

A White-Noise-On-Jerk Motion Prior for Continuous-Time Trajectory Estimation on $SE(3)$

Tim Y. Tang, David J. Yoon, and Timothy D. Barfoot

Abstract—Simultaneous trajectory estimation and mapping (STEAM) offers an efficient approach to continuous-time trajectory estimation, by representing the trajectory as a Gaussian process (GP). Previous formulations of the STEAM framework use a GP prior that assumes white-noise-on-acceleration, with the prior mean encouraging constant body-centric velocity. We show that such a prior cannot sufficiently represent trajectory sections with non-zero acceleration, resulting in a bias to the posterior estimates.

This paper derives a novel motion prior that assumes white-noise-on-jerk, where the prior mean encourages constant body-centric acceleration. With the new prior, we formulate a variation of STEAM that estimates the pose, body-centric velocity, and body-centric acceleration. By evaluating across several datasets, we show that the new prior greatly outperforms the white-noise-on-acceleration prior in terms of solution accuracy.

I. INTRODUCTION

State estimation techniques for mobile robotics have been predominantly formulated in discrete time. While discrete-time techniques are sufficient for many applications, they are not ideal for high-rate sensors that take measurements continuously along a trajectory (e.g., scanning-while-moving lidars), or a combination of asynchronous sensors. Continuous-time estimation techniques are much more suitable in these cases, since measurements can be incorporated at any time along the trajectory, without needing to include an additional state at every measurement time. Moreover, continuous-time techniques have the advantage that the posterior estimates can be queried at any time along the trajectory, not just at measurement times.

Continuous-time estimation techniques can be categorized into two types: parametric and nonparametric. Parametric approaches typically represent the trajectory using a finite set of temporal basis functions. Our work focuses on the nonparametric approach in which the trajectory is represented as a Gaussian process, with time as the only input variable. While model fidelity in parametric approaches are affected by choices regarding trajectory representation and discretization, the GP approach relies heavily on the continuous-time prior distribution for solution quality.

Current formulations of the GP approach to continuous-time trajectory estimation employ a white-noise-on-acceleration (WNOA) prior, or one that assumes the prior mean is constant-velocity. While this choice of prior is appropriate for certain types of motion, we argue that it

The authors are affiliated with the Autonomous Space Robotics Lab at University of Toronto, Toronto, Ontario, Canada. {tim.tang, david.yoon}@robotics.utoronto.ca, tim.barfoot@utoronto.ca

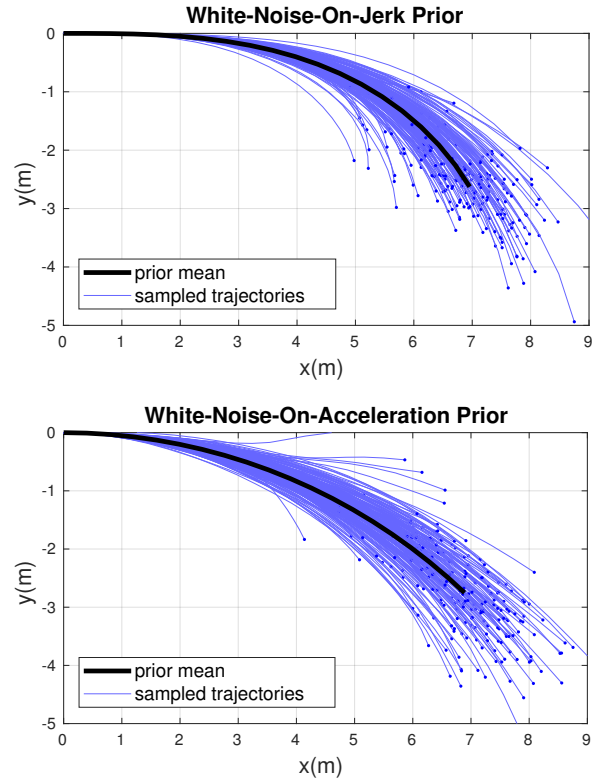


Fig. 1: Existing formulations of STEAM use a white-noise-on-acceleration motion prior (bottom), which have trouble representing trajectories with non-zero acceleration, such as in the motion of a vehicle in urban driving. We propose a white-noise-on-jerk motion prior (top), which is more suitable for representing these types of trajectories.

is insufficient for representing trajectories with non-zero acceleration, such as in the motion of a vehicle in urban driving. We show that a bias can occur when the motion prior does not sufficiently represent the underlying trajectory.

With this in mind, we derive a white-noise-on-jerk (WNOJ) motion prior, which assumes the prior mean is constant-acceleration. Our derivation starts with the same form of physically motivated stochastic differential equation (SDE) for describing motion as in the WNOA prior.

By evaluating on several real-world lidar datasets, we show that our variation of STEAM with the WNOJ prior greatly outperforms the current formulation of STEAM, which employs a WNOA prior. In particular, the use of WNOJ prior results in reduced bias and improved odometry accuracy to the estimated trajectory. We perform the experimental evaluation using lidar-only motion estimation, as this is a problem particularly suitable for continuous-time methods. The contribution of this paper, however, can be applied to

any choice of sensor suite.

In Section II we review previous work. An overview of our existing continuous-time lidar-only estimator is provided in Section III. In Section IV we identify a source of estimator bias that relates to the choice of GP prior. Section V presents the derivation of a white-noise-on-jerk motion prior, which is compared against the white-noise-on-acceleration prior experimentally in Section VI. In Section VII we give concluding remarks and discuss future work.

II. RELATED WORK

Early works on continuous-time estimation are mostly parametric approaches, which represent the trajectory using temporal basis functions. Jung and Taylor [1] first presented an estimator where the sensor trajectory is modelled by spline functions. Furgale et al. [2] derived the simultaneous localization and mapping (SLAM) problem in continuous time, and showed a small number of basis functions can sufficiently represent the state. Anderson and Barfoot [3] derived a relative coordinate formulation by estimating the body-centric velocity. Lovegrove et al. [4] applied continuous-time estimation in visual-inertial SLAM. Recent work by Dubé et al. [5] explored strategies for selecting knot sampling in parametric approaches to continuous-time trajectories.

Batch nonparametric approaches, which represent the trajectory as a Gaussian process, were first formulated by Tong et al. [6]. The smoothness assumption is handled in a principled manner through the underlying GP prior. Barfoot et al. [7] extended the GP approach to STEAM, which employs a WNOA motion prior, by jointly estimating the pose and velocity. This choice of GP motion prior results in the inverse kernel matrix being exactly sparse, leading to a very efficient formulation. Anderson and Barfoot [8] extended [7] to matrix Lie groups. Boots et al. [9] reformulated STEAM from batch estimation to an incremental algorithm. STEAM has been applied to motion planning [10], crop monitoring [11], and visual teach and repeat [12].

Without explicitly treating the trajectory as a continuous function of time, some estimators use interpolation between discrete poses to compensate for motion distortion, particularly in the case of scanning lidars. Bosse and Zlot [13], [14] used cubic splines to enforce smoothness for a trajectory estimated using data from a 2D spinning lidar, and linearly interpolated between sampled poses. Dong et al. [15] performed visual odometry from lidar intensity images, and interpolated on rotation and translation using a scheme detailed in [16]. State-of-the-art lidar-only motion estimation algorithm, LOAM [17], interpolates on $SE(3)$ between adjacent discrete poses, using a scheme similar to [13]. Unlike continuous-time methods, these methods need to make ad-hoc assumptions about trajectory smoothness in order to carry out interpolation.

