Parallel Manipulators

Towards New Applications

Parallel Manipulators

Towards New Applications

Edited by
Huapeng Wu

I-Tech

Published by I-Tech Education and Publishing

I-Tech Education and Publishing
Vienna
Austria

www.i-techonline.com
Additional copies can be obtained from:
publication@ars-journal.com

First published April 2008
Printed in Croatia

A catalogue record for this book is available from the Austrian Library.
Parallel Manipulators, Towards New Applications, Edited by Huapeng Wu
p. cm.
ISBN 978-3-902613-40-0
1. Parallel Manipulators. 2. New Applications. I. Huapeng Wu

Preface

In recent years, parallel kinematics mechanisms have attracted a lot of attention from the academic and industrial communities due to potential applications not only as robot manipulators but also as machine tools. Generally, the criteria used to compare the performance of traditional serial robots and parallel robots are the workspace, the ratio between the payload and the robot mass, accuracy, and dynamic behaviour. In addition to the reduced coupling effect between joints, parallel robots bring the benefits of much higher payload-robot mass ratios, superior accuracy and greater stiffness; qualities which lead to better dynamic performance. The main drawback with parallel robots is the relatively small workspace.

A great deal of research on parallel robots has been carried out worldwide, and a large number of parallel mechanism systems have been built for various applications, such as remote handling, machine tools, medical robots, simulators, micro-robots, and humanoid robots.

This book opens a window to exceptional research and development work on parallel mechanisms contributed by authors from around the world. Through this window the reader can get a good view of current parallel robot research and applications.

The book consists of 23 chapters introducing both basic research and advanced developments. Topics covered include kinematics, dynamic analysis, accuracy, optimization design, modelling, simulation and control of parallel robots, and the development of parallel mechanisms for special applications. The new algorithms and methods presented by the contributors are very effective approaches to solving general problems in design and analysis of parallel robots.

The goal of the book is to present good examples of parallel kinematics mechanisms and thereby, we hope, provide useful information to readers interested in building parallel robots.

Editor

Huapeng Wu
Institute of Mechatronics and Virtual Engineering
Lappeenranta University of Technology
Finland

Contents

1

Control of Cable Robots for Construction Applications

Alan Lytle, Fred Proctor and Kamel Saidi
National Institute of Standards and Technology
United States of America

1. Introduction

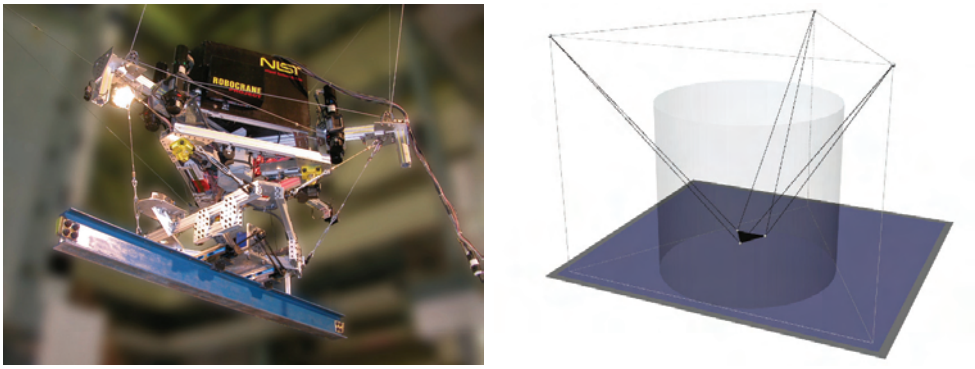
The Construction Metrology and Automation Group at the National Institute of Standards and Technology (NIST) is conducting research to provide standards, methodologies, and performance metrics that will assist the development of advanced systems to automate construction tasks. This research includes crane automation, advanced site metrology systems, laser-based 3D imaging, calibrated camera networks, construction object identification and tracking, and sensor integration and process control from Building Information Models. The NIST RoboCrane has factored into much of this research both as a robotics test platform and a sensor/target positioning apparatus. This chapter provides a brief review of the RoboCrane platform, an explanation of control algorithms including the NIST GoMotion controller, and a discussion of crane task decomposition using the Four Dimensional/Real-time Control System approach.

