vicon2gt:
Derivations and Analysis

Patrick Geneva - pgeneva@udel.edu
Guoquan Huang - ghuang@udel.edu

Department of Mechanical Engineering
University of Delaware, Delaware, USA

RPNG

Robot Perception and Navigation Group (RPNG)
Tech Report - RPNG-2020-VICON2GT
Last Updated - November 15, 2020

Contents

1 Problem Formulation

The goal of this utility is to construct a groundtruth trajectory pose of the inertial measurement unit (IMU) sensor. To do so, we typically rely on an external motion capture system (e.g. an OptiTrack[1] or Vicon[2] system), which can provide accurate 6 degree-of-freedom (DoF) pose measurements of markers placed on the sensor rig. While it is straightforward to capture a 6 DoF pose once the motion capture system is calibrated, how we can leverage these poses to robustly recover the pose of the inertial state of the trajectory need carefully studying. In the following we will introduce our toolbox developed specifically for recovering a 15 DoF state containing the orientation, position, velocity and biases of the IMU sensor frame at every timestep over a dataset. This requires handling of the calibration of the motion capture system world frame in respect to the IMU global inertial frame, the motion capture marker frame, and the time offset between the two systems. We will first introduce the state which we are estimating and the frame of references and then dive into how each measurement model is formulated.

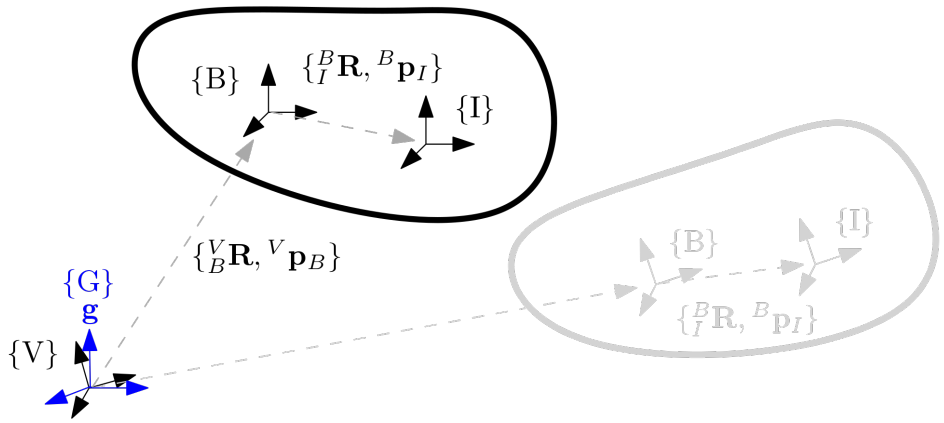

Figure 1: Sensor frames in the system. The motion capture frame $\{V\}$ which poses are capture in and is not gravity aligned along with the motion capture marker body frame $\{B\}$ and inertial IMU frame $\{I\}$ can be seen. Also seen is the gravity vector $\mathbf{g}$ which is is perfectly along the z-axis in the the global inertial frame $\{G\}$.

Shown in Figure 1, we have a series of frames that we need to define. The first is what we consider is the "global" frame in this estimation problem. As compared to estimating in the inertial global frame, we instead estimate in the motion capture system frame of reference $\{V\}$. This is primarily motivated to simplify the motion capture measurement model and allow explicit definition of the frame which gravity is represented in. This motion capture system global frame $\{V\}$ is what the collected poses are in and thus is defined during the motion capture system setup. The poses collected are of the markers which are attached to the sensor platform and thus the frame $\{B\}$ is arbitrarily defined to how the markers are placed and can change from dataset to dataset. This marker frame itself is not normally very useful to the end user.

The trajectory frame we are interested in is the IMU sensor frame $\{I\}$ which is defined by the accelerometer sensor axis sensor and is rigidly connected to the marker body frame. The issue that arises is that our global motion capture frame is *not* gravity aligned, thus we have to have special care when handling our gravity as it is defined in our global inertial frame. Specifically we estimate

[1] https://optitrack.com/
[2] https://www.vicon.com/

the following states:

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_{I_0}^\top & \cdots & \mathbf{x}_{I_N}^\top & \mathbf{x}_C^\top & {}^V_G\theta_x & {}^V_G\theta_y \end{bmatrix}^\top \tag{1}$$

$$\mathbf{x}_{I_i} = \begin{bmatrix} {}^{I_i}_V\bar{q}^\top & \mathbf{b}_{g,i}^\top & {}^V\mathbf{v}_{I_i}^\top & \mathbf{b}_{a,i}^\top & {}^V\mathbf{p}_{I_i}^\top \end{bmatrix}^\top \tag{2}$$

$$\mathbf{x}_C = \begin{bmatrix} {}^I_B\bar{q}^\top & {}^I\mathbf{p}_B^\top & {}^Vt_I \end{bmatrix}^\top \tag{3}$$

where we are estimating N inertial states at an arbitrary frequency, along with a calibration state $\mathbf{x}_C$ containing the spacial-temporal parameters between the motion capture and IMU sensors, and the rotation between the motion capture frame and global inertial frame. ${}^{I_k}_V\bar{q}$ is the unit quaternion parameterizing the rotation $\mathbf{R}({}^{I_k}_V\bar{q}) = {}^{I_k}_V\mathbf{R}$ from the global frame of reference $\{V\}$ to the IMU local frame $\{I_k\}$ at time t_k [1], $\mathbf{b}_\omega$ and $\mathbf{b}_a$ are the gyroscope and accelerometer biases, and ${}^V\mathbf{v}_{I_k}$ and ${}^V\mathbf{p}_{I_k}$ are the velocity and position of the IMU expressed in the global frame, respectively. It is important to note here that we are estimating the IMU states at the *true* IMU clock time, meaning that the states which occur at time t_k are in the IMU clock frame, ${}^It_k$, and can be related a time in motion capture clock by:

