# Proposed Project Description

#### Interface for quick volumetric error compensation of machine tools

J. Creamer

## Introduction

In order to begin implementation of table based volumetric compensation on the production floor, a machine must be able to be calibrated by maintenance personnel. Current practice uses several Matlab files which are relatively complex to take the raw measurement data and generate compensation tables. A user interface which simplifies this process is therefore necessary. Such a program would take raw comma separated measurement data and a machine configuration and output compensation tables.

## Process overview

The first step in the calibration of a 5-axis machine tool is to collect measurements throughout the joint space. A laser tracker is used to measure an active target mounted in the machine tool spindle at 200-500 positions throughout the joint space. These positions are measured using two different tool lengths, which allows both position and orientation errors of the tool to be modeled. Each tool length is measured on site using the laser tracker and a metrology bench. Any necessary link lengths are recorded from the controller. This data along with the machine configuration are used to build and error model. Model parameters are identified using a maximum likelihood estimator. The resulting parameters are used to populate compensation tables for the specific machine tool controller.

## Program overview

Inputs:

* Long tool measurement .csv file
* Short tool measurement .csv file
* Command file as .mpf from controller or Matlab variable
* Measured tool lengths and controller offsets and link lengths
* Machine tool configuration
* Controller profile

Outputs:

* Table file (.txt or .mpf)
* Performance indicators (Mean and Max residual error, chi squared, calculation time for full tables, same for controller tailored tables+ which tables were selected)
* Plot of table functions



Figure 2.1: Chart of program functions

### Measurement alignment

Often one or more measurements is skipped, repeated, or taken in the wrong location. The long and short tool measurements are each taken at the same set of commanded positions, but differ in tool length, so there is a constant vector difference between each long and corresponding short tool measurement. Any time this offset varies by more than 2-3 standard deviations, there is a problem. This section of the program should align measurements or remove them so that all remaining measurements have an offset within the expected range.

### Input machine configuration

The user selection of a machine configuration should be done via either a drop down menu or the selection of a picture/illustration (preferred). This needs to trigger the use of the correct forward kinematics function for other scripts. Detailed information about the machine tool kinematics is included in Appendix A. The configuration could be identified by specifying rotary or translational joint and a unit vector defining the direction.

### Calculate transformation

The first three long and short tool measurements, tool lengths, offset, and first three commanded positions are fed in to a Matlab script running a least-squares algorithm which fits a transformation between measurement coordinates and machine coordinates. The least-squares algorithm will be provided as a Matlab script of compiled library. It can be reference in [1].

### Align commands

Even though the long and short tool measurements are aligned with one another, they may not align with the commanded positions. The forward kinematics can be used to predict the measured position of the machine for a given command, and the actual measurement should be reasonably close to the predicted value. If the command and measurement do not match, the error between the actual and predicted measurement will be several orders of magnitude larger than expected. Typical errors for a 5-axis machine tool are on the order of 10-2 in, but errors for misaligned commands are 101 to 102 in.

### Fit a full error model

A “full” model of the actual kinematics is fit using the Implicit Loop Method, a maximum likelihood estimator described by Hollerbach et. al. [2]. The ILM requires formatted, aligned data, tool lengths and relevant link lengths, and the transformation from the measurement frame. This section will be provided as a Matlab script of compiled library. It will output a goodness of fit value, the calculation time, the mean and maximum residual error, and plots of the identified table functions compiled in to a performance report.

### Select relevant compensation tables

Based on the capability of the selected machine controller, the appropriate tables to populate will be selected using a genetic algorithm based on the example provided by Haupt and Haupt [3]. The model from the best solution will then be used to populate tables and generate a performance report. This will be provided as a Matlab script of compiled library.

## References

|  |  |
| --- | --- |
| [1] | K. Arun, T. Huang and S. Blostein, "Least- Squares Fitting of tTo 3-D Point Sets," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 9, no. 5, pp. 698-700, 1987. |
| [2] | C. Hollerbach, J. Wampler and T. Arai, "An Implicit Loop Method for Kinematic Calibration and Its Application to Closed-Chain Mechanisms," *IEEE Trans. Robotics and Automation,* vol. 11, no. 5, pp. 710- 724, 1995. |
| [3] | R. L. Haupt and S. E. Haupt, Practical Genetic Algorithms, Hoboken, NJ: John Wiley & Sons, 2004. |

## Appendix A

In order to characterize machine error from tooltip measurements, it is necessary to know what tooltip position is expected for a set of joint commands. This is done by transforming from some base frame of the machine, through a coordinate system at each of the joints, into a coordinate frame at the tooltip.

The nominal kinematics for an n-link machine tool described by the Zero Reference model is given by,



where  is the joint command vector, and, if the kth joint is rotary, **T***k*(*qk*) is



where **p***o* is a point that joint axis passes through and **R**k(*qk*) is

 .

with **u***k*=[*ux uy uz*]T a unit vector defining the direction of the joint axis with respect to the reference coordinate system, and c(*q*k) and s(*q*k) represent cos(*q*k) and sin(*q*k), respectively, and *v(qk)* is 1-cos(*qk*).

For a translational joint, **T***k*(*qk*) is



where **u***k* is the unit vector defining the kth joint axis with respect to the reference coordinate system.

Because of inaccuracies in machine tool fabrication and construction, the actual kinematics of the machine tool are never exactly equivalent to those of the nominal kinematic model. The joint perturbation model of the actual kinematics is given by,



where,  is a vector of joint perturbation functions that perturb the nominal joint variables. First-order joint perturbation functions are described by,



where, *fij(qi)* is a function mapping the joint command, *qi*, on axis *i* onto the appropriate perturbation on joint *j*.

The full kinematics for a general machine are



where **T**0 is the transformation from the measurement frame to the base frame and **T**t is a transformation from the center of the last joint to the tool tip, given as



where the vector **u**=[*ux uy uz*] is a unit vector describing the tool direction with respect to a reference frame and *TL* and *Toffset* are the tool length and gage line offset, respectively.

The transformation from the laser tracker home to machine home, **T0,** is calculated using a least squares method described by Arun et. al. [1].