

# Towards Consistent Batch State Estimation Using a Time-Correlated Measurement Noise Model

David J. Yoon and Timothy D. Barfoot

**Abstract**—In this paper, we present an algorithm for learning time-correlated measurement covariances for application in batch state estimation. We parameterize the inverse measurement covariance matrix to be block-banded, which conveniently factorizes and results in a computationally efficient approach for correlating measurements across the entire trajectory. We train our covariance model through supervised learning using the groundtruth trajectory. In applications where the measurements are time-correlated, we demonstrate improved performance in both the mean posterior estimate and the covariance (i.e., improved estimator consistency). We use an experimental dataset collected using a mobile robot equipped with a laser rangefinder to demonstrate the improvement in performance. We also verify estimator consistency in a controlled simulation using a statistical test over several trials.

## I. INTRODUCTION

In the research field of probabilistic robotics, we formulate state estimation using probability theory in order to handle the uncertainty of our sensor measurements [1]. Modern algorithms are capable of solving for the trajectory using all measurements in a batch solution [2], [3], resulting in accurate estimates in real-world applications.

Estimator performance is affected by the uncertainty we assign to the noisy measurements, i.e., our sensor noise models. Ideally, we wish to learn our noise models from data. In most applications, where the noise is assumed Gaussian, common practice is to assume that measurements acquired at different times are corrupted by statistically independent noise. Using the groundtruth trajectory, the sample measurement covariance at the marginal level (i.e., at a single timestep) can then be applied as the noise model. However, the uncorrelated assumption is not always accurate (see Figure 1). Not accounting for the correlations degrades the quality of the estimator and can lead to overconfident estimates [4]. Computing the sample covariance over the entire trajectory is also not tractable, as we often work with trajectories that are many thousands of timesteps in length.

In order to improve the state of the art in batch state estimation, we developed a data-driven approach for learning time-correlated noise models in a supervised learning fashion (i.e., using the groundtruth trajectory). By parameterizing our noise model as the inverse covariance, our method is computationally efficient for large trajectories. To the best of our knowledge, learning time-correlated measurement covariances for batch state estimation has not been

All authors are with the University of Toronto Institute for Aerospace Studies (UTIAS), 4925 Dufferin St, Ontario, Canada. david.yoon@robotics.utias.utoronto.ca, tim.barfoot@utoronto.ca

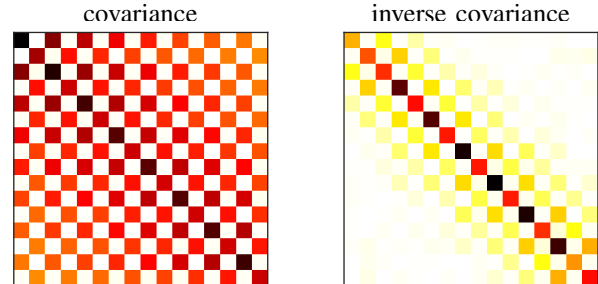


Fig. 1: A visualization of the sample measurement covariance (left) and its inverse (right) for a 2D landmark over 8 timesteps, showing the presence of time correlation in a real dataset. Darker colours indicate larger magnitudes and lighter colours are magnitudes closer to zero. The covariance appears dense, but its inverse appears banded, which our proposed method exploits.

demonstrated before in the mobile robotics research field. We verify that our proposed method results in a consistent batch estimator in a controlled simulation via a statistical test over several trials. We demonstrate improved performance in both the posterior mean and covariance (as gauged by estimator consistency) using an experimental dataset with time-correlated measurements.

## II. RELATED WORK

In applications where the sensor measurements are not time-correlated, several options exist for learning measurement covariances. The default method is to assume the noise characteristics remain constant throughout the trajectory, in which case the maximum likelihood sample covariance is a reasonable choice [2]. If the groundtruth trajectory is unknown (i.e., unsupervised training), Expectation-Maximization (EM) can be applied to iteratively refine the trajectory estimate and measurement covariance [5]–[7].

In some applications, the noise characteristics may vary throughout the trajectory. An early approach for handling this problem was the adaptive Kalman Filter [8], [9], a reactive method that refines the sample covariance over a trailing window of frames. Rather than reacting, which may be delayed, a predictive approach, CELLO, was presented by Vega-Brown et al. [10]. By introducing known features that describe the measurements, local kernel estimates of the covariance were learned by weighting measurements based on feature similarity. Landry et al. [11] adapt CELLO to model the uncertainty of pointcloud registration for Iterative Closest Point (ICP). Later work extended CELLO to unsupervised training using EM [12], [13].

Deep learning presents another option for learning a varying noise model. Brossard et al. [14] regress the noise variances of a vehicle model that penalizes lateral and vertical



