50 pin ribbon cables of the Servo To Go card and the outside wires for limit switches and brake controller are regular terminal blocks. Improving the isolation of the signals and shielding the control components from the electromagnetic disturbance was outside the scope of this thesis research. In this research, software measures were taken to avoid the fake home limit switch signals and realize the closed loop homing function.

Reducing the number of axes being homed simultaneouly is one option available by software. Homing one pair of axes simultaneously with the other two pair of axes dwelling at the current position (start positions for homing or home positions) was tested first. The result indicated that, sometimes, only one axis in the homing pair detects fake home limit switch signal, which usually happens to axis 4 or 5 of the last pair of axes. Based on this experiment result, the final closed loop homing procedure was designed as shown in Figure 4.6. Note that this procedure requires the operator to move each axis carriage into its pre-home zone by eye using M_JOGGING commands. This homing procedure homes one axis at a time with the other axes dwelling at their current positions in a pre-home zone or at the captured home positions. The pre-home zone is in the neighborhood of the home position and was determined through experiment. The pre-home zone begins at tool tip positon of (0,0,1990mm) with all the six carriages locating inside 40mm range from their home positions. With this new proceudre the home position of each axis was found without a fake limit switch signal being detected. The experiment measurements of the home position H_L_i and the measurements for homing repeatability were then completed. These test results are detailed in Chapter 6.

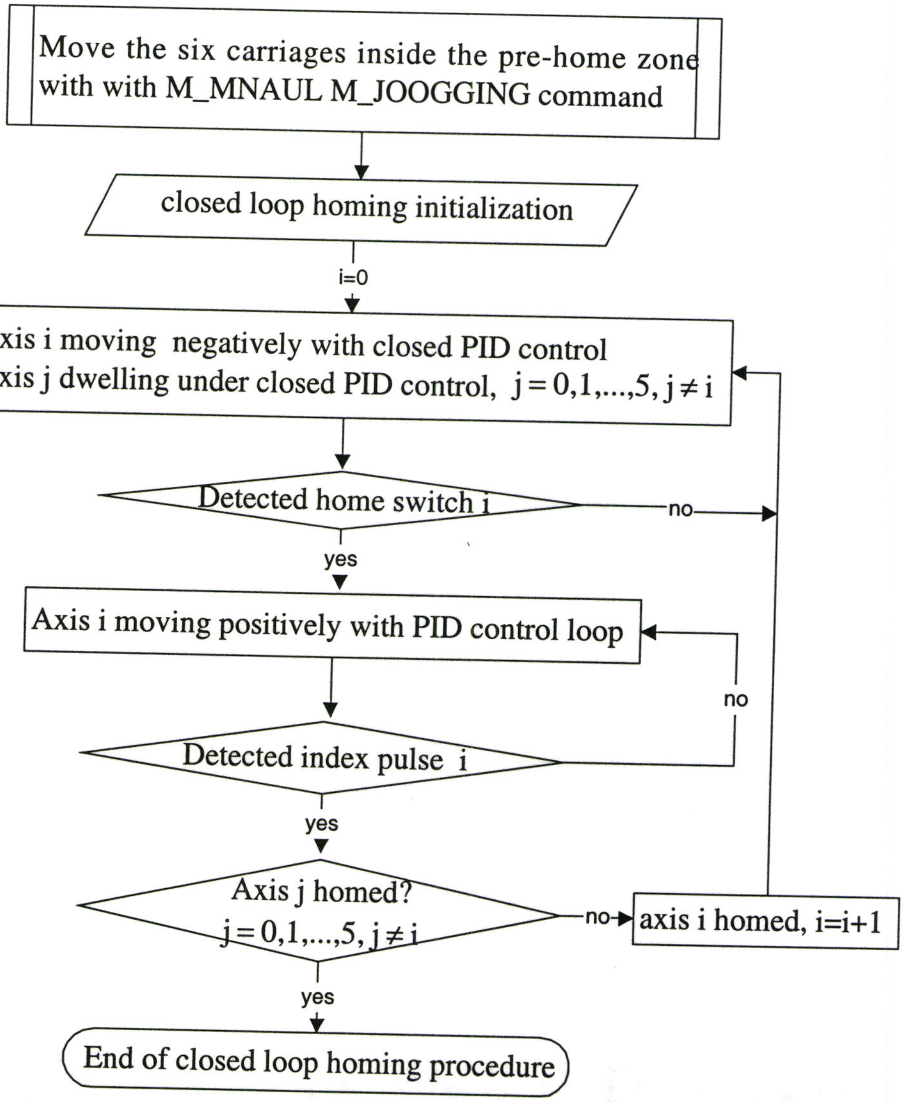


Figure 4.6 Closed loop homing procedure for the Tiara hexapod

4.8 Safety Functions of the Servo Control Software

In addition to the hardware reconfiguration, software improvement and development for the servo control system of the Tiara hexapod in the proceeding sections, a pneumatic cylinder compensator and a series of software watchdog functions were also developed for this machine for safety and normal operation purposes.

With the existing design the platform falls down about 10mm when the machine stops any kind of motions. This is due to the inclined layout of the leg-platform system causing the gravity force and moment to act on axes 2 and 3. During the very short time when the machine stops its motion, the two brakes on axes 2 and 3 can not produce enough reaction torque, so the two carriages of axes 2 and 3 move down more than 10mm before stopping. Software measures such as increasing delay time for disabling the amplifiers after enabling the brakes were not found to be effective.

A mechanical compensator mainly consisting of a pneumatic cylinder and two Cardan joints was suggested as a solution by the author of this thesis. This compensator was designed, manufactured and installed on the machine by another team member, Mr. Simon S.F. Chang. With the help of this compensator, the falling downwards of the platform has been reduced to about 1mm. This compensator made the machine motion measurement presented in Chapter 6 feasible. For installation and adjustment of the pneumatic compensator, a specific brake control function was also developed in the servo control software.

During testing and improving the servo control software, a series of software safety watchdog functions were developed for this machine. The first one prevents machine malfunction due to failure of the signal generator and the online interpolation. The second one responds to the monitoring status returned by the low level interface. If an activated limit switch is detected, except in homing mode, or an amplifier fault is detected, the servo control software will call the low level interface functions to inhibit the servo card, stop all motors and enable the all brakes. The third one stops the machine when one or more amplifiers are disabled due to axes actual position being outside its safety limit in any of the motion modes. The last one prevents the operator from selecting the incorrect CNC mode and command. If the user by mistake sets the unavailable CNC mode such as M_MDI, selects the unsafe CNC command types such as H_OL and H_TO_CENTER or sets one mode with a mismatching command, the servo control software will automatically stop the machine. Those software safety watchdogs make the machine work safely.

4.9 Summary

This chapter outlined the hardware and software structure of the servo control system for the Tiara Hexapod. Problems related to the hardware and software were found and were solved through reconfiguring and rewiring hardware; and modifying and developing software functions. The main motion functions of the servo control system were developed and extensively tested both on a single axis test system and the complete six axis machine. A homing function was realized under closed loop control. This function enables each axis to reach its home position from a pre-home zone. An axis jogging function was developed that moves the six axes simultaneously in absolute, incremental or continuous modes. With this function, the machine can approach the pre-home zone from any location for successful homing, or move out from a singular position if it occurs. Software safety watchdog functions were also developed that allows the machine to operate safey.

Chapter 5

Development of the Online Interpolator

5.1 Introduction

This chapter describes the online implementation of the interpolator for the Tiara hexapod, which converts the tool path to the motion trajectory for the servo controller. The chapter begins by introducing the command generation of the machine, including command generation architecture, motion equations and profiles, definitions of the time parameters, as well as machine dynamic limitations. Next, the motion equation solutions and time parameters are derived in great detail for the single move motion interpolation. Then, the blended move type motion interpolation is presented. Following this, the online interpolator software structure is described. Interpolated motion examples are then shown with discussion. The chapter closes with a summary

5.2 Command Generation for the Tiara Hexapod

5.2.1 Command Generation Architecture

As a CNC machine tool, the Tiara hexapod was designed to machine aluminium parts for the automobile industry. It follows the three-step sequence described by Chou and Yang [41] to manufacture a part from its CAD model: tool path planning, tool path conversion to motion trajectory and motion trajectory realization. Tool path planning is outside the scope of this thesis. Motion trajectory realization deals with controller design and real time implementation, which has been implemented as described in chapter 4.

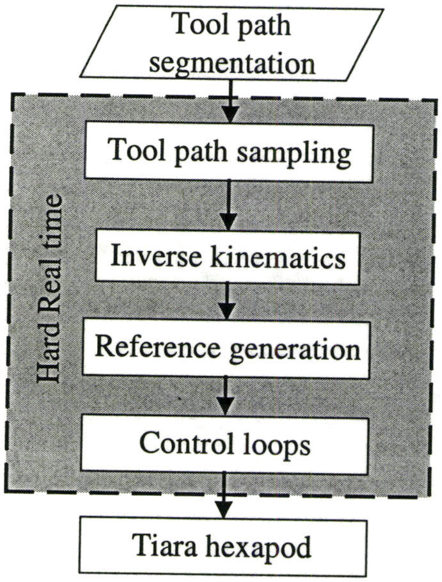


Figure 5.1 Command generation architecture for the Tiara hexapod

The second step, also referred to as command generation, transforms the tool path information generated offline in the first step into the axis motion commands in real time. It involves the kinematics of the coordinated motion, machine dynamics and interpolator design. The key aspect of motion command generation is the process for computing the tool path as a set of curves. This process is known as interpolation and is performed by an interpolator, which is the main concern of this chapter. There are two main categories of interpolators for CNC machine tools. One is Reference Pulse Interpolators; the other is Reference Word or Sampled-Data Interpolators [39]. The online real time command generation of the Tiara hexapod shown in Figure 5.1 is designed based on the Reference Word technique.

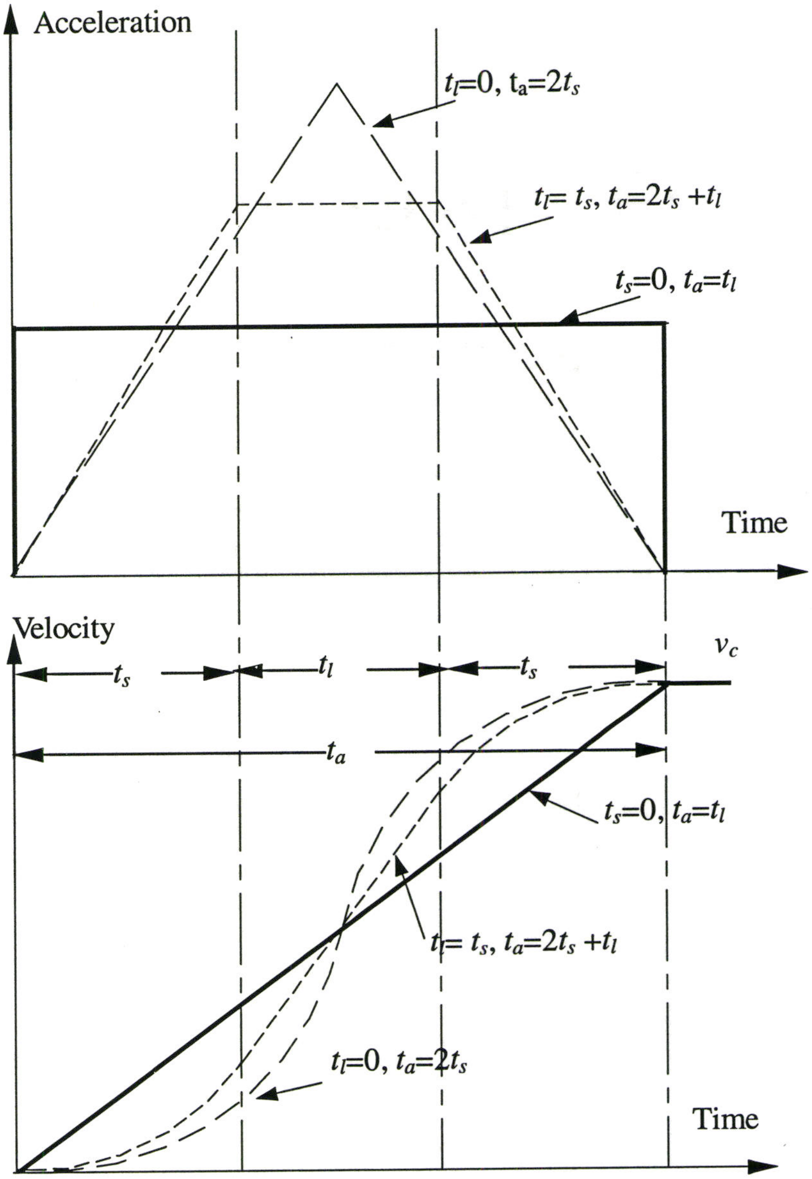
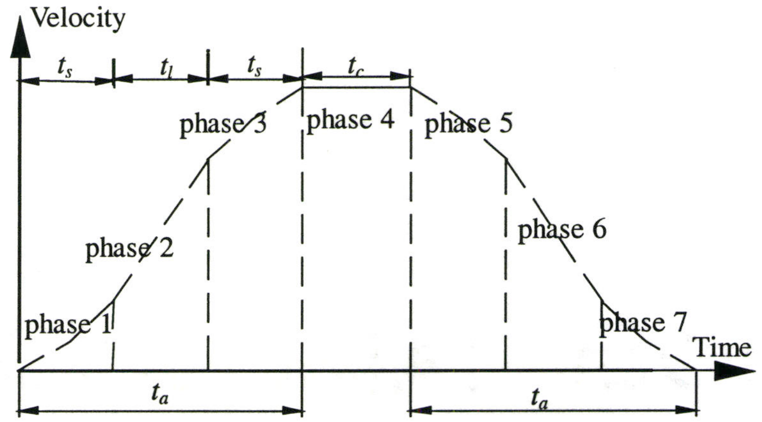
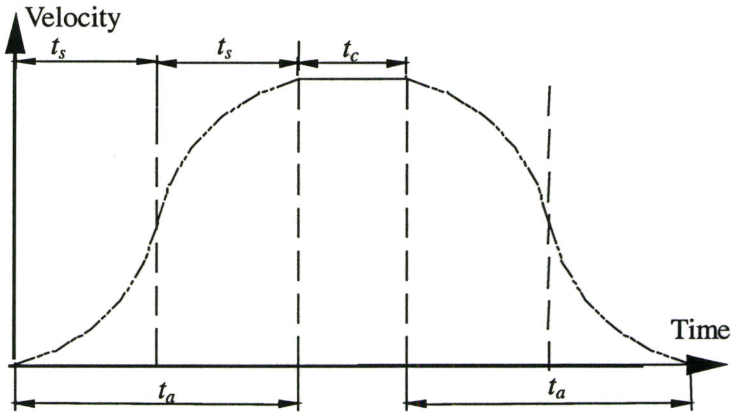


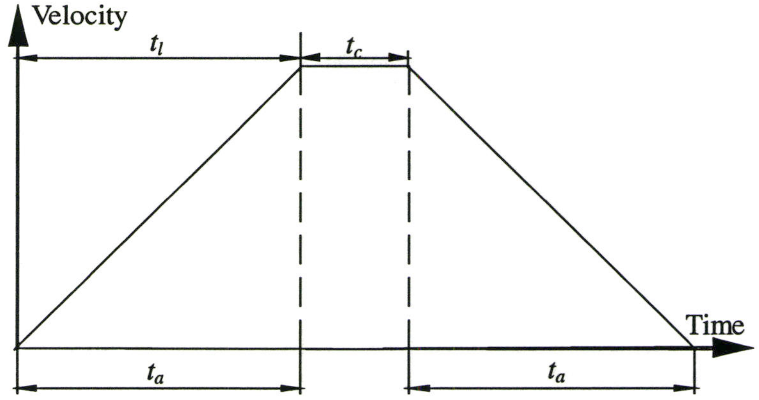
Figure 5.2 Acceleration and velocity profiles [45]



c) Partial S curve velocity profile



b) Pure S curve velocity profile



a) Trapezoidal velocity profile

Figure 5.3 Three possible velocity profiles for a single move

5.2.2 Acceleration Profiles and Time Parameters

Inside the command generation architecture, the online tool path sampling follows the trajectory generation principle of the PMAC [54], to generate full or partial second order velocity profiles with the triangular or trapezoidal acceleration profiles shown in Figure 5.2. Figure 5.2 shows three kinds of possible motion acceleration profiles, namely step, triangular and trapezoidal. They produce three velocity profiless: linear varying velocity, pure S curve velocity consisting of two parabolic curves, and partial S curve velocity composed by two parabolic curves and one straight line segment. The acceleration time parameters labelled in this Figure and in Figure 5.3 are used by the interpolator and are defined as follows.

t_s is the acceleration time for generating the S curve velocity. t_l is the acceleration time for producing linear varying velocity. t_c defined in Figure 5.3 is the duration of constant velocity with zero acceleration. t_a is the total acceleration/deceleration time, while t_m is the total move time formed by t_a and t_c. . The relationship among these time parameters are expressed as

$$\begin{aligned} t_a &= 2t_s + t_l \\ t_m &= t_a + t_c \end{aligned} \tag{5.1}$$

Any single motion must start from a stop and end to a stop, so the total motion time for this kind of single motion consists of an acceleration/deceleration period t_a, constant velocity time period t_c, and a return to a stop period t_a. Three possible velocity profiles of this kind of simple single motion are shown in Figure 5.3. In these three velocity profiles, the velocity starts from zero. After the acceleration period including up to three acceleration phases, it reaches the target velocity v_c and maintains it in phase 4 for a period of time t_c. Then, the velocity returns to zero over the decelration period, which consists of up to three phases, named phase 5, 6 and 7. It can be seen from Figure 5.3 that the velocity varies smoothly for S curve or partial S curve profiles, while the velocity in trapezoidal profile changes discontinuously. This is due to the step changes in the acceleration which have an infinite derivative. The derivative of acceleration is termed jerk.

5.2.3 Kinematics of Motion Equations

The relationships among jerk, acceleration and velocity mentioned above, as well as the resulting position are expressed by the well-known kinematics of motion equations, which can be founded in a similar format in [50][54][60] as

$$J_i(t) = J_{ini,i} \tag{5.2}$$

$$a_i(t) = a_{ini,i} + J_i(t)(t - t_{ini}) \tag{5.3}$$

$$v_i(t) = v_{ini,i} + a_{ini,i}(t - t_{ini}) + J_i(t)(t - t_{ini})^2 / 2 \tag{5.4}$$

$$p_i(t) = p_{ini,i} + v_{ini,i}(t - t_{ini}) + a_{ini,i}(t - t_{ini})^2 / 2 + J_i(t)(t - t_{ini})^3 / 6 \tag{5.5}$$

Where $J_i(t), a_i(t), v_i(t)$ and $p_i(t)$ are the current jerk, acceleration, velocity and position for axis i at given time t. $J_{ini,i}, a_{ini,i}, v_{ini,i}$ and $p_{ini,i}$ are the initial values respectively. t_{ini} is the initial time of the current move time segment.

The velocity profiles shown in Figure 5.3 are the solutions of Equation 5.4 for different jerk and acceleration profiles. For a single move motion of the tool tip, the total solutions of the motion equations for axis 0 are shown in Figure 5.4. In this single move motion with piecewise constant jerk profile, the acceleration is integrated as a trapezoidal profile. Consequently, the velocity profile concistss of partial S curves. The resulting moving position profile is a cubic curve. The online interpolator for the Tiara hexapod controls the jerk, acceleration and velocity of each axis in this way so that the tool tip on the platform can move along an interpolated path with smoothly changing velocity, avoiding the sudden changes in position and velocity that can cause a mechanical component to wear and reduce the machining accuracy.

