2012 International Conference on Indoor Positioning and Indoor Navigation, 13-15th November 2012

Fusing the Information from Two Navigation Systems Using an Upper Bound on Their Maximum Spatial Separation

Isaac Skog, John-Olof Nilsson, Dave Zachariah, and Peter Händel
KTH Royal Institute of Technology, ACCESS Linnaeus Centre, Signal Processing Lab, Stockholm, Sweden

***Abstract*—A method is proposed to fuse the information from two navigation systems whose relative position is unknown, but where there exists an upper limit on how far apart the two systems can be. The proposed information fusion method is applied to a scenario in which a pedestrian is equipped with two foot-mounted zero-velocity-aided inertial navigation systems; one system on each foot. The performance of the method is studied using experimental data. The results show that the method has the capability to significantly improve the navigation performance when compared to using two uncoupled foot-mounted systems.**

***Index Terms*—Pedestrian navigation, Inertial navigation, Constraints, Zero-velocity detection.**

I. Introduction

Currently, there is no navigation technology that, on its own, can provide a reliable, robust, and infrastructure-free solution to the problem of positioning a pedestrian in all kinds of indoor environments. Only a navigation system that acts in cooperation with a multitude of navigation technologies (i.e., sub-navigation systems) with complementary properties has the potential to fully solve this pedestrian indoor navigation problem [1]. Because navigation technologies with complementary properties are generally based upon different physical phenomena, the most favorable location of the different sub navigation systems on the body of the user also differs. For example, a zero-velocity-aided inertial navigation system is best mounted on the foot of its user, since it then becomes stationary on a regular basis whenever the user walks; a vision-aided inertial navigation system is best mounted on the shoulder of the user, since it then has a clear view of the environment in front of its user but is excited by smaller angular rates than if it is mounted on the user's head; and a ultra-wide band transceiver is best mounted on the head of its user because it is then least likely that other body parts will block the radio signals.

If not all sub-navigation systems are located on the same point of the user, they will track the position of different points of the user's body. Hence, there is a question of how to combine these navigation solutions by means of the different sub-navigation systems. Because the human body is non-rigid, the relative positions of the subsystems are not fixed and one cannot directly relate the navigation solution of one subsystem to another. However, as illustrated in Fig. 1, there is an upper limit on how spatially separated the different subsystems can be. In this paper, we will therefore propose a method for fusing the information from two non-collocated navigation systems when there exists an upper bound on how far apart the systems can be.

Part of this work has been funded by The Swedish Agency for Innovation Systems (VINNOVA).

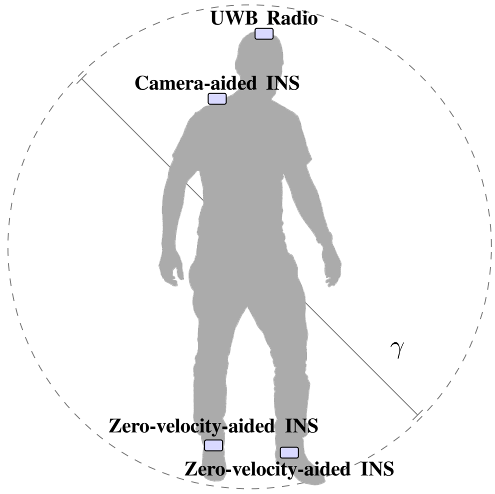

Fig. 1: Illustration of the possible placements of the subsystem in a pedestrian navigation system and the maximum spatial separation γ between the subsystems.

The outline of the paper is a follows. In Section II, we describe the problem of fusing the information from two navigation systems using prior information about their maximum separation in space, and we propose a method to solve it. In Section III, we apply the proposed method to the problem of fusing the navigation solutions of two foot-mounted zero-velocity-aided inertial navigation systems. Then in Section IV, we describe an experiment that we conducted to evaluate the proposed method and present results. Finally, in Section V, we draw conclusions.

Reproducible Research: The data and Matlab code used in this paper are available at www.openshoe.org.

II. Applying an inequality constraint to the navigation solution using a projection

In this section, we will show how knowledge of the maximum distance between two navigation systems can be used

978-1-4673-1954-6/12/$31.00 © 2012 IEEE

to fuse the navigation information from the two systems.