1.1 The NIST RoboCrane

RoboCrane was first developed by the NIST Manufacturing Engineering Laboratory's (MEL) Intelligent Systems Division (ISD) in the late 1980s as part of a Defense Advanced Research Project Agency (DARPA) contract to stabilize crane loads (Albus et al., 1992). The basic RoboCrane is a parallel kinematic machine actuated through a cable support system. The suspended moveable platform is kinematically constrained by maintaining tension due to gravity in all six support cables. The support cables terminate in pairs at three vertices attached to an overhead support. This arrangement provides enhanced load stability over beyond traditional lift systems and improved control of the position and orientation (pose) of the load. The suspended moveable platform and the overhead support typically form two opposing equilateral triangles, and are often referred to as the "lower triangle" and "upper triangle," respectively.

The version of RoboCrane used in this research is the Tetrahedral Robotic Apparatus (TETRA). In the TETRA configuration, all winches, amplifiers, and motor controllers are located on the moveable platform as opposed to the support structure. The upper triangle only provides the three tie points for the cables, allowing the device to be retrofitted to existing overhead lift mechanisms. Although the TETRA configuration is presented in this chapter, the control algorithms and the Four Dimensional/Real-time Control System (4D/RCS), for 3D + time/Real-time Control System, task decomposition are adaptable to

many different crane configurations. The functional RoboCrane design can be extended and adapted for specialized applications including manufacturing, construction, hazardous waste remediation, aircraft paint stripping, and shipbuilding. Figure 1 depicts the RoboCrane TETRA configuration (a) and the representative work volume (b). Figure 2 shows additional retrofit configurations of the RoboCrane platform, and Figure 3 shows implementations for shipbuilding (Bostelman et al., 2002) and aircraft maintenance.



(a) (b)

Fig. 1. RoboCrane – TETRA configuration (a); Rendering of the RoboCrane environment. The shaded cylinder represents the nominal work volume (b).

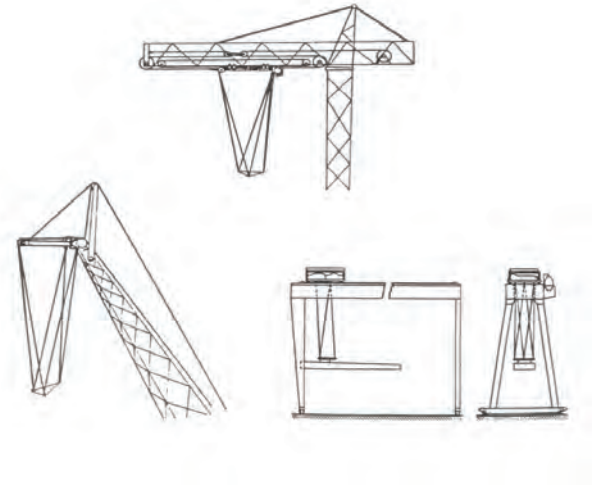


Fig. 2. Illustrations of RoboCrane in possible retrofitted configurations: Tower Crane (top), Boom Crane (lower left) and Gantry Bridge Crane (lower right).