$${}^It = {}^Vt + {}^Vt_I \tag{4}$$

The inertial state $\mathbf{x}_{I_i}$ lies on the manifold defined by the product of the unit quaternions $\mathbb{H}$ with the vector space $\mathbb{R}^{12}$ (i.e. $\mathcal{M} = \mathbb{H} \times \mathbb{R}^{12}$) and has 15 DoF. For quaternions, we define the quaternion boxplus operation as:

$$\bar{q}_1 \boxplus \delta\boldsymbol{\theta} \triangleq \begin{bmatrix} \frac{\delta\boldsymbol{\theta}}{2} \\ 1 \end{bmatrix} \otimes \bar{q}_1 \simeq \bar{q}_2 \tag{5}$$

We additionally define our rotation from the gravity aligned inertial frame to the motion capture frame using the following roll-pitch-yaw rotation:

$${}^V_G\mathbf{R} = \mathbf{R}_y({}^V_G\theta_y)\mathbf{R}_x({}^V_G\theta_x) \tag{6}$$

$$= \begin{bmatrix} \cos{}^V_G\theta_y & 0 & \sin{}^V_G\theta_y \\ 0 & 1 & 0 \\ -\sin{}^V_G\theta_y & 0 & \cos{}^V_G\theta_y \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos{}^V_G\theta_x & -\sin{}^V_G\theta_x \\ 0 & \sin{}^V_G\theta_x & \cos{}^V_G\theta_x \end{bmatrix} \tag{7}$$

Note that here we are fixing the yaw to be zero since the gravity aligned frame has arbitrary yaw and thus this rotation is only 2 DoF. For vector variables, the "boxplus" and "boxminus" operations, which map elements to and from a given manifold [2], equate to simple addition and subtraction of their vectors.

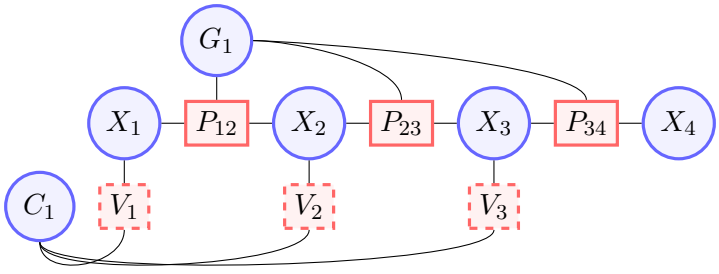

Figure 2: Example of a factor graph that our system created. States that will be estimated are denoted in circles and measurements are denoted in squares. Note that we differentiate interpolated factors with dashed outlines. For visualization we have grouped all calibration parameters into a single node C_1, and G_1 is the rotation from gravity aligned to motion capture frame.

An overview of the non-linear factor graph that we will solve is shown in Figure 2. In the following section we will explicitly explain which states the measurements are a function of. As a sneak-peak, the inertial IMU preintegration will be a function of the two bounding states and the inertial gravity frame transformation, while the motion capture 6 DoF pose measurements are a function of the spacial-temporal parameters between the two sensors.

2 Continuous Preintegration

2.1 Inertial Measurement Model

We model the linear acceleration and angular velocity inertial measurements as:

$$\boldsymbol{\omega}_m = \boldsymbol{\omega} + \mathbf{b}_w + \mathbf{n}_w \tag{8}$$

$$\mathbf{a}_m = \mathbf{a} + \mathbf{b}_a + \mathbf{n}_a + {}^{I}_{V}\mathbf{R}^{V}_{G}\mathbf{R}\,{}^{G}\mathbf{g} \tag{9}$$

where ${}^{G}\mathbf{g}$ is the gravity in the global *gravity aligned inertial frame*, $\{G\}$, which is $[0, 0, 9.81]$. The motion capture frame, $\{V\}$, is not gravity aligned, thus we need to take into account that there is a rotation into the inertial frame. This rotation is only a 2 DoF orientation where the yaw rotation around the gravity vector has been fixed to zero (e.g., a rotation about yaw will not change the direction of gravity in the $\{G\}$ frame). $\boldsymbol{\omega}$ is the angular velocity, $\mathbf{a}$ is the linear acceleration, and $\mathbf{n}_w,\ \mathbf{n}_a$ are the continuous measurement noises. The underlying IMU dynamics are [3]:

$$\dot{{}^{I}_{V}\bar{q}} = \frac{1}{2}\boldsymbol{\Omega}(\boldsymbol{\omega}_m - \mathbf{b}_w - \mathbf{n}_w)\,{}^{I}_{V}\bar{q} \tag{10}$$

$$\dot{\mathbf{b}}_w = \mathbf{n}_{wb} \tag{11}$$

$$\dot{{}^{V}\mathbf{v}_I} = {}^{V}_{I}\mathbf{R}\,(\mathbf{a}_m - \mathbf{b}_a - \mathbf{n}_a) - {}^{V}_{G}\mathbf{R}\,{}^{G}\mathbf{g} \tag{12}$$

$$\dot{\mathbf{b}}_a = \mathbf{n}_{ab} \tag{13}$$

$$\dot{{}^{V}\mathbf{p}_I} = {}^{V}\mathbf{v}_I \tag{14}$$

where $\mathbf{n}_{wb},\ \mathbf{n}_{ab}$ are the random walk noises and $\boldsymbol{\Omega}(\cdot)$ is:

$$\boldsymbol{\Omega}(\boldsymbol{\omega}) = \begin{bmatrix} -\lfloor\boldsymbol{\omega}\times\rfloor & \boldsymbol{\omega} \\ -\boldsymbol{\omega}^{\top} & 0 \end{bmatrix} \tag{15}$$