This choice of parameterization factors the conditional likelihood,  $p(\mathbf{y}|\mathbf{x})$ , in a convenient way. Substituting  $\mathbf{R}^{-1} = \mathbf{S}^T \mathbf{W} \mathbf{S}$  with bandwidth  $b$  into  $-\ln p(\mathbf{y}|\mathbf{x})$ ,

$$\begin{aligned} & \frac{1}{2} \mathbf{e}_y^T \mathbf{R}^{-1} \mathbf{e}_y - \frac{1}{2} \ln |\mathbf{R}^{-1}| = \\ & \sum_{k=b+1}^K \frac{1}{2} \left( \mathbf{e}_{k-b:k}^T [\mathbf{S}_{k,b:1} \mathbf{1}]^T \mathbf{W}_k [\mathbf{S}_{k,b:1} \mathbf{1}] \mathbf{e}_{k-b:k} - \ln |\mathbf{W}_k| \right) \\ & + \frac{1}{2} \mathbf{e}_{1:b}^T [\mathbf{S}_{b,b:1} \mathbf{1}]^T \mathbf{W}_b [\mathbf{S}_{b,b:1} \mathbf{1}] \mathbf{e}_{1:b} - \frac{1}{2} \ln |\mathbf{W}_b| \\ & + \vdots \text{ (factors between)} \\ & + \frac{1}{2} \mathbf{e}_{y,1}^T \mathbf{W}_1 \mathbf{e}_{y,1} - \frac{1}{2} \ln |\mathbf{W}_1|, \end{aligned} \quad (4)$$

$$\mathbf{e}_{k_1:k_2}^T = [\mathbf{e}_{y,k_1}^T \quad \mathbf{e}_{y,k_1+1}^T \quad \cdots \quad \mathbf{e}_{y,k_2}^T], \quad (5)$$

$$\mathbf{S}_{k,b':1} = [\mathbf{S}_{k,b'} \quad \mathbf{S}_{k,b'-1} \quad \cdots \quad \mathbf{S}_{k,1}]. \quad (6)$$

For example with bandwidth  $b = 1$ ,

$$\begin{aligned} & \frac{1}{2} \mathbf{e}_y^T \mathbf{R}^{-1} \mathbf{e}_y - \frac{1}{2} \ln |\mathbf{R}^{-1}| = \\ & \sum_{k=2}^K \frac{1}{2} \left( \mathbf{e}_{k-1:k}^T [\mathbf{S}_{k,1} \mathbf{1}]^T \mathbf{W}_k [\mathbf{S}_{k,1} \mathbf{1}] \mathbf{e}_{k-1:k} - \ln |\mathbf{W}_k| \right) \\ & + \frac{1}{2} \mathbf{e}_{y,1}^T \mathbf{W}_1 \mathbf{e}_{y,1} - \frac{1}{2} \ln |\mathbf{W}_1|. \end{aligned} \quad (7)$$

Expanding the product of the correlation parameters and the errors,  $[\mathbf{S}_{k,1} \mathbf{1}] \mathbf{e}_{k-1:k} = \mathbf{S}_{k,1} \mathbf{e}_{y,k-1} + \mathbf{e}_{y,k}$ , which we recognize to be similar to the differencing approach for handling correlations in filters [24], but with the measurement errors instead of the measurements,  $\mathbf{y}_{k-1}$  and  $\mathbf{y}_k$ .

### C. Learning a Constant Noise Model

When we expect the noise characteristics to not vary throughout the trajectory, we can set  $\mathbf{W}_k = \mathbf{W}_*$  and  $\mathbf{S}_{k,b:1} = \mathbf{S}_*$  constants for  $k = b+1, b+2, \dots, K$ .

Given a training dataset of length  $N$ ,  $\mathcal{D} = \{\mathbf{x}_{1:N}, \mathbf{y}_{1:N}\}^2$ , where  $\mathbf{x}_{1:N}$  is the groundtruth trajectory, we can optimize for  $\theta_* = \{\mathbf{W}_*, \mathbf{S}_*\}$  using a maximum<sup>3</sup> likelihood objective:

$$\mathcal{L} = \sum_{i=b+1}^N \frac{1}{2} \left( \mathbf{e}_{i-b:i}^T [\mathbf{S}_* \mathbf{1}]^T \mathbf{W}_* [\mathbf{S}_* \mathbf{1}] \mathbf{e}_{i-b:i} - \ln |\mathbf{W}_*| \right), \quad (9)$$

where we dropped the factors corresponding to  $i = 1, \dots, b$ . We assume the noise models for these factors are known. In practice, they can be approximated by optimizing a similar objective to (9), but for smaller bandwidths.

We take the partial derivatives of  $\mathcal{L}$  with respect to  $\theta_*$  and set them to zero to optimize (9). Starting with  $\mathbf{S}_*$ ,

$$\frac{\partial \mathcal{L}}{\partial \mathbf{S}_*} = \sum_{i=b+1}^N \mathbf{W}_* \left( \mathbf{e}_{y,i} \mathbf{e}_{i-b:i-1}^T + \mathbf{S}_* \mathbf{e}_{i-b:i-1} \mathbf{e}_{i-b:i-1}^T \right). \quad (10)$$

<sup>2</sup>We index using  $i = 1, 2, \dots, N$  to distinguish the training dataset from the test dataset, which uses  $k = 1, 2, \dots, K$ .

<sup>3</sup>We are technically minimizing the objective here since we choose to work with the negative logarithm.

