Homework Assignment 2

Advanced robotics & Robotics Systems

Oleg Rodionov

February 29, 2020

Abstract

The homework task is devoted to calibration of cylindrical robot. The main goal is to improve accuracy of manipulator's tool by compensation technique. Errors related to offsets of joints and links can be efficiently compensated by modifying internal parameters of the robot.

Calibration process consists of 4 main steps:

1. Modeling
2. Measuring
3. Identification
4. Implementation

With given assumptions of robot's stiffness, it's required to implement identification part elastostatic calibration and after that realize compensation technique to reduce an error.

Contents

GitHub Link

My code you can on GitHub repository you can find here.

Robot description and Inverse Kinematic problem

A picture of cylindrical robot you can see on the Figure 1. It's comprised of 2 links, one revolute and two translational joints. First we need to solve forward and inverse kinematics problems to build a VJM model.

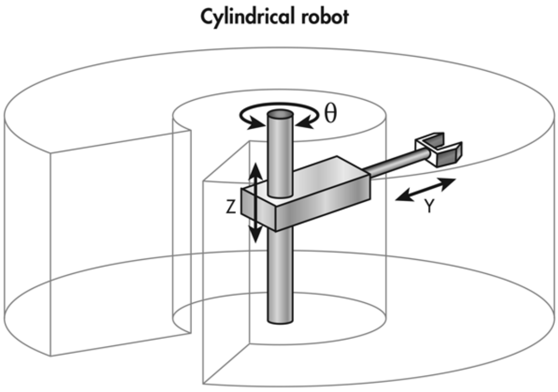

Figure 1: Cylindrical robot.

The sketches of front view and top view, you can see on figures 2 and 3 correspondingly.

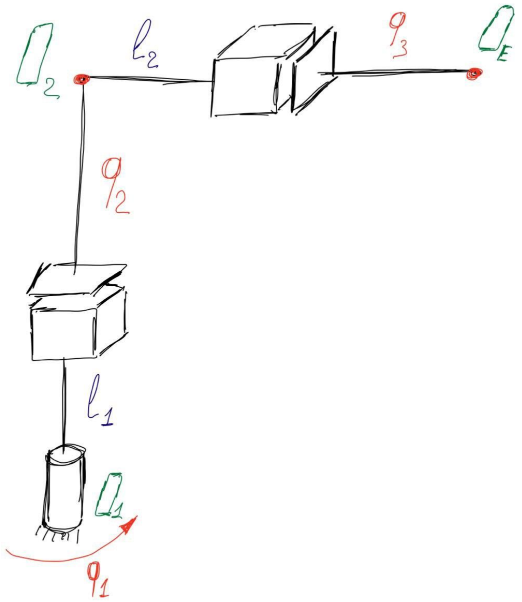

Figure 2: Front view scheme of robot .

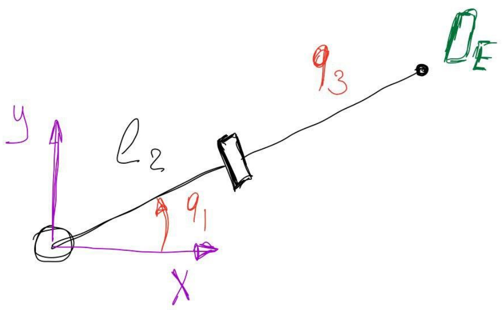

Figure 3: Top view scheme of robot.

By knowing the position of end-effector O_E (x_e, y_e, z_e) from the top view 3 we can easily define q_1 and q_3 as:

$$q_1 = atan2(y_e, x_e) \tag{1}$$

$$q_3 = \sqrt{x_e^2 + y_e^2} - l_2 \tag{2}$$

And q_2 remains constant in spite of changing q_1. So, we get the follow equation for q_2:

$$q_2 = z_e - l_1; \tag{3}$$

Reduced Virtual joint model

Now we want to describe *reduced* virtual joint model of this manipulator. In general, it may have more than 250 parameters and identification of this number of parameters is a very complicated task. So, we assume, that all links are rigid and the stiffness of the robot is lumped in joints. It's common approach for this task, because if we introduce flexible links, it will require the precise CAD model, but manufacturers of the robot usually don't provide CAD models). The VJM approach allows us to reduce number of parameters up to three, because we have only 3 stiffness virtual joints, with known initial stiffness coefficients. The model is presented the fig. 4.