2.2 Continuous Preintegrated Measurements

Continuous preintegration [4] is the factorization of the integration of equation (10)-(14) between two state timesteps as follows (the IMU frame $\{I_k\}$ has been shorten to $\{k\}$):

$${}^{V}\mathbf{p}_{k+1} = {}^{V}\mathbf{p}_k + {}^{V}\mathbf{v}_k\Delta T - \frac{1}{2}{}^{V}_{G}\mathbf{R}\,{}^{G}\mathbf{g}\Delta T^2 + {}^{V}_{k}\mathbf{R}\,{}^{k}\boldsymbol{\alpha}_{k+1} \tag{16}$$

$${}^{V}\mathbf{v}_{k+1} = {}^{V}\mathbf{v}_k - {}^{V}_{G}\mathbf{R}\,{}^{G}\mathbf{g}\Delta T + {}^{V}_{k}\mathbf{R}\,{}^{k}\boldsymbol{\beta}_{k+1} \tag{17}$$

$${}^{k+1}_{V}\bar{q} = {}^{k+1}_{k}\bar{q} \otimes {}^{k}_{V}\bar{q} \tag{18}$$

where ΔT is the difference between the bounding pose timestamps (t_k, t_{k+1}) and ${}^{k}\boldsymbol{\alpha}_{k+1}, {}^{k}\boldsymbol{\beta}_{k+1}$ are defined by the following integrations of the IMU measurements:

$$ {}^{k}\boldsymbol{\alpha}_{k+1} = \int_{t_k}^{t_{k+1}} \int_{t_k}^{s} {}^{k}_{u}\mathbf{R} \left(\mathbf{a}_m - \mathbf{b}_a - \mathbf{n}_a \right) du ds \tag{19} $$

$$ {}^{k}\boldsymbol{\beta}_{k+1} = \int_{t_k}^{t_{k+1}} {}^{k}_{u}\mathbf{R} \left(\mathbf{a}_m - \mathbf{b}_a - \mathbf{n}_a \right) du \tag{20} $$

We note that the preintegrated measurements, ${}^{k}\boldsymbol{\alpha}_{k+1}$, ${}^{k}\boldsymbol{\beta}_{k+1}$, ${}^{k+1}_{k}\bar{q}$ are dependent on the *true* biases. This dependency is addressed through a first order Taylor series expansion about the current bias estimates $\bar{\mathbf{b}}_w$ and $\bar{\mathbf{b}}_a$ at time t_k:

$$ {}^{k}\boldsymbol{\alpha}_{k+1} \simeq {}^{k}\breve{\boldsymbol{\alpha}}_{k+1} + \frac{\partial \boldsymbol{\alpha}}{\partial \mathbf{b}_a}\Big|_{\bar{\mathbf{b}}_a} \Delta\mathbf{b}_a + \frac{\partial \boldsymbol{\alpha}}{\partial \mathbf{b}_w}\Big|_{\bar{\mathbf{b}}_w} \Delta\mathbf{b}_w \tag{21} $$

$$ {}^{k}\boldsymbol{\beta}_{k+1} \simeq {}^{k}\breve{\boldsymbol{\beta}}_{k+1} + \frac{\partial \boldsymbol{\beta}}{\partial \mathbf{b}_a}\Big|_{\bar{\mathbf{b}}_a} \Delta\mathbf{b}_a + \frac{\partial \boldsymbol{\beta}}{\partial \mathbf{b}_w}\Big|_{\bar{\mathbf{b}}_w} \Delta\mathbf{b}_w \tag{22} $$

$$ {}^{k+1}_{k}\bar{q} \simeq \bar{q}(\Delta\mathbf{b}_w)^{-1} \otimes {}^{k+1}_{k}\breve{q} \tag{23} $$

where ${}^{k}\breve{\boldsymbol{\alpha}}_{k+1}$, ${}^{k}\breve{\boldsymbol{\beta}}_{k+1}$, ${}^{k+1}_{k}\breve{q}$ are the preintegrated measurements evaluated at the current bias estimates. In particular, ${}^{k+1}_{k}\breve{q}$ can be found using the zeroth order quaternion integrator [1]. The quaternion which models multiplicative orientation corrections due to linearized bias change is:

$$ \bar{q}(\Delta\mathbf{b}_w) = \begin{bmatrix} \frac{\boldsymbol{\theta}}{||\boldsymbol{\theta}||} \sin\frac{||\boldsymbol{\theta}||}{2} \\ \cos\frac{||\boldsymbol{\theta}||}{2} \end{bmatrix} \tag{24} $$

$$ \boldsymbol{\theta} = \frac{\partial \bar{q}}{\partial \mathbf{b}_w}\Big|_{\bar{\mathbf{b}}_w} \left(\mathbf{b}_{w(k)} - \bar{\mathbf{b}}_w \right) \tag{25} $$

