

# A System-Level Cooperative Multi-Agent GNSS Positioning Solution

Greiff, Marcus; Di Cairano, Stefano; Kim, Kyeong Jin; Berntorp, Karl

TR2023-135 November 03, 2023

## Abstract

We present a multi-agent cooperative estimation method for improving the performance of global navigation satellite systems (GNSSs). The proposed method uses existing receiver technology, avoids inter-agent communication, and minimizes the computational overhead in the agents. The method is based on recursive mixed-integer Kalman filtering for a system characterized by several agents in a bipartite star graph structure, where the nodes in one of the vertex sets perform local filtering based on local information, and a single node in the other vertex set estimates all of the system states using inter-agent error correlations in the context of partially overlapping local state spaces. We conduct extensive Monte-Carlo simulation studies in an urban driving scenario using a road map from an actual city, incorporating real satellite trajectories and realistic ionospheric bias modeling. In addition, we perform a hardware-in-the-loop study. The results indicate that the method can correct erroneous estimates in faulty agents by leveraging cooperation with other agents, improving accuracy from decimeter level to centimeter level for that particular agent. When all agents have similar residual biases, expected improvements in the root-mean-square position error typically range between 20–100%.

*IEEE Transactions on Control Systems Technology 2023*



# A System-Level Cooperative Multi-Agent GNSS Positioning Solution

Marcus Greiff<sup>1</sup>, Stefano Di Cairano<sup>1</sup>, Kyeong Jin Kim<sup>1</sup>, and Karl Berntorp<sup>1,\*</sup>

**Abstract**—We present a multi-agent cooperative estimation method for improving the performance of global navigation satellite systems (GNSSs). The proposed method uses existing receiver technology, avoids inter-agent communication, and minimizes the computational overhead in the agents. The method is based on recursive mixed-integer Kalman filtering for a system characterized by several agents in a bipartite star graph structure, where the nodes in one of the vertex sets perform local filtering based on local information, and a single node in the other vertex set estimates all of the system states using inter-agent error correlations in the context of partially overlapping local state spaces. We conduct extensive Monte-Carlo simulation studies in an urban driving scenario using a road map from an actual city, incorporating real satellite trajectories and realistic ionospheric bias modeling. In addition, we perform a hardware-in-the-loop study. The results indicate that the method can correct erroneous estimates in faulty agents by leveraging cooperation with other agents, improving accuracy from decimeter level to centimeter level for that particular agent. When all agents have similar residual biases, expected improvements in the root-mean-square position error typically range between 20–100%.

## I. INTRODUCTION

The classical *GNSS positioning problem* concerns the estimation of a single (hereafter referred to as local) receiver’s states from a set of code and carrier-phase measurements, acquired from one or several constellations of satellites [1]. The measurement equation, from which the receiver state is inferred, is time-varying, nonlinear in the position of the receiver, and incorporates biases [2]. For the carrier-phase measurements, some of the biases are integer-valued, commonly referred to as *ambiguities* [3]. If leveraging that these are integer-valued, the estimator can significantly increase the estimation accuracy at the cost of a more computationally complex (mixed-integer) estimation problem [1].

In this paper, we focus on the case of *multiple receivers* and propose an algorithm for cooperative estimation of multiple agents. We envision GNSS receivers that can stream relevant local statistics from various agents to a fusion center (FC), where the information is fused, before being propagated back to improve the local receiver estimates. With this cooperative estimation approach, we reuse existing GNSS technologies and minimize computational overhead in the receivers. The objective is to use the FC to fuse the local receiver estimates, simultaneously including modeled inter-receiver noise covariance and shared biases, thereby improving the local estimates by having additional information available. In the proposed strategy, the local receivers use their own internal estimation method and each has a nominal performance even during a

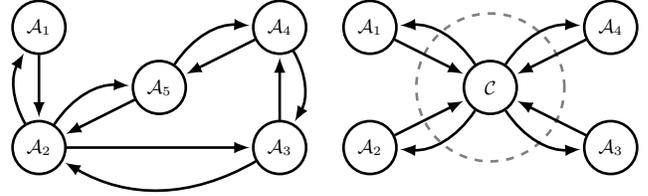


Fig. 1. *Left*: Communication graph for a completely decentralized scheme. *Right*: The star communication graph for the cooperative positioning strategy considered in this paper, partitioned into the vertex sets  $\mathcal{A}$  and  $\mathcal{C}$ .

complete loss of communication, and the FC adds performance improvement whenever communication is enabled. The works [4] and [5] have developed consensus algorithms by assuming static receivers whose states are configured in the same state-space. Our proposed solution, however, provides an estimation method for dynamic receivers with partially overlapping local state-spaces, for example, when the receivers share some bias states. The proposed solution also avoids additional burden of the receivers’ onboard processors, which have limited computational capabilities [6]. In the following, we interchangeably refer to a receiver as an *agent* denoted by  $\mathcal{A}_i$  with an internal state  $\mathbf{x}_k^i \in D_i$ . The set of  $N$  agents is denoted by  $\mathcal{A} = \{\mathcal{A}_i\}_{i=1}^N$  and the FC is denoted by  $\mathcal{C}$  with a global state vector  $\mathbf{x}_k^g \in D$  where  $D_i \subset D$ .

### A. Prior Work

Many solutions have been proposed for the GNSS positioning problem, with seminal works in [1], [3], [7]–[9], showing that the optimal solution to the mixed-integer least-squares (MILS) problem can be computed by solving a relaxed LS problem generating a *real-valued solution*, with the subsequent solution of an NP-hard integer LS (ILS) problem using various integer search methods (see [9] and references therein). If a sufficiently good integer ambiguity hypothesis is found, often determined by a ratio test [10], the real-valued solution can be conditioned on this integer hypothesis, resulting in a *fixed solution* and a reduction in estimate variance. This highlights the importance of integer fixation and spurred work on various filtering methods [2], [11]–[13], where nonlinear Kalman filters (KFs) are employed in combination with integer fixation, which is the state-of-the-art in high-precision GNSS implementations, such as the Real-Time Kinematic (RTK) software library [13]. However, alternative filtering methods have been proposed, for example, particle filters [14] and mixture KFs with marginalized ambiguities [15]–[17].

An improvement of single-receiver solutions can be done in two ways: (i) acquisition of more informative measurements, or (ii) improved filtering methods for inference based on current code and carrier-phase measurements. While there is significant concurrent work being done on improving the measurement quality in line with (i), such as launching the QZSS

<sup>1</sup>Mitsubishi Electric Research Labs (MERL), 02139 Cambridge, MA, USA.

\*Corresponding author. Email: greiff@merl.com, dicairano@ieee.org, kkim@merl.com, karl.o.berntorp@ieee.org

constellation in the Japan-Oceania region [18], a compatible and less costly source of high-quality measurements can be found by sharing data among the receivers. This is true if the receivers are in close geographical proximity, since then the biases associated with the same satellite are correlated.

To illustrate this, consider the distributed estimation of the parameters of a regional single-layer ionospheric delay model (see [19]), as done in [4], [5], where data from a large number of static receivers are shared over a sparse communication graph. Due to the significant correlation in the receiver biases, these parameters are approximately the same for each satellite in each receiver, enabling a filtering solution that significantly improves the ionospheric bias estimates [4]. This idea of leveraging geographical proximity of the receivers in the construction of the estimation algorithms can be done in a mobile receiver setting as well, but introduces complications.

### B. Contributions

Fig. 1 shows the communication graph in a decentralized scheme (left), as considered in [4], [5], and a bipartite star communication graph considered in this paper (right). Posing the multi-agent GNSS positioning problem for dynamic receivers in this manner facilitates a solution where data are not directly shared between agents, where the agents do not incur additional, possible large-scale, computations, and where the FC (here  $\mathcal{C}$ ) can fully utilize the local information. However, such an approach results in three significant technical hurdles, especially when considering moving receivers that are to function independently if communication with the FC fails:

- P1:** The agents have different and partially overlapping state-spaces due to different visible satellites in the agents. The local state-spaces relate to the global state vector as in the third subplot in Fig 2, not as in the first subplot assumed in the cited works. This complicates operations such as the averaging of local estimates, rendering standard consensus filters (see, e.g., [4], [5], [20]) and intersection methods (see e.g. [21], [22]) nontrivial to implement.
- P2:** Optimal fusion of the local estimates requires knowledge of the inter-agent estimate cross-covariance, which is complicated by the different local integer fixation employed in high-performance GNSS positioning [13].
- P3:** With an FC computing correction terms for the local estimators, there may be non negligible delays in the communication between  $\mathcal{A}$  and  $\mathcal{C}$ . The data may arrive asynchronously in  $\mathcal{C}$ , and the filters in  $\mathcal{A}$  will run at different rates than in  $\mathcal{C}$ . Properly dealing with this is essential to ensure numerical robustness in practice.

In contrast to previous work, we model cross-covariance in the GNSS measurements under certain assumptions, as well as combining solutions to **P1**, **P2**, and **P3**, facilitating a cooperative GNSS positioning service that can be implemented to enhance the local estimates of moving receivers. The novelties of the approach are; (i) the use of partially overlapping local state-spaces and subsequent use of optimal fusion, addressing **P1**; (ii) the explicit computation of the local error covariance of the relaxed estimates, which is made possible by the use of the dual density mixed-integer Kalman filters (MIKFs) in [23] and thereby addressing **P2**; and (iii) a method for back-propagation

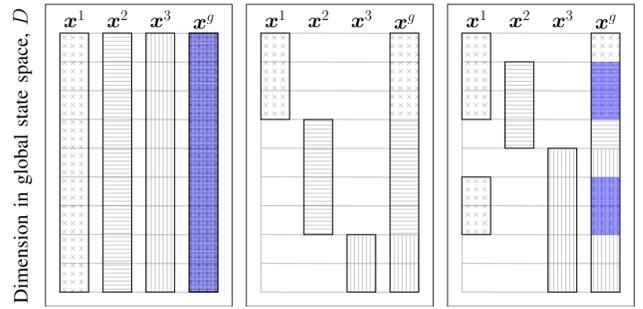


Fig. 2. Three agents with states  $\{x^1, x^2, x^3\}$ , with global state vector  $x^g \in D = \mathbb{R}^9$  including all unique local states. All states that are shared among the agents are highlighted in blue in the global state vector. *Left:* Local states and global states reside in the same spaces  $D_1 = D_2 = D_3 = D$ , i.e., each agent is estimating the same state vector. *Center:* Local and global states reside in different spaces, and  $D_1 \times D_2 \times D_3 = D$ , i.e., each local state is unique, but when stacked they reside in the global state space. *Right:* Local state spaces are partially overlapping, that is,  $D_1 \times D_2 \times D_3 \neq D$  and  $D_i \neq D_j$ .

of the improved estimates from the FC to the local agents, addressing **P3**. The method is evaluated in a Monte-Carlo study using an urban driving example that includes a car simulator, real GPS constellation trajectories, and ionospheric corrections realized from a perturbed Klobuchar model [24]. In addition, we validate the method using an Orolia hardware-in-the-loop (HIL) simulator [25], showing similar results as in the simulation study and validating our modeling assumptions.

### C. Outline and Notation

Sec. II describes the general GNSS measurement models, estimation models, and underlying assumptions. Sec. III develops the proposed cooperative positioning solution by addressing problems **P1**, **P2**, and **P3**. Sec. V presents a numerical evaluation of the proposed algorithm for vehicle positioning, and the conclusions in Sec. VI closes the paper.

Vectors are denoted by  $x \in \mathbb{R}^n$  with  $x_i$  being the  $i$ th element of  $x$ . Matrices are written as  $X$ , and the element on row  $i$  and column  $j$  of  $X$  is denoted with  $X_{ij}$ . The notation  $X_{(i,j)}$  indicates block  $(i,j)$  in a block-structured matrix  $X$ . The  $M \times M$  identity matrix is denoted by  $I_M$ ,  $\mathbf{1}_M$  indicates an  $M$ -length column matrix of ones, and  $A \otimes B$  denotes the Kronecker product between  $A$  and  $B$ . The smallest and largest eigenvalues of a matrix  $\Sigma$  are denoted with  $\lambda(\Sigma)$  and  $\bar{\lambda}(\Sigma)$ , respectively. To make the concatenation of vectors concise, we let  $(x; y) \triangleq [x^T y^T]^T \in \mathbb{R}^{n+m}$  for any  $x \in \mathbb{R}^n$  and  $y \in \mathbb{R}^m$ . The notation  $x \sim \mathcal{N}(\mu, \Sigma)$  indicates that  $x$  is Gaussian distributed with mean  $\mu$  and covariance  $\Sigma$ . The notation  $\hat{x}_{k|k}$  denotes the estimate of  $x$  at time step  $k$  given the set of measurements from time step 0 to time step  $k$ ,  $y_{0:k} \triangleq \{y_0, \dots, y_k\}$ , and  $\hat{x}_{k|k-1}$  is the one-step prediction of  $\hat{x}_{k-1|k-1}$ . With  $p(x_{0:k}|y_{0:k})$ , we denote the posterior density function of the state trajectory  $x_{0:k}$  from time step 0 to time step  $k$  given the measurements  $y_{0:k}$ , and  $p(x_k|y_{0:k})$  denotes the corresponding marginal (filtering) posterior.  $\text{diag}(\cdot)$  is a matrix composition, where the arguments form blocks on the diagonal, with all off-diagonal blocks set to zero.

## II. MODELING AND PROBLEM FORMULATION

In this section, the local and global state vectors and the GNSS measurement model are introduced, both in context of the local models in the agents and the global model in the FC.

**Definition 1** Let  $\mathcal{A}_i$  be an agent with associated (possibly time-varying) set of visible satellites  $\mathcal{S}_i$ , which is a subset of the possibly visible satellites,  $\mathcal{S} = \{1, \dots, N_s\} \subset \mathbb{N}$ , where

$$\mathcal{S}_i = \{s \in \mathcal{S} \mid s \text{ is observed by } \mathcal{A}_i\} \subseteq \mathcal{S}, \quad |\mathcal{S}_i| = N_s^i \leq N_s.$$

We consider the code and carrier-phase measurements from satellite  $s$  to the agent  $\mathcal{A}_i$  at a time  $t_k$  corresponding to the discrete time step  $k$ , using standard GNSS measurement models [11], [26]–[28]. By utilizing a base receiver (reference),  $\mathcal{B}$ , mounted at a known location broadcasting to the target receiver,  $\mathcal{A}_i$ , several error sources can be mitigated through differencing techniques [28]. Forming single differenced (SD) or double differenced (DD) measurements from pairs of satellites and the receivers reduces modeling errors, either explicitly or approximately. The errors that remain after the DD operation are estimated using established methods (e.g., [24], [29]), leading to *pseudo-range correction* (PRC) and *carrier-phase correction* (CPC) terms  $PRC_{i,k}^s$  and  $CPC_{i,k}^s$ , respectively, for each combination of agent  $\mathcal{A}_i$  and satellite  $s \in \mathcal{S}_i$ . In conventional notation,  $P_{i,k}^s$  is the code (pseudo-range) measurement, and  $\rho_{i,k}^s = \|\mathbf{p}_k^s - \mathbf{p}_{i,k}\|_2$  is the Euclidean distance between agent  $\mathcal{A}_i$  and satellite  $s$ , where  $\mathbf{p}_k^s, \mathbf{p}_{i,k} \in \mathbb{R}^3$  are the coordinates of satellite  $s$  and agent  $\mathcal{A}_i$ , respectively, in earth-centered earth-fixed (ECEF) coordinates [30]. Furthermore,  $I_{i,k}^s$  is the ionospheric delay;  $\epsilon_{i,k}^s$  is the code observation noise;  $\Phi_{i,k}^s$  is the carrier-phase observation;  $\lambda$  is the carrier wavelength;  $N_{i,k}^s$  is the integer ambiguity; and  $\eta_{i,k}^s$  denotes carrier observation noise. The noise sources are assumed to be zero-mean Gaussian, and we let  $\text{Cov}(\epsilon_{i,k}^{s_l}, \epsilon_{i,k}^{s_m}) = \text{Cov}(\eta_{i,k}^{s_l}, \eta_{i,k}^{s_m}) = 0$  if  $s_l \neq s_m$ , with  $\text{Cov}(\epsilon_{i,k}^{s_l}, \epsilon_{i,k}^{s_l}) = (\sigma_{\epsilon,k}^{s_l})^2$  and  $\text{Cov}(\eta_{i,k}^{s_l}, \eta_{i,k}^{s_l}) = (\sigma_{\eta,k}^{s_l})^2$  when  $s_l = s_m$ . When omitting terms that vanish under DD, the code and phase measurements of a satellite set  $\mathcal{S}_i$  and an agent  $\mathcal{A}_i$  at time step  $k$ , can be written in vector form as

$$\mathbf{P}_{i,k}^{\mathcal{S}_i} = \boldsymbol{\rho}_{i,k}^{\mathcal{S}_i} + \mathbf{I}_{i,k}^{\mathcal{S}_i} + \mathbf{PRC}_{i,k}^{\mathcal{S}_i} + \boldsymbol{\epsilon}_{i,k}^{\mathcal{S}_i}, \quad (1a)$$

$$\boldsymbol{\Phi}_{i,k}^{\mathcal{S}_i} = \boldsymbol{\rho}_{i,k}^{\mathcal{S}_i} + \lambda \mathbf{N}_{i,k}^{\mathcal{S}_i} - \mathbf{I}_{i,k}^{\mathcal{S}_i} + \mathbf{CPC}_{i,k}^{\mathcal{S}_i} + \boldsymbol{\eta}_{i,k}^{\mathcal{S}_i}, \quad (1b)$$

where  $\boldsymbol{\rho}_{i,k}^{\mathcal{S}_i} = (\rho_{i,k}^{s_1}, \dots, \rho_{i,k}^{s_{N_s^i}})^T \in \mathbb{R}^{N_s^i}$ , and the other vectors in (1) defined in analogous fashion.