Setting to zero, we obtain

$$\mathbf{S}_* = - \left( \sum_{i=b+1}^N \mathbf{e}_{y,i} \mathbf{e}_{i-b:i-1}^T \right) \left( \sum_{i=b+1}^N \mathbf{e}_{i-b:i-1} \mathbf{e}_{i-b:i-1}^T \right)^{-1} \quad (11)$$

Taking the partial derivative<sup>4</sup> of  $\mathcal{L}$  with respect to  $\mathbf{W}_*$ ,

$$\frac{\partial \mathcal{L}}{\partial \mathbf{W}_*} = \sum_{i=b+1}^N \frac{1}{2} \left( [\mathbf{S}_* \mathbf{1}] \mathbf{e}_{i-b:i} \mathbf{e}_{i-b:i}^T [\mathbf{S}_* \mathbf{1}]^T - \mathbf{W}_*^{-1} \right). \quad (12)$$

Setting to zero, we obtain

$$\mathbf{W}_* = (N-b) \left( \sum_{i=b+1}^N [\mathbf{S}_* \mathbf{1}] \mathbf{e}_{i-b:i} \mathbf{e}_{i-b:i}^T [\mathbf{S}_* \mathbf{1}]^T \right)^{-1}, \quad (13)$$

where  $\mathbf{S}_*$  is known from (11).

### D. Learning a Varying Noise Model

When the noise characteristics are expected to vary throughout the trajectory, we can formulate a prediction scheme through machine learning by introducing a feature,  $\psi \in \mathbb{R}^L$ , for each measurement. Given a training dataset,  $\mathcal{D} = \{\mathbf{x}_{1:N}, \mathbf{y}_{1:N}, \psi_{1:N}\}$ , our noise model for a target feature,  $\psi_*$ , at test time is

$$\theta_* = \theta(\psi_* | \mathcal{D}) = \{\mathbf{W}(\psi_* | \mathcal{D}), \mathbf{S}(\psi_* | \mathcal{D})\}. \quad (14)$$

We choose to apply the local kernel estimation approach of Vega-Brown et al. [33] and adapt it to predict our correlated noise parameters. We first review their methodology for uncorrelated noise prediction (i.e.,  $\theta(\psi_* | \mathcal{D}) = \mathbf{W}(\psi_* | \mathcal{D})$ ), then present how we adapt their methodology.

Vega-Brown et al. [33] formulate an estimator for  $\theta_*$  given a target feature,  $\psi_*$ . A joint posterior between  $\theta_*$  and  $\theta_{1:N}$  is formed using Bayes' rule and  $\theta_{1:N}$  is then marginalized out, resulting in

$$p(\theta_* | \psi_*, \mathcal{D}) \propto \left( \prod_{i=1}^N p(\mathbf{y}_i | \mathbf{x}_i, \theta_*, \psi_i, \psi_*) \right) p(\theta_* | \psi_*), \quad (15)$$

where we note the uncorrelated factorization of the likelihood terms. They show that an Inverse-Wishart distribution can be applied as the prior on the covariance, which results in a Student's t distribution over the noise [13]. In our work we will apply an uninformative prior,  $p(\theta_* | \psi_*) \propto 1$ , which is an adaptation of their prior work, CELLO [10], [12].

The key innovation of Vega-Brown et al. [33] is their choice in modelling the likelihood in (15) (referred to as the extended likelihood), as

$$p(\mathbf{y}_i | \mathbf{x}_i, \theta_*, \psi_i, \psi_*) \propto p(\mathbf{y}_i | \mathbf{x}_i, \theta_*)^{h(\psi_i, \psi_*)}, \quad (16)$$

where  $h(\psi_i, \psi_*)$  is a kernel function. The extended likelihood is modelled using the known likelihood,  $p(\mathbf{y}_i | \mathbf{x}_i, \theta_*)$ . Motivated by enforcing a notion of smoothness, they derived this model by bounding the information divergence between the two likelihood distributions. We refer to Vega-Brown et al. [33] for a more detailed explanation.

<sup>4</sup>While  $\mathbf{W}_*$  is a symmetric matrix, we do not need to take a structured derivative since we are setting the result to zero.

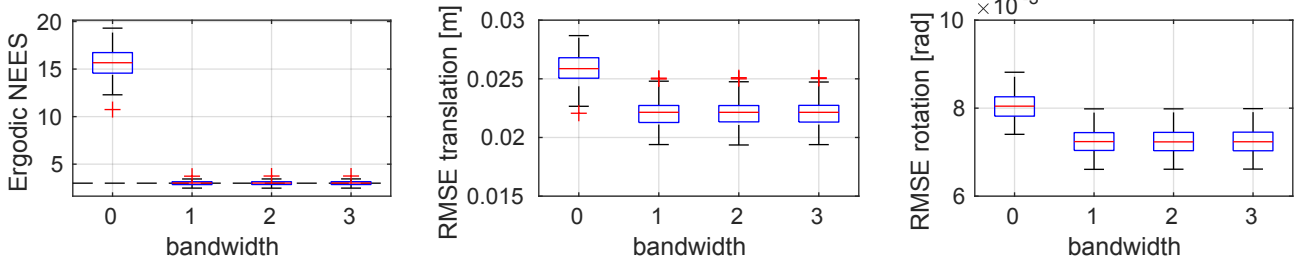


Fig. 2: Simulations: Box plots of the ergodic Normalized Estimation Error Squared (NEES), Root Mean Squared Error (RMSE) translation, and RMSE rotation over 100 simulation trials. Ideally the ergodic NEES should be the dimension of the state,  $D = 3$ . Not modelling the time correlations (bandwidth 0) achieves the worst performance. A bandwidth of 1 is enough to fully model the correlations in the simulated data. The proposed learning algorithm is capable of learning the correct noise properties even when the model is over-parameterized (bandwidth greater than 1 in this case).