1.2 Motivation for current research

Productivity gains in the U.S. construction sector have not kept pace with other industrial sectors such as manufacturing and transportation. These other industries have realized their productivity advances primarily through the integration of information, communication,

automation, and sensing technologies. The U.S. construction industry lags these other sectors in developing and adopting these critical, productivity-enhancing technologies. Leading industry groups, such as the Construction Industry Institute (CII), Construction Users Roundtable (CURT) and FIATECH, have identified the critical need for fully integrating and automating construction processes.
Robust field-automation on dynamic and cluttered construction sites will require advanced capabilities in construction equipment automation, site metrology, 3D imaging, construction object identification and tracking, data exchange, site status visualization, and design data integration for autonomous system behavior planning. The NIST Construction Metrology and Automation Group (CMAG) is conducting research to provide standards, methodologies, and performance metrics that will assist the development, integration, and evaluation of these technologies. Of particular interest are new technologies and capabilities for automated placement of construction components.

(a) (b)

Fig. 3. The NIST Flying Carpet – a platform for ship access in drydocks (a) and the NIST Aircraft Maintenance Project (AMP) – a platform for aircraft access in hangars (b).

2. RoboCrane kinematics

From (Albus et al., 1992), given an initial condition where the overhead support and the suspended platforms are represented by parallel, equilateral triangles with centers aligned along the vertical axis Z, (see Figure 4), the positions of the upper triangle with vertices A, B, and C and lower triangle with vertices D, E, and F are expressed as

$$\mathbf{A}=\begin{bmatrix} -b \\ -\frac{1}{3}b\sqrt{3} \\ -h \end{bmatrix} \quad \mathbf{B}=\begin{bmatrix} b \\ -\frac{1}{3}b\sqrt{3} \\ -h \end{bmatrix} \quad \mathbf{C}=\begin{bmatrix} 0 \\ \frac{2}{3}b\sqrt{3} \\ -h \end{bmatrix}$$
$$\mathbf{D}=\begin{bmatrix} 0 \\ -\frac{2}{3}a\sqrt{3} \\ 0 \end{bmatrix} \quad \mathbf{E}=\begin{bmatrix} a \\ \frac{1}{3}a\sqrt{3} \\ 0 \end{bmatrix} \quad \mathbf{F}=\begin{bmatrix} -a \\ \frac{1}{3}a\sqrt{3} \\ 0 \end{bmatrix} \tag{1}$$

With the positions of the vertices of triangles ABC and DEF as described in equations (1), when the lower platform is moved to a new position and orientation (D´E´F´) through a translation of

$$\mathbf{U} = \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix} \tag{2}$$

and a rotation of

$$R_{yxz}(\gamma,\theta,\phi) = R_z(\phi)\cdot R_x(\theta)\cdot R_y(\gamma) \tag{3}$$

the cable lengths can be expressed as

$$\mathbf{L_1} = \begin{bmatrix} -b + \frac{2}{3}aQ_{12}\sqrt{3} - u_x \\ -\frac{1}{3}b\sqrt{3} + \frac{2}{3}aQ_{22}\sqrt{3} - u_y \\ -h + \frac{2}{3}aQ_{32}\sqrt{3} - u_z \end{bmatrix} \qquad \mathbf{L_2} = \begin{bmatrix} -b + \frac{2}{3}aQ_{12}\sqrt{3} - u_x \\ -\frac{1}{3}b\sqrt{3} + \frac{2}{3}aQ_{22}\sqrt{3} - u_y \\ -h + \frac{2}{3}aQ_{32}\sqrt{3} - u_z \end{bmatrix}$$

$$\mathbf{L_3} = \begin{bmatrix} b - aQ_{11} - \frac{1}{3}aQ_{12}\sqrt{3} - u_x \\ -\frac{1}{3}b\sqrt{3} - aQ_{21} - \frac{1}{3}aQ_{22}\sqrt{3} - u_y \\ -h - aQ_{31} - \frac{1}{3}aQ_{32}\sqrt{3} - u_z \end{bmatrix} \quad \mathbf{L_4} = \begin{bmatrix} -aQ_{11} - \frac{1}{3}aQ_{12}\sqrt{3} - u_x \\ \frac{2}{3}b\sqrt{3} - aQ_{21} - \frac{1}{3}aQ_{22}\sqrt{3} - u_y \\ -h - aQ_{31} - \frac{1}{3}aQ_{32}\sqrt{3} - u_z \end{bmatrix} \tag{4}$$