**Definition 2** Let  $\mathbf{v} \in \mathbb{R}^{N_s}$  be a vector where each component is unique to each satellite, and let  $\mathbf{v}^i \in \mathbb{R}^{N_s^i}$  be the vector corresponding to  $\mathcal{S}_i$ . Then  $\mathbf{M}^i : \mathbb{R}^{N_s} \rightarrow \mathbb{R}^{N_s^i}$  is a binary incidence matrix associated with each  $\mathcal{A}_i$  such that  $\mathbf{v}^i = \mathbf{M}^i \mathbf{v}$ .

**Definition 3** Let  $\mathbf{S}^i : \mathbb{R}^{N_s^i} \rightarrow \mathbb{R}^{N_s^i-1}$  be a linear difference operator  $\mathbf{S}^i = [\mathbf{1}_{(N_s^i-1)} \quad -\mathbf{I}_{(N_s^i-1)}]$  associated with agent  $\mathcal{A}_i$ .

These definitions allow the construction of local DD measurement models that can be related in a global model using linear maps. Considering (1) we let  $\mathbf{P}_{i,k}^{\mathcal{S}_i} = \mathbf{M}^i \mathbf{P}_{i,k}^{\mathcal{S}}$  and  $\boldsymbol{\Phi}_{i,k}^{\mathcal{S}_i} = \mathbf{M}^i \boldsymbol{\Phi}_{i,k}^{\mathcal{S}}$ , and define SD measurements for agent  $\mathcal{A}_i$ ,

$$\Delta \mathbf{P}_{i\mathcal{B},k}^{\mathcal{S}_i} \triangleq \mathbf{P}_{\mathcal{B},k}^{\mathcal{S}_i} - \mathbf{P}_{i,k}^{\mathcal{S}_i}, \quad \Delta \boldsymbol{\Phi}_{i\mathcal{B},k}^{\mathcal{S}_i} \triangleq \boldsymbol{\Phi}_{\mathcal{B},k}^{\mathcal{S}_i} - \boldsymbol{\Phi}_{i,k}^{\mathcal{S}_i}. \quad (2)$$

Using Definition 3, the DD measurements for agent  $\mathcal{A}_i$  with respect to a base station  $\mathcal{B}$  and satellites  $\mathcal{S}_i$  are defined as

$$\mathbf{S}^i \Delta \mathbf{P}_{i\mathcal{B},k}^{\mathcal{S}_i} = \mathbf{S}^i \mathbf{M}^i [\mathbf{P}_{\mathcal{B},k}^{\mathcal{S}_i} - \mathbf{P}_{i,k}^{\mathcal{S}_i}], \quad (3a)$$

$$\mathbf{S}^i \Delta \boldsymbol{\Phi}_{i\mathcal{B},k}^{\mathcal{S}_i} = \mathbf{S}^i \mathbf{M}^i [\boldsymbol{\Phi}_{\mathcal{B},k}^{\mathcal{S}_i} - \boldsymbol{\Phi}_{i,k}^{\mathcal{S}_i}]. \quad (3b)$$

## A. State-Vector Definitions

The state  $\mathbf{x}^i$  of agent  $\mathcal{A}_i$  includes real-valued kinematic states,  $\boldsymbol{\kappa}^i$ , with the agent's position  $\mathbf{p}^i \in \mathbb{R}^3$  contained in  $\boldsymbol{\kappa}^i$ . In addition, we consider a set of biases defined for each agent. Typically, the biases include integer valued ambiguity biases  $\mathbf{n}^i$ , ionospheric biases  $\boldsymbol{\theta}^{I,i}$ , shared code correction terms  $\boldsymbol{\theta}^{P,i}$ , and shared phase correction terms  $\boldsymbol{\theta}^{C,i}$ .

**Assumption 1** Ionospheric delays from a satellite  $s$  are similar in agents  $\mathcal{A}_i$  and  $\mathcal{A}_j$  (i.e.,  $I_{\mathcal{A}_i,k}^s \approx I_{\mathcal{A}_j,k}^s$ ).

**Assumption 2** Each agent  $\mathcal{A}_i$  only knows the correction terms for satellites  $s \in \mathcal{S}_i$ .

The validity of Assumption 1 depends on several external factors, but in practice, it is valid when two receivers are within 3–10km from each other [31]. Assumption 2 relates to observability, and is essential if the agents are to function safely in the absence of communication with the FC.

Assumption 1 implies that a subset of the ionospheric biases in agent  $\mathcal{A}_i$ , specifically those for  $s \in \mathcal{S}_i \cap \mathcal{S}_j$ , are shared and approximately equal to the corresponding biases in agent  $\mathcal{A}_j$ . Consequently, we define the residual biases for the set  $\mathcal{S}$ , as

$$\boldsymbol{\theta}_k^I \triangleq \mathbf{I}_{\mathcal{B},k}^{\mathcal{S}} - \mathbf{I}_{i,k}^{\mathcal{S}} = \mathbf{I}_{\mathcal{B},k}^{\mathcal{S}} - \mathbf{I}_{j,k}^{\mathcal{S}} \in \mathbb{R}^{N_s}, \quad (4a)$$

$$\boldsymbol{\theta}_k^P \triangleq \mathbf{PRC}_{\mathcal{B},k}^{\mathcal{S}} - \mathbf{PRC}_{i,k}^{\mathcal{S}} = \mathbf{PRC}_{\mathcal{B},k}^{\mathcal{S}} - \mathbf{PRC}_{j,k}^{\mathcal{S}}, \quad (4b)$$

$$\boldsymbol{\theta}_k^C \triangleq \mathbf{CPC}_{\mathcal{B},k}^{\mathcal{S}} - \mathbf{CPC}_{i,k}^{\mathcal{S}} = \mathbf{CPC}_{\mathcal{B},k}^{\mathcal{S}} - \mathbf{CPC}_{j,k}^{\mathcal{S}}. \quad (4c)$$

While the ionospheric and correction biases are shared among various agents, the ambiguities can take any integer value. A cycle slip in one receiver does not necessarily imply the presence of a cycle slip in another receiver. The integer ambiguities are unique to each agent, and we estimate the DD ambiguities with a unique ambiguity vector for each agent,

$$\mathbf{n}_k^i \triangleq \mathbf{S}^i (\mathbf{N}_{\mathcal{B},k}^{\mathcal{S}_i} - \mathbf{N}_{i,k}^{\mathcal{S}_i}) \in \mathbb{Z}^{N_s^i-1}. \quad (5)$$

Hence, the global state vector in the FC is

$$\mathbf{x}_k^g = (\boldsymbol{\kappa}_k^1; \mathbf{n}_k^1; \dots; \boldsymbol{\kappa}_k^N; \mathbf{n}_k^N; \boldsymbol{\theta}_k^I; \boldsymbol{\theta}_k^P; \boldsymbol{\theta}_k^C) \in D. \quad (6)$$

Similarly, the local state vector of  $\mathcal{A}_i$  is

$$\mathbf{x}_k^i = (\boldsymbol{\kappa}_k^i; \mathbf{n}_k^i; \boldsymbol{\theta}_k^{I,i}; \boldsymbol{\theta}_k^{P,i}; \boldsymbol{\theta}_k^{C,i}) \in D_i, \quad (7)$$

where, for instance,  $\boldsymbol{\theta}_k^{I,i} = \mathbf{M}^i \boldsymbol{\theta}_k^I$  as per Definition 1.

## B. Estimation Model

1) *Measurement Model*: Let  $\Delta \boldsymbol{\rho}_{i\mathcal{B},k}^{\mathcal{S}_i} = \boldsymbol{\rho}_{\mathcal{B},k}^{\mathcal{S}_i} - \boldsymbol{\rho}_{i,k}^{\mathcal{S}_i}$ . For agent  $\mathcal{A}_i$ , the DD measurement is modeled in the local state  $\mathbf{x}^i$  in defined in (7) as

$$\mathbf{y}_k^{P,i} = \mathbf{S}^i \Delta \boldsymbol{\rho}_{i\mathcal{B},k}^{\mathcal{S}_i} + \mathbf{S}^i \boldsymbol{\theta}_k^{I,i} + \mathbf{S}^i \boldsymbol{\theta}_k^{P,i} + \mathbf{r}_k^{P,i}, \quad (8a)$$

$$\mathbf{y}_k^{\Phi,i} = \mathbf{S}^i \Delta \boldsymbol{\rho}_{i\mathcal{B},k}^{\mathcal{S}_i} + \lambda \mathbf{n}_k^i - \mathbf{S}^i \boldsymbol{\theta}_k^{I,i} + \mathbf{S}^i \boldsymbol{\theta}_k^{C,i} + \mathbf{r}_k^{\Phi,i}, \quad (8b)$$

$$\mathbf{y}_k^{I,i} = \boldsymbol{\theta}_k^{I,i} + \mathbf{r}_k^{I,i}, \quad (8c)$$

$$\mathbf{y}_k^{PRC,i} = \boldsymbol{\theta}_k^{P,i} + \mathbf{r}_k^{PRC,i}, \quad (8d)$$

$$\mathbf{y}_k^{CPC,i} = \boldsymbol{\theta}_k^{C,i} + \mathbf{r}_k^{CPC,i}, \quad (8e)$$

$$\mathbf{r}_k^i = (\mathbf{r}_k^{P,i}; \mathbf{r}_k^{\Phi,i}; \mathbf{r}_k^{I,i}; \mathbf{r}_k^{PRC,i}; \mathbf{r}_k^{CPC,i}), \quad (8f)$$

where  $\mathbf{r}_k^i \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_{(i,i),k})$  is Gaussian noise. More compactly,

$$\mathbf{y}_k^i = (\mathbf{y}_k^{P,i}; \mathbf{y}_k^{\Phi,i}; \mathbf{y}_k^{I,i}; \mathbf{y}_k^{PRC,i}; \mathbf{y}_k^{CPC,i}), \quad (9)$$

where  $\mathbf{y}_k^i \in \mathbb{R}^{5N_s^i-2}$  because of taking difference of satellites to generate (8a) and (8b). For each agent  $\mathcal{A}_i$ , the measurement model in (8) is compactly written as

$$\mathbf{y}_k^i = \mathbf{h}^i(\mathbf{x}_k^i) + \mathbf{r}_k^i. \quad (10)$$

Equations (8c)–(8e) are nonstandard and included to make the state vector  $\mathbf{x}^i$  observable with respect to the locally available information, given assumptions on the number of visible satellites, the dynamics of  $\mathbf{x}^i$ , and the problem geometry (to be detailed in Sec. II-B2). Hence, the shared biases  $\{\boldsymbol{\theta}_k^{I,i}, \boldsymbol{\theta}_k^{P,i}, \boldsymbol{\theta}_k^{C,i}\}$  can deviate from their models,  $\{\mathbf{y}_k^{I,i}, \mathbf{y}_k^{PRC,i}, \mathbf{y}_k^{CPC,i}\}$ , permitting adjustments through the cooperative positioning.

In the FC, all of the available local measurements are stacked into a global measurement vector, defined by the composition

$$\mathbf{y}_k^g = (\mathbf{y}_k^1; \mathbf{y}_k^2; \dots; \mathbf{y}_k^N) \quad (11)$$

where  $\mathbf{y}_k^g \in \mathbb{R}^{\sum_{i=1}^N (5N_s^i-2)}$  and

$$\mathbf{y}_k^g = \mathbf{h}(\mathbf{x}_k^g) + \mathbf{r}_k, \quad (12)$$

with  $\mathbf{h}(\mathbf{x}_k^g) = (\mathbf{h}^1(\mathbf{x}_k^1); \dots; \mathbf{h}^N(\mathbf{x}_k^N))$  and similar for  $\mathbf{r}_k$ . In (10) and (12), the noise  $\mathbf{r}_k^i$  is zero-mean Gaussian, with

$$\mathbf{R}_{(i,i),k} = \text{diag}(\mathbf{R}_{(i,i),k}^{PP}, \mathbf{R}_{(i,i),k}^{\Phi\Phi}, \mathbf{R}_{(i,i),k}^{II}, \mathbf{R}_{(i,i),k}^{PRC}, \mathbf{R}_{(i,i),k}^{CPC}),$$

where  $\mathbf{R}_{(i,i),k}^{II}, \mathbf{R}_{(i,i),k}^{PRC}, \mathbf{R}_{(i,i),k}^{CPC}$  reflect the relative uncertainty of the correction terms computed in agent  $\mathcal{A}_i$ . Assumption 1 implies that the shared biases are similar in agents sufficiently close to each other. The covariance in the measurement noise of shared biases between two agents  $\mathcal{A}_i$  and  $\mathcal{A}_j$ ,  $i \neq j$ , is

$$\mathbf{R}_{(i,i),k}^{II} = \sigma_I^2 \mathbf{I}, \quad \mathbf{R}_{(i,j),k}^{II} = \frac{1}{2} \sigma_I^2 \mathbf{M}^i (\mathbf{M}^j)^\top, \quad (13a)$$

$$\mathbf{R}_{(i,i),k}^{PRC} = \sigma_{PRC}^2 \mathbf{I}, \quad \mathbf{R}_{(i,j),k}^{PRC} = \frac{1}{2} \sigma_{PRC}^2 \mathbf{M}^i (\mathbf{M}^j)^\top, \quad (13b)$$

$$\mathbf{R}_{(i,i),k}^{CPC} = \sigma_{CPC}^2 \mathbf{I}, \quad \mathbf{R}_{(i,j),k}^{CPC} = \frac{1}{2} \sigma_{CPC}^2 \mathbf{M}^i (\mathbf{M}^j)^\top, \quad (13c)$$

for modeling uncertainty defined by  $\sigma_I, \sigma_{PRC}, \sigma_{CPC} > 0$ . The following holds for the measurement noise cross-covariance in  $\mathbf{r}_k^i$  and  $\mathbf{r}_k^j$ .