We adapt their methodology for time-correlated noise by applying the correlated factorization scheme shown in (4), replacing the uncorrelated factorization in (15):

$$p(\theta_* | \psi_*, \mathcal{D}) \propto p(\mathbf{y}_{1:b} | \mathbf{x}_{1:b}, \psi_{1:b}) \quad (17)$$

$$\times \left( \prod_{i=b+1}^N p(\mathbf{y}_i | \mathbf{y}_{i-b:i-1}, \mathbf{x}_{i-b:i}, \theta_*, \psi_i, \psi_*) \right) p(\theta_* | \psi_*).$$

We use the negative logarithm of (17) as our training objective, optimizing for  $\theta_* = \{\mathbf{W}_*, \mathbf{S}_*\}$ . As mentioned, we will apply an uninformative prior,  $p(\theta_* | \psi_*) \propto 1$ . Modelling the extended likelihood using our known likelihood and kernel function, the objective function is

$$\mathcal{L}_h = \sum_{i=b+1}^N \frac{h_i}{2} (\mathbf{e}_{i-b:i}^T [\mathbf{S}_* \mathbf{1}]^T \mathbf{W}_* [\mathbf{S}_* \mathbf{1}] \mathbf{e}_{i-b:i} - \ln |\mathbf{W}_*|), \quad (18)$$

where  $h_i = h(\psi_i, \psi_*)$  and, similar to (9), we drop the lower bandwidth factors corresponding to  $i = 1, 2, \dots, b$ . The resulting objective (18) is similar to (9), but now with each term weighted by evaluations of the kernel,  $h_i$ . Taking the partial derivatives and setting them to zero,

$$\mathbf{S}(\psi_* | \mathcal{D}) = - \left( \sum_{i=b+1}^N h_i \mathbf{e}_{y,i} \mathbf{e}_{i-b:i-1}^T \right) \times \left( \sum_{i=b+1}^N h_i \mathbf{e}_{i-b:i-1} \mathbf{e}_{i-b:i-1}^T \right)^{-1}, \quad (19)$$

$$\mathbf{W}(\psi_* | \mathcal{D}) = \left( \frac{1}{H} \sum_{i=b+1}^N h_i [\mathbf{S}_* \mathbf{1}] \mathbf{e}_{i-b:i} \mathbf{e}_{i-b:i}^T [\mathbf{S}_* \mathbf{1}]^T \right)^{-1}, \quad (20)$$

where  $H = \sum_{i=b+1}^N h(\psi_i, \psi_*)$ . For our work we use the squared exponential kernel,

$$h(\psi_i, \psi_*) = \exp \left( -\frac{1}{2} (\psi_i - \psi_*)^T \mathbf{M} (\psi_i - \psi_*) \right). \quad (21)$$

The weight matrix,  $\mathbf{M}$ , is a hyperparameter that we train via maximum likelihood as suggested by Vega-Brown et al. [10].

## IV. EXPERIMENTAL RESULTS

### A. Dataset

A robot with a 2D laser rangefinder and wheel odometers is driven around tubular landmarks (see Figure 3). The groundtruth for the robot trajectory and landmark positions



Fig. 3: The dataset for demonstrating the proposed method was acquired by a robot with a laser rangefinder and wheel odometers that was driven around tubular landmarks. A Vicon motion capture system provides the groundtruth.

are accurately provided by a Vicon motion capture system. The rangefinder measurements are preprocessed into 2D point landmark measurements. The dataset is approximately 12000 timesteps in length with a discrete-time sampling period of  $\Delta t = 0.1s$ . This dataset has been used before for demonstrating advances in batch estimation [7], [34].

We can use the groundtruth to check for time correlations in the measurements. We stack the landmark measurements over a short window of 8 timesteps without overlap and calculate the sample measurement covariance<sup>5</sup>. The magnitude of the resulting covariance and its inverse are visualized in Figure 1, clearly showing signs of time correlations (i.e., they do not appear to be block-diagonal). A similar experiment for the velocity measurements reveals that they are also time-correlated, but not as severely as the landmark measurements.

### B. Problem Setup

We perform localization as a batch state estimation problem, where the data association and true landmark locations (i.e., the map) are assumed to be known. Our state is  $\mathbf{x} = \{\mathbf{T}_1, \mathbf{T}_2, \dots, \mathbf{T}_K\}$ , where  $\mathbf{T}_k = \mathbf{T}_{k,i} \in SE(2)$  is the relative transform between the robot frame at time  $k$  and a fixed reference frame,  $i$ . We use the odometer measurements in a constant-velocity error function (motion model),

$$\mathbf{e}_{v,k} = \ln \left( \exp(\Delta t \mathbf{v}_k^\wedge) \mathbf{T}_{k-1} \mathbf{T}_k^{-1} \right)^\vee, \quad (22)$$

where  $\mathbf{v}_k^T = [-v_k \ 0 \ -\omega_k]$  are the forward, lateral, and rotational speeds<sup>6</sup> in the robot frame, and  $\exp(\cdot)$  and  $\ln(\cdot)$

<sup>5</sup>One may be tempted to use this covariance for estimation by grouping measurements, but correlations will be neglected (grouping without overlap) or measurements will be double-counted (grouping with overlap).

<sup>6</sup>The negative sign is from our convention for the exponential map [2].

TABLE I: Experiments: Quantitative results on the dataset with the best results in bold font. Note that the ideal value for the ergodic NEES in this problem is 3, the dimension of the state. The trend is that the SVD-FEAT-X methods perform the best, with larger bandwidth settings generally performing better.

