IEEE Robotics and Automation Letters (RAL) paper presented at the
2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
Madrid, Spain, October 1-5, 2018

On the Comparison of Gauge Freedom Handling in Optimization-based Visual-Inertial State Estimation

Zichao Zhang, Guillermo Gallego, Davide Scaramuzza

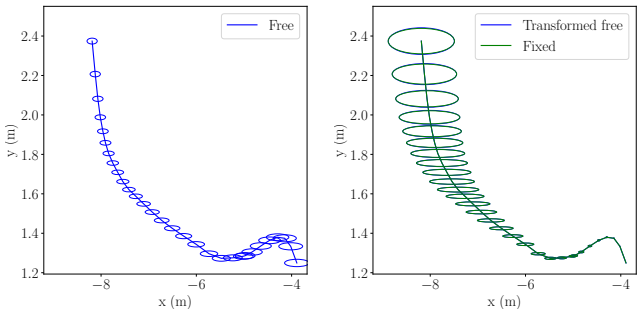
***Abstract*— It is well known that visual-inertial state estimation is possible up to a four degrees-of-freedom (DoF) transformation (rotation around gravity and translation), and the extra DoFs ("gauge freedom") have to be handled properly. While different approaches for handling the gauge freedom have been used in practice, no previous study has been carried out to systematically analyze their differences. In this paper, we present the first comparative analysis of different methods for handling the gauge freedom in optimization-based visual-inertial state estimation. We experimentally compare three commonly used approaches: fixing the unobservable states to some given values, setting a prior on such states, or letting the states evolve freely during optimization. Specifically, we show that (*i*) the accuracy and computational time of the three methods are similar, with the free gauge approach being slightly faster; (*ii*) the covariance estimation from the free gauge approach appears dramatically different, but is actually tightly related to the other approaches. Our findings are validated both in simulation and on real-world datasets and can be useful for designing optimization-based visual-inertial state estimation algorithms.**

I. Introduction

Visual-inertial (VI) sensor fusion is an active research field in robotics. Cameras and inertial sensors are complementary [1], and a combination of both provides reliable and accurate state estimation. While the majority of the research on VI fusion focuses on filter-based methods [2], [3], [4], nonlinear optimization has become increasingly popular within the last few years. Compared with filter-based methods, nonlinear optimization based methods suffer less from the accumulation of linearization errors. Their main drawback, high computational cost, has been mitigated by the advance of both hardware and theory [5], [6]. Recent work [5], [7], [8], [9] has shown impressive real-time VI state estimation results in challenging environments using nonlinear optimization.

Although these works share the same underlying principle, i.e., solving the state estimation as a nonlinear least squares optimization problem, they use different methods to handle the unobservable DoF in VI systems. It is well known that for a VI system, global position and yaw are not observable [3], [10], which in this paper we call *gauge freedom* following the convention from the field of bundle adjustment [11]. Given this gauge freedom, a natural way to get a unique solution is to fix the corresponding states (i.e., parameters) in the optimization [12]. Another possibility is to set a prior on the unobservable states, and the prior essentially acts as a virtual measurement in the optimization [5], [8], [13], [7]. Finally, one may instead allow the optimization algorithm to change the unobservable states freely during the iterations. While these three methods all prove to work in the existing literature, there is no comparison study of their differences in VI state estimation: they are often presented as implementation details and therefore not well studied and understood. Moreover, although the similar problem for vision-only bundle adjustment has already been studied (e.g., [11], [14] with 7 unobservable DoFs in the monocular case), to the best of our knowledge, such a study has not been done for VI systems (which have 4 unobservable DoFs).

In this work, we present the first comparative analysis of the different approaches for handling the gauge freedom in optimization-based visual-inertial state estimation. We compare these approaches, namely the *gauge fixation* approach, the *gauge prior* approach and the *free gauge* approach on simulated and real-world data in terms of their accuracy, computational cost and estimated covariance (which is of interest for, e.g., active SLAM [16]). While all these methods



(a) Free gauge approach (b) Gauge fixation approach

Fig. 1: Different pose uncertainties of the keyframes on the Machine Hall sequence of the EuRoC MAV Dataset [15] (MAV moving toward the negative x direction). The left plot shows the uncertainties from the free gauge approach, where no reference frame is selected. On the right we set the reference frame to be the first frame, and, consequently, the uncertainties grow as the VI system moves. For visualization purposes, the uncertainties have been enlarged. We can clearly identify the difference in the parameter uncertainties from free gauge and gauge fixation approaches. However, by using the covariance transformation in Section VI-B, we show that the free gauge covariance can be transformed to satisfy the gauge fixation condition. The transformed uncertainties agree well with the gauge fixation ones.

Manuscript received February 24, 2018; accepted April 16, 2018. Date of publication May 4, 2018; date of current version May 23, 2018. This letter was recommended for publication by Associate Editor S. Julier and Editor F. Chaumette upon evaluation of the reviewers' comments. This work was supported by the DARPA FLA program, the Swiss National Center of Competence Research Robotics, through the Swiss National Science Foundation, and the SNSF-ERC starting grant. *(Corresponding author: Zichao Zhang.)*

The authors are with the Robotics and Perception Group, Department of Informatics, University of Zürich, 8006 Zürich, and also with the Department of Neuroinformatics, University of Zürich and ETH Zürich, 8092 Zurich, Switzerland (e-mail: zzhang@ifi.uzh.ch; guillermo.gallego@ifi.uzh.ch; davide.scaramuzza@ieee.org).

Digital Object Identifier 10.1109/LRA.2018.2833152

have similar performance in terms of estimation error, the free gauge approach is slightly faster, due to the fewer iterations required for convergence. We also find that, as mentioned by [7], in the free gauge approach, the resulting covariance from the optimization is not associated to any particular reference frame (as opposed to the one from the gauge fixation approach), which makes it difficult to interpret the uncertainties in a meaningful way. However, in this work we further show that by applying a covariance transformation, the free gauge covariance is actually closely related to other approaches (see Fig. 1).

The rest of the paper is organized as follows. In Section II, we introduce the optimization-based VI state estimation problem and its non-unique solution. In Section III we present different approaches for handling gauge freedom. Then we describe the simulation setup for our comparison study in Section IV. The detailed comparison in terms of accuracy/timing and covariance is presented in Sections V and VI, respectively. Finally, we show experimental results on real-world datasets in Section VII.

II. Problem Formulation and Indeterminacies

The problem of visual-inertial state estimation consists of inferring the motion of a combined camera-inertial (IMU) sensor and the locations of the 3D landmarks seen by the camera as the sensor moves through the scene. By collecting the equations of the visual measurements (image points) and the inertial measurements (accelerometer and gyroscope), the problem can be written as a non-linear least squares (NLLS) optimization one, where the goal is to minimize the objective function (e.g., assuming Gaussian errors)

$$J(\boldsymbol{\theta}) \doteq \underbrace{\|\mathbf{r}^V(\boldsymbol{\theta})\|^2_{\Sigma_V}}_{\text{Visual}} + \underbrace{\|\mathbf{r}^I(\boldsymbol{\theta})\|^2_{\Sigma_I}}_{\text{Inertial}}, \tag{1}$$

where $\|\mathbf{r}\|^2_\Sigma = \mathbf{r}^\top \Sigma^{-1} \mathbf{r}$ is the squared Mahalanobis norm of the residual vector $\mathbf{r}$, weighted using the covariance matrix Σ of the measurements. The cost (1) can be used in full smoothing [5] or fixed-lag smoothing [7] approaches.