A. Problem formulation

Consider a scenario where we have two navigation system, and let $\mathbf{x}_k^{(i)} \in \mathbb{R}^{n_i}$, $i = 1, 2.$, be the vector containing the true state of the i:th navigation system at time instant $k \in \mathbb{N}^+$. Further, assume that the $s \in \mathbb{N}^+$ first elements of the vector $\mathbf{x}_k^{(i)}$ represents the position of the navigation system. Next, define the joint state vector

$$\mathbf{x}_k \stackrel{\text{def}}{=} \begin{bmatrix} \left(\mathbf{x}_k^{(1)}\right)^T & \left(\mathbf{x}_k^{(2)}\right)^T \end{bmatrix}^T \tag{1}$$

and the matrix

$$\mathbf{L} \stackrel{\text{def}}{=} \begin{bmatrix} \mathbf{I}_s & \mathbf{0}_{3,n_1} & -\mathbf{I}_s & \mathbf{0}_{3,n_2} \end{bmatrix}. \tag{2}$$

Here $\mathbf{I}_q$ and $\mathbf{0}_{q,r}$ denote a identity matrix of size q and a zero matrix of size q times r, respectively. Moreover, $(\cdot)^T$ denotes the transpose operation. Now, if there is an upper bound γ on how far separated the two navigation systems can be, it most hold that $\|\mathbf{L}\,\mathbf{x}_k\|^2 \leq \gamma^2 \;\forall k$. Hence, we would like the joint navigation solution $\widehat{\mathbf{x}}_k \in \mathbb{R}^m$ $(m = n_1 + n_2)$, defined as

$$\widehat{\mathbf{x}}_k \stackrel{\text{def}}{=} \begin{bmatrix} \left(\widehat{\mathbf{x}}_k^{(1)}\right)^T & \left(\widehat{\mathbf{x}}_k^{(2)}\right)^T \end{bmatrix}^T, \tag{3}$$

to also fulfill this condition. Here and through out the paper the circumflex diacritic $\widehat{\cdot}$ is used to indicate an estimate quantity.

One way of imposing the constraint to the joint navigation solution is to, if $\|\mathbf{L}\,\widehat{\mathbf{x}}_k\|^2 > \gamma^2$, project the joint navigation solutions onto the subspace $\{\mathbf{x} \in \mathbb{R}^m : \|\mathbf{L}\,\mathbf{x}\|^2 \leq \gamma^2\}$. (See e.g., [2] for an alternative approach to impose the constraint.) One projection $p(\widehat{\mathbf{x}}_k)$ that does this is given by [3], [4]

$$p(\widehat{\mathbf{x}}_k) \stackrel{\text{def}}{=} \underset{\mathbf{x}}{\operatorname{argmin}}\Big(\|\widehat{\mathbf{x}}_k - \mathbf{x}\|^2_{\mathbf{P}_k^{-1}}\Big) \quad \text{s.t.} \quad \|\mathbf{L}\,\mathbf{x}\|^2 \leq \gamma^2, \tag{4}$$

where

$$\|\widehat{\mathbf{x}}_k - \mathbf{x}\|^2_{\mathbf{P}_k^{-1}} = (\widehat{\mathbf{x}}_k - \mathbf{x})^T\,\mathbf{P}_k^{-1}(\widehat{\mathbf{x}}_k - \mathbf{x}), \tag{5}$$

and $\mathbf{P}_k$ is the covariance matrix of the joint navigation solution $\widehat{\mathbf{x}}_k$. That is, the projection is the solution to an inequality constrained weighted least squares problem.

B. The solution to the constrained least square problem

The solution to the inequality constrained weighted least squares problem is a stationary point of the Lagrange function (with the Lagrange multiplier λ) [5]

$$J(\mathbf{x}, \lambda) \stackrel{\text{def}}{=} \|\widehat{\mathbf{x}}_k - \mathbf{x}\|^2_{\mathbf{P}_k^{-1}} + \lambda\,\psi(\mathbf{x}), \tag{6}$$

where

$$\psi(\mathbf{x}) \stackrel{\text{def}}{=} \|\mathbf{L}\,\mathbf{x}\|^2 - \gamma^2. \tag{7}$$

The stationary points of the Lagrange function are given by the solutions to the normal equations

$$\frac{\partial J(\mathbf{x},\lambda)}{\partial \mathbf{x}} = \mathbf{0} \quad \leftrightarrow \quad \left(\mathbf{P}_k^{-1} + \lambda \mathbf{L}^T\,\mathbf{L}\right)\mathbf{x} = \mathbf{P}_k^{-1}\,\widehat{\mathbf{x}}_k \tag{8}$$

$$\frac{\partial J(\mathbf{x},\lambda)}{\partial \lambda} = 0 \quad \leftrightarrow \quad \psi(\mathbf{x}) = 0. \tag{9}$$

