Mobile Manipulation Calibration

Adam Heins

November 7, 2023

1 Introduction

The frames of interest are listed in Table 1.

Table 1: Coordinate frames.

Name	Subscript
World	w
Mobile base	b
Base of arm	a
End effector	e
Tool	t

The pose $\boldsymbol{T}_{wt}$ of an arbitrary tool attached to the end effector can be computed using the sequence of transforms

$$\boldsymbol{T}_{wt} = \boldsymbol{T}_{wb}(\boldsymbol{q}_b)\boldsymbol{T}_{ba}\boldsymbol{T}_{ae}(\boldsymbol{q}_a)\boldsymbol{T}_{et}, \tag{1}$$

where $\boldsymbol{T}_{wb}$ depends on the base configuration $\boldsymbol{q}_b = [x_b, y_b, \theta_b]^T$ and $\boldsymbol{T}_{ae}$ depends on the arm configuration $\boldsymbol{q}_a$.

2 Vicon Zero Pose

The `vicon_bridge` package allows one to provide a *zero pose* to specify the desired origin of a given model. Given the true robot frame $\{r\}$ and the Vicon model frame $\{v\}$, we have

$$\boldsymbol{T}_{wr} = \boldsymbol{T}_{wv}\boldsymbol{T}_{vr},$$

where $\boldsymbol{T}_{vr}$ is a constant transform we need to calibrate. The zero pose $\boldsymbol{T}^0_{wv}$ is such that $\boldsymbol{T}_{wr} = \boldsymbol{I}$, which implies $\boldsymbol{T}^0_{wv}\boldsymbol{T}_{vr} = \boldsymbol{I}$ and thus

$$\boldsymbol{T}^0_{wv} = \boldsymbol{T}^{-1}_{vr} = \begin{bmatrix} \boldsymbol{C}_{rv} & \boldsymbol{r}^{vr}_r \\ \boldsymbol{0}^T & 1 \end{bmatrix}$$

3 Base Calibration

3.1 Center of Rotation

We need to calibration the Vicon measured base poses $\hat{\boldsymbol{T}}_{wv}$ so that the origin of $\boldsymbol{T}_{wb}$ is correct (i.e., we want it at the center of rotation, such that there is no translational motion when the base is commanded to rotate). Starting at an arbitrary configuration $\hat{\boldsymbol{T}}_{wv,0}$, we will move the base to a sequence of desired yaw angles θ_i^d and obtain the corresponding measured configurations $\hat{\boldsymbol{T}}_{wv,i}$. We want to find $\boldsymbol{r}_v^{bv}$ such that $\hat{\boldsymbol{r}}_{wb,i} = \hat{\boldsymbol{r}}_{wb,0}$ is satisfied as closely as possible for each i, which yields the least-squares problem

$$\underset{\boldsymbol{r}_v^{bv}}{\operatorname{argmin}} \ \frac{1}{2}\sum_i \left\| \hat{\boldsymbol{C}}_{wv,i}\boldsymbol{r}_v^{bv} + \hat{\boldsymbol{r}}_{w,i}^{vw} - \hat{\boldsymbol{C}}_{wv,0}\hat{\boldsymbol{r}}_v^{bv} - \hat{\boldsymbol{r}}_{w,0}^{vw} \right\|^2. \tag{2}$$

This gives us $\boldsymbol{r}_v^{bv}$, but what we want for the Vicon zero pose is $\boldsymbol{r}_b^{vb} = -\boldsymbol{C}_{bv}\boldsymbol{r}_v^{bv}$, so we need to negate the value and be careful to rotate it from the Vicon frame to the base frame if $\boldsymbol{C}_{bv} \neq \boldsymbol{I}$.

We can write (2) in the form $\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2$ by taking $\boldsymbol{x} = \boldsymbol{r}_v^{bv}$ and

$$\boldsymbol{A} = \begin{bmatrix} \hat{\boldsymbol{C}}_{wv,0} - \hat{\boldsymbol{C}}_{wv,0} \\ \vdots \\ \hat{\boldsymbol{C}}_{wv,n} - \hat{\boldsymbol{C}}_{wv,0} \end{bmatrix}, \qquad \boldsymbol{b} = \begin{bmatrix} \hat{\boldsymbol{r}}_{w,0}^{vw} - \hat{\boldsymbol{r}}_{w,0}^{vw} \\ \vdots \\ \hat{\boldsymbol{r}}_{w,0}^{vw} - \hat{\boldsymbol{r}}_{w,n}^{vw} \end{bmatrix}.$$

3.2 Orientation (Yaw)

We also need to calibrate the Vicon measured base poses $\hat{\boldsymbol{T}}_{wv}$ so that the yaw angle of the model is aligned with the actual forward direction of $\boldsymbol{T}_{wb}$; that is, the direction of pure forward velocity. Starting at a particular pose $\hat{\boldsymbol{T}}_{wv,0}$, we command the Ridgeback to drive forward and periodically measure the pose $\hat{\boldsymbol{T}}_{wv,i}$. Ideally, the position of the base would be aligned with the x-axis of the initial frame. Let $\boldsymbol{p}_i = \boldsymbol{C}_{wv,0}^T(\boldsymbol{r}_{w,i}^{vw} - \boldsymbol{r}_{w,0}^{vw})$. The yaw error is $\Delta\theta_i = \operatorname{arctan2}(p_{y,i}, p_{x,i})$. We then solve the problem

$$\underset{\theta_{vb}}{\operatorname{argmin}} \ \frac{1}{2}\sum_i (\theta_{vb} - \Delta\theta_i)^2, \tag{3}$$

with corresponding Vicon zero orientation $\boldsymbol{C}_{bv} = \boldsymbol{C}_z(-\theta_{vb})$.