The visual term in (1) consists of the reprojection error between the measured image points $\mathbf{x}_{ij}$ and the predicted ones $\hat{\mathbf{x}}_{ij}$ by a metric reconstruction. Assuming a pinhole camera model, $\hat{\mathbf{x}}_{ij}(\boldsymbol{\theta}) \propto \mathtt{K}_i(\mathtt{R}_i^\top | -\mathtt{R}_i^\top \mathbf{p}_i)(\mathbf{X}_j^\top, 1)^\top$, where $(\mathtt{R}_i, \mathbf{p}_i)$ are the extrinsic parameters of the i-th camera ($i = 0, \dots, N-1$) and $\mathbf{X}_j$ are the 3D Euclidean coordinates of the j-th landmark point ($j = 0, \dots, K-1$). We assume that the intrinsic calibrations $\mathtt{K}_i$ are noise-free. The inertial term in (1) consists of the error between the inertial measurements and the predicted ones by a model of the trajectory of the IMU. For example, [17] considers the error in the raw acceleration and angular velocity measurements, whereas [5] considers errors in equivalent, lower rate measurements (inertial preintegration terms at the rate of the visual data). In this work, we consider the latter formulation, although most of the results do not depend on the choice of formulation.

The parameters of the problem (also known as *state*),

$$\boldsymbol{\theta} \doteq \{\mathbf{p}_i, \mathtt{R}_i, \mathbf{v}_i, \mathbf{X}_j\}, \tag{2}$$

comprise the camera motion parameters[1] (extrinsics and linear velocity) and the 3D scene (landmarks).

The accelerometer and gyroscope biases are usually expressed in the IMU frame and thus not affected by a fixation of the coordinate frame. Therefore, we exclude the biases from the state and assume that the IMU measurements are already corrected. A full description of the inertial and visual measurement models is out of the scope of this work, and we refer the reader to [5] for details.

A. Solution Ambiguities and Geometrical Equivalence

When addressing the VI state estimation problem, it is essential to note that the objective function (1) is *invariant* to certain transformations of the parameters $\boldsymbol{\theta}' = g(\boldsymbol{\theta})$, i.e.,

$$J(\boldsymbol{\theta}) = J(g(\boldsymbol{\theta})). \tag{3}$$

Specifically, g, defined by homogeneous matrices of the form

$$g \doteq \begin{pmatrix} \mathtt{R}_z & \mathbf{t} \\ 0 & 1 \end{pmatrix}, \tag{4}$$

is a 4-DoF transformation consisting of an arbitrary translation $\mathbf{t} \in \mathbb{R}^3$ and a rotation $\mathtt{R}_z = \mathrm{Exp}(\alpha \mathbf{e}_z)$ by an arbitrary angle (yaw) $\alpha \in (-\pi, \pi)$ around the gravity axis $\mathbf{e}_z = (0, 0, 1)^\top$. For notation simplicity, we define the mapping $\mathrm{Exp}(\boldsymbol{\theta}) \doteq \exp(\boldsymbol{\theta}^\wedge)$, where exp is the exponential map of the Special Orthogonal group $SO(3)$, and $\boldsymbol{\theta}^\wedge$ is the skew-symmetric matrix associated with the cross-product, i.e., $\mathbf{a}^\wedge \mathbf{b} = \mathbf{a} \times \mathbf{b}, \forall \mathbf{b}$. This is the well-known Rodrigues formula.

Applying a transformation (4) to the reconstruction (2) gives another reconstruction $g(\boldsymbol{\theta}) = \boldsymbol{\theta}' \equiv \{\mathbf{p}'_i, \mathtt{R}'_i, \mathbf{v}'_i, \mathbf{X}'_j\}$,

$$\begin{aligned} \mathbf{p}'_i &= \mathtt{R}_z \mathbf{p}_i + \mathbf{t} & \mathtt{R}'_i &= \mathtt{R}_z \mathtt{R}_i \\ \mathbf{v}'_i &= \mathtt{R}_z \mathbf{v}_i & \mathbf{X}'_j &= \mathtt{R}_z \mathbf{X}_j + \mathbf{t} \end{aligned} \tag{5}$$

Both parameters $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ represent the same underlying scene geometry (camera trajectory and 3D points), i.e., they are *geometrically equivalent*. They generate the same predicted measurements; and, therefore, the same error (1).

As a consequence of the invariance (3), the parameter space $\mathcal{M}$ can be partitioned into disjoint sets of geometrically equivalent reconstructions. Each of these sets is called an *orbit* [11] or a leaf [14]. Formally, the orbit associated to $\boldsymbol{\theta}$ is the 4D manifold

$$\mathcal{M}_{\boldsymbol{\theta}} \doteq \{g(\boldsymbol{\theta}) \,|\, g \in \mathcal{G}\}, \tag{6}$$

where $\mathcal{G}$ is the group of transformations of the form (4). Note that the objective function (1) is constant on each orbit.

The main consequence of the invariance (3) is that (1) does not have a *unique* minimizer because there are infinitely many reconstructions that achieve the same minimum error: all the reconstructions on the orbit (6) of minimal cost (see Fig. 2), differing only by 4-DoF transformations (4). Hence, the VI estimation problem has some *indeterminacies* or unobservable states: there are not enough equations to completely specify a unique solution.

[1] For simplicity, we assume that the coordinate frames of the camera and the IMU coincide, e.g., by compensating the camera-IMU calibration [18].

TABLE I: Three gauge handling approaches considered. ($n = 9N + 3K$ is the number of parameters in (2))

	Size of parameter vec.	Hessian (Normal eqs)
Fixed gauge	$n-4$	inverse, $(n-4)\times(n-4)$
Gauge prior	n	inverse, $n \times n$
Free gauge	n	pseudoinverse, $n \times n$

B. Additional Constraints: Specifying a Gauge

The process of completing (1) with additional constraints

$$\mathbf{c}(\boldsymbol{\theta}) = \mathbf{0} \tag{7}$$

that yield a unique solution is called specifying a *gauge* $\mathcal{C}$ [14], [11]. In other words, equations (7) select a representative of the orbit (6), i.e., to remove the indeterminacy within the equivalence class. In VI, this is achieved by specifying a reference coordinate frame for the 3D reconstruction. For example, the *standard gauge* in camera-motion estimation consists of selecting the reconstruction that has the reference coordinate frame located at the first ($i = 0$) camera position and with zero yaw. These constraints specify a unique transformation (4), and therefore, a unique solution $\boldsymbol{\theta}_C = \mathcal{C} \cap \mathcal{M}_{\boldsymbol{\theta}}$ among all equivalent ones. By construction, gauges $\mathcal{C}$ are transversal to orbits $\mathcal{M}_{\boldsymbol{\theta}}$, so that $\boldsymbol{\theta}_C \neq \emptyset$ [14].

III. Optimization and Gauge Handling

From an optimization point of view, the minimization of the NLLS function (1) using the Gauss-Newton algorithm presents some difficulties. Even if we use a minimal parametrization for all elements of the state (parameter vector) $\boldsymbol{\theta}$, the Hessian matrix of (1), which drives the parameter updates, is singular due to the unobservable DoFs. More specifically, it has a rank deficiency of four, corresponding to the 4-DoFs in (4).