**Proposition 1** Assume that the single difference in (2) of two agents  $\mathcal{A}_i$  and  $\mathcal{A}_j$  with  $i \neq j$  is done with respect to the same base station  $\mathcal{B}$ . Then, the measurement noise covariance matrix of the DD measurements in between  $\mathcal{A}_i$  and  $\mathcal{A}_j$  is

$$\text{Cov} \left( \begin{bmatrix} \mathbf{r}_k^{P,i} \\ \mathbf{r}_k^{\Phi,i} \\ \mathbf{r}_k^{P,j} \\ \mathbf{r}_k^{\Phi,j} \end{bmatrix}, \begin{bmatrix} \mathbf{r}_k^{P,i} \\ \mathbf{r}_k^{\Phi,i} \\ \mathbf{r}_k^{P,j} \\ \mathbf{r}_k^{\Phi,j} \end{bmatrix} \right) = \begin{bmatrix} \mathbf{R}_{(i,i),k}^{PP} & \mathbf{0} & \mathbf{R}_{(i,j),k}^{PP} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_{(i,i),k}^{\Phi\Phi} & \mathbf{0} & \mathbf{R}_{(i,j),k}^{\Phi\Phi} \\ \mathbf{R}_{(j,i),k}^{PP} & \mathbf{0} & \mathbf{R}_{(j,j),k}^{PP} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_{(j,i),k}^{\Phi\Phi} & \mathbf{0} & \mathbf{R}_{(j,j),k}^{\Phi\Phi} \end{bmatrix},$$

$$\mathbf{R}_{(i,i),k}^{PP} = 2\mathbf{S}^i \mathbf{M}^i \mathbf{R}_k^{\epsilon\epsilon} (\mathbf{M}^i)^\top (\mathbf{S}^i)^\top, \quad (14a)$$

$$\mathbf{R}_{(i,i),k}^{\Phi\Phi} = 2\mathbf{S}^i \mathbf{M}^i \mathbf{R}_k^{\eta\eta} (\mathbf{M}^i)^\top (\mathbf{S}^i)^\top, \quad (14b)$$

$$\mathbf{R}_{(i,j),k}^{PP} = \mathbf{S}^i \mathbf{M}^i \mathbf{R}_k^{\epsilon\epsilon} (\mathbf{M}^j)^\top (\mathbf{S}^j)^\top, \quad (14c)$$

$$\mathbf{R}_{(i,j),k}^{\Phi\Phi} = \mathbf{S}^i \mathbf{M}^i \mathbf{R}_k^{\eta\eta} (\mathbf{M}^j)^\top (\mathbf{S}^j)^\top. \quad (14d)$$

**Proof 1** The proof is sketched in Appendix A. ■

In this paper, we model the measurement noise covariance

$$\mathbf{R}_{\epsilon\epsilon,k} = \text{diag}((\sigma_{\epsilon,k}^1)^2, \dots, (\sigma_{\epsilon,k}^{N_s})^2), \quad (15a)$$

$$\mathbf{R}_{\eta\eta,k} = \text{diag}((\sigma_{\eta,k}^1)^2, \dots, (\sigma_{\eta,k}^{N_s})^2), \quad (15b)$$

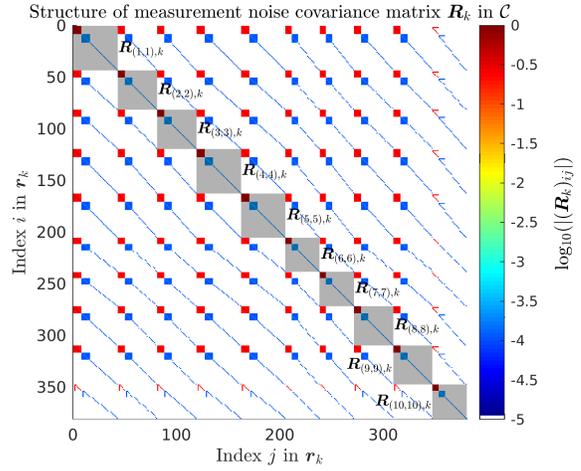


Fig. 3. Structure of the measurement noise covariance matrix in an  $N = 10$  agent model, with the block diagonal elements  $\mathbf{R}_{(i,i),k}$  in gray. The colors indicate the magnitude of each element in  $\mathbf{R}_k$ .

as uncorrelated, but in general the inter-agent measurement noise covariance can be expressed as in Proposition 1 even when the measurement noise is correlated and time-varying.

From (13), Proposition 1, Assumption 1, and the model in (8), the inter-agent noise cross-covariance is

$$\begin{aligned} \mathbf{R}_{(i,j),k} &\triangleq \text{Cov}(\mathbf{r}_k^i, \mathbf{r}_k^j) \\ &= \text{diag}(\mathbf{R}_{(i,j),k}^{PP}, \mathbf{R}_{(i,j),k}^{\Phi\Phi}, \mathbf{R}_{(i,j),k}^{II}, \mathbf{R}_{(i,j),k}^{PRC}, \mathbf{R}_{(i,j),k}^{CPC}). \end{aligned}$$

The resulting structured measurement noise covariance in the global model is illustrated in Fig. 3, and leveraging this additional information in the FC will be shown to improve the estimation accuracy in the numerical results (see Sec. V).

2) *Prediction Model*: The integer ambiguities in (5) that appear in the carrier-phase measurements (8b) are constant over long periods of time before single ambiguities instantaneously take new integer values, known as *cycle slip*. The time evolution of these ambiguities can be modeled as a random walk driven by a discrete noise-process. However, instead of driving the random walk with Gaussian noise, the noise is defined such that with a small probability  $b \in [0, 1]$  it attains a random integer in  $[-a, a] \subset \mathbb{Z} \setminus \{0\}$ , and with a probability  $1 - b$  it is 0 (see the modeling in [23, Sec. IV.B]).

Such noise is radically different from the Gaussian process noise assumed in Kalman filtering. To address this, one approach is to solve a *relaxed estimation problem*, in which the integer ambiguities are configured on the real numbers, detect a cycle slip when it occurs and adapt the corresponding Gaussian process noise accordingly. To this end, we adopt the cycle-slip detection in [23, Sec. IV.B], which assesses if the predicted measurement significantly differs from the acquired measurement. This is done by ascribing all of the variation in a DD code measurement to specific dimensions  $j$  of  $\mathbf{n}_k^i$ , denoted as  $\delta n_{j,k}^i$ , and comparing this to a threshold  $d > 0$ . This gives rise to an incidence vector  $\mathbf{c}_k^i$ , with elements

$$\mathbf{c}_{j,k}^i = \begin{cases} 1 & \text{if } |\delta n_{j,k}^i| > d \\ 0 & \text{otherwise} \end{cases}. \quad (16)$$

Hence, in the relaxed estimation model, the ambiguity prediction is described by a random walk with time-varying noise,

$$\mathbf{n}_k^i = \mathbf{n}_{k-1}^i + \mathbf{q}_{k-1}^{n,i}, \quad (17a)$$

$$\mathbf{q}_{k-1}^{n,i} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_{(i,i),k-1}^{nn}), \quad (17b)$$

$$\mathbf{Q}_{(i,i),k-1}^{nn} = \text{diag}(\mathbf{c}_k^i \sigma_{jump}^2 + (\mathbf{1} - \mathbf{c}_k^i) \sigma_{stay}^2), \quad (17c)$$

where the variance  $\sigma_{jump}^2$  reflects the width of the domain of the integer jump process  $a$  and the associated probability  $b$ , and  $\sigma_{stay}^2$  can be seen as regularizing process noise [23].

Since the correction terms are slowly time varying, it is appropriate to model them as Gaussian-distributed random walks. Consequently, the local model for agent  $\mathcal{A}_i$  is

$$\begin{aligned} \mathbf{x}_{k+1}^i &= \mathbf{F}^i(\mathbf{x}_k^i) + \mathbf{q}_k^i, & \mathbf{q}_k^i &\sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k^i), \\ \mathbf{F}^i(\mathbf{x}_k^i) &= (\mathbf{f}^i(\boldsymbol{\kappa}_k^i); \mathbf{n}_k^i; \boldsymbol{\theta}_k^{I,i}; \boldsymbol{\theta}_k^{P,i}; \boldsymbol{\theta}_k^{C,i}), \end{aligned} \quad (18)$$

where  $\mathbf{f}^i$  is the  $i$ th local state-transition (motion) model, which in general is nonlinear. The prediction model in the FC is

$$\begin{aligned} \mathbf{x}_{k+1}^g &= \mathbf{F}(\mathbf{x}_k^g) + \mathbf{q}_k, & \mathbf{q}_k &\sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k), \\ \mathbf{F}(\mathbf{x}_k^g) &= (\mathbf{f}^1(\mathbf{x}_k^1); \mathbf{n}_k^1; \dots; \mathbf{f}^N(\mathbf{x}_k^N); \mathbf{n}_k^N; \boldsymbol{\theta}_k^I; \boldsymbol{\theta}_k^P; \boldsymbol{\theta}_k^C). \end{aligned} \quad (19)$$

To describe the receiver motion, various prediction models can be considered (see [32]). Regardless of the motion model used, let the process noise associated with  $\boldsymbol{\kappa}_k^i$  be denoted by  $\mathbf{Q}_{(i,i),k}^{\boldsymbol{\kappa}\boldsymbol{\kappa}}$ , and the noise associated with the bias states  $(\boldsymbol{\theta}_k^{I,i}; \boldsymbol{\theta}_k^{P,i}; \boldsymbol{\theta}_k^{C,i})$  be denoted by  $\mathbf{Q}_{(i,i),k}^{\boldsymbol{\theta}\boldsymbol{\theta}} = h\sigma_B^2 \mathbf{I}$ . Hence,

$$\mathbf{Q}_k^i = \text{diag}(\mathbf{Q}_{(i,i),k}^{\boldsymbol{\kappa}\boldsymbol{\kappa}}, \mathbf{Q}_{(i,i),k}^{nn}, \mathbf{Q}_{(i,i),k}^{\boldsymbol{\theta}\boldsymbol{\theta}}), \quad (20)$$

and the expression for  $\mathbf{Q}_k$  follows from (6) and (7).

Fig. 4 illustrates the relationship between the local and the global models for a typical  $N = 10$  agent problem (similar to Fig. 2). Such models are used in the simulations in Sec. V, indicating both the scale and the complexity of the problem.

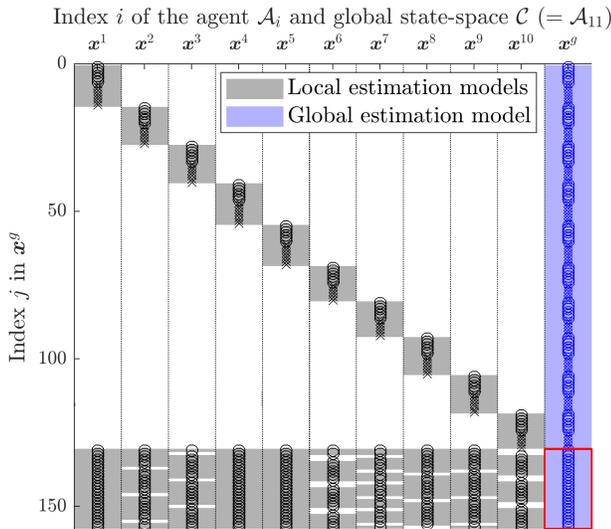


Fig. 4. Illustration of how the states in the local estimation models (gray) relate to the global state vector (blue) with the shared states in the global model highlighted with the red box. The real-valued states are marked with "o" and the integer valued ambiguity states are marked with "x".

Given how the biases enter in (8), and with a local model of the form (18), the local state vector  $\mathbf{x}_k^i$  of agent  $\mathcal{A}_i$  is observable in  $\mathbf{y}_k^i$  under mild assumptions.

**Remark 1** For instance, take a constant-velocity (CV) model,

$$\mathbf{f}^i(\boldsymbol{\kappa}_k^i) = \begin{bmatrix} \mathbf{I} & h\mathbf{I} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \boldsymbol{\kappa}_k^i, \quad \mathbf{Q}_{(i,i),k}^{\boldsymbol{\kappa}\boldsymbol{\kappa}} = \begin{bmatrix} \frac{h^3}{2} \mathbf{I} & \frac{h^2}{2} \mathbf{I} \\ \frac{h^2}{2} \mathbf{I} & h\mathbf{I} \end{bmatrix}, \quad (21)$$

with the kinematic states of the receiver,  $\boldsymbol{\kappa}_k^i = (\mathbf{p}_k^i; \mathbf{v}_k^i) \in \mathbb{R}^6$ , defined as a composition of the receiver position and velocity. Consider the measurement equation in (10) at a point  $\boldsymbol{\kappa} = \boldsymbol{\kappa}_0 \in \mathbb{R}^6$ , and let  $[(\partial \Delta \rho_{\mathcal{A}_i \mathcal{B}, k}^S) / (\partial \boldsymbol{\kappa}_k)]|_{\boldsymbol{\kappa}_k = \boldsymbol{\kappa}_0} \triangleq [\mathbf{J}, \mathbf{0}] \in \mathbb{R}^{N_s^i \times 6}$ . It follows that  $\text{rank}(\mathbf{S}\mathbf{J}) \geq 3$  if  $N_s^i \geq 4$  and the satellites are located at different points in space, with at least three directions  $\mathbf{p}_k^s - \mathbf{p}_{\mathcal{A}_i, k}$  being nonparallel. Under these conditions, with  $\mathbf{A} \triangleq [(\partial \mathbf{F}^i) / (\partial \mathbf{x}_k^i)]|_{\boldsymbol{\kappa}_k = \boldsymbol{\kappa}_0}$  and  $\mathbf{C} \triangleq [(\partial \mathbf{h}^i) / (\partial \mathbf{x}_k^i)]|_{\boldsymbol{\kappa}_k = \boldsymbol{\kappa}_0}$ , for all  $\boldsymbol{\kappa}_0$ ,

$$\text{rank} \left( \begin{bmatrix} \mathbf{C} \\ \mathbf{C}\mathbf{A} \end{bmatrix} \right) = \text{dim}(\mathbf{x}^i). \quad (22)$$

### C. Problem Formulation

The information sent from an agent  $\mathcal{A}_i$  to the global center  $\mathcal{C}$  contains the code and carrier-phase measurements, as well as an estimate of the local state  $\mathbf{x}^i$ . The local estimator producing the state estimate is Bayesian as outlined in Assumption 3.

**Assumption 3** Each agent  $\mathcal{A}_i$  executes a recursive estimator that at each time step  $k$  updates a posterior distribution  $p(\mathbf{x}_k^i | \mathbf{y}_{0:k}^i)$  and extracts a first and second moment estimate,  $\{\hat{\mathbf{x}}_{k|k}^i, \boldsymbol{\Sigma}_{k|k}^i\}$ , such that  $p(\mathbf{x}_k^i | \mathbf{y}_{0:k}^i) \approx \mathcal{N}(\hat{\mathbf{x}}_{k|k}^i, \boldsymbol{\Sigma}_{k|k}^i)$ . Similarly, the recursive estimator updates the posterior with a virtual measurement described by a first and second moment  $\{\hat{\mathbf{z}}_k, \boldsymbol{\Sigma}_{z,k}\}$ , resulting in  $p(\mathbf{x}_k^i | \mathbf{y}_{0:k}^i, \mathbf{z}_k)$ . The virtual measurement is elaborated upon in Sec. III-D.

Based on Assumption 3, the measurement model (12), and prediction models in (18) and (19), we seek a positioning system where the FC receives the set of code and carrier-phase measurements from each agent,  $\mathbf{y}_k^i$ , as well as the first and second moment estimates  $\{\hat{\mathbf{x}}_{k|k}^i, \boldsymbol{\Sigma}_{k|k}^i\}_{i=1}^N$  of the state (7). The FC then determines a first and second moment  $\{\hat{\mathbf{z}}_k, \boldsymbol{\Sigma}_{z,k}\}$  for each agent  $\mathcal{A}_i$  that can be used to update the posterior distribution of  $p(\mathbf{x}_k^i | \mathbf{y}_{0:k}^i)$ , to result in an improved local posterior distribution  $p(\mathbf{x}_k^i | \mathbf{y}_{0:k}^i, \mathbf{z}_k) \approx p(\mathbf{x}_k^i | \mathbf{y}_{0:k}^i)$  that includes knowledge of partially shared biases states and inter-agent measurement noise cross-correlation.

