Kinematics Calibration

Kinematic Calibration and the Product of Exponentials Formula

Advances in Robot Kinematics and Computational Geometry, 1994

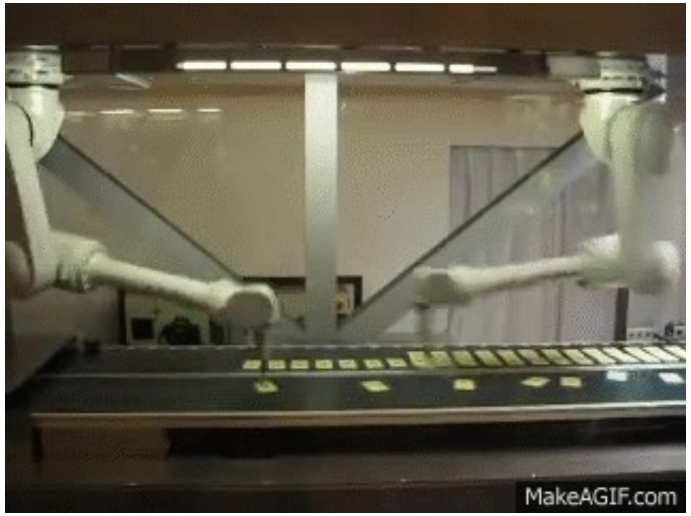
Prof. Driggs-Campbell

March 10, 2026

Fun Facts about Industrial Robots

- As of 2026, there are over 4.5 million industrial robots in operation
  - In 2025, nearly 550,000 new robots were installed in industrial operations
- Majority of industrial robots are used in automotive and electronics assembly
  - Metals (welding) and machinery industries are the fastest growing areas
- Industry robots typically operate for 10 years (80,000-10,000 hours)
  - In contrast, domestic robots (vacuums, mowers) typically operate for less than two hours per day and last about 5 years

Robot Lifespans and Calibration



- D-H parameters are extremely sensitive to small kinematic variations
    - Especially when join axes are nearly parallel
- PoE is a *continuous parametric model,* the exponential coordinates vary smoothly with variation sin the joint axes
    - The Jacobian also has a nice compact expression!

PoE-Based Calibration Steps

1. **Modeling:** Find screw axes as representation for parameterization
   - ω_i and v_i provide all information about links and joints
2. **Linearization:** Use the Jacobian to linearize the PoE formula
3. **Data Collection:** Cather ground truth tool pose across various postures
4. **Optimization:** Update screw axes to minimize different between measured data and computed FK pose

Effective calibration can improve position control errors by up to 90% and increase precision by orders of magnitude!

Linearization for Calibration

Linearizing the PoE Formula

- Using the fact that for curve in SE(3), $X(t) = e^{A(t)}$:

$$\dot{X}X^{-1} = \int_0^1 e^{A(t)s}\dot{A}(t)e^{-A(t)s}ds$$

- We may linearize the PoE formula as follows, were $A = \begin{bmatrix} [\omega_i] & v_i \\ 0 & 0 \end{bmatrix}$:

$$\begin{aligned} df \cdot f^{-1} &= A_1\,dx_1 + \mathrm{Ad}_{e^{A_1x_1}}(A_2)\,dx_2 + \ldots + \mathrm{Ad}_{e^{A_1x_1}\cdots e^{A_{n-1}x_{n-1}}}(A_n)\,dx_n \\ &+ x_1\int_0^1 \mathrm{Ad}_{e^{A_1x_1s}}(dA_1)\,ds + x_2\mathrm{Ad}_{e^{A_1x_1}}\left(\int_0^1 \mathrm{Ad}_{e^{A_2x_2s}}(dA_2)\,ds\right) \\ &+ \ldots + x_n\mathrm{Ad}_{e^{A_1x_1}\cdots e^{A_{n-1}x_{n-1}}}\left(\int_0^1 \mathrm{Ad}_{e^{A_nx_ns}}(dA_n)\,ds\right) \\ &+ \mathrm{Ad}_{e^{A_1x_1}\cdots e^{A_nx_n}}\left(\int_0^1 \mathrm{Ad}_{e^{\Gamma s}}(d\Gamma)\,ds\right) \end{aligned}$$

x_i is joint value
Γ is M parameters

Why linearize?

Least-Squares Solutions

Least Squares Example

Least Squares for Kinematic Calibration

- Errors and Jacobians in a single equation:
$$[y_i \cdots y_m]^\top = [\mathbf{A_1} \cdots \mathbf{A_m}]^\top \mathbf{p} = \mathbf{Y} = \boldsymbol{\mathcal{A}}\mathbf{p}$$
- Least squares solution:
$$\mathbf{p} = (\mathcal{A}^\top \mathcal{A})^{-1} \mathcal{A}^\top \mathbf{Y}$$
- Parameter update:
$$\mathbf{P}' = \mathbf{P} + \mathbf{p}$$
- Since this is a nonlinear problem, we iteratively solve until variations in **p** approach zero (parameters converge)
  - At each iteration, the new parameters for $[S_i]$ are used

A few things to note

- In the general formulation, we are free to optimize both ω_i, v_i simultaneously
  - This captures a helical joint! And useful since no joint is perfect and kinematic imperfections may result in a slightly helical motion
  - Can add constraints if you have perfect joints
- Any optimization or estimation algorithms can be used – not just least squares!
- This method only accounts for geometric errors
  - Other errors include backlash, gear transmission error, joint compliance may impact your model

Summary

- **Calibration** is super important for precise tasks!
  - Parameters vary from manufacturing defects or wear and tear over time
- The **PoE formulation is great for calibration** due to its simple and continuous representation
- Presented a method for **linearizing** and optimizing PoE parameters using **least squares** method
- Note that no method is perfect, but this geometric calibration can improve accuracy from the order of cm to sub-mm