While various motion estimation methods have made assumptions with the trajectory being constant-velocity [18], [7], [8], [19], [20], a constant-acceleration trajectory assumption has been used for tracking control [21], manipulator motion planning [22], and manipulator state estimation [23].

To the best of our knowledge, the derivation we present in this paper is the first attempt at modelling the trajectory as constant-acceleration mean (white-noise-on-jerk) in the context of continuous-time trajectory estimation on $SE(3)$.

III. CONTINUOUS-TIME ESTIMATOR

In this section, we give details on our existing continuous-time lidar odometry algorithm, which uses the STEAM framework with a WNOA motion prior [8]. This serves as the baseline against which the WNOJ prior will be evaluated.

A. WNOA GP Prior

Our goal is to employ a class of GP priors that leads to an efficient formulation and a simple solution [7] [16]. This class of GP priors is based on linear time-invariant (LTI) stochastic differential equations (SDEs) of the form

$$\begin{aligned}\dot{\gamma}(t) &= \mathbf{A}\gamma(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{L}\mathbf{w}(t), \\ \mathbf{w}(t) &\sim \mathcal{GP}(\mathbf{0}, \mathbf{Q}_c\delta(t-t')), \end{aligned} \quad (1)$$

where $\gamma(t)$ is the state at timestep t , $\mathbf{u}(t)$ is an exogenous input, and $\mathbf{w}(t)$ is a zero-mean, white-noise GP with *power spectral density matrix*, $\mathbf{Q}_c \in \mathbb{R}^{6 \times 6}$ [16]. If $\mathbf{u}(t) = \mathbf{0}$, then for the mean function we have the simple solution

$$\tilde{\gamma}(\tau) = \Phi(\tau, t_k)\tilde{\gamma}(t_k), \quad (2)$$

where $\tilde{\gamma}$ is the prior mean, and $\Phi(\tau, t_k)$ is the *state transition function* from timestep t_k to timestep τ .

B. GP Prior for $SE(3)$

In $SE(3)$, a physically-motivated GP prior is the following SDE:

$$\begin{aligned}\dot{\mathbf{T}}(t) &= \boldsymbol{\varpi}(t)^\wedge \mathbf{T}(t), \\ \boldsymbol{\varpi}(t) &= \mathbf{w}'(t), \quad \mathbf{w}(t) \sim \mathcal{GP}(\mathbf{0}, \mathbf{Q}_c\delta(t-t')), \end{aligned} \quad (3)$$

where $\mathbf{T}(t) = \exp(\boldsymbol{\xi}(t)^\wedge) \in SE(3)$ is the pose with $\boldsymbol{\xi}(t) = [\boldsymbol{\rho}(t)^T \quad \boldsymbol{\phi}(t)^T]^T \in \mathbb{R}^6$, where $\boldsymbol{\xi}$ is the vector-space representation of the pose. $\boldsymbol{\rho}(t) = [\rho_1(t) \quad \rho_2(t) \quad \rho_3(t)]^T$, $\boldsymbol{\phi}(t) = [\phi_1(t) \quad \phi_2(t) \quad \phi_3(t)]^T$, where $\boldsymbol{\rho}$ and $\boldsymbol{\phi}$ are the translational and rotational components of $\boldsymbol{\xi}$, respectively. $\boldsymbol{\varpi}(t) = [\boldsymbol{\nu}(t)^T \quad \boldsymbol{\omega}(t)^T]^T \in \mathbb{R}^6$ is the body-centric velocity. $(\cdot)^\wedge$ converts $\boldsymbol{\xi}(t) \in \mathbb{R}^6$ into a member of *Lie algebra*, $\mathfrak{se}(3)$ [24] [16]. The state is

$$\mathbf{x}(t) = \{\mathbf{T}(t), \boldsymbol{\varpi}(t)\}. \quad (4)$$

However, it can be seen that the SDE in (3) is nonlinear, and therefore cannot be cast into the form of (1) and solved efficiently [8]. Instead, [8] defines a *local* pose variable:

$$\boldsymbol{\xi}_i(t) := \ln(\mathbf{T}(t)\mathbf{T}_i^{-1})^\vee, \quad t_i \leq t \leq t_{i+1}, \quad (5)$$

which is a function of the *global* pose variables, $\mathbf{T}(t)$ and \mathbf{T}_i , where $\mathbf{T}_i = \mathbf{T}(t_i)$. For simplicity we use \mathbf{T}_i to denote $\mathbf{T}_{i,0}$. Here $(\cdot)^\vee$ is the inverse of $(\cdot)^\wedge$, and converts a member of $\mathfrak{se}(3)$ to $\boldsymbol{\xi} \in \mathbb{R}^6$ [24] [16].

Using *local* variables, [8] defines a sequence of *local* priors that can be cast into a LTI SDE of the form in (1), with

$$\gamma_i(t) := \begin{bmatrix} \boldsymbol{\xi}_i(t) \\ \dot{\boldsymbol{\xi}}_i(t) \end{bmatrix}, \quad \mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \quad \mathbf{L} = \begin{bmatrix} \mathbf{0} \\ \mathbf{1} \end{bmatrix}, \quad (6)$$

where $\gamma_i(t)$ is defined as the *local* state. Under this formulation, we have white-noise on the second derivative of $\xi_i(t)$, i.e. $\ddot{\xi}_i(t) = \mathbf{w}(t)$. Furthermore, we have the following [8]:

$$\dot{\xi}_i(t) = \mathcal{J}(\xi_i(t))^{-1} \varpi(t), \quad (7)$$

where $\mathcal{J}(\xi) \in \mathbb{R}^{6 \times 6}$ is the *left Jacobian* of $SE(3)$ [24] [16].

C. Cost Terms in Optimization

For our estimator, the negative-log-likelihood objective function consists of prior and measurement cost terms:

$$J = \underbrace{\sum_i J_i}_{\text{prior}} + \underbrace{\sum_j J_j}_{\text{measurement}}. \quad (8)$$

Since our estimator is only for odometry, we do not keep landmarks as part of the state, as in the full STEAM problem. The optimization problem is then