**Remark 2** Assumption 3 restricts our approach to be used in combination with statistical estimators. However, GNSS estimators either produce general posterior distributions (e.g., [17], [26]) or are designed to be moment estimators (e.g., [12], [23], [33]). The capability to extract a first and second moment is readily handled by statistical estimators such as the EKF, LTKFs, and sequential Monte-Carlo methods (e.g., particle filters). The capability to include first and second moments to update the posterior distribution means that the estimator in each agent needs to incorporate information that is Gaussian distributed, which is handled already by the above-mentioned methodologies.

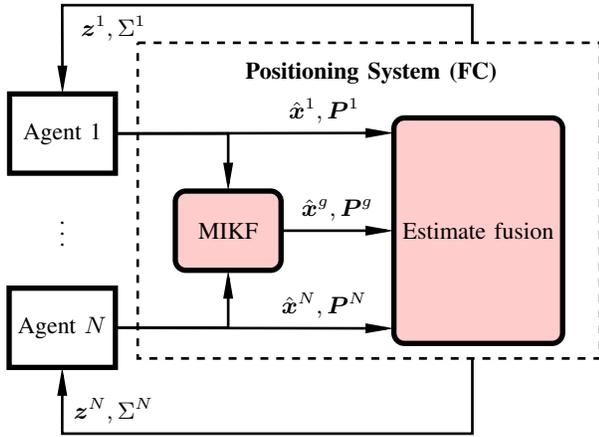


Fig. 5. Architecture of the proposed cooperative GNSS positioning system. The proposed positioning solution consists of; (i) an estimator (MIKF) that has access to all available satellite measurements across the  $N$  different agents and leverage measurement cross-correlation; (ii) an estimate fusion block that fuses the global estimate with the local estimates using inter-agent cross-covariance estimation; and (iii) a method for propagating back the updated estimate to the local agents.

### III. COOPERATIVE GNSS POSITIONING SYSTEM

In this section we present the proposed GNSS-based positioning system. The key ingredients enabling our approach are; (i), from (13) and Proposition 1, the FC  $\mathcal{C}$  has an information advantage because there is substantial cross-covariance between the agents that can be leveraged in the FC (see Fig 3); and (ii), from Assumption 1, there are residual biases that are similar across multiple agents (see Fig 4).

Fig. 5 shows a high-level diagram of the positioning system. A set of agents transmits local state and measurement information to the FC. The FC consists of three building blocks. First, an MIKF that utilizes the partially linear structure in the estimation model. In the FC, the MIKF utilizes cross-covariances between measurements from different agents. Second, the FC performs fusion between local estimates and the estimates from the MIKF operating on the global information. This step is crucial when the local agents operate at different rates than the FC and/or the local agents use additional onboard sensing that is not transmitted to the FC. Third, the FC back-propagates the updated state information by a *virtual measurement* to the local agents.

In the following, we explain the different components of the positioning system. We start with a brief review of the MIKF that is executed in the local agents and describe the recursive estimator that produces a global state estimate leveraging all satellite measurements.<sup>1</sup> Then, we describe how to optimally fuse the local state estimates in  $\mathcal{A}_i$  with the global state estimate in  $\mathcal{C}$ , solving **P1**. To do this, we derive an expression, which is exact in the linear case, to recursively estimate the inter-agent covariance, solving **P2**. Finally, we provide a method for propagating the fused estimate from  $\mathcal{C}$  to  $\mathcal{A}_i$  by using *virtual measurements*,  $z^i$ , solving **P3**.

<sup>1</sup>Note that the MIKF fulfills Assumption 3, but any other method could be used, as long as it satisfies Assumption 3

#### A. Mixed-Integer Kalman Filter and the Fusion Center

The dominating paradigm in GNSS positioning is to solve relaxed estimation problems, where all of the states in the local state vector are real-valued. Then, based on the most likely integer ambiguity hypothesis found by an ILS, a state estimate is determined. For the purposes of this paper, an appealing method is given in [23], which uses the partially linear moment approximations proposed in [34] to estimate two separate densities over the real-valued states. This filter, MIKF, recursively estimates three separate probability densities,

- Relaxed:  $p(\kappa_k^i, \mathbf{n}_k^i, \boldsymbol{\theta}_k^{I,i}, \boldsymbol{\theta}_k^{P,i}, \boldsymbol{\theta}_k^{C,i} | \mathbf{y}_{0:k}^i) \approx \mathcal{N}(\hat{\mathbf{x}}_{k|k}^i, \mathbf{P}_{k|k}^i)$
- Fixed:  $p(\kappa_k^i, \boldsymbol{\theta}_k^{I,i}, \boldsymbol{\theta}_k^{P,i}, \boldsymbol{\theta}_k^{C,i} | \mathbf{y}_{0:k}^i, \mathbf{n}_{0:k}^i) \approx \mathcal{N}(\check{\mathbf{x}}_{k|k}^i, \check{\mathbf{P}}_{k|k}^i)$
- Integer:  $\mathbf{n}_k^{I,i} = \underset{\mathbf{n}_k^i \in \mathbb{Z}^{N_s^i}}{\operatorname{argmax}} p(\kappa_k^i, \mathbf{n}_k^i, \boldsymbol{\theta}_k^{I,i}, \boldsymbol{\theta}_k^{P,i}, \boldsymbol{\theta}_k^{C,i} | \mathbf{y}_{0:k}^i)$

The resulting statistics,  $\{\hat{\mathbf{x}}_{k|k}^i, \mathbf{P}_{k|k}^i, \check{\mathbf{x}}_{k|k}^i, \check{\mathbf{P}}_{k|k}^i, \mathbf{n}_k^{I,i}\}$ , are known locally and stored in agent  $\mathcal{A}_i$ , and are updated recursively when receiving the measurements  $\mathbf{y}_k^i$  using [23, Algorithm 1]. This is represented by the function

$$\mathcal{A}_i \leftarrow \text{MIKF\_update}(\mathcal{A}_i, \mathbf{y}_k^i). \quad (23)$$

The FC has access to the global model and employs the MIKF update over the relaxed estimates  $\{\hat{\mathbf{x}}_{k|k}^g, \mathbf{P}_{k|k}^g\}$ , written as

$$\mathcal{C} \leftarrow \text{MIKF\_update}(\mathcal{C}, \mathbf{y}_k^g). \quad (24)$$

In addition, the FC updates a third density, which is the fused estimates of all of the local and global states,  $\mathcal{N}(\hat{\mathbf{x}}_{k|k}^f, \mathbf{P}_{k|k}^f)$ . This density is defined in the global state space and its computation is described in Sec. III-B.

**Remark 3** *The FC has two main information advantages compared to each local agent. First, (14) implies that for satellites measuring more than one agent at the same time and using the same base station, the measurement covariance has cross-correlation. Second, the sharing of some of the residual biases implies there is considerable mutual information in the states of the local receivers.*

**Remark 4** *The MIKF in [23, Algorithm 1] includes estimating a Gaussian posterior defined over the complete local state space in the relaxed density, and is updated by a Kalman-type update. As such, the MIKF satisfies Assumption 3.*

#### B. Fusing Local and Global Estimates

The estimates  $\{\hat{\mathbf{x}}_{k|k}^i, \mathbf{P}_{k|k}^i\}$  in Sec. III-A are based on the global model, making use of known inter-agent noise correlation and sharing of residual biases that individual agents cannot utilize. However, the agents might incorporate additional sensors, and their filters are typically run at faster update rates. By fusing the local estimates with global estimates, we can therefore potentially improve performance in a fused estimate, compared to simply running the MIKF in Sec. III-A with the global estimation model in (12) and (19).

There exist numerous methods for fusing first and second moments of estimates or *information items* from different agents. For instance, fusing  $\mathbf{x}^i \sim \mathcal{N}(\boldsymbol{\mu}^i, \boldsymbol{\Sigma}^i)$  with  $\mathbf{x}^g \sim \mathcal{N}(\boldsymbol{\mu}^g, \boldsymbol{\Sigma}^g)$  into a combined information item  $\mathbf{x}^f \sim \mathcal{N}(\boldsymbol{\mu}^f, \boldsymbol{\Sigma}^f)$ , can be done by convex combinations (CC),

covariance intersection (CI), or inverse CI (ICI) [21], [22]. These methods do not assume knowledge of a cross-covariance  $\Sigma^{gi} = \mathbb{E}[(\mathbf{x}^g - \boldsymbol{\mu}^g)(\mathbf{x}^i - \boldsymbol{\mu}^i)^\top]$ , which leads to CC, CI, and ICI overestimating the fused covariance. In addition, these methods assume that  $\boldsymbol{\mu}^i$  and  $\boldsymbol{\mu}^g$  are defined in the same space (i.e., the left-most scenario in Fig. 2). For a known cross-covariance  $\Sigma^{gi}$ , the Bar-Shalom/Campo rule (BC) [35] is the minimum mean-square error (MMSE) fusion for Gaussian distributions with known cross-covariance. In its original formulation, the BC rule assumes a fully overlapping state space (i.e.,  $(\mathbf{x}^i; \mathbf{x}^g) = (\mathbf{x}^f; \mathbf{x}^f)$ ). However, it can be extended to a general case where the local state vectors partially overlap [21] (i.e., the right-most scenario in Fig. 2). Due to a linear relationship between the the local-state space of  $\mathcal{A}_i$  and the global state space in  $\mathcal{C}$ , we build a mapping between the global state  $\mathbf{x}^g$  in the FC and the stacking of the local state  $\mathbf{x}^i$  of each agent  $\mathcal{A}_i$ ,  $\mathbf{x}^l$ ,

$$\underbrace{\begin{bmatrix} \mathbf{x}_k^1 \\ \mathbf{x}_k^2 \\ \vdots \\ \mathbf{x}_k^{N-1} \\ \mathbf{x}_k^N \end{bmatrix}}_{\triangleq \mathbf{x}_k^l} = \underbrace{\begin{bmatrix} \mathbf{T}_u^1 & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{T}_s^1 \\ \mathbf{0} & \mathbf{T}_u^2 & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{T}_s^2 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{T}_u^{N-1} & \mathbf{0} & \mathbf{T}_s^{N-1} \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{T}_u^N & \mathbf{T}_s^N \end{bmatrix}}_{\triangleq \mathbf{T}_{g \rightarrow l}} \mathbf{x}_k^g, \quad (25a)$$

where

$$\mathbf{T}_u^i \triangleq \begin{bmatrix} \mathbf{I}_{N_u^i \times N_u^i} \\ \mathbf{0}_{3N_s^i \times N_u^i} \end{bmatrix}, \quad \mathbf{T}_s^i \triangleq \begin{bmatrix} \mathbf{0}_{N_u^i \times N_s} \\ \mathbf{1}_3 \otimes \mathbf{M}^i \end{bmatrix}, \quad (25b)$$

with  $N_u^i \triangleq \dim(\mathbf{k}^i) + \dim(\mathbf{n}^i)$  denoting the number of states that are unique to agent  $\mathcal{A}_i$ . Eq. (25) follows from the state-space definitions provided in (6) and (7). By Assumption 3 and assuming that it is possible to estimate the inter-agent cross covariance  $\Sigma^{gl}$ ,

$$\begin{bmatrix} \mathbf{x}^l \\ \mathbf{x}^g \end{bmatrix} \sim \mathcal{N} \left( \begin{bmatrix} \boldsymbol{\mu}^l \\ \boldsymbol{\mu}^g \end{bmatrix}, \begin{bmatrix} \Sigma^l & \Sigma^{lg} \\ \Sigma^{gl} & \Sigma^g \end{bmatrix} \right) \triangleq \mathcal{N}(\boldsymbol{\mu}^p, \Sigma^p). \quad (26)$$

The local and global state estimates,  $\mathbf{x}^l$  and  $\mathbf{x}^g$ , relate to the fused information item in the state-space of the FC by

$$\begin{bmatrix} \mathbf{x}^l \\ \mathbf{x}^g \end{bmatrix} = \begin{bmatrix} \mathbf{T}_{g \rightarrow l} \\ \mathbf{I} \end{bmatrix} \mathbf{x}^f \triangleq \mathbf{H}^f \mathbf{x}^f. \quad (27)$$

Here, the optimally fused estimate  $\mathbf{x}^f \sim \mathcal{N}(\boldsymbol{\mu}^f, \Sigma^f)$  is the solution to the WLS-problem

$$\mathbf{K}^f = ((\mathbf{H}^f)^\top (\Sigma^p)^{-1} \mathbf{H}^f)^{-1} (\mathbf{H}^f)^\top (\Sigma^l)^{-1}, \quad (28a)$$

$$\boldsymbol{\mu}^f = \mathbf{K}^f \boldsymbol{\mu}^p, \quad (28b)$$

$$\Sigma^f = \mathbf{K}^f \Sigma^p (\mathbf{K}^f)^\top = ((\mathbf{H}^f)^\top (\Sigma^p)^{-1} \mathbf{H}^f)^{-1}. \quad (28c)$$

If  $\mathbf{T}_{g \rightarrow l} = \mathbf{I}$ , (28) simplifies to

$$\boldsymbol{\mu}^f = \boldsymbol{\mu}^l + (\Sigma^l - \Sigma^{lg}) \bar{\Sigma}^{-1} (\boldsymbol{\mu}^g - \boldsymbol{\mu}^l), \quad (29a)$$

$$\Sigma^f = \Sigma^l - (\Sigma^l - \Sigma^{lg}) \bar{\Sigma}^{-1} (\Sigma^l - \Sigma^{gl}), \quad (29b)$$

$$\bar{\Sigma} = (\Sigma^l + \Sigma^g - \Sigma^{lg} - \Sigma^{gl}). \quad (29c)$$

However, (28) allows general state-space descriptions and therefore solves Problem **P1**. Given (27) defined by (25), we refer to such a generalized BC fusion through (28) of the information items in (26), by the function

$$[\boldsymbol{\mu}_k^f, \Sigma_k^f] \leftarrow \text{gbc\_fusion}(\boldsymbol{\mu}_k^p, \Sigma_k^p, \mathbf{T}_{g \rightarrow l}). \quad (30)$$

### C. Inter-Agent Covariance Estimation for Optimal Fusion

The generalized BC-fusion (28) relies on knowledge of the cross-covariance of the estimate errors in forming (26). In the general case, there is no analytical formula for determining the cross-covariance of inter-agent estimation errors. However, for a first-order expansion of the system nonlinearities, we obtain a recursive method of computing the cross-covariances involved in the generalized BC-fusion, as summarized in Proposition 2.