Important: remember to take this into account when computing the overall base zero pose including the position, as discussed above.

We can write (3) in the form $\|\boldsymbol{A}x - \boldsymbol{b}\|^2$ by taking $x = \theta_{bv}$, $\boldsymbol{A}$ a vector of ones, and $\boldsymbol{b}$ a vector of the angle errors $\Delta\theta_i$.

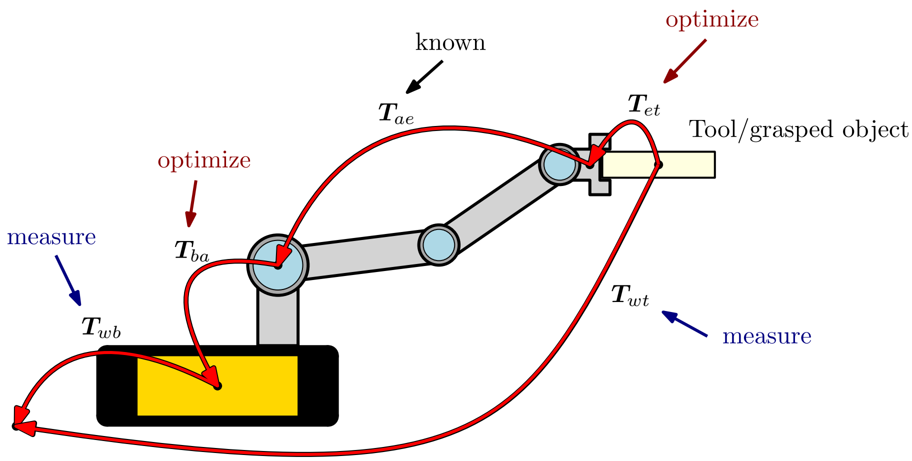

Figure 1: Frames on the robot being calibrated.

4 Arm–End Effector–Tool Calibration

Having calibrated the base pose $\boldsymbol{T}_{wb}$, we now wish to calibrate (static) the transforms between the base and arm $\boldsymbol{T}_{ba}$ and the EE and tool $\boldsymbol{T}_{et}$. We will assume that factory calibration for the arm transform $\boldsymbol{T}_{ae}(\boldsymbol{q}_a)$ is good enough.

To do so, we will collect a sequence of pairs $(\boldsymbol{q}_{a,i}, \hat{\boldsymbol{T}}_{wt,i})$ by moving the arm to the sequence of configurations $\boldsymbol{q}_{a,i}$. We will then solve the (nonlinear) least squares problem

$$\underset{\boldsymbol{T}_{ba},\boldsymbol{T}_{wt}}{\operatorname{argmin}} \ \frac{1}{2}\sum_i \| \boxminus (\boldsymbol{T}_{wt}(\boldsymbol{q}), \hat{\boldsymbol{T}}_{wt})\|^2. \tag{4}$$

over the manifold $SE(3)$, where $\boldsymbol{T}_{wt}$ is computed using (1) and

$$\boxminus (\boldsymbol{T}_1, \boldsymbol{T}_2) = \log(\boldsymbol{T}_1^{-1}\boldsymbol{T}_2)^{\vee} \tag{5}$$

is the error between the poses. We can include any prior information about the transforms as an initial guess for the solver.

5 Force Sensor Calibration

Here our goal is to find the orientation $\boldsymbol{C}_{bf}$ of the force sensor frame $\{f\}$ about the x- and y-axes of a frame $\{b\}$ with a gravity-aligned z-axis. To do so, we will zero the sensor and then attach a mass m to it and collect a set of force measurements $\{\boldsymbol{f}_{f,i}\}_{i=1}^{n}$. We then solve the nonlinear least-squares problem

$$\underset{\Delta\boldsymbol{C}\in SO(3)}{\operatorname{argmin}} \ \frac{1}{2}\sum_i \|\Delta\boldsymbol{C}\bar{\boldsymbol{C}}_{bf}\boldsymbol{f}_{f,i} - m\boldsymbol{g}\|^2$$

for the orientation offset $\Delta \boldsymbol{C}$ such that $\boldsymbol{C}_{bf} = \Delta \boldsymbol{C} \bar{\boldsymbol{C}}_{bf}$ with $\bar{\boldsymbol{C}}_{bf}$ a nominal guess for the orientation.

Calibrating the yaw angle offset would be more complicated, and at least require multiple orientations of $\boldsymbol{C}_{bf}$ to identify.