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} J(\mathbf{x}), \quad (9)$$

where the state \mathbf{x} consists of all trajectory poses and velocities, as defined in (4). We solve the optimization problem using Gauss-Newton, where each \mathbf{T} and ϖ are updated using an $SE(3)$ perturbation scheme [24] [8]:

$$\mathbf{T}_{\text{op},i} \leftarrow \exp(\delta \xi_i^\wedge) \mathbf{T}_{\text{op},i}, \quad \varpi_{\text{op},i} \leftarrow \varpi_{\text{op},i} + \delta \varpi_i, \quad (10)$$

where $(\cdot)_{\text{op}}$ is the operating point. Each prior cost term is

$$J_i = \frac{1}{2} \mathbf{e}_i^T \mathbf{Q}_i^{-1} \mathbf{e}_i. \quad (11)$$

In terms of *local* pose variables, each prior error term is

$$\mathbf{e}_i = \gamma_i(t_{i+1}) - \hat{\gamma}_i(t_{i+1}) - \Phi(t_{i+1}, t_i)(\gamma_i(t_i) - \hat{\gamma}_i(t_i)), \quad (12)$$

where the *local* state variables are defined as [8]

$$\gamma_i(t_i) = \begin{bmatrix} \mathbf{0} \\ \varpi_i \end{bmatrix}, \quad \gamma_i(t_{i+1}) = \begin{bmatrix} \ln(\mathbf{T}_{i+1,i})^\vee \\ \mathcal{J}_{i+1,i}^{-1} \varpi_{i+1} \end{bmatrix}, \quad (13)$$

with $\mathbf{T}_{i+1,i} := \mathbf{T}_{i+1} \mathbf{T}_i^{-1}$, $\mathcal{J}_{i+1,i} := \mathcal{J}(\ln(\mathbf{T}_{i+1} \mathbf{T}_i^{-1}))^\vee$. The state transition function can be computed as in [16],

$$\Phi(t, t_i) = \exp(\mathbf{A} \Delta t_i) = \begin{bmatrix} \mathbf{1} & \Delta t_i \mathbf{1} \\ \mathbf{0} & \mathbf{1} \end{bmatrix}, \quad (14)$$

and the inverse covariance matrix is [24] [8]

$$\mathbf{Q}_i(t)^{-1} = \begin{bmatrix} 12\Delta t_i^{-3} \mathbf{Q}_c^{-1} & -6\Delta t_i^{-2} \mathbf{Q}_c^{-1} \\ -6\Delta t_i^{-2} \mathbf{Q}_c^{-1} & 4\Delta t_i^{-1} \mathbf{Q}_c^{-1} \end{bmatrix}, \quad (15)$$

where $\Delta t_i = t - t_i$. Using the relationship between *local* and *global* state variables, we can re-write the prior error term in terms of *global* state variables as [8]

$$\mathbf{e}_i = \begin{bmatrix} \ln(\mathbf{T}_{i+1} \mathbf{T}_i^{-1})^\vee - (t_{i+1} - t_i) \varpi_i \\ \mathcal{J}_{i+1,i}^{-1} \varpi_{i+1} - \varpi_i \end{bmatrix}. \quad (16)$$

Each measurement cost term is

$$J_j = \frac{1}{2} \frac{u_j^2}{1 + u_j^2}, \quad (17)$$

where we have chosen the Geman-McClure robust cost [16]. Each u_j is a whitened error norm. Given a point \mathbf{p} measured

at time $t = \tau$, and let \mathbf{q} be its matched point, expressed in the reference frame \mathcal{F}_0 . Define a measurement error term:

$$\mathbf{g}_j = \mathbf{D}(\mathbf{p} - \mathbf{T}_\tau \mathbf{q}), \quad (18)$$

where $\mathbf{D} \in \mathbb{R}^{3 \times 4}$ is a projection matrix. If \mathbf{p} lies on a plane, then we formulate a point-to-plane whitened error norm:

$$u_j^{\text{plane}} = \sqrt{\mathbf{g}_j^T (\beta \mathbf{n} \mathbf{n}^T) \mathbf{g}_j}, \quad (19)$$

where \mathbf{n} is the surface normal of \mathbf{p} , and β is a scale factor. If \mathbf{p} does not lie on a plane, we formulate a point-to-point whitened error norm:

$$u_j^{\text{point}} = \sqrt{\mathbf{g}_j^T \mathbf{R}_j^{-1} \mathbf{g}_j}, \quad (20)$$

where \mathbf{R}_j is the associated measurement covariance.

Our lidar odometry algorithm utilizes sliding-window optimization, and runs in an iterative fashion where matched pairs of points are found in each iteration. Please refer to our previous work [25] for further details on our odometry pipeline, such as point matching and keypoint selection.

D. Querying the Trajectory

Our formulation allows us to incorporate measurements at any time along the trajectory, not just at timesteps kept in the state vector as for discrete-time methods. Suppose we have a measurement at $t = \tau$ as in (18), and that $t_i < \tau < t_{i+1}$, where t_i and t_{i+1} are knot times in the state. We can interpolate for the state at τ using results from [8]:

$$\gamma_i(\tau) = \Lambda(\tau) \gamma_i(t_i) + \Omega(\tau) \gamma_i(t_{i+1}), \quad (21)$$

where $\Lambda(\tau) \in \mathbb{R}^{12 \times 12}$ and $\Omega(\tau) \in \mathbb{R}^{12 \times 12}$ are [7]

$$\begin{aligned} \Lambda(\tau) &= \Phi(\tau, t_i) - \Omega(\tau) \Phi(t_{i+1}, t_i), \\ \Omega(\tau) &= \mathbf{Q}_i(\tau) \Phi(t_{i+1}, t_i)^T \mathbf{Q}_i(\tau)^{-1}. \end{aligned} \quad (22)$$

Again, using our knowledge on the relationship between *local* and *global* state variables as in (13), we can re-formulate (21) using *global* state variables. While interpolating for the body-centric velocity at an arbitrary time might be of interest to certain applications, for lidar-only odometry we are mainly interested in pose interpolation:

$$\mathbf{T}_\tau = \exp((\Lambda_{12}(\tau) \varpi_i + \Omega_{11}(\tau) \ln(\mathbf{T}_{i+1,i})^\vee + \Omega_{12}(\tau) \mathcal{J}_{i+1,i}^{-1} \varpi_{i+1})^\wedge) \mathbf{T}_i, \quad (23)$$

where Λ_{mn} and Ω_{mn} are $\mathbb{R}^{6 \times 6}$ sub-blocks of Λ and Ω . This is a principled approach for querying the trajectory that comes directly from standard GP interpolation [26]. It can be seen that, given a measurement at $t = \tau$ with $t_i < \tau < t_{i+1}$, the result in (23) allows updates to temporally adjacent state variables at t_i and t_{i+1} in the optimization process.

IV. ESTIMATOR BIAS

As shown in Section VI (Figures 4, 5, 7), there are noticeable biases in our baseline estimator, particularly in the directions of z (ρ_3), roll (ϕ_1), and pitch (ϕ_2). See Figure 2 for the coordinate system for our estimator.

There could be many sources that might cause the estimated trajectory to be biased, such as poor sensor calibration,



Fig. 2: The coordinate system for our estimator. Roll (ϕ_1), pitch (ϕ_2), and yaw (ϕ_3) are rotations about the $x(\rho_1)$, $y(\rho_2)$, and $z(\rho_3)$ axes, respectively.

or choosing a measurement covariance that does not reflect the sensor noise characteristics. In this paper, we focus on a very specific source of estimator bias, which results from the motion prior being insufficient to represent the underlying continuous-time trajectory.