There are several ways to mitigate this issue, as summarized in Table I. One of them is to optimize in a smaller parameter space where there are no unobservable states, and therefore the Hessian is invertible. This essentially enforces hard constraints on the solution (*gauge fixation* approach). Another one is to augment the objective function with an additional penalty (which yields an invertible Hessian) to favor that the solution satisfies certain constraints, in a soft manner (*gauge prior* approach). Lastly, one can use the pseudoinverse of the singular Hessian to implicitly provide additional constraints (parameter updates with smallest norm) for a unique solution (*free gauge* approach). The first two strategies require VI problem-specific knowledge (which state to constrain), whereas the last one is generic.

A. Rotation Parametrization for Gauge Fixation or Prior

One problem with the gauge fixation and gauge prior approaches is that fixing the 1-DoF yaw rotation angle of a camera pose is not straightforward, as we discuss next.

The standard method to update orientation variables (i.e., rotations) during the iterations of the NLLS solver (Gauss-Newton or Levenberg-Marquardt–LM) of (1) is to use local coordinates, where, at the q-th iteration, the update is

$$\mathtt{R}^{q+1} = \mathrm{Exp}(\delta\boldsymbol{\phi}^q)\mathtt{R}^q. \tag{8}$$

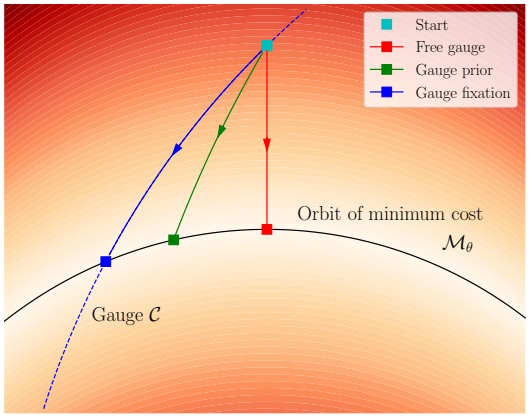


Fig. 2: Illustration of the optimization paths taken by different gauge handling approaches. The gauge fixation approach always moves on the gauge $\mathcal{C}$, thus satisfying the gauge constraints. The free gauge approach uses the pseudoinverse to select parameter steps of minimal size for a given cost decrease, and therefore, moves perpendicular to the isocontours of the cost (1). The gauge prior approach follows a path in between the gauge fixation and free gauge approaches. It minimizes a cost augmented by (11), so it may not exactly end up on the orbit of minimum visual-inertial cost (1).

Setting the z component of $\delta\boldsymbol{\phi}^q$ to 0 allows fixating the yaw with respect to $\mathtt{R}^q$. However, concatenating several such updates (Q iterations), $\mathtt{R}^Q = \prod_{q=0}^{Q-1} \mathrm{Exp}(\delta\boldsymbol{\phi}^q)\mathtt{R}^0$, does not fixate the yaw with respect to the initial rotation $\mathtt{R}^0$, and therefore, this parametrization cannot be used to fix the yaw-value of $\mathtt{R}^Q$ to that of the initial value $\mathtt{R}^0$.

Although yaw fixation or prior can be applied to any camera pose, it is a common practice to use the first camera. Thus, for the rotations of the other camera poses, we use the standard iterative update (8), and, for the first camera, $\mathtt{R}_0$, we use a more convenient parametrization. Instead of directly using $\mathtt{R}_0$, we use a left-multiplicative increment:

$$\mathtt{R}_0 = \mathrm{Exp}(\Delta\boldsymbol{\phi}_0)\mathtt{R}_0^0, \tag{9}$$

where the rotation vector $\Delta\boldsymbol{\phi}_0$ is initialized to zero and updated. Indeed, the rotation vector formulation has a singularity at $\|\Delta\boldsymbol{\phi}_0\| = \pi$, but it is applicable when the initial rotation is close to the optimal value ($\|\Delta\boldsymbol{\phi}_0\| < \pi$), which is often the case in real systems (e.g., initial values are provided by a front-end, such as [5]).

B. Different Approaches for Handling Gauge Freedom

Based on the previous discussion, *gauge fixation* consists of fixing the position and yaw angle of the first camera pose throughout the optimization. This is achieved by setting

$$\mathbf{p}_0 = \mathbf{p}_0^0, \qquad \Delta\boldsymbol{\phi}_{0z} \doteq \mathbf{e}_z^\top \Delta\boldsymbol{\phi}_0 = 0, \tag{10}$$

where $\mathbf{p}_0^0$ is the initial position of the first camera. Fixing these values of the parameter vector is equivalent to setting the corresponding columns of the Jacobian of the residual vector in (1) to zero, namely $\mathtt{J}_{\mathbf{p}_0} = \mathbf{0}$, $\mathtt{J}_{\Delta\boldsymbol{\phi}_{0z}} = \mathbf{0}$.

The *gauge prior* approach adds to (1) a penalty

$$\|\mathbf{r}_0^P\|^2_{\Sigma_0^P}, \quad \text{where} \quad \mathbf{r}_0^P(\boldsymbol{\theta}) \doteq (\mathbf{p}_0 - \mathbf{p}_0^0,\ \Delta\boldsymbol{\phi}_{0z}). \tag{11}$$

The choice of Σ_0^P in (11) will be discussed in Section V.

Finally, the *free gauge* approach lets the parameter vector evolve freely during the optimization. To deal with the singular Hessian, we may use the pseudoinverse or add some damping (Levenberg-Marquardt algorithm) so that the NLLS problem has a well-defined parameter update.

A comparison of the paths followed in parameter space during the optimization iterations of the three approaches is illustrated in Fig. 2.

Next, we show an experimental comparison of the three gauge handling approaches.

IV. Comparison Study: Simulation Setup

A. Data Generation

We use three 6-DoF trajectories for our experiments, namely a sine-like shape one, an arc-like one and a rectangular one. We denote them as *sine*, *arc* and *rec* respectively. We consider two landmark configurations: *plane*, where the 3D points are roughly distributed on several planes and *random*, where the 3D points are generated randomly along the trajectory. Fig. 3 shows some simulation setup examples.

To generate the inertial measurements, we fit the trajectories using B-splines and then sample the accelerations and angular velocities. The sampled values are corrupted with biases and additive Gaussian noise, and then are used as inertial measurements. For the visual measurements, we project the 3D points through a pinhole camera model to get the corresponding image coordinates and then corrupt them with additive Gaussian noise.

B. Optimization Solver

To solve the VI state estimation problem (1), we use the LM algorithm in the Ceres solver [19]. We implement the different approaches for handling the gauge freedom described in Section III. For each trajectory, we sample several keyframes along the trajectory. Our parameter space contains the states (i.e., position, rotation and velocity) at these keyframes and the positions of the 3D points. The initial states are disturbed randomly from the groundtruth.

C. Evaluation

1) Accuracy: To evaluate the accuracy of an estimated state, we first calculate a transformation to align the estimation and the groundtruth. The transformation is calculated from the first poses of both trajectories. Note that the transformation has four DoFs, i.e., a translation and a rotation around the gravity vector. After alignment, we calculate the root mean squared error (RMSE) of all the keyframes. Specifically, we use the Euclidean distance for position and velocity errors. For rotation estimation, we first calculate the relative rotation (in angle-axis representation) between the aligned rotation and the groundtruth, and then use the angle of the relative rotation as the rotation error.