5.2.4 Machine Dynamic Limitations and Motion parameters

For any motion interpolation, the motion parameters include motion distance, motion time or motion velocity. In CNC machining, the motion velocity or the tool tip feedrate v_0 is usually defined by the user. To achieve this feedrate in a certain period of time, the jerk and acceleration values in Equations 5.2 and 5.3 might exceed the values the machine can reach, which will cause wear or damage to the machine components. Chou and Yang in [41] pointed out that the existence of jerk in a mechanical system causes vibration. In CNC machining, jerk should be kept as small as possible for achieving desired machining accuracy and long machine life. The user therefore has to define the machine dynamic limitations, i.e. the peak values for jerk, acceleration and velocity for the machine. According to system limitation of the Tiara hexapod, these values were defined in [54] as $jerk_{peak} = 400000\text{mm/s}^3$, $a_{peak} = 10000\text{mm/s}^2$ and $v_{peak} = 600\text{mm/s}$

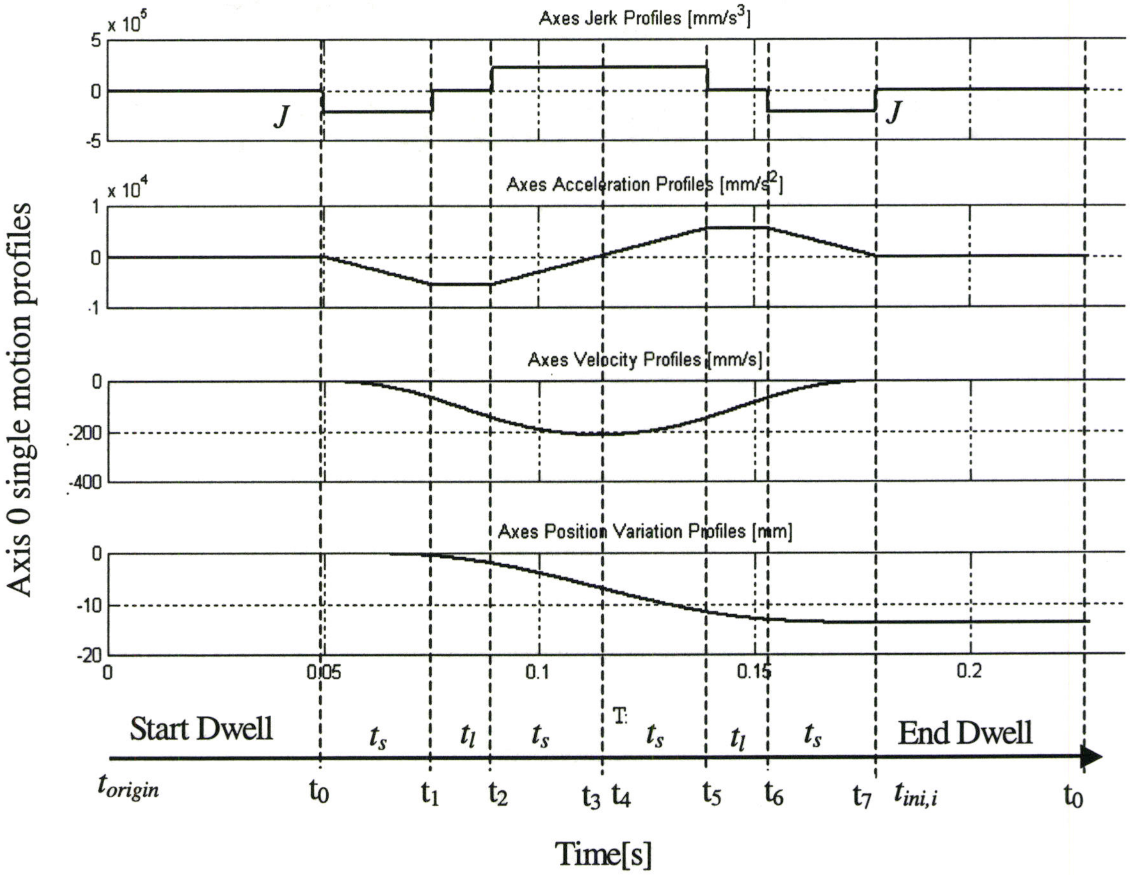


Figure 5.4 Axis 0 single motion profiles

The motion parameters are directly related to the specified tool path. The tool path of the Tiara hexapod consists of a series of short linear segments or blocks. Two vectors $G=[G_x,G_y,G_z]^T$ and $n=[n_x,n_y,n_z]^T,\|n\|=1$, which were defined in Chapter 3 as the tool tip's position and the platform orientation, define the start and end of each move block. The distance of each move block is computed with the move start and end positions' vectors as

$$\Delta G=\|G_{end}-G_{start}\| \tag{5.6}$$

With the specified feed rate v_0 for each move block, the total move time t_m for each block can be easily computed with Equation 5.7

$$t_m = \frac{\Delta G}{v_0} \tag{5.7}$$

From inverse kinematics model Equation 3.33, it is known that when tool tip is located at the start pose G_{start} and n_{start}, one of the six axes of the Tiara hexapod will stay at $\lambda_i(G_{start}, n_{start})$ position on its axis, while for the end pose G_{end} and n_{end}, each axis will locate at $\lambda_i(G_{end}, n_{end})$. In order to move the tool tip from G_{start} to G_{end} in t_m time, the six axes of the machine have to move a distance $p_{move,i}$ simultaneously from $\lambda_i(G_{start}, n_{start})$ to $\lambda_i(G_{end}, n_{end})$ within time period t_m.

$$p_{move,i} = \lambda_i(G_{end}, n_{start}) - \lambda_i(G_{start}, n_{start}), i = 0,1,\ldots,5 \tag{5.8}$$

The time t_m and the inverse kinematics model guarantee that the coordinated motion of the six axes will produce the desired tool tip motion.

The five-degree-of-freedom tool tip motion along the specified tool path block now has been transferred to a one-degree-of-freedom motion of six individual axes. For each linear tool path block, the inverse kinematics solution is applied twice: once at the start and once at the end pose of this block. The online interpolation then begins to deals with individual axis motion: determining the jerk, acceleration, velocity and the corresponding time parameters to smoothly complete the motion distance $p_{move,i}$ during the time period t_m.

5.3 Single Move Motion Interpolation

5.3.1 Motion Equation Solutions and Motion Profiles

A single move motion is the simplest type of motion. It starts with zero velocity and accelerates within time t_a to the target velocity $v_{c,i}$, then keeps this velocity for a duration t_c (t_c can be 0), and finally returns to a stop within a time period of t_a. This kind of motion is useful for diagnosing and measuring the machine performance. Figure 5.4 shows typical single move motion profiles with one time period *block[id].t_start* for dwell motion added at the beginning and end of the motion for measurement purpose. The dwell motion period at the beginning of the block is defined as phase 0 or Start Dwell, while the one at the end of the motion is defined as phase 8 or End Dwell, to comply with the phases' definition from Figure 5.3. Therefore, there are totally nine motion phases for this single move. The constant velocity period $t_c = 0$.The start time and time duration for each phase are labelled along the time axis in Figure 5.4. With the move time given by Equation 5.7 and the distance by Equation 5.8, the target velocity for this move is obtained by

$$v_{c,i} = \frac{p_{move,i}}{t_m}, i = 0,1,...,5 \tag{5.9}$$

The total moving time for this move is $t_m + t_a = 2t_s + t_l + t_c + 2t_s + t_l$. In order to determine the motion time in different phases of this single move, the kinematics equations 5.2-5.5 are applied to each of the motion phases of Figure 5.4 in the Tables

5.1-5.5.

The start time of the tool path motion is named the origin time t_{origin}. If the current single move is the first block of tool path, then initial time of the phase 0 is $t_{ini,i} = t_{origin}$, $\mathrm{t0} = t_{origin} + \mathrm{block[id].t\_start}$. Otherwise, $t_{ini,i} = \mathrm{t7}$, $\mathrm{t0} = \mathrm{t7} + \mathrm{block[id].t\_start}$. Under both cases, the initial velocity and position are set as the end velocity and position of the previous single move as $v_{c,i,prev}$ and $p_{disp,i,prev}$. For single move motion $v_{c,i,prev}$ is always zero, while $p_{ini,i}$ equals zero only when starting from the home position. So the initial values are $J_{ini,i} = a_{ini,i} = 0, v_{ini,i} = v_{c,i,prev} = 0, p_{ini,i} = p_{disp,i,prev}$. Beginning from phase 0, the motion functions and final values for the first five phases are expressed in Tables 5.1 to 5.5.

Table 5.1 Phase 0 Start Dwell $t \in [t_{ini,i} \quad \mathrm{t0}]$

Motion profiles	Final values of the phase
$J(t) = 0$	$J(\mathrm{t0}) = 0$
$a(t) = 0$	$a(\mathrm{t0}) = 0$
$v(t) = v_{c,i,prev} = 0$	$v(\mathrm{t0}) = v_{c,i,prev} = 0$
$p(t) = p_{ini,i} + v_{c,i,prev}(t - t_{ini,i})$	$p(\mathrm{t1}) = p_{ini,i}$

Table 5.2 Phase 1 FIRST_TS_ACC_SUB, $t \in [\mathrm{t0} \quad \mathrm{t1}], \mathrm{t1} = \mathrm{t0} + t_s$

Motion profiles	Final values of the phase
$J(t) = J$	$J(\mathrm{t1}) = J$
$a(t) = Jt$	$a(\mathrm{t1}) = Jt_s$
$v(t) = v_{c,i,prev} + \frac{1}{2} J \cdot (t - \mathrm{t0})^2$	$v(\mathrm{t1}) = v_{c,i,prev} + \frac{1}{2} J \cdot t_s^{\ 2}$
$p(t) = p(\mathrm{t0}) + v_{c,i,prev}(t - \mathrm{t0}) + \frac{1}{6} J \cdot (t - \mathrm{t0})^3$	$p(\mathrm{t1}) = p(\mathrm{t0}) + v_{c,i,prev} t_s + \frac{1}{6} J \cdot t_s^{\ 3}$

Table 5.3 Phase 2 TL_ACC_SUB $t \in [\mathrm{t1}\ \ \mathrm{t2}], \mathrm{t2} = \mathrm{t0} + t_s + t_l$

Motion profiles	Final values of the phase
$J(t) = 0$	$J(\mathrm{t2}) = 0$
$a(t) = Jt_s$	$a(\mathrm{t2}) = Jt_s$
$v(t) = v_{c,i,prev} + \frac{1}{2}J \cdot t_s^{\ 2} + Jt_s(t - \mathrm{t1})$	$v(\mathrm{t2}) = v_{c,i,prev} + \frac{1}{2}J \cdot t_s^{\ 2} + Jt_s t_l$
$p(t) = p(\mathrm{t1}) + v_{c,i,prev}(t - \mathrm{t1})$ $+\frac{1}{2}Jt_s^{\ 2}(t - \mathrm{t1}) + \frac{1}{2}Jt_s(t - \mathrm{t1})^2$	$p(\mathrm{t2}) = p(\mathrm{t1}) + v_{c,i,prev}(t_s + t_l) + \frac{1}{6}J \cdot t_s^{\ 3}$ $+\frac{1}{2}Jt_s^{\ 2}t_l + \frac{1}{2}Jt_s t_l^2$

Table 5.4 Phase 3 SECOND_TS_ACC_SUB $t \in [\mathrm{t2}\ \ \mathrm{t3}], \mathrm{t3} = \mathrm{t0} + 2t_s + t_l$

Motion profiles	Final values of the phase
$J(t) = -J$	$J(\mathrm{t3}) = -J$
$a(t) = Jt_s - J(t - \mathrm{t2})$	$a(\mathrm{t3}) = 0$
$v(t) = v_{c,i,prev} + \frac{1}{2}J \cdot t_s^{\ 2} + Jt_s t_l$ $+ Jt_s(t - \mathrm{t2}) - \frac{1}{2}J(t - \mathrm{t2})^2$	$v(\mathrm{t3}) = v_{c,i,prev} + Jt_s(t_s + t_l) = v_{c,i}$
$p(t) = p(\mathrm{t2}) + v_{c,i,prev}(t - \mathrm{t2}) + (\frac{1}{2}J \cdot t_s^{\ 2} + Jt_s t_l) \cdot$ $(t - \mathrm{t2}) + \frac{1}{2}Jt_s(t - \mathrm{t2})^2 - \frac{1}{6}J(t - \mathrm{t2})^3$	$p(\mathrm{t3}) = p(\mathrm{t2}) + v_{c,i,prev}t_s + \frac{5}{6}J \cdot t_s^{\ 3}$ $+ Jt_s^{\ 2}t_l$

Table 5.5 Phase 4 CST_VEL_SUB $t \in [\mathrm{t3}\ \ \mathrm{t4}], \mathrm{t4} = \mathrm{t0} + t_a + t_c$

Motion profiles	Final values of the phase
$J(t) = 0$	$J(\mathrm{t4}) = 0$
$a(t) = 0$	$a(\mathrm{t4}) = 0$
$v(t) = v_{c,i}$	$v(\mathrm{t4}) = v_{c,i}$
$p(t) = p(\mathrm{t3}) + v_{c,i}(t - \mathrm{t3})$	$p(\mathrm{t4}) = p(\mathrm{t3}) + v_{c,i}t_c$

Based on the above derivations, the relationship of constant jerk and target velocity is constructed from phase 3 in Table 5.4 as

$$J = \frac{v_{c,i} - v_{c,i,prev}}{t_s(t_s + t_l)} \tag{5.10}$$

The constant acceleration is obtained from phase 2 in Table 5.3 as

$$a = \frac{v_{c,i} - v_{c,i,prev}}{(t_s + t_l)} \tag{5.11}$$

The moving distance during the acceleration period is obtained at the end of phase 3 as

$$p_{acc} = p(\text{t3}) = \frac{1}{2}(v_{c,i,prev} + v_{c,i})(2t_s + t_l) = \frac{1}{2}(v_{c,i,prev} + v_{c,i})t_a \tag{5.12}$$

While the moving distance during the time period t_m is

$$p(\text{t4}) = p_{acc} + v_{c,i}t_c \tag{5.13}$$

For a single move, the original velocity equals to the previous velocity, $v_{c,i,prev} = 0$, so the jerk becomes $J = \frac{v_{c,i}}{t_s(t_s + t_l)}$. The moving distance during acceleration/deceleration period becomes $p(t_a) = \frac{1}{2}v_{c,i}t_a$

The remaining phases of the single move motion, phase 5, FIRST_TS_DEC_SUB, with time $t \in [\text{t4 t5}]$, $\text{t5} = \text{t4} + t_s$; phase 6, TL_DEC_SUB, with time $t \in [\text{t5 t6}]$, $\text{t6} = \text{t4} + t_s + t_L$; phase 7 SECOND_TS_DEC_SUB, with $t \in [\text{t6 t7}]$, $\text{t7} = \text{t4} + 2t_s + t_L$ form a return to a stop period, which is also a deceleration period. During that period, the acceleration time parameters are same as those in phase 1 to 3 with similar motion equations. The previous target velocity $v_{c,i,prev}$ is updated as velocity $v_{c,i}$ of phase 4, t0 is replaced by t4, and the target velocity is 0. Plugging the previous target velocity and current target velocity into Equation 5.10, the jerk in the return to stop period of a single move is opposite to that in acceleration period.

$$J_{return\_to\_stop} = \frac{0 - v_{c,i}}{t_s(t_s + t_l)} \tag{5.14}$$

The moving distance in Equation 5.12 during this return to a stop period is always

$$p_{return\_to\_stop} = \frac{1}{2} v_{c,i} t_a \tag{5.15}$$

For a single move starting and ending at a stop, the total moving distance $p_{move,i}$ reached at the end of phase 7 is

$$\begin{aligned} p(\mathrm{t}7) &= p(\mathrm{t}0 + t_m + t_a) = p_{acc/dec} + v_{c,i}t_c + p_{return\_to\_stop} \\ &= \frac{1}{2}v_{c,i}t_a + v_{c,i}t_c + \frac{1}{2}v_{c,i}t_a = v_{c,i}(t_a + t_c) = v_{c,i}t_m \end{aligned} \tag{5.16}$$

Equation 5.16 has verified the Equation 5.9. It indicates that, for a single move with the defined target velocity, the motion is exactly finished during the $t_m + t_a$ time period. The motion distance consists of three parts, acceleration motion distance p_{acc}, constant velocity motion distance $v_{c,i}t_c$ and return to a stop or deceleration motion distance $p_{return\_to\_stop}$.

5.3.2 Determination of Acceleration Time Parameters

For a given length block with specified tool tip feedrate, the move time t_m is identical for the six axes, so the time parameters t_s, t_l and t_c have to be same for all axes. They are determined according to the machine dynamic limitations on jerk and acceleration, and the maximum target velocity among the six axes during that move.

From the target velocity obtained at the end of phase 3 with Equation 5.10, it can be found that, the maximum target velocity among the six axes is achieved or reduced to zero in a minimum time of $2t_s$ given the maximum jerk J_{peak}. t_s is therefore determined by

$$t_s = \sqrt{\frac{\max(\left|v_{c,i} - v_{c,i,prev}\right|, \left|0 - v_{c,i}\right|,)}{J_{peak}}}, i = 0,1,\ldots,5 \tag{5.17}$$

The motion is further constrained by the peak acceleration. This value and Equation 5.11 may be used to determine the linear varying velocity acceleration time t_l. The maximum acceleration can be achieved at the end of phase 1, with the maximum jerk J_{peak} held constant over the time period $t_{s,peak}$

$$t_{s,peak} = \frac{a_{peak}}{J_{peak}} \tag{5.18}$$

If $t_s > t_{s,peak}$ for a axis, then the acceleration obtained at the end of phase 1 for that axis will exceed the preset maximum acceleration $a(t_s) = J_{peak} t_s > a_{peak}$. If this happens then the acceleration for all the axes has to be decreased in proportion so that no axis acceleration exceeds this limit. Therefore, the S cerve velocity acceleration time computed by Equation 5.17 must be reduced to $t_s = t_{s,peak}$, and the acceleration at the end of phase 1 for that axis has to decrease to $a(t_s) = J_{peak} t_{s,jerk} = a_{peak}$.

Consequently, the target velocity cannot be achieved or reduced to 0 within two period of t_s, so a period of constant acceleration time t_l must be added as follows

$$\text{If } t_s > t_{s,peak}, \text{ then } t_s = t_{s,peak}, \text{and } t_l = \frac{\max(\left|v_{c,i} - v_{c,i,prev}\right|, \left|0 - v_{c,i}\right|,)}{J_{peak} t_s} - t_s \tag{5.19}$$

If $t_s \le t_{s,peak}$, the acceleration obtained at the end of phase 1 for any axis is less than the preset maximum acceleration $a(t_s) = J_{peak} t_s \le a_{peak} = J_{peak} t_{s,peak}$, the maximum target velocity is achieved within the two t_s perods defined by Equation 5.17. In this case, no linear varying velocity acceleration time is needed, so

$$\text{If } t_s \le t_{s,peak}, \quad t_l = 0 \tag{5.20}$$

The total acceleration time is $t_a = 2t_s + t_l$ in all situations.

5.3.3 Constant Velocity Time and Short Moves

The total acceleration time t_a is the minimum time for an individual move or a blended move to reach the target velocity. If the distance of a single move is long enough and feedrate specified is not too large, the move time t_m is usually greater than the acceleration time t_a. During this move, there is a period of time t_c with constant target velocity $v_{c,i}$. t_c is obtained from Equation 5.1 as

$$t_c = t_m - t_a \tag{5.21}$$

However, if the move distance is so short that the move time t_m is less than the acceleration time t_a, the move cannot reach its target velocity and it will spend its entire time in acceleration (yielding a triangular rather than trapezoidal acceleration profile). The equations for determining the acceleration time parameters are invalid because of the unavailable target velocity. The acceleration time parameters have to be adjusted in the following way:

$$\text{If } t_a > t_m \text{, then } t_a = t_m \text{, and } t_c = 0 \tag{5.22}$$

Furthermore, the minimum time for such a move is $2t_s$ therfore

$$\text{If } t_s < t_{s,peak} \text{, then } t_s = \frac{1}{2}t_a = \frac{1}{2}t_m \text{, and } t_l = 0 \tag{5.23}$$

Otherwise,

$$\text{If } t_s > t_{s,peak} \text{, then } t_s = t_{s,peak} \text{, and } t_l = t_a - 2t_s \tag{5.24}$$

Under this case, the target velocity is not reached, but the move still keeps on the tool path.

5.4 Blended Move Motion Interpolation

According to the PMAC definition in [45], for the linear blended move, an axis moves toward the target position at a designated speed, accelerating to and decelerating from this speed in a controlled fashion. If more than one move is specified in succession with no pause in between, the first move will blend into the second with the same type of controlled acceleration as is done to and from a stop.

The profiles for a two block blended move obtained with the online interpolator are shown in Figure 5.5. The doted curves indicated the single move of first block shown in Figure 5.4. In a single move, after the motion reaching the target velocity at time point t_m, a return to a stop period will begin to decrease the acceleration and velocity to 0 and finish the defined move distance $p_{move,i}$. For blended moves, there is no pause between the success motion blocks. At the point the current block for a single move would start to decelerate to a stop the blending of motion begins. At the time point t_m, the current block's position and velocity are defined as the initial values for the next blended block as $p_{ini,i}$ and $v_{c,i}$, $_{prev}$ as shown in Figure 5.5.

The blended move begins with an acceleration period in the same way as a single move, from a nonzero initial velocity $v_{c,i,prev}$ at $p_{ini,i}$ to its target velocity $v_{c,i}$. The subsequent move blocks are blended in the same way, except the last move needs a return to a stop period to finish the entire tool path motion. The total moving distance in a blended move is the summation of the length of the blocks computed by Equation 5.8 as follows

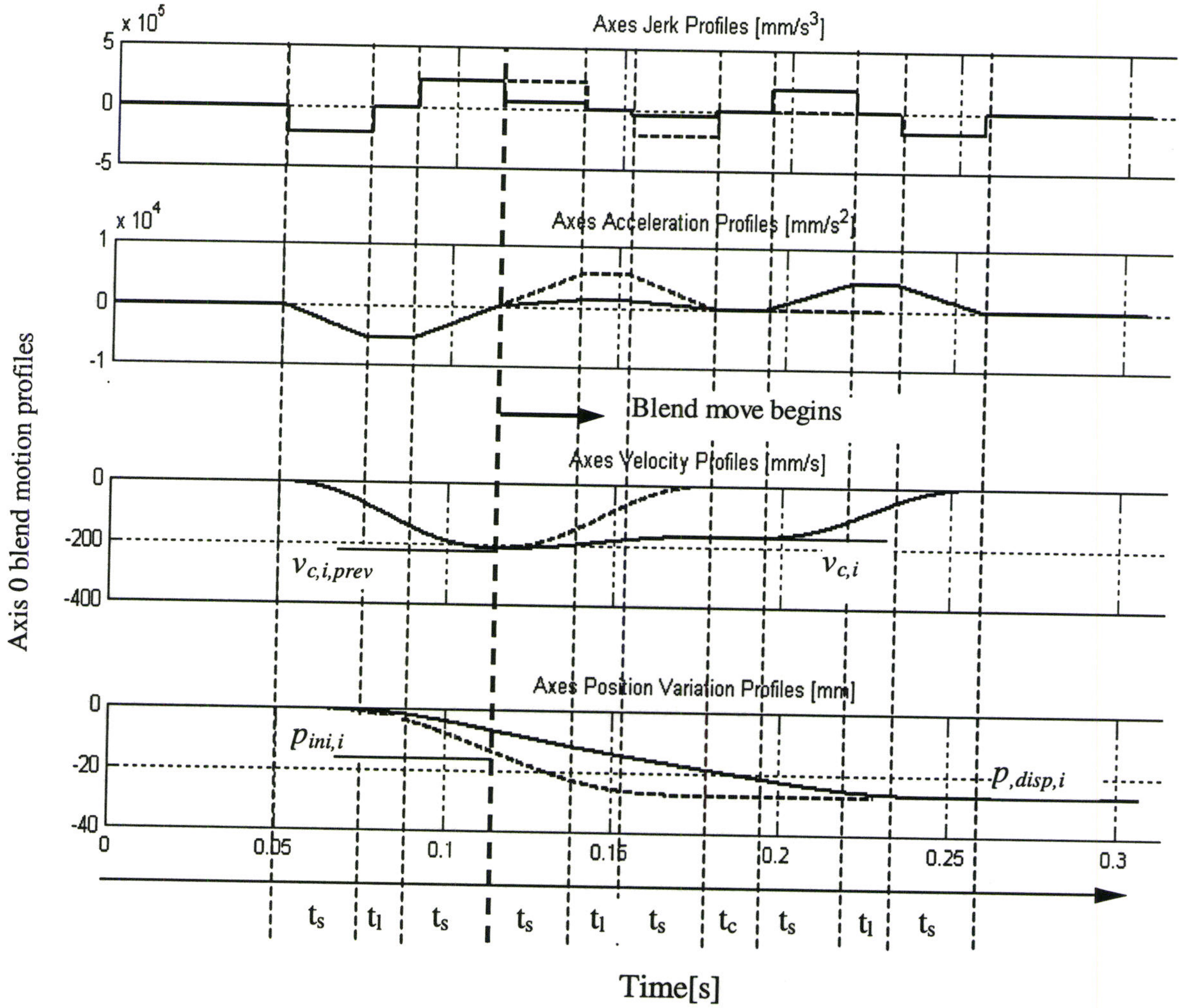


Figure 5.5 Motion profiles for a two block blended move of axis 0

$$p_{disp,i} = \sum_{first\ block}^{last\ block} p_{move,i} \tag{5.25}$$

In order to determine the acceleration time parameters during a blended move, the proposed interpolator assumes that the next move to be blended is the last move in which a return to a stop period is added. So the motion distance, from $p_{ini,i}$ to the end of the assumed last move $p_{disp,i}$, is similar to the distance in Equation 5.16 for a single move, which consists of three parts as

$$\begin{aligned} p_{disp,i} - p_{ini,i} &= p_{acc/dec} + v_{c,i}t_c + p_{return\_to\_stop} \\ &= \frac{1}{2}(v_{c,i,prev} + v_{c,i})t_a + v_{c,i}t_c + \frac{1}{2}v_{c,i}t_a \end{aligned} \tag{5.26}$$

The target velocity for each axis during blended move is consequently expressed as

$$v_{c,i} = \frac{p_{disp,i} - p_{ini,i} - v_{c,i,prev}t_a / 2}{t_m} \tag{5.27}$$

Equation 5.27 define the target velocity as a function of t_a ($=t_s+t_l$), while in equation 5.17 to 5.19, the time parameters t_a ($=t_s+t_l$) are defined as the function of $v_{c,i}$.