**Proposition 2** Assume that each  $\mathcal{A}_i$  estimates a relaxed Gaussian distribution  $\mathcal{N}(\hat{\mathbf{x}}_{k|k}^i, \mathbf{P}_{k|k}^{ii})$  of the state  $\mathbf{x}_k^i$  using a MIKF, with the measurement  $\mathbf{y}_k^i$  incorporated in the relaxed estimate using a Kalman gain  $\mathbf{K}_k^i$ . Assume that there exists a global estimate  $\mathcal{N}(\hat{\mathbf{x}}_{k|k}^g, \mathbf{P}_{k|k}^g)$  where the measurement  $\mathbf{y}_k$  is incorporated by a Kalman-type filter with the Kalman gain  $\mathbf{K}_k^g$ . Define  $\hat{\mathbf{x}}_{k|k}^l = (\hat{\mathbf{x}}_{k|k}^1; \cdots; \hat{\mathbf{x}}_{k|k}^N)$ . Then, the cross-covariance

$$\mathbf{P}_{k|k}^{ll} \triangleq \mathbb{E}[(\mathbf{x}_k^l - \hat{\mathbf{x}}_{k|k}^l)(\mathbf{x}_k^l - \hat{\mathbf{x}}_{k|k}^l)^\top] \triangleq \begin{bmatrix} \mathbf{P}_{k|k}^{11} & \cdots & \mathbf{P}_{k|k}^{1N} \\ \vdots & \ddots & \vdots \\ \mathbf{P}_{k|k}^{N1} & \cdots & \mathbf{P}_{k|k}^{NN} \end{bmatrix}$$

can be approximated recursively by a first-order Taylor expansion of the nonlinearities in  $\mathbf{F}^i(\mathbf{x}^i)$  and  $\mathbf{h}^i(\mathbf{x}^i)$  as

$$\begin{aligned} \mathbf{A}_k^i &= \left. \frac{\partial \mathbf{F}^i(\mathbf{x}^i)}{\partial \mathbf{x}^i} \right|_{\mathbf{x}^i = \hat{\mathbf{x}}_{k-1|k-1}^i}, \quad \mathbf{C}_k^i = \left. \frac{\partial \mathbf{h}^i(\mathbf{x}^i)}{\partial \mathbf{x}^i} \right|_{\mathbf{x}^i = \hat{\mathbf{x}}_{k-1|k-1}^i}, \\ \mathbf{P}_{k|k-1}^{ij} &\approx \mathbf{A}_k^i \mathbf{P}_{k|k-1}^{ij} \mathbf{A}_k^j + \mathbf{T}_{g \rightarrow l}^i \mathbf{Q}_k (\mathbf{T}_{g \rightarrow l}^j)^\top, \\ \mathbf{P}_{k|k}^{ij} &\approx (\mathbf{I} - \mathbf{K}_k^i \mathbf{C}_k^i) \mathbf{P}_{k|k-1}^{ij} (\mathbf{I} - \mathbf{K}_k^j \mathbf{C}_k^j)^\top + \mathbf{K}_k^i \mathbf{R}_k^{ij} (\mathbf{K}_k^j)^\top. \end{aligned} \quad (31a)$$

Similarly, the cross-covariance of the joint distribution of local and global estimates is

$$\begin{aligned} \mathbf{P}_{k|k}^{lg} &\triangleq \mathbb{E}[(\mathbf{x}_k^l - \hat{\mathbf{x}}_{k|k}^l)(\mathbf{x}_k^g - \hat{\mathbf{x}}_{k|k}^g)^\top] \\ &\approx (\mathbf{I} - \mathbf{K}_k^l \mathbf{C}_k^l) (\mathbf{A}_k^l \mathbf{P}_{k-1|k-1}^{lg} (\mathbf{A}_k^g)^\top \\ &\quad + \mathbf{T}_{g \rightarrow l} \mathbf{Q}_k) (\mathbf{I} - \mathbf{K}_k^g \mathbf{C}_k^g)^\top + \mathbf{K}_k^l \mathbf{R}_k (\mathbf{K}_k^g)^\top, \end{aligned} \quad (31b)$$

where

$$\begin{aligned} \mathbf{A}_k^g &= \left. \frac{\partial \mathbf{F}(\mathbf{x})}{\partial \mathbf{x}} \right|_{\mathbf{x} = \hat{\mathbf{x}}_{k-1|k-1}^g}, \quad \mathbf{C}_k^g = \left. \frac{\partial \mathbf{h}(\mathbf{x})}{\partial \mathbf{x}} \right|_{\mathbf{x} = \hat{\mathbf{x}}_{k-1|k-1}^g}, \\ \mathbf{A}_k^l &\triangleq \text{diag}(\mathbf{A}_k^1, \cdots, \mathbf{A}_k^N), \quad \mathbf{C}_k^l \triangleq \text{diag}(\mathbf{C}_k^1, \cdots, \mathbf{C}_k^N), \\ \mathbf{K}_k^l &\triangleq \text{diag}(\mathbf{K}_k^1, \cdots, \mathbf{K}_k^N). \end{aligned}$$

**Proof 2** The result follows from writing the estimation errors of  $\mathbf{x}_k^l$  and  $\mathbf{x}_k^g$  recursively, and is a straight-forward adaptation of the derivation in [36, Appendix B]. ■

While approximate, the recursive method in (31) facilitates the implementation of (30), solving Problem **P2** but requiring

extensive bookkeeping. For future reference, we define a set of relevant inter-agent estimate error covariance matrices, as

$$\mathcal{P} = \{\mathbf{P}_{k|k}^{ij} | (i, j) \in [1, N]^2 \subset \mathbb{N}^2, i < j\} \cup \mathbf{P}_{k|k}^{lg}, \quad (32)$$

and we denote an update of the elements of this set by

$$\mathcal{P} \leftarrow \text{update\_crosscov}(k, \mathcal{P}, \mathcal{A}, \mathcal{C}). \quad (33)$$

#### D. Back-Propagating the Global Estimates

When the FC has determined an updated fused state estimate  $\hat{\mathbf{x}}^{f,i}$  and associated covariance  $\mathbf{P}^{f,i}$  for agent  $\mathcal{A}_i$ , this information needs to be propagated back for use in the agent  $\mathcal{A}_i$ . The straightforward approach is to submit back the fused state estimate and covariance. Sometimes this is not the statistically correct approach, since typically the FC and local agent update at different update rates and with different sensor information. This implies that; (i) the fused estimate may correspond to a different time compared to the current time, such that there needs to be time-correlation of the fused estimate and the current time; (ii) the fused estimate may differ from the local estimate such that direct application of the fused estimate can cause large jumps, which is unsuitable when using GNSS in a control system; and (iii), since the local agent estimator relies on recursive updates from a previous estimate according to systematic Bayesian principles, when overwriting the local estimator with the fused estimate, such Bayesian principles are violated. Instead, the statistically correct approach is to let the local agent treat the fused estimate as an extra, virtual measurement on the local states, that, when used in the Kalman update in absence of other measurements, causes the local agent estimates to converge to the global estimate. When the local agents utilize additional onboard sensing, as will typically be the case, the virtual measurements acts as yet another measurement update. This is summarized in Proposition 3.

**Proposition 3** Consider the system

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{q}_k, \quad \mathbf{q}_k \sim (\mathbf{0}, \mathbf{Q}_k), \quad (34a)$$

$$\mathbf{z}_k = \mathbf{x}_k + \mathbf{r}_k, \quad \mathbf{r}_k \sim (\mathbf{0}, \mathbf{R}_k). \quad (34b)$$

Then, for any distribution  $p(\mathbf{x}_{k-1} | \mathbf{z}_{0:k-1}) = \mathcal{N}(\boldsymbol{\mu}_{k-1}, \boldsymbol{\Sigma}_{k-1})$ , there exists a set of measurement realization and variances  $\{\mathbf{z}_k, \mathbf{Q}_k, \mathbf{R}_k\}$  such that  $p(\mathbf{x}_k | \mathbf{z}_{0:k}) = \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$  for any sought posterior  $\{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}$ ,

$$\mathbf{z}_k = \boldsymbol{\mu}_{k-1} + (\boldsymbol{\Sigma}_{k-1} + \mathbf{Q}_k + \mathbf{R}_k) \mathbf{N}_k^{-1} (\boldsymbol{\mu}_k - \boldsymbol{\mu}_{k-1}), \quad (35a)$$

$$\mathbf{R}_k = -(-\boldsymbol{\Sigma}_k^{-1} + \mathbf{N}_k^{-1})^{-1}, \quad (35b)$$

$$\mathbf{N}_k = \boldsymbol{\Sigma}_{k-1} + \mathbf{Q}_k. \quad (35c)$$

Furthermore, the smallest positive semi-definite  $\mathbf{Q}_k \succeq 0$  that yields a positive definite  $\mathbf{R}_k \succ 0$  is

$$\mathbf{Q}_k = \boldsymbol{\Sigma}_k - \boldsymbol{\Sigma}_{k-1} + |\min(\lambda(\boldsymbol{\Sigma}_k - \boldsymbol{\Sigma}_{k-1}), 0)| \mathbf{I}. \quad (36)$$

**Proof 3** The proof follows from the Woodbury matrix identity.

We denote the generation of a virtual measurement using Proposition 3 by

$$\mathbf{z}_k^i \leftarrow \text{set\_virtualmeas}(\boldsymbol{\mu}_k^i, \boldsymbol{\Sigma}_k^i). \quad (37)$$

#### E. Algorithm Outline

Proposition 3 provides a way to compute a virtual measurement that translates a prior distribution in the local agent to a posterior produced by the FC. After computing a relaxed fused density based on all of the available information, we can propagate this to any agent using a virtual measurement update. Hence, Proposition 3 provides a means to feed back (back-propagate) the information from  $\mathcal{C}$  to the local agents  $\mathcal{A}_i$ . Each local agent executes with a sampling period  $\delta t = t_k - t_{k-1}$ , usually determined as the sampling period of the fastest sensor in the local agent. At each time step  $k$ , the local agent transmits its estimated state and code and carrier-phase measurements to the global agent  $\mathcal{C}$ . The global agent, however, executes with a sampling period  $\Delta t > \delta t$ , by virtue of solving a high-dimensional estimation problem. The combination of different timings, larger sampling period, and communication times in the central computing unit, can lead to a delay from when the measurements and estimates from each agent  $i$  have been gathered, until the updated virtual measurements  $\mathbf{z}^i$  has been transmitted from the FC  $\mathcal{C}$  to an agent  $\mathcal{A}_i$ . In such a case, the virtual measurement can be an out-of-sequence measurement (OOSM), which is a measurement that arrives after more recent measurements have already been processed. While it is possible to have OOSMs both in the communication from FC to agents, and from agents to FC, we primarily focus on the former as the FC is likely to be run at lower rates.

There are numerous ways to incorporate an OOSM. If the local agents store all measurements and the computational power is large enough, the optimal solution is to rerun the filter from time  $t_\tau$  up until the current time  $t_k$  including the virtual measurement  $\mathbf{z}_\tau$ , before the next measurement arrives. However, in many cases this is impractical, both because of computational limitations and because storing large buffers of measurements is not supported in the hardware. In this case, there are multiple solutions available, which are optimal in the linear case and readily extendable to the nonlinear setting (e.g., [37]–[40]). These methods essentially amount to different versions of smoother implementations.

While we are primarily concerned with the communication and back-propagation of information from the FC to each agent, we also note that a similar smoothing approach can be taken if there are delays in the communication from the local agents to the global FC. Here, an OOSM update can be done along the lines of [37]–[40] instead of conducting in-sequence filtering. However, caution needs to be taken with respect to the updating of the cross-covariances in (33) using Proposition 2. Here, it is advised to run the global FC at a slight delay, and perform a forward pass incorporating the available measurements prior to the BC fusion in (30).

Algorithm 1 summarizes the proposed cooperative positioning solution for the case where the back propagation is done with a virtual measurement but without OOSMs. We emphasize that the last step in the algorithm can be implemented in various ways using OOSMs, but the exact choice of method will depend on the nature of the communication errors in a real implementation. Since the choice of OOSM approach will depend on the delays, local computational complexity,

additional sensing used in the local estimators, and memory capabilities of the receivers, we validate Algorithm 1 in simulations without OOSMs. The effect of introducing OOSMs, depending on the method used will generally result in a slight performance degradation, an increase of computational burden, or both, but it is not a limitation of the method in general.

---

**Algorithm 1** The cooperative estimation algorithm.

---

```

Initialize:  $\{\hat{\mathbf{x}}_{-1|-1}^j, \mathbf{P}_{-1|-1}^j\}_{j=1}^N$ 
1: for  $k = 0, 1, \dots$  do
2:   for  $i = 1, \dots, N$  do
3:      $\mathcal{A}_i \leftarrow \text{MIKF\_update}(\mathcal{A}_i, \mathbf{y}_k^i)$ 
4:   end for
5:   Transmit  $\{\mathbf{y}_k, \hat{\mathbf{x}}_{k|k}^i, \mathbf{P}_{k|k}^i\}_{i=1}^N$  from  $\mathcal{A}$  to  $\mathcal{C}$ 
   // Estimate inter-agent cross-covariance
6:    $\mathcal{P} \leftarrow \text{update\_crosscov}(k, \mathcal{P}, \mathcal{A}, \mathcal{C})$ 
   // Fuse and back propagate
7:   if Fuse and back propagate then
   // Compute MSE-optimal fusion of estimates
8:    $[\boldsymbol{\mu}_k^f, \boldsymbol{\Sigma}_k^f] \leftarrow \text{gbc\_fusion}(\boldsymbol{\mu}_k^p, \boldsymbol{\Sigma}_k^p, \mathbf{T}_{g \rightarrow l})$ 
9:   for  $i = 1, \dots, N$  do
   // Define virtual measurement update
10:     $\mathbf{z}_k^i \leftarrow \text{set\_virtualmeas}(\boldsymbol{\mu}_k^i, \boldsymbol{\Sigma}_k^i)$ 
11:     $\mathcal{A}_i \leftarrow \text{MIKF\_update}(\mathcal{A}_i, \mathbf{z}_k^i)$ 
12:   end for
13: end if
14: end for

```

---

#### IV. EVALUATION PRELIMINARIES

To evaluate the proposed cooperative estimation algorithm, we present results from two urban-driving vehicle simulation studies, as well as a simulation study with HIL. The first study in Sec. V-A is conducted without back-propagation (i.e., Algorithm 1 without using virtual measurements) of the global estimates to the local agents. The purpose is to demonstrate the advantage of using global information by the reduction of the variance of the estimates of the global, relaxed solution when including known global noise correlations, and highlight performance gains that can be made in the cooperative positioning. The second study in Sec. V-B includes the fusion and back-propagation (i.e., the complete Algorithm 1) of information from the FC to the agents, showing an improvement in the local estimate each time a virtual measurement is included, demonstrating a clear performance improvement among the agents participating in the cooperative positioning. In the second study we show how the cooperative positioning can significantly increase performance of agents that have faulty and inconsistent ionospheric bias models.

##### A. Performance Metrics

There are various metrics in which the estimates can be measured. Here, we consider an empirical mean-square error (MSE) of the estimates, computed from  $N_{MC} = 100$  Monte-Carlo (MC) runs. Let  $\mathcal{N}(\hat{\mathbf{x}}_{k|k}^{(j)}, \mathbf{P}_{k|k}^{(j)})$  be the Gaussian posterior distribution in any of the local, global or fused, relaxed or fixed

estimates at time step  $k$  as computed in stimulation  $j$ . Then, we define

$$\text{MSE}(\hat{\mathbf{x}}_k) \triangleq \frac{1}{N_{MC}} \sum_{j=1}^{N_{MC}} \|\mathbf{x}_{k|k}^{(j)} - \hat{\mathbf{x}}_{k|k}^{(j)}\|_2^2. \quad (38)$$

This enables investigating estimate consistency. To this end, we define a measure of average posterior covariance (APC),

$$\text{APC}(\mathbf{P}_k) \triangleq \frac{1}{N_{MC}} \sum_{j=1}^{N_{MC}} \text{Trace}(\mathbf{P}_{k|k}^{(j)}). \quad (39)$$

If the filter is consistent, we should expect that  $\text{MSE} \approx \text{APC}$  when  $N_{MC}$  is large enough. Additionally, as the proposed solution entertains relaxed, fixed, and fused estimates of high-dimensional state-vectors, we define a convenient scalar error measure. If a simulation contains  $K$  time steps, we let  $\mathcal{E}(\hat{\mathbf{x}}) \triangleq K^{-1} \sum_{k=1}^K \text{MSE}(\hat{\mathbf{x}}_k)$  denote a time-averaged MSE. Finally, we consider a relative average empirical MSE of two estimates  $\hat{\mathbf{x}}^i$  in  $\mathcal{A}_i$  and  $\mathbf{T}_{g \rightarrow l}^i \hat{\mathbf{x}}$  in  $\mathcal{C}$ , as  $\mathcal{R}(\hat{\mathbf{x}}^i) \triangleq \mathcal{E}(\hat{\mathbf{x}}^i) / \mathcal{E}(\mathbf{T}_{g \rightarrow l}^i \hat{\mathbf{x}})$ .