where $\Delta\mathbf{b}_w := \mathbf{b}_{w(k)} - \bar{\mathbf{b}}_w$ and $\Delta\mathbf{b}_a := \mathbf{b}_{a(k)} - \bar{\mathbf{b}}_a$ are the differences between the true biases and the current bias estimate used as the linearization point. The new preintegration measurements can now be computed *once* and changes in the bias estimates can be taken into account through the above Taylor series. The final measurement residual is as follows:

$$ r_I(\mathbf{x}) = \begin{bmatrix} 2\text{vec}\Big({}^{k+1}_{V}\bar{q} \otimes {}^{k}_{V}\bar{q}^{-1} \otimes {}^{k+1}_{k}\breve{q}^{-1} \otimes \bar{q}(\Delta\mathbf{b}_w) \Big) \\ \mathbf{b}_{w,k+1} - \mathbf{b}_{w,k} \\ \begin{pmatrix} {}^{k}_{V}\mathbf{R}\Big({}^{V}\mathbf{v}_{k+1} - {}^{V}\mathbf{v}_k + {}^{V}_{G}\mathbf{R}\,{}^{G}\mathbf{g}\Delta T \Big) \\ -{}^{k}\breve{\boldsymbol{\beta}}_{k+1} - \frac{\partial \boldsymbol{\beta}}{\partial \mathbf{b}_a}\Big|_{\bar{\mathbf{b}}_a} \Delta\mathbf{b}_a - \frac{\partial \boldsymbol{\beta}}{\partial \mathbf{b}_w}\Big|_{\bar{\mathbf{b}}_w} \Delta\mathbf{b}_w \end{pmatrix} \\ \mathbf{b}_{a,k+1} - \mathbf{b}_{a,k} \\ \begin{pmatrix} {}^{k}_{V}\mathbf{R}\Big({}^{V}\mathbf{p}_{k+1} - {}^{V}\mathbf{p}_k - {}^{V}\mathbf{v}_k\Delta T + \frac{1}{2}{}^{V}_{G}\mathbf{R}\,{}^{G}\mathbf{g}\Delta T^2 \Big) \\ -{}^{k}\breve{\boldsymbol{\alpha}}_{k+1} - \frac{\partial \boldsymbol{\alpha}}{\partial \mathbf{b}_a}\Big|_{\bar{\mathbf{b}}_a} \Delta\mathbf{b}_a - \frac{\partial \boldsymbol{\alpha}}{\partial \mathbf{b}_w}\Big|_{\bar{\mathbf{b}}_w} \Delta\mathbf{b}_w \end{pmatrix} \end{bmatrix} $$

where $\text{vec}(\cdot)$ returns the vector portion of the quaternion (i.e., the top three elements) and the bias errors are the difference between biases in the bounding states.

We use combined continuous preintegration factors that included both the inertial and bias errors together and relate to the full 15 degree-of-freedom state (see Equation 1). This combined continuous preintegration factor better models the measurement error state dynamics due to bias drift over the integration interval.

2.3 Continuous Preintegrated Jacobians

The analytical Jacobians needed for graph optimization, bias Jacobians, and *closed-form* preintegrated measurements are included in the preintegration technical report [5] where the above Jacobians correspond to the following equations:

$$\frac{\partial \boldsymbol{\alpha}}{\partial \mathbf{b}_a}\Big|_{\bar{\mathbf{b}}_a} = \text{Equation (49)} \tag{26}$$

$$\frac{\partial \boldsymbol{\alpha}}{\partial \mathbf{b}_w}\Big|_{\bar{\mathbf{b}}_w} = \text{Equations (53)-(58),(84)} \tag{27}$$

$$\frac{\partial \boldsymbol{\beta}}{\partial \mathbf{b}_a}\Big|_{\bar{\mathbf{b}}_a} = \text{Equation (49)} \tag{28}$$

$$\frac{\partial \boldsymbol{\beta}}{\partial \mathbf{b}_w}\Big|_{\bar{\mathbf{b}}_w} = \text{Equations (59)-(61),(84)} \tag{29}$$

$$\frac{\partial \bar{q}}{\partial \mathbf{b}_w}\Big|_{\bar{\mathbf{b}}_w} = \text{Equation (81)} \tag{30}$$

We additionally have the following two Jacobians in respect to the two angles which rotate the gravity vector into the motion capture frame. This can be found by directly taking the derivative in respect to the rotation matrix:

$$\frac{\partial \boldsymbol{\alpha}}{\partial [{}^V_G\theta_x \ {}^V_G\theta_y]} = \frac{1}{2} {}^k_V\mathbf{R} \Delta T^2 \mathbf{H}_z \tag{31}$$

$$\frac{\partial \boldsymbol{\beta}}{\partial [{}^V_G\theta_x \ {}^V_G\theta_y]} = {}^k_V\mathbf{R} \Delta T \mathbf{H}_z \tag{32}$$

where we define the derivative in respect to ${}^V_G\mathbf{R}{}^G\mathbf{g}$ as:

$$\mathbf{H}_z = 9.81 \begin{bmatrix} -\sin {}^V_G\theta_y \sin {}^V_G\theta_x & \cos {}^V_G\theta_y \cos {}^V_G\theta_x \\ -\cos {}^V_G\theta_x & 0 \\ -\cos {}^V_G\theta_y \sin {}^V_G\theta_x & -\sin {}^V_G\theta_y \cos {}^V_G\theta_x \end{bmatrix} \tag{33}$$