If there is a unique solution to the constraint least squares problem in (4), then the stationary point of the Lagrange function that corresponds to the solution to the constrained least squares problem, is the unique stationary point for which $\lambda > 0$ [5]. Let this stationary point be denote by $\{\mathbf{x}^*, \lambda^*\}$. Further, note that if the solution is unique, then $\left(\mathbf{P}^{-1} + \lambda^*\mathbf{L}^T\mathbf{L}\right)$ must have full rank and the projection function $p(\widehat{\mathbf{x}}_k)$ defined in (4) can equivalently be written as

$$p(\widehat{\mathbf{x}}) \stackrel{\text{eqv}}{=} \Pi(\lambda^*)\,\widehat{\mathbf{x}}_k \tag{10}$$

where

$$\Pi(\lambda) \stackrel{\text{def}}{=} \left(\mathbf{P}_k^{-1} + \lambda\,\mathbf{L}^T\mathbf{L}\right)^{-1}\mathbf{P}_k^{-1}, \tag{11}$$

and

$$\lambda^* \stackrel{\text{def}}{=} \{\lambda \in \mathbb{R}^+ : \psi(\Pi(\lambda)\,\widehat{\mathbf{x}}_k) = 0\}. \tag{12}$$

The function $\psi(\Pi(\lambda)\,\widehat{\mathbf{x}}_k)$ is a nonlinear polynomial function in λ, and to find its roots, one must in most cases resort to some numerical root finding method such as the Bisection method or Newton's method. We have used the method described in [3] and not experienced any convergence problems.

C. The covariance of the projected navigation state vector

The covariance $\mathbf{P}_k^*$ of the projected joint navigation solution can be approximated as [4]

$$\mathbf{P}_k^* = \nabla p\,\mathbf{P}_k(\nabla p)^T, \tag{13}$$

where ∇p is the Jacobian matrix of the projection function $p(\mathbf{x})$ with respect to $\mathbf{x}$ evaluated around $\widehat{\mathbf{x}}_k$, i.e.,

$$\nabla p \stackrel{\text{def}}{=} \left[\frac{\partial p(\mathbf{x})}{\partial [\mathbf{x}]_1}\Big|_{\mathbf{x}=\widehat{\mathbf{x}}_k} \cdots \frac{\partial p(\mathbf{x})}{\partial [\mathbf{x}]_m}\Big|_{\mathbf{x}=\widehat{\mathbf{x}}_k}\right].$$

Here $[\mathbf{x}]_i$ denotes the i:th element of the vector $\mathbf{x}$. To find the Jacobian matrix, we may first note that

$$\begin{aligned}\frac{\partial p(\mathbf{x})}{\partial [\mathbf{x}]_i} &= \frac{\partial \Pi(\lambda^*)}{\partial [\mathbf{x}]_i}\mathbf{x} + \Pi(\lambda^*)\,\frac{\partial \mathbf{x}}{\partial [\mathbf{x}]_i} \\ &= \frac{\partial \Pi(\lambda^*)}{\partial \lambda^*}\frac{\partial \lambda^*}{\partial [\mathbf{x}]_i}\mathbf{x} + \Pi(\lambda^*)\,\mathbf{e}_i,\end{aligned} \tag{14}$$

where $\mathbf{e}_i$ denotes the i:th natural basis vector. The partial derivative of $\Pi(\lambda^*)$ with respect to λ^* is given by

$$\begin{aligned}\frac{\partial \Pi(\lambda^*)}{\partial \lambda^*} &= \left(\frac{\partial\left(\mathbf{P}_k^{-1} + \lambda^*\mathbf{L}^T\mathbf{L}\right)^{-1}}{\partial \lambda^*}\right)\mathbf{P}_k^{-1} \\ &= -\left(\mathbf{P}_k^{-1} + \lambda^*\mathbf{L}^T\mathbf{L}\right)^{-1}\left(\frac{\partial\left(\mathbf{P}_k^{-1} + \lambda^*\mathbf{L}^T\mathbf{L}\right)}{\partial \lambda^*}\right) \\ &\quad \cdot\left(\mathbf{P}_k^{-1} + \lambda^*\mathbf{L}^T\mathbf{L}\right)^{-1}\mathbf{P}_k^{-1} \\ &= -(\mathbf{P}_k^{-1} + \lambda^*\mathbf{L}^T\mathbf{L})^{-1}\mathbf{L}^T\mathbf{L}\Pi(\lambda^*).\end{aligned}$$