$$\mathbf{L_5} = \begin{bmatrix} aQ_{11} - \frac{1}{3}aQ_{12}\sqrt{3} - u_x \\ \frac{2}{3}b\sqrt{3} + aQ_{21} - \frac{1}{3}aQ_{22}\sqrt{3} - u_y \\ -h + aQ_{31} - \frac{1}{3}aQ_{32}\sqrt{3} - u_z \end{bmatrix} \quad \mathbf{L_6} = \begin{bmatrix} -b + aQ_{11} - \frac{1}{3}aQ_{12}\sqrt{3} - u_x \\ -\frac{1}{3}b\sqrt{3} + aQ_{21} - \frac{1}{3}aQ_{22}\sqrt{3} - u_y \\ -h + aQ_{31} - \frac{1}{3}aQ_{32}\sqrt{3} - u_z \end{bmatrix}$$

where

$$\begin{array}{lll} \mathbf{L_1} = \mathbf{A} - \mathbf{D'} & \mathbf{L_2} = \mathbf{B} - \mathbf{D'} & \mathbf{L_3} = \mathbf{B} - \mathbf{E'} \\ \mathbf{L_4} = \mathbf{C} - \mathbf{E'} & \mathbf{L_5} = \mathbf{C} - \mathbf{F'} & \mathbf{L_6} = \mathbf{A} - \mathbf{F'} \end{array} \tag{5}$$

and $\mathbf{Q_{ij}}$ represents an element in the following rotation matrix:

$$\mathbf{Q} = \begin{bmatrix} \cos(\gamma)\cos(\phi) - \sin(\gamma)\sin(\theta)\sin(\phi) & -\cos(\theta)\sin(\phi) & \sin(\gamma)\cos(\phi) + \cos(\gamma)\sin(\theta)\sin(\phi) \\ \cos(\gamma)\sin(\phi) - \sin(\gamma)\sin(\theta)\cos(\phi) & \cos(\theta)\cos(\phi) & \sin(\gamma)\sin(\phi) - \cos(\gamma)\sin(\theta)\cos(\phi) \\ -\sin(\gamma)\cos(\theta) & \sin(\theta) & \cos(\gamma)\cos(\theta) \end{bmatrix} \tag{6}$$

Therefore, for any new desired pose of the moving platform described by equations (2) and (3), the required cable lengths to achieve that pose can be calculated by the inverse kinematic equations shown in equations (4).

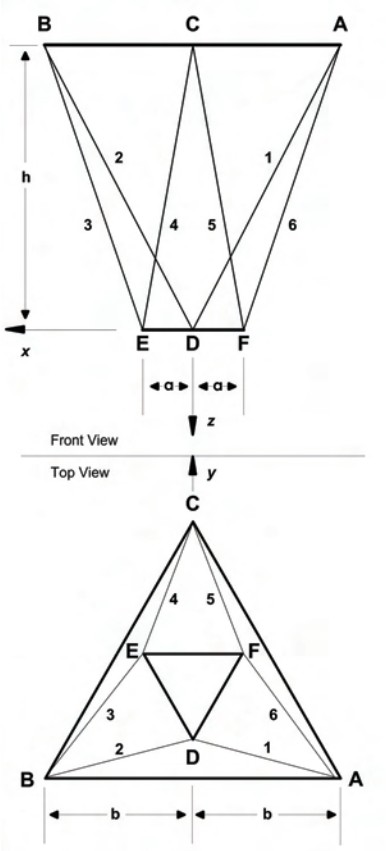


Fig. 4. Graphical representation of the RoboCrane cable support structure.

3. Measuring RoboCrane pose

The controller's estimate of the actual pose of RoboCrane differs from the actual pose due to several sources of error. Position feedback is provided through motor encoders that measure rotational position. Cable length is computed by multiplying the rotational position by the winch drum radius, with a suitable scale factor and offset.

