THE CATHOLIC UNIVERSITY OF AMERICA
DEPARTMENT OF ELECTRICAL ENGINEERING

TRAJECTORY PLANNING AND CONTROL OF A 6 DOF MANIPULATOR WITH STEWART PLATFORM-BASED MECHANISM

Charles C. Nguyen
Principal Investigator and Associate Professor
and
Sami Antrazi
Graduate Research Assistant

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

July 1990

SUMMARY OF THE REPORT

This report presents the research results obtained from the research grant entitled "Control of Robot Manipulator Compliance," funded by the Goddard Space Flight Center (NASA) under a research grant with Grant Number NAG 5-780, for the period between February 1st, 1990 and July 31, 1990.

This report deals with the trajectory planning and control of a robot manipulator that has 6 degrees of freedom and was designed based on the mechanism of the Stewart Platform. First the main components of the manipulator will be described and its operation will be explained. We then briefly present the solutions for the forward and inverse kinematics of the manipulator. After that, two trajectory planning schemes will be developed using the manipulator inverse kinematics to track straight lines and circular paths. Finally experiments conducted to study the performance of the developed planning schemes in tracking a straight line and a circle will be presented and discussed.

1 Introduction

Successful robotic assembly of parts requires that the robot end-effector be able to perform very precise motion. As a result, research effort has been enormously spent [1,12] to study the application of closed-kinematic chain (**CKC**) mechanism in the design of robot manipulators performing high precision motion because CKC mechanism generally provides better positioning capability than open-kinematic chain (**OKC**) mechanism. CKC mechanism was first implemented in the design of the Stewart Platform [1] originally intended for simulating aircraft motion. Later, the Stewart platform attracted considerable interest of robotic researchers and was proposed in the design of several robot manipulators and end-effectors [2,8]. Hunt [2,3] developed numerous structural designs for robot manipulators using the mechanism of the Stewart Platform which initiated intensive research in [4,5]. Sugimoto and Duffy [6] employed the theory of linear algebra and screw systems to obtain a general method which describes the instantaneous link motion of a single closed-loop mechanism. Yang and Lee [7] studied the feasibility of manipulators designed based on the Stewart Platform whose inverse dynamics and kinematics were later investigated by Do and Yang [8] and Sugimoto [9]. Nguyen and Pooran applied the Lagrangian formulation to derive dynamical equations for a six-degree-of-freedom (**DOF**) CKC manipulator [10] whose kinematics was also developed in [11]. Later, they derived a learning-based control scheme for the above manipulator to perform repetitive tasks [12].

A Stewart platform-based manipulator was designed and built at the Goddard Space Flight Center (**GSFC**) [13] to serve as a testbed for evaluating the feasibility of autonomous assembly of parts in space. In this report, we first describe the main components of the manipulator and explain its operation. We then briefly review the forward and inverse kinematics of the manipulator. After that, a Cartesian trajectory planning scheme is developed and employed with a trajectory control scheme using the manipulator inverse kinematics. Finally experiments conducted to study the performance of the developed trajectory planning and control scheme are presented and discussed.

2 The 6 DOF Robot Manipulator

This section is devoted to briefly describe the main components of the manipulator and present the solutions for its inverse and forward kinematics.

2.1 Description of Hardware

Figures 1 and 2 show the robot manipulator which was designed based on the Stewart Platform mechanism and mainly consists of a lower base platform, an upper payload platform, a compliant platform, a gripper and six linear actuators. The movable payload platform is supported above the stationary base platform by six axially extensible rods with ballnuts and ballscrews providing the extensibility. Stepper motors were selected