2) Computational Efficiency: To evaluate the computational cost, we record the convergence time and number of iterations of the solver. We run each configuration (i.e., the combination of trajectory and points) for 50 trials and calculate the average time and accuracy metrics.

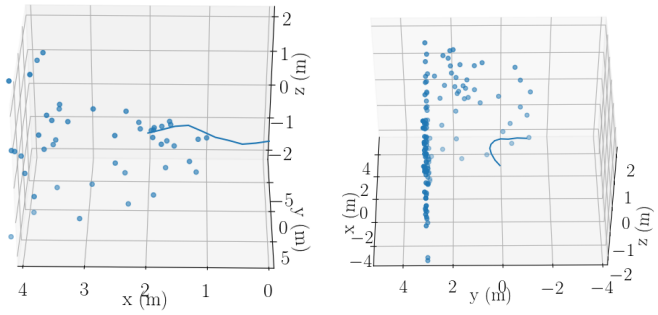


Fig. 3: Sample simulation scenarios. The left one shows a *sine* trajectory with randomly generated 3D points, and the right one shows an *arc* trajectory with the 3D points distributed on two planes.

3) Covariance: We also compare the covariances produced by the optimization algorithm, which are of interest for applications such as active SLAM [20]. The covariance matrix of the estimated parameters is given by the inverse of the Hessian. For the free gauge approach, the Moore-Penrose pseudoinverse is used, since the Hessian is singular [11].

V. Comparison Study: Timing and Accuracy

A. Gauge Prior: Choosing the Appropriate Prior Weight

Before comparing the three approaches from Section III, we need to choose the prior covariance Σ_0^P in the gauge prior approach. A common choice is $\Sigma_0^P = \sigma_0^2 \mathtt{I}$, for which the prior (11) becomes $\|\mathbf{r}_0^P\|^2_{\Sigma_0^P} = w^P \|\mathbf{r}_0^P\|^2$, with $w^P = 1/\sigma_0^2$. We tested a wide range of the prior weight w^P on different configurations and the results were similar. Therefore, we will look at one configuration in detail. Note that $w^P = 0$ is essentially the free gauge approach, whereas $w^P \to \infty$ is the gauge fixation approach.

1) Accuracy: Fig. 4 shows how the RMSE changes with the prior weight. It can be seen that the estimation errors of different prior weights are very similar (note the numbers on the vertical axis). While there is no clear optimal prior weight for different configurations of trajectories and 3D points, the RMSE stabilizes at one value after the weight increases above a certain threshold (e.g., 500 in Fig. 4).

2) Computational Cost: Fig. 5 illustrates the computational cost for different prior weights. Similarly to Fig. 4, the number of iterations and the convergence time stabilize when the prior weight is above a certain value. Interestingly, there is a peak in the computational time when the prior weight increases from zero to the threshold where it stabilizes. The same behavior is observed for all configurations. To investigate this behavior in detail, we plot in Fig. 6 the prior error with respect to the average reprojection error at each iteration for several prior weight values. The position prior error is the Euclidean distance between the current estimate of the first position and its initial value, the yaw prior error is the z-component of the relative rotation of the current estimate of the first rotation with respect to its initial value, and the average reprojection error is the total visual residual averaged by the number of observed 3D points in all keyframes. For very large prior weights (10^8 in the plot), the algorithm decreases the reprojection error while keeping the

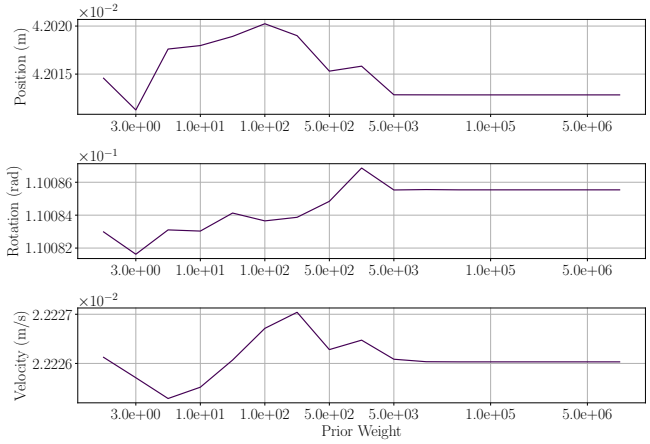


Fig. 4: RMSE in position, orientation and velocity for different prior weights

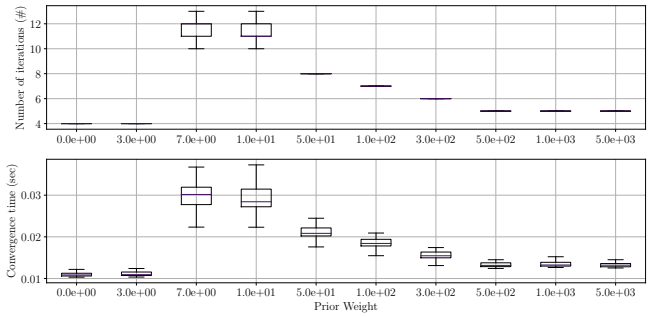


Fig. 5: Number of iterations and computing time for different prior weights.

prior error almost equal to zero. In contrast, for smaller prior weights (e.g., 50–500), the optimization algorithm reduces the reprojection error during the first two iterations at the expense of increasing the prior error. Then the optimization algorithm spends many iterations fine-tuning the prior error while keeping the reprojection error small (moving along the orbit), hence the computational time increases.

3) Discussion: While the accuracy of the solution does not significantly change for different prior weights (Fig. 4), a proper choice of the prior weight is required in the gauge prior approach to keep the computational cost small (Fig. 5). Extremely large weights are discarded since they sometimes make the optimization unstable. We observe similar behavior for different configurations (trajectory and points combination). Therefore, in the rest of the section we use a proper prior weight (e.g., 10^5) for the gauge prior approach.

B. Accuracy and Computational Effort

We compare the performance of the three approaches on the six combinations of simulated trajectories (*sine*, *arc* and *rec*) and 3D points (*plane* and *random*). We optimize the objective function for differently perturbed initializations and observe that the results are similar. For the results presented in this section, we perturb the groundtruth positions by a random vector of $5\,\mathrm{cm}$ (with respect to a trajectory of $5\,\mathrm{m}$), the orientations by a random rotation of 6 degrees, the velocities by a uniformly distributed variable in $[-0.05, 0.05]\ \mathrm{m/s}$ (with respect to a mean velocity of $2\,\mathrm{m/s}$) and the 3D point positions by a uniform random variable in $[-7.5, 7.5]\ \mathrm{cm}$.