However, the winch drum radius is not constant, but varies depending on the amount of cable that has already been wrapped around the drum, increasing its radius. It is possible to keep track of this and change the radius continually, by building a table that relates motor rotational position with effective radius.

Another source of error is that the cable length is affected by sag due to gravity. This sag depends on the pose of the platform and its load. Compensation can be achieved using an iterative process that begins with the nominal cable lengths, computes the platform pose using the forward kinematics equations, and determines the tensions on each of the cables using the transpose of the Jacobian matrix and the weight of the platform. The tensions can be used to generate the actual catenary curve of the cable, taking its nominal length as the

length of the hanging catenary curve. This process is repeated iteratively, with the nominal cable length as the fixed arc length of the catenary, and the chord between its endpoints as the continually revised length used by the forward kinematics.
Calibration errors in the mounting points of the ends of the cables further contribute to pose error. In practice these are not fixed points, but vary as the angles of the cables change the contact point to the pulleys or eye bolts that affix the ends. Even if these contact points were constant, their actual locations can be difficult to measure with precision, given their large displacement over a typical work volume.
Given these many sources of error, it is desirable to be able to measure the pose of the platform directly. There are many commercial systems for this purpose. An initial approach to external measurement implemented on RoboCrane uses a laser-based, large-scale, site measurement system (SMS).

3.1 The site measurement system (SMS)

A laser-based site measurement system (SMS) is used to track RoboCrane's pose and to measure object locations within the work volume. The SMS uses stationary, active-beacon laser transmitters and mobile receivers to provide millimeter-level position data at an update rate of approximately 20 Hz. This technology was chosen based upon a combination of factors including indoor/outdoor operation, accuracy, update-rate, and support for multiple receivers.
Each SMS transmitter emits two rotating, fanned laser beams and a timing pulse. Elevation is calculated from the time difference between fanned beam strikes. Azimuth is referenced from the timing pulse. The field of view of each transmitter is approximately 290° in azimuth and ± 30° in elevation/declination.
Similar to GPS, the SMS does not restrict the number of receivers. Line-of-sight to at least two transmitters must be maintained by each receiver in order to calculate that receiver's position. The optical receivers each track up to four transmitters and wirelessly transmit timing information to a base computer for position calculation.
For tracking RoboCrane's pose, four laser transmitters are positioned and calibrated on the work volume perimeter, and three SMS receivers are mounted on RoboCrane near the vertices of the lower triangle. The receiver locations are registered to the manipulator during an initial setup process in the local SMS coordinate frame. A transmitter and an optical receiver are shown in Figure 5. The SMS receivers mounted on RoboCrane are shown in Figure 6.

(a) (b)

Fig. 5. An SMS laser transmitter (a) and an SMS optical receiver (b).

The drawback of these systems is the added cost, and the need to maintain lines-of-sight between the platform and transmitters, potentially interfering with intended use. The benefits of accurate pose measurement are often significant enough to warrant their use.
In the first implementation of the SMS to track RoboCrane, position estimates were obtained at several stopping points during RoboCrane's trajectory, and these estimates were used as coarse correction factors for the encoder positions. Current work is focused on a dynamic tracking approach to eliminate the need for stopping points.

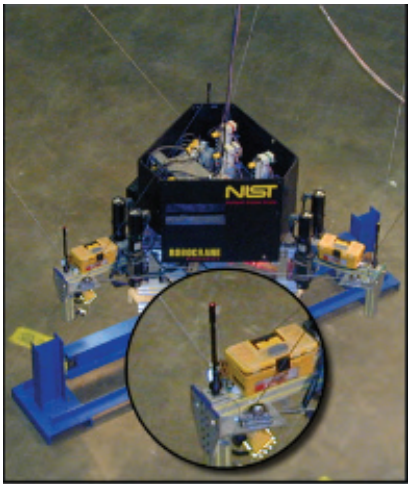


Fig. 6. The SMS on RoboCrane showing a close-up view of one of the three receivers.