##### B. Evaluation Setup

We consider an  $N = 10$  agent scenario. For simplicity, we restrict the scope to the GPS satellite constellation in the evaluation. Of all 32 satellites in the GPS constellation, only those with an elevation angle greater than 15deg are considered possibly visible and constitute  $\mathcal{S}$ , resulting in  $|\mathcal{S}| = 9$ . Of these, a random subset of the satellites is seen by the agent  $\mathcal{A}_i$ , resulting in a time-varying  $\mathcal{S}_i$ , thereby simulating occultations that may occur in an urban environment. The setup leads to 159 states in  $\mathbf{x}_k^g$  and 390 unique measurements in the global measurement vector  $\mathbf{y}_k^g$ . Synthetic data are generated over  $T = 200$ s and sampled at 10Hz, with the nominal parameters stated in Appendix B. The estimation model is given by (18) and (19), and for simplicity a (nearly) CV model in (21) is used in all agents. However, for data generation, (i) the true kinematic states of agent  $\mathcal{A}_i$ , with positions and velocities  $\boldsymbol{\kappa}_k^i = (\mathbf{p}_k^i; \mathbf{v}_k^i) \in \mathbb{R}^6$ , are sampled from a car movement simulator, (ii) the ambiguities evolve by an integer random walk (see [23, Sec. IV.B]), and (iii) the ionospheric biases,  $\boldsymbol{\theta}_k^{I,i}$ , are sampled from an ionospheric delay simulator. A complete description of the parameter variation introduced in each Monte-Carlo simulation is summarized in Appendix B.

1) *Car Simulator:* To generate the synthetic data, a set of 10 agents  $\{\mathcal{A}_i\}_{i=1}^N$  are initialized randomly on the streets of Boston, USA, and made to follow the shortest route to a random destination (see Fig. 6). The agents are constrained to follow routes extracted using the open-source routing machine (OSRM) tool [41], with a path velocity in a local Cartesian ENU-frame sampled from a Gaussian distribution, with a mean velocity dictated by the path curvature. For this purpose, let  $\mathbf{p}_{ENU}^i : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^3$  be a function that maps a positive path distance to a coordinate in the ENU frame, and let  $c : \mathbb{R}_{\geq 0} \rightarrow [0, \infty)$  denote a measure of curvature of a path as a function of the path distance (0 being a straight path). The agents are assigned velocity references

$$\mathbf{v}_{r,k}^i = \begin{cases} v_{min} & \text{if } \max_{d \in [d_k^i, d_k^i + T_h v_{k-1}^i]} c(\mathbf{p}_{ENU}(d)) > c_{max} \\ v_{max} & \text{otherwise,} \end{cases} \quad (40)$$

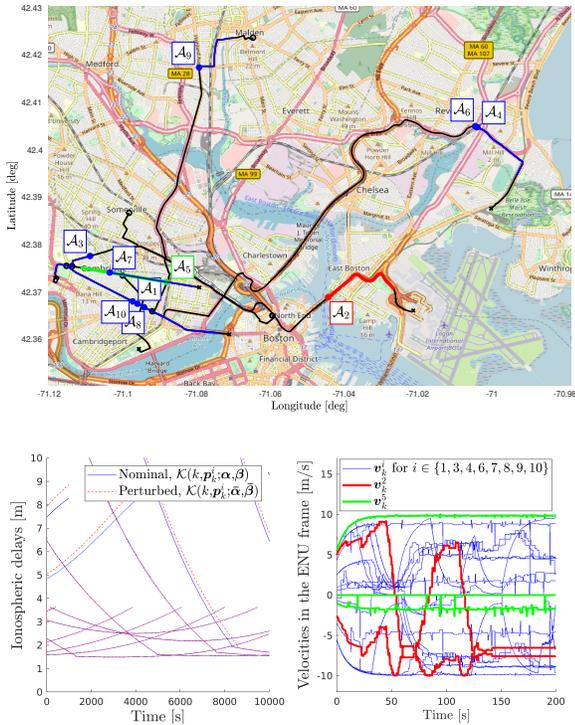


Fig. 6. *Top*: Movement of the vehicles in time over the Boston road network, with paths  $\mathbf{p}_{ENU}^i$  (black) and trajectories of the vehicles with a tag  $\mathcal{A}_i$  identifying each agent. The red color indicates the path of agent  $\mathcal{A}_2$ , and the green color indicates the trajectory of agent  $\mathcal{A}_5$ . *Bottom, left*: The nominal and perturbed ionospheric biases from the Klobuchar model parameterized as in Sec. IV-B, corresponding to agent  $\mathcal{A}_5$ . *Bottom, right*: The velocity profiles of the agents in the local ENU frame with the velocities of agent  $\mathcal{A}_2$  changing significantly in time, violating the assumption of constant velocity, and the velocities of agent  $\mathcal{A}_5$ , being approximately constant in time.

and the mean dynamics of the agent path velocity is

$$v_k^i = v_{k-1}^i + (h/\tau)(v_{r,k-1}^i - v_{k-1}^i), \quad (41)$$

where the path distance at time step  $k$  is sampled as

$$d_k^i \sim \mathcal{N}(d_{k-1}^i + hv_{k-1}^i, \sigma_v^2). \quad (42)$$

The kinematic state  $\kappa^i$  of the agent  $\mathcal{A}_i$  at time step  $k$  is defined by the position  $\mathbf{p}_{ENU}^i(d_k^i)$ , and the driving behavior of the vehicles is defined by  $\{T_h, \tau, \sigma_v, v_{min}, v_{max}, c_{max}\}$ . In the evaluation, we use a horizon length of  $T_h = 10s$ , with  $\tau = 10s$ ,  $\sigma_v = 0.1m/s$ ,  $v_{min} = 1m/s$ , and  $v_{max} = 10m/s$ . This results in the behavior in Fig. 6, which clearly differs from the predicted behavior by the CV motion model. Hence, our evaluation accounts for a realistic modeling mismatch.

2) *Ionospheric Delay Simulator*: To generate realistic ionospheric delays for the measurements of different agents, the movement of the GPS constellation is simulated forward in time using real ephemeris data. The true ionospheric biases are generated from a Klobuchar model [24], denoted by  $\mathcal{K} : \mathbb{R}^{12} \rightarrow \mathbb{R}_{>0}$ . This function takes a time index, a position, and a set of parameters; in this paper either a set of nominal parameters  $(\alpha, \beta)$  corresponding to early January of 2007 in [42], or a set of perturbed parameters,  $(\tilde{\alpha}, \tilde{\beta})$ , given in

Appendix B. Hence,  $\mathcal{K}$  models an ionospheric delay of a satellite  $s \in \mathcal{S}_i$  at a time step  $k$ , with the agent position  $\mathbf{p}_k^i$ ,

$$I_{i,k}^s = \mathcal{K}(k, \mathbf{p}_k^i; \alpha, \beta). \quad (43)$$

This model is commonly used to correct for ionospheric delays, but its parameters are never perfectly known. As such, the correction terms in the receivers are simulated using a set of perturbed Klobuchar parameters  $(\tilde{\alpha}, \tilde{\beta})$ . In the synthetic data, the true ionospheric delay difference terms do not evolve by the random walk as assumed in the estimation model (18), (19). Rather, the ionospheric bias term evolves by

$$I_{i,k}^s = \mathcal{K}(k, \mathbf{p}_k^i; \alpha, \beta) - \mathcal{K}(k, \mathbf{p}_k^i; \tilde{\alpha}, \tilde{\beta}), \quad (44)$$

with the global bias residual vector  $\theta_k^I$  in the synthetic data defined as in (4a). Similarly to the difference between the true vehicle dynamics and the simplified model in the estimator, we should expect a prediction error in the ionospheric biases residuals to degrade the resulting MSE of the estimates.

## V. NUMERICAL RESULTS

Here, we present the results using the simulation setup described in Sec. IV. First, we present the results from a simulation study where we compare the estimation difference between having global and local information. Then, we show the simulation results when including back-propagation of the global information to the local agents. Finally, we conclude the evaluation by presenting results including HIL data.

### A. Cooperative Estimation Without Back-Propagation

To demonstrate the benefit of including global in relation to only using local information (i.e., a conventional GNSS positioning approach), the system states are collaboratively estimated without back-propagation, cutting off Algorithm 1 after the generalized BC-fusion step. The cooperative estimation scheme is subsequently run with EKF moment approximations, using adaptive ambiguity priors and cycle-slip detection as described in [23], implementing the modified LAMBDA decorrelation scheme in [43] with a bootstrapping method [9].

Since the fusion is based on the estimated variance, an important aspect of the estimator is its ability to produce consistent estimates, that is, that the estimated confidence is consistent with the true confidence. To study this, we compare the empirical MSE and APC of the estimates in the MC simulation study. Fig. 7 shows the estimates of the states in agent  $\mathcal{A}_1$ . The MSE and APC are plotted for the relaxed estimates  $\{\hat{\mathbf{p}}_k^1, \hat{\mathbf{v}}_k^1, \hat{\mathbf{n}}_k^1, \hat{\boldsymbol{\theta}}_k^{I,1}, (\hat{\boldsymbol{\theta}}_k^{P,1}, \hat{\boldsymbol{\theta}}_k^{C,1})\}$  (having real-valued ambiguities) and for the fixed estimates  $\{\check{\mathbf{p}}_k^1, \check{\mathbf{v}}_k^1, \check{\mathbf{n}}_k^1, \check{\boldsymbol{\theta}}_k^{I,1}, (\check{\boldsymbol{\theta}}_k^{P,1}, \check{\boldsymbol{\theta}}_k^{C,1})\}$  (when fixing the ambiguities to integers) in  $\log_{10}$  scale. The relaxed estimates are roughly consistent except for the the velocity estimates. This is due to the expected modeling errors, as the simple CV motion model does not perfectly capture receiver motion governed by the car simulator. However, when investigating the fixed estimate density in the lower plot, the estimates no longer seem to be consistent in the second moment of the position estimates. One explanation for this is that the statistical assumption that the ILS problem is solved exactly in the MIKF is violated

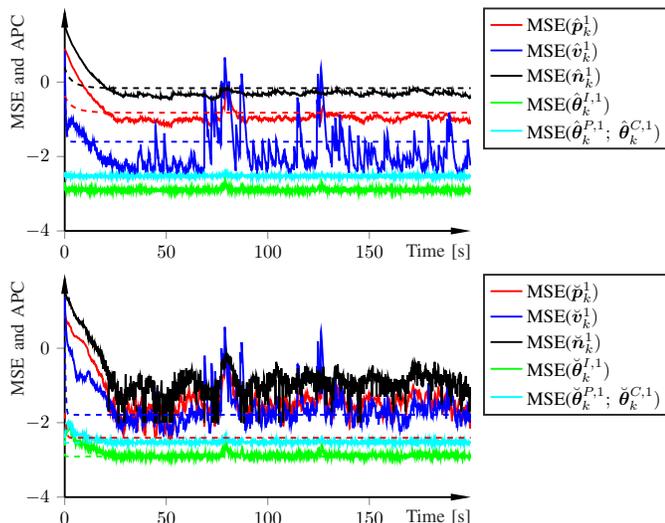


Fig. 7. Estimate MSE in time with associated APC (solid and dashed, respectively) of the states in agent  $\mathcal{A}_1$ . *Top*: Relaxed estimate density. *Bottom*: Fixed estimate density.

in practice, as it is an NP-hard problem, which is actually solved by heuristic methods. Hence, we should expect to have some variance in the fixed ambiguity estimate in time. This is clearly seen by that we do not achieve a perfect ambiguity fixation (the empirical MSE of ambiguity estimates in black is nonzero in Fig. 7, lower plot). From Fig. 7, we conclude that the relaxed estimates are consistent, motivating the approach of fusing the relaxed estimates prior to fixation, as opposed to fusing the fixed estimates. Also, note that in the local filters, the fixed estimates yield a significant improvement in the empirical MSE, with a tendency to under-estimate the estimate error variance. Again, this is only shown for agent  $\mathcal{A}_1$  in Fig. 7 but similar conclusions can be drawn for the other agents.

When evaluating the complete solution in Algorithm 1, the FC essentially estimates the Cramér-Rao performance bound. Hence, when investigating performance improvements in the local agents, the FC is the benchmark to compare against. To relate Fig. 7 to the other agents and how the introduction of global information increases performance, we summarize the results from the MC-study in Table I, which shows the relative performance decrease for the relaxed estimates relative to global information when using only local information. There is a significant variation in the performance decrease between global and local information for the different agents, especially in the kinematic states. One reason is the large prediction errors arising from the relatively simple CV prediction model. For instance, agent  $\mathcal{A}_2$ , which from Table I gains a lot from using global information, takes many sharp turns with significant changes in the velocities, therefore violating the assumption of a near-constant velocity in (21). Agent  $\mathcal{A}_5$ , on the other hand, follows Massachusetts avenue, a wide and fairly straight road, at a near constant speed throughout the simulation, and its estimate MSE is much lower as a result. Hence, there is less performance decrease when only relying on local information. From Table I, it is clear that significant performance gains are possible by utilizing the global information and modeling shared biases in the global state

TABLE I  
RELATIVE AVERAGE EMPIRICAL MSE IN  $\mathcal{A}$  AND  $\mathcal{C}$  FOR THE EXAMPLE WITH GOOD BIAS MODELS, NO BACK-PROPAGATION.

Agent	$\mathcal{R}(\hat{p}^i)$	$\mathcal{R}(\hat{v}^i)$	$\mathcal{R}(\hat{n}^i)$	$\mathcal{R}(\hat{\theta}^{I,i})$	$\mathcal{R}(\hat{\theta}^{C,i})$	$\mathcal{R}(\hat{\theta}^{P,i})$
$\mathcal{A}_1$	1.00	1.48	0.98	1.70	1.74	1.70
$\mathcal{A}_2$	1.83	1.45	1.10	3.39	1.74	1.74
$\mathcal{A}_3$	1.32	1.91	1.23	2.75	1.74	1.74
$\mathcal{A}_4$	1.62	1.70	1.35	2.00	1.74	1.70
$\mathcal{A}_5$	1.12	1.15	1.32	1.70	1.74	1.70
$\mathcal{A}_6$	1.66	1.45	1.51	2.04	1.74	1.70
$\mathcal{A}_7$	1.32	1.12	1.48	1.66	1.74	1.70
$\mathcal{A}_8$	1.12	1.12	1.26	1.78	1.74	1.70
$\mathcal{A}_9$	1.35	2.00	1.12	2.69	1.74	1.74
$\mathcal{A}_{10}$	1.32	1.48	1.35	1.86	1.74	1.70

vector, especially for the difficult cases where the estimates degrade the most. For instance, in the problematic vehicle  $\mathcal{A}_2$ , the empirical RMSE of the ionospheric residuals is 0.1229m when using local information, compared with 0.0671m when using global information. However, an even greater strength of the cooperative positioning solution is its correction of poor and inconsistently set bias models in the various agents, as will be shown in the next section.

### B. Cooperative Estimation With Back-Propagation

In this subsection, we run Algorithm 1 including the estimate fusion proposed in Sec. III-B using generalized BC fusion from the inter-agent covariance estimates computed using the recursion proposed in Sec. III-C. The global information is subsequently back-propagated to the local agents at a rate of 2Hz, while the filters in the agents are run at 10Hz. By modeling shared bias states in the global model, enabled by Assumption 1, locally faulty correction terms can be adjusted using the global information to enhance the estimates. For instance, if agent  $\mathcal{A}_1$  has a faulty ionospheric bias correction term  $I_1^s$  for some  $s \in \mathcal{S}_1$ , and there exists other agents participating in the cooperative positioning for which this satellite is also visible, we should expect the estimation performance of  $\mathcal{A}_1$  to be significantly improved by the cooperative positioning. To this end, we introduce a local correction term in  $\mathcal{A}_1$  that is significantly different from the true bias, which is accurately estimated in the other participating agents. We introduce the faulty correction term in  $\mathcal{A}_1$  by adding a 0.3m effect of ionospheric delay to one of the biases corresponding to satellite  $s_\delta = 2 \in \mathcal{S}_1$ , as

$$I_{1,k}^{s_\delta} \triangleq \mathcal{K}(k, \mathbf{p}_k^i; \boldsymbol{\alpha}, \boldsymbol{\beta}) - \mathcal{K}(k, \mathbf{p}_k^1; \tilde{\boldsymbol{\alpha}}, \tilde{\boldsymbol{\beta}}) + 0.3, \quad (45)$$

in generating the synthetic data for agent  $\mathcal{A}_1$ .