Folds	Ergodic NEES					RMSE translation [m]					RMSE rotation [rad]				
	01	02	03	04	AVG	01	02	03	04	AVG	01	02	03	04	AVG
P2P-CONST	15.4	15.4	19.5	15.9	<b>16.6</b>	0.0215	0.0211	0.0240	0.0226	<b>0.0223</b>	0.0175	0.0167	0.0188	0.0149	<b>0.0169</b>
SVD-CONST	12.4	16.2	16.9	12.9	<b>14.6</b>	0.0219	0.0225	0.0241	0.0226	<b>0.0228</b>	0.0185	0.0187	0.0198	0.0179	<b>0.0187</b>
SVD-FEAT-0	11.4	14.1	15.0	13.5	<b>13.5</b>	0.0213	0.0215	0.0233	0.0224	<b>0.0221</b>	0.0150	0.0144	<b>0.0164</b>	0.0150	<b>0.0152</b>
SVD-FEAT-1	4.53	5.25	5.64	5.85	<b>5.32</b>	<b>0.0196</b>	0.0196	0.0218	0.0218	<b>0.0207</b>	0.0134	<b>0.0128</b>	<b>0.0164</b>	0.0121	<b>0.0137</b>
SVD-FEAT-3	3.71	3.85	4.77	4.61	<b>4.23</b>	<b>0.0196</b>	0.0193	0.0219	<b>0.0217</b>	<b>0.0206</b>	<b>0.0131</b>	0.0130	0.0165	0.0119	<b>0.0136</b>
SVD-FEAT-5	<b>3.41</b>	<b>3.37</b>	<b>4.44</b>	<b>4.42</b>	<b>3.91</b>	<b>0.0196</b>	<b>0.0189</b>	<b>0.0216</b>	0.0218	<b>0.0205</b>	0.0133	0.0129	0.0166	<b>0.0117</b>	<b>0.0136</b>

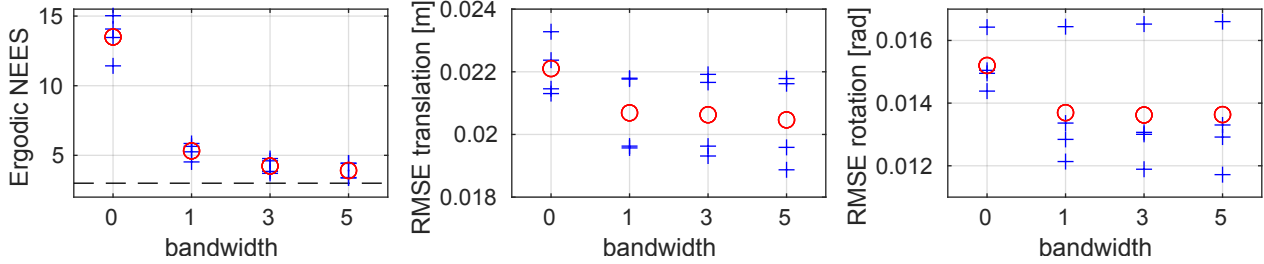


Fig. 4: Experiments: Plots of the results in Table I for SVD-FEAT-X methods. A ‘+’ marks an evaluation of a single fold, and a ‘o’ indicates the average. The biggest improvements are gained by changing the bandwidth to  $b = 1$  from  $b = 0$  (i.e., no time correlations). In general we still see minor improvements for  $b > 1$ , the exception being the rotation error for fold 03. However, the average rotation error still appears to decrease and stabilize for larger  $b$ .

are the  $SE(2)$  exponential map and its inverse, respectively.

There are a varying number of landmarks measured at each timestep throughout the dataset. Learning a correlated noise model at the landmark-measurement level will require tedious bookkeeping of the observed landmarks and ignore potential correlations between landmarks at a single timestep. We instead preprocess the measurements into pose pseudomeasurements,  $\mathbf{T}_{mk} = \mathbf{T}_{mk,i} \in SE(2)$ , and learn a varying noise model for the pseudomeasurements. This is a common approach for learning noise models for rich sensor data [11], [15]–[18], [35]. We compute the pseudomeasurements using Singular Value Decomposition (SVD) and use

$$\mathbf{e}_{y,k} = \ln(\mathbf{T}_{mk} \mathbf{T}_k^{-1})^\vee. \quad (23)$$

We learn our noise model by defining a feature,  $\psi_k \in \mathbb{R}^6$ , that comprises the number of visible landmarks and their spread<sup>7</sup> in the robot frame at time  $k$ .

### C. Simulation

We simulate multiple trials using a Bayesian experiment setup [2] and evaluate consistency using the NEES  $\chi^2$  statistical test [23], [36]. Measurements are simulated by

$$\mathbf{y}_k = \mathbf{g}(\mathbf{x}_k, \ell) + \mathbf{w}_k, \quad \mathbf{w}_k = \mathbf{S}' \mathbf{w}_{k-1} + \mathbf{n}, \quad (24)$$

where  $\mathbf{y}_k$  is a stacked vector of all landmark measurements,  $\mathbf{g}(\cdot, \ell)$  is a nonlinear model that computes landmark measurements in the robot frame at pose  $\mathbf{x}_k$ ,  $\ell$  are the true landmark positions (i.e., map), and  $\mathbf{n} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}')$ . Measurements outside a  $270^\circ$  field-of-view and greater than a maximum range of  $5m$  are not counted, varying the number of visible landmarks at each timestep<sup>8</sup>. The noise parameters are set as  $\mathbf{R}' = (0.03m)^2 \times \mathbf{1}$  and  $\mathbf{S}' = 0.9 \times \mathbf{1}$ . For each simulation

trial we generate a training sequence for the noise model and a test sequence for evaluation.