3.2 Dynamic pose measurement

A commanded pose will generally result in a different actual pose due to various sources of system error such as those discussed previously. This relationship is depicted as

$$\mathbf{N} \rightarrow \mathbf{X} \rightarrow \mathbf{A} \tag{7}$$

or, in matrix form,

$$\mathbf{NX} = \mathbf{A} \tag{8}$$

where $\mathbf{N}$ is the commanded pose, $\mathbf{X}$ is the perturbation that includes all the sources of error, and $\mathbf{A}$ is the actual pose that results. The effects of $\mathbf{X}$ can be cancelled by commanding an adjusted pose, $\mathbf{N}^*$, where

$$\mathbf{N}^* = \mathbf{N}\mathbf{X}^{-1} \tag{9}$$

Using the adjusted pose allows us to achieve the original desired pose since

$$\mathbf{N}^*\mathbf{X} = \mathbf{N} \tag{10}$$

In general, most of the sources of error are unknown and variable, so computing $\mathbf{X}^{-1}$ apriori is not feasible. However, $\mathbf{X}^{-1}$ can be estimated by comparing a previously commanded

adjusted pose, $\mathbf{N}^*$, with the resulting actual pose, $\mathbf{A}^*$, as measured by the SMS. For time step, (i-1)

$$\mathbf{N}^*_{i-1}\mathbf{X}_{i-1} = \mathbf{A}^*_{i-1} \tag{11}$$

And the inverse of $\mathbf{X}_{i-1}$ can be calculated as

$$\left(\mathbf{X}_{i-1}\right)^{-1} = \left(\mathbf{A}^*_{i-1}\right)^{-1}\mathbf{N}^*_{i-1} \tag{12}$$

For the current time step, (i), the commanded adjusted pose can be calculated as

$$\mathbf{N}^*_i = \mathbf{N}_i\left(\mathbf{X}_{i-1}\right)^{-1} \tag{13}$$

where $\mathbf{N}_i$ is the desired pose for the current time step and $\mathbf{X}_{i-1}$ is the perturbation from the previous time step. Therefore,

$$\mathbf{N}^*_i\mathbf{X}_i \approx \mathbf{N}_i \tag{14}$$

If the platform is moving, then the cancellation is not perfect, since we are trying to cancel this time step's unknown perturbation transform with the inverse from the previous time step, which will be slightly different. If the platform is stationary, the two converge and the cancellation becomes perfect.

Platform motion has a more pronounced practical effect due to measurement latency for $\mathbf{A}$. When computing $\mathbf{X}^{-1}$, it is important that the $\mathbf{N}$ and $\mathbf{A}$ poses are synchronized. If the measured $\mathbf{A}$ pose lags the nominal $\mathbf{N}$ pose, then the compensation will have the effect of *leading* the motion. When speed slows, this leading will become an overshoot, and the platform will oscillate.

In the presence of measurement latency, one solution is to only compute the compensating transform $\mathbf{X}^{-1}$ when the platform is stationary. With this method, the platform is moved into an area of interest, held stationary for at least the latency period, and $\mathbf{X}^{-1}$ is computed. From that point, iteration is suppressed, and the compensating transform is constant. As the platform moves away from the compensation point, its accuracy diminishes.

If the latency is constant and can be measured, a solution is to keep a time history of nominal poses and their associated inverse transforms, and look back into this history by the amount of latency to associate a pair $\mathbf{N}$ and $\mathbf{X}^{-1}$ to the latent $\mathbf{A}$ measurement. If the measurements can be timestamped, then the same technique can be supplemented with timestamps to make the association. This technique can be used in the presence of variable measurement latency.