Thus, we can express the Jacobian matrix ∇p as

$$\begin{aligned}\nabla p &= \Pi(\lambda^*) - (\mathbf{P}_k^{-1} + \lambda^*\mathbf{L}^T\mathbf{L})^{-1}\mathbf{L}^T\mathbf{L}\Pi(\lambda^*)\widehat{\mathbf{x}}_k\nabla\lambda^* \\ &= \Pi(\lambda^*) - (\mathbf{P}_k^{-1} + \lambda^*\mathbf{L}^T\mathbf{L})^{-1}\mathbf{L}^T\mathbf{L}p(\widehat{\mathbf{x}}_k)\nabla\lambda^*,\end{aligned} \tag{15}$$

where $\nabla\lambda^*$ denotes the Jacobian matrix of λ^* with respect to $\mathbf{x}$ evaluated around $\widehat{\mathbf{x}}_k$. From the implicit function theorem it follows that the Jacobian matrix $\nabla\lambda^*$ is given by

$$\nabla\lambda^* = -\left(\frac{\partial\psi(\Pi(\lambda)\,\widehat{\mathbf{x}}_k)}{\partial\lambda}\Big|_{\lambda=\lambda^*}\right)^{-1}\nabla\psi, \tag{16}$$

where $\nabla\psi$ is the Jacobian matrix of $\psi(\Pi(\lambda^*)\,\mathbf{x})$ with respect to $\mathbf{x}$, evaluated around $\widehat{\mathbf{x}}_k$. The partial derivative and the Jacobian matrix in (16) are

$$\begin{aligned}\frac{\partial\psi(\Pi(\lambda)\,\widehat{\mathbf{x}}_k)}{\partial\lambda}\Big|_{\lambda=\lambda^*} &= \frac{\partial}{\partial\lambda}\Big(\widehat{\mathbf{x}}_k^T\Pi(\lambda)^T\mathbf{L}^T\mathbf{L}\Pi(\lambda)\widehat{\mathbf{x}}_k\Big)\Big|_{\lambda=\lambda^*}\\ &= \widehat{\mathbf{x}}_k^T\Big(\frac{\partial\Pi(\lambda)^T}{\partial\lambda}\Big|_{\lambda=\lambda^*}\mathbf{L}^T\mathbf{L}\Pi(\lambda^*)\\ &\quad+\Pi(\lambda^*)^T\mathbf{L}^T\mathbf{L}\frac{\partial\Pi(\lambda)}{\partial\lambda}\Big|_{\lambda=\lambda^*}\Big)\widehat{\mathbf{x}}_k\\ &= 2\,\widehat{\mathbf{x}}_k^T\Pi(\lambda^*)^T\mathbf{L}^T\mathbf{L}\frac{\partial\Pi(\lambda)}{\partial\lambda}\Big|_{\lambda=\lambda^*}\widehat{\mathbf{x}}_k\\ &= -2\,\widehat{\mathbf{x}}_k^T\Pi(\lambda^*)^T\mathbf{L}^T\mathbf{L}\\ &\quad\cdot\big(\mathbf{P}_k^{-1}+\lambda^*\,\mathbf{L}^T\mathbf{L}\big)^{-1}\mathbf{L}^T\mathbf{L}\Pi(\lambda^*)\widehat{\mathbf{x}}_k\\ &= -2\,p(\widehat{\mathbf{x}}_k)^T\mathbf{L}^T\mathbf{L}\,\big(\mathbf{P}_k^{-1}+\lambda^*\,\mathbf{L}^T\mathbf{L}\big)^{-1}\\ &\quad\cdot\mathbf{L}^T\mathbf{L}\,p(\widehat{\mathbf{x}}_k)\end{aligned} \tag{17}$$

and

$$\begin{aligned}\nabla\psi &= 2\,\widehat{\mathbf{x}}_k^T\Pi(\lambda^*)^T\mathbf{L}^T\mathbf{L}\Pi(\lambda^*)\\ &= 2\,p(\widehat{\mathbf{x}}_k)^T\mathbf{L}^T\mathbf{L}\Pi(\lambda^*),\end{aligned} \tag{18}$$

respectively. Thus, the Jacobian matrix $\nabla\lambda^*$ is given by

$$\nabla\lambda^* = \frac{p(\widehat{\mathbf{x}}_k)^T\mathbf{L}^T\mathbf{L}\,\Pi(\lambda^*)}{p(\widehat{\mathbf{x}}_k)^T\mathbf{L}^T\mathbf{L}(\mathbf{P}_k^{-1}+\lambda^*\,\mathbf{L}^T\mathbf{L})^{-1}\mathbf{L}^T\mathbf{L}\,p(\widehat{\mathbf{x}}_k)} \tag{19}$$

Introducing $\mathbf{z}_k \overset{\text{def}}{=} \mathbf{L}^T\mathbf{L}\,p(\widehat{\mathbf{x}}_k)$ and inserting (19) into (15) yields the Jacobian matrix ∇p of the projection $p(\mathbf{x})$

$$\begin{aligned}\nabla p &= \Pi(\lambda^*) - \frac{(\mathbf{P}_k^{-1}+\lambda^*\mathbf{L}^T\mathbf{L})^{-1}\mathbf{z}_k\mathbf{z}_k^T\,\Pi(\lambda^*)}{\mathbf{z}_k^T(\mathbf{P}_k^{-1}+\lambda^*\,\mathbf{L}^T\mathbf{L})^{-1}\mathbf{z}_k}\\ &= \left(\mathbf{I}_m - \frac{(\mathbf{P}_k^{-1}+\lambda^*\mathbf{L}^T\mathbf{L})^{-1}\mathbf{z}_k\mathbf{z}_k^T}{\mathbf{z}_k^T(\mathbf{P}_k^{-1}+\lambda^*\,\mathbf{L}^T\mathbf{L})^{-1}\mathbf{z}_k}\right)\Pi(\lambda^*)\end{aligned} \tag{20}$$

from which the covariance approximation (13) is given.

