Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

State Estimation with Event-Based Inputs Using Stochastic Triggers

Benjamin Noack, Christopher Funk, Susanne Radtke, and Uwe D. Hanebeck

Intelligent Sensor-Actuator-Systems Laboratory
Institute for Anthropomatics and Robotics
Karlsruhe Institute of Technology (KIT), Germany
benjamin.noack@ieee.org, christopher.funk@kit.edu,
susanne.radtke@kit.edu, uwe.hanebeck@ieee.org

Abstract: Event-based communication and state estimation offer the potential to improve resource utilization in networked sensor and control systems significantly. Sensor nodes can trigger transmissions when data are deemed useful for the remote estimation units. To improve the estimation performance, the remote estimator can exploit the implicit information conveyed by the event trigger even if no transmission is triggered. The implicit information is typically incorporated into the measurement update of a remote Kalman filter. In this paper, event-triggered transmissions of input data are investigated that enter the prediction step of the remote estimator. By employing a stochastic trigger, the implicit input information remains Gaussian and can easily be incorporated into the remote Kalman filter. The proposed event-based scheme is evaluated in remote tracking scenarios, where system inputs are transmitted aperiodically.

Keywords: Kalman Filtering, Event-based Estimation, Stochastic Triggering, Networked Estimation, Gaussianity-preserving Triggers.

1. INTRODUCTION

Cheap integrated sensors have widely permeated industries, from consumer products over transportation systems to manufacturing processes. Sensor data can be acquired ubiquitously and pervasively through wide-area networks. The transmission and processing of sensor data are closely interrelated aspects of effective resource allocation in wireless networks. Recent trends in networked data processing indicate a paradigm shift from time-periodic to data-driven or event-based transmission schedules. In this regard, event-based state estimation techniques have become a major subject of current research endeavors (Sijs and Lazar, 2012; Shi et al., 2016; Battistelli et al., 2018). The underlying mechanism to trigger an event evaluates how useful current sensor readings are for the remote estimation unit. Sijs and Lazar (2012) propose the use of send-on-delta triggering rules, and Trimpe and D'Andrea (2014) employ event triggers that rely on the predicted error variance of the measurements. As discussed by Wu et al. (2013), finding a trade-off between estimation quality and communication rate is decisive for the effectiveness of event-based estimation. For this reason, Molin et al. (2015) consider a joint objective function that comprises both the estimation error and a communication penalty. Trimpe (2014) and Bian et al. (2018) discuss the estimation stability under event-based transmissions in more detail. Sijs and Noack (2017) study the impact of imperfect communication links on the estimation quality at the remote estimation system. Solutions to such open issues are key to leverage the full potential of event-based estimation for resource-efficient sensor data processing.

The gist of event-based estimation is the design of event-triggering criteria that are the tool to mediate between energy saving and estimation accuracy. Sijs et al. (2014) study different event-triggering criteria with respect to estimation errors and communication resources. Lowering the communication rate does not imply a proportional degradation of the estimation quality. On the contrary, the event trigger itself conveys information about the data and can be exploited by the remote estimation system even when no transmission has been triggered. More precisely, Sijs et al. (2013, 2015) have demonstrated that the remote estimator can interpret the absence of sensor data as implicit—also called negative—measurement information. The estimator can hence perform time-periodic measurement updates (Sijs and Lazar, 2012) though sensor data are transmitted aperiodically. The herein used implicit information assumes a deterministic trigger design, which features a set-membership representation of the actual sensor signal. Sijs et al. (2013, 2015) and Shi et al. (2016) apply hybrid state estimators to incorporate the set-membership implicit information. These hybrid concepts embody an amalgamation of Kalman filtering with ellipsoidal calculus (Noack et al., 2012). Such bounding techniques are also exploited by Dormann et al. (2018) and Battistelli et al. (2018) in distributed Kalman filtering to derive data-driven fusion algorithms. However, the inherent set-membership nature of deterministic triggers calls for a more complex estimator design.

A viable alternative to deterministic triggers is a stochastic trigger design that allows for a Gaussian representation of the implicit measurement information. Han et al. (2015) study open- and closed-loop trigger mechanisms, where the latter requires the remote system to feed back its estimates to the sensor. Weerakkody et al. (2016) extend event-based estimation to multi-sensor data fusion with stochastic triggers, and Mohammadi and Plataniotis (2017) provide a Gaussianity-preserving formulation in the information form of the Kalman filter. Andrén and Cervin (2016) let the sensor predict the system behavior to further reduce the communication rate without information loss at the receiver. The trigger decision proposed by Wu et al. (2016) rests upon a local Kalman filter, which compares the current estimate with the prediction from the last event. Schmitt et al. (2019) employ an FIR filter for the local prediction at the sensor, which is more resilient towards outliers and initialization errors.

Research on event-based estimation has in common that the remote estimator incorporates implicit information in the measurement update of a Kalman filter or related filtering concepts. For this purpose, Andrén and Cervin (2016) express the measurement update in terms of probability density functions, and implicit measurements relate to the likelihoods used in Bayes' theorem. The purpose of this paper is to introduce event-based estimation for sensor readings that are used in the prediction step of a Kalman filter. An example is an external tracking system for a mobile robotic platform; the onboard IMU provides data at a high rate that needs to be accessed by the remote tracking system for state prediction. To reduce the communication rate between platform and tracking system, we examine the use of stochastic trigger mechanisms. In the prediction step of the remote Kalman filter, implicit information about the input data, e.g., IMU measurements, can be exploited when no transmission occurs. We point out that implicit input information can easily be incorporated into a linear Kalman filter due to the stochastic trigger design. The proposed event-based estimator is also evaluated in two tracking scenarios. In both scenarios, a mobile platform transmits its sensor readings of forces driving the system to a remote tracking system.