The marginal NEES at time  $k$  is computed as  $\epsilon_{x,k} = \mathbf{e}_{x,k}^T \hat{\mathbf{P}}_k^{-1} \mathbf{e}_{x,k}$ , where  $\mathbf{e}_{x,k}$  is the marginal error and  $\hat{\mathbf{P}}_k$  is the marginal posterior covariance at time  $k$ . The NEES  $\chi^2$  statistical test aggregates  $\epsilon_{x,k}$  over all  $N_t$  trials and checks if the result is within a lower and upper bound:

$$\mathcal{Q}_{\chi^2(N_t D)}(\ell) \leq \sum_{j=1}^{N_t} \epsilon_{x,k}^j \leq \mathcal{Q}_{\chi^2(N_t D)}(u), \quad (25)$$

where  $\ell$  and  $u$  are lower and upper confidence bounds, respectively, and  $\mathcal{Q}_{\chi^2(N_t D)}(\cdot)$  is the quantile function for the  $\chi^2$  distribution with  $N_t D$  degrees of freedom ( $D = 3$  is the dimension of the state).

Simulating  $N_t = 100$  trials, we apply our proposed approach for the SVD pseudomeasurements. With bandwidth  $b = 1$ , 0.2022% of 3000 test timesteps fall outside a 99.8% confidence interval ( $\ell = 0.001$ ,  $u = 0.999$ ), which is reasonable. Using a 95% confidence interval ( $\ell = 0.025$ ,  $u = 0.975$ ), 5.09% fall outside the bounds, which is also reasonable. Similar results are produced for  $b > 1$ , demonstrating that we can over-parameterize and still learn an accurate covariance model<sup>9</sup>. When  $b = 0$  (i.e., original CELLO [10]), none of the trials fall within reasonable confidence intervals because the resulting estimator is overconfident.

We can also take an *ergodic* assumption and compute the average NEES over the entire test sequence in each trial:  $\frac{1}{K} \sum_{k=1}^K \epsilon_{x,k}$ . The ergodic NEES does not take into account the time correlations in the estimated posterior and thus is not valid for a statistical test. It is still useful to compute as the value on average should match the dimension of the

<sup>7</sup>2D sample mean and covariance of the landmark point measurements.

<sup>8</sup>These specifications are similar to the real dataset.

<sup>9</sup>Our experiments show that more training data is required for over-parameterized settings of the bandwidth. In other words, performance may be degraded if  $b$  is set too large with insufficient training data.

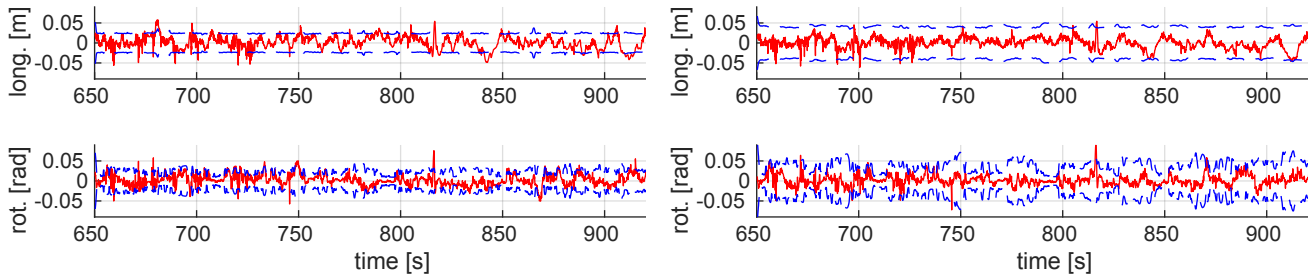


Fig. 5: Experiments: Shown above are error plots with a  $3\sigma$  uncertainty envelope. We plot the longitudinal translation errors and the rotational errors. We omit the lateral translation errors in the interest of space. SVD-FEAT-0 (left column) does not model any time correlations. SVD-FEAT-5 (right column), which learns an inverse measurement covariance with a block-bandwidth of 5, appears to have its errors more contained within the uncertainty envelope.

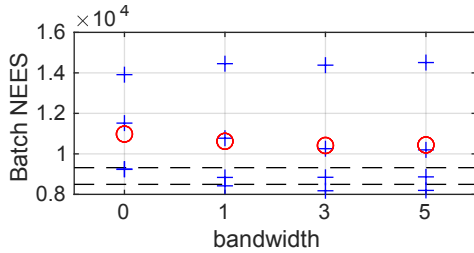


Fig. 6: Experiments: The batch NEES for SVD-FEAT- $X$  with varying bandwidths. A ‘+’ marks a single fold, and a ‘o’ marks the average. Dashed lines indicate the 99.8% confidence interval, which most points do not meet. Meeting this confidence interval is the subject of future work.

state,  $D = 3$ . We plot the ergodic NEES as a box plot in Figure 2 for varying bandwidths,  $b$ , along with box plots of the RMSE for translation and rotation. Performance on all three metrics is poor when  $b = 0$ . When  $b = 1, 2$ , and  $3$ , performance is improved and similar.