2.4 Continuous Preintegrated Covariance

To find the covariance of the above residual, we can look at the continuous IMU error state dynamics. Consider the time $t_\tau \in [t_k, t_{k+1}]$. Defining ${}^k_\tau\breve{\mathbf{R}}$ as the rotation from the IMU frame at t_τ to the beginning IMU time t_k, $\hat{\mathbf{a}}$ as the corrected acceleration ($\hat{\mathbf{a}} = \mathbf{a}_m - \bar{\mathbf{b}}_a$), and $\hat{\boldsymbol{\omega}}$ as the corrected angular velocity ($\hat{\boldsymbol{\omega}} = \boldsymbol{\omega}_m - \bar{\mathbf{b}}_\omega$), the linearized *measurement* error state system can be defined as

the following:

$$\begin{bmatrix} {}^{\tau}_{k}\dot{\delta\boldsymbol{\theta}} \\ \dot{\delta\mathbf{b}}_{\omega} \\ {}^{k}\dot{\delta\boldsymbol{\beta}}_{\tau} \\ \dot{\delta\mathbf{b}}_{a} \\ {}^{k}\dot{\delta\boldsymbol{\alpha}}_{\tau} \end{bmatrix} = \begin{bmatrix} -\lfloor\hat{\boldsymbol{\omega}}\times\rfloor & -\mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ -{}^{k}_{\tau}\breve{\mathbf{R}}\lfloor\hat{\mathbf{a}}\times\rfloor & \mathbf{0}_3 & \mathbf{0}_3 & -{}^{k}_{\tau}\breve{\mathbf{R}} & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix} \begin{bmatrix} {}^{\tau}_{k}\delta\boldsymbol{\theta} \\ \delta\mathbf{b}_{\omega} \\ {}^{k}\delta\boldsymbol{\beta}_{\tau} \\ \delta\mathbf{b}_{a} \\ {}^{k}\delta\boldsymbol{\alpha}_{\tau} \end{bmatrix} + \begin{bmatrix} -\mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & -{}^{k}_{\tau}\breve{\mathbf{R}} & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix} \begin{bmatrix} \mathbf{n}_{\omega} \\ \mathbf{n}_{b\omega} \\ \mathbf{n}_{a} \\ \mathbf{n}_{a\omega} \end{bmatrix}$$

$$\Rightarrow \quad \dot{\mathbf{r}} = \mathbf{F}\mathbf{r} + \mathbf{G}\mathbf{n} \tag{34}$$

It can be noted that the above is equivalent to the standard VINS error state propagation equations in the local frame of reference [6]. Based on the above equations, we can define the state transition matrix $\boldsymbol{\Phi}(t_{\tau+1}, t_{\tau})$ which describes how the error transitions across the measurement interval $[t_{\tau}, t_{\tau+1}] \subset [t_k, t_{k+1}]$. Starting with covariance $\mathbf{P}_k = \mathbf{0}_{15\times15}$ we perform the following propagation for all IMU measurements in the preintegration interval $[t_k, t_{k+1}]$.

$$\mathbf{P}_{\tau+1} = \boldsymbol{\Phi}(t_{\tau+1}, t_{\tau})\mathbf{P}_{\tau}\boldsymbol{\Phi}(t_{\tau+1}, t_{\tau})^{\top} + \mathbf{Q}_{p_{\tau}}$$

$$\mathbf{Q}_{p_{\tau}} = \int_{t_{\tau}}^{t_{\tau+1}} \boldsymbol{\Phi}(t_{\tau+1}, u)\mathbf{G}(u)\mathbf{Q}_c\mathbf{G}(u)^{\top}\boldsymbol{\Phi}(t_{\tau+1}, u)^{\top}du$$

where $\mathbf{Q}_c$ is the continuous time noise covariance matrix. The final covariance of the preintegrated measurement is the ending cumulative covariance $\mathbf{P}_{k+1}$.

3 Motion Capture 6 DoF Pose Factor

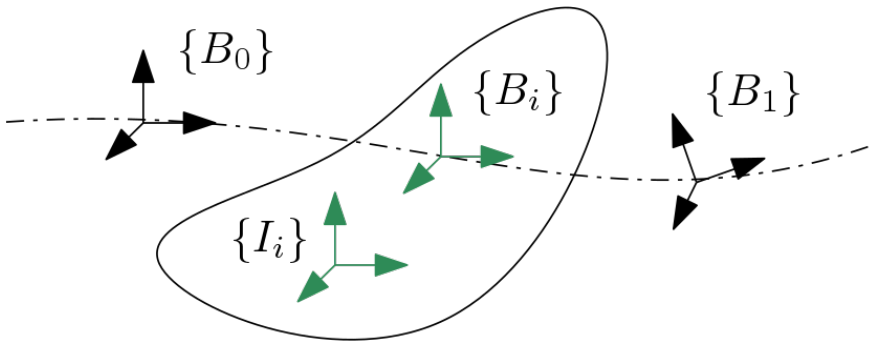

Figure 3: Example interpolation problem where two bounding motion capture poses $\{B_0\}$ and $\{B_1\}$. The motion capture pose is first interpolated to the pose time creating frame $\{B_i\}$, then the rigid extrinsic transformation can transform it into the IMU sensor frame $\{I_i\}$.