Consider a very simple estimation problem in which a robot, initially stationary at time t_0 , travels from \mathcal{F}_0 at t_0 to \mathcal{F}_1 at t_1 under constant acceleration. The robot only travels forward, therefore motion only occurs in the ρ_1 direction. The robot measures a single point \mathbf{p} at t_1 , which is matched against \mathbf{q} measured at t_0 . Keeping \mathbf{T}_0 and ϖ_0 fixed, the state we wish to estimate is

$$\mathbf{x} = \{\mathbf{T}_1, \varpi_1\}. \quad (24)$$

We can define the following ground truth quantities:

$$\begin{aligned} \xi_{\text{gt},1} &= [\tfrac{1}{2}a(t_1 - t_0)^2 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T, \\ \mathbf{T}_{\text{gt},1} &= \exp(\xi_{\text{gt},1}^\wedge), \\ \varpi_{\text{gt},1} &= [a(t_1 - t_0) \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T, \end{aligned} \quad (25)$$

where a is the acceleration in ρ_1 . We can define the following measurement error equation and the associated measurement Jacobian [24]:

$$\mathbf{g} = \mathbf{D}(\mathbf{p} - \mathbf{T}_{\text{op},1}\mathbf{q}), \quad \mathbf{G} = -\mathbf{D}(\mathbf{T}_{\text{op},1}\mathbf{q})^\odot, \quad (26)$$

where the $(\cdot)^\odot$ operator is defined in [24]. We also have a WNOA prior error term \mathbf{e} that can be constructed from (16). For simplicity, we assume $t_1 - t_0 = 1$, $\mathbf{Q} = \mathbf{R} = \mathbf{1}$, and that the measurement is noise-free. Initializing the state variables at ground truth, we have $\mathbf{g} = \mathbf{0}$ as the measurement is noise-free. However, \mathbf{e} is not zero since the motion is not constant-velocity. Performing Gauss-Newton for one iteration using the perturbation scheme in [8] (Equation (10)), we have:

$$\delta\xi_1 = \frac{1}{m} \begin{bmatrix} -\frac{1}{4}a((a^2 - 4x)^2 + 32(y^2 + z^2 + 1)) \\ ay(a + 4x) \\ az(a + 4x) \\ 0 \\ 8az \\ -8ay \end{bmatrix}, \quad (27)$$

where $m = (a^2 - 4x)^2 + 16(y^2 + z^2 + 2)$, and $\mathbf{T}_{\text{op},1}\mathbf{q} = [x \ y \ z \ 1]^T$. Equation (27) shows that our simple problem results in perturbations to degrees of freedom where

there is no motion (ρ_2, ρ_3, ϕ_2 , and ϕ_3), effectively creating a bias. Moreover, the perturbations to these DOFs depend on $[x \ y \ z]^T$, the Cartesian coordinate of the transformed point, $\mathbf{T}_{\text{op},1}\mathbf{q}$.

We can draw the observation that when the motion prior cannot sufficiently describe the underlying trajectory, such as when the prior mean is constant-velocity but the trajectory is constant-acceleration, then the estimator will be biased in certain degrees of freedom. Particularly, the induced bias is a function of the Cartesian coordinates of points. The bias stems from the optimizer's desire to keep the cost low; the prior cost is made smaller by increasing the measurement cost in the overall objective function (8). It can be shown easily that this source of bias will not occur when we use an estimator with a WNOJ prior, as derived in Section V.

Equation (27) is computed assuming the robot has forward acceleration. However, a similar case can be made for motion with angular acceleration, such as when initiating a turn.

V. WHITE-NOISE-ON-JERK MOTION PRIOR

Here we derive a white-noise-on-jerk motion prior. Instead of modelling the acceleration as a zero-mean, white-noise Gaussian process as in the case of a WNOA prior [8], we now explicitly estimate the following state:

$$\mathbf{x}(t) = \{\mathbf{T}(t), \varpi(t), \dot{\varpi}(t)\}. \quad (28)$$

where $\dot{\varpi}(t) \in \mathbb{R}^6$ is the body-centric acceleration.

Extending the idea of *local* pose variables as presented in Section III-B, we can define a sequence of *local* white-noise-on-jerk priors as a LTI SDE in the form of (1):

$$\gamma_i(t) := \begin{bmatrix} \xi_i(t) \\ \dot{\xi}_i(t) \\ \ddot{\xi}_i(t) \end{bmatrix}, \quad \mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}, \quad \mathbf{L} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{1} \end{bmatrix}. \quad (29)$$

We now have white-noise on the third derivative (jerk) of $\xi_i(t)$, $\ddot{\xi}_i(t) = \mathbf{w}(t)$, where $\mathbf{w}(t) \sim \mathcal{GP}(\mathbf{0}, \mathbf{Q}_c \delta(t - t'))$. For the WNOJ prior, the state transition function is now

$$\Phi(t, t_i) = \exp(\mathbf{A}\Delta t_i) = \begin{bmatrix} \mathbf{1} & \Delta t_i \mathbf{1} & \frac{1}{2} \Delta t_i^2 \mathbf{1} \\ \mathbf{0} & \mathbf{1} & \Delta t_i \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} \end{bmatrix}, \quad (30)$$

and the covariance matrix can be computed as

$$\begin{aligned} \mathbf{Q}_i(t) &= \int_0^{\Delta t_i} \exp(\mathbf{A}(\Delta t_i - s)) \mathbf{L} \mathbf{Q}_c \mathbf{L}^T \exp(\mathbf{A}(\Delta t_i - s))^T ds \\ &= \begin{bmatrix} \frac{1}{20} \Delta t_i^5 \mathbf{Q}_c & \frac{1}{8} \Delta t_i^4 \mathbf{Q}_c & \frac{1}{6} \Delta t_i^3 \mathbf{Q}_c \\ \frac{1}{8} \Delta t_i^4 \mathbf{Q}_c & \frac{1}{3} \Delta t_i^3 \mathbf{Q}_c & \frac{1}{2} \Delta t_i^2 \mathbf{Q}_c \\ \frac{1}{6} \Delta t_i^3 \mathbf{Q}_c & \frac{1}{2} \Delta t_i^2 \mathbf{Q}_c & \Delta t_i \mathbf{Q}_c \end{bmatrix}. \end{aligned} \quad (31)$$

The inverse covariance matrix is then

$$\begin{aligned} \mathbf{Q}_i(t)^{-1} &= \\ &= \begin{bmatrix} 720 \Delta t_i^{-5} \mathbf{Q}_c^{-1} & -360 \Delta t_i^{-4} \mathbf{Q}_c^{-1} & 60 \Delta t_i^{-3} \mathbf{Q}_c^{-1} \\ -360 \Delta t_i^{-4} \mathbf{Q}_c^{-1} & 192 \Delta t_i^{-3} \mathbf{Q}_c^{-1} & -36 \Delta t_i^{-2} \mathbf{Q}_c^{-1} \\ 60 \Delta t_i^{-3} \mathbf{Q}_c^{-1} & -36 \Delta t_i^{-2} \mathbf{Q}_c^{-1} & 9 \Delta t_i^{-1} \mathbf{Q}_c^{-1} \end{bmatrix}. \end{aligned} \quad (32)$$

Figure 1 shows trajectories sampled from a white-noise-on-jerk prior distribution where the prior mean is constant-acceleration, compared with trajectories sampled from a white-noise-on-acceleration prior distribution where the prior mean is constant-velocity. We argue that the WNOJ prior is more suitable for representing motion with non-zero acceleration trajectory sections, such as in urban driving.