#### D. Real Data

We divide the dataset into 4 equally-sized folds ( $\sim 3000$  timesteps). We test on each fold by training on the 3 other folds. We compare the following methods:

- 1) *P2P-CONST*: Replaces the SVD pseudomeasurements with the original 2D landmark measurements.
- 2) *SVD-CONST*: Learn a constant covariance for the SVD pseudomeasurements (no correlations are modelled).
- 3) *SVD-FEAT- $X$* : Learn varying noise parameters for the SVD pseudomeasurements with bandwidth  $b = X$ .

For P2P-CONST and SVD-CONST, we train a constant covariance for the motion model in (22). For SVD-FEAT- $X$ , we also band the motion model covariance by  $b = X$  to model time correlations in the velocity.

Table I shows our quantitative results. The best performers are the SVD-FEAT- $X$  methods, with larger bandwidth settings fairing better in general. Figure 4 plots the results of the SVD-FEAT- $X$  methods, visualizing a general trend in performance improvement. Figure 5 qualitatively compares error plots between SVD-FEAT-0 and SVD-FEAT-5. The errors appear to be more contained within the  $3\sigma$  uncertainty envelope when the time correlations are modelled.

TABLE II: Timing results on SVD-FEAT- $X$  for varying bandwidth.

Bandwidth	0	1	2	3	4	5
Predict	2.35s	3.05s	3.66s	4.06s	4.51s	4.74s
Optimize	0.85s	1.00s	1.07s	1.34s	1.47s	1.62s

## V. DISCUSSION AND FUTURE WORK

We present improvements in batch estimation for both the posterior mean and covariance. We demonstrate consistency using a statistical test in a controlled simulation and show improvements to the ergodic NEES on real data, both of which evaluate at the marginal level. For future work, we will focus on achieving consistency using real data through a statistical test. In theory, computing the *batch* NEES over the entire trajectory should account for the correlations in the posterior, in contrast to the ergodic NEES, and be valid for a  $\chi^2$  statistical test. The batch NEES is  $\epsilon = \mathbf{e}^T \hat{\mathbf{P}}^{-1} \mathbf{e}$ , where  $\hat{\mathbf{P}}$  is the posterior covariance of the entire trajectory and  $\mathbf{e}$  is the full-trajectory error. Figure 6 shows the batch NEES using the SVD-FEAT- $X$  methods on our dataset with the 99.8% confidence interval indicated by the dashed lines. Our methods tend to still be overconfident on some folds, suggesting there are remaining nonidealities for which we must account. One such nonideality might be that the groundtruth (i.e., the Vicon motion capture) is not truly perfect. We may need to account for this uncertainty in both training and evaluation. Unsupervised training using EM [5], [12] may be of value here and is also an item for future work.

The average wall-clock times for the SVD-FEAT- $X$  methods are shown in Table II for a 3000 timestep test sequence. ‘Predict’ evaluates the noise parameters for every timestep (see (19), (20)). ‘Optimize’ runs Gauss-Newton (GN) until convergence and Takahashi’s method [37] for efficiently computing the marginal posterior covariances. The wall-clock times appear to increase linearly by bandwidth. However, even the slowest times are reasonable given the sequence length. There is potential for online application via a sliding window, which we leave as future work targeting larger-scale trajectories and richer sensor data.

## VI. CONCLUSION

We propose a method for modelling and learning a time-correlated noise model. Our method is applicable to batch state estimation and is capable of modelling correlations over long time periods. We train our models using the groundtruth trajectory and show improvements in estimator performance in simulation and on an experimental dataset.

## ACKNOWLEDGMENT

We thank the Natural Sciences and Engineering Research Council of Canada (NSERC) for supporting this work.