In order to meet the position requirement of equation 5.26 and the dynamic limitation specification on each axis, a convergence algorithm applied in [54] for simulated motion is adopted online to decouple $v_{c,i}$ from t_a. This algorithm includes four steps:

Step 1: Initialize the convergence loop. Estimate the target velocity $v_{c,i}$ for the blended move via Equation 5.9 with the defined move distance $p_{move,i}$ and move time t_m. With that $v_{c,i}$, calculate the acceleration time parameters t_s and t_l from Equations 5.17~5.21.

Step 2: With the calculated $t_a(t_s+t_l)$, recompute $v_{c,i}$ using Equation 5.27 to meet the position requirements.

Step 3: Recalculate the new time parameters t_s, t_l and t_c using Equation 5.17~5.21 in order to respect each axis preset peak values for jerk and acceleration.

Step 4: Determine the position error at the end of the blended move (since the start of the blended move) with Equation 5. 28

$$p_{error,i} = p_{disp,i} - p_{ini,i} - v_{c,i,prev} t_a / 2 - v_{c,i} t_m \tag{5.28}$$

The stop criteria of this convergence algorithm is to minimize the position error to be less than one BLU. If this criteria is satisfied, the current calculated target $v_{c,i}$ and time parameters t_s, t_l and t_c are applied to the blended move. Other wise the convergence loop repeats from steps 2 to 4.

5.5 Interpolator Overview

Based the above ideas, the six axis motion interpolator was developed as a real time thread code which has been integrated with the servo control loop as shown in Figure 4. 5. During interpolation, the tool path is first interpreted as a series of short linear motion blocks by the standalone module Wr_Tp module. This module belongs to Soft Real Time level interface and runs in Linux environment. It assigns each block with the following parameters: the start and end points, single move or blended move type, the corresponding feedrate, as well as the dwell time between successive blocks if they are in single move type. It finally sets the status of each block as B_INITIALIZED. Then the real time supervisory thread code Get_TP_Thread_Code() (M2 in Figure 4.5) loads the initialized blocks' data structure to shared memory for online interpolation. It sets the tool path status as INITIALIZED.

The online interpolation motion thread code, Interp_Servo_Thread_Code() is then executed in a real time “WHILE ” loop to generate the set-point commands for the servo control loop. The core of the Interp_Servo_Thread_Code() is the function Hexapod_Motion_Interpolation(). Block information is the input to this function, while the motion acceleration, velocity and position profiles are the output. The function uses the variable interp_step as a counter of sampling periods of duration Ts(Ts=0.3ms). The total number of motion profile points, PROF_N_SAMPLES, which is obtained through offline simulation is used as a terminal flag for interpolation motion. The displacement from the first block to the current one is recorded and updated at each sampling period. Hexapod_Motion_interpolation(), labelled as N0 in Figure 5.6, consists mainly of 11 function modules labelled from N1 to N11. The N1, N2 and N3 functions form three stages of N0. If N1 (Next_Move_Initialization with flowchart shown in Figure 5.7) initializes the first block or the next block and checks block status successfully, then N2 (Compute_Sub_Move in Figure 5.8) will compute the profiles of the active sub-move. Following the successful return of N2, the running absolute time increases, and N3 (Set_Sub_Move) will be run to determine the next active sub-move with respect to the global time t.

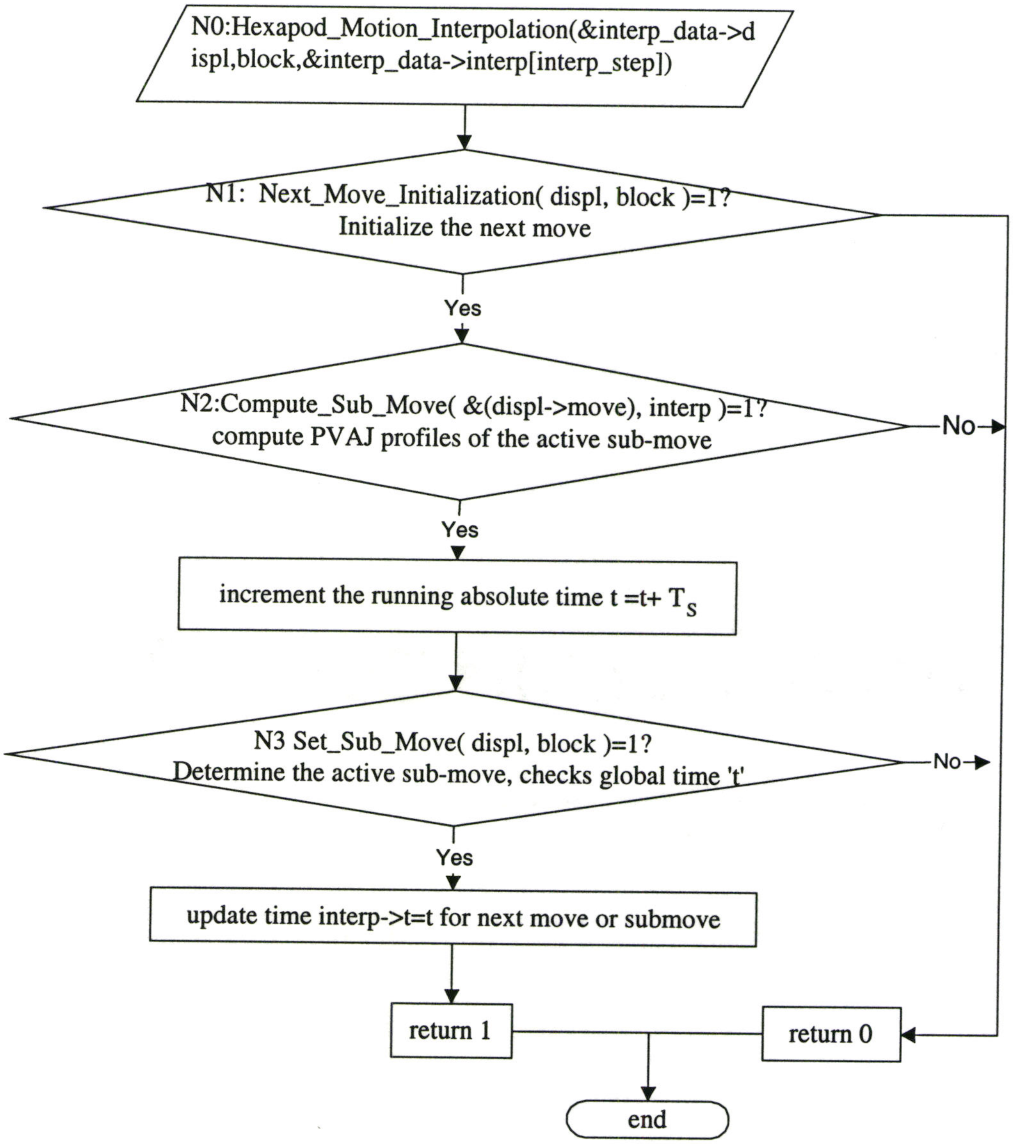


Figure 5.6 Flow chart of Hexapod_Motion_Interpolation function

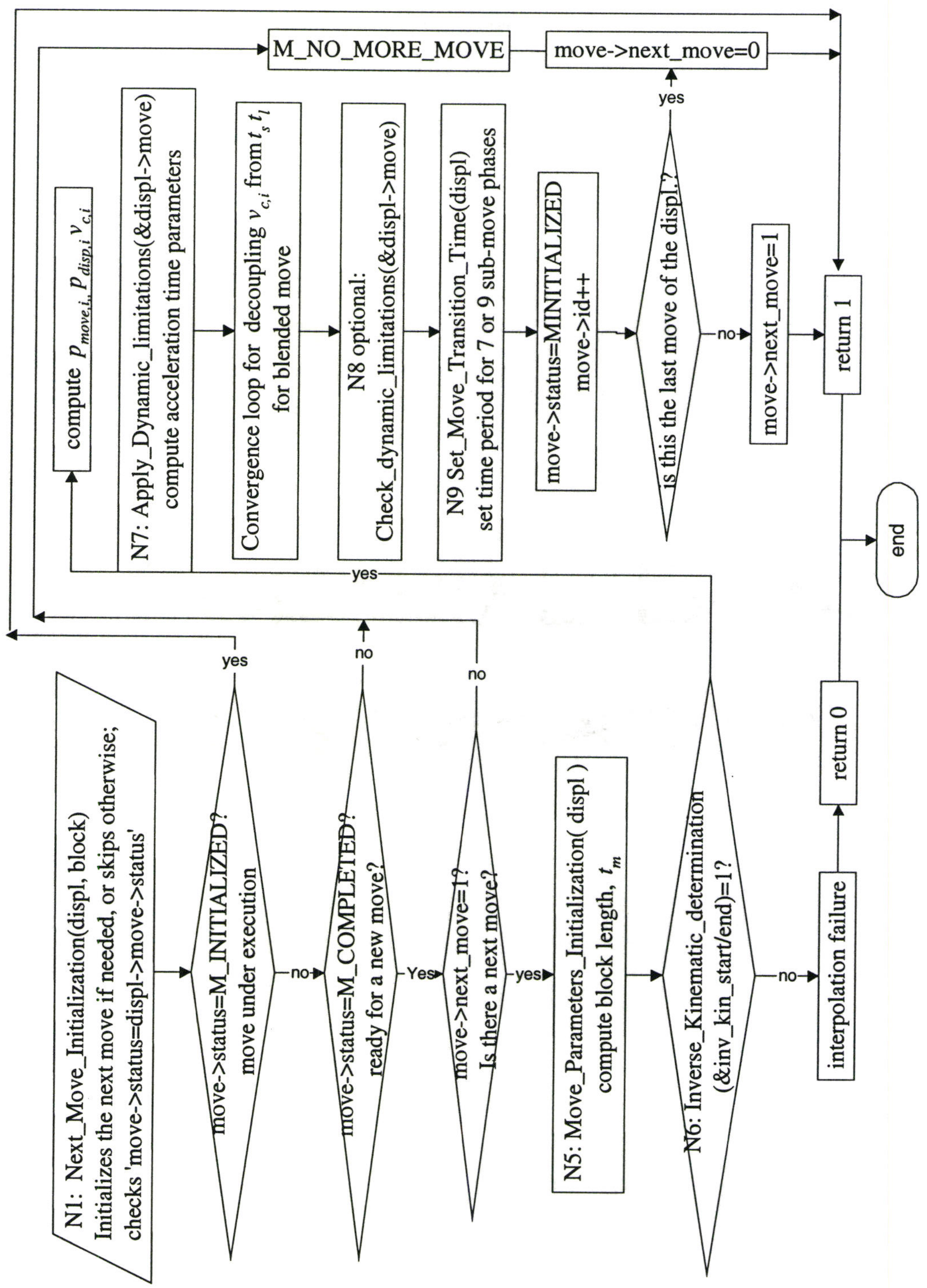


Figure 5.7 Module N1: Next_Move_Initialization

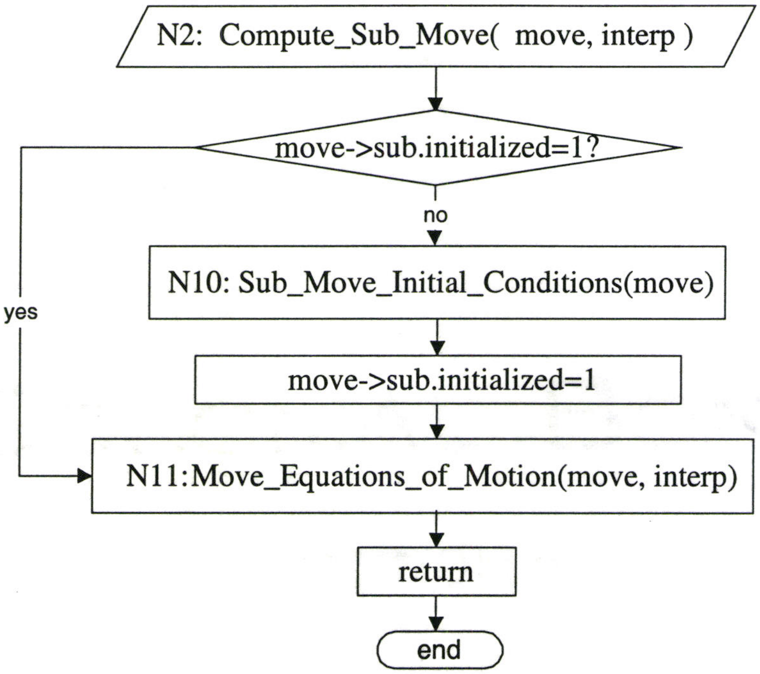


Figure 5.8 Module N2: Compute_Sub-Move

5.6 Real Time Online Interpolation Results

With the online interpolation code developed above, the motion of the machine in single move and blended move type was extensively tested for different feedrates. Because the machine is still being developed and has not been calibrated, all the motion tests are carried out under relatively low speed with the a maximum feedrate tested of 80mm/s.

5.6.1 Single Move Motion Interpolation

The single move motion was tested and used for the repeatability and straightness measurement of the machine that is described in chapter 6. One of the straightness measurement motion tool paths is shown in Figure 5.9. It is a planar line from (0,

-100mm, 2064mm) to (-200mm, 100mm, 2064mm) in machine coordinate system $B - X_B Y_B Z_B$ The vertical lines in the figure denotes the orientation of the platform at each block's end.

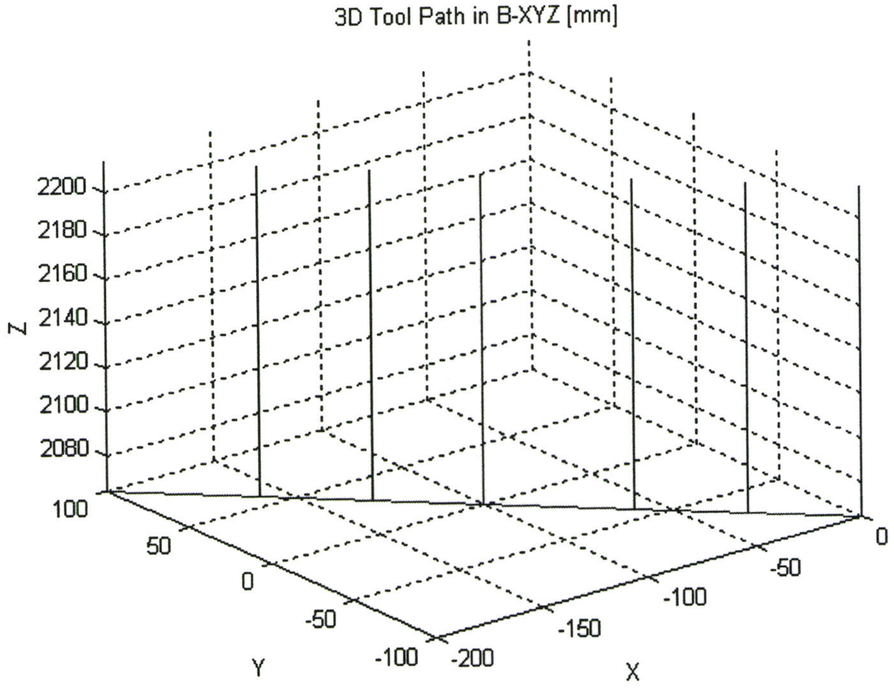


Figure 5.9 Tool path in single move motion used for straightness measurement

The motion runs in single move type for one or two bi-directional cycles with different feedrates for different blocks, so the tool path consists of 12 or 24 blocks. For the 12 blocks motion, the feedrate is 20mm/s for the first and last bock, and 32mm/s for the others. The dwell time is 0.05s for the first and last block, and 15s for others. For the 24 blocks' motion, the corresponding feedrates are doubled. With the sampling period of 0.3ms, for the 12 blocks' motion, the total motion profile points generated is 514,721.

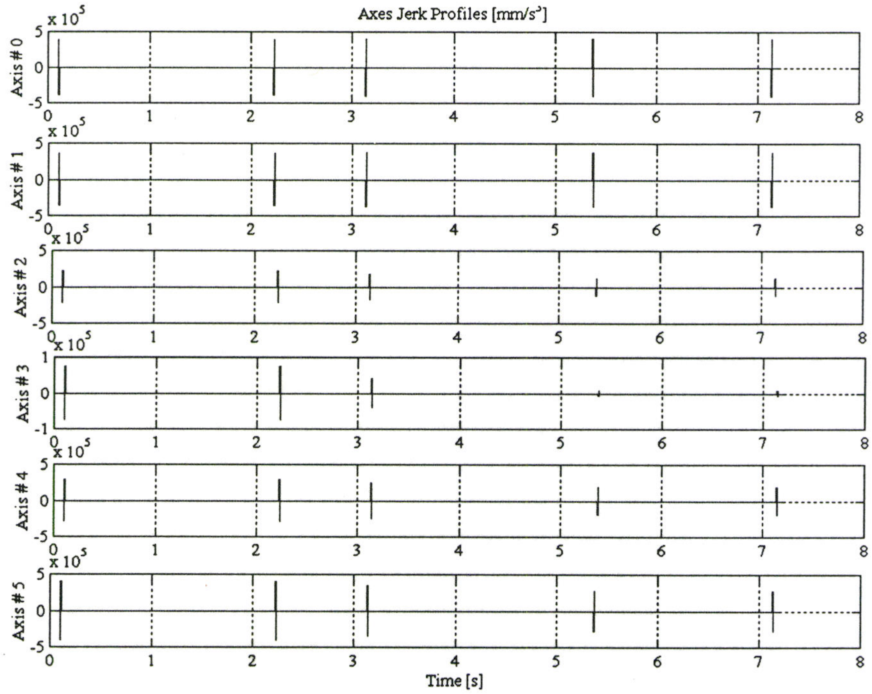


Figure 5.10 Jerk profiles for a three block single move motion

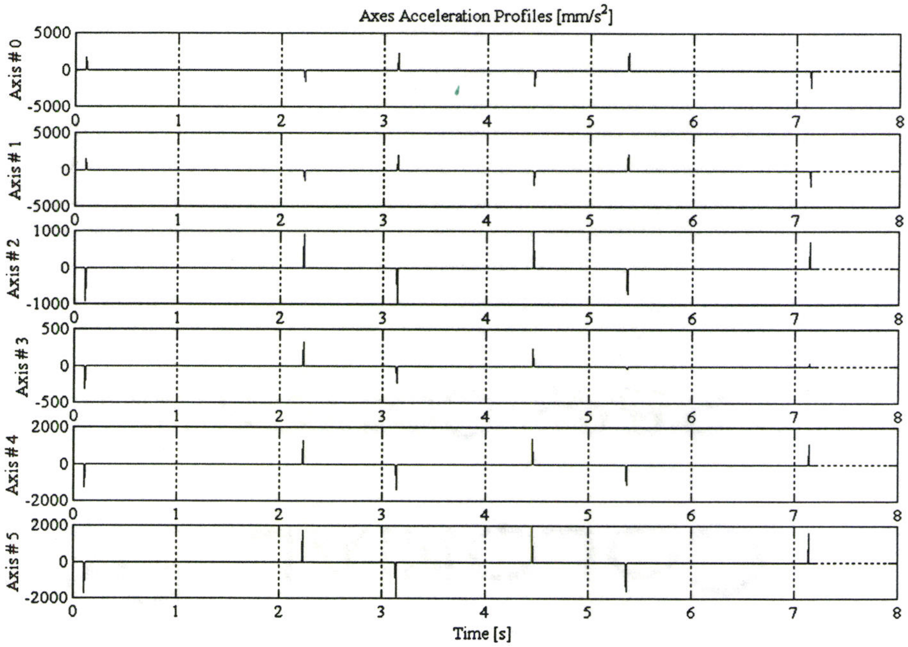


Figure 5.11 Acceleration profiles for a three block single move motion

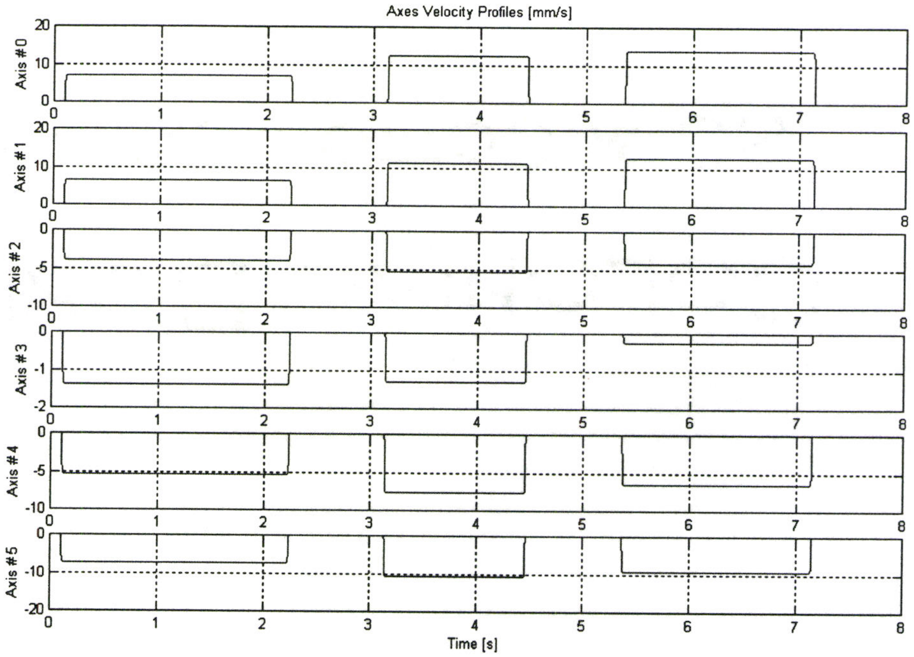


Figure 5.12 Velocity profiles for a three block single move motion

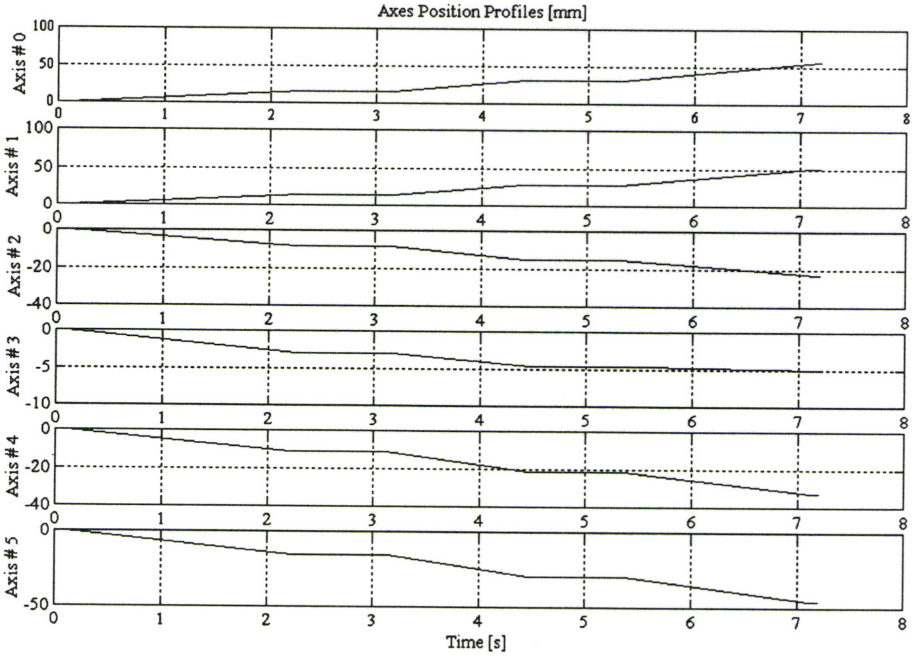


Figure 5.13 Position profiles for a three block single move motio

Due to the huge amount of profile points, for plotting purpose Figures 5.10 to 5.13 only shows the motion profiles for the first three blocks with dwell time reduced to 0.9s between the second and the third blocks.