A. Prior Error Term

In *local* pose variables, the prior error term is the same as in (12). We wish to then express the prior error in terms of \mathbf{T} , $\boldsymbol{\varpi}$, and $\dot{\boldsymbol{\varpi}}$. The relationship between $\xi_i(t)$ and $\dot{\xi}_i(t)$ and *global* state variables are shown in Equations (5) and (7). To express $\dot{\xi}_i(t)$ in terms of *global* state variables, we have

$$\begin{aligned}\ddot{\xi}_i(t) &= \frac{d}{dt}(\dot{\xi}_i(t)) = \frac{d}{dt}(\mathcal{J}(\xi_i(t))^{-1}\boldsymbol{\varpi}(t)) \\ &= \frac{d}{dt}(\mathcal{J}(\xi_i(t))^{-1})\boldsymbol{\varpi}(t) + \mathcal{J}(\xi_i(t))^{-1}\dot{\boldsymbol{\varpi}}(t).\end{aligned}\quad (33)$$

We can write the inverse left Jacobian of $SE(3)$ as a power-series expansion [16]:

$$\mathcal{J}(\xi)^{-1} = \sum_{n=0}^{\infty} \frac{B_n}{n!} (\xi^\wedge)^n = B_0 \mathbf{1} + \frac{B_1}{1!} \xi^\wedge + \frac{B_2}{2!} (\xi^\wedge)^2 + \dots, \quad (34)$$

where the coefficients, B_n , are the *Bernoulli numbers*. The operator $(\cdot)^\wedge$ is defined as [24] [16]

$$\xi^\wedge = \begin{bmatrix} \boldsymbol{\rho} \end{bmatrix}^\wedge = \begin{bmatrix} \phi^\wedge & \boldsymbol{\rho}^\wedge \\ \mathbf{0} & \phi^\wedge \end{bmatrix}. \quad (35)$$

It can be shown easily that $\frac{d}{dt}(\xi^\wedge) = \dot{\xi}^\wedge$, therefore

$$\begin{aligned}\frac{d}{dt}(\mathcal{J}^{-1}) &= \frac{B_1}{1!} \dot{\xi}^\wedge + \frac{B_2}{2!} (\dot{\xi}^\wedge \xi^\wedge + \xi^\wedge \dot{\xi}^\wedge) + \\ &\frac{B_3}{3!} (\dot{\xi}^\wedge (\xi^\wedge)^2 + \xi^\wedge \dot{\xi}^\wedge \xi^\wedge + (\xi^\wedge)^2 \dot{\xi}^\wedge) + \dots\end{aligned}\quad (36)$$

As it turns out, we cannot express $\frac{d}{dt}(\mathcal{J}^{-1})$ analytically in terms of \mathcal{J}^{-1} or \mathcal{J} , which are familiar terms with which to work. We instead resort to the first-order approximation [16] that

$$\mathcal{J}(\xi)^{-1} \approx \mathbf{1} - \frac{1}{2} \xi^\wedge. \quad (37)$$

Our approximation is reasonable as long as ξ is small, which it will be in our case. Under this approximation, we have

$$\frac{d}{dt}(\mathcal{J}(\xi)^{-1}) \approx -\frac{1}{2} \dot{\xi}^\wedge, \quad (38)$$

and finally

$$\begin{aligned}\ddot{\xi}_i(t) &= \frac{d}{dt}(\mathcal{J}(\xi_i(t))^{-1})\boldsymbol{\varpi}(t) + \mathcal{J}(\xi_i(t))^{-1}\dot{\boldsymbol{\varpi}}(t) \\ &\approx -\frac{1}{2} \dot{\xi}_i(t)^\wedge \boldsymbol{\varpi}(t) + \mathcal{J}(\xi_i(t))^{-1}\dot{\boldsymbol{\varpi}}(t) \\ &= -\frac{1}{2} (\mathcal{J}(\xi_i(t))^{-1}\boldsymbol{\varpi}(t))^\wedge \boldsymbol{\varpi}(t) + \mathcal{J}(\xi_i(t))^{-1}\dot{\boldsymbol{\varpi}}(t).\end{aligned}\quad (39)$$

The *local* state variables can then be written as

$$\begin{aligned}\gamma_i(t_i) &= \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\varpi}_i \\ \dot{\boldsymbol{\varpi}}_i \end{bmatrix}, \\ \gamma_i(t_{i+1}) &= \begin{bmatrix} \ln(\mathbf{T}_{i+1,i})^\vee \\ \mathcal{J}_{i+1,i}^{-1}\boldsymbol{\varpi}_{i+1} \\ -\frac{1}{2}(\mathcal{J}_{i+1,i}^{-1}\boldsymbol{\varpi}_{i+1})^\wedge \boldsymbol{\varpi}_{i+1} + \mathcal{J}_{i+1,i}^{-1}\dot{\boldsymbol{\varpi}}_{i+1} \end{bmatrix},\end{aligned}\quad (40)$$

where we have made use of the identity $\mathbf{x}^\wedge \mathbf{x} = \mathbf{0}$ [16].

In terms of *global* state variables, the prior error term is

$$\mathbf{e}_i = \begin{bmatrix} \ln(\mathbf{T}_{i+1,i})^\vee - (t_{i+1} - t_i)\boldsymbol{\varpi}_i - \frac{1}{2}(t_{i+1} - t_i)^2 \dot{\boldsymbol{\varpi}}_i \\ \mathcal{J}_{i+1,i}^{-1}\boldsymbol{\varpi}_{i+1} - \boldsymbol{\varpi}_i - (t_{i+1} - t_i)\dot{\boldsymbol{\varpi}}_i \\ -\frac{1}{2}(\mathcal{J}_{i+1,i}^{-1}\boldsymbol{\varpi}_{i+1})^\wedge \boldsymbol{\varpi}_{i+1} + \mathcal{J}_{i+1,i}^{-1}\dot{\boldsymbol{\varpi}}_{i+1} - \dot{\boldsymbol{\varpi}}_i \end{bmatrix}. \quad (41)$$

Suppose we assume the trajectory has zero acceleration (which is assumed by a prior mean that is constant-velocity), $\dot{\boldsymbol{\varpi}}_i = \dot{\boldsymbol{\varpi}}_{i+1} = \mathbf{0}$, and also make the assumption that $\mathcal{J}_{i+1,i}^{-1}\boldsymbol{\varpi}_{i+1} \approx \boldsymbol{\varpi}_{i+1}$. In this case, the last component in (41) becomes zero, and the first two components become identical to the WNOA prior as in (16); we have essentially recovered the prior error equation for the WNOA prior.

B. Querying the Trajectory

We start from the same interpolation equation using *local* state variables (21). For the WNOJ prior, the interpolation coefficients $\Lambda(\tau)$ and $\Omega(\tau)$ can be computed from (22), using $\Phi \in \mathbb{R}^{18 \times 18}$ and $\mathbf{Q}_i \in \mathbb{R}^{18 \times 18}$ from (30) and (31).