D. Handling navigation systems with attitude estimates

In the problem formulation in Section II-A it was assumed that all navigation states of the i:th navigation system could be represented by a vector $\mathbf{x}_k^i$, defined in $\mathbb{R}^{n_i}$. This may cause a problem if the state vector of the i:th navigation system also include the attitude of the navigation platform. This since an angle is only defined on $[0, 2\pi)$ and a sequence of rotations does not commute and can therefore not be represented by a "proper" vector. Hence, the proposed projection method will, not without modifications, work as intended with navigation state vectors that includes attitude states.

The way to get around this problem is to: (a) note that the constraint in the weighted least squares minimization, only depends upon the position states and that the minimization only affects all the other states via the weighting matrix $\mathbf{P}_k^{-1}$; and (b) to assume that the coupling between the states, represented by the weighting matrix, is such that a change in position only has a small effect on the attitude. This means that before the minimization we may substitute the elements of the state vector that represent the attitude of the platform with zero and then do the minimization (projection). The "attitude" elements in the output vector of the minimization will then be the small attitude perturbations that can be used to calculate the correct (projected) attitude states using, for example, the method described in [6, p. 200]

III. Foot-to-foot distance inequality constraint zero-velocity aided inertial navigation

In this section, we will use the projection method described in the previous section to formulate a Kalman filter algorithm for fusing the information from two foot-mounted zero-velocity-aided inertial navigation systems using an upper bound on γ the maximum foot-to-foot distance. We will begin by introducing the navigation equations and the state-space model used in the filter.

A. The navigation equations and the state-space model

Let $\widehat{\mathbf{x}}_k \in \mathbb{S}^{18}$ be the joint navigation solution of the two foot-mounted inertial navigation systems, and let the function f denote the mechanized navigation equations that the inertial navigation systems uses to compute their navigation solutions. That, is

$$\widehat{\mathbf{x}}_k^{(i)} = f(\widehat{\mathbf{x}}_{k-1}^{(i)}, \widetilde{\mathbf{s}}_k^{(i)}, \widetilde{\omega}_k^{(i)}) \quad \widehat{\mathbf{x}}_0^{(i)} = \{\text{Initial Condition}\}. \tag{21}$$

Here $\widetilde{\mathbf{s}}_k^{(i)} \in \mathbb{R}^3$ and $\widetilde{\omega}_k^{(i)} \in \mathbb{R}^3$ denote the, by the i:th system, measured specific force and angular rate vector, respectively. Further, let the time dynamics of the perturbations (errors) $\delta\mathbf{x}_k^{(i)} \in \mathbb{R}^9$ in the navigation solution of the i:th system be described by the state-space model

$$\delta\mathbf{x}_k^{(i)} = \mathbf{F}_k^{(i)}\,\delta\mathbf{x}_{k-1}^{(i)} + \mathbf{G}_k^{(i)}\,\mathbf{w}_k^{(i)}. \tag{22}$$

Here $\mathbf{F}_k^{(i)}$ and $\mathbf{G}_k^{(i)}$ denote the state transition and noise gain matrix, respectively. The vector $\mathbf{w}_k^{(i)} \in \mathbb{R}^6$ denotes the process noise, which is assumed white and to have the covariance matrix $\mathbf{Q}^{(i)}$. Since the attitude perturbations, the attitude estimates, and the true attitude in general has a nonlinear dependence, the relationship between the perturbation vector $\delta\mathbf{x}_k^{(i)}$, the navigation solution $\widehat{\mathbf{x}}_k^{(i)}$, and the true navigation state $\mathbf{x}_k^{(i)}$ is also generally given by a nonlinear function. For the system parameterizations we have a hand, let this function be denoted by Γ, and defined so that $\mathbf{x}_k^{(i)} = \Gamma(\widehat{\mathbf{x}}_k^{(i)}, \delta\mathbf{x}_k^{(i)})$.