Figures 5.10 to 5.13 indicate that the coordinated motion of the six axes is insured through the inverse kinematics model of the machine and the identical motion times. Each constant jerk profile is limited by the preset peak jerk value that can be observed in Figure 5.10. The acceleration profiles in Figure 5.11 are also inside the limits of the preset acceleration values. For the specified tool path and feedrate, the acceleration profiles are triangular, because the relative low target velocity can be achieved within $2t_s$. As a consequence, the resulting velocity profiles in Figure 5.12 have a long constant velocity period with very short period of pure S curves in the acceleration and return to a stop phases. The feature of a single move, starting from a stop and ending to a stop, is fully realized in the velocity profiles in Figure 5.12 and the position profiles in Figure 5.13.

5.6.2 Blended Move Motion Interpolation Result

Blended move motions were tested for a tool path with a planar rectangular shape. This shape was chosen to allow visuaaly observing the movement of the tool tip and of each axis. The motion was also implemented at relative low feedrate. In order to detail the online interpolated motion profiles of a blended move, a rectangular tool path including 10 linear motion blocks shown in Figure 5.14, was simulated at feedrate of 235mm/s. One starting block begins at the origin of $B - X_B Y_B Z_B$ was added before the

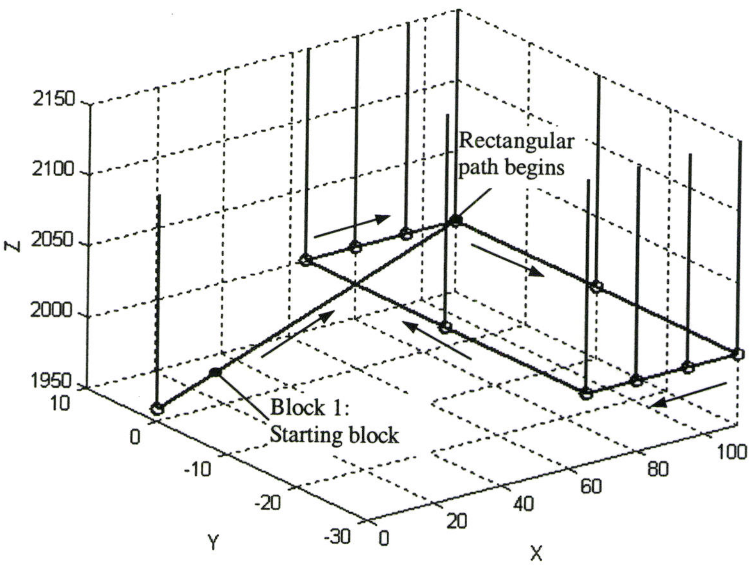


Figure 5.14 Rectangular tool path used for the blended move motion

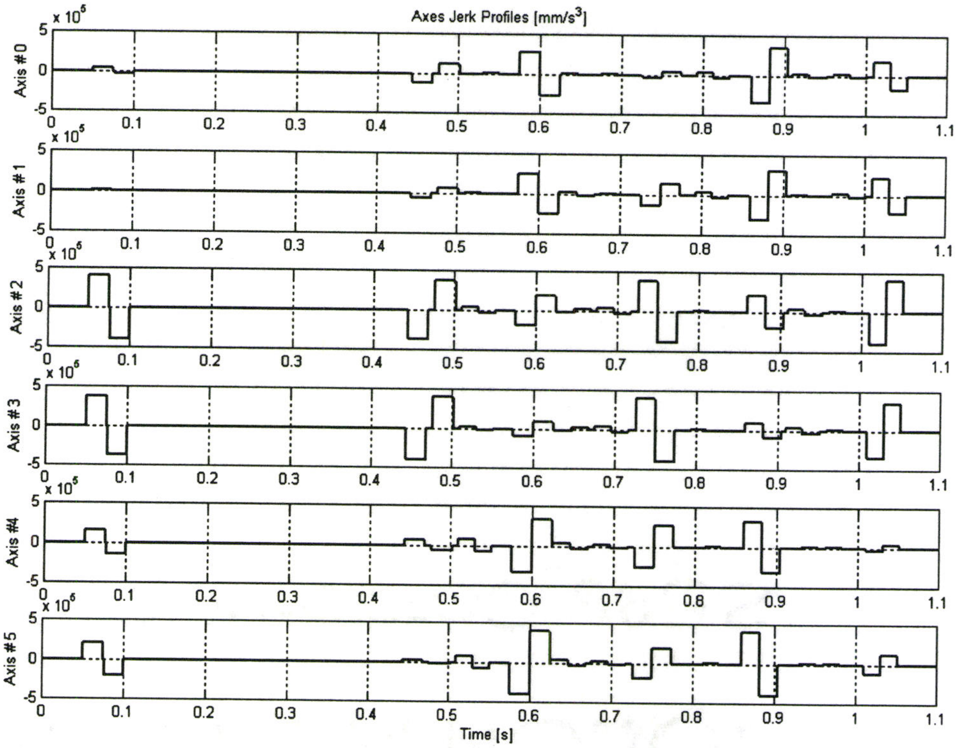


Figure 5.15 Jerk profiles for rectangular tool path blended move motion

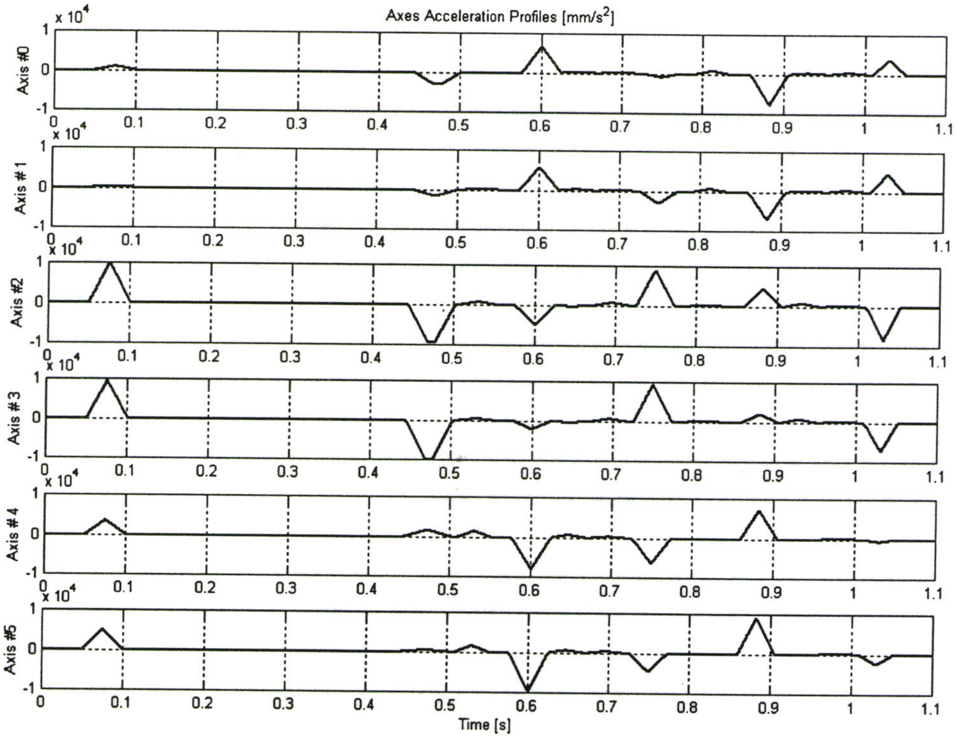


Figure 5.16 Acceleration profiles for rectangular tool path blended move motion

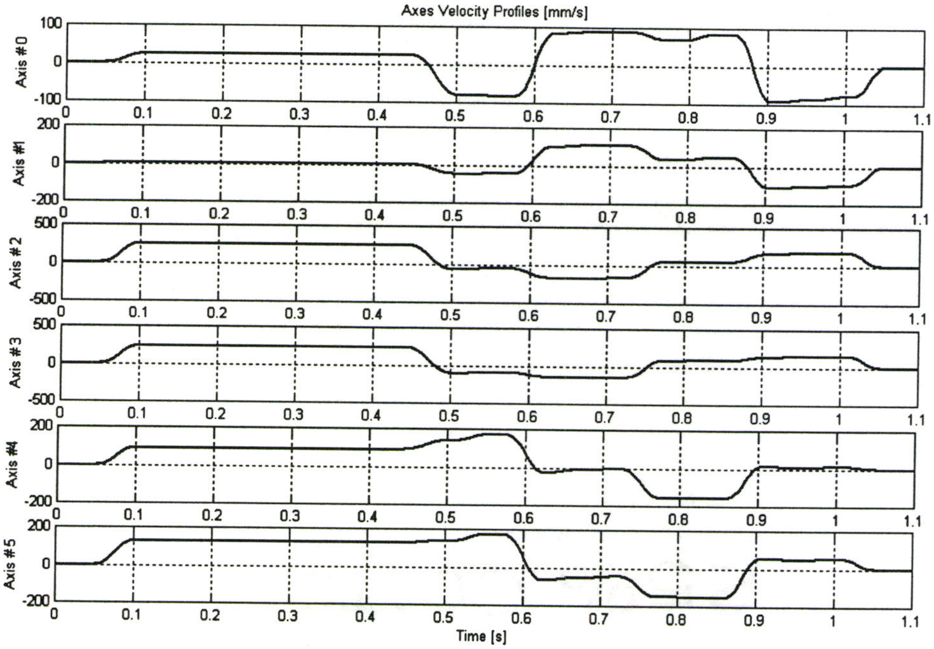


Figure 5.17 Velocity profiles for rectangular tool path blended move motion

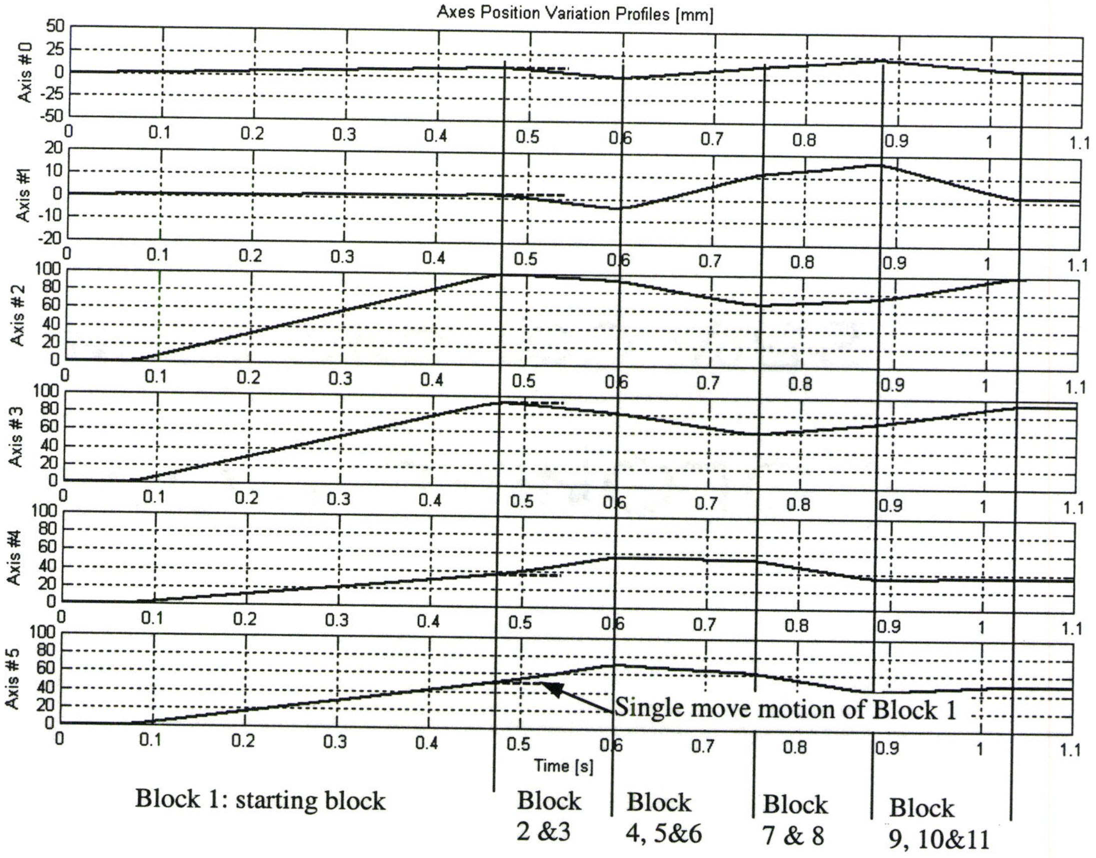


Figure 5.18 Position profiles for rectangular tool path blended move motion

studied rectangular tool path. The resulting jerk, acceleration, velocity and position profiles online generated are graphically illustrated in Figure 5.15 to 5.18 respectively. Note that a blended rectangular motion is actually impossible.

The piecewise constant jerk, triangular acceleration, S-curve velocity and cubic position profiles for six axes are clearly visible in these figures. The preset peak values for jerk and acceleration at each acceleration period and decelration period are always respected. The acceleration profiles are still triangular in shape since the maximum target velocity can be achieved in $2t_s$. Without the linear varying velocity acceleration time period t_l, the velocity profiles consists of two pure S curves at the acceleration and deceleration periods and one constant velocity line during each move time t_m. The motion is blended between the successive blocks with the velocity smoothly changed in pure S curve as shown in Figure 5.17.

The position profiles in Figure 5.18 changes their values and directions simultaneously with the other profiles. In addition, they provide useful information for visually observing the motion of the tool tip when it moves along different coordinate axis of $\mathrm{B-X_B Y_B Z_B}$. During the second and third blocks of the blended motion indicated in Figure 5.14, the tool tip moves along negative $\mathrm{Y_B}$ axis direction in machine coordinate system $\mathrm{B-X_B Y_B Z_B}$. To achieve this movement, according to the layout of the machine and the definition of the machine $\mathrm{B-X_B Y_B Z_B}$ system, the 0 & 1 axes and the 2 & 3 axes should move up (in the negative direction) towards their home positions, while the axes 4 & 5 should move down (in the positive directions). This observation is fully verified by the position profiles for the second and third blocks given in Figure 5.18.

5.7 Conclusion

The six axes motion interpolator for Tiara hexapod was developed based on the idea and definition for linear blending move of PMAC [45] and online implemented for single move and blended move motions.

The interpolator uses the inverse kinematics model and the identical motion time to coordinate the motions of six parallel axes to produce the desired tool tip motion with specified feedrate. It generates a smooth pure, or partial S-curve velocity profile and cubic position profiles for each axis based on piecewise constant jerk profiles. The constant jerk and the resulting acceleration values are limited by two interpolator parameters, the peak jerk and the peak acceleration, which are defined by the user according to the machine system dynamic limitations and machining operation.

The developed interpolator, which consists of eleven dependant and two standalone modular functions, has been successfully integrated with the improved servo control software. Extensive interpolation simulations, tests and measurements for single move and blended move with different tool path and different feedrate were carried out. The simulation and test results demonstrated the desired interpolation properties were fully realized.

Chapter 6

Motion Control Experiments

6.1 Introduction

In this chapter, the performance of the improved servo control motion of axis system and online interpolation motion of tool tip are experimentally evaluated for the un-calibrated Tiara hexapod.

To begin with, the limitation of the measurement of hexapod machines and the Tiara hexapod specifically are described. Following this, the experimental verification of axis motion and interpolation motions are performed. Next, the homing position and closed loop homing repeatability are measured. Then, the linear motion repeatability and accuracy measurements are carried out. The main factors affecting the motion accuracy are analyzed with forward kinematic simulation. The chapter closes with a set of conclusions.

6.2 Limitations of the Measurement of the Tiara Hexapod

As mentioned in chapter 2, one key area interest is standard test methods and measurement procedures to evaluate the performance of a hexapod machine. For conventional CNC machine tools, a number of standards such as ASME B5.54 [61] have defined tests that give an indication of machine tool performance. These tests are designed to analyze the performance of a specific orthogonal axis, or a single rotary axis, and are now well proven and industrially accepted.

For hexapod type machine, there is no orthogonal linear or rotary axes, so many of the tests available from the standards for conventional CNC machine measurement and testing may not be appropriate for such machines. Some of the tests, such as motion repeatability and accuracy tests are still useful as they allow direct comparisons between hexapod machines and conventional machines that provide information on the basic performance of any of machine tool. The conventional metrology systems for such tests including laser interferometers, ball bars, electronic levels and capacitance gages can be applied to the hexapod machine tests and measurements. However, special experiment set-ups and machine motions need designed to obtain the desired result.

Meanwhile, a hexapod is very likely to have irregular workspaces as presented in chapter 3 for the Tiara Hexapod, so test results for stiffness and linear motion accuracy considerably depend on where in the machine's workspace you measure them. This means that it is not straight forward deciding where to run a particular test.

Regarding the Tiara hexapod, it is still under development without kinematic calibration. How far the nominal parameters derived in chapter 3 are from the actual parameters is unknown. So the motion error might be too big to be measured with laser interferometer and proximity sensors, the two currently available measurement device for Tiara hexapod. In addition, the Tiara hexapod is not equipped with a worktable. For experiment purpose, a conventional milling pallet has been clamped on the frame bottom plate of the Tiara hexapod as a temporary worktable. The relationship between this "worktable" and the machine coordinate system is unknown and needs to be constructed with spatial measurement device in the future calibration procedure. Limited by these factors, in this research, the measurements are carried out in the machine workspace near this temporary worktable, with the objective to evaluate basic motion performance of the machine and to decide upon the proper measurement device, which will be used to calibrate the machine in the future research.

6.3 Experimental Verification of Motion Programs

Each of the motion functions developed in Chapter 4 and 5 are tested in this section to see if the motion programs do what they are programmed to do. For each of the motions described below, the motion verification is completed by visual inspection and measurement. The motions include axis motions and linear interpolated motions along different tool paths. The verification is achieved by moving the platform along specified motions paths and ensuring the movement occurs in the correct direction and in the correct shape.

6.3.1 Axis Motion Tests

Axis motion tests include axis jogging motion (M_AXIS_JOGGING) and closed loop homing motion (H_CL) under M_MANUAL operation mode, the automatic A_RUN motion under M_AUTOMATIC operation mode with the signal generator.

Many axis jogging motion tests were carried out with various motion parameters including the axis position, jogging speed and direction, as well as the jogging type as shown in Figure 4.5. The axis jogging motion distance along each axis was measured and verified by the servo control software. The motion test shows that axis jogging motion can move the six axes to or from some specific positions. With this function, the machine can come out of the possible singularity area or it can approach the pre-home zone for closed loop homing.

Closed loop homing motion was tested following the procedure describe in Figure 4.6 with different homing speeds. With the help of the installed mechanical compensator, each axis reached the same homing position in various test motions. At the home positions, the parameter H_L_i defined in Chapter 3, the closed loop homing motion repeatability of each axis were measured and are presented in section 6.4.

Automatic axis motion with A_RUN command was also tested with the built in signal generator. Motion with different signal types (i,e, RAMP, SINE_WAVE and SQUARE_WAVE) were performed individually with various moving speeds and time durations. Under each case, the six axes began and ended the motion at the correct times.