Substituting with *global* state variables for the WNOJ prior using (40), the pose interpolation equation is

$$\begin{aligned}\mathbf{T}_\tau &= \exp((\Lambda_{12}(\tau)\boldsymbol{\varpi}_i + \Lambda_{13}(\tau)\dot{\boldsymbol{\varpi}}_i + \Omega_{11}(\tau)\ln(\mathbf{T}_{i+1,i})^\vee \\ &+ \Omega_{12}(\tau)\mathcal{J}_{i+1,i}^{-1}\boldsymbol{\varpi}_{i+1} + \Omega_{13}(\tau)(-\frac{1}{2}(\mathcal{J}_{i+1,i}^{-1}\boldsymbol{\varpi}_{i+1})^\wedge \boldsymbol{\varpi}_{i+1} \\ &+ \mathcal{J}_{i+1,i}^{-1}\dot{\boldsymbol{\varpi}}_{i+1}))^\wedge) \mathbf{T}_i,\end{aligned}\quad (42)$$

where $t_i < \tau < t_{i+1}$. Λ_{mn} and Ω_{mn} are $\mathbb{R}^{6 \times 6}$ sub-blocks of $\Lambda(\tau) \in \mathbb{R}^{18 \times 18}$ and $\Omega(\tau) \in \mathbb{R}^{18 \times 18}$.

Again, if we assume that $\dot{\boldsymbol{\varpi}}_i = \dot{\boldsymbol{\varpi}}_{i+1} = \mathbf{0}$, and $\mathcal{J}_{i+1,i}^{-1}\boldsymbol{\varpi}_{i+1} \approx \boldsymbol{\varpi}_{i+1}$, the terms with coefficients Λ_{13} and Ω_{13} become zeros. Similar to the case with the prior error term, we can essentially recover the pose interpolation equation for the WNOA prior as in (23).

VI. EXPERIMENTAL VALIDATION

To evaluate the white-noise-on-jerk prior we derived, we formulated a variation of our continuous-time lidar odometry estimator that employs the WNOJ prior. The new estimator is evaluated on various Velodyne lidar datasets, and the odometry errors are compared against the baseline estimator presented in Section III, which employs a WNOA prior. To ensure a fair comparison, all other aspects of STEAM such as constructing measurement terms, and the other components of lidar odometry such as the point matching method, are kept the same. Any differences between the two estimators arise solely from their different motion priors. Our evaluations are odometric, and we make no attempt to use mapping or loop-closure to reduce the estimation error.

The power spectral density matrix, \mathbf{Q}_c , is the only hyperparameter for our lidar odometry algorithm. In practice, \mathbf{Q}_c is designed to be a diagonal matrix, where tuning is done by choosing the diagonal elements from a list of candidates based on our prior knowledge about the trajectory. While we do not make use of the nonholomic constraint in our estimator, we penalize acceleration or jerk in each DOF differently, by scaling each diagonal element of \mathbf{Q}_c relative to the others. For both types of motion priors, we tuned \mathbf{Q}_c to achieve the best performance on the training set (sequences 0 to 10) of the KITTI odometry benchmark [27]. \mathbf{Q}_c was then kept the same when evaluating on all other datasets.

A. KITTI Odometry Benchmark

Sequences 0 to 10 are the training sequences of KITTI. Sequences 11 to 21 are the test sequences, where the ground truths are not publicly available. The KITTI benchmark evaluates percentage translation errors across path segments of lengths 100, 200, ..., 800 meters, and an average over all path segments is computed¹. A total error averaged over path segments evaluated for all sequences is also reported.

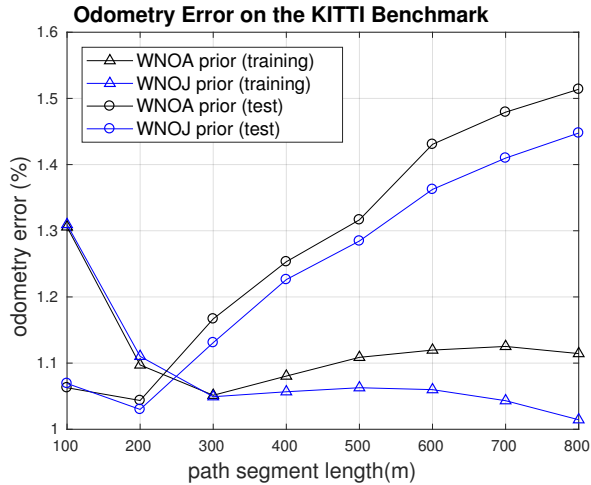


Fig. 3: Odometry error for the baseline estimator which employs the WNOA prior, and for the new estimator which employs the WNOJ prior.

The baseline estimator that employs a WNOA prior achieved an overall error of 1.13% on the training set, and 1.26% on the test set. Our new estimator that employs a WNOJ prior achieved an overall error of 1.10% on the training set, and 1.22% on the test set. A detailed break-down of the error for various path segment lengths is presented in Figure 3. The new estimator using WNOJ prior outperforms the baseline estimator for almost all path segment lengths. Figure 4 shows a sequence where the odometry biases are noticeably reduced when we use the WNOJ prior.

Our baseline estimator, submitted as *STEAM-L*, currently ranks at #11 on the KITTI leader board among lidar-only methods, while our new estimator with the WNOJ prior, *STEAM-L WNOJ*, ranks at #10. Both estimators are odometric and make no use of loop-closure. By choosing a motion

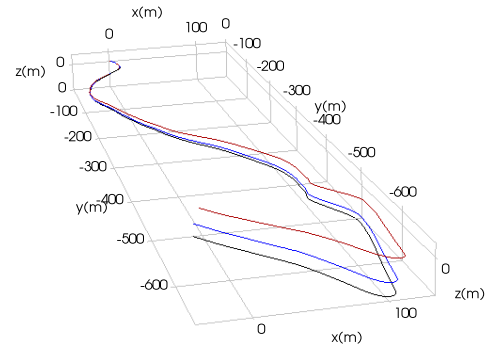


Fig. 4: 3D plots of odometry estimates for sequence 10: baseline estimator using WNOA prior (black) vs. new estimator using WNOJ prior (blue) when compared against ground truth (red).

prior that we believe is more representative of real-world vehicle trajectories, we achieved consistent improvements to our baseline method, which is already fairly accurate.

The lidar point-clouds from the KITTI odometry benchmark were post-processed by the dataset authors to compensate motion distortion. As a result, all points in a sensor revolution can be treated as being measured at exactly the same time, and we do not need to rely on the motion prior for interpolating the pose as in Equations (23) and (42). The prior cost terms (11), however, are still used to smooth the trajectory. Nevertheless, for undistorted data such as in the KITTI dataset, it is not necessary to use a continuous-time estimation framework. Even though the new estimator with the WNOJ prior outperformed our baseline estimator on the KITTI benchmark, we argue that datasets with motion-distorted point-clouds are more suitable for comparing continuous-time methods.

B. University of Toronto Dataset

A dataset was collected by our test vehicle (Figure 6) along different routes around University of Toronto (U of T). This resulted in 9 sequences of Velodyne data where each is at least 1.7km in distance. 6-DOF ground truth is available via an on-board Applanix positioning and orientation system (POS). For consistency, to evaluate for odometry errors we use the same method as the KITTI benchmark, where translational errors are evaluated across path segments of lengths 100, 200, ..., 800 meters. This is a motion-distorted lidar dataset, as we do not employ external sensors or ground truth to compensate the point-clouds. We rely solely on the continuous-time estimator for handling motion distortion.