Fig. 8 displays the estimation performance of  $\mathcal{A}_1$  both without (upper plot) and with (lower plot) back-propagation of the global information to the local agent. Without back-propagation, the estimation performance of the fixed density is clearly inconsistent with the estimated  $2\sigma$  confidence interval, and the empirical RMSE of the ionospheric bias estimates in agent  $\mathcal{A}_1$  is roughly 0.3m, about the size of the error introduced in (45). However, when back-propagating the global estimates to the local agent, the error is significantly reduced

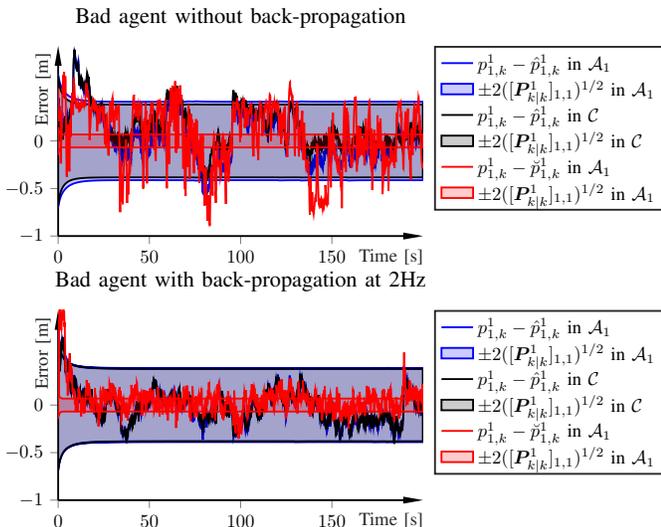


Fig. 8. The mean estimate position error for one Monte-Carlo run in the E-direction of agent  $\mathcal{A}_1$  when its ionospheric bias model has an error of 0.3m and inconsistent with the remaining agents. Filled areas are estimated  $2\sigma$ -confidence interval for the local error (blue), global error (black), and local fixed error (red). *Top*: Example of typical performance without back-propagation. *Bottom, left*: Example of performance with back-propagation.

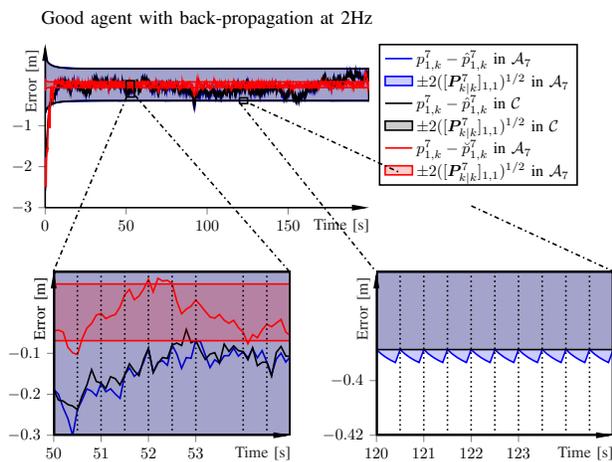


Fig. 9. Mean estimate position error in one of the Monte Carlo simulations with an estimated  $2\sigma$ -confidence interval for the local error (blue), global error (black). *Top*: The position estimate error in the E-direction of agent  $\mathcal{A}_7$ . *Bottom, left*: Zoom on the estimate errors in  $\mathcal{A}_7$  and  $\mathcal{C}$ , which are identical on each back-propagation. *Bottom, right*: Zoom on the estimate variance, which is identical on each back-propagation, before slowly increasing toward the asymptotic variance of the estimate in the local measurement information.

and of the same size as the estimated  $2\sigma$  confidence interval, around 0.08m, a visible and clear performance increase.

To illustrate the back-propagation of global information to the local agents, Fig. 9 shows the positioning performance of one of the agents with good bias estimates,  $\mathcal{A}_7$ . The relaxed estimate mean and variance in  $\mathcal{A}_7$  (black) and  $\mathcal{C}$  (blue) are equal each time the estimates are fused and back-propagated, and the variance in the local agent increases between each back-propagation. Performance of the collaborative estimation improves with the rate at which the back-propagation is done.

The possible performance gains in of the estimates entertained in the agents mainly depend on how the Cramér-Rao

TABLE II  
RELATIVE AVERAGE EMPIRICAL MSE IN  $\mathcal{A}$  AND  $\mathcal{C}$  FOR THE EXAMPLE WITH BAD BIAS MODELS IN  $\mathcal{A}_1$ , NO BACK-PROPAGATION.

Agent	$\mathcal{R}(\hat{p}^i)$	$\mathcal{R}(\hat{v}^i)$	$\mathcal{R}(\hat{n}^i)$	$\mathcal{R}(\hat{\theta}^{I,i})$	$\mathcal{R}(\hat{\theta}^{C,i})$	$\mathcal{R}(\hat{\theta}^{P,i})$
$\mathcal{A}_1$	1.79	1.17	5.29	55.78	1.72	1.78
$\mathcal{A}_2$	1.09	1.25	1.08	0.76	1.74	2.04
$\mathcal{A}_3$	1.11	1.25	1.07	0.72	1.74	1.91
$\mathcal{A}_4$	1.52	1.21	1.22	0.74	1.73	1.83
$\mathcal{A}_5$	1.43	1.10	1.26	0.68	1.72	1.81
$\mathcal{A}_6$	1.14	1.21	1.02	0.74	1.72	1.83
$\mathcal{A}_7$	1.10	1.10	0.97	0.68	1.73	1.83
$\mathcal{A}_8$	1.00	1.07	0.97	0.79	1.73	1.83
$\mathcal{A}_9$	1.26	1.28	1.19	0.73	1.74	1.94
$\mathcal{A}_{10}$	0.94	1.13	0.91	0.78	1.72	1.79

bound differs when using global and local information, and implicitly on the noise levels of the system, the amount of inter-agent cross-correlation, and the sampling periods of the estimators. To investigate this more rigorously, we show the results from a MC study of 100 runs without back-propagation in Table II and with back-propagation at 2Hz in Table III. From these two tables, some conclusions can be drawn. First, without propagating back the global estimates,  $\mathcal{A}_1$ , which has the erroneous bias model, has 1.79 times worse positioning performance than the global agent, but after back-propagation, the difference has shrunk to 1.09. Second, we note a significant performance increase of the positional empirical MSE when enabling the back-propagation of information. This holds across most agents, but is especially true for the agent with a poor ionospheric bias model. Consequently, even when there are unmodeled errors in the correction terms, back-propagation rates as low as 2Hz can yield significant performance improvements in the position estimates of the local agents. Third, two out of the ten agents ( $\mathcal{A}_8$  and  $\mathcal{A}_{10}$ ) have a minor increase in positional MSE. This can be attributed to the fact that the simulation includes significant modeling errors, both in the prediction model and by the inconsistent bias models introduced for agent  $\mathcal{A}_1$ . It could also be an indicator that more MC runs are required for the case of inconsistent bias models. As the rate of back-propagation approaches the rate at which the local filters are run, the entries in Table III will approach 1 in every cell, and if making it arbitrarily slow the entries will approach those in Table II. Thus, even in the cases where (i) the back-propagation rates are relatively slow, (ii) there are significant modeling errors, and (iii) some of the agents have inconsistent bias models violating Assumption 1, a vast majority of the agents still benefit significantly from participating in the cooperative positioning.

In the considered simulation, the average computational time is 2ms/time step in each agent, increasing to 40ms/time step in the global model (when implemented in nonoptimized Matlab code and run on a 2.80GHz 11th Gen Intel i7). This makes real-time execution at 10Hz possible, but additional computational overhead may be present in practice, making the algorithm at least quasi-real-time. Due to the computational complexity scaling with the number of states in the global model cubed, it is advised to partition the agents into smaller subsets using the clustering method in [31] for deployment at

TABLE III  
RELATIVE AVERAGE EMPIRICAL MSE IN  $\mathcal{A}$  AND  $\mathcal{C}$  FOR THE EXAMPLE WITH BAD BIAS MODELS IN  $\mathcal{A}_1$ , WITH BACK-PROPAGATION AT 2Hz.

Agent	$\mathcal{R}(\hat{p}^i)$	$\mathcal{R}(\hat{v}^i)$	$\mathcal{R}(\hat{n}^i)$	$\mathcal{R}(\hat{\theta}^{I,i})$	$\mathcal{R}(\hat{\theta}^{C,i})$	$\mathcal{R}(\hat{\theta}^{P,i})$
$\mathcal{A}_1$	1.09	1.67	1.52	22.79	1.58	2.81
$\mathcal{A}_2$	1.06	1.11	1.02	1.74	1.59	1.77
$\mathcal{A}_3$	1.08	1.14	0.99	0.62	1.58	1.69
$\mathcal{A}_4$	1.05	1.13	1.00	0.58	1.58	1.68
$\mathcal{A}_5$	1.03	1.11	1.02	0.54	1.58	1.69
$\mathcal{A}_6$	1.02	1.16	1.01	0.55	1.59	1.69
$\mathcal{A}_7$	1.01	1.09	1.03	1.60	1.59	1.63
$\mathcal{A}_8$	1.03	1.05	1.03	1.63	1.59	1.66
$\mathcal{A}_9$	1.08	1.14	1.00	0.59	1.58	1.73
$\mathcal{A}_{10}$	1.06	1.08	1.01	0.60	1.59	1.65

a larger scale.

### C. Hardware-in-the-Loop Simulation Study

To demonstrate and test the validity of Assumptions 1–3, we provide an HIL simulation study where the output of the car simulator in Sec. IV-B1 is passed to the Orolia GSG-5/6 series GNSS hardware simulator [25], as outlined in Fig. 10. The Orolia device takes the position trajectories of the receivers as a set of NMEA files, and runs a simulation using the u-blox ZED-F9P chip set. Here, we have no knowledge of the exact models and noise levels used by the Orolia device, and each simulation for the 10 agents and the single base station is done independently. The output of the HIL simulator is a set of satellite trajectories in a GSG format [25] sampled at 1Hz, GNSS observation and navigation data to a RINEX format [44] sampled at approximately 10Hz, and a broadcast file (BRDC) containing the parameters of a Klobuchar model [24]. As each receiver is simulated independently in the Orolia device, the time-stamping in the RINEX observation files is nonuniform. This implies that the satellite trajectories are not the same in the different agents in the GSG data. Consequently, we preprocess the GSG and RINEX data to associate each instance of measurement and satellite position with a time step in the estimation algorithm at 10Hz. Furthermore, in this processing, we also apply a measurement mask to remove the DD measurements based on a threshold on an elevation angle and the signal-to-noise (SNR) ratio associated with the GNSS measurements. Both of these steps (re-sampling of the satellite trajectories and measurement masking) are necessary in real applications to achieve good positioning performance.

From this preprocessing step, we obtain a set of local DD measurements at 10Hz for each agent. The correction terms are computed using the delay simulator in Sec. IV-B2, and the cooperative positioning scheme in Algorithm 1 is subsequently run on the data corresponding to the GPS constellation and the  $L_1$  frequency band. As the noise statistics used to generate the data are unknown, the parameters of the filters are hand tuned and modified slightly from those used in the previous simulations. In particular, the measurement noise associated with the code measurements is reduced. The output of the local receivers are compared with the ground truth in an ENU frame, yielding the positioning errors summarized in Table IV.

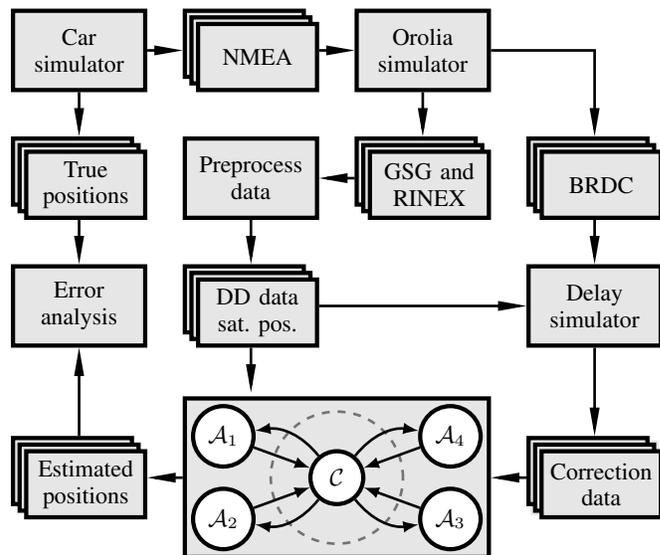


Fig. 10. Sketch of the setup and information flow in the HIL simulations.

TABLE IV  
POSITIONING RMSE FOR THE DIFFERENT AGENTS WITH COOPERATIVE POSITIONING ON HIL DATA WITH BACK-PROPAGATION AT 2Hz.

Agent	$\sqrt{\mathcal{E}(p_{1k}^i)}$ [cm]	$\sqrt{\mathcal{E}(p_{2k}^i)}$ [cm]	$\sqrt{\mathcal{E}(p_{3k}^i)}$ [cm]
$\mathcal{A}_1$	14.1	6.0	22.7
$\mathcal{A}_2$	15.8	8.6	9.8
$\mathcal{A}_3$	10.2	8.8	31.7
$\mathcal{A}_4$	16.1	7.7	24.6
$\mathcal{A}_5$	8.9	10.5	16.6
$\mathcal{A}_6$	5.8	8.0	26.5
$\mathcal{A}_7$	15.2	6.6	27.8
$\mathcal{A}_8$	9.6	6.7	24.0
$\mathcal{A}_9$	4.5	5.9	22.5
$\mathcal{A}_{10}$	7.9	6.3	25.1
Average	<b>10.8</b>	<b>7.5</b>	<b>23.1</b>

The results in Table IV are similar to the performance seen in the quantitative simulation study using back-propagation in ideal conditions (compare the RMSE in Table IV with the fixed estimate (red) in Fig. 9). The estimation error is significantly larger in the vertical direction. This is often the case for GNSS due to the dilution of precision, but is also partly explained by discretization errors in processing the satellite trajectories. The previously mentioned GSG files are available at 1Hz, and to get the required resolution of 10Hz the trajectories are interpolated linearly, which introduces errors as the satellite trajectories are far from linear. Such errors are likely to have a greater impact on the vertical positioning performance. We emphasize that the results the HIL experiment include:

- data processing artifacts such as interpolation errors in the satellite trajectories as would be present in practice;
- time-stamping discrepancies between the measurement data in the agents resulting from the HIL simulator;
- modeling mismatches in the estimation model, as we do not know the noise statistics used in the Orolia simulator;
- known errors in the prediction model, as the car move-

ment generated by the simulator in Sec IV-B1 does not obey the CV assumption used to define the estimation model (see Fig. 6).

Despite these sources of errors, the proposed cooperative positioning method achieves high positioning performance on the data generated by the Orolia hardware in the loop simulator. Furthermore, this performance is consistent with the simulation results reported in Sec. V-A and Sec. V-B.

## VI. CONCLUSIONS

This paper presents a cooperative GNSS positioning algorithm where several participating agents collaboratively estimate a set of states. For the mobile receiver case, the state spaces of the agents are different but partially overlapping. In addition, the agents use double-difference measurements with respect to the same base station, there is significant cross-correlation in the inter-agent carrier-phase and code measurement noise. Consequently, the agents can greatly benefit from sharing information facilitated by a fusion center. The proposed algorithm fuses the relaxed estimates of the agents centrally, incorporating assumptions on shared biases and global noise models, and propagates this fused estimate back to the agents. The method improves local estimation performance with a minimal increase in the computational overhead of the receivers, since the fusion is done centrally. Such receivers are often equipped with processors with limited capabilities. We explicitly quantified the improvements for a realistic urban driving scenario with 10 agents in Sec. V-A, showing a significant improvement in the empirical MSE of the estimates when leveraging the inter-agent noise correlations and assumptions on shared biases. Furthermore, we verified the method's robustness with respect to inconsistent bias models in Sec. V-B. Despite being heavily model-based, the method is capable of correcting the locally faulty modeled biases, as Fig. 8 illustrates. Even with the various intentionally induced modeling errors, the position MSE improves in a vast majority of the agents and significantly enhanced in the agent with a poor bias model. The HIL evaluation in Sec. V-C indicates that our method performs similarly to the simulated setup, implying that some of our assumptions are realistic also when including real GNSS hardware in the loop.