to drive the ballscrews to extend or shorten the actuator lengths whose variations will in turn produce the motion of the payload platform and consequently the motion of a gripper attached to the compliant platform. Each end of the actuator links is mounted to the platforms by 2 rotary joints whose axes intersect and are perpendicular to each other. Passive compliance is provided through the compliant platform, which is suspended from the payload platform by six spring-loaded pistons arranged in the Stewart Platform mechanism. The compliance is passively provided by permitting strain on two opposing springs acting in the pistons. Thus the pistons are compressed and extended when resistive and gravitational forces are applied on the gripper. The rotation of each stepper motor is controlled by sending out proper commands to an indexer which then transmits proper pulse sequences to the stepper motor drive. Therefore, precise gripper motion can be produced by properly controlling the motions of six manipulator legs. Figure 2 presents the planning and control scheme employed to control the motion of the manipulator gripper. A Cartesian path specified with desired starting and ending velocities and accelerations is converted into 6 Cartesian trajectories using a Cartesian trajectory planning scheme. Then based upon the desired Cartesian trajectories, joint-space trajectories will be determined by a planner which sends proper commands through the RS232 port of a personal computer to the indexers. The indexer will then transmit pulses to the stepper motor drives where microstepping permits each revolution (360^0) of the stepper motor to be equivalent to 25,000 steps. Therefore the drive rotates the stepper motor one angular increment of $\frac{360^0}{25,000} = 0.0144^o$, each time it receives one step pulse. Furthermore, through the linear motion converter system consisting of the ballnut and the ballscrew, each angular increment (=1step) is converted into 8 μ-inches of linear translation of the manipulator leg.

2.2 Solution of the Inverse Kinematics

The lengths of the manipulator legs are selected as joint variables. To define the Cartesian variables we proceed to assign coordinate frame $\{\mathbf{A}\}$ to the movable payload platform and $\{\mathbf{B}\}$ to the base platform as shown in Figure 2. The position and orientation of Frame $\{\mathbf{A}\}$ with respect to Frame $\{\mathbf{B}\}$ are selected as the Cartesian variables in the sense that the position of Frame $\{\mathbf{A}\}$ is the position of its origin with respect to Frame $\{\mathbf{B}\}$. Now denoting the angle between AA_i and $\mathbf{x}_A$ by λ_i, and the angle between BB_i and $\mathbf{x}_B$ by Λ_i for i=1,2,...,6, we have obtain $\Lambda_i = 60(i-1)^o$; $\lambda_i = 60(i-1)^o$ for i=1,3,5 and $\Lambda_i = \Lambda_{i-1} + \theta_B$; $\lambda_i = \lambda_{i-1} + \theta_A$ for i =2,4,6. Furthermore, if Vector ${}^A\mathbf{a}_i = (a_{ix}\ a_{iy}\ a_{iz})^T$ describes the position of the attachment point A_i with respect to Frame $\{\mathbf{A}\}$, and Vector ${}^B\mathbf{b}_i = (b_{ix}\ b_{iy}\ b_{iz})^T$ the position of the attachment point B_i with respect to Frame $\{\mathbf{B}\}$, then they can be written as ${}^A\mathbf{a}_i = [\ r_A cos(\lambda_i)\ \ r_A sin(\lambda_i)\ \ 0\ ]^T$ and ${}^B\mathbf{b}_i = [\ r_B cos(\Lambda_i)\ \ r_B sin(\Lambda_i)\ \ 0\ ]^T$ for i=1,2,...,6 where r_A represents the radius of the payload platform and r_B that of the base platform.

We proceed to consider the vector diagram for an ith actuator given in Figure 4. The length vector ${}^B\mathbf{q}_i = (q_{ix}\ q_{iy}\ q_{iz})^T$, expressed with respect to Frame $\{\mathbf{B}\}$ can be

computed by

$$^{B}\mathbf{q}_i = {}^{B}\mathbf{a}_i - {}^{B}\mathbf{b}_i \tag{1}$$

where Vector $^{B}\mathbf{a}_i$ and Vector $^{B}\mathbf{b}_i$ describe the positions of A_i and B_i, respectively both in terms of Frame $\{\mathbf{B}\}$. However, $^{B}\mathbf{a}_i$ can be computed by