The core methodology of this measurement model is based on [7] which has been expanded to support time offset calibration. As shown in Figure 3, we construct a measurement by calculating an artificial pose $\{B_i\}$ which should occur at the state time. This can then be related to our state through the extrinsic transformation between the motion capture marker frame and the IMU sensor frame. This means that this measurement function is only a function of the state it is interpolated to and the calibration.

Another way to approach this problem (which we do not use), is to instead interpolate two bounding states to each motion capture pose measurement, $\{B_0\}$ and $\{B_1\}$, and introduce a measurement for every motion capture pose measurement. This method likely has more robustness to noisy motion capture poses but likely more sensitivity to the interpolation model if state frequencies are low.

3.1 Measurement Model

We are estimating the state $\mathbf{x}_{I_i}$ which has occurred at time ${}^{I}t_i$ in the IMU clock frame. We can first calculate a synthetic measurement which is of the motion capture marker frame at this time instance through the following:

$$\{{}^{B_i}_{V}\mathbf{R}, {}^{V}\mathbf{p}_{B_i}\} = g\Big(\{{}^{B_0}_{V}\mathbf{R}, {}^{V}\mathbf{p}_{B_0}\}, \{{}^{B_1}_{V}\mathbf{R}, {}^{V}\mathbf{p}_{B_1}\}, {}^{V}t_I\Big) \tag{35}$$

This function $g(\cdot)$ is *not* the measurement function, but instead is how we calculate our noisy measurement. The measurement is a function of the bounding motion capture poses and the time offset between the motion capture system and IMU. Specifically it is defined as:

$${}^{B_i}_{V}\mathbf{R} = \text{Exp}\left(\lambda\ \text{Log}({}^{B_1}_{V}\mathbf{R}\,{}^{B_0}_{V}\mathbf{R}^{\top})\right){}^{B_0}_{V}\mathbf{R} \tag{36}$$

$${}^{V}\mathbf{p}_{B_i} = (1-\lambda){}^{V}\mathbf{p}_{B_0} + \lambda\,{}^{V}\mathbf{p}_{B_1} \tag{37}$$

$$\lambda = \frac{({}^{V}t_i - {}^{V}t_{B_0})}{({}^{V}t_{B_1} - {}^{V}t_{B_0})} = \frac{({}^{I}t_i - {}^{V}t_I - {}^{V}t_{B_0})}{({}^{V}t_{B_1} - {}^{V}t_{B_0})} \tag{38}$$

where we have the bounding poses $\{B_0\}$ and $\{B_1\}$ which were collected at time ${}^{V}t_{B_0}$ and ${}^{V}t_{B_1}$ in the motion capture clock frame. We wish to interpolate to the state time ${}^{I}t_i$, thus we calculate the time in the motion capture clock as ${}^{V}t_i = {}^{I}t_i - {}^{V}t_I$. As in [7] the measurement covariance is propagate through the following covariance propagation:

$$\boldsymbol{P}_i = \boldsymbol{H}_u \boldsymbol{P}_{0,1} \boldsymbol{H}_u{}^{\top} \tag{39}$$

Having calculated now the measurement $\{{}^{B_i}_{V}\breve{\mathbf{R}}, {}^{V}\breve{\mathbf{p}}_{B_i}\}$ and its measurement noise $\boldsymbol{P}_i$, we can formulate the measurement function which relates this measurement to our state estimates through our spacial calibration parameters:

$$\{{}^{B_i}_{V}\mathbf{R}, {}^{V}\mathbf{p}_{B_i}\} = h\Big(\{{}^{I_i}_{V}\mathbf{R}, {}^{V}\mathbf{p}_{I_i}\}, \{{}^{I}_{B}\mathbf{R}, {}^{I}\mathbf{p}_{B}\}\Big) + \mathbf{n}_{pose} \tag{40}$$

$${}^{B_i}_{V}\mathbf{R} = {}^{I}_{B}\mathbf{R}^{\top}\,{}^{I_i}_{V}\mathbf{R} + \mathbf{n}_{ori} \tag{41}$$

$${}^{V}\mathbf{p}_{B_i} = {}^{V}\mathbf{p}_{I_i} + {}^{I_i}_{V}\mathbf{R}^{\top}\,{}^{I}\mathbf{p}_{B} + \mathbf{n}_{pos} \tag{42}$$

We thus have the following residual:

$$r_V(\mathbf{x}) = \begin{bmatrix} 2\text{vec}\Big({}^{I}_{B}\bar{q} \otimes {}^{B_i}_{V}\bar{q} \otimes {}^{B_i}_{V}\breve{\bar{q}}^{-1}\Big) \\ {}^{V}\mathbf{p}_{I_i} + {}^{I_i}_{V}\mathbf{R}^{\top}\,{}^{I}\mathbf{p}_{B} - {}^{V}\breve{\mathbf{p}}_{B_i} \end{bmatrix} + \mathbf{n}_{pose} \tag{43}$$

3.2 Measurement Jacobians

We can define the following linearized residual system as:

$$\tilde{r}_V(\mathbf{x}) = \frac{\partial h}{\partial \mathbf{x}_{I_i}}\tilde{\mathbf{x}}_{I_i} + \frac{\partial h}{\partial \mathbf{x}_{C}}\tilde{\mathbf{x}}_{C} - \frac{\partial g}{\partial {}^{V}t_I}{}^{V}\tilde{t}_I + \mathbf{n}_{pose} \tag{44}$$