In conclusion, it is possible to devise algorithms that utilize preexisting GNSS hardware and leverage additional centralized computational power and communication to enhance the local estimates in a set of participating receivers. In the proposed cooperative positioning solution, this is done safely, as each agent functions independently of the FC during a loss of communication by design. Naturally, there are difficulties in scaling up the proposed method to a much larger number of vehicles. In such a scenario, one approach is to cluster the agents into smaller groups such that the computation in the fusion center can be done in quasi real-time. To demonstrate this, future work will include the evaluation of the cooperative positioning solution in real urban driving scenarios.

## APPENDIX A PROOF DETAILS

**Proof 1 (of Proposition 1)** *The double difference operator is a linear operator by (3) as  $M^i$  and  $S^i$  are linear. Thus,*

$$S^i M^i P_{i,k} - \mathbb{E}[S^i M^i P_{i,k}] = S^i M^i (\epsilon_{\mathcal{B},k}^S - \epsilon_{i,k}^S), \quad (46a)$$

$$S^i M^i \Phi_{i,k} - \mathbb{E}[S^i M^i \Phi_{i,k}] = S^i M^i (\eta_{\mathcal{B},k}^S - \eta_{i,k}^S). \quad (46b)$$

*Therefore, as  $R_k^{\epsilon\epsilon} = \text{Cov}(\epsilon_{\mathcal{B},k}^S, \epsilon_{\mathcal{B},k}^S) = \text{Cov}(\epsilon_{i,k}^S, \epsilon_{i,k}^S)$  and  $\text{Cov}(\epsilon_{\mathcal{B},k}^S, \epsilon_{i,k}^S) = \mathbf{0}$ , standard Gaussian algebra yields*

$$R_{(i,i),k}^{PP} = \text{Cov}(S^i M^i P_{i,k}, S^i M^i P_{i,k}) \quad (47a)$$

$$= \mathbb{E}[S^i M^i P_{i,k} - \mathbb{E}[S^i M^i P_{i,k}]](\star)^\top \quad (47b)$$

$$= S^i M^i (\text{Cov}(\epsilon_{\mathcal{B},k}^S, \epsilon_{\mathcal{B},k}^S) + \text{Cov}(\epsilon_{i,k}^S, \epsilon_{i,k}^S))(M^i)^\top (S^i)^\top \\ = 2S^i M^i (R_k^{\epsilon\epsilon})(M^i)^\top (S^i)^\top. \quad (47c)$$

*For the inter-agent measurement cross-covariance, we also get terms  $\text{Cov}(\epsilon_{i,k}^S, \epsilon_{j,k}^S) = \mathbf{0}$ , yielding the cross-covariance*

$$R_{(i,j),k}^{PP} = \text{Cov}(S^i M^i P_{i,k}, S^j M^j P_{j,k}) \quad (48a)$$

$$= S^i M^i \text{Cov}(\epsilon_{\mathcal{B},k}^S, \epsilon_{\mathcal{B},k}^S)(M^j)^\top (S^j)^\top \quad (48b)$$

$$= S^i M^i R_k^{\epsilon\epsilon} (M^j)^\top (S^j)^\top, \quad (48c)$$

*as the base station is shared among the agents. The same applies to the carrier phase measurements, resulting in (14) and the block-structured noise in the proposition. ■*

**Proof 2 (of Proposition 3)** *With the premise and notation of Proposition 3, we seek a triple  $\{z_k, R_k, Q_k\}$  such that*

$$\mu_k = \mu_{k-1} + K(z_k - \mu_{k-1}) \quad (49a)$$

$$\Sigma_k = (I - K)(\Sigma_{k-1} + Q_k) \quad (49b)$$

$$K = (\Sigma_{k-1} + Q_k)(\Sigma_{k-1} + Q_k + R_k)^{-1}. \quad (49c)$$

*Let  $N_k = (\Sigma_{k-1} + Q_k)$ , then*

$$(N_k^{-1} - N_k^{-1} \Sigma_k N_k^{-1})^{-1} - N_k = R_k \quad (50)$$

*From the Woodbury matrix identity (see, e.g., [45]),*

$$(N_k^{-1} - N_k^{-1} \Sigma_k N_k^{-1})^{-1} = N_k - (-\Sigma_k^{-1} + N_k^{-1})^{-1}. \quad (51)$$

*Insertion into (50) yields*

$$R_k = -(-\Sigma_k^{-1} + N_k^{-1})^{-1} = -(-\Sigma_k^{-1} + (\Sigma_{k-1} + Q_k)^{-1})^{-1} \quad (52)$$

*Picking any  $Q_k$  of sufficient magnitude, yields  $R_k \succ \mathbf{0}$ , as*

$$\lambda(\Sigma_{k-1} + Q_k) < \bar{\lambda}(\Sigma_k) \Rightarrow -\Sigma_k^{-1} + (\Sigma_{k-1} + Q_k)^{-1} \prec \mathbf{0} \\ \Leftrightarrow R_k \succ \mathbf{0}. \quad (53)$$

*Therefore,*

$$R_k \succ \mathbf{0} \Leftrightarrow I \prec (\Sigma_{k-1} + Q_k) \Sigma_k^{-1}, \quad (54)$$

*which holds for all  $Q_k = \Sigma_k - \Sigma_{k-1} + \epsilon I$ , with  $\epsilon \geq |\min(\lambda(\Sigma_k - \Sigma_{k-1}), 0)|$ . Furthermore, we have that*

$$\mu_k = \mu_{k-1} + N_k (N_k + R_k)^{-1} (z_k - \mu_{k-1}) \Leftrightarrow \quad (55a)$$

$$z_k = \mu_{k-1} + (N_k + R_k) N_k^{-1} (\mu_k - \mu_{k-1}) \quad (55b)$$

*which always exists, as  $N_k \succ \mathbf{0}$ . In summary, a triple  $\{z_k, R_k, Q_k\}$  solving (49) is given by (52), (55b), and defining  $Q_k$  with  $\epsilon = |\min(\lambda(\Sigma_k - \Sigma_{k-1}), 0)|$ . ■*

APPENDIX B  
NOMINAL SIMULATION PARAMETERS

In each MC run, the initial estimation errors of the real valued states are sampled uniquely for each run in accordance with the filter prior, and the car movement is realized differently (the routes remain the same, but the starting point and movement along the routes differ). The initial DD ambiguities are sampled from a uniform distribution from  $-10$  to  $10$ , independently in each dimension. The realization of these initial errors is different in each MC run, and all measurement and process noise as well as the true ambiguity trajectories are realized differently in each MC run. The modeling errors with respect to the Klobuchar delay model were realized differently for each run, and this model was parameterized by

$$\alpha = (1.7; -1.7; -6.0; 7.2) \cdot 10^{-8}, \quad \tilde{\alpha} \sim \mathcal{N}(\tilde{\alpha}, 0.1 \text{diag}(|\alpha|))$$

$$\beta = (0.9; -0.9; -2.0; 5.1) \cdot 10^5, \quad \tilde{\beta} \sim \mathcal{N}(\tilde{\beta}, 0.1 \text{diag}(|\beta|))$$

where the absolute value  $|\cdot|$  is taken element-wise.

For the HIL experiments, a single simulation was carried out for each agent using the vehicle simulator, and this generated one set of GNSS measurements per agent. In this study, only the initial estimation errors were sampled differently.

REFERENCES

- [1] P. Teunissen, "A canonical theory for short GPS baselines. Part III: the geometry of the ambiguity search space," *J. Geodesy*, vol. 71, no. 8, pp. 486–501, 1997.
- [2] A. Leick, L. Rapoport, and D. Tatarikov, *GPS satellite surveying*. John Wiley & Sons, 2015.
- [3] P. Teunissen, P. De Jonge, and C. Tiberius, "The LAMBDA method for fast GPS surveying," in *Int. Symp. GPS Technology Applications*, Bucharest, Romania, 1995.
- [4] A. Khodabandeh, P. J. Teunissen, and S. Zaminpardaz, "Consensus-based distributed filtering for GNSS," *Kalman Filters: Theory for Advanced Applications*, pp. 273–304, 2018.
- [5] A. Khodabandeh and P. Teunissen, "Distributed least-squares estimation applied to GNSS networks," *Measurement Science and Technology*, vol. 30, no. 4, p. 044005, 2019.
- [6] S. Di Cairano and I. V. Kolmanovskiy, "Real-time optimization and model predictive control for aerospace and automotive applications," in *American Cont. Conf. (ACC)*. IEEE, 2018, pp. 2392–2409.
- [7] P. D. Jonge and C. Tiberius, *The LAMBDA Methods for Integer Ambiguity Estimation: Implementation Aspects*. Verlag der Delft Univ. Technol., 1996, vol. 12.
- [8] P. Teunissen, "The invertible GPS ambiguity transformations," *Manuscripta Geodaetica*, vol. 20, pp. 489–497, 1995.
- [9] P. J. Teunissen, "GPS carrier phase ambiguity fixing concepts," in *GPS for Geodesy*. Springer, 1998, pp. 319–388.
- [10] P. J. Teunissen and S. Verhagen, "The GNSS ambiguity ratio-test revisited: a better way of using it," *Survey Review*, vol. 41, no. 312, pp. 138–151, 2009.
- [11] M. Sahmoudi and R. Landry, "A nonlinear filtering approach for robust multi-GNSS RTK positioning in presence of multipath and ionospheric delays," *Selected Topics Sig. Proc.*, vol. 3, no. 5, pp. 764–776, 2009.
- [12] T. Takasu and A. Yasuda, "Kalman-filter-based integer ambiguity resolution strategy for long-baseline RTK with ionosphere and troposphere estimation," in *ION GNSS*, 2010, pp. 161–171.
- [13] T. Takasu, "RTKLIB manual ver. 2.4. 2," *RTKLIB: An Open Source Program Package for GNSS Positioning*, pp. 29–49, 2013.
- [14] B. Azimi-Sadjadi and P. Krishnaprasad, "Approximate nonlinear filtering and its application in navigation," *Automatica*, pp. 945–956, 2005.
- [15] K. Berntorp, A. Weiss, and S. Di Cairano, "GNSS ambiguity resolution by adaptive mixture Kalman filter," in *Int. Conf. Information Fusion*. IEEE, 2018, pp. 1–5.
- [16] M. Greiff and K. Berntorp, "Optimal measurement projections with adaptive mixture Kalman filtering for GNSS positioning," in *American Cont. Conf.* IEEE, 2020, pp. 4435–4441.
- [17] K. Berntorp, A. Weiss, and S. Di Cairano, "Integer ambiguity resolution by mixture Kalman filter for improved GNSS precision," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 56, no. 4, pp. 3170–3181, 2020.
- [18] A. Hauschild, P. Steigenberger, and C. Rodriguez-Solano, "Signal, orbit and attitude analysis of Japan's first QZSS satellite Michibiki," *GPS solutions*, vol. 16, no. 1, pp. 127–133, 2012.
- [19] A. Mannucci, B. Wilson, D. Yuan, C. Ho, U. Lindqwister, and T. Runge, "A global mapping technique for GPS-derived ionospheric total electron content measurements," *Radio science*, vol. 33, no. 3, pp. 565–582, 1998.
- [20] D. Ding, Z. Wang, and B. Shen, "Recent advances on distributed filtering for stochastic systems over sensor networks," *Int. J. General Systems*, vol. 43, no. 3–4, pp. 372–386, 2014.
- [21] B. Noack, J. Sijts, and U. D. Hanebeck, "Fusion strategies for unequal state vectors in distributed Kalman filtering," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 3262–3267, 2014.
- [22] B. Noack, J. Sijts, M. Reinhardt, and U. D. Hanebeck, "Decentralized data fusion with inverse covariance intersection," *Automatica*, vol. 79, pp. 35–41, 2017.
- [23] M. Greiff, K. Berntorp, S. Di Cairano, and K. J. Kim, "Mixed-integer linear regression Kalman filters for GNSS positioning," in *IEEE Conf. Control Techn. Applications*, San Diego, CA, Aug. 2021.
- [24] J. Klobuchar, "Ionospheric time-delay algorithm for single-frequency GPS users," *IEEE Trans. Aerosp. Electron. Syst.*, pp. 325–331, 1987.
- [25] Orolia, "GSG-5/6 series GNSS simulator user manual," 2019, last accessed 08-06-2023. [Online]. Available: [www.orolia.com](http://www.orolia.com)
- [26] B. Azimi-Sadjadi and P. Krishnaprasad, "Integer ambiguity resolution in GPS using particle filtering," in *Amer. Cont. Conf.*, 2001, pp. 3761–3766.
- [27] P. Teunissen, "Theory of carrier phase ambiguity resolution," *Wuhan Univ. J. Natural Sciences*, vol. 8, no. 2, p. 471, 2003.
- [28] G. Blewitt, "Basics of the GPS technique: observation equations," *Geodetic Applications of GPS*, pp. 10–54, 1997.
- [29] J. Saastamoinen, "Contributions to the theory of atmospheric refraction," *Bull. Géodésique*, vol. 105, no. 1, pp. 279–298, 1973.
- [30] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins, *Global Positioning System: theory and practice*. Springer-Verlag, 1997.
- [31] M. Greiff, S. Di Cairano, and K. Berntorp, "Dynamic clustering for gnss positioning with multiple receivers," in *2022 25th International Conference on Information Fusion (FUSION)*, 2022, pp. 1–7.
- [32] X. Rong Li and V. P. Jilkov, "Survey of maneuvering target tracking . part I: Dynamic models," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 39, no. 4, pp. 1333–1364, 2003.
- [33] S. Zhao, X. Cui, F. Guan, and M. Lu, "A Kalman filter-based short baseline RTK algorithm for single-frequency combination of GPS and BDS," *Sensors*, vol. 14, no. 8, pp. 15 415–15 433, 2014.
- [34] M. Greiff, A. Robertsson, and K. Berntorp, "Exploiting linear substructure in linear regression Kalman filters," in *Conf. Decision and Control*, 2020, pp. 2942–2948.
- [35] Y. Bar-Shalom and L. Campo, "The effect of the common process noise on the two-sensor fused-track covariance," *IEEE Trans. Aerosp. Electron. Syst.*, no. 6, pp. 803–805, 1986.
- [36] V. Shin, Y. Lee, and T.-S. Choi, "Generalized millman's formula and its application for estimation problems," *Signal Processing*, vol. 86, no. 2, pp. 257–266, 2006.
- [37] K. Zhang, X. R. Li, and Y. Zhu, "Optimal update with out-of-sequence measurements," *Trans. Sig. Proc.*, vol. 53, no. 6, pp. 1992–2004, 2005.
- [38] W. Koch and F. Govaers, "On accumulated state densities with applications to out-of-sequence measurement processing," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 47, no. 4, pp. 2766–2778, 2011.
- [39] S. Zhang and Y. Bar-Shalom, "Optimal update with multiple out-of-sequence measurements with arbitrary arriving order," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 48, no. 4, pp. 3116–3132, 2012.
- [40] K. Berntorp, A. Robertsson, and K.-E. Årzén, "Rao-Blackwellized particle filters with out-of-sequence measurement processing," *IEEE Trans. Signal Process.*, vol. 62, no. 24, pp. 6454–6467, 2014.
- [41] OSRM, "Homepage of the open source road map project," 2021, last accessed 08-06-2022. [Online]. Available: <http://www.project-osrm.org/>
- [42] C.-M. Lee, K.-D. Park, J. Ha, and S. Lee, "Generation of Klobuchar coefficients for ionospheric for simulation," *J. Astronomy and Space Sciences*, vol. 27, no. 2, pp. 117–122, 2010.
- [43] X.-W. Chang, X. Yang, and T. Zhou, "MLAMBDA: a modified LAMBDA method for integer least-squares estimation," *J. Geodesy*, vol. 79, no. 9, pp. 552–565, 2005.
- [44] RINEX Working Group, "RINEX: The Receiver Independent Exchange Format, Version 3.03," International GNSS Service, Tech. Rep., 2015. [Online]. Available: <https://files.igs.org/pub/data/format/rinex303.pdf>
- [45] K. B. Petersen, M. S. Pedersen *et al.*, "The matrix cookbook," *Technical University of Denmark*, vol. 7, no. 15, p. 510, 2008.