Now, if we assume that the perturbations in the two foot-mounted inertial navigation systems are independent of each other, then the time dynamics of the perturbations $\delta\mathbf{x}_k \in \mathbb{R}^{18}$ in the joint navigation solution can be modeled as

$$\delta\mathbf{x}_k = \mathbf{F}_k\,\delta\mathbf{x}_{k-1} + \mathbf{G}_k\,\mathbf{w}_k, \tag{23}$$

where

$$\mathbf{F}_k = \begin{bmatrix} \mathbf{F}_k^{(1)} & \mathbf{0}_{9,9} \\ \mathbf{0}_{9,9} & \mathbf{F}_k^{(2)} \end{bmatrix}, \quad \mathbf{G}_k = \begin{bmatrix} \mathbf{G}_k^{(1)} & \mathbf{0}_{9,6} \\ \mathbf{0}_{9,6} & \mathbf{G}_k^{(2)} \end{bmatrix}, \tag{24}$$

and $\mathbf{w}_k = \begin{bmatrix} (\mathbf{w}_k^{(1)})^T & (\mathbf{w}_k^{(2)})^T \end{bmatrix}^T$. Since, the perturbations in the two systems are assumed independent, the covariance matrix of the process noise $\mathbf{w}_k$ has the structure

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}^{(1)} & \mathbf{0}_{6,6} \\ \mathbf{0}_{6,6} & \mathbf{Q}^{(2)} \end{bmatrix}. \tag{25}$$

B. The zero-velocity updates and the observation equation

Next, let $T_k^{(i)}$ an $\eta^{(i)}$ denote the, by the i:th system's zero-velocity detector, calculated test statistic and detection threshold, respectively. See [7] for different ways to calculate the zero-velocity detection test statistics. If $T_k^{(i)} \leq \eta^{(i)}$, the detector chooses the hypothesis that the system has zero-velocity. However, since there is nothing as a perfect detector, the system may be subjected to some small motions even though the detector declares the system to have zero velocity. Let the velocity of these small motions be denoted by $\mathbf{v}_k^{(i)}$, and assume that the velocity can be modeled as additive white noise with covariance matrix $\mathbf{R}^{(i)}$. Then, we can define the following zero-velocity observation (measurement) equation for the state-space model of the perturbations in the joint navigation solution.

If $T_k^{(1)} \leq \eta^{(1)}$ or $T_k^{(2)} \leq \eta^{(2)}$, then

$$-\mathbf{H}_k\,\widehat{\mathbf{x}}_k = \mathbf{H}_k\,\delta\mathbf{x}_k + \mathbf{v}_k, \tag{26}$$

where

$$\mathbf{H}_k = \begin{cases} \begin{bmatrix} \mathbf{H} & \mathbf{0}_{3,9} \end{bmatrix}, & T_k^{(1)} \leq \eta^{(1)} \;\&\; T_k^{(2)} > \eta^{(2)} \\ \begin{bmatrix} \mathbf{0}_{3,9} & \mathbf{H} \end{bmatrix}, & T_k^{(1)} > \eta^{(1)} \;\&\; T_k^{(2)} \leq \eta^{(2)} \\ \begin{bmatrix} \mathbf{H} & \mathbf{0}_{3,9} \\ \mathbf{0}_{3,9} & \mathbf{H} \end{bmatrix}, & T_k^{(1)} \leq \eta^{(1)} \;\&\; T_k^{(2)} \leq \eta^{(2)} \end{cases}, \tag{27}$$

$$\mathbf{v}_k = \begin{cases} \mathbf{v}_k^{(1)}, & T_k^{(1)} \leq \eta^{(1)} \;\&\; T_k^{(2)} > \eta^{(2)} \\ \mathbf{v}_k^{(2)}, & T_k^{(1)} > \eta^{(1)} \;\&\; T_k^{(2)} \leq \eta^{(2)} \\ \begin{bmatrix} \mathbf{v}_k^{(1)} \\ \mathbf{v}_k^{(2)} \end{bmatrix}, & T_k^{(1)} \leq \eta^{(1)} \;\&\; T_k^{(2)} \leq \eta^{(2)} \end{cases}, \tag{28}$$

and

$$\mathbf{H} = \begin{bmatrix} \mathbf{0}_{3,3} & \mathbf{I}_3 & \mathbf{0}_{3,3} \end{bmatrix}. \tag{29}$$

Further, if we assume that the small motions during the zero-velocity updates of the two systems are independent of each other, then the covariance of $\mathbf{v}_k$ is given by

$$\mathbf{R}_k = \begin{cases} \mathbf{R}^{(1)}, & T_k^{(1)} \leq \eta^{(1)} \;\&\; T_k^{(2)} > \eta^{(2)} \\ \mathbf{R}^{(2)}, & T_k^{(1)} > \eta^{(1)} \;\&\; T_k^{(2)} \leq \eta^{(2)} \\ \begin{bmatrix} \mathbf{R}^{(1)} & \mathbf{0}_{3,3} \\ \mathbf{0}_{3,3} & \mathbf{R}^{(2)} \end{bmatrix}, & T_k^{(1)} \leq \eta^{(1)} \;\&\; T_k^{(2)} \leq \eta^{(2)} \end{cases}. \tag{30}$$