The organization and contributions of this paper can be summarized as follows: Our goal is to model implicit input information as measurements of the actual input. For this purpose, we first study how measured inputs enter the time update of the Kalman filter. In particular, the best linear unbiased estimate of the actual input is used, and the noise characteristics of the measured input represent an additional error covariance matrix. Second, we discuss how the implicit information retrieved from the stochastic event trigger is translated into a measurement of the actual input, which can then be exploited in the time update according to the preceding considerations.

2. GAUSSIANITY-PRESERVING TRIGGERING

This section introduces the notations used throughout this paper and gives an overview of Gaussianity-preserving trigger designs.

2.1 System Model & State Estimation

Discrete-time linear system and measurement models are considered, which are governed by

$$\underline{\mathbf{x}}_{k+1} = \mathbf{A}_k\, \underline{\mathbf{x}}_k + \mathbf{B}_k\, \underline{\mathbf{u}}_k + \underline{\mathbf{w}}_k \;, \quad (1)$$

$$\underline{\mathbf{z}}_{k+1} = \mathbf{C}_{k+1}\, \underline{\mathbf{x}}_{k+1} + \underline{\mathbf{v}}_{k+1} \;, \quad (2)$$

where $\underline{\mathbf{x}}_k \in \mathbb{R}^{n_x}$ is the state at time step $k \in \mathbb{N}$, and $\underline{\mathbf{z}}_{k+1} \in \mathbb{R}^{n_z}$ denotes the observation. The time-variant process and measurement matrices are given by $\mathbf{A}_k \in \mathbb{R}^{n_x \times n_x}$ and $\mathbf{C}_{k+1} \in \mathbb{R}^{n_z \times n_x}$, respectively. The process noise $\underline{\mathbf{w}}_m \sim \mathcal{N}(\underline{0}, \mathbf{Q}_m)$ and measurement noise $\underline{\mathbf{v}}_n \sim \mathcal{N}(\underline{0}, \mathbf{R}_n)$ are white and mutually uncorrelated for arbitrary $m, n \in \mathbb{N}$. Inputs $\underline{\mathbf{u}}_k \in \mathbb{R}^{n_u}$ affect (1) through $\mathbf{B}_k \in \mathbb{R}^{n_x \times n_u}$. To estimate the state, a discrete-time Kalman filter is considered. The time update or prediction step yields

$$\hat{\underline{\mathbf{x}}}_{k+1|k} = \mathbf{A}_k\, \hat{\underline{\mathbf{x}}}_{k|k} + \mathbf{B}_k\, \underline{\mathbf{u}}_k \;, \quad (3)$$

$$\mathbf{P}_{k+1|k} = \mathbf{A}_k\, \mathbf{P}_{k|k}\, \mathbf{A}_k^{\mathrm{T}} + \mathbf{Q}_k \;, \quad (4)$$

where $\mathbf{P}_{k+1|k}$ is the covariance of the estimation error $\hat{\underline{\mathbf{x}}}_{k+1|k} - \underline{\mathbf{x}}_{k+1}$. The measurement update with the observation $\underline{\mathbf{z}}_{k+1}$ is obtained by

$$\hat{\underline{\mathbf{x}}}_{k+1|k+1} = \hat{\underline{\mathbf{x}}}_{k+1|k} + \mathbf{K}_{k+1}\left(\underline{\mathbf{z}}_{k+1} - \mathbf{C}_{k+1}\, \hat{\underline{\mathbf{x}}}_{k+1|k}\right) , \quad (5)$$

$$\mathbf{P}_{k+1|k+1} = \left(\mathbf{I} - \mathbf{K}_{k+1}\, \mathbf{C}_{k+1}\right) \mathbf{P}_{k+1|k} \;, \quad (6)$$

$$\mathbf{K}_{k+1} = \mathbf{P}_{k+1|k}\, \mathbf{C}_{k+1}^{\mathrm{T}} \Big(\mathbf{C}_{k+1} \mathbf{P}_{k+1|k}\, \mathbf{C}_{k+1}^{\mathrm{T}} + \mathbf{R}_{k+1}\Big)^{-1} ,$$

where $\mathbf{K}_{k+1}$ is the Kalman gain. The zero-mean error $\hat{\underline{\mathbf{x}}}_{k+1|k+1} - \underline{\mathbf{x}}_{k+1}$ has the covariance matrix $\mathbf{P}_{k+1|k+1}$.

2.2 Stochastic Triggering

Let $\underline{\mathbf{y}}_k \in \mathbb{R}^{n_y}$ represent the sensor data for which a trigger decision is to be made. The variable $\gamma_k = 1$ denotes that an event is triggered, and $\underline{\mathbf{z}}_k$ is sent to the receiver. For $\gamma_k = 0$, no transmission is triggered. To determine γ_k, an independently and identically distributed random variable $\boldsymbol{\xi}_k$ is generated, which is uniformly distributed over $[0, 1]$. The decision scheme is given by

$$\gamma_k = \begin{cases} 1, & \boldsymbol{\xi}_k > \phi(\underline{\mathbf{y}}_k - \underline{c}_k)\,, \\ 0, & \boldsymbol{\xi}_k \leq \phi(\underline{\mathbf{y}}_k - \underline{c}_k)\,, \end{cases}$$

with $\phi(\underline{\mathbf{y}}_k - \underline{c}_k) = \exp\left(-\frac{1}{2}\,(\underline{\mathbf{y}}_k - \underline{c}_k)^{\mathrm{T}}\, \mathbf{Z}_k^{-1}\, (\underline{\mathbf{y}}_k - \underline{c}_k)\right)$ to compare $\underline{\mathbf{y}}_k$ against a chosen $\underline{c}_k \in \mathbb{R}^{n_y}$. The matrix $\mathbf{Z}_k$ is a design parameter to tune the transmission rate. Due to the design of $\phi(\cdot)$ and the properties of $\boldsymbol{\xi}_k$, the transmission probability given $\underline{\mathbf{y}}_k$ yields

$$\Pr\big\{\gamma_k = 1 \,\big|\, \underline{\mathbf{y}}_k\big\} = 1 - \phi(\underline{\mathbf{y}}_k - \underline{c}_k)\,,$$
$$\Pr\big\{\gamma_k = 0 \,\big|\, \underline{\mathbf{y}}_k\big\} = \phi(\underline{\mathbf{y}}_k - \underline{c}_k)\,. \quad (7)$$

The likelihood $\Pr\big\{\gamma_k = 0 \,\big|\, \underline{\mathbf{y}}_k\big\}$ is exploited in the following to infer information about $\underline{\mathbf{y}}_k$ when the sensor does not trigger a transmission.