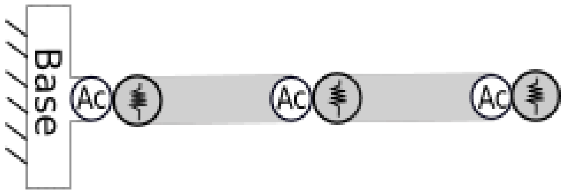

Figure 4: VJM model of the robot

And the expression, taking into account virtual springs is:

$$T = T_{base}R_z(q_1)R_z(\Theta_a^1)T_z(Link_1)Tz(q_2)Tz(\Theta_a^2)T_x(Link_2)T_x(q_3)T_x(\Theta_a^3) \tag{4}$$

where:

- q_i - the actuator displacement;
- $\theta(i)$ - virtual joints' coordinates for actuator and of the links.

The direction of springs for revolute joint - rotation with respect to axis z, for translational q_2 - displacement along z axis, and for q_3 - along local x axis.

Here is the table with stiffness coefficients and limits

Joint	**Stiffness,** $10^6 * \frac{N}{m}$	**Lower limit, rad**	**Upper limit, rad**
1	1	$-\pi$	π
2	2	0	0.5
3	0.5	0	0.5

Table 1: Joint stiffness coefficients and limits

Measurements

We don't have experimental data, so we need to obtain it from our model. Furthermore, we can add noise to simulate real conditions. Let's assume that it has zero mean and std $\sigma = 1 \bullet 10^-5$ metres. To obtain the data we can randomize end-effector positions, solve inverse kinematics problem for each point, after that calculate Jacobians and calculate deflection at each point according to this formula:

$$\Delta \boldsymbol{t} = \boldsymbol{J}_\theta \boldsymbol{K}_\theta^{-1} \boldsymbol{J}_\theta^T \boldsymbol{w} \tag{5}$$

where:

- $\Delta \boldsymbol{t}$ - the end-effector deflection;
- $\boldsymbol{J}_\theta$ - the Jacobian matrix with respect to $\boldsymbol{\theta}$
- $\boldsymbol{K}_\theta$ - the virtual stiffness diagonal matrix
- w - wrench general vector

We can take only linear part of the Jacobians and force part of wrenches, because we are interested in linear displacement of end-effector. After that we are ready to the identification.

Parameters identification

The main goal here is to find unknown stiffness parameters of the model. We can tune it according to the experimental data, obtained in previous section. We can rewrite equation 5 in the form:

$$\Delta \boldsymbol{t} = \sum_{i=1}^{n} (\boldsymbol{J}_{\theta,i} \boldsymbol{K}_{\theta,i}^{-1} \boldsymbol{J}_{\theta,i}^T) \boldsymbol{w} \tag{6}$$

- n is a number of experiments
- $\boldsymbol{K}_{\theta,i}$ - the link or joint compliances to be identified
- $\boldsymbol{J}_{\theta,i}$ - sub-Jacobians ($\boldsymbol{J}_\theta = [\boldsymbol{J}_{\theta,1}, \boldsymbol{J}_{\theta,2}...]$)

Also we know, that the values are corrupted by noise and can be represented as:

$$\Delta \boldsymbol{t} = \boldsymbol{A}_k(\boldsymbol{q}, \boldsymbol{w})\boldsymbol{k} + \boldsymbol{\epsilon} \tag{7}$$

where:

- $\boldsymbol{A}_k = [\boldsymbol{J}_{\theta,1}\boldsymbol{J}_{\theta,1}^T)\boldsymbol{w}, ..., \boldsymbol{J}_{\theta,n}\boldsymbol{J}_{\theta,n}^T)\boldsymbol{w}]$ is an observation matrix;
- $\boldsymbol{k} = (\boldsymbol{k}_{\theta,1}, \boldsymbol{k}_{\theta,2}, ..., \boldsymbol{k}_{\theta,n})^T$ is a compliance vector;
- $\boldsymbol{e}$ - measurement noise.