Thus, the measurement Jacobians are as follows:

$$\frac{\partial h}{\partial \mathbf{x}_{I_i}} = \begin{bmatrix} {}^{I}_{B}\mathbf{R}^{\top} & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ -{}^{I_i}_{V}\mathbf{R}^{\top}\lfloor {}^{I}\mathbf{p}_B \times \rfloor & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix} \tag{45}$$

$$\frac{\partial h}{\partial \mathbf{x}_{C}} = \begin{bmatrix} -{}^{I}_{B}\mathbf{R}^{\top} & \mathbf{0}_3 & 0 \\ \mathbf{0}_3 & {}^{I_i}_{V}\mathbf{R}^{\top} & 0 \end{bmatrix} \tag{46}$$

$$\frac{\partial g}{\partial {}^{V}t_I} = \frac{\partial g}{\partial \lambda}\frac{\partial \lambda}{\partial {}^{V}t_I} \tag{47}$$

$$= \begin{bmatrix} -{}^{B_i}_{B_0}\mathbf{R}\ \mathrm{J}_r\left(\lambda\ {}^{B_1}_{B_0}\boldsymbol{\phi}\right){}^{B_1}_{B_0}\boldsymbol{\phi} \\ {}^{V}\mathbf{p}_{B_1} - {}^{V}\mathbf{p}_{B_0} \end{bmatrix} \frac{-1}{({}^{V}t_{B_1} - {}^{V}t_{B_0})} \tag{48}$$

where we define the following:

$${}^{B_i}_{B_0}\mathbf{R} = \mathrm{Exp}\left(\lambda\ \mathrm{Log}({}^{B_1}_{V}\mathbf{R}{}^{B_0}_{V}\mathbf{R}^{\top})\right) \tag{49}$$

$${}^{B_1}_{B_0}\boldsymbol{\phi} = \mathrm{Log}({}^{B_i}_{B_0}\mathbf{R}) \tag{50}$$

$$\mathrm{J}_r(\phi) = \mathbf{I} - \frac{1 - cos(\parallel \phi \parallel)}{\parallel \phi \parallel^2}\lfloor \phi \times \rfloor + \frac{\parallel \phi \parallel - sin(\parallel \phi \parallel)}{\parallel \phi \parallel^3}\lfloor \phi \times \rfloor^2 \tag{51}$$

4 Simulation Results

We now wish to look at how sensitive the system is to motion capture pose noise. This is helpful in practice since we wish to know under what realistic noise scenarios we will be able to accurately recover the trajectory of the system. We generate the simulated trajectory using a modified version of the OpenVINS simulator [8]. We choose the TUM-VI corridor1 trajectory as our simulated pose trajectory which was generated with a visual-inertial system. A 200 Hz IMU, 20 Hz camera, and 100 Hz 6 DoF motion capture system was simulated. For each simulation, the system started with identity spacial transforms and a time offset value of zero. The simulator picks a random motion capture to IMU to marker body orientation $\sigma = 0.1$ and position $\sigma = 0.2$, and time offset $\sigma = 0.05$ (we used the same distributions for all 3 vector dimensions). We additionally select a random ${}^{V}_{G}\mathbf{R}$ with $\sigma = 0.1\pi \approx 10^{\circ}$ and simulate the system such that the B-spline trajectory is in the fixed inertial frame (thus all groundtruths from multiple different motion capture frame transforms are the same). As per standard practice, the inertial measurement readings were corrupted using the random walk biases and corresponding white noises, while the motion capture poses were corrupted using an additive white noise.

We observed that there was typically a small position offset between the groundtruth and optimized trajectory caused by errors in the marker to IMU transformation. Thus as compared to not performing alignment, we perform position and yaw alignment between the simulated groundtruth and optimized trajectories.

Table 1: Average absolute trajectory error (degrees / meters) over 10 Monte-Carlo runs. Each column is the motion capture orientation noises (deg), while each row is a different position motion capture noise (m).

	0.057	**0.286**	**0.573**	**2.864**	**5.730**
0.001	0.020 / 0.001	0.106 / 0.007	0.300 / 0.021	0.464 / 0.045	0.441 / 0.045
0.005	0.037 / 0.002	0.157 / 0.009	0.318 / 0.020	0.466 / 0.045	0.441 / 0.045
0.010	0.115 / 0.006	0.211 / 0.012	0.348 / 0.024	0.462 / 0.045	0.440 / 0.045
0.050	0.051 / 0.012	0.363 / 0.024	0.409 / 0.029	0.409 / 0.039	0.421 / 0.043
0.100	0.050 / 0.022	0.371 / 0.029	0.424 / 0.032	0.362 / 0.035	0.391 / 0.040

Shown in Table 1, we can see that as both orientation and position error increases, the orientation error plateaus at around half a degree even under 6 degree and 10cm pose errors. Position errors increase as motion capture measurements become noisier, but impressively never go above 5cm event with 10cm noise levels. This confirms that the system is able to reconstruct the trajectory in high noise level cases where the noise of the motion capture system and IMU sensor are known. Depending on the level of accuracy required, being conservative in the estimates might prove useful to ensure proper recovery.