3. TRIGGERING ON INPUT INFORMATION

Stochastic triggering has been investigated against the background of the measurement update in event-based Kalman filtering. In many localization and navigation tasks, inputs are noisy sensor readings, and their noise characteristics are modeled as part of the process noise. The input information is often provided by onboard sensors like inertial measurement units (IMU) and needs to be transmitted to remote systems if an external tracker is used. For the sensor readings of inputs, a corresponding measurement model

$$\underline{\mathbf{z}}_k^{(u)} = \mathbf{C}_k^{(u)}\, \underline{\mathbf{u}}_k + \underline{\mathbf{v}}_k^{(u)} \tag{8}$$

is considered, where $\mathbf{C}_k^{(u)}$ is the measurement mapping with $\mathrm{rank}(\mathbf{C}_k^{(u)}) = n_u$ and $\underline{\mathbf{v}}_k^{(u)} \sim \mathcal{N}(\underline{0}, \mathbf{R}_k^{(u)})$ is an additive noise term. The mapping $\mathbf{C}_k^{(u)}$ is typically assumed to be the identity matrix if the input is measured directly. In the following, we examine how $\underline{\mathbf{z}}_k^{(u)}$ is incorporated in the prediction step and how implicit information on $\underline{\mathbf{z}}_k^{(u)}$ can be exploited between triggering events.

3.1 Estimates of Uncertain Inputs

For the typical case $\mathbf{C}_k^{(u)} = \mathbf{I}$, the intuitive and common way to incorporate measured inputs is to interpret them directly as inputs in (1) and the measurement noise as part of the process noise $\underline{\mathbf{w}}_k$. To extend this intuition to implicit input information, this subsection sheds more light on the estimation of and with uncertain inputs. Such problems have been studied against the background of unknown biases affecting the system. Friedland (1969) has suggested to augment the state vector by the bias term and formulated the corresponding filtering equations. The measurement equations considered therein rely upon a feedthrough of the bias term. Gillijns and Moor (2007) consider the problem of simultaneous input and state estimation, which they tackle by a best linear unbiased estimate (BLUE) of the uncertain inputs and which corresponds to a weighted least-squares estimate. Like in the case of bias estimation, the inputs are supposed to be unknown quantities that need to be inferred from the state observations in (2). Bai et al. (2018) estimate time-varying nonlinear uncertain dynamics affecting the process model by using the state observations and an augmented state representation. These concepts strive for the simultaneous estimation of input and state information.

In the case considered in this paper, measurements of input information are available through (8). For this reason, we can directly infer estimates of the actual inputs, which circumvents the need for simultaneous input and state estimation. We also see that the input estimate naturally arises out of the best linear predictor.

By exploiting the measurement equation (8), the BLUE of $\underline{\mathbf{u}}_k$ yields

$$\hat{\underline{\mathbf{u}}}_k = \mathbf{L}_k\, \underline{\mathbf{z}}_k^{(u)} \tag{9}$$

with covariance matrix

$$\mathbf{P}_k^{(u)} = \mathbf{L}_k\, \mathbf{R}_k^{(u)}\, \mathbf{L}_k^{\mathrm{T}} = \left(\big(\mathbf{C}_k^{(u)}\big)^{\mathrm{T}} \big(\mathbf{R}_k^{(u)}\big)^{-1} \mathbf{C}_k^{(u)} \right)^{-1} \tag{10}$$

and gain matrix

$$\mathbf{L}_k = \left(\big(\mathbf{C}_k^{(u)}\big)^{\mathrm{T}} \big(\mathbf{R}_k^{(u)}\big)^{-1} \mathbf{C}_k^{(u)} \right)^{-1} \big(\mathbf{C}_k^{(u)}\big)^{\mathrm{T}} \big(\mathbf{R}_k^{(u)}\big)^{-1}, \tag{11}$$

which represents a weighted least squares estimate of $\underline{\mathbf{u}}_k$. For $\mathbf{C}_k^{(u)} = \mathbf{I}$, the estimate simplifies to $\hat{\underline{\mathbf{u}}}_k = \underline{\mathbf{z}}_k^{(u)}$ with $\mathbf{P}_k^{(u)} = \mathbf{R}_k^{(u)}$. The input estimate $\hat{\underline{\mathbf{u}}}_k$ is used in the prediction formula (3) in place of the actual input while $\mathbf{P}_k^{(u)}$ enters (4) as an additional noise covariance. This result can also be obtained directly by striving for a best linear prediction given $\hat{\underline{\mathbf{u}}}_k$, which is discussed in Appendix A. Either way, the prediction step of the Kalman filter is altered to

$$\hat{\underline{\mathbf{x}}}_{k+1|k} = \mathbf{A}_k\, \hat{\underline{\mathbf{x}}}_{k|k} + \mathbf{B}_k\, \mathbf{L}_k\, \underline{\mathbf{z}}_k^{(u)} \overset{(9)}{=} \mathbf{A}_k\, \hat{\underline{\mathbf{x}}}_{k|k} + \mathbf{B}_k\, \hat{\underline{\mathbf{u}}}_k\,, \tag{12}$$

$$\mathbf{P}_{k+1|k} = \mathbf{A}_k\, \mathbf{P}_{k|k}\, \mathbf{A}_k^{\mathrm{T}} + \mathbf{Q}_k^{'} \tag{13}$$

with

$$\mathbf{Q}_k^{'} = \mathbf{B}_k\, \mathbf{P}_k^{(u)}\, \mathbf{B}_k^{\mathrm{T}} + \mathbf{Q}_k\,,$$

and the input estimate $\hat{\underline{\mathbf{u}}}_k$ is directly used in the prediction.