$$^{B}\mathbf{a}_i = {}^{B}_{A}\mathbf{R}\ {}^{A}\mathbf{a}_i + {}^{B}\mathbf{d} \tag{2}$$

where $^{B}_{A}\mathbf{R}$ is the matrix representing the orientation of Frame $\{\mathbf{A}\}$ with respect to Frame $\{\mathbf{B}\}$ and Vector $^{B}\mathbf{d}$ contains the Cartesian coordinates x, y, z of the origin, A of Frame $\{\mathbf{A}\}$ with respect to Frame $\{\mathbf{B}\}$ such that $^{B}\mathbf{d} = [\ x \quad y \quad z\ ]^T$.

Now substituting (2) into (1) and using $l_i = \sqrt{q_{ix}^2 + q_{iy}^2 + q_{iz}^2}$, we obtain after intensive simplifications:

$$\begin{aligned} l_i^2 &= x^2 + y^2 + z^2 + r_A^2 + r_B^2 + 2(r_{11}a_{ix} + r_{12}a_{iy})(x - b_{ix}) \\ &\quad +2(r_{21}a_{ix} + r_{22}a_{iy})(y - b_{iy}) + 2(r_{31}a_{ix} + r_{32}a_{iy})z - 2(xb_{ix}), \end{aligned} \tag{3}$$

for i=1,2,...,6 where r_{ij} for i,j=1,2,3 are the elements of the matrix $^{B}_{A}\mathbf{R}$.

Equation (3) represents the solution to the inverse kinematic problem in the sense that for a given Cartesian configuration, composed of the position and orientation specified by $^{B}\mathbf{b}_i$ and $^{B}_{A}\mathbf{R}$, respectively, the actuator lengths l_i for i=1,2,...,6, can be computed using (3).

2.3 Solution of the Forward Kinematics

The forward kinematics concerns with the determination of the Cartesian position and orientation of the payload platform when the actuator lengths l_i for i=1,2,...,6, are given. Consequently, the forward kinematic problem can be formulated as to find x, y, z (position) and r_{ij} (orientation) for i=j=1,2,3 to satisfy Equation (3) for a given set of l_i for i=1,2,...,6. Thus the problem is reduced to solving 6 highly nonlinear simultaneous equations with 6 unknowns, 3 of which represent the orientation. A closed-form solution does not generally exist, which leads us to seek an iterative numerical method to solve the above set of nonlinear equations. One widely used technique for solving nonlinear equations is the Newton-Raphson method, which is employed in this report to solve the forward kinematic problem.

3 The Trajectory Planning Schemes

Two trajectory planning schemes developed to control the motion of the robot manipulator gripper are presented in this section. The first scheme is designed for tracking straight lines while the second for arbitrary paths.

3.1 The Straight-Line Trajectory Planning Scheme

The indexer has two main modes of operation: the *normal mode* and the *continuous mode*. In the normal mode, based on the information about the velocity v_f, acceleration a, and the distance to be traveled Δ_l, the indexer will determine the appropriate leg velocity profiles which are either a trapezoid or a triangle depending on the relationship between the given information. For simplicity, the straight-line planning scheme will only utilize the trapezoidal profile which is illustrated in Figure 5 where t_a, t_c, and t_d denote the acceleration time, the constant velocity time, and the deceleration time, respectively. In addition the indexer requires that $t_a = t_d$. By inspection, we found that $\Delta_l = v_f(t_c + t_d)$ and $v_f = at_a$. To track a path in a 3-dimensional space, the positions of x- y- and z-coordinates must always be linearly related to each other anytime during the tracking. Intuitively, if the leg displacements are planned such that their velocities are linearly related to each other, then the resulting Cartesian motion should be a linear path. Computer simulation which utilized the manipulator forward kinematics and was performed to verify the above fact has agreed with our intuition. The following algorithm facilitates the trajectory planning for straight lines.

Algorithm 1: Straight Line Trajectory Planning

1. Use the manipulator inverse kinematics to compute the leg lengths corresponding to the starting point P_s and the final point P_f of the straight line, namely l_is and l_if for i=1,2,...,6.