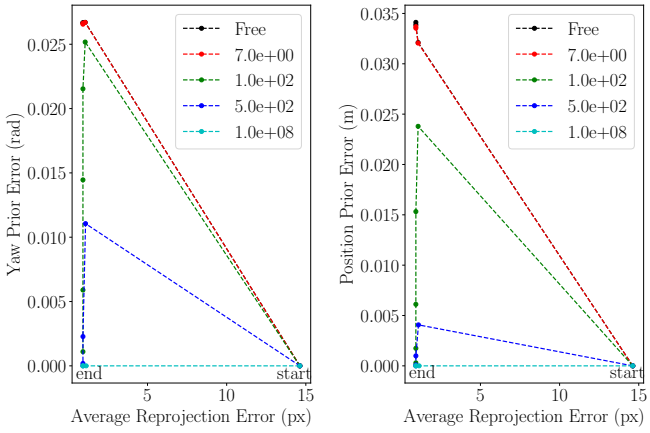


Fig. 6: Prior error vs. average reprojection error for some representative prior weights. Each dot in the plot stands for an iteration with the corresponding prior weight. The optimization starts from the bottom-right corner, where the reprojection errors are the same and the prior errors are zero. As the optimization proceeds, the reprojection error decreases and there are different behaviors for different prior weights regarding the prior error. Note that the free gauge case behaves as the zero prior weight.

The average RMSEs of 50 trials are listed in Table II. We omit the results for the gauge prior approach because they are identical to the ones from the gauge fixation approach up to around 8 digits after the decimal. It can be seen that there are only small differences between the free gauge approach and the gauge fixation approach, and neither of them has a better accuracy in all simulated configurations.

The convergence time and number of iterations are plotted in Fig. 7. The computational cost of the gauge prior approach and the gauge fixation approach are almost identical. The free gauge approach is slightly faster than the other two. Specifically, except for the *sine* trajectory with random 3D points, the free gauge approach takes fewer iterations and less time to converge. Note that the gauge fixation approach takes the least time per iteration due to the smaller number of variables in the optimization (see Table I).

C. Discussion

Based on the results in this section, we conclude that:

- The three approaches have almost the same accuracy.
- In the gauge prior approach, one needs to select the proper prior weight to avoid increasing the computational cost.
- With a proper weight, the gauge prior approach has almost the same performance (accuracy and computational cost) as the gauge fixation approach.
- The free gauge approach is slightly faster than the others, because it takes fewer iterations to converge (cf. [14]).

While it may be possible to fix the unobservable DoFs (recall that we use a tailored parametrization (9) to fix the yaw DoF), the free gauge approach has the additional advantage that is generic, i.e., not specific of VI, and therefore it does not require any special treatment on rotation parametrization.

VI. Comparison Study: Covariance

A. Covariance Comparison

Given a high prior weight, as discussed in the previous section, the covariance matrix from the gauge prior approach

TABLE II: RMSE on different trajectories and 3D points configurations. The smallest errors (e.g., $\mathbf{p}$ gauge fixation vs. $\mathbf{p}$ free gauge) are highlighted.

Configuration	Gauge fixation			Free gauge		
	$\mathbf{p}$	ϕ	$\mathbf{v}$	$\mathbf{p}$	ϕ	$\mathbf{v}$
sine plane	0.04141	0.1084	**0.02182**	0.04141	0.1084	0.02183
arc plane	**0.02328**	0.6987	0.01303	0.02329	0.6987	0.01303
rec plane	**0.01772**	0.1668	0.01496	0.01774	0.1668	**0.01495**
sine random	0.03932	0.0885	0.01902	**0.03908**	**0.0874**	**0.01886**
arc random	0.02680	0.6895	0.01167	**0.02678**	0.6895	**0.01166**
rec random	**0.02218**	0.1330	0.009882	0.02220	0.1330	**0.009881**

Position, rotation and velocity RMSE are measured in m, deg and m/s, respectively.

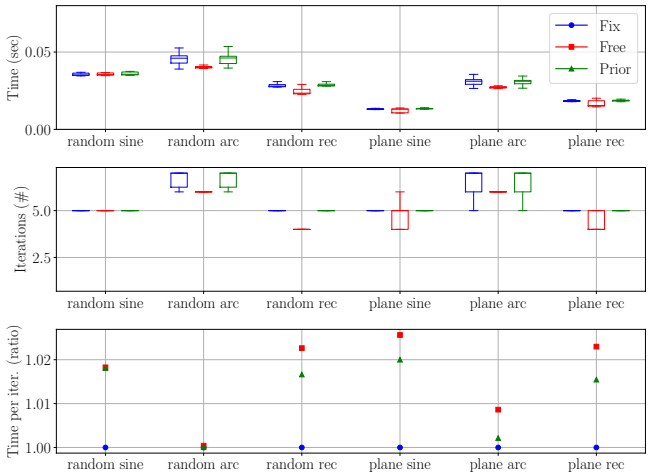


Fig. 7: Number of iterations, total convergence time and time per iteration for all configurations. The time per iteration is the ratio with respect to the gauge fixation approach (in blue), which takes least time per iteration.

is similar to the gauge fixation approach and therefore omitted here. We only compare the covariances of the free gauge approach and the gauge fixation approach in this section. An example of the covariance matrices of the free gauge and gauge fixation approaches is visualized in Fig. 9. If we look at the top-left block of the covariance matrix, which corresponds to the position components of the states: (*i*) for the gauge fixation approach (Fig. 9c), the uncertainty of the first position is zero due to the fixation, and the position uncertainty increases afterwards (cf. Fig. 1b); (*ii*) in contrast, the uncertainty in the free gauge case (Fig. .9a) is "distributed" over all the positions (cf. Fig. 1a). This is due to the fact that the free gauge approach is not fixed to any reference frame. Therefore, the uncertainties directly read from the free gauge covariance matrix are not interpretable in a geometrically-meaningful way. However, this does not mean the covariance estimation from the free gauge approach is useless: it can be transformed to a geometrically-meaningful form by enforcing a gauge fixation condition, as we show next.

B. Covariance Transformation

Covariances are averages of squared perturbations of the estimated parameter. A perturbation $\Delta\boldsymbol{\theta}$ of a reconstruction $\boldsymbol{\theta}$ can be decomposed into two components: one parallel to the orbit $\mathcal{M}_{\boldsymbol{\theta}}$ (6) and one parallel to the gauge $\mathcal{C}$ (7). The component of $\Delta\boldsymbol{\theta}$ parallel to the orbit $\mathcal{M}_{\boldsymbol{\theta}}$ is not geometrically meaningful since the perturbed reconstruction is also in the

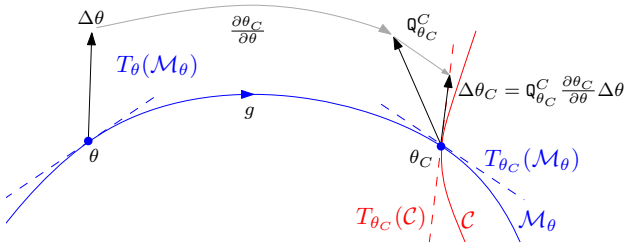


Fig. 8: Illustration of the covariance transformation in the parameter space. $\mathcal{M}_{\boldsymbol{\theta}}$ is the subspace that contains all the parameters that are equivalent to free gauge estimation $\boldsymbol{\theta}$ (i.e., different by a 4-DoF transformation). C is that subspace that contains all the parameters that satisfy the gauge fixation condition (10). We first transform $\boldsymbol{\theta}$ to the gauge fixation estimation $\boldsymbol{\theta}_C$ along $\mathcal{M}_{\boldsymbol{\theta}}$, together with the perturbation $\Delta\boldsymbol{\theta} \mapsto (\partial\boldsymbol{\theta}_C/\partial\boldsymbol{\theta})\Delta\boldsymbol{\theta}$. Then we project the perturbation onto the tangent space to the gauge $T_{\boldsymbol{\theta}_C}(\mathcal{C})$, parallel to the $\mathcal{M}_{\boldsymbol{\theta}}$, using the projector $\mathtt{Q}^C_{\boldsymbol{\theta}_C}$. The average of the outer product of these transformed perturbations is the covariance $\mathrm{Cov}(\boldsymbol{\theta}_C)$.

orbit (thus, arbitrarily large perturbations produce no change of the scene geometry). Therefore, only perturbations along the gauge $\mathcal{C}$, $\Delta\boldsymbol{\theta}_C$, represent changes of the reconstructed geometry and are therefore meaningful. Such perturbations live on the tangent space $T_{\boldsymbol{\theta}_C}(\mathcal{C})$. Hence, geometrically-meaningful perturbations are gauge-dependent [14], [11].