By introducing an event-based trigger mechanism at the sensor, the measurements in (8) are only available at event times. In the following subsection, we discuss the input estimation in the case event-triggered Kalman filtering. In particular, input estimates need to be retrieved from the implicit information conveyed by the trigger decision when not event occurs.

3.2 Kalman Filtering with Implicit Input Information

The use of event triggers implies that two cases need to be distinguished. Either the input measurement is available for state prediction, i.e., $\gamma_k = 1$, or no event has been triggered, i.e., $\gamma_k = 0$, and the estimator is ignorant about the actual measurement $\underline{\mathbf{z}}_k^{(u)}$.

The first case ($\gamma_k = 1$) has been discussed in the previous subsection, and $\underline{\mathbf{z}}_k^{(u)}$ is used directly. The corresponding likelihood yields

$$\Pr\big\{\underline{\mathbf{z}}_k^{(u)} \,\big|\, \underline{\mathbf{u}}_k\big\} = \frac{1}{\sqrt{(2\pi)^{n_z} \det\big(\mathbf{R}_k^{(u)}\big)}} \cdot \exp\left(-\frac{1}{2} \big(\underline{\mathbf{z}}_k^{(u)} - \mathbf{C}_k^{(u)}\, \underline{\mathbf{u}}_k\big)^{\mathrm{T}} \big(\mathbf{R}_k^{(u)}\big)^{-1} \big(\underline{\mathbf{z}}_k^{(u)} - \mathbf{C}_k^{(u)} \underline{\mathbf{u}}_k\big) \right)$$

which encompasses the probabilistic representation of the measurement model (8). More precisely, the input estimate (9) with covariance (10) is the maximum likelihood (ML) estimate for $\Pr\big\{\underline{\mathbf{z}}_k^{(u)} \,\big|\, \underline{\mathbf{u}}_k\big\}$.

In the second case ($\gamma_k = 0$), we can infer information about the input if the estimator is aware of the trigger likelihood (7), i.e.,

$$\Pr\big\{\gamma_k = 0 \,\big|\, \underline{\mathbf{z}}_k^{(u)}\big\} = \phi(\underline{\mathbf{z}}_k^{(u)} - \underline{c}_k)$$

for chosen triggering constant $\underline{c}_k$. As a result,

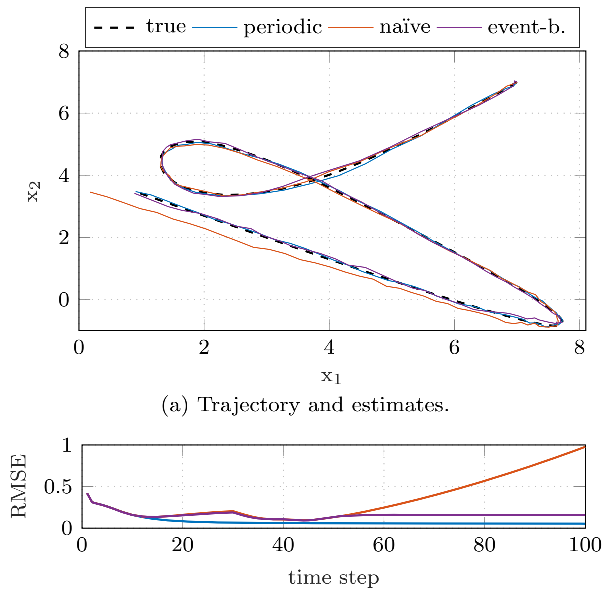
$$\begin{aligned}
\Pr\left\{\gamma_k = 0 \,\middle|\, \underline{\mathbf{u}}_k\right\} &= \int_{\mathbb{R}^{n_z}} \Pr\left\{\gamma_k = 0, \underline{\mathbf{z}}_k^{(u)} \,\middle|\, \underline{\mathbf{u}}_k\right\} \mathrm{d}\, \underline{\mathbf{z}}_k^{(u)} \\
&= \int_{\mathbb{R}^{n_z}} \Pr\left\{\gamma_k = 0 \,\middle|\, \underline{\mathbf{z}}_k^{(u)}, \underline{\mathbf{u}}_k\right\} \cdot \Pr\left\{\underline{\mathbf{z}}_k^{(u)} \,\middle|\, \underline{\mathbf{u}}_k\right\} \mathrm{d}\, \underline{\mathbf{z}}_k^{(u)} \\
&= a \cdot \int_{\mathbb{R}^{n_z}} \exp\left(-\frac{1}{2} \big(\underline{\mathbf{z}}_k^{(u)} - \underline{c}_k\big)^{\mathrm{T}} \mathbf{Z}_k^{-1} \big(\underline{\mathbf{z}}_k^{(u)} - \underline{c}_k\big) \right) \cdot \\
&\quad \exp\left(-\frac{1}{2} \big(\underline{\mathbf{z}}_k^{(u)} - \mathbf{C}_k^{(u)}\, \underline{\mathbf{u}}_k\big)^{\mathrm{T}} \big(\mathbf{R}_k^{(u)}\big)^{-1} \big(\underline{\mathbf{z}}_k^{(u)} - \mathbf{C}_k^{(u)} \underline{\mathbf{u}}_k\big) \right) \mathrm{d}\, \underline{\mathbf{z}}_k^{(u)} \\
&= a \cdot \exp\left(-\frac{1}{2} \big(\underline{c}_k - \mathbf{C}_k^{(u)}\, \underline{\mathbf{u}}_k\big)^{\mathrm{T}} \big(\mathbf{Z}_k + \mathbf{R}_k^{(u)}\big)^{-1} \big(\underline{c}_k - \mathbf{C}_k^{(u)}\, \underline{\mathbf{u}}_k\big) \right)
\end{aligned}$$