We are implementing elastostatic calibration in order to increase accuracy of stiffness model, i.e. update stiffness coefficients of the string. That's why we apply Least squares method on experimental data:

$$\sum_{i=1}^{m} ||\Delta\boldsymbol{t}_i - \boldsymbol{A}_k(\boldsymbol{q}_i, \boldsymbol{w}_i)\boldsymbol{k}|| \rightarrow \min \boldsymbol{k}. \tag{8}$$

where m is a number of experiments.(50 in my case). The solution has the following view:

$$\widehat{\boldsymbol{k}} = (\sum_{i=1}^{m} \boldsymbol{A}_i^T \boldsymbol{A}_i)^{-1} (\sum_{i=1}^{m} \boldsymbol{A}_i^T \Delta\boldsymbol{t}_i) \tag{9}$$

Important no notice, that we artificially added noise in deflections, so it is taken into account in identification technique.

Upgraded coefficients are presented in the table 2. It distinguishes from initial ones due to noise influence.

Joint	**Stiffness,** $10^5 * \frac{N}{m}$
1	7.8
2	20.9
3	5.2

Table 2: New Joint stiffness values

Compensation

Now we can use new stiffness matrix to compensate an error. Let's choose the desired position(a circle, for example) and values of Force(I've chosen values [500, -250, 1000]). We want the end-effector to "draw" this trajectory. But robot will make a mistake, even with "calibrated" stiffness matrix(the value of error is about 1 millimeter). The main idea of compensation technique: we know the mistake of the end effector due to flexibility and we can't get rid off the mistake. Let's assign to robot another desired trajectory(updated), with which robot will also have a mistake, but this will lead it to the initial desired trajectory. Because we precisely know the deflections, we can easily implement it.

The results of these technique are presented on figures correspondingly. The first picture we will have, when we don't add a noise artificially, and the second, when we do.

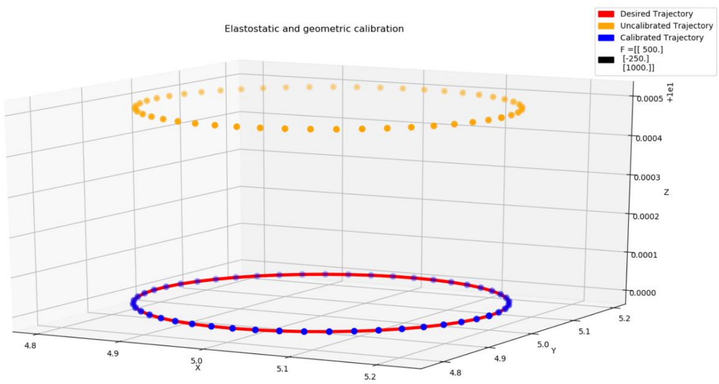

Figure 5: Compensated and uncompensated tool trajectories without random noise in the model.

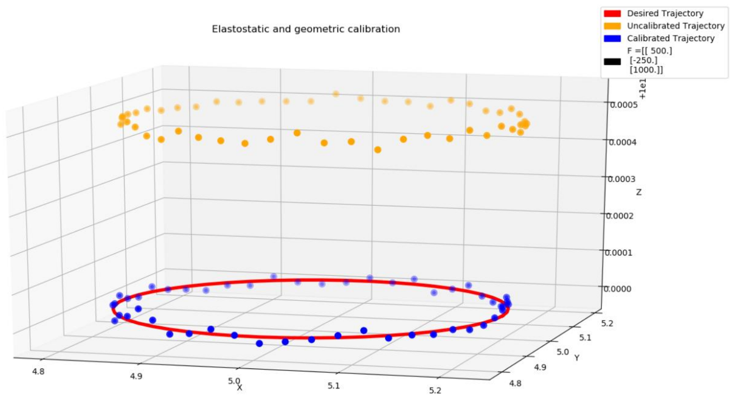

Figure 6: Compensated and uncompensated tool trajectories with random noise in the model.

Conclusions

We can conclude, that compensation technique works well for industrial manipulators and enable to achieve high accuracy. Elastostatic calibration allows us to increase precion of flexible model and it's parameters, but it depends on number of parameters, initial values of these parameters and presence of noise. In absence of noise it's not effective and we will obtain same coefficients as we had before calibration.

In practice we will always have noise and mean square error technique helps to reduce it. But as you see on figures 5 and 6 the noise affects on the results and eventually, we will have sufficiently accurate model.

Efficiency of calibrated of calibrated robots is higher than uncalibrated, because one of the most important advantages of industrial robots is accuracy of the end effector.