The covariance from the free gauge approach $\mathrm{Cov}^*(\boldsymbol{\theta})$ at an estimate $\boldsymbol{\theta}$ can be transformed into the covariance of a given gauge fixation $\mathcal{C}$ (10) by the following formula [14]:

$$\mathrm{Cov}(\boldsymbol{\theta}_C) \approx \left(\mathtt{Q}^C_{\boldsymbol{\theta}_C}\frac{\partial\boldsymbol{\theta}_C}{\partial\boldsymbol{\theta}}\right)\overset{*}{\mathrm{Cov}}(\boldsymbol{\theta})\left(\mathtt{Q}^C_{\boldsymbol{\theta}_C}\frac{\partial\boldsymbol{\theta}_C}{\partial\boldsymbol{\theta}}\right)^\top, \quad (12)$$

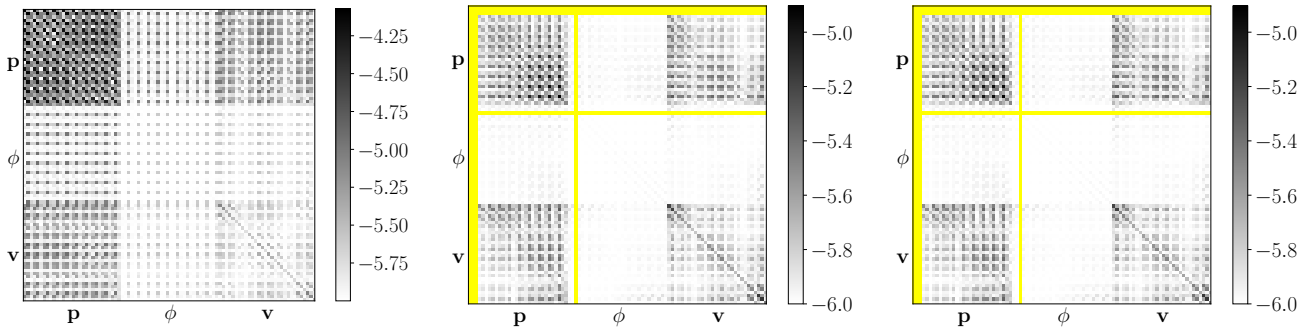
where $\boldsymbol{\theta}_C = \mathcal{C}\cap\mathcal{M}_{\boldsymbol{\theta}} = g(\boldsymbol{\theta})$ is the equivalent parameter that satisfies the gauge. Specifically, $g \equiv \{\mathtt{R}_z, \mathbf{t}\}$ (4) is obtained by "pushing" $\boldsymbol{\theta}$ along $\mathcal{M}_{\boldsymbol{\theta}}$ (Fig. 8) until it meets $\mathcal{C}$, satisfying

$$\begin{aligned} \mathbf{p}_0^C &= \mathtt{R}_z\mathbf{p}_0 + \mathbf{t}, \\ 0 &= \mathbf{e}_z^\top \mathrm{Log}(\mathtt{R}_z\,\mathrm{Exp}(\Delta\boldsymbol{\phi}_0)), \end{aligned} \quad (13)$$

where $\{\mathbf{p}_0, \Delta\boldsymbol{\phi}_0\} \in \boldsymbol{\theta}$ and $\mathbf{p}_0^C \in \boldsymbol{\theta}_C$. Recall that the rotation of the first camera pose is parameterized differently (9), and therefore should be transformed as $\Delta\boldsymbol{\phi}_0^C = \mathrm{Log}(\mathtt{R}_z\,\mathrm{Exp}(\Delta\boldsymbol{\phi}_0))$, where Log is the inverse operator of Exp, defined in Section II-A.

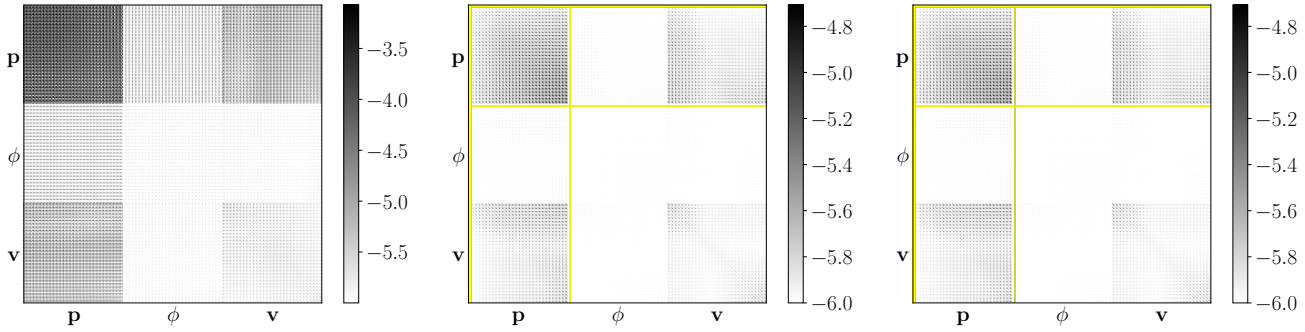
The transformation rule (12) consists of two operations (also illustrated in Fig. 8): (*i*) transferring perturbations along the orbit $\mathcal{M}_{\boldsymbol{\theta}}$ (operator $\partial\boldsymbol{\theta}_C/\partial\boldsymbol{\theta}$), and (*ii*) projecting the perturbations on the tangent space to the gauge $T_{\boldsymbol{\theta}_C}(\mathcal{C})$ (operator $\mathtt{Q}^C_{\boldsymbol{\theta}_C}$). These operators are specified in Appendix I.

In Fig. 9, we show an example of covariance transformation on simulated data. Because VI systems are mostly used for motion estimation, we only show the covariance of the motion parameters. To better appreciate the entries of the covariance in spite of their magnitude difference, we use a logarithmic scale for visualization. Specifically, we plot $\log_{10}(|\sigma_{ij}| + \varepsilon)$, where $\mathrm{Cov} \equiv \Sigma = (\sigma_{ij})$ is the covariance matrix, and $\varepsilon = 10^{-7}$ defines the value corresponding to the white color. We transform the free gauge covariance to the reference frame specified by the gauge fixation constraint (10). It can be seen that the transformed covariance agrees well with the covariance from the the gauge fixation, with a very small relative error in Frobenius norm (0.11 %).