The U of T dataset features driving in urban scenes where the vehicle's speed is generally under 50km/h. However, the vehicle needs to constantly slow down for traffic, or take a turn at an intersection. Since the vehicle's trajectory contains many sections where the velocity changes consistently, this dataset is much more suitable for motion estimation using a WNOJ prior, than a WNOA prior.

The results for the baseline estimator using WNOA prior and the new estimator using WNOJ are compared in Table I. The errors are higher than in the KITTI dataset, mostly

¹ The evaluation metric is provided in KITTI's development kit.

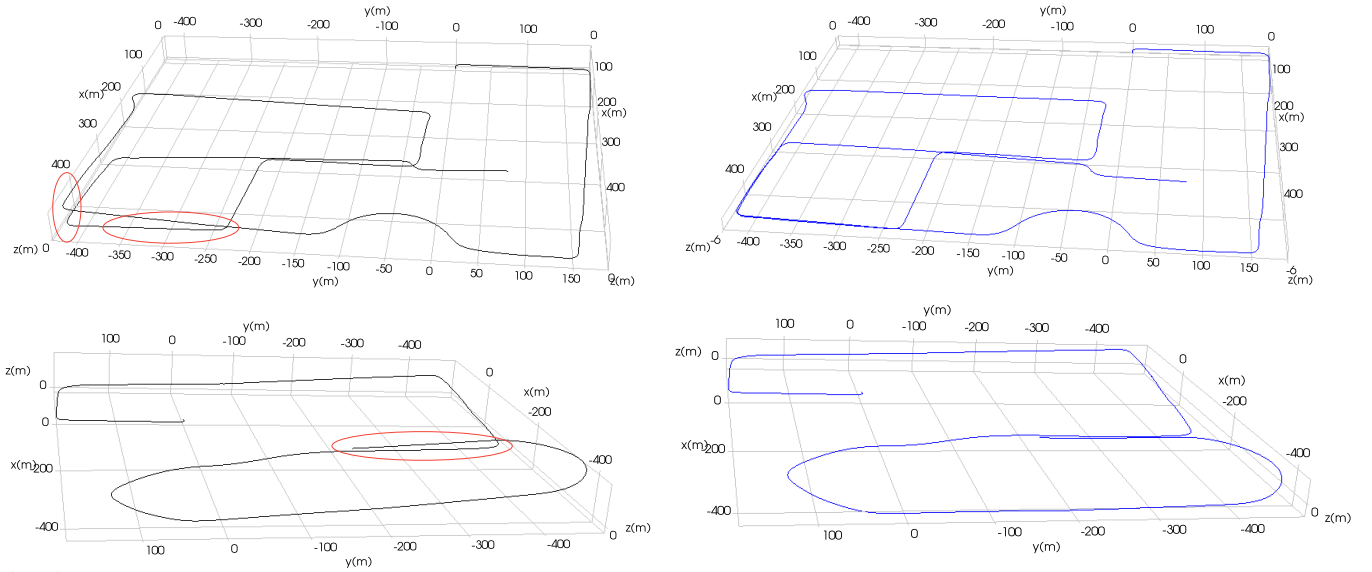


Fig. 5: 3D plots of odometry estimates for sequence 0 (top) and sequence 4 (bottom) of the University of Toronto dataset from the same perspective. Black is odometry using the baseline estimator with WNOA prior, and blue is the new estimator using WNOJ prior. Left: due to biases, odometry does not overlap when the vehicle travels back to a path it has been before (circled). Right: the biases are significantly reduced when using WNOJ prior.



Fig. 6: The Buick test vehicle used, equipped with a Velodyne HDL-64E lidar, and an Applanix POS-LV 210 system for ground truth.

TABLE I: Odometry errors for the baseline estimator using WNOA prior and new estimator using WNOJ prior, evaluated on the U of T dataset.

Sequence no.	Distance (km)	Baseline estimator with WNOA prior (%)	New estimator with WNOJ prior (%)
0	3.34	1.5326	1.2663
1	2.21	1.3706	1.2797
2	3.04	1.3967	1.3485
3	2.91	1.7980	1.5844
4	2.99	1.6100	1.4307
5	1.71	2.4319	2.1696
6	3.48	2.1322	2.054
7	3.04	1.2932	1.2122
8	2.92	1.6988	1.5327
overall	25.63	1.6736	1.5235

because the point-clouds are distorted. The WNOJ prior resulted in smaller error for all sequences, and an overall of 9% error reduction. Figure 5 shows comparison plots of the estimated trajectory using the WNOA prior and WNOJ prior for two sequences from the U of T dataset. The estimated trajectory using a WNOJ prior is significantly more accurate.

Again, since the lidar data are motion-distorted, we need to interpolate the pose for each point measurement. We argue that the WNOJ prior (42) offers a more suitable interpolation

scheme than the WNOA prior (23). Results for the U of T dataset achieved a greater reduction in error from using the WNOJ prior than the KITTI dataset, which makes no use of the interpolation scheme.

C. Richmond Hill Dataset

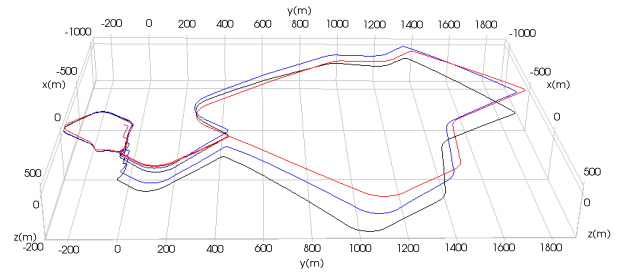


Fig. 7: 3D plots of odometry estimates for sequence 1 of the Richmond Hill dataset: baseline estimator using WNOA prior (black) vs. new estimator using WNOJ prior (blue) when compared against ground truth (red).

A dataset was collected in the city of Richmond Hill, North of Toronto, using our test vehicle. This resulted in three long sequences more than 60km in total. The Richmond Hill dataset features driving in suburban areas, and on highways, which contain less useful geometry and structure. Moreover, the test vehicle was driving more than 90km/h on highways, making this a highly challenging dataset. Similar to the U of T dataset, the point-clouds are motion-distorted.

The results are summarized in Table II. The new estimator using the WNOJ prior outperformed the baseline estimator for all sequences, resulting in a reduction of the overall error by 8.4%. Figure 7 shows a comparison plot of odometry estimates using the WNOJ prior and the WNOA prior, where the odometry from the new estimator is noticeably less biased than the odometry from the baseline estimator, when compared against ground truth.

TABLE II: Errors for the baseline estimator using WNOA prior and new estimator using WNOJ prior, evaluated on the Richmond Hill dataset.

Sequence no.	Distance (km)	Baseline estimator with WNOA prior (%)	New estimator with WNOJ prior (%)
0 (suburban)	17.91	1.5887	1.5094
1 (urban)	7.49	2.3229	1.9363
2 (highway)	35.01	2.3449	2.1627
overall	60.41	2.1180	1.9409

D. Remarks

We found that the new estimator using the WNOJ prior is more sensitive to \mathbf{Q}_c than the baseline estimator using the WNOA prior. This is noticeable because of the difference on the coefficient of each term in the inverse covariance matrix, $\mathbf{Q}_i(t)^{-1}$, between the WNOJ prior (32) and the WNOA prior (15). Despite this, we achieved an improvement on all datasets using \mathbf{Q}_c tuned on the KITTI training set alone. We plan on releasing our datasets for public use in the future.