2. Compute $\Delta_{li} = l_if - l_is$ for i=1,2,...,6 and find Δ_{lk} whose absolute value is the biggest.

3. Select a_k and v_fk for the k-th leg such that $a_k \leq a_max$; and $v_fk \leq v_max$; $v_fk \leq a_k\sqrt{|\Delta_{lk}|}$ to ensure trapezoidal profile where a_max and v_max denote the maximum acceleration and velocity of the stepper motor, and then compute $t_a = \frac{v_fk}{a_k} = t_d$ and $t_c = \frac{\Delta_{lk}}{v_fk} - t_a$.

4. For i$\neq$k; i=1,2,..,6 compute $a_i = \frac{\Delta_{li}}{t_a(t_a+t_c)}$ and $v_fi{=}t_aa_i$.

3.2 The Trajectory Planning Scheme For Arbitrary Curves

In the *continuous mode* of operation, the indexer requires the acceleration a, final velocity v_f and the direction of the rotation (direction of the linear displacement) of the stepper motor. The stepper motor will accelerate to the velocity v_f and continue to run at this velocity until new velocity and acceleration are given in the same direction of rotation. The current planning mainly consists of dividing a space curve into n segments and then planning the velocity profiles of the manipulator legs in the continuous mode

so that each segment will be reached within a specified time. The following algorithm is proposed to plan for arbitrary space curves.

Algorithm 2: Arbitrary Curve Trajectory Planning

1. Divide the space curve into n segments.
2. Use the manipulator inverse kinematics to compute the leg lengths corresponding to each segment point on the curve, namely l_ij for i=1,2,...,6 (leg number) and j=1,2,...,n+1 (segment point number)
3. Compute $\Delta_{ij} = l_i, j+1 - l_i, j$ for i=1,2,...,6 and j=1,2,...n.
4. For each segment, select an appropriate travel time t_j for j=1,2,...,n, and compute the corresponding acceleration and final velocity at the end of each segment.

In general, the travel times for the segments are constant and equal to each other during the tracking of curves which do not require the change of leg direction. However when direction of any leg has to change, the travel time can be selected efficiently using the *look ahead method.* Using this method, the algorithm looks at the next segment point and determine if any change in leg direction is necessary. For example if the direction of a leg requires direction change, then its travel time will be recomputed to ensure that the velocity at the end of the segment will be zero to allow direction change. After that, the recomputed travel time will be set for the remaining legs for the next two segments. Finally the travel time of all legs will be set back to the old value before the leg direction change occurs. The above process can be repeated any time a leg direction change is necessary.

4 Experimental Study

This section presents the results of experiments conducted to evaluate the performance of the developed trajectory planning schemes. In particular, Algorithm 1 was used to track a straight line in the x-y plane of the base frame {B}, which is described by $y(t) = x(t)$. The required profiles of the manipulator legs required to track the straight line were determined before the tracking using Algorithm 1 and are illustrated in Figure 6. The path that the manipulator gripper actually tracked, is presented in Figure 7 together with the desired path. Figure 7 shows excellent tracking capability with insignificant errors. Algorithm 2 was applied to compute the leg velocity profiles shown in Figure 8 to track a circle in the x-y plane of the {B} with a radius of 0.8 inches. Figure 9 shows the desired path plotted together with the path the manipulator actually tracked. The average errors for x- y- and z-coordinates were computed to be 1.096×10^{-4} inches, 2.326×10^{-4} inches, and 2.974×10^{-5} inches, respectively.

5 Conclusion

The problem of trajectory planning for a Stewart Platform-based manipulator was addressed in this report. We first described the main components of the manipulator and briefly presented the development of its inverse and forward kinematics. We then employed the normal and continuous modes of operation of the stepper motor indexer to derive two trajectory planning schemes for tracking straight paths and circular paths, respectively. Experiments performed to evaluate the developed algorithms showed excellent tracking results. Future research will focus on investigating the application of the algorithms to carry out assembly tasks such as mating space connectors with or without force and position feedback. In addition, development of a 6 DOF force sensor system based on the mechanism of the Stewart Platform is under way.