## REFERENCES

- [1] S. Thrun, W. Burgard, and D. Fox, "Probabilistic robotics (intelligent robotics and autonomous agents)," 2005.
- [2] T. D. Barfoot, *State Estimation for Robotics*. Cambridge University Press, 2017.
- [3] F. Dellaert and M. Kaess, *Factor Graphs for Robot Perception*. Now Publishers Inc., 2017.
- [4] C. Merfels and C. Stachniss, "Sensor fusion for self-localisation of automated vehicles," *PFG—Journal of Photogrammetry, Remote Sensing and Geoinformation Science*, vol. 85, no. 2, pp. 113–126, 2017.
- [5] R. H. Shumway and D. S. Stoffer, "An approach to time series smoothing and forecasting using the em algorithm," *Journal of time series analysis*, vol. 3, no. 4, pp. 253–264, 1982.
- [6] Z. Ghahramani and G. E. Hinton, "Parameter estimation for linear dynamical systems," U. of Toronto, Tech. Rep. CRG-TR-96-2, 1996.
- [7] T. D. Barfoot, J. R. Forbes, and D. J. Yoon, "Exactly sparse gaussian variational inference with application to derivative-free batch nonlinear state estimation," *International Journal of Robotics Research*, 2020.
- [8] R. Mehra, "On the identification of variances and adaptive kalman filtering," *IEEE Transactions on automatic control*, vol. 15, no. 2, pp. 175–184, 1970.
- [9] R. F. Stengel, *Optimal control and estimation*, 1994.
- [10] W. Vega-Brown, A. Bachrach, A. Bry, J. Kelly, and N. Roy, "CELLO: A fast algorithm for covariance estimation," in *International Conference on Robotics and Automation*, 2013, pp. 3160–3167.
- [11] D. Landry, F. Pomerleau, and P. Giguere, "CELLO-3D: Estimating the covariance of ICP in the real world," in *International Conference on Robotics and Automation*, 2019.
- [12] W. Vega-Brown and N. Roy, "CELLO-EM: Adaptive sensor models without ground truth," in *International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 1907–1914.
- [13] V. Peretroukhin, W. Vega-Brown, N. Roy, and J. Kelly, "PROBE-GK: Predictive robust estimation using generalized kernels," in *International Conference on Robotics and Automation*, 2016, pp. 817–824.
- [14] M. Brossard, A. Barrau, and S. Bonnabel, "AI-IMU dead-reckoning," *IEEE Transactions on Intelligent Vehicles*, vol. 5, no. 4, pp. 585–595, 2020.
- [15] K. Liu, K. Ok, W. Vega-Brown, and N. Roy, "Deep inference for covariance estimation: Learning Gaussian noise models for state estimation," in *International Conference on Robotics and Automation*, 2018.
- [16] R. L. Russell and C. Reale, "Multivariate uncertainty in deep learning," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.
- [17] I. Torroba, C. I. Sprague, N. Bore, and J. Folkesson, "Point-NetKL: Deep inference for GICP covariance estimation in bathymetric SLAM," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4078–4085, 2020.
- [18] A. De Maio and S. Lacroix, "Deep bayesian ICP covariance estimation," in *International Conference on Robotics and Automation*, 2022.
- [19] S. J. Julier and J. K. Uhlmann, "A non-divergent estimation algorithm in the presence of unknown correlations," in *American Control Conference*, vol. 4, 1997, pp. 2369–2373.
- [20] S. J. Julier, "Fusion of dependent information in posegraphs," in *International Conference on Multisensor Fusion and Integration for Intelligent Systems*, 2012, pp. 504–509.
- [21] B. Noack, S. J. Julier, and U. D. Hanebeck, "Treatment of biased and dependent sensor data in graph-based SLAM," in *International Conference on Information Fusion*, 2015, pp. 1862–1867.
- [22] B. Noack, U. Orguner, and U. D. Hanebeck, "Nonlinear decentralized data fusion with generalized inverse covariance intersection," in *International Conference on Information Fusion*, 2019, pp. 1–7.
- [23] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 2004.
- [24] A. Bryson Jr and L. Henrikson, "Estimation using sampled data containing sequentially correlated noise," *Journal of Spacecraft and Rockets*, vol. 5, no. 6, pp. 662–665, 1968.
- [25] K. Wang, Y. Li, and C. Rizos, "Practical approaches to Kalman filtering with time-correlated measurement errors," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 2, pp. 1669–1681, 2012.
- [26] G. Chang, "On Kalman filter for linear system with colored measurement noise," *Journal of Geodesy*, vol. 88, no. 12, pp. 1163–1170, 2014.
- [27] M. G. Petovello, K. O'Keefe, G. Lachapelle, and M. E. Cannon, "Consideration of time-correlated errors in a kalman filter applicable to GNSS," *Journal of Geodesy*, vol. 83, no. 1, pp. 51–56, 2009.
- [28] Y. Xu, Y. S. Shmaliy, T. Shen, D. Chen, M. Sun, and Y. Zhuang, "Ins/uwb-based quadrotor localization under colored measurement noise," *IEEE Sensors Journal*, vol. 21, no. 5, pp. 6384–6392, 2020.
- [29] X. Wang, Y. Liang, Q. Pan, C. Zhao, and F. Yang, "Nonlinear gaussian smoothers with colored measurement noise," *IEEE Transactions on Automatic Control*, vol. 60, no. 3, pp. 870–876, 2014.
- [30] K. Lee and E. N. Johnson, "State estimation using gaussian process regression for colored noise systems," in *IEEE Aerospace Conference*. IEEE, 2017, pp. 1–8.
- [31] S. J. Julier, R. De Nardi, and J. D. Nelson, "Multi-rate estimation of coloured noise models in graph-based estimation algorithms," in *International Conference on Information Fusion*, 2012, pp. 2087–2093.
- [32] S. Demko, W. F. Moss, and P. W. Smith, "Decay rates for inverses of band matrices," *Mathematics of Computation*, vol. 43, no. 168, pp. 491–499, 1984.
- [33] W. R. Vega-Brown, M. Doniec, and N. G. Roy, "Nonparametric bayesian inference on multivariate exponential families," *Advances in Neural Information Processing Systems*, vol. 27, 2014.
- [34] 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*, vol. 10, 2014, pp. 1–10.
- [35] J. N. Wong, D. J. Yoon, A. P. Schoellig, and T. D. Barfoot, "Variational inference with parameter learning applied to vehicle trajectory estimation," *IEEE Robotics and Automation Letters*, 2020.
- [36] Z. Chen, C. Heckman, S. Julier, and N. Ahmed, "Weak in the NEES?: Auto-tuning kalman filters with bayesian optimization," in *International Conference on Information Fusion*, 2018, pp. 1072–1079.
- [37] K. Takahashi, J. Fagan, and M.-S. Chen, "A sparse bus impedance matrix and its application to short circuit study," in *Proceedings of the PICA Conference*, 1973.