Controller latency also has an effect on the accuracy of the compensating transform. Figure 7 shows the magnitude of the translation portion of the compensating transform during tests with four different trajectory cycle times. In each test, the platform speed varied from 1 cm/s to 10 cm/s. These tests were done with a simulated measurement system that simulates actual position from the servo position run through the forward kinematics. In this case, the compensating transform should be small, and in fact it goes to zero as the

motion pauses between each speed setting. It is apparent from these figures that as the platform speed increases, the magnitude of the compensating transform increases, as is expected from servo lag. It is also apparent that as cycle time increases, so does the magnitude of the transform. This is due to the uncertainty between when nominal position is registered by the controller, and when it is read out some fraction of a period later.

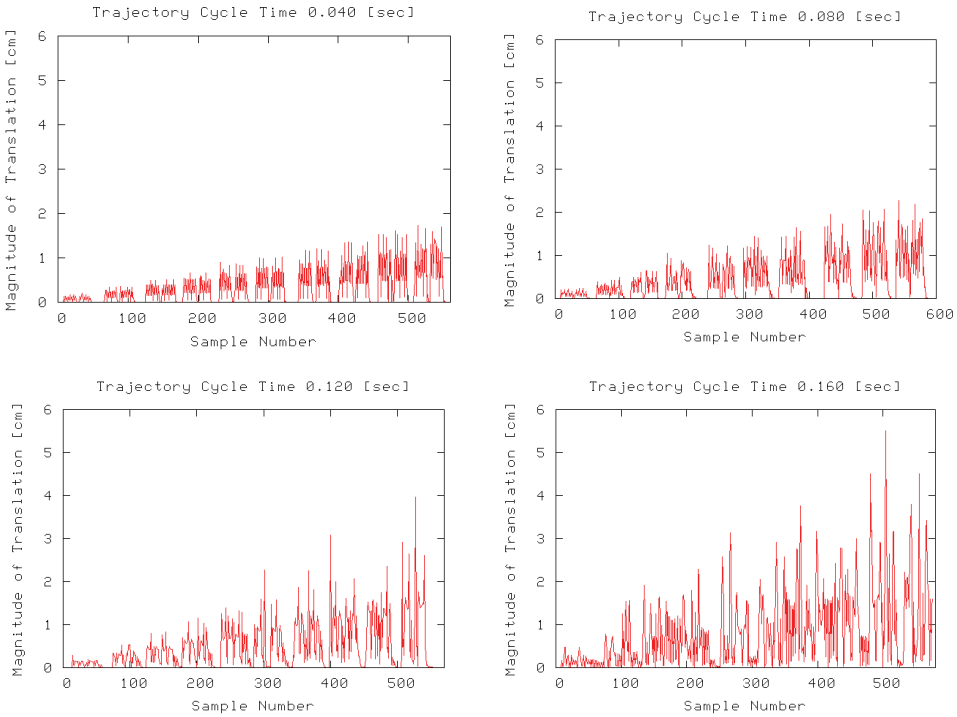


Fig. 7. Compensating transform magnitude (translation only) for four different trajectory cycle times. As the trajectory cycle time increases, the magnitude increases, and becomes more noisy as a result of the increased uncertainty in the latency between control output and compensation. (Note: Figures intended as qualitative examples of cycle time effects.)

Whenever a new $\mathbf{X}^{-1}$ transform is written to the controller, it has the potential to cause a jump in motion. To prevent this, transforms are "walked in" according to speed and acceleration limits. A large change in the transform will appear as a relatively quick but controlled move to the new, more accurate position. The effect of compensation is illustrated in Figure 8. The square path in the lower left of the figure is the uncompensated path, which is offset and slightly skewed from the ideal path due to kinematic miscalibration. Shortly after the second pass around the square path, compensation was turned on and its effects walked in over several seconds. This interval appears as the two line segments connecting the square paths. The square path in the upper right is the compensated path, whose adherence to the nominal edges at 0 cm and 10 cm is quite good.