6.3.2 Linear Interpolation Motion Verification

Linear interpolation motions include single move type motion and blended move motion. The testing of linear motion is ensuring that the linear motion with specified tool tip position and orientation correctly occurs with respect to the various motion parameters configuration. As stated in section 5.5.1, the following parameters affect the linear motion: the start and end position and orientation of each tool path block; the corresponding federate; single move or blended move type of the motion; and the dwell time between successive blocks for single move type motion. All of these parameters were configured with different values to test the resulting motion of the tool tip. The single move type motions along X_B , Y_B axes and $X_B Y_B$ plane diagonal direction of the machine coordinate system $B - X_B Y_B Z_B$ were observed. These motions moved at different feedrates for each block and dwelled for a specified time period between blocks.

To make it easy to observe the motion shape and direction changes of blended move type motion, a series of rectangular tool path were designed on the $X_B Y_B$ plane of machine coordinate system $B - X_B Y_B Z_B$ similar to that shown in Figure 5.14. Various blended move type motions occurred as the specified tool paths had different block feedrates. The six axes position changes of those rectangular tool paths were found to be similar to the profiles shown in Figure 5.18. The orientation changes of the platform between blocks in linear interpolation motions were also clearly observed. With the linear single move and blended move motions verified, the position repeatability and accuracy were measured as described in section 6.5.

6.4 Homing Position and Repeatability Measurement

Previous sections of this chapter have detailed the qualitative verification of the motion functions, motion direction changes and the generated tool paths shapes of the axis motions and interpolated motions. The next step in the motion testing was to quantitatively test the axis motion and the generated tool path. The homing position and homing motion repeatability of each axis were measured first. Those measurements were carried out first without the legs and platform attached on the slide joints, and later verified with the legs, platform and the mechanical compensator assembled on the machine.

The homing position of each axis is defined by the parameter H_L_i as shown in Figure 3.5. It is the distance between the slide end block and the slide carriage when the slide carriage stays at its home position. This home position was measured with a digital caliper with resolution 0.01mm. The results are shown in Appendix A, Table A.6

The homing position repeatability for the six axes were measured individually according to the ASME B5.54 standard [61] with Mitutoyo dial indicator (resolution 0.001in(25.4microns)) which was attached to the frame with a magnetic stand. The measurement was performed without the legs and platform attached on the axis slide joints due to two reasons. First, with legs and platform attached and before installation of the mechanical compensator, the platform will fall down and cause the carriages to move away from their homing positions after brakes are first enabled. Second, it was difficult and unsafe to go inside the machine to read the dial indicator, which was located at different positions more than 1.5m above the ground on the machine frame. The measurement for each axis was performed first when it was dwelling at the home position. The results for axis 1 and 4 are shown below in Figure 6.1 and 6.2, others are listed in Appendix C. The measurement results at the dwell state show that, when the machine is under control in the closed loop homing procedure, the homing motion repeatability is less than 0.001in(25.4microns). During this procedure, the brakes were disabled, the braking plate of each axis brake could freely rotate with the axis screw shaft as shown in Figure 6.3. Each axis dwelled at its home position controlled by the servo controller.

In order to find the influence of gravity on the carriage, joints, leg and platform, as well as brake function to the home position, another measurement with the data plotted in Figure 6.1 was performed on axis 1. This measurement was performed when the brakes were enabled and the machine was stopped. On the carriage of axis 1, a fixture was attached to support the power and signal cables of electrical spindle. After all the axes reached their home positions and kept dwelling for a specified period of time, the six brakes were enabled, the braking plate inside the brake, which has a hexagon hole to fit the hexagon braking shaft as shown in Figure 6.3, was then pressed on the brake body by the spring to stop the rotation of the axis. Due to the gravity of the spindle wire cables and its supporting fixture, the clearance and wear between the hexagon hole of brake plate and the hexagon braking shaft, the carriage of axis 1 moved down from its home position. The dial indicator reading for the homing position was reduced between 0.008"and 0.010" as shown in Figure 6.1, the homing repeatability measured with the brake enabled became worse with a value of about 0.002in (50.8 microns).

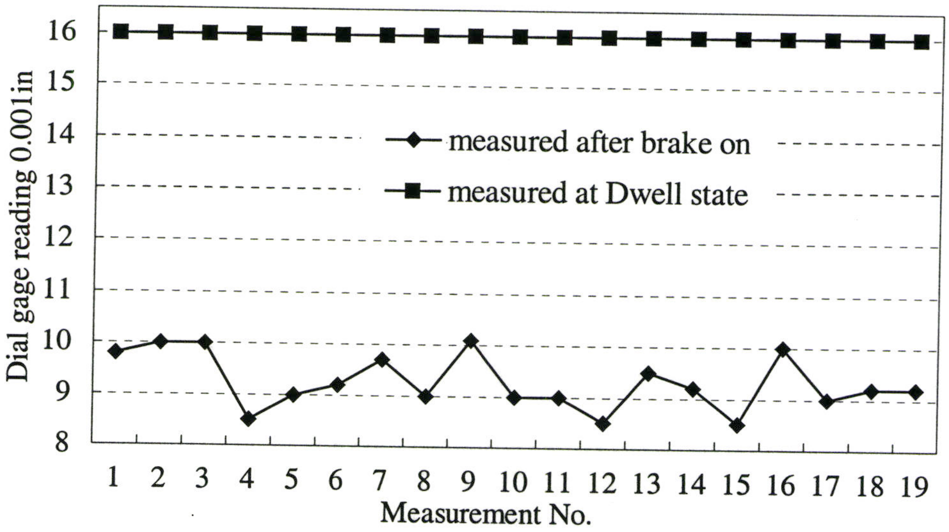


Figure 6.1 Homing repeatability measurement results for Axis 1

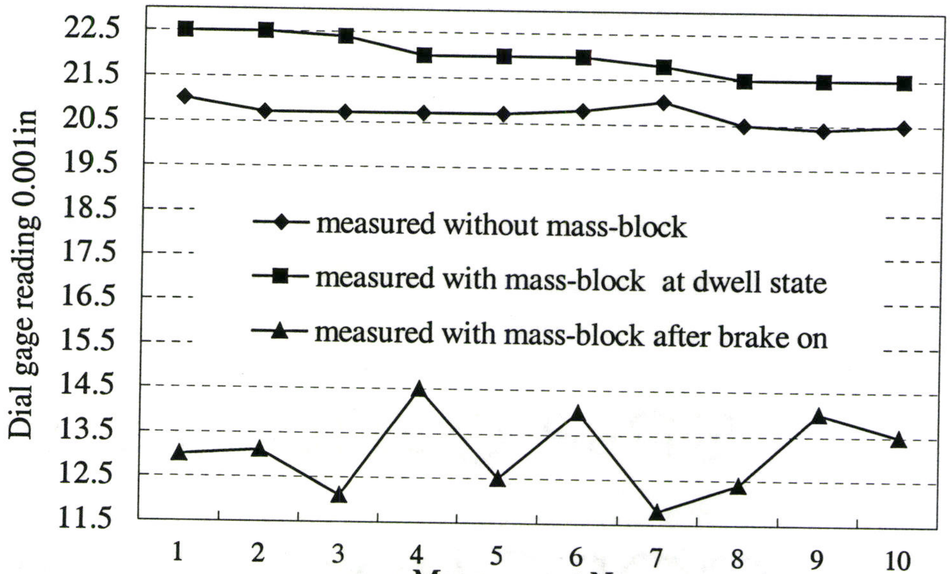
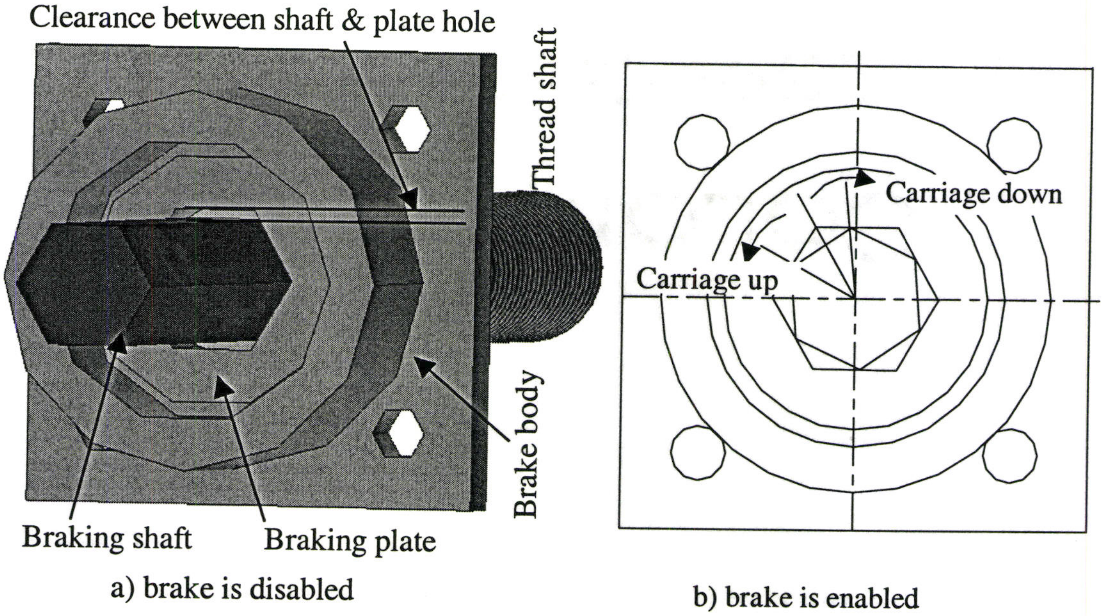


Figure 6.2 Homing repeatability measurement results for Axis 4

To verify this observation, two extra measurements were carried out at the dwell state and the brake enabled on state for Axis 4 with a 10kg block attached to the carriage. At the dwell state, the home position changes with the dial indicator measurement reading between 0.0215in and 0.0225in, the homing repeatability is still less than 0.001in(25.4microns) as shown in Figure 6.2. While the brake is enabled, the dial indicator reading falls down into the region from 0.0115" to 0.0145", the homing repeatability worsens to about 0.003in(76.2micron).



a) brake is disabled b) brake is enabled

Figure 6.3 Axis brake activating states

The above homing positions measurements under dwell state indicate that the closed loop homing function for the Tiara hexapod can make six axes reach their home position with a repeatability of 0.001in(25.4microns), when the machine is under servo control. The homing repeatability measurements after machine is stopped and brakes are enabled show that the gravity acting on the legs and platform system and the brake design affect the home positions. The homing repeatability measured after the machine was stopped was about 0.003in(76.2microns).

6.5 Linear Motion Positional Repeatability and Accuracy Measurements

Generally, a linear positioning accuracy test aims to assess the individual linear axes of a conventional CNC machine with orthogonal axes. The Tiara hexapod has virtual X_B, Y_B and Z_B axes in the machine coordinate system that are used for simplify control of the machine for the operator. These axes could be selected as the directions for linear motion accuracy measurement. The machine was controlled to move along a virtual axis direction, stopping at preset displacement intervals where the positional error was recorded with dial indicators. Three or more bi-directional runs were measured for each axis direction motion, the accuracy and repeatability figures were calculated from the complete set of data according to ASME B5.54 [61].

6.5.1 Linear Motion Repeatability Measurement Setup

The linear motion repeatability measurement setup of the Tiara hexapod is shown in Figure 6.4. The mechanical compensator mentioned in Chapter 4 and the milling pallet as temporary worktable were installed on the machine to make the measurement available. A “test block”, an aluminium cubic block, was bolted on the platform of the machine. Three dial indicators with 0.001in (25.4microns) resolution were installed with adjustable magnetic stands on the temporary worktable. They were arranged in such a way that their measuring probes touched and were perpendicular to the three mutually perpendicular surfaces of the test block. Their readings indicated the X, Y and Z direction repeatability at the measured point.

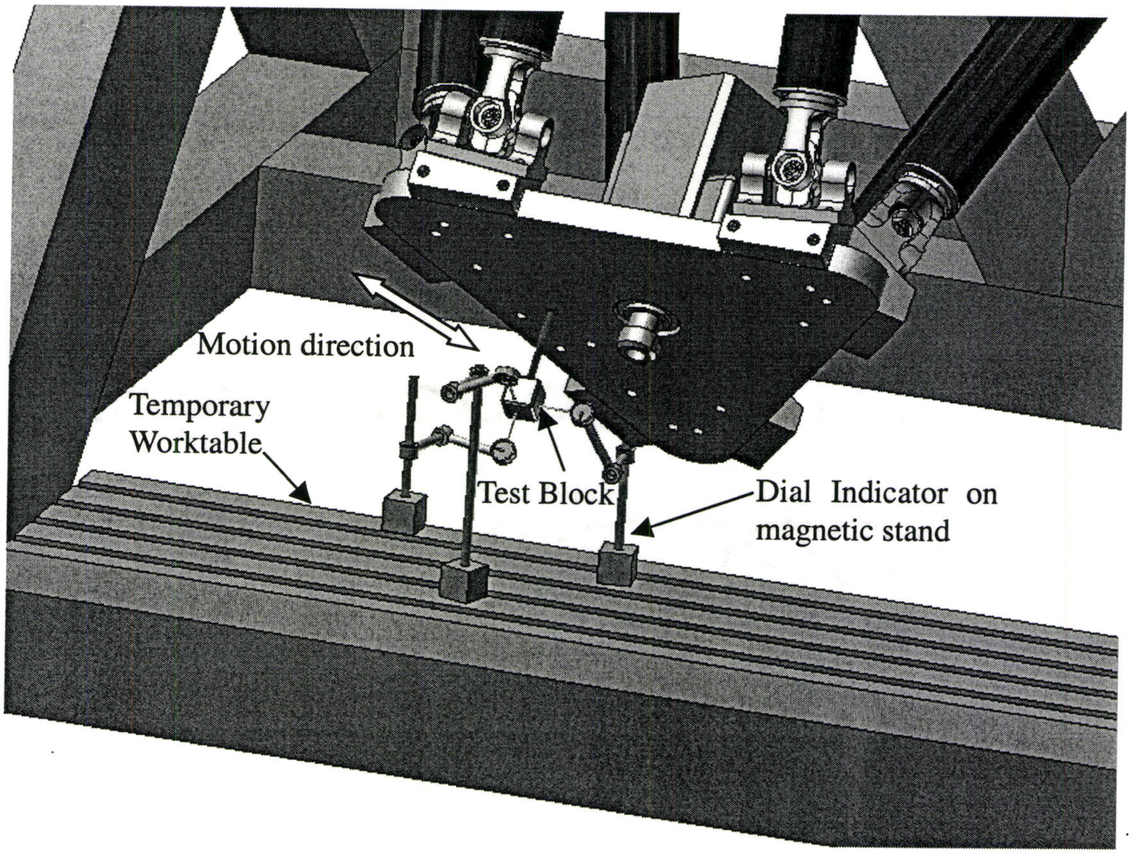


Figure 6.4 Linear motion repeatability measurement setup

The machine was programmed to perform a single move motion described in section 5.6.1 along one of the three motion lines in the machine coordinate system $B-X_BY_BZ_B$. It was programmed to move at 16mm/s from the start position to the end position, dwell at the end point for 20 seconds, and then return to the start point at 32mm/s and dwell there for another 20 seconds. The three dial indicator readings at the measured position, the start position or the end position, were recorded during the 20 seconds dwell interval.

Linear motion repeatability measurements were carried out along the following three straight lines:

- The first was in the X_BY_B plane along the X_B direction from point (-200 100) to point (-100,100). The measurements were done at the start position.
- The second was along the Z_B direction moving 100mm. The measurement were done at the start position.
- The third was in the X_BY_B plane along diagonal direction from point (-100, 0) to point (-200,100). The measurements were done at the end position.

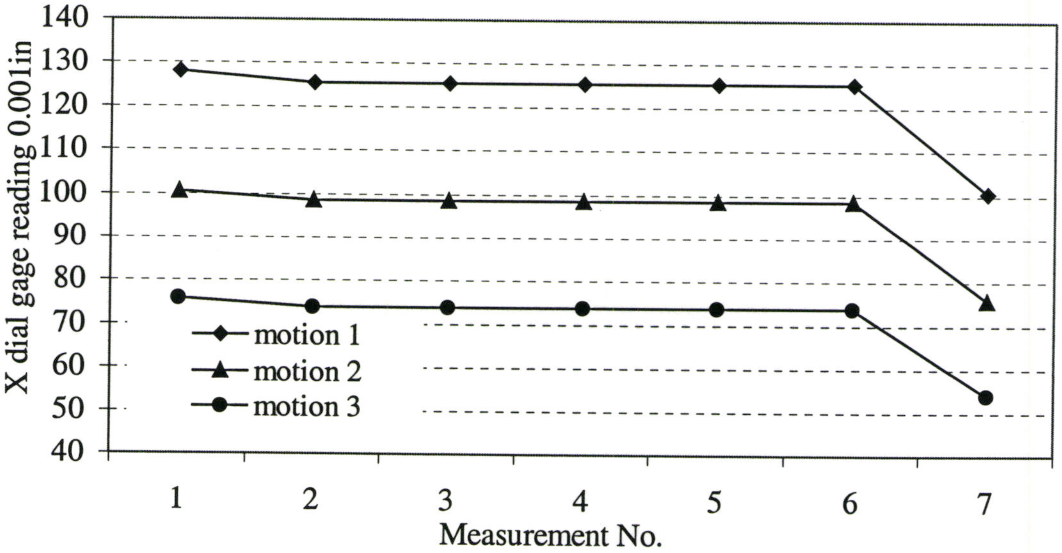


Figure 6.5 X direction repeatability of X_B direction motion measurements

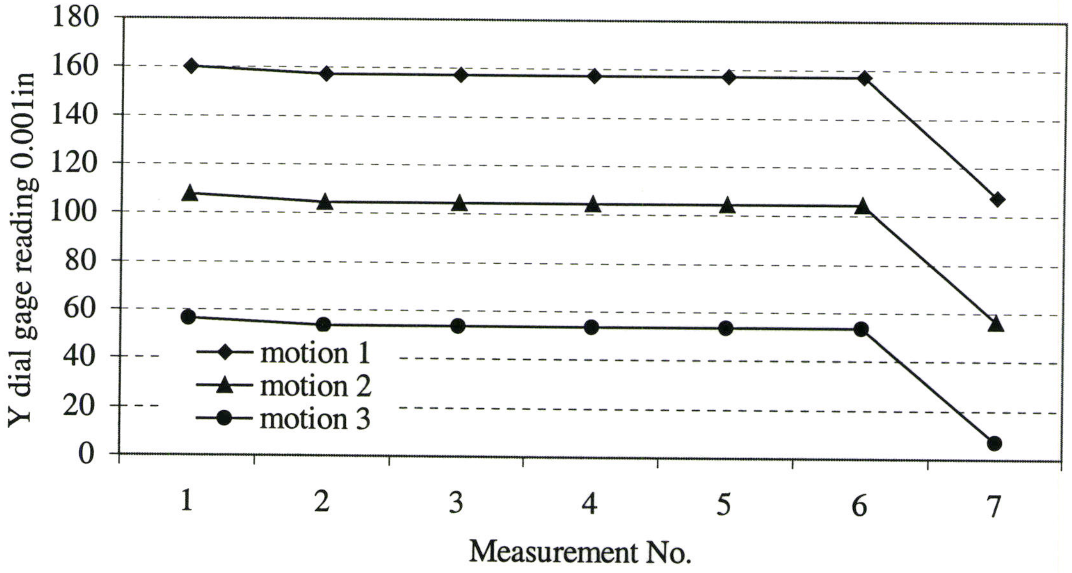


Figure 6.6 Y direction repeatability of X_B direction motion measurements

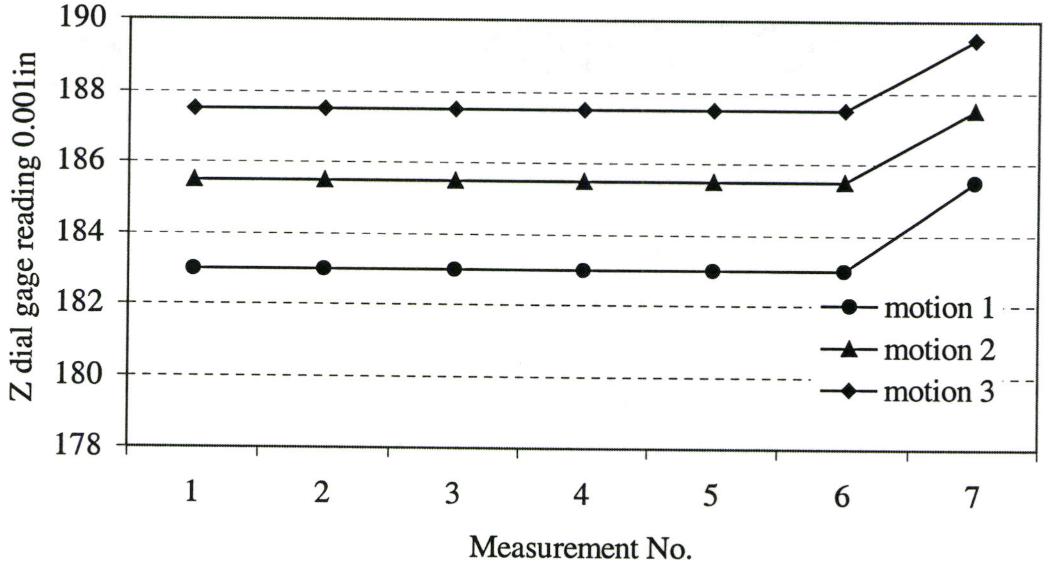


Figure 6.7 Z direction repeatability of X_B direction motion measurements

6.5.2 Linear Motion Repeatability Measurement Results

The repeatability measurement results for linear motion along X_B direction are shown in Figures 6.5, 6.6 and 6.7. The X direction repeatability measurement results for linear motion along the X_BY_B diagonal direction are plotted in Figure 6.8. The other measurement results for X_BY_B diagonal motion and Z_B direction motion are graphically shown in Appendix C.

For the X_B direction motion repeatability measurements shown in Figures 6.5 to 6.7, the three dial gages were arranged to measure the start position. The dial gages (X,Y,Z) reading of (128,160, 183) indicates the start position before motion 1 begins. When the machine starts to run in single move motion, the brakes are disabled and then DC motors are activated to make the machine stay at the start position for the Start Dwell period. At that moment, because of the gravity of the leg-platform system and previously mentioned clearance inside the brakes, the platform will change its position by 0.003in in the X direction and 0.003in the Y direction. After the Start Dwell period, the machine moves to the end position, stays there for 20 seconds, then it returns to the start position at 32mm/s. The dial gages readings at the start position now become (125, 157, 183). In the remaining measurements for motion 1, the dial gages readings at start position stayed at (125, 157, 183), so the repeatability is less than 0.001in (25.4 microns). This period corresponds to the straight lines from measurement No.2 to No. 6 in Figure 6.5, 6.6 and 6.7.