Algorithm 1 Pseudo code for the proposed Kalman filter algorithm.

```
1:  k ← 0, c_Z ← −τ_Z
2:  x̂_k ← Process{ Joint initial navigation state }
3:  P_k ← Process{ Initial covariance matrix }
4:  loop
5:    k ← k + 1
6:    [x̂_k]_{1:9} ← f([x̂_{k−1}]_{1:9}, s̃_k^{(1)}, ω̃_k^{(1)})
7:    [x̂_k]_{10:18} ← f([x̂_{k−1}]_{10:18}, s̃_k^{(2)}, ω̃_k^{(2)})
8:    P_k ← F_k P_{k−1} F_k^T + G_k Q G_k^T
9:    T_k^{(1)} ← Process{ Zero-velocity detector for system #1 }
10:   T_k^{(2)} ← Process{ Zero-velocity detector for system #2 }
11:   if T_k^{(1)} ≤ η^{(1)} or T_k^{(2)} ≤ η^{(2)} then
12:     K_k ← P_k H_k^T (H_k P_k H_k^T + R_k)^{−1}
13:     δx̂_k ← −K_k H_k x̂_k
14:     [x̂_k]_{1:9} ← Γ([x̂_k]_{1:9}, [δx̂_k]_{1:9})
15:     [x̂_k]_{10:18} ← Γ([x̂_k]_{10:18}, [δx̂_k]_{10:18})
16:     P_k ← (I_18 − K_k H_k) P_k
17:     if ‖L x̂_k‖^2 > γ^2 and k − c_Z > τ_Z then
18:       x̂_k ← p(x̂_k)
19:       P_k ← ∇p P_k (∇p)^T
20:       c_Z ← k
21:     end if
22:   end if
23: end loop
```

C. The state constrained Kalman filter algorithm

With the navigation equations and the state-space model for the system perturbation with zero-velocity updates defined as in Section III-A and III-B, the pseudo code for an inequality constrained Kalman filter algorithm that fuses the information from two foot-mounted zero-velocity aided navigation system is given in Algorithm 1.

The algorithm works as follows. First, the joint navigation state vector and the covariance of the joint navigation state vector are initialized at line 2 and 3, respectively. Then, at line 6 and 7, the navigation state vector is updated using the current inertial measurement unit data. After that, at line 8, the covariance of the updated joint navigation state vector is calculated. Thereafter, at line 9 and 10 the zero-velocity detectors calculates their test statistics. The test statistics are compared with the detection thresholds at line 11. If any or both of the two test statistics are below their detection thresholds, one or both systems are assumed to be stationary and a zero-velocity update is done by executing line 12 to 16 of the pseudo code. That is, at line 12 the Kalman gain is calculated, which then at line 13 is used to estimate the perturbations in the joint navigation solution. Thereafter, at line 14 and 15 the estimated perturbations are used to correct the navigation solution. As a final part of the zero-velocity update the posteriori covariance of the joint navigation solution is calculated at line 16.

The solution to the constrained least square problem in-

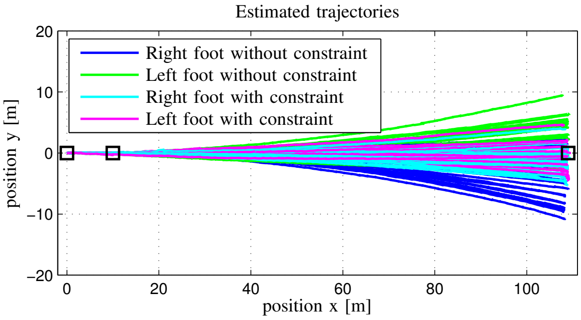

Fig. 2: Estimated trajectories from walking along a 110[m] straight line. The black boxes indicate the location of the starting position (0[m]), the heading reference point (10[m]), and the stop position (110[m]).