can be computed with constant terms being summarized in a and by exploiting that γ_k is independent of $\underline{\mathbf{u}}_k$ given $\underline{\mathbf{z}}_k^{(u)}$. Accordingly, an ML estimate can be computed for $\Pr\left\{\gamma_k = 0 \,\middle|\, \underline{\mathbf{u}}_k\right\}$ that yields

$$\hat{\underline{\mathbf{u}}}_k^{\gamma} = \mathbf{L}_k^{\gamma}\, \underline{c}_k \tag{14}$$

with covariance matrix

$$\begin{aligned}
\mathbf{P}_k^{\gamma} &= \mathbf{L}_k^{\gamma} \left(\mathbf{Z}_k + \mathbf{R}_k^{(u)}\right) \left(\mathbf{L}_k^{\gamma}\right)^{\mathrm{T}} \\
&= \left(\big(\mathbf{C}_k^{(u)}\big)^{\mathrm{T}} \big(\mathbf{Z}_k + \mathbf{R}_k^{(u)}\big)^{-1} \mathbf{C}_k^{(u)} \right)^{-1}
\end{aligned} \tag{15}$$



(a) Trajectory and estimates.

(b) Root-mean-squared error.

Fig. 1. Simulation results for near-constant-velocity model.

and gain matrix

$$\mathbf{L}_k^\gamma = \left((\mathbf{C}_k^{(u)})^\mathrm{T} (\mathbf{Z}_k + \mathbf{R}_k^{(u)})^{-1} \mathbf{C}_k^{(u)} \right)^{-1} \cdot (\mathbf{C}_k^{(u)})^\mathrm{T} (\mathbf{Z}_k + \mathbf{R}_k^{(u)})^{-1} .$$

The estimate (14) is inferred from the triggering variable $\underline{c}_k$. The send-on-delta trigger scheme, for instance, sets $\underline{c}_k$ to the last received $\underline{\mathbf{z}}_{k_\text{last}}^{(u)}$ during the most recent triggering event. It can also be observed that $\mathbf{P}_k^\gamma - \mathbf{P}_k^{(u)} \geq \mathbf{0}$ is positive semi-definite, i.e., $\mathbf{Z}_k$ in (15) inflates the covariance matrix. Between triggering events, the resulting formulas for the prediction step are then given by

$$\hat{\underline{\mathbf{x}}}_{k+1|k} \overset{(14)}{=} \mathbf{A}_k \, \hat{\underline{\mathbf{x}}}_{k|k} + \mathbf{B}_k \, \hat{\underline{\mathbf{u}}}_k^\gamma , \qquad (16)$$

$$\mathbf{P}_{k+1|k} = \mathbf{A}_k \, \mathbf{P}_{k|k} \, \mathbf{A}_k^\mathrm{T} + \mathbf{Q}_k^\gamma \qquad (17)$$

with

$$\mathbf{Q}_k^\gamma = \mathbf{B}_k \, \mathbf{P}_k^\gamma \, \mathbf{B}_k^\mathrm{T} + \mathbf{Q}_k , \qquad (18)$$

i.e., the input estimate (14) with covariance (15) is used.

Remark 1: In the linear Gaussian case, the input estimates can be inferred as ML estimates either from the input measurements $\underline{\mathbf{z}}_k^{(u)}$ if available or from the implicit information $\gamma_k = 0$ when no transmission is triggered. The derived event-based prediction bears a strong resemblance with the commonly considered event-based filtering step, where implicit information is exploited in the update equations (5) and (6). It is worth to emphasize that implicit information is a key ingredient of the prediction step when no transmission event is triggered: Complete ignorance about the actual input $\underline{\mathbf{u}}_k$ implies *infinite* uncertainty $\mathbf{Q}_k^\gamma$, which renders the prediction useless. Strictly speaking, the prediction (16) and (17) cannot be carried out without any information that could be inferred about the actual input $\underline{\mathbf{u}}_k$. In general, assumptions about the input forces driving the system are made and are a strong argument in favor of implicit information being exploited in event-based estimation.

Remark 2: Detailed studies on communication rate, stability, and convergence are left to future work. However, the corresponding analyses can be adopted from Han et al. (2015), i.e., the behavior of event-based input triggering resembles the measurement update with implicit measurements. For instance, a lower and upper bound on $\mathbf{P}_{k|k}$ can be derived by designing Kalman filters with error dynamics according to (4) and (17), respectively.

4. SIMULATIONS

To evaluate the proposed concept, a send-on-delta trigger mechanism as used by Schmitt et al. (2019) is studied. In both evaluation scenarios, the trigger variable $\underline{c}_k$ is set to the last transmitted input $\underline{\mathbf{u}}_{k_\text{last}}$ at time step k_last. The event-based approach is compared against a periodic transmission of input data and a naïve approach, which also relies on the event-based transmission schedule. Due to the send-on-delta scheme, the naïve and event-based estimator design resemble each other in that they both use the same trigger decision and hold $\underline{\mathbf{u}}_{k_\text{last}}$ when no transmission is triggered. In doing so, they have the same transmission rate. However, they differ in that the naïve approach does not exploit the implicit information. Both simulations confirm that the implicit information significantly contributes to improving the estimation quality.