After the measurement of motion 1 was finished, the machine returned to the start point (125,157,183). The brakes were enabled and the DC motors were disabled, and the machine was stopped. Again at that moment, even with the help of the mechanical compensator, because the gravity of the leg-platform and the clearance inside the brake, the platform does fall down again. This is demonstrated by the change of dial gages readings (125,157,183) for measurement No.6 to (100.5, 107.5,185.5) for measurement No.7. The measurements for motions 2 and 3 along X_B direction motion gave the similar results.

The measurements for the second direction motion, the Z_B direction motion, which were performed at the motion start position, produced similar results as shown in Appendix C.

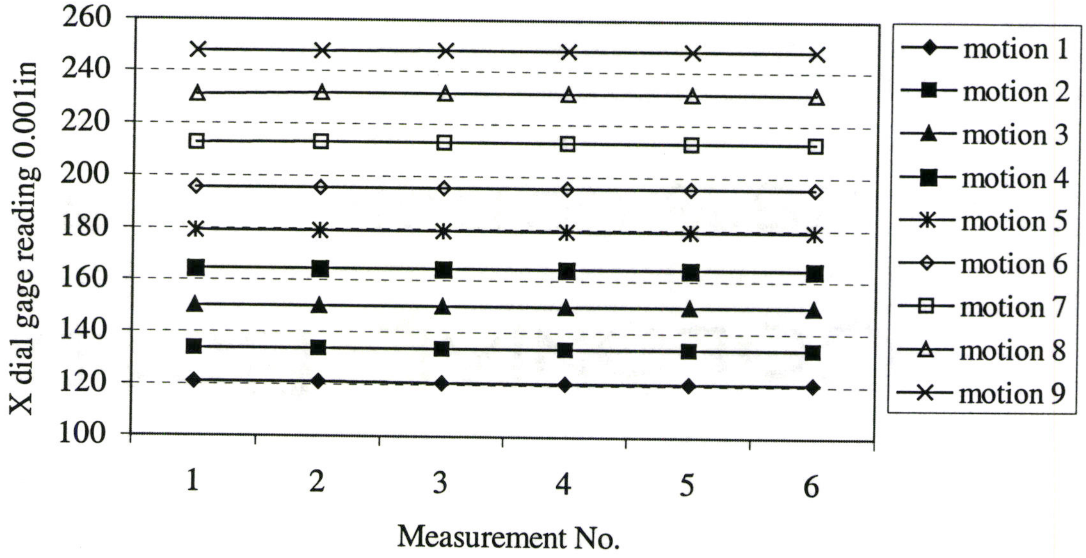


Figure 6.8 X direction repeatability of X_BY_B diagonal motion

The aim of diagonal direction motion measurements for a conventional CNC machine tool is to evaluate the repeatability of the coordinated motion of two or more axes. The Tiara hexapod requires actuated movement of all six axes for any straight line, which implies that any straight line movement through the workspace may be considered as a diagonal. Therefore the measurement results for diagonal motion repeatability should not in theory be considerably different to the linear motion along X_B axis or Z_B axis. For these measurements, the measured point was the end position. The measurement motions were repeated nine times. During each measurement motion, the end position was measured three times. After each motion, the falling down phenomena of platform was observed again. The measurements from the X direction dial gage are shown in Figure 6.8, the other two dial gages readings are given in Appendix C. The measured results show that the repeatability along the X_BY_B plane diagonal direction motion is the same as those for the X_B and Z_B direction motions, which is less than 0.001in(25.4 microns).

Limited by the resolution of the dial gages, the linear motion repeatability for the Tiara hexapod without calibration can be concluded to be less than 0.001inch (25.4 microns). This repeatability value defines the upper bound of motion accuracy enhancement through calibration.

6.5.3 Linear Motion Positional Accuracy Measurement Results

According to ASME B5.54 [61], each individual axis of the machine has six error terms that can be measured. They are the linear displacement error, two straightness errors perpendicular to the linear axis direction, and three angular errors. Laser interferometer is strongly recommended for measuring displacement error. However, for the Tiara hexapod, the virtual X_BY_B plane of machine coordinate system B-$X_BY_BZ_B$ has a

45 degree inclined angle with the world coordinate system O-$X_O Y_O Z_O$, so a special fixture set-up needs to be designed and manufactured for laser interferometer displacement measurement.

Generally, the straightness range of a laser interferometer measurement is +/-1mm, for the un-calibrated Tiara hexapod, it is not guaranteed that the programmed motion along the virtual axis has a straightness within this +/-1mm range. Therefore, even with the specially designed fixture setup, the laser interferometer cannot be used to measure the linear motion errors of the current Tiara hexapod currently.

Limited by these factors, the linear motion displacement error and three rotational errors measurement cannot be carried out in this research. According to ASME B5.54, the two straightness errors tests were performed with a mechanical straightedge and a dial indicator. The measurement setup is shown in Figure 6.9.

A one meter long, high carbon steel square bar, ground flat, which had been used for measuring the linear motion accuracy of a robot by Colella in [50], was used as the straightedge. Limited by the physical dimensions of the temporary worktable, the linear motion straightness measurement motion path was designed along $X_B Y_B$ plane diagonal direction from point (-100, 0) to point (-200,100), as was shown in Figure 5.9. The straightedge was clamped on the pallet as shown in Figure 6.9. It was aligned parallel to the motion path defined by the start and end points so that it is within the measurement range of the dial indicator. The dial indicator used previous was attached to the platform as shown in the figure.

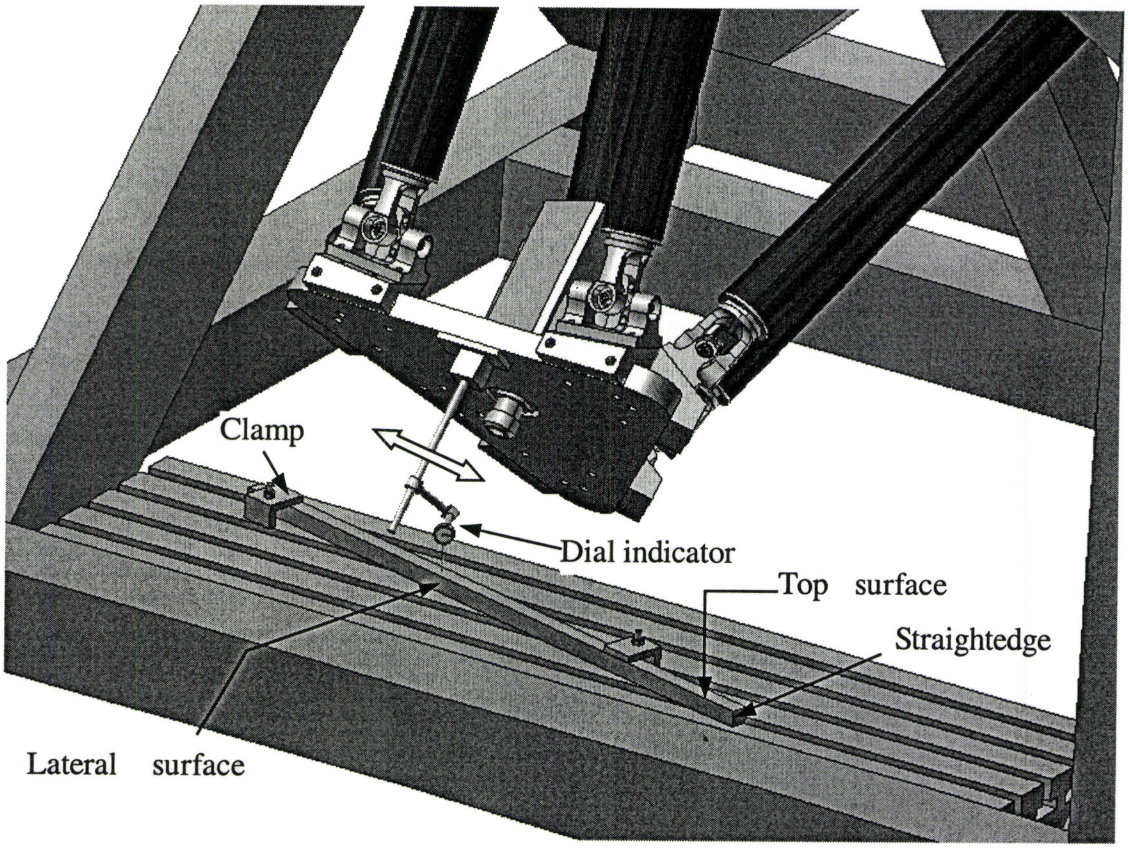


Figure 6.9 Linear motion straightness measurement setup

In each measurement motion, the machine ran in single move motions for one or two bi-directional cycles with different feedrates for different blocks. For one bi-directional cycle, the tool path consisted of 12 blocks; the feedrate was 20mm/s for the first and last bock, and 32mm/s for the others. For the two cycles motion, the total number of blocks was 24, the feedrate was doubled to 40mm/s for the first and last block, and to 64mm/s for the others. The dwell times were kept the same in both cycle motions for reading the dial indicator, 0.05s for the first and last block, and 15s for other blocks.

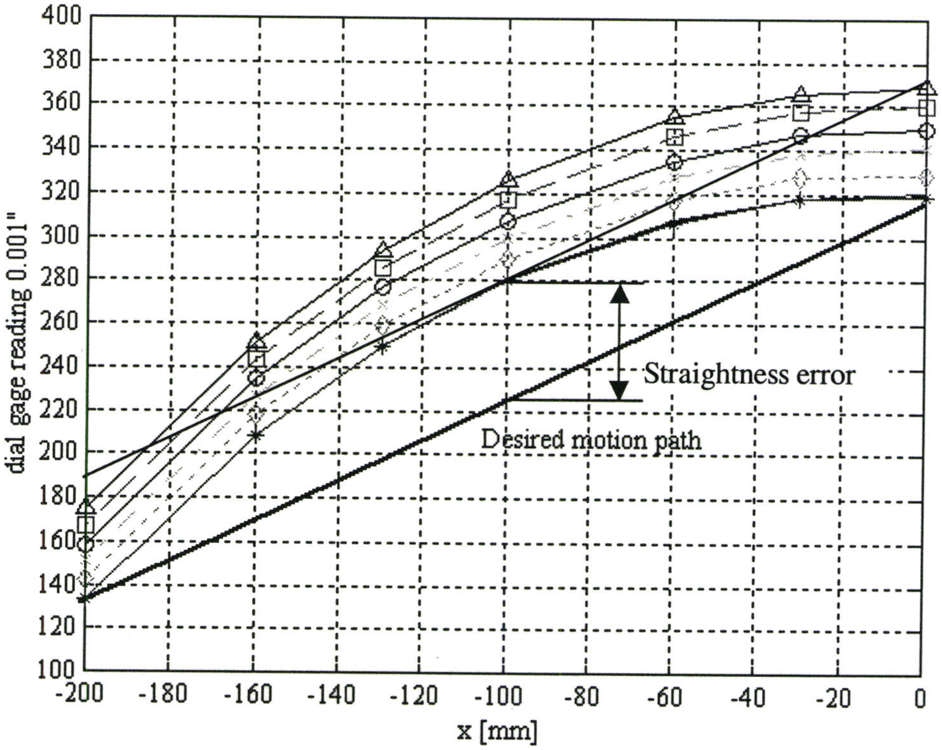


Figure 6.10 Straightness measurements along top surface of straightedge

Straightness errors were measured at the motion block end positions along the top surface and the lateral surface (see Figure 6.9). Measurements for each straightness error were repeated six times, and the results were plotted in Figure 6.10 and 6.11.

The solid line in each of the graphs represents the desired motion path, which the dial indicator would indicate if the machine moved in a perfectly straight line along the straightedge. It is not perfectly horizontal in the graph as it is impossible to manually line up the straightedge such that it is exactly parallel to the programmed motion path of the machine. The plotted dial gage reading at each block's end position indicates seven points on the actual motion path of the machine. There were totally six measured motion paths. Because of the platform falling downwards problem experienced previously in the motion repeatability measurements, there exists a constant offset of 0.01in(0.254mm) between any two successive measured motion paths, and the six measured motion paths are therefore in the same arc shapes.

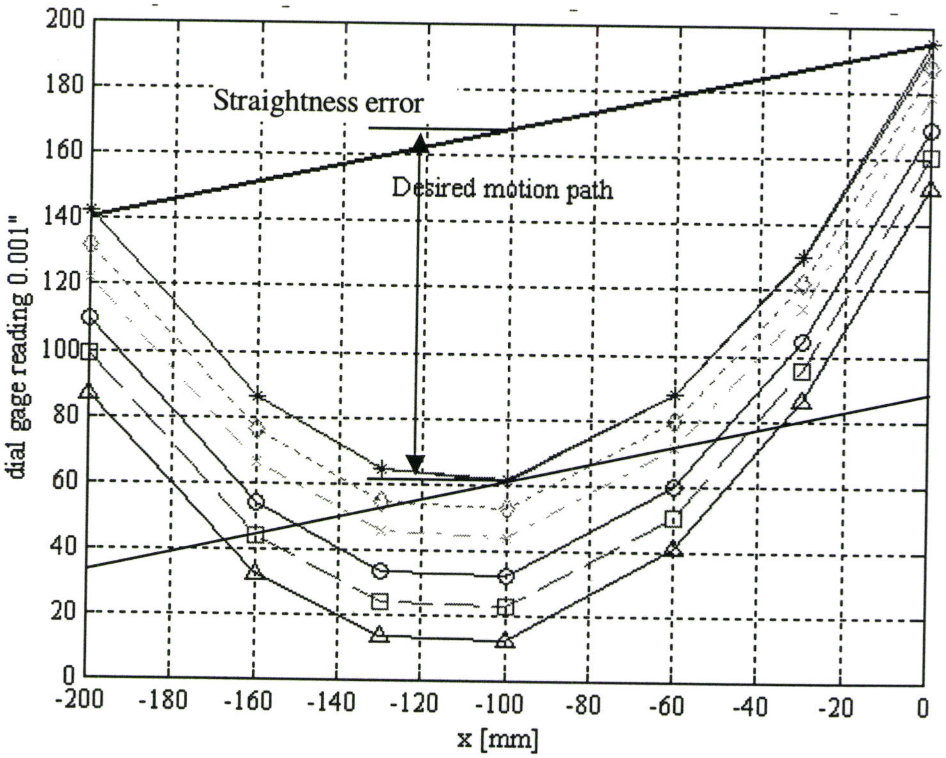


Figure 6.11 Straightness measurements along lateral surface of straightedge

At each measured point, there is a deviation of dial gage reading from the desire motion path. A line is drawn that passes through the dial gage reading point with the maximum deviation and that is parallel to the desired motion path. According to ASME B5.54, the total range between this line and desired motion path is considered as the straightness for that motion. For the measured motions in this experiment, the straightness error along the top surface of the straightedge in Figure 6.10 is about

0.055in(1.40mm), the straightness error along the lateral surface of the straightedge is about 0.108in(2.74mm) in Figure 6.11.

These results show that the linear motion positional accuracy of the Tiara hexapod without calibration is very low. This is due to the inaccurate nominal kinematic parameters used by the interpolator. With this low accuracy, in the future calibration procedure of the Tiara hexapod, some measurement device, such as the proximity sensor with sensing range of 2mm, which was used in [50] for robot motion accuracy measurement, will be needed. The laser interferometer with straightness range of +/-1mm cannot be used for measurement. The survey theodolite, which has been successful used by Masory and Yan in [15] and [28] to calibrate the AFU Stewart platform, is suitable for the Tiara hexapod calibration,

In order to find the main factors among the nominal kinematic parameters influence the motion position accuracy of the Tiara hexapod, the next section adopts forward kinematic simulation to analyze the error sources of the Tiara hexapod.

6.6 Motion Error Sources Analysis

6.6.1 Introduction to Hexapod Forward Kinematics

Given the tool pose or tool path, and other kinematic parameters, finding the slide joint positions or leg lengths of a hexapod is termed inverse kinematic problem. Given the slide positions/leg lengths, the problem of finding the actual tool pose is termed the forward kinematics.

It is well known that for hexapod, the inverse problem is simple with a closed form solution, while the forward one is complicated. For some modified configurations of the six DOF hexapod, such as relocating the joint positions on the triangular platform or using combined joints, the closed form forward solution might be available. For the general six DOF hexapod, the closed form analytical solution is still not available. Several alternative methods exist [34][53].

One analytical solution method based on kinematics mapping for forward kinematics produced a 40th order polynomial. This closed form solution guarantees convergence, but the procedure to obtain the coefficients of the polynomial is computational expensive.

Sensor based forward kinematics is another solution. By introducing redundant sensing, the forward kinematics problem may become simpler in closed form but the inverse problem may become more complex. However, for hexapod, especially for those prototypes being developed at laboratories without well-developed internal measuring and sensing system, modification has to be made by introducing some extra sensors that are not used by the original control system. More over, where and how to place the redundant sensors on an already manufactured hexapod is not a trivial problem, it must be foreseen while designing the machine. Furthermore, the accuracy of the extra sensors must be at least five times as high as the desired accuracy of the platform. Suitable sensors are not available on the market at reasonable costs.

The most common method is to apply numerical method to solve a group of non-linear equations. The advantages of this method are that the mathematical model is simple which avoids the complexity of mathematical deduction. The disadvantage is that, its computation speed is slow, it cannot give all the pose solution of the machine and, the final results are determined by the selection of original values. However, since this method can be applied to any kind of parallel structure, for those parallel structure without closed form solution it is still valuable. In order to quantitatively analyze the motion error sources, the following section will use the numerical method to solve the forward kinematics problem of the Tiara hexapod in order to quantitatively analyze the motion error sources.

6.6.2 Forward Kinematics Model of the Tiara Hexapod

The kinematics model diagram of the Tiara hexapod is shown in Figure 3.11. The pose of the platform consisting of position vector G = $[Gx, Gy, Gz]^T$ and orientation vector $\vec{n}=[n_x\ n_y\ n_z]^T$ can be defined by the following pose vector [51][52]:

$$ {}^{B}\chi=[X_G,Y_G,Z_G,n_x,n_y,n_z]^T \tag{6.1}$$

The forward kinematics problem aims to determine the pose $\chi=[\mathrm{X_G},\mathrm{Y_G},\mathrm{Z_G},\mathrm{n_x},\mathrm{n_y},\mathrm{n_z}]^{\mathrm{T}}=[\chi_1,\chi_2,\chi_3,\chi_4,\chi_5,\chi_6]^{\mathrm{T}}$ of the platform, given slide joint center position λ_i along each slide *i*=1,2,.., 6.

Rearranging the Equation 3.32 as a set of six scalar functions *f*

$$f_i(\chi)=\sqrt{(X_{S_i}-X_{P_i})^2+(Y_{S_i}-Y_{P_i})^2+(Z_{S_i}-Z_{P_i})^2}-L_i=0 \tag{6.2}$$

where $\begin{bmatrix} X_{S_i} \\ Y_{S_i} \\ Z_{S_i} \end{bmatrix}=\begin{bmatrix} X_{F_{1_i}} \\ Y_{F_{1_i}} \\ Z_{F_{1_i}} \end{bmatrix}+\lambda_i\begin{bmatrix} U_x \\ U_y \\ Z_z \end{bmatrix}$, $\begin{bmatrix} X_{P_i} \\ Y_{P_i} \\ Z_{P_i} \\ 1 \end{bmatrix}_B={}^{B}_{P}[T]\cdot\begin{bmatrix} {}^{P}X_{P_i} \\ {}^{P}Y_{P_i} \\ {}^{P}Z_{P_i} \\ 1 \end{bmatrix}$

This is the forward kinematic problem of the hexapod. In Equation 6.2, the slide joint center home position F_{l_i}, the unit orientation vector of the slides $\vec{U}_i$, the displacement λ_i of slide joint center from F_{l_i} along the slide, leg length L_i, and platform joint center position ${}^{P}P_i$ are all known, the unknowns are six components of the pose vector χ included in transformation matrix ${}^{B}_{P}[T]$ defined by Equation 3.20.

Therefore, in order to solve the forward kinematic problem, the set of six non-linear Equations (6.2) with six unknowns has to be solved. There is clearly no closed-form solution as mentioned before. So the Newton-Raphson method that has been used in [51][52][54] is applied to solve the forward kinematic problem.

For the planar rectangular tool path shown in Figure 5.14, the setpoint position profiles produced by the online interpolator for six axes are shown in Figure 5.18. Those set-point positions, the slide joint center positions λ_i, are the input of the forward kinematic problem, while the pose is the output of the forward kinematic problem .

6.6.3 Motion Error Source Analysis Results

As mentioned in chapter 3, the kinematic parameters seen by the interpolator include joint center positions on the six slides and platform, the leg lengths, and the slide orientation vectors. Any inaccuracy of those parameters might cause the pose inaccuracy and finally the motion positional errors. In order to figure out the influence of those error sources on the motion accuracy, the forward kinematic simulations considered seven cases of input parameters as follows: nominal parameters (case 1); slide joint 1 position with 0.05mm coordinates error (case 2); platform joint 1 position with 0.05mm coordinates error (case 3); leg length with 1mm error (case 4); slide 1 with 89 degree angle between its axis and the X-axis, meaning a 1 degree angular error (case 5); slide 1 with 89 degree angle between its axis and the Y-axis which also means a 1 degree angular error (case 6); and slide1 has 1 degree angular errors with both X and Y axis (case 7).

Under each case, the obtained motion path positions were plotted in 2D & 3D format and compared with the desired motion path. Its individual components were also plotted versus motion time as a 5 DOF tool path. With the nominal kinematic parameters, the forward kinematic simulation results for the desired rectangular motion path were plotted in Figures 6.12, 6.13 and 6.14. The maximum motion path errors for the seven cases were summarized in Table 6.1

During the simulation procedure, several other tool paths that had been run successfully online were used to generate the set-point position profiles. Those generated set-point position profiles were then input to the forward kinematics algorithm to try to predict the tool paths. Unfortunately, most of those attempts did not produce reasonable results. Some even did not converge. This verifies that the numerical forward kinematic algorithm based on the Newton-Raphson method cannot give all the pose solutions of the hexapod machine.