References

[1] Stewart, D., "A Platform with Six Degrees of Freedom," *Proc. Institute of Mechanical Engineering*, vol. 180, part 1, No. 5, pp. 371-386, 1965-1966.

[2] Hunt, K. H., *Kinematic Geometry of Mechanisms*, Oxford University, London 1978.

[3] Hunt, K. H., "Structural Kinematics of in-parallel-actuated Robot Arms," *Trans. ASME, J. Mech., Transmis., Automa. in Des.*, Vol. 105, pp. 705-712, 1983.

[4] Fichter, E.F. and MacDowell, E.D., "A Novel Design for a Robot Arm," *ASME Int. Computer Technology Conference*, San Francisco, pp. 250-256, 1980

[5] Fichter, E.F., "A Stewart Platform-Based Manipulator: General Theory and Practical Construction," *Int. Journal of Robotics Research*, pp. 157-182, Summer 1986

[6] Sugimoto, K. and Duffy, J., "Application of Linear Algebra to Screw Systems," *Mech. Mach. Theory*, Vol. 17, No. 1, pp. 73-83, 1982.

[7] Yang, D. C. and Lee, T. W., "Feasibility Study of a Platform Type of Robotic Manipulators from a Kinematic Viewpoint," *Trans. ASME Journal of Mechanisms, Transmissions, and Automation in Design*, Vol. 106. pp. 191-198, June 1984.

[8] Do, W.Q.D. and Yang, D.C.H., "Inverse Dynamics of a Platform Type of Manipulating Structure," *ASME Design Engineering Technical Conference*, Columbus, Ohio, pp. 1-9, 1986.

[9] Sugimoto, K., "Kinematic and Dynamic Analysis of Parallel Manipulators by Means of Motor Algebra," *ASME Journal of Mechanisms, Transmissions, and Automation in Design*, pp. 1-5, Dec. 1986.

[10] Nguyen, C.C., Pooran, F.J., "Dynamic Analysis of a 6 DOF CKCM Robot End-Effector for Dual-Arm Telerobot Systems," *Journal of Robotics and Autonomous Systems*, Vol. 5, pp. 377-394, 1989.

[11] Nguyen, C.C., Pooran, F.J., "Kinematic Analysis and Workspace Determination of a 6 DOF CKCM Robot End-Effector" *Journal of Mechanical Technology*, Vol. 20, pp. 283-294, 1989.

[12] Nguyen, C. C., Pooran, F.J., "Learning-Based Control of a Closed-Kinematic Chain Robot End-Effector Performing Repetitive Tasks", to appear in *Journal of Microcomputer Applications*, Vol. 8, No. 3, 1989.

[13] Premack, Timothy et al, "Design and Implementation of a Compliant Robot with Force Feedback and Strategy Planning Software," *NASA Technical Memorandum 86111*, 1984.

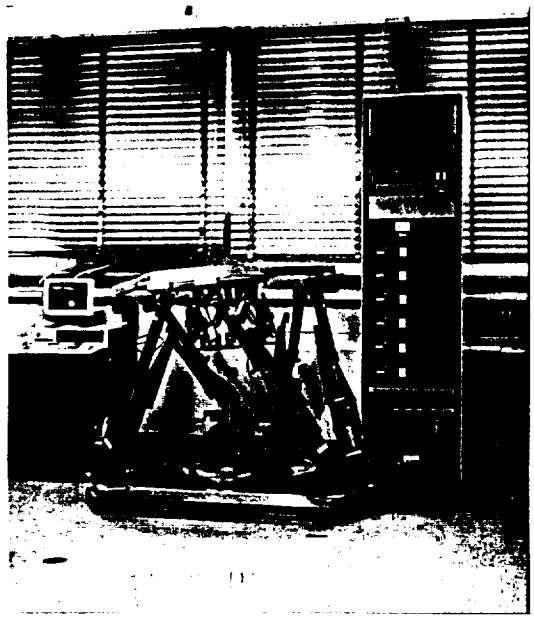


Figure 1: The robot manipulator.