4.1 Near-Constant-Velocity Model

In the first example, a near-constant-velocity model with the system matrices

$$\mathbf{A}_k = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{B}_k = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \mathbf{Q}_k = \begin{bmatrix} \frac{\Delta t^4}{4} & 0 & \frac{\Delta t^3}{2} & 0 \\ 0 & \frac{\Delta t^4}{4} & 0 & \frac{\Delta t^3}{2} \\ \frac{\Delta t^3}{2} & 0 & \Delta t^2 & 0 \\ 0 & \frac{\Delta t^3}{2} & 0 & \Delta t^2 \end{bmatrix},$$

and initial state $\underline{\mathbf{x}}_0 = [7, 7]^\mathrm{T}$ is considered. The measurement model (2) is defined by the matrices

$$\mathbf{C}_k = \left[\begin{smallmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{smallmatrix}\right], \, \mathbf{R}_k = 0.01 \, \mathbf{I} .$$

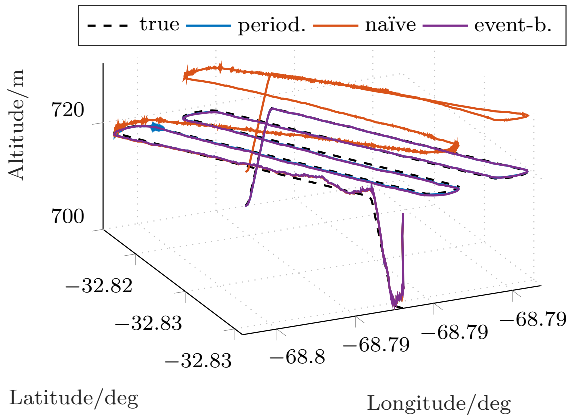
The velocity inputs $\underline{\mathbf{u}}_k$ are computed such that the state trajectory follows the dashed line in Fig. 1(a). The same figure depicts the estimation results in the periodic, naïve, and event-based case. In the second and third case, the estimators are only aware of the velocities when events are triggered. In this example, the velocities $\underline{\mathbf{u}}_k$ are directly accessed, i.e., $\mathbf{P}_k^{(u)} = \mathbf{0}$. The trigger matrix is set to $\mathbf{Z}_k = 0.2 \, \mathbf{I}$. Hence, the noise matrix (18) used by the event-based estimator becomes

$$\mathbf{Q}_k^\gamma = \mathbf{B}_k \, \mathbf{Z}_k \, \mathbf{B}_k^\mathrm{T} + \mathbf{Q}_k$$

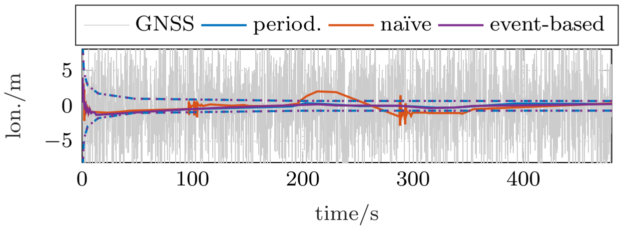
to model the implicit information when no transmission is triggered. Fig. 1(b) depicts the root-mean-squared error (RMSE) for 10 000 Monte-Carlo simulations. The event-based case provides a slightly higher error than the periodic case but outperforms the naïve case. The event trigger lowers the communication rate to 38 % of the periodic case.

4.2 Simulation of Integrated Navigation System

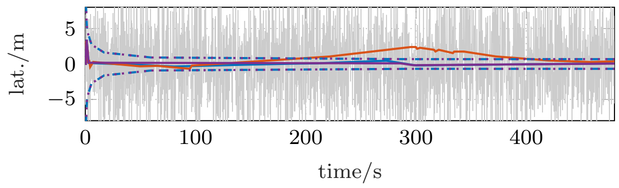
The second evaluation scenario uses the simulation framework NaveGo for navigation systems developed by Gonzalez et al. (2015). To investigate event-based transmission schemes, the INS/GNSS example based on synthetic data provided in the file `navego_example_synth.m` has been extended. The accelerations and turn rates measured by



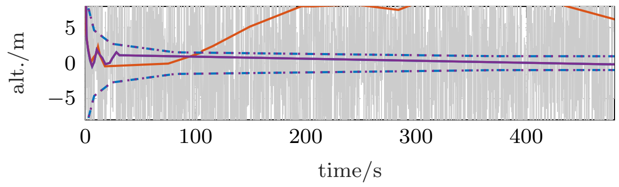
(a) Actual and estimated trajectories.



(b) Longitude errors.



(c) Latitude errors.



(d) Altitude errors.

Fig. 2. INS/GNSS navigation example with ADIS16405 IMU error profile. 3σ-bounds for the periodic and event-based case are shown as dashed lines.

the simulated onboard IMU in the body frame are each three-dimensional vectors. For the six-dimensional joint vector, a trigger for event-based transmissions is defined with

$$\mathbf{Z}_k = \begin{bmatrix} \mathbf{Z}^{\text{acc}} & \mathbf{0} \\ \mathbf{0} & \mathbf{Z}^{\text{rate}} \end{bmatrix}, \quad \mathbf{Z}^{\text{acc}} = 0.008\,\mathbf{I}, \quad \mathbf{Z}^{\text{rate}} = 0.004\,\mathbf{I}\,.$$

For all other parameters, the predefined settings in the NaveGo simulation framework have been adopted, which also provides error profiles of an ADIS16405 IMU. Details and parameter settings can be found in Gonzalez et al. (2019). The framework employs an extended Kalman filter, which is a complementary filter as proposed by Farrell (2008). The error state representation comprises 15 state components.

The extended scenario using event-based estimation assumes that the object is tracked by a remote estimation system that receives the IMU data from the object and fuses the data with GNSS measurements. Again, a send-on-delta scheme with trigger matrix $\mathbf{Z}_k$ is used to trigger transmissions from the object to the remote tracker. For the ADIS16405 error profile, the estimation results and errors are depicted in Fig. 2, where only 33 % of the measurements have been transmitted. We have also conducted simular experiments with the parameters of a tactical grade ADIS16488 IMU, where even rates below 10 % can be achieved. It should be noticed that the simulated data is rather smooth and higher rates are expected in real systems. Experiments with a robotic platform and real data will be part of future work. The event-based estimator exploiting the implicit information conveyed by the send-on-delta trigger reaches a similar estimation quality as the periodic filter design. The send-on-delta trigger implies using the last received input when no event is triggered. This strategy is also pursued by the naïve estimator. However, it does not exploit the correct error covariance (17), which is related to the implicit information. The incorrect error covariance leads to performance degradation of the naïve approach as it can be seen, e.g., in Fig. 2(b). With a more sophisticated trigger design, e.g., a predictive trigger, the event-based estimator should further outperform the naïve approach at an even lower transmission rate, which will be part of prospective research.