5 Realworld Results

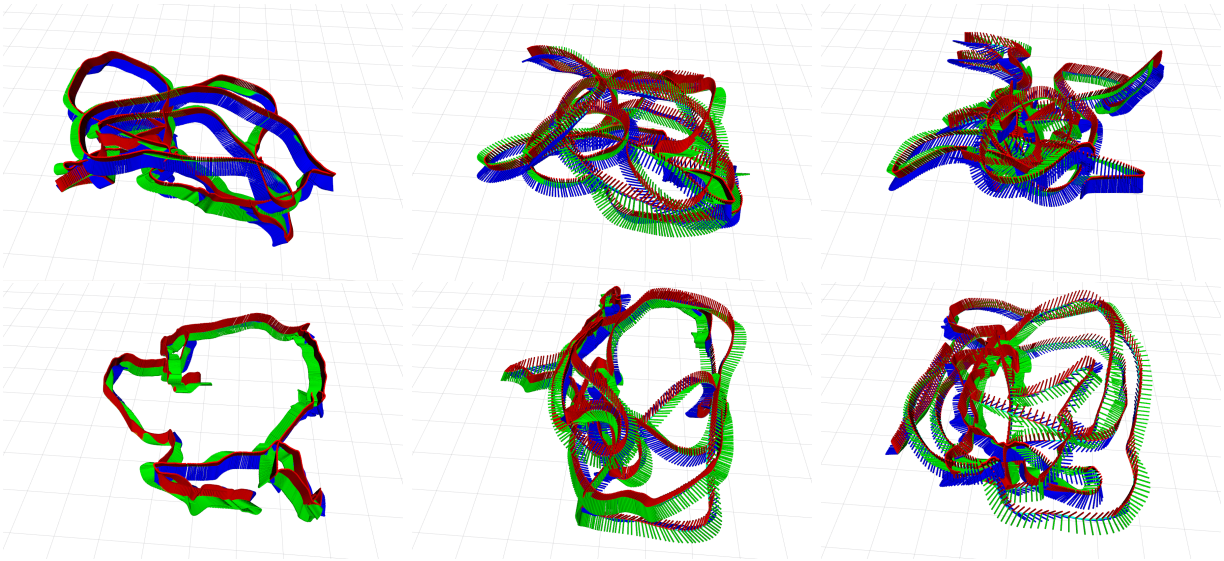

Figure 4: Example generated trajectories on the EuRoC MAV dataset.

Table 2: Average absolute trajectory error (degrees / meters) compared to the provided groundtruth of the EuRoC MAV dataset. Position and yaw alignment was performed.

	V1_01	**V1_02**	**V1_03**	**V2_01**	**V2_02**	**V2_03**
Ori. (deg)	5.789	1.969	2.269	5.779	0.853	0.791
Std. Ori. (deg)	0.161	0.170	0.144	0.978	0.406	0.279
Pos. (m)	**0.036**	0.009	0.006	**0.068**	0.015	0.018
Std. Pos. (m)	0.013	0.004	0.003	0.022	0.007	0.009

Some example recovered trajectories from optimizing the IMU and raw motion capture messages in the EuRoC MAV datasets [9] bag files can be seen in Figure 4. Shown in Table 2, we compare the absolute trajectory error after position and yaw alignment of the generated groundtruth to that provided by the dataset. This dataset provided groundtruth has been time aligned to take into account the motion capture to IMU clocks, and optimized the IMU sensor frame, temporal time offset, and spacial transform between the groundtruth and inertial frame. It is interesting that the V1_01 and V2_01 runs have large position and orientation errors suggesting further investigation. We have noted before that we found that the V1_01 orientation estimates to be poor when compared to the output of an visual-inertial estimator (as compared to the other datasets).

References

[1] Nikolas Trawny and Stergios I. Roumeliotis. *Indirect Kalman Filter for 3D Attitude Estimation*. Tech. rep. University of Minnesota, Dept. of Comp. Sci. & Eng., Mar. 2005.

[2] Christoph Hertzberg et al. "Integrating generic sensor fusion algorithms with sound state representations through encapsulation of manifolds". In: *Information Fusion* 14.1 (2013), pp. 57–77.

[3] Averil B. Chatfield. *Fundamentals of High Accuracy Inertial Navigation*. Reston, VA: American Institute of Aeronautics and Astronautics, Inc., 1997.

[4] Kevin Eckenhoff, Patrick Geneva, and Guoquan Huang. "Closed-form Preintegration Methods for Graph-based Visual-Inertial Navigation". In: *International Journal of Robotics Research* 38.5 (2019), pp. 563–586.

[5] Kevin Eckenhoff, Patrick Geneva, and Guoquan Huang. *Continuous Preintegration Theory for Visual-Inertial Navigation*. Tech. rep. RPNG-2018-CPI. Available: http://udel.edu/~ghuang/papers/tr_cpi.pdf. University of Delaware, 2018.

[6] A. I. Mourikis and S. I. Roumeliotis. "A multi-state constraint Kalman filter for vision-aided inertial navigation". In: *Proceedings of the IEEE International Conference on Robotics and Automation*. Rome, Italy, 2007, pp. 3565–3572.

[7] Patrick Geneva, Kevin Eckenhoff, and Guoquan Huang. "Asynchronous Multi-Sensor Fusion for 3D Mapping and Localization". In: *Proc. of the IEEE International Conference on Robotics and Automation*. Brisbane, Australia, 2018.

[8] Patrick Geneva et al. "OpenVINS: A Research Platform for Visual-Inertial Estimation". In: *Proc. of the IEEE International Conference on Robotics and Automation*. Paris, France, 2020. URL: https://github.com/rpng/open_vins.

[9] Michael Burri et al. "The EuRoC micro aerial vehicle datasets". In: *The International Journal of Robotics Research* 35.10 (2016), pp. 1157–1163.