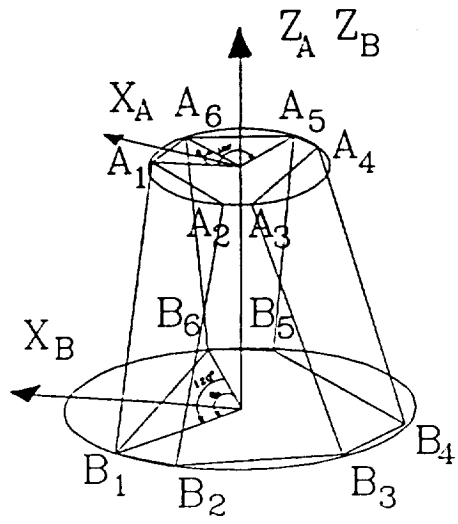


Figure 2: Frame assignment for platforms.

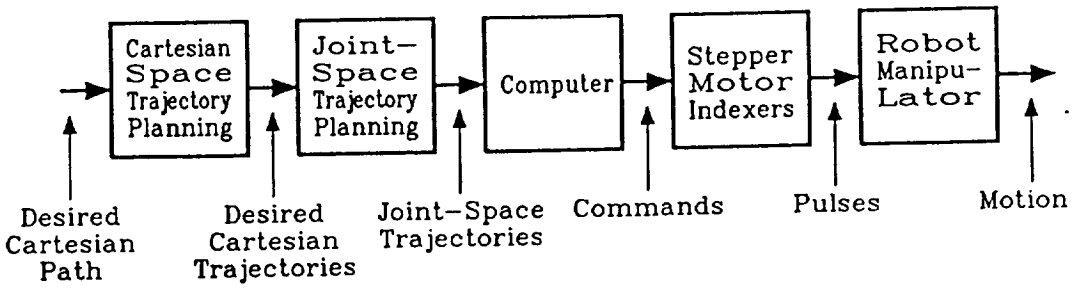


Figure 3: Trajectory planning and control of the robot manipulator.

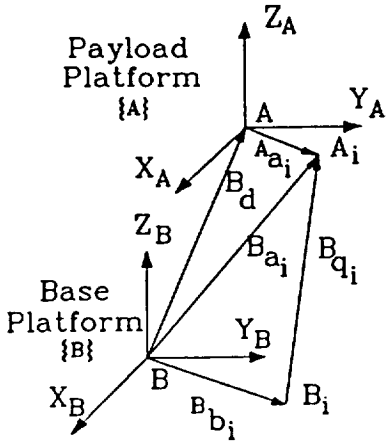


Figure 4: Vector diagram for the ith leg.

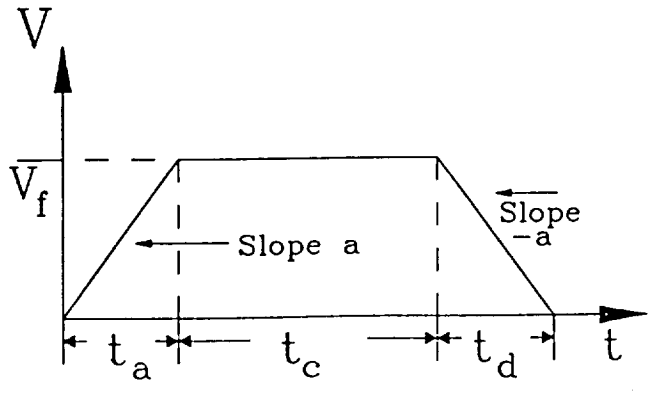


Figure 5: Trapezoidal velocity profile.