From the maximum error data shown in Table 6.1, on one hand, one can see that even with the nominal parameters in case 1, the maximum error is still 1mm, so the numerical forward method can not give an accurate solution, it only provide relative values about the pose errors caused by different error sources. On the other hand, the data shows that the joint position error, leg length error, and slide orientation error all affect

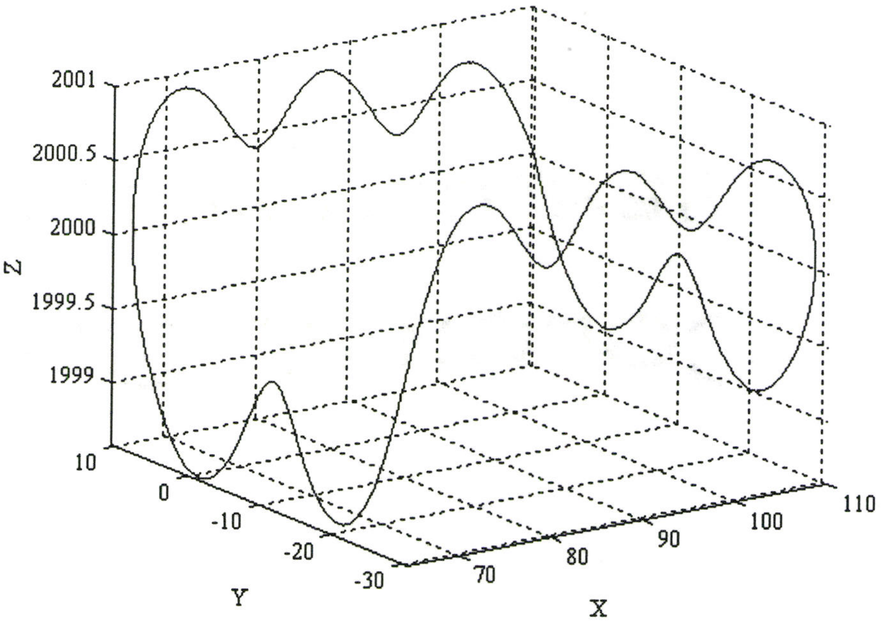


Figure 6.12 3D tool path of forward kinematic solution with nominal parameters

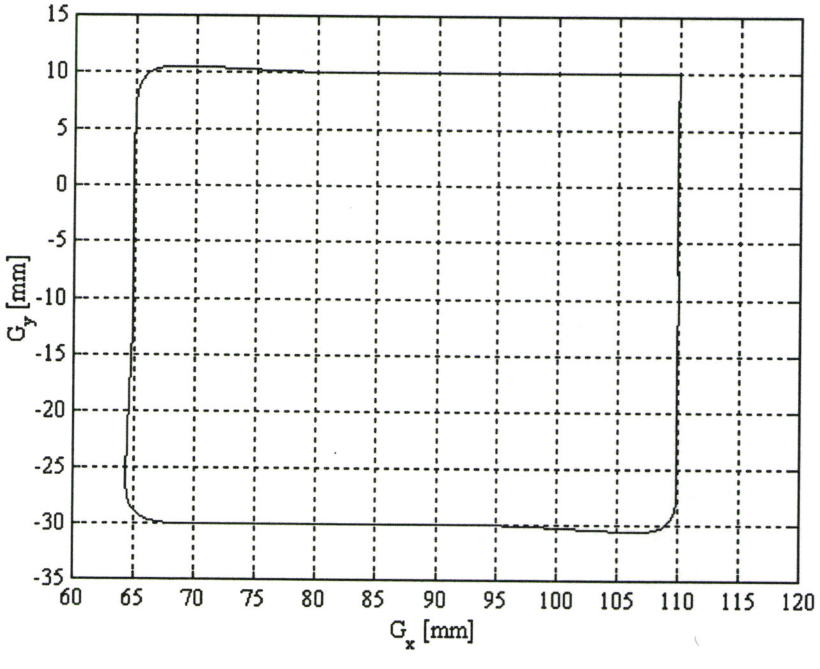


Figure 6.13 2D tool path of forward kinematic solution with nominal parameters

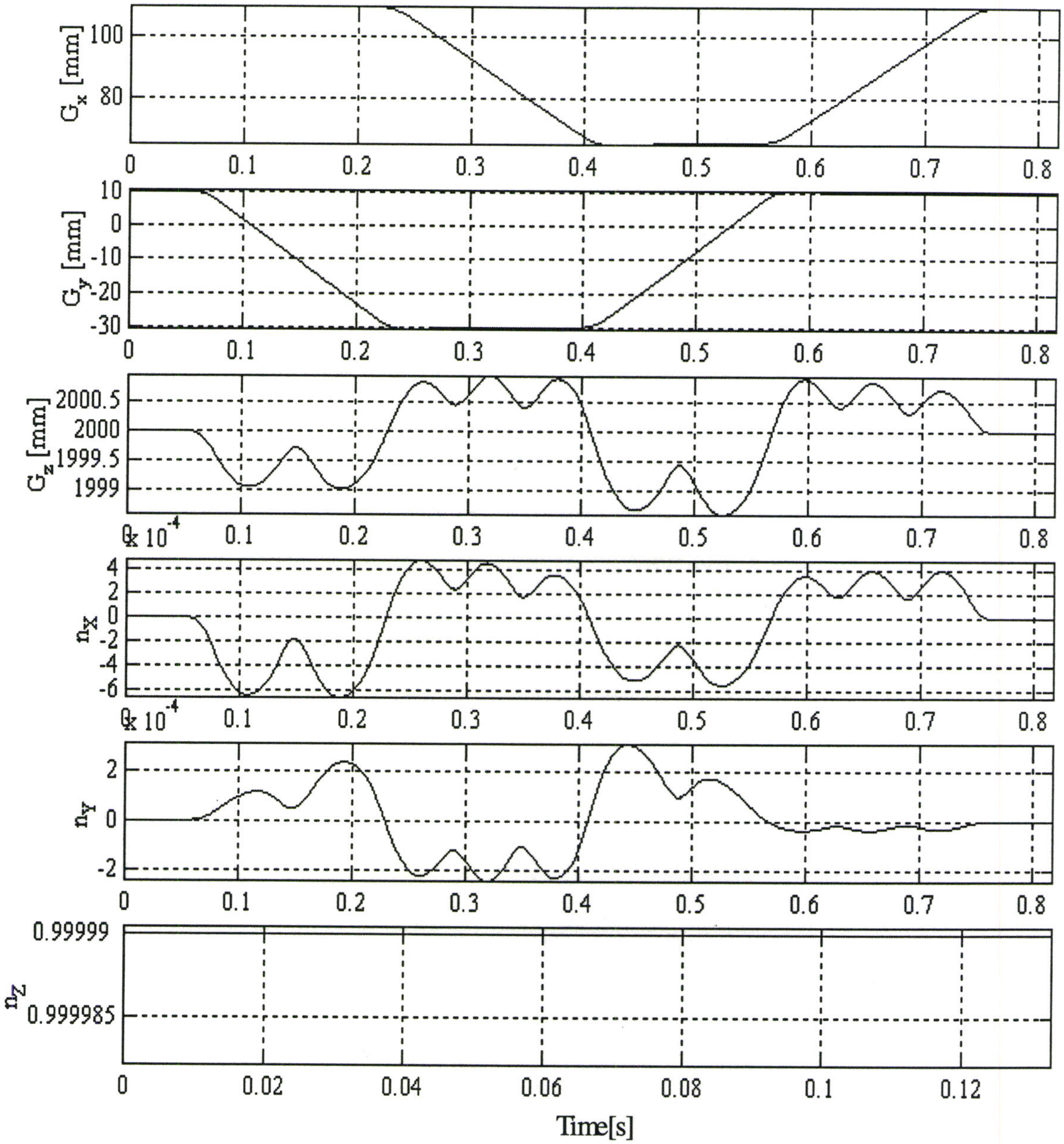


Figure 6.14 5 DOF tool path of forward kinematic solution with nominal parameters

the pose accuracy. Among these, the slide orientation error is the most significant factor influencing pose accuracy. Platform joint position error ranks second. Slide joint position error is the next error source of pose error. Leg length error is a minor factor influencing pose error.

Table 6.1 Pose errors for each case comparing with desired pose

Case	Max error in G_x	Max error in G_y	Max error in G_z	Max error in n_x	Max error in n_y	Max error in n_z
1	1mm	1mm	1mm	-6×10^{-4}	-6×10^{-2}	10^{-5}
2	2mm	1mm	20mm	-8×10^{-3}	6×10^{-3}	10^{-5}
3	10mm	5mm	140mm	0.03	-0.02	10^{-5}
4	1mm	2mm	3mm	-8×10^{-4}	4×10^{-4}	10^{-5}
5	7mm	2.5mm	80mm	0.02	-0.015	2×10^{-5}
6	40mm	350mm	420mm	0.75	0.25	10^{-5}
7	6mm	4mm	80mm	0.025	-0.017	2×10^{-5}

6.7 Conclusion

This chapter has presented all of the experiment measurements performed on the Tiara hexapod. The experimental verification of the motion functions shows that both axis motions and interpolated linear motions can realized for specified movements with the correct motion directions and shapes, as well as within the specified time period.

The homing position and repeatability measurements indicate all six axes can successfully reach their home positions with a repeatability less than 0.001in (25.4 microns). The linear motion positional repeatability along the X_B, Z_B and X_BY_B plane diagonal directions were determined to be less than 0.001in (25.4microns). The linear motion straightness accuracy was found to be very low with a maximum error of 0.108in (2.74mm). Forward kinematic simulation finds that inaccuracy of the joint center position, leg length and slide orientation affects the motion accuracy. Slide orientation error is the most significant error source and leg length error is the least.

Chapter 7

Conclusions & Recommendations

This thesis has detailed the improvement of kinematic model, improvement of the servo control system, the implementation of online interpolator and the experimental measurement of the Tiara hexapod. This chapter summarizes the achievement of this thesis and presents suggestions for future work.

7.1 Achievements

All of the objectives outlined in the introduction have been realized. The improvement and implementation of the kinematic model, servo control system and online interpolator have been successfully completed. Specifically, the following has been achieved:

1. The nominal model of the constructed Tiara hexapod was improved using Homogeneous Transformation Matrices, a geometrical model and physical measurements. Detailed derivations of the nominal kinematic model were broken into a number of different steps, following the assembling sequence of the machine. Parameters were provided at each step to allow easy modification, so the nominal model is fairly simple to update to the accurate model.

2. The servo control software and hardware configurations were improved to guarantee safe motion control of the machine. Each axis servo system of the machine was tuned to produce excellent position control results, with rise time varying from 0.15s to 0.167s and maximum following error from 3 to 9 encoder counts for the six axes. The improved control software provides functions to control the motion of the six axes of the machine simultaneously and safely. The improved and new functions include: axis jogging functions, closed loop homing functions, automatic running functions, safety watchdog functions, as well as brake control functions. Each of the functions was tested and produced the desired result.

3. The interpolator was successfully implemented online. It generates a smooth pure, or partial S-curve velocity profile and cubic position profile for each axis with the piecewise constant jerk and trapezoidal acceleration profiles that respect the dynamic limitations of the machine. It allows the machine to move in its workspace in a single move or a linear blended move motion. The interpolated motions were thoroughly tested with a maximum feedrate at 64mm/s and proven to reliably work as designed.

4. The basic performance of the machine was evaluated. Axis system motions and interpolated linear motions were verified and measured. The homing positions of all six axes were obtained with a repeatability less than 0.001in(25.4microns). The positional repeatability of linear motions along different tool path was less than 0.001in(25.4microns), limited by the dial gage resolution. While the linear motion positional accuracy was found to be very low with a maximum error of 0.108in (2.74mm). This is due to the inaccuracy of joint centre positions, leg length and slide orientation.

7.2 Recommendations for Future Work

Beyond the scope of this thesis, the following are recommendations for future research work related to exploring the full capabilities of the Tiara hexapod:

1. Improving signal isolation and electromagnetic shielding of the control system. Optical isolation needs to be applied between the servo card and the outside signal circuits. Electromagnetic disturbance to the spindle controller and the servo control system needs studying and correcting. In addition, a more powerful brakes system should be considered for replacing the current ones.

2. Kinematic calibration is the next critical step for fully exploring the capabilities of Tiara hexapod. Based on experimental measurement results in this thesis, the suitable metrological device for calibration is the spatial measurement device, e.g. a survey theodolite or a laser tracking system.

3. Applying advanced control techniques in the servo control system is one of the main aspects that will improve the machine performance. The unique layout of Tiara hexapod makes the load on each single axis vary at different poses, which represents a great challenge for controller design. The current fixed gains PID controller is only suitable for the testing phase of the machine. Future research work should be directed to implement more adaptive control techniques in the servo control system.

4. Enriching the functions of the interpolator. G code interpreter and the corresponding interpolation functions, such as circular & parametric curve interpolation, and look-ahead function need developing and implementing online.

Appendix A

Nominal Parameters of theTiara Hexapod

A.1 Slide System Parameters in the Machine Coordinate System

Table A.1 *F_L_slide_center*(mm)

Axis 0	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
813.2504	731.0461	-31.0461-	-813.2504	-82.2043	82.2043
-374.6089	-516.9911	-516.9911	-374.6089	891.6000	891.6000
0	0	0	0	0	0

Table A.2 *Slide_joint_base_vector* (mm)

Axis 0	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
-48.9737	-48.9737	48.9737	48.9737	0	0
28.2750	28.2750	28.2750	28.2750	-56.5500	-56.5500
51.5000	51.5000	51.5000	51.5000	51.5000	51.5000

Table A.3 *F_L_joint_base_center* (mm)

Axis 0	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
764.2767	682.0723	-682.0723	-764.2767	-82.2043	82.2043
-346.3339	-488.7161	-488.7161	-346.3339	835.0500	835.0500
51.5000	51.5000	51.5000	51.5000	51.5000	51.5000

Table A.4 *Slide_joint_vector* (mm)

Axis 0	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
-19.1592	-23.9692	23.9692	19.1592	-4.8100	4.8100
16.6156	8.2845	8.2845	16.6156	-24.9001	-24.9001
48.8041	48.8041	48.8041	48.8041	48.8041	48.8041

Table A.5 *F_l_temp* (mm)

Axis 0	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
745.1175	658.1031	-658.1031	-745.1175	-87.0143	87.0143
-329.7183	-480.4316	-480.4316	-329.7183	810.1499	810.1499
100.3041	100.3041	100.3041	100.3041	100.3041	100.3041

Table A.6 *H_L* (mm)

Axis 0	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
46	37	44.1	44.3	45.7	42.0

Table A.7 *F_l_z* (mm)

Axis 0	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
0	0	0	0	0	0
0	0	0	0	0	0
98.5	89.5	96.6	96.8	98.2	94.5

Table A.8 *F_l* (mm)

Axis 0	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
745.1175	658.1031	-658.1031	-745.1175	-87.0143	87.0143
-329.7183	-480.4316	-480.4316	-329.7183	810.1499	810.1499
198.8041	189.8041	196.9041	197.1041	198.5041	194.8041

Table A.9 *H_U*(mm)

Axis 0	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
47.1	36.9	37.8	35.7	35.1	36.1

Table A.10 *Lact* (mm)

Axis 0	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
721.9	741.1	733.1	735.0	734.2	736.9

Table A.11 F_u (mm)

Axis 0	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
745.1175	658.1031	-658.1031	-745.1175	-87.0143	87.0143
-329.7183	-480.4316	-480.4316	-329.7183	810.1499	810.1499
920.7041	930.9041	930.0041	932.1041	932.7041	931.7041

A.2 Platform System Parameters in the Platform Coordinate System

Table A.12 *Platform_pin_hole_location* (mm)

Axis 0	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
259.3842	64.7701	-64.7701	-259.3842	-194.6141	194.6141
74.9655	-262.1161	-262.1161	74.9655	187.1506	187.1506
-203.7450	-203.7450	-203.7450	-203.7450	-203.7450	-203.7450

Table A.13 *Platform_joint_base_vector*(mm)

Axis 0	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
0	0	0	0	0	0
0	0	0	0	0	0
-31.0700	-31.0700	-31.0700	-31.0700	-31.0700	-31.0700

Table A.14 *Platform_joint_base_center*(mm)

Axis 0	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
259.3842	64.7701	-64.7701	-259.3842	-194.6141	194.6141
74.9655	-262.1161	-262.1161	74.9655	187.1506	187.1506
-234.8150	-234.8150	-234.8150	-234.8150	-234.8150	-234.8150

Table A.15 *Platform_joint_vector*(mm)

Axis 0	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
19.4882	23.7573	-23.7593	-19.4882	4.2711	-4.2711
-16.1833	-8.7856	-8.7856	-16.1833	24.9689	24.9689
-48.8192	-48.8192	-48.8192	-48.8192	-48.8192	-48.8192

Table A.16 *Platform_joint_center* (mm)

Axis 0	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
278.8724	88.5294	-88.5294	-278.8724	-190.3431	190.3431
58.7822	-270.9017	-270.9017	58.7822	212.1195	212.1195
-283.6342	-283.6342	-283.6342	-283.6342	-283.6342	-283.6342

A.3 Leg length Parameters and Joints Axes Parameters

Table A.17 *Leg_Distance* (mm)

Axis 0	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
1418	1420	1418	1418	1420	1416

Table A.18 *Leg_Gap* (mm)

Axis 0	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
8.14	8.03	7.38	7.86	8.10	7.98

Table A.19 *Leg_length* (mm)

Axis 0	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
1596.14	1598.03	1595.38	1595.86	1598.10	1593.98

Table A.20 *Platform_ joint_axis1_vector*(mm)

Axis 0	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
0.5291	-0.4678	0.4678	-0.5291	-0.9970	0.9970
0.8457	-0.8811	-0.8811	0.8457	0.0354	0.0354
-0.0692	-0.0692	-0.0692	-0.0692	-0.0692	-0.0692

Table A.21 *Slide joint_axis1_vector*(mm)

Axis 0	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
-0.5291	0.4678	-0.4678	0.5291	0.9970	-0.9970
-0.8457	0.8811	0.8811	-0.8457	-0.0354	-0.0354
0.0692	0.0692	0.0692	0.0692	0.0692	0.0692

Table A.22 *Platform_joint_installation angles* (°)

Axis 0	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
32.2520	151.7755	151.7755	32.2520	87.9718	87.9718

Table A.23 *Slide_ joint_installation angles* (°)

Axis 0	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5
28.2245	147.7480	147.7480	28.2245	92.0282	92.0282

Appendix B

Modified IO Ports Configurations

Table B.1 Keyword and Mask code for input signals

Key word	Mask code	Key word	Mask code
MAX_LIM_0	0x00010000	MAX_LIM_3	0x00008000
MIN_LIM_0	0x80000000	MIN_LIM_3	0x00000400
FAULT_0	0x00040000	FAULT_3	0x00000800
MAX_LIM_1	0x00080000	MAX_LIM_4	0x00001000
MIN_LIM_1	0x00100000	MIN_LIM_4	0x00002000
FAULT_1	0x00200000	FAULT_4	0x00004000
MAX_LIM_2	0x00400000	MAX_LIM_5	0x10000000
MIN_LIM_2	0x00800000	MIN_LIM_5	0x20000000
FAULT_2	0x00000100	FAULT_5	0x40000000

Table B.2 Digital output bit function and connector pin mapping

Bit	Function	Connector/pin
A0	None	P1/47
A1	None	P1/45
A2	0 Amplifier enable	P1/43
A3	1 Amplifier enable	P1/41
A4	2 Amplifier enable	P1/39
A5	3 Amplifier enable	P1/37
A6	4 Amplifier enable	P1/35
A7	5 Amplifier enable	P1/33
D0	0-5 Brake enable	P2/31
D1	(none)	P2/29
D2	(none)	P2/27
D3	(none)	P2/25

Table B.3 Digital input bit function and connector pin mapping

Bit	Function	Connector/pin
C0	0 +limit switch	P1/15
C1	(none)	P1/13
C2	0 Amplifier fault	P1/11
C3	1 +limit switch	P1/9
C4	1 –limit switch	P1/7
C5	1 Amplifier fault	P1/5
C6	2 +limit switch	P1/3
C7	2 –limit switch	P1/1
B0	2 Amplifier fault	P1/31
B1	(none)	P1/29
B2	3 –limit switch	P1/27
B3	3 Amplifier fault	P1/25
B4	4 +limit switch	P1/23
B5	4 –limit switch	P1/21
B6	4 Amplifier fault	P1/19
B7	3 + limit switch	P1/17
D4	5 +limit switch	P2/23
D5	5 –limit switch	P2/21
D6	5 Amplifier fault	P2/19
D7	0 –limit switch	P2/17