5. CONCLUSION & OUTLOOK

The commonly considered measurement update with event-triggered sensor data has been extended to time updates with event-triggered input data. When no transmission is triggered, the remote Kalman filter can infer implicit information about the input data. Due to the stochastic trigger design, the design matrix $\mathbf{Z}_k$ appears as additional process noise in the prediction step. The evaluation scenarios underpin the effectiveness of the proposed stochastic input triggers in reducing the communication rate without significantly impairing the estimation quality. Further research will be dedicated to a more sophisticated trigger design that equips the sensor with a predictive assessment of the sensor signal. In doing so, the transmission rate can be further reduced. Unreliable communication links and estimator properties like stability and convergenge rate also need to be addressed. For the considered scenarios, a validation against real data is necessary. Also, a combination with advanced pre-integration techniques, as proposed by Forster et al. (2017), may represent a promising research direction.

REFERENCES

Andrén, M.T. and Cervin, A. (2016). Event-Based State Estimation Using an Improved Stochastic Send-on-Delta Sampling Scheme. In *Proc. 2016 2nd Int. Conf. on Event-based Control, Communication, and Signal Processing (EBCCSP 2016)*. Krakow, Poland.

Bai, W., Xue, W., Huang, Y., and Fang, H. (2018). On Extended State Based Kalman Filter Design for a Class of Nonlinear Time-varying Uncertain Systems. *Science China Information Sciences*, 61(4), 1–16.

Battistelli, G., Chisci, L., and Selvi, D. (2018). A Distributed Kalman Filter with Event-Triggered Communication and Guaranteed Stability. *Automatica*, 93, 75–82.

Bian, X., Li, X.R., and Jilkov, V.P. (2018). Remote State Estimation with Data-driven Communication and Guaranteed Stability. In *Proc. 21st Int. Conf. on Inform. Fusion (Fusion 2018)*. Cambridge, UK.

Dormann, K., Noack, B., and Hanebeck, U.D. (2018). Optimally Distributed Kalman Filtering with Data-Driven Communication. *Sensors*, 18(4).

Farrell, J.A. (2008). *Aided Navigation: GPS with High Rate Sensors*. McGraw-Hill Education Ltd.

Forster, C., Carlone, L., Dellaert, F., and Scaramuzza, D. (2017). On-Manifold Preintegration for Real-Time Visual–Inertial Odometry. *IEEE Trans. on Robotics*, 33(1), 1–21.

Friedland, B. (1969). Treatment of Bias in Recursive Filtering. *IEEE Trans. on Autom. Control*, 14(4).

Gillijns, S. and Moor, B.D. (2007). Unbiased Minimum-variance Input and State Estimation for Linear Discrete-time Systems with Direct Feedthrough. *Automatica*, 43(5), 934–937.

Gonzalez, R., Catania, C., and Dabove, P. (2019). NaveGo: An Open-Source MATLAB/GNU-Octave Toolbox for Processing Integrated Navigation Systems and Performing Inertial Sensors Profiling Analysis. Version 1.2. URL: `https://github.com/rodralez/NaveGo`.

Gonzalez, R., Giribet, J.I., and Patiño, H.D. (2015). NaveGo: A Simulation Framework for Low-Cost Integrated Navigation Systems. *Journal of Control Engineering and Applied Informatics*, 17(2), 110–120.

Han, D., Mo, Y., Wu, J., Weerakkody, S., Sinopoli, B., and Shi, L. (2015). Stochastic Event-Triggered Sensor Schedule for Remote State Estimation. *IEEE Trans. on Autom. Control*, 60(10), 2661–2675.

Mohammadi, A. and Plataniotis, K.N. (2017). Event-Based Estimation With Information-Based Triggering and Adaptive Update. *IEEE Trans. on Signal Processing*, 65(18), 4924–4939.

Molin, A., Sandberg, H., and Johansson, K.H. (2015). Consistency-Preserving Event-Triggered Estimation in Sensor Networks. In *Proc. 54th IEEE Conf. on Decision and Control (CDC 2015)*, 7494–7501. Osaka, Japan.

Noack, B., Pfaff, F., and Hanebeck, U.D. (2012). Optimal Kalman Gains for Combined Stochastic and Set-Membership State Estimation. In *Proc. of the 51st IEEE Conf. on Dec. and Control (CDC 2012)*. Maui, HI, USA.

Schmitt, E.J., Noack, B., Krippner, W., and Hanebeck, U.D. (2019). Gaussianity-Preserving Event-Based State Estimation with an FIR-Based Stochastic Trigger. *IEEE Control Systems Letters*, 3(3), 769–774.

Shi, D., Shi, L., and Chen, T. (2016). *Event-Based State Estimation: A Stochastic Perspective*, volume 41 of *Studies in Systems, Decision and Control*. Springer International Publishing, Cham.

Sijs, J., Kester, L., and Noack, B. (2014). A Study on Event Triggering Criteria for Estimation. In *Proc. of the 17th Intern. Conf. on Inform. Fusion (Fusion 2014)*. Salamanca, Spain.

Sijs, J. and Lazar, M. (2012). Event Based State Estimation with Time Synchronous Updates. *IEEE Trans. on Autom. Control*, 57(10), 2650–2655.

Sijs, J. and Noack, B. (2017). Event-based State Estimation in a Feedback Loop with Imperfect Communication Links. In *Proceedings of the 20th IFAC World Congress (IFAC 2017)*. Toulouse, France.