(a) Free gauge covariance (b) Transformed free gauge covariance (c) Gauge fixation covariance

Fig. 9: Covariance of free gauge (Fig. 9a) and gauge fixation (Fig. 9c) approaches using $N = 10$ keyframes. In the middle (Fig. 9b), the free gauge covariance transformed using (12) shows very good agreement with the gauge fixation covariance: the relative difference between them is $\|\Sigma_b - \Sigma_c\|_F / \|\Sigma_c\|_F \approx 0.11\%$ ($\|\cdot\|_F$ denotes Frobenius norm). For better visualization, the magnitude of the covariance entries is displayed in logarithmic scale. The yellow bands of the gauge fixation and transformed covariances indicate zero entries due to the fixed 4-DoFs (the position and the yaw angle of the first camera).



(a) Free gauge covariance (b) Transformed free gauge covariance (c) Gauge fixation covariance

Fig. 10: Covariance comparison and transformation using $N = 30$ keyframes of the EuRoC Vicon 1 sequence (VI1). Same color scheme as in Fig. 9. The relative difference between (b) and (c) is $\|\Sigma_b - \Sigma_c\|_F / \|\Sigma_c\|_F \approx 0.02\%$. Observe that, in the gauge fixation covariance, the uncertainty of the first position and yaw is zero, and it grows for the rest of the camera poses (darker color), as illustrated in Fig. 1b.

C. Discussion

In this section, we have seen that the parameter covariance from the free gauge approach is different from the other approaches and cannot be directly interpreted in a meaningful way. However, we can actually transform the free gauge covariance into the gauge fixation one by a linear transformation (12). The covariance transformation method in Section VI-B, which is a special case of the general theory in [14], not only provides insights into the differences and connections of the compared methods, but it can also be useful for covariance calculation if the optimization method is used as a black box (i.e., cannot directly calculate the covariance—inverse of the Hessian matrix—from the Jacobians of the measurement model).

VII. Experiments on Real-World Datasets

We performed the same experimental comparison as in the simulation on two sequences from the EuRoC MAV Dataset [15]: *Machine Hall 1* (MH1) and *Vicon Room 1* (VI1). We used a semi-direct visual odometry algorithm (SVO [21]) to provide the initialization of the parameters in the optimization problem (1). We used the stereo setup of SVO to remove scale ambiguity. As for the biases, we used the groundtruth values in the dataset. The evaluation method described in Section IV was used. Note that we did not run the optimization over the full trajectories but on shorter segments, which is enough to demonstrate the differences of the three methods. The computational cost of the three different approaches is plotted in Fig. 11. The results are consistent with our simulation experiments: the free gauge approach, which requires fewer iterations to converge, is faster than the other two, The accuracies are reported in Table III, and all three methods have similar estimation error. In Fig. 10, we observe, as in Fig. 9, the aparent difference between the covariances and further show that, by applying (12), we can calculate the covariance in a certain reference frame using the free gauge covariance, and the result agrees well with the covariance from actually fixating the gauge (cf. Fig. 10b and Fig. 10c).

VIII. Conclusion

In this work, we presented the first comparison study of different approaches, namely the gauge fixation approach, the gauge prior approach and the free gauge approach, for handling the gauge freedom in optimization-based visual-inertial state estimation. We showed in simulation as well as on real-world datasets that all these methods have similar accuracy and efficiency, with the free gauge approach being slightly faster due to fewer iterations in the optimization. However, one major difference we identified is the estimated

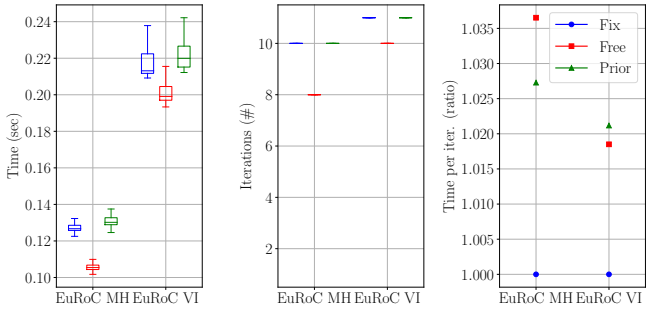


Fig. 11: Computational cost of the three different methods for handling gauge freedom on two sequences from the EuRoC dataset. The time per iteration is the ratio with respect to the gauge fixation approach.

TABLE III: RMSE on EuRoC datasets. Same notation as in Table II.

Sequence	Gauge fixation			Free gauge		
	$\mathbf{p}$	$\boldsymbol{\phi}$	$\mathbf{v}$	$\mathbf{p}$	$\boldsymbol{\phi}$	$\mathbf{v}$
EuRoC MH	0.06936	**0.07845**	0.03092	**0.06918**	0.07857	**0.03091**
EuRoC VI	0.07851	0.4382	0.04644	0.07851	0.4382	0.04644

covariance from the optimization algorithms are different, especially for the free gauge approach. To better understand the connection between the different approaches, we showed how to transform the free gauge covariance to satisfy the gauge fixation condition, which indicates the covariances from different approaches are actually closely related.

Appendix I
Operators For Covariance Transformation

The Jacobian $\frac{\partial \boldsymbol{\theta}_C}{\partial \boldsymbol{\theta}}$ in (12) is computed from g according to the relations (5) and the chosen parametrization of $\boldsymbol{\theta}, \boldsymbol{\theta}_C$. It is a block-diagonal, full-rank square matrix of size $9N + 3K$. Differentiating on (5), we obtain the matrices in the diagonal, $\partial \mathbf{p}_i^C / \partial \mathbf{p}_i = \partial \mathbf{v}_i^C / \partial \mathbf{v}_i = \partial \mathbf{X}_j^C / \partial \mathbf{X}_j = \mathtt{R}_z$. Differentiating the rotation parameters, we have, for the first camera pose (parametrization (9)), $\partial \Delta\boldsymbol{\phi}_0^C / \partial \Delta\boldsymbol{\phi}_0 = \mathtt{J}_r^{-1}(\Delta\boldsymbol{\phi}_0^C)\, \mathtt{J}_r(\Delta\boldsymbol{\phi}_0)$, where $\mathtt{J}_r$ is the right Jacobian of $SO(3)$ [22, p. 40], and for the remaining poses (parametrization (8)), $\partial \delta\boldsymbol{\phi}_i^C / \partial \delta\boldsymbol{\phi}_i = \mathtt{R}_z$.

The oblique projector $\mathtt{Q}_{\boldsymbol{\theta}_C}^C$ in (12) is given by

$$\mathtt{Q}_{\boldsymbol{\theta}_C}^C \doteq \mathtt{I} - \mathtt{U}_{\boldsymbol{\theta}_C} (\mathtt{V}_{\boldsymbol{\theta}_C}^\top \mathtt{U}_{\boldsymbol{\theta}_C})^{-1} \mathtt{V}_{\boldsymbol{\theta}_C}^\top, \tag{14}$$