Appendix C

Experiment Measurement Results

C.1 Homing Repeatability Measurements Results

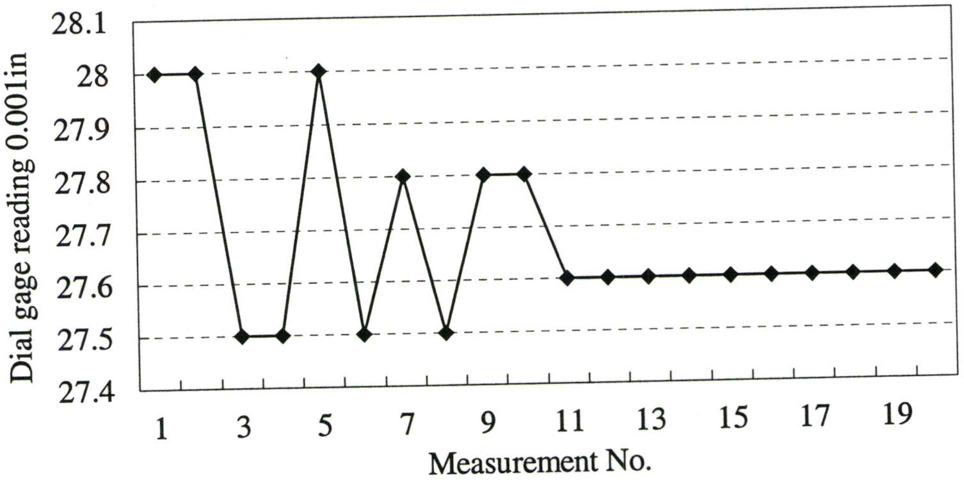


Figure C.1 Homing repeatability measurement results for Axis 0

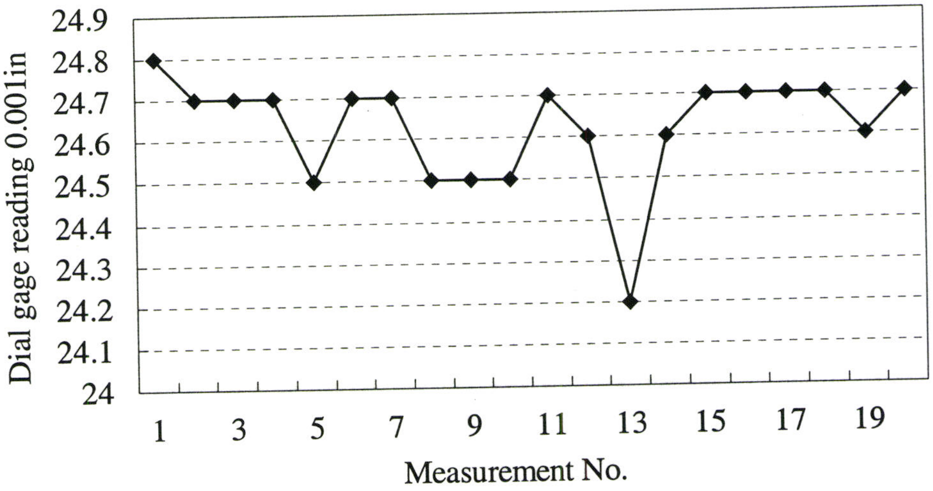


Figure C.2 Homing repeatability measurement results for Axis 2

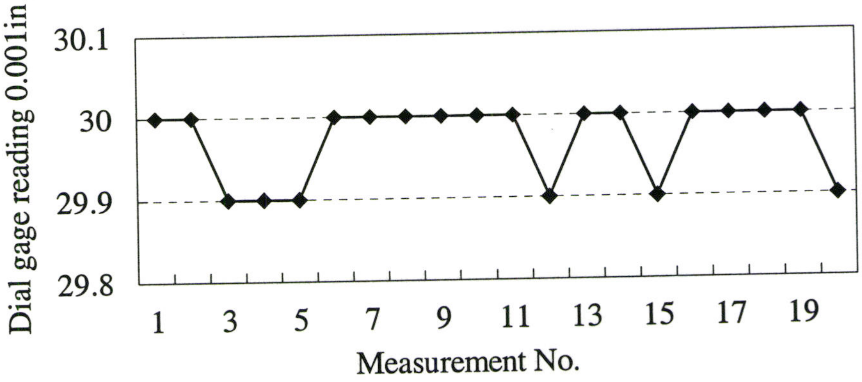


Figure C.3 Homing repeatability measurement results for Axis 3

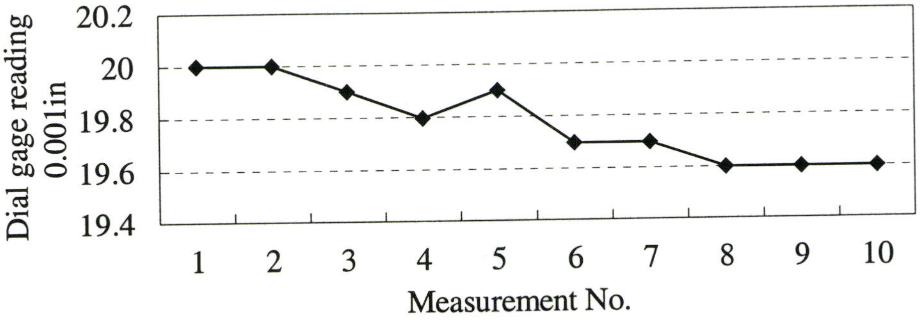


Figure C.4 Homing repeatability measurement results for Axis 5

C.2 Linear Motion Repeatability Measurements Results

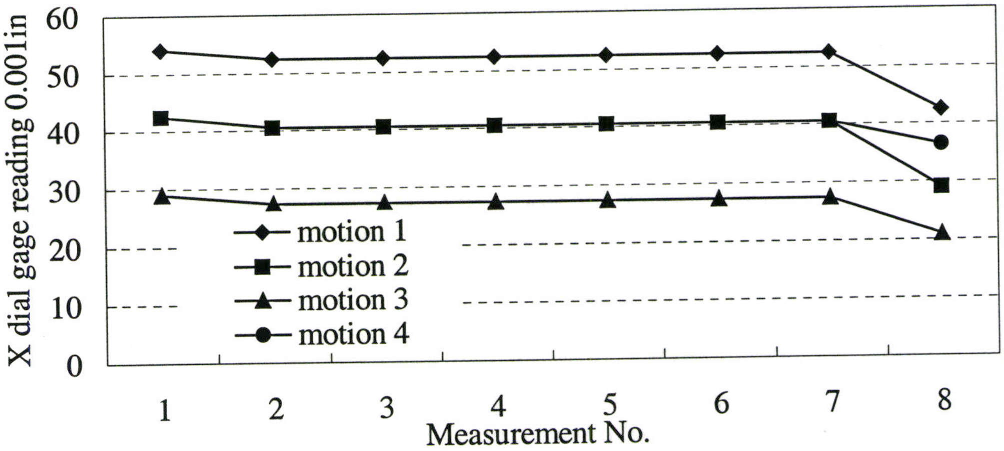


Figure C.5 X direction repeatability of Z_B direction motion measurements

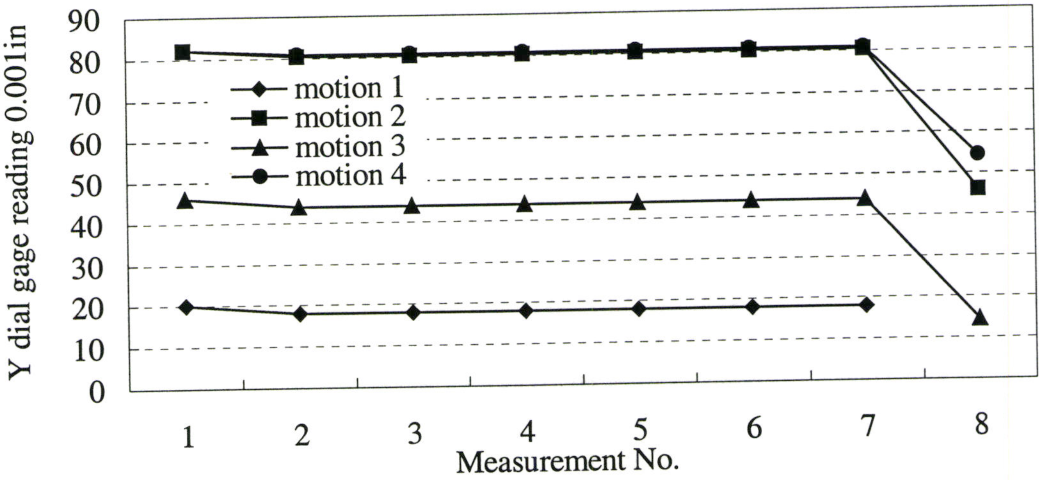


Figure C.6 Y direction repeatability of Z_B direction motion measurements

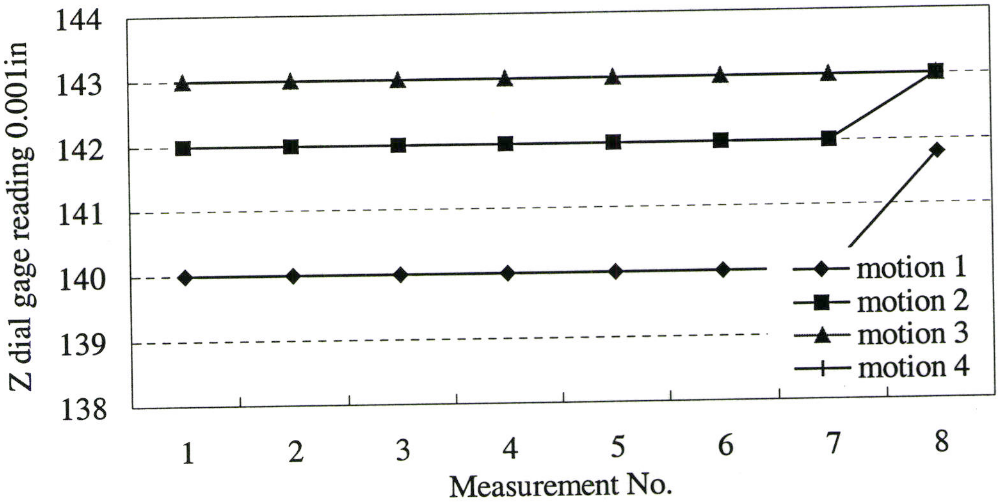


Figure C.7 Z direction repeatability of Z_B direction motion measurements

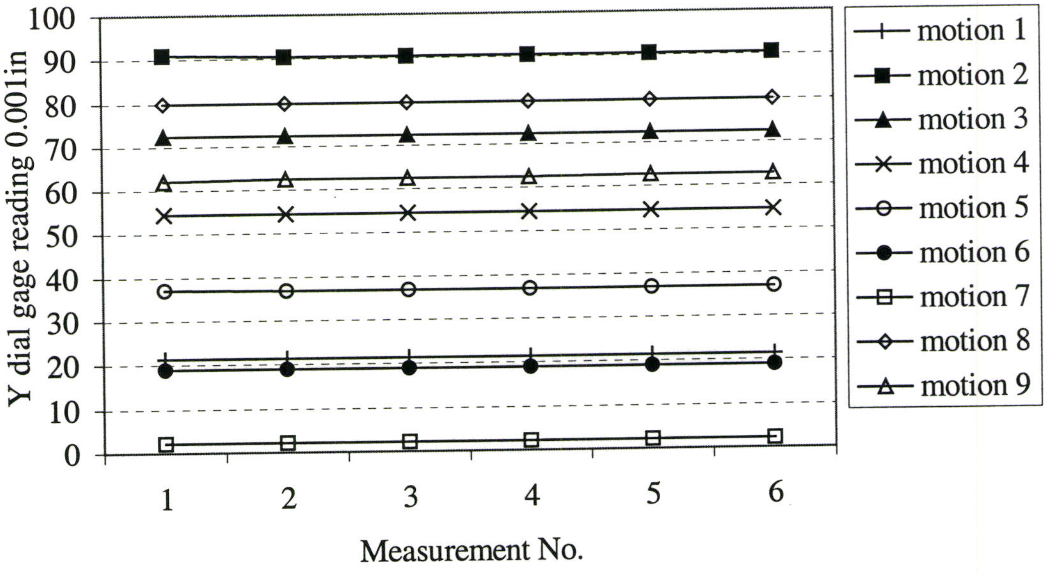


Figure C.8 Y direction repeatability of $X_B Y_B$ diagonal motion

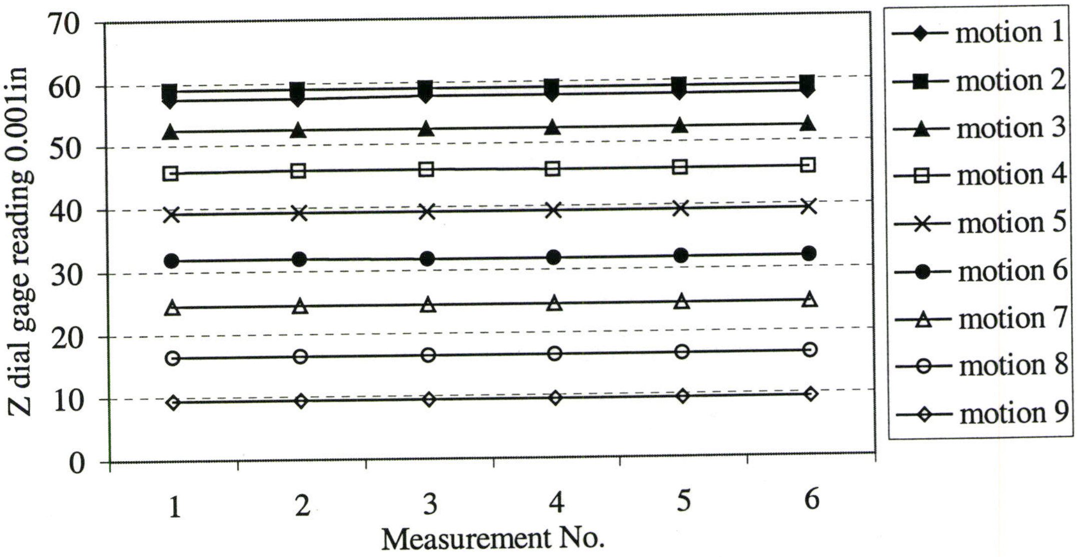


Figure C.9 Z direction repeatability of $X_B Y_B$ diagonal motion

References

[1] M. Weck, M. Giesler, A. Meylahn and D. Saimer. Parallel kinematics: the importance of enabling technologies. *Parallel Kinematic Machines Theoretical Aspects and Industrial Requirements*, 283-294, Springer, 1999.

[2] R.Neugebauer, F. Wieland, M. Schwaar and C. Hochmuth. Experiences with a hexapod-based machine tool. *Parallel Kinematic Machines Theoretical Aspects and Industrial Requirements*, 313-326, Springer, 1999.

[3] H. K. Tonshoff and H. Grendel. A systematic comparison of parallel kinematics. *Parallel Kinematic Machines Theoretical Aspects and Industrial Requirements*, 295-312, Springer, 1999.

[4] M. Weck and D. Staimer. Parallel kinematic machine tools-current state and future potentials. *Annals of the CIRP*, 51(2):1-13, 2002

[5] G. Prischow. Parallel kinematic machines (PKM) –limitations and new solutions, *Annals of the CIRP* , 49(1): 275-280, 2000.

[6] F. Resteriner, R. Neugebauer, S. spiewak and F. Wieland. Putting parallel kinematic machines (PKM) to productive work. *Annals of the CIRP* , 48(1): 345-350, 1999.

[7] J. Tlusty, J. Ziegert and S. Ridgeway. Fundamental comparison of the use of serial and parallel kinematics for machine tools. *Annals of the CIRP* , 48(1): 351-356, 1999.

[8] S. Sartori and G. Zhang. Geometric error measurement and compensation of

machines. *Annals of the CIRP* , 44(2): 599-609, 1995.

[9] R. Ramesh and M.A. Mannan, An. Poo. Error compensation in machine tools- a review Part I: geometric, cutting-force induced and fixture-dependant errors. *Int. J. of Machine Tools and Manufacture*, 40:1235-1256, 2000.

[10] O. Masory, J. Wang and H. Zhuang. On the accuracy of a Stewart platform- part 2 kinematic calibration and compensation. In *Proc. of the IEEE Int. Conf on Rob. & Aut.*, Atlanta, GA, 725-731,1993.

[11] G. Duelen and K. Schroer. Robot calibration-method and results. *Robotics & computer-integrated Manufacturing*, 8(4):223-231, 1991.

[12] W. K. Veitschegger and Ch. Wu. Robot calibration and compensation. *IEEE J. of Robotics and Automation*, 4(6): 643-656, Dec. 1988.

[13] J. Wang and O. Masory. On the accuracy of a Stewart platform- part 1 the effect of manufacturing tolerances. In *Proc. of the IEEE Int. Conf on Rob. & Aut.*, 1:114-120,1993.

[14] A. J. Patel and K. F. Ehmann. Volumetric error analysis of a Stewart platform based machine tool, *Annals of the CIRP* , 46(1): 287-290, 1997,

[15] H. Zhuang, O.Masory and J. Yan. Kinematic calibration of a Stewart platform using pose measurement obtained by a single Theodolite. *ISPE*, 329-334, 1995.

[16] H. Zhang, J. Yuan and O. Masory. Calibration of Stewart platforms and other parallel manipulators by minimizing inverse kinematic residuals. *J. of Robotic Systems*, 15(7): 395-405, 1998.

[17] J. A. Soons. On the geometric and thermal errors of a hexapod machine tool.

Parallel Kinematic Machines Theoretical Aspects and Industrial Requirements, 151-170, Springer, 1999.

[18] A.J.Wavering. Parallel kinematic machine research at NIST: past, present and future. *Parallel Kinematic Machines Theoretical Aspects and Industrial Requirements*, 17-32, Springer, 1999.

[19] J. Song and J. Mou. Error modeling and compensation for parallel kinematic machines. *Parallel Kinematic Machines Theoretical Aspects and Industrial Requirements*, 171-188, Springer, 1999.

[20] G.. Belforte, B. Bona, E. Canuto and F. Donati, etc. Coordinate measuring machines and machine tools self-calibration and error correction. *Annals of the CIRP*, 36(1):359-364, 1987.

[21] H. Zhuang. Self-calibration of parallel mechanisms with a case study on Stewart platforms. *IEEE Trans. on Rob. & Aut.*, 13(3): 387-397, June 1997.

[22] H. Zhuang, R. Sudhakar and Z. S. Roth. A new method for pose fitting from two 3d point sets and its application to robot localization. In *Proc. of the IEEE Int. Conf on Rob. & Aut.*, 655-660,1996.

[23] H. Zhuang and L. Liu. Determination of number of independent parameters for the self-calibration of parallel-link mechanisms. *Proceedings of the ASME Manufacturing Science and Engineering Division*, 8:699-703, 1998.

[24] H. Zhuang, L. Liu and O. Masory. Autonomous calibration of hexapod machine tools. *Manufacturing Science and Engineering*, 122: 140-149, Feb. 2000,

[25] U. Heisel. Precision requirements of hexapod machines and investigation Results.

Parallel Kinematic Machines Theoretical Aspects and Industrial Requirements, 131-150, Springer, 1999.

[26] J. C. Ziegert, B. Jokiel and C. -C. Huang. Calibration and self-calibration of hexapod machine tools. *Parallel Kinematic Machines Theoretical Aspects and Industrial Requirements*, 205-216, Springer, 1999.

[27] N. P. Powell, B. D. Whittingham and N. Z. Gindy. Parallel link mechanism machine tools: acceptance testing and performance analysis. *Parallel Kinematic Machines Theoretical Aspects and Industrial Requirements*, 327-344, Springer, 1999.

[28] O. Masory and J. Yan. Measurement of pose repeatability of Stewart platform. *J. of Robotic Systems*, 12(12) 821-832,1995.

[29] K. Lau, R. Hocken and L. Haynes. Robot performance measurements using automatic laser tracking techniques. *Robotics & computer-integrated Manufacturing*, 2(34) 227-236, 1985.

[30] J. P. Prenniger, M. Vincze and H. Gander. Contactless position and orientation Measurement of robot end-effectors. *IEEE Trans. on Rob. & Aut.*, 3(3): 180-185, June 1997.

[31] M. Hebsacker. Hexaglide 6 DOF and Triglide 3 DOF parallel manipulators. *Parallel Kinematic Machines Theoretical Aspects and Industrial Requirements*, 345-356, Springer, 1999.

[32] W. Khalil and S. Besnard. Self-calibration of stewart-gouph parallel robots without extra sensors. *IEEE Trans. on Rob. & Aut.*, 15(6): 1116-1121, Dec. 1999.

[33] H. Zhuang and Z. S. Roth. Method for kinematic calibration of Stewart platforms. *J.*

of robotics systems, 10(3) 391-405, 1993.

[34] Z. Huang. *Parallel Kinematic Robot Mechanisms Theory and Control*, Mechanical Industry Press, Beijing, China, 1997.

[35] R. Xu and S. Cai. Servo controller of virtual axes machine tools, *http://www.mirl.itri.org.tw/magazine/.*

[36] J. Destefani. Return of the Hexapods, *http:/www.sme.org/manufacturing engineering,* February 2003.

[37] Y. Koren and O. Masory. Reference-pulse circular interpolators for CNC systems. *J. of Engineering for industry*, 103: 131-136, Feb. 1981.

[38] O. Masory and Y. Koren, Reference–word circular interpolators for CNC systems. *J. of Engineering for industry*, 104:400-405, Nov. 1982.

[39] Y. Koren. *Computer Control of Manufacturing Systems*. McGraw-Hill Book Company, 1983

[40] Y. Koren, C. –C. Lo and M. Sphpitalni. CNC interpolators: algorithm and analysis, *Manufacturing Science and Engineering*, 1993.

[41] J. J Chou, and D.C. H. Yang. Command generation for three axes CNC machining. *J. of Engineering for industry*, 113:305-310, 1991.

[42] Y. Koren and R. –Sh. Lin. Five axis surface interpolators. *Annals of the CIRP*, 44(1): 379-382, 1995.

[43] J. P. Merlet, M.-W. Perng and D. Daney. Optimal trajectory planning of a 5 axis machine tool based on 6-axis parallel manipulator. *In Advances in Robot Kinemtics (ARK)*, 315-322, Piran, June 2000.

[44] C. C. Nguyen, S. S. Antrazi and J. -Y. Park. Trajectory planning and control of a Stewart platform-based end-effector with passive compliance for part assembly. *J. of Intelligent and Robotic Systems*, 6:263-281, 1992.

[45] *PMAC User's Manual –5*, Delta Tau Data System, Inc. March 1998.

[46] F. Pierrot and T. Shibukawac. From hexa to hexaM. *Parallel Kinematic Machines Theoretical Aspects and Industrial Requirements*. 357-364, Springer, 1999.

[47] B. S. El-Khasawneh and P. M. Ferreira. The tetrahedral tripod. *Parallel Kinematic Machines Theoretical Aspects and Industrial Requirements*, 419-430, Springer, 1999.

[48] R. W. Teltz and M.A. Elbestawi. Design basis and implementation of an open architecture machine tool controller. *Manufacturing Science and Engineering*, 1997.

[49] R.W. Teltz. *Open Architecture Control for Intelligent Machining Systems*. PhD thesis, McMaster University, Hamilton, Ontario, 1998.

[50] M. J. Colella. *Development of an Open Architecture Multi-Robot Controller*, Master thesis, McMaster University, Hamilton, Ontario, 1997.

[51] F. Audren, *Optimization of the Tiara Hexapod. Technical Report*, IMMRC of McMaster University, Hamilton, Ontario,1999

[52] Dan N. Centea, etc. Development of the Tiara hexapod-a machine tool based on parallel kinematic structures. *Manufacturing Science and Engineering*, 10:809-878, 1999.

[53] P. A. Meyer. *Kinematic modeling, accuracy analysis and calibration of a hexapod machine tool*, Master thesis, McMaster University, Hamilton, Ontario, 2000.

[54] H. Lacheray. *Motion interpolation and control of a hexapod machine tool*. Master