Sijs, J., Noack, B., and Hanebeck, U.D. (2013). Event-based State Estimation with Negative Information. In *Proc. of the 16th Intern. Conf. on Inform. Fusion (Fusion 2013)*. Istanbul, Turkey.

Sijs, J., Noack, B., Lazar, M., and Hanebeck, U.D. (2015). Event-based control and signal processing. chapter Time-Periodic State Estimation with Event-Based Measurement Updates, 261–279. CRC Press.

Trimpe, S. (2014). Stability Analysis of Distributed Event-Based State Estimation. In *Proc. 53rd IEEE Conf. Dec. and Control (CDC 2014)*. Los Angeles, CA, USA.

Trimpe, S. and D'Andrea, R. (2014). Event-Based State Estimation with Variance-Based Triggering. *IEEE Trans. on Autom. Control*, 59(12), 3266–3281.

Weerakkody, S., Mo, Y., Sinopoli, B., Han, D., and Shi, L. (2016). Multi-Sensor Scheduling for State Estimation With Event-Based, Stochastic Triggers. *IEEE Trans. on Autom. Control*, 61(9), 2695–2701.

Wu, J., Jia, Q.S., Johansson, K.H., and Shi, L. (2013). Event-Based Sensor Data Scheduling: Trade-Off Between Communication Rate and Estimation Quality. *IEEE Trans. on Autom. Control*, 58(4), 1041–1046.

Wu, J., Ren, X., Han, D., Shi, D., and Shi, L. (2016). Finite-Horizon Gaussianity-Preserving Event-Based Sensor Scheduling in Kalman Filter Applications. *Automatica*, 72, 100–107.

Appendix A. PREDICTION OF STATE ESTIMATES WITH INPUT MEASUREMENTS

The prediction step in (3) and (4) incorporates possibly noisy inputs $\underline{\mathbf{u}}_k$. The following considerations briefly outline how the prediction step can be directly derived for measurements $\underline{\mathbf{z}}_k^{(u)}$ of $\underline{\mathbf{u}}_k$ according to eq. (8). In the time update of the Kalman filter, we strive for finding the best unbiased prediction

$$\hat{\underline{\mathbf{x}}}_{k+1|k} = \mathbf{M}_k\, \hat{\underline{\mathbf{x}}}_{k|k} + \mathbf{N}_k\, \underline{\mathbf{z}}_k^{(u)} \tag{A.1}$$

where the available input measurements $\underline{\mathbf{z}}_k^{(u)}$ are used. The corresponding prediction error yields

$$\begin{aligned}\underline{\mathbf{x}}_{k+1} - \hat{\underline{\mathbf{x}}}_{k+1|k} =& \mathbf{A}_k\, \underline{\mathbf{x}}_k + \mathbf{B}_k\, \underline{\mathbf{u}}_k + \underline{\mathbf{w}}_k \\ &- \big(\mathbf{M}_k\, \hat{\underline{\mathbf{x}}}_{k|k} + \mathbf{N}_k\, \underline{\mathbf{z}}_k^{(u)}\big) \\ =& \mathbf{A}_k \big(\underline{\mathbf{x}}_k - \hat{\underline{\mathbf{x}}}_{k|k}\big) + \big(\mathbf{A}_k - \mathbf{M}_k\big)\, \hat{\underline{\mathbf{x}}}_{k|k} + \underline{\mathbf{w}}_k \\ &+ \big(\mathbf{B}_k - \mathbf{N}_k\, \mathbf{C}_k^{(u)}\big)\, \underline{\mathbf{u}}_k + \mathbf{N}_k\, \underline{\mathbf{v}}_k^{(u)} \ .\end{aligned}$$

Without going into much detail, we see that $\mathbf{M}_k = \mathbf{A}_k$ and $\mathbf{N}_k\, \mathbf{C}_k^{(u)} = \mathbf{B}_k$ give the unbiasedness required to minimize the mean-squared prediction error. The matrix $\mathbf{N}_k$ takes the form $\mathbf{N}_k = \mathbf{B}_k \big(\mathbf{C}_k^{(u)}\big)^*$ with the second matrix being a pseudo inverse. The pseudo inverse $\big(\mathbf{C}_k^{(u)}\big)^*$ has to be determined so that the error covariance related to $\mathbf{N}_k\, \underline{\mathbf{v}}_k^{(u)}$ is minimized. It can be shown that this is achieved when the pseudo inverse equals (11), i.e., $\mathbf{N}_k = \mathbf{B}_k\, \mathbf{L}_k$. The corresponding error covariance then yields

$$\mathrm{Cov}\{\mathbf{N}_k\, \underline{\mathbf{v}}_k^{(u)}\} = \mathbf{B}_k\, \mathbf{L}_k \big(\mathbf{R}_k^{(u)}\big)^{-1} \mathbf{L}_k^{\mathrm{T}}\, \mathbf{B}_k^{\mathrm{T}} \overset{(10)}{=} \mathbf{B}_k\, \mathbf{P}_k^{(u)}\, \mathbf{B}_k^{\mathrm{T}} \ .$$

The prediction error then reduces to

$$\underline{\mathbf{x}}_{k+1} - \hat{\underline{\mathbf{x}}}_{k+1|k} = \mathbf{A}_k \big(\underline{\mathbf{x}}_k - \hat{\underline{\mathbf{x}}}_{k|k}\big) + \mathbf{B}_k\, \mathbf{L}_k\, \underline{\mathbf{v}}_k^{(u)} + \underline{\mathbf{w}}_k \ .$$

Using the determined parameters of (A.1) and the prediction error, we arrive at the equations (12) and (13). The following properties can be noticed: The Kalman filter estimates correspond to the conditional means given $\underline{\mathbf{z}}_k^{(u)}$ in place of $\underline{\mathbf{u}}_k$, the covariance (10) is part of the process noise matrix $\mathbf{Q}_k'$, and most importantly, the input estimate (9) naturally enters the prediction equations to determine the best unbiased prediction.