where $\mathtt{I}$ is the identity matrix, $\mathtt{U}_{\boldsymbol{\theta}_C}$ is a basis for the tangent space to the orbit at $\boldsymbol{\theta}_C$, $T_{\boldsymbol{\theta}_C}(\mathcal{M}_{\boldsymbol{\theta}})$, and $\mathtt{V}_{\boldsymbol{\theta}_C}$ is a basis for the orthogonal complement of the tangent space to the gauge $\mathcal{C}$ at $\boldsymbol{\theta}_C$, $(T_{\boldsymbol{\theta}_C}(\mathcal{C}))^\perp$ (Fig. 8). Both $\mathtt{U}_{\boldsymbol{\theta}_C}$ and $\mathtt{V}_{\boldsymbol{\theta}_C}$ are $(9N{+}3K){\times}4$ matrices and their specific form depend on the choice of parametrization and gauge constraints. Matrix $\mathtt{U}_{\boldsymbol{\theta}_C}$ can be obtained by applying to the parameter $\boldsymbol{\theta}_C$ an infinitesimal transformation (4), $\delta g \doteq \{\Delta\mathtt{R}_z, \Delta\mathbf{t}\}$. The resulting parameter can be written as $\delta g(\boldsymbol{\theta}_C) \approx \boldsymbol{\theta}_C + D(\boldsymbol{\theta}_C)$, where the generators of the infinitesimal gauge [14] $D(\boldsymbol{\theta}_C) \doteq \mathtt{U}_{\boldsymbol{\theta}_C}(\Delta\alpha, \Delta\mathbf{t}^\top)^\top$ are linearly-related with $(\Delta\alpha, \Delta\mathbf{t}^\top)^\top$, the local coordinates describing δg. The rows of $\mathtt{U}_{\boldsymbol{\theta}_C}$ are

$$\begin{aligned}
&\mathtt{U}_{\mathbf{p}_i^C} = \left[\mathbf{e}_z \times \mathbf{p}_i^C, \mathtt{I}\right] && \mathtt{U}_{\mathbf{v}_i^C} = \left[\mathbf{e}_z \times \mathbf{v}_i^C, \mathtt{0}\right] \\
&\mathtt{U}_{\Delta\boldsymbol{\phi}_0^C} = \left[\mathtt{J}_l^{-1}(\Delta\boldsymbol{\phi}_0^C)\mathbf{e}_z, \mathtt{0}\right] && \mathtt{U}_{\delta\boldsymbol{\phi}_i^C} = \left[\mathbf{e}_z, \mathtt{0}\right],\ i \neq 0 \\
&\mathtt{U}_{\mathbf{X}_j^C} = \left[\mathbf{e}_z \times \mathbf{X}_j^C, \mathtt{I}\right],
\end{aligned} \tag{15}$$

where $\mathtt{J}_l$ is the left Jacobian of $SO(3)$ [22, p. 40].

Matrix $\mathtt{V}_{\boldsymbol{\theta}_C}$ is given by the derivative of the constraints (7), $\mathtt{V}_{\boldsymbol{\theta}_C}^\top \doteq \frac{\partial \mathbf{c}}{\partial \boldsymbol{\theta}}(\boldsymbol{\theta}_C)$. In case of the gauge fixation (10), only two derivatives are non-vanishing: $\partial(\mathbf{p}_0 - \mathbf{p}_0^0)/\partial \mathbf{p}_0 = \mathtt{I}$ and $\partial(\mathbf{e}_z^\top \Delta\boldsymbol{\phi}_0)/\partial \Delta\boldsymbol{\phi}_0 = \mathbf{e}_z^\top$.

References

[1] P. Corke, J. Lobo, and J. Dias, "An introduction to inertial and visual sensing," *Int. J. Robot. Research*, vol. 26, no. 6, pp. 519–535, 2007.

[2] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint Kalman filter for vision-aided inertial navigation," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, Apr. 2007, pp. 3565–3572.

[3] E. Jones and S. Soatto, "Visual-inertial navigation, mapping and localization: A scalable real-time causal approach," *Int. J. Robot. Research*, vol. 30, no. 4, Apr 2011.

[4] M. Li and A. I. Mourikis, "High-precision, consistent EKF-based visual-inertial odometry," *Int. J. Robot. Research*, vol. 32, no. 6, pp. 690–711, 2013.

[5] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "On-manifold preintegration for real-time visual-inertial odometry," *IEEE Trans. Robot.*, vol. PP, no. 99, pp. 1–21, Feb 2016.

[6] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. Leonard, and F. Dellaert, "iSAM2: Incremental smoothing and mapping using the Bayes tree," *Int. J. Robot. Research*, vol. 31, pp. 217–236, Feb. 2012.

[7] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, "Keyframe-based visual-inertial SLAM using nonlinear optimization," *Int. J. Robot. Research*, 2015.

[8] Z. Yang and S. Shen, "Monocular visual-inertial state estimation with online initialization and camera-IMU extrinsic calibration," *IEEE Trans. Autom. Sci. Eng.*, vol. 14, no. 1, pp. 39–51, Jan 2017.

[9] H. Rebecq, T. Horstschaefer, and D. Scaramuzza, "Real-time visual-inertial odometry for event cameras using keyframe-based nonlinear optimization," in *British Machine Vis. Conf. (BMVC)*, Sept. 2017.

[10] J. Kelly and G. S. Sukhatme, "Visual-inertial sensor fusion: Localization, mapping and sensor-to-sensor self-calibration," *Int. J. Robot. Research*, vol. 30, no. 1, pp. 56–79, 2011.

[11] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon, "Bundle adjustment – a modern synthesis," in *Vision Algorithms: Theory and Practice*, ser. LNCS, W. Triggs, A. Zisserman, and R. Szeliski, Eds., vol. 1883. Springer Verlag, 2000, pp. 298–372.

[12] S. Leutenegger, P. Furgale, V. Rabaud, M. Chli, K. Konolige, and R. Siegwart, "Keyframe-based visual-inertial SLAM using nonlinear optimization," in *Robotics: Science and Systems (RSS)*, 2013.

[13] S. Shen, N. Michael, and V. Kumar, "Tightly-coupled monocular visual-inertial fusion for autonomous flight of rotorcraft MAVs," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2015, pp. 5303–5310.

[14] K. Kanatani and D. D. Morris, "Gauges and gauge transformations for uncertainty description of geometric structure with indeterminacy," *IEEE Trans. Inf. Theory*, vol. 47, no. 5, pp. 2017–2028, Jul 2001.

[15] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. Achtelik, and R. Siegwart, "The EuRoC MAV datasets," *Int. J. Robot. Research*, 2015.

[16] A. J. Davison and R. M. Murray, "Simultaneous localization and map-building using active vision," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 24, no. 7, 2002.

[17] A. Patron-Perez, S. Lovegrove, and G. Sibley, "A spline-based trajectory representation for sensor fusion and rolling shutter cameras," *Int. J. Comput. Vis.*, vol. 113, no. 3, pp. 208–219, 2015.

[18] P. Furgale, J. Rehder, and R. Siegwart, "Unified temporal and spatial calibration for multi-sensor systems," in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, 2013.

[19] A. Agarwal, K. Mierle, and Others, "Ceres solver," http://ceres-solver.org.

[20] H. Carrillo, I. Reid, and J. A. Castellanos, "On the comparison of uncertainty criteria for active slam," in *2012 IEEE International Conference on Robotics and Automation*, May 2012, pp. 2080–2087.

[21] C. Forster, Z. Zhang, M. Gassner, M. Werlberger, and D. Scaramuzza, "SVO: Semidirect visual odometry for monocular and multicamera systems," *IEEE Trans. Robot.*, vol. PP, no. 99, pp. 1–17, 2017.

[22] G. S. Chirikjian, *Stochastic Models, Information Theory, and Lie Groups, Volume 2: Analytic Methods and Modern Applications (Applied and Numerical Harmonic Analysis)*. Birkhauser, 2012.