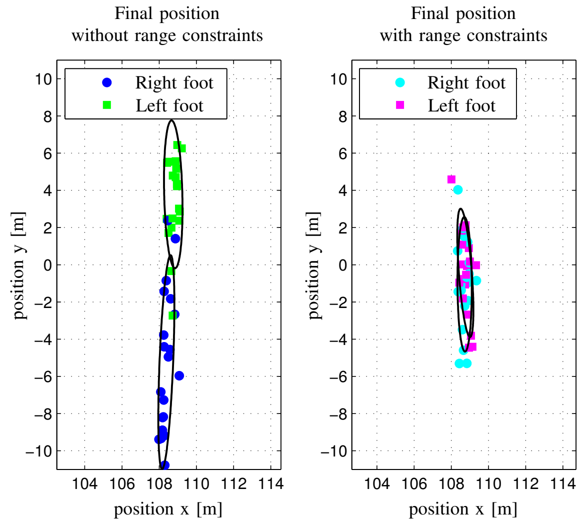

Fig. 3: Scatter plot of end position of the two systems with and without the range constraint. 1σ confidence ellipsoids are shown in black. It is clearly seen that the mean error and covariance of the final position estimates are significantly reduced by applying the range constraint.

corporates information that is not well captured by only the mean and the covariance. Therefore, to avoid a biased estimate and numerical problems the constraint is only applied if a minimum number of samples τ_Z has elapsed since the previous time that it was applied. Consequently, at line 17, the algorithm checks to determine whether the navigation solution does not fulfill the maximum foot-to-foot distance constraint and a minimum time has elapsed since the previous time that the constraint was applied. If this is the case, at lines 18-19, the navigation solution is projected onto the subspace that fulfills the constraint and the covariance of the projected estimate is calculated. We have typically used a τ_Z corresponding to approximately 1[s].

IV. Experiment and Results

To test the proposed algorithm, the following experiment was conducted. A user, equipped with one OpenShoe navigation system on each foot, walked 110 meters on level ground along a straight line at a normal gait speed (approx. 5 km/h). Refer to www.openshoe.org or [8] for details about the OpenShoe navigation system. Twenty such trajectories with 4 different OpenShoe units (different IMUs) were recorded. To get the same initial and final positions and a heading reference, plates with imprints of the shoes were positioned at 0[m], 10[m], and 110[m]. The initial heading was set such that the estimated position at the 10[m] plate, without using the constraint was, on the x-axis (Adjustments were made for the spacing between the foots.).

The inertial measurement unit data collected from the two navigation systems was then processed with Algorithm 1. The processing was done with the maximum foot-to-foot distance set to infinity ($\gamma = \infty$[m]), giving two uncoupled systems, and set to 1 meter ($\gamma = 1$[m]), giving the constrained system. The estimated trajectories are shown in Fig. 2. Corresponding scatter plots with 1σ confidence ellipsoids of the final horizontal position estimates are shown in Fig. 3. Applying the range constraints can be seen to have significantly reduced the mean error and covariance of the final position estimates. As noted in [9], there are symmetric systematic modeling errors which will cancel out. Therefore, two foot-mounted zero-velocity aided inertial navigation systems are believed to be a favorable application and the experiment shows that constraints can be used to fuse the information from two navigation systems.

V. Conclusions

We have suggested a method to fuse the information from two navigation system using an upper bound on their spatial separation. The solution is based on a constrained least square problem formulation. The use of the method has been demonstrated on the fusion of information from two foot-mounted zero-velocity update aided inertial navigation systems. In this setup, the use of range constraint has been shown to significantly improve the navigation performance.

References

[1] J. Rantakokko, *et al.*, "Accurate and reliable soldier and first responder indoor positioning: multisensor systems and cooperative localization," *IEEE Wireless Commun. Mag.*, vol. 18, pp. 10 –18, Apr. 2011.

[2] D. Zachariah, *et al.*, "Bayesian estimation with distance bounds," *IEEE Trans. Signal Process.*, To appear. [Online]. Available: 10.1109/LSP.2012.2224865

[3] Y. Chun and E. Blasch, "Kalman filtering with nonlinear state constraints," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 45, no. 1, pp. 70–84, Jan. 2009.

[4] S. Julier and J. LaViola, "On Kalman filtering with nonlinear equality constraints," *IEEE Trans. Signal Process.*, vol. 55, no. 6, pp. 2774–2784, June 2007.

[5] W. Gander, "Least squares with a quadratic constraint," *Numerische Mathematik*, vol. 36, no. 3, pp. 291–307, 1980.

[6] J. A. Farrell and M. Barth, *The Global Positioning System and Inertial Navigation*. McGraw-Hill, 1998.

[7] I. Skog, *et al.*, "Zero-velocity detection – An algorithm evaluation," *IEEE Trans. Bio-Med. Eng.*, vol. 57, no. 11, pp. 2657–2666, Nov. 2010.

[8] J.-O. Nilsson, *et al.*, "Foot-mounted INS for everybody – An open-source embedded implementation," in *Proc. of The Position Location and Navigation System (PLANS) Conference*, SC, USA, Apr. 2012.

[9] J.-O. Nilsson, *et al.*, "A note on the limitations of ZUPTs and the implications on sensor error modeling," in *Proc. of The Indoor Positioning and Indoor Navigation (IPIN) Conference*, Syd, Australia, Nov. 2012.