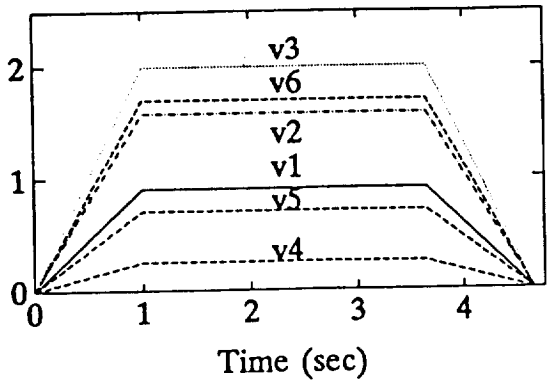


Figure 6: Leg velocity profiles (Alg. 1).
vertical axis=velocity (rev/sec)

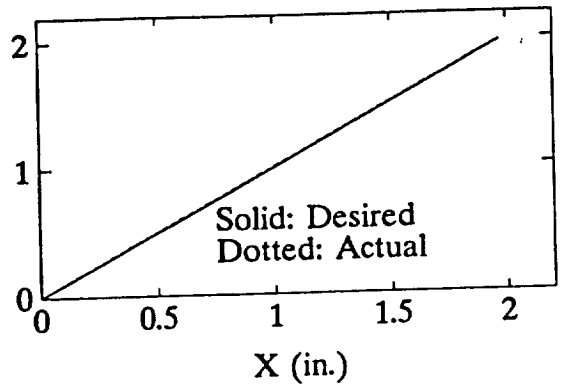
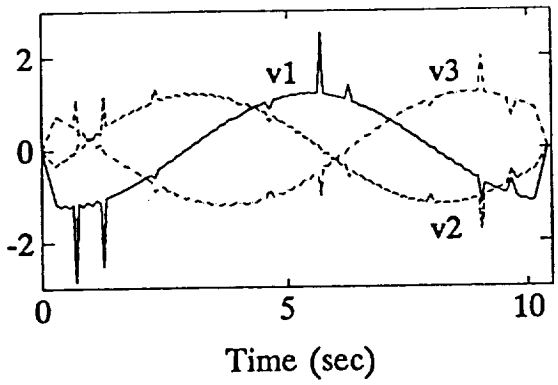


Figure 7: Tracking a straight line.
vertical axis=y-axis (inch)



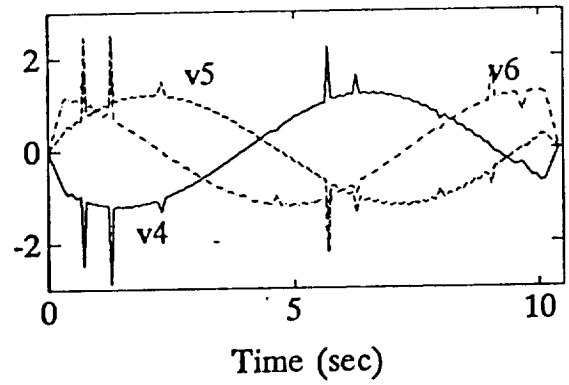


Figure 8: Velocity profiles for 6 legs (Algorithm 2).
vertical axis=velocity (revolution/sec)

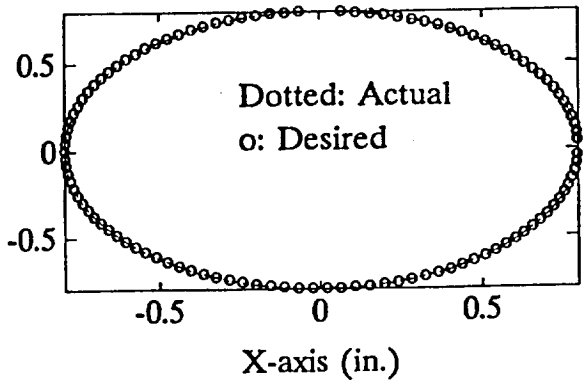


Figure 9: Tracking a circular path.