The cost of STEAM increases linearly with the number of state variables, therefore optimizing for (8) is now 50% more expensive when using the WNOJ prior. The overall increase in runtime is smaller, since the cost for the other components of the estimator, such as point matching, stays invariant.

VII. CONCLUSION AND FUTURE WORK

In this paper, we showed that in continuous-time trajectory estimation, a source of estimator bias can arise when the motion prior cannot sufficiently represent the underlying trajectory. The main contribution of this paper is the derivation of a white-noise-on-jerk motion prior for continuous-time trajectory estimation on $SE(3)$. We showed that the new prior outperforms the existing white-noise-on-acceleration prior employed by STEAM on various lidar datasets, both with and without motion distortion.

Our new formulation of STEAM using the WNOJ prior now has accelerations in the state. Therefore, an extension would be to formulate an estimator that incorporates acceleration measurements from an inertial measurement unit (IMU) directly, rather than pre-integrating to a fixed timestep as is done in many existing inertial estimators.

ACKNOWLEDGMENT

The authors would like to thank Applanix Corporation and the Natural Sciences and Engineering Research Council of Canada (NSERC) for supporting this work, and General Motors (GM) Canada for donating the test vehicle.

REFERENCES

- [1] S.-H. Jung and C. J. Taylor, "Camera Trajectory Estimation Using Inertial Sensor Measurements and Structure from Motion Results," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 2. IEEE, 2001, pp. II-II.
- [2] P. Furgale, T. D. Barfoot, and G. Sibley, "Continuous-Time Batch Estimation Using Temporal Basis Functions," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*. IEEE, 2012, pp. 2088–2095.
- [3] S. Anderson and T. D. Barfoot, "Towards Relative Continuous-Time SLAM," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 1033–1040.

- [4] S. Lovegrove, A. Patron-Perez, and G. Sibley, "Spline Fusion: A Continuous-Time Representation for Visual-Inertial Fusion with Application to Rolling Shutter Cameras," in *BMVC*, 2013.
- [5] R. Dubé, H. Sommer, A. Gawel, M. Bosse, and R. Siegwart, "Non-uniform Sampling Strategies for Continuous Correction Based Trajectory Estimation," in *Robotics and Automation (ICRA), 2016 IEEE International Conference on*. IEEE, 2016, pp. 4792–4798.
- [6] C. H. Tong, P. Furgale, and T. D. Barfoot, "Gaussian Process Gauss-Newton for Non-parametric Simultaneous Localization and Mapping," *The International Journal of Robotics Research*, vol. 32, no. 5, pp. 507–525, 2013.
- [7] T. D. Barfoot, C. H. Tong, and S. Särkkä, "Batch Continuous-Time Trajectory Estimation as Exactly Sparse Gaussian Process Regression," in *Robotics: Science and Systems*, 2014.
- [8] S. Anderson and T. D. Barfoot, "Full STEAM ahead: Exactly Sparse Gaussian Process Regression for Batch Continuous-Time Trajectory Estimation on $SE(3)$," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*. IEEE, 2015, pp. 157–164.
- [9] X. Yan, V. Indelman, and B. Boots, "Incremental Sparse GP Regression for Continuous-Time Trajectory Estimation and Mapping," *Robotics and Autonomous Systems*, vol. 87, pp. 120–132, 2017.
- [10] M. Mukadam, J. Dong, F. Dellaert, and B. Boots, "Simultaneous Trajectory Estimation and Planning via Probabilistic Inference," in *Proceedings of Robotics: Science and Systems (RSS)*, 2017.
- [11] J. Dong, J. G. Burnham, B. Boots, G. Rains, and F. Dellaert, "4D Crop Monitoring: Spatio-temporal Reconstruction for Agriculture," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*.
- [12] M. Warren, M. Paton, K. MacTavish, A. P. Schoellig, and T. D. Barfoot, "Towards Visual Teach and Repeat for GPS-Denied Flight of a Fixed-Wing UAV," in *Field and Service Robotics*. Springer, 2018, pp. 481–498.
- [13] M. Bosse and R. Zlot, "Continuous 3D Scan-Matching with a Spinning 2D Laser," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*. IEEE, 2009, pp. 4312–4319.
- [14] R. Zlot and M. Bosse, "Efficient Large-Scale 3D Mobile Mapping and Surface Reconstruction of an Underground Mine," in *Field and service robotics*. Springer, 2014, pp. 479–493.
- [15] H. Dong and T. D. Barfoot, "Lighting-invariant Visual Odometry Using Lidar Intensity Imagery and Pose Interpolation," in *Field and Service Robotics*. Springer, 2014, pp. 327–342.
- [16] T. D. Barfoot, *State Estimation for Robotics*. Cambridge University Press, 2017.
- [17] J. Zhang and S. Singh, "LOAM: Lidar Odometry and Mapping in Real-time," in *Robotics: Science and Systems*, vol. 2, 2014, p. 9.
- [18] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "MonoSLAM: Real-time Single Camera SLAM," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 6, pp. 1052–1067, 2007.
- [19] S. Anderson and T. D. Barfoot, "RANSAC for Motion-distorted 3D Visual Sensors," in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*. IEEE, 2013, pp. 2093–2099.
- [20] J. Hedborg, P.-E. Forssén, M. Felsberg, and E. Ringaby, "Rolling Shutter Bundle Adjustment," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 1434–1441.
- [21] J. T. Jang, S. T. Moon, S. Han, H. C. Gong, G.-H. Choi, I. H. Hwang, and J. Lyoo, "Trajectory Generation with Piecewise Constant Acceleration and Tracking Control of a Quadcopter," in *Industrial Technology (ICIT), 2015 IEEE International Conference on*.
- [22] M. Mukadam, J. Dong, X. Yan, F. Dellaert, and B. Boots, "Continuous-time Gaussian process motion planning via probabilistic inference," *The International Journal of Robotics Research*, vol. 37, no. 11, pp. 1319–1340, 2018.
- [23] B. Olofsson, J. Antonsson, H. G. Kortier, B. Bernhardsson, A. Robertsson, and R. Johansson, "Sensor Fusion for Robotic Workspace State Estimation," *IEEE/ASME Transactions on Mechatronics*, 2016.
- [24] T. D. Barfoot and P. T. Furgale, "Associating Uncertainty With Three-Dimensional Poses for Use in Estimation Problems," *IEEE Transactions on Robotics*, vol. 30, no. 3, pp. 679–693, 2014.
- [25] T. Y. Tang, D. J. Yoon, F. Pomerleau, and T. D. Barfoot, "Learning a Bias Correction for Lidar-only Motion Estimation," *15th Conference on Computer and Robot Vision (CRV)*, 2018.
- [26] C. E. Rasmussen, "Gaussian Processes in Machine Learning," in *Advanced lectures on machine learning*. Springer, 2004, pp. 63–71.
- [27] A. Geiger, P. Lenz, and R. Urtasun, "Are We Ready for Autonomous Driving? The KITTI Vision Benchmark Suite," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.