Robot's Velocity and Tilt Estimation Through Computationally Efficient Fusion of Proprioceptive Sensors Readouts

Paweł Wawrzyński, *Member, IEEE*
Institute of Control and Computation Engineering
Warsaw University of Technology
00-665 Warsaw, Poland,
p.wawrzynski@elka.pw.edu.pl

***Abstract*— In this paper a method is introduced that combines Inertial Measurement Unit (IMU) readouts with low accuracy and temporarily unavailable velocity measurements (e.g., based on kinematics or GPS) to produce high accuracy estimates of velocity and orientation with respect to gravity. The method is computationally cheap enough to be readily implementable in sensors. The main area of application of the introduced method is mobile robotics.**

***Keywords*—velocity estimation, Kalman filter, mobile robotics.**

I. Introduction

Knowledge of velocity of a robot or its specific part is usually crucial for their efficient control. Orientation of the robot with respect to gravity vector is important especially in legged robots, where it enables balancing of their body.

Velocity estimation is possible with a system based on vision [1] or Global Positioning System (GPS) [2], [3], [4]. However, in many applications where autonomy of the robot is required, exteroceptive sensors are unwanted. It is known that velocity and orientation may be estimated by proper integration of Inertial Measurement Unit (IMU) readouts. However, IMU can not be used alone, as it is prone to the so-called 'drift' effect which makes the estimates practically useless [5].

In [6] we proposed a method that combines IMU readouts with velocity measurements that come from robot kinematics to produce velocity and orientation estimates of a legged robot. The estimates are generally of high accuracy even if measurements are of low accuracy and are temporarily unavailable. In this paper we present a version of this method that is so computationally inexpensive that it may be in principle implemented on electronics within a sensor (e.g., IMU).

The structure of this paper is as follows: Sec. II presents the formal problem definition, and the notation used throughout the paper. In Sec. III basic tools for the velocity estimation are presented along with Extended Kalman Filter sensor fusion. Sec. IV introduces the contribution of this paper, which is a computationally inexpensive filter for velocity estimation. Experimental data analysis and discussions are given in Sec. V. Finally, in Sec. VI a brief summary of the results is given.

II. Problem formulation

A point in the robot body is given with Inertial Measurement Unit (IMU) attached to it. IMU measures acceleration and angular velocity. An auxiliary sensor measures velocity of IMU in a drift-less manner. This auxiliary measurement may result from leg's kinematics, angular velocity of wheels, or GPS. It may be temporarily unavailable because of no leg touching the ground, inaccessibility of GPS satellites, and so on.

At each moment we wish accurate measurement of velocity and tilt of IMU in the frame in which this sensor takes measurements. This frame, hereafter called *IMU frame* is immobile with respect to the ground and at each moment it is parallel to IMU. Tilt is expressed as gravity vector in IMU frame. It is understood that having velocity and tilt in IMU frame, we are able to express velocity and orientation of IMU in any coordinate system. We are not able to express global yaw, but without external reference point we are not able to estimate it with high accuracy anyway.

We require that velocity and tilt estimates are recursive, and their update on the basis of coming data are based on limited computational effort. In essence, we require that the whole computational burden may be handled by a microprocessor within IMU.

Notation

Whenever it matters, we assume right-hand coordinate system and right-wise rotations for positive angle. We also apply the following notation conventions.

- $\mathbf{a} \in R^3$ is an acceleration vector measured by IMU, i.e., a sum of linear acceleration and acceleration due to gravity, in IMU frame.
- $\boldsymbol{\omega} \in R^3$ is the angular velocity of IMU in IMU frame.
- $\mathbf{g} \in R^3$ is the gravity vector in IMU frame.
- $\mathbf{v} \in R^3$ is the velocity vector of IMU in IMU frame.
- δ is the constant time elapsing between consecutive IMU measurements.

$\mathbf{r}(\mathbf{x},\boldsymbol{\beta}) \in R^3$ is a vector that is a result of simultaneous rotations of vector $\mathbf{x}$ about each axis of the frame by the angles contained in vector $\boldsymbol{\beta} \in R^{3\times 1}$. Gyroscope measurements will be handled with this kind of rotation in Sec. III.

$\widehat{x}$ is an estimate of the true value x.

x^S is a measurement of the true value x whereas S denotes the source of this measurement, e.g., $\mathbf{a}^A$ is the accelerometer readout, and $\boldsymbol{\omega}^G$ is the gyroscope readout.

$\widetilde{x} = \widehat{x} - x$ $(= x^S - x)$ is an error of the estimate $\widehat{x}$ (or measurement x^S).

"$\leftarrow$": This paper focuses on recursive estimates i.e., the estimates that are computed on the basis of their previous values, .e.g,

$$\widehat{\mathbf{x}}_t = f(\widehat{\mathbf{x}}_{t-1}) \tag{1}$$

where t is time. Definitions such as (1) will be written in the simpler form

$$\widehat{\mathbf{x}} \leftarrow f(\widehat{\mathbf{x}}). \tag{2}$$

III. Accurate solution

This section presents the method of estimation of velocity and tilt introduced in [6]. This method will serve as a basis to introduce a simplified in the following section.

A. Dynamics of tilt and velocity in IMU frame

Let us consider a mobile IMU, the gravity vector, $\mathbf{g}$, and velocity of the sensor, $\mathbf{v}$, both in IMU frame. Suppose in a time period of infinitesimal length $\delta > 0$ angular velocity of the sensor is constant, and equal to $\boldsymbol{\omega}$. Within the period, the gravity remains constant, but in IMU frame it is rotating with angular velocity of $-\boldsymbol{\omega}$. Hence, within the period $\mathbf{g}$ changes to

$$\mathbf{g} \leftarrow \mathbf{r}(\mathbf{g}, -\boldsymbol{\omega}\delta). \tag{3}$$

Suppose within the period the sensor is moving with constant linear acceleration and finally it perceives acceleration, $\mathbf{a}$, which is a sum of the sensor's linear acceleration, and the gravity, $\mathbf{g}$, both in IMU frame. The velocity of the sensor in IMU frame changes due to (i) the rotation of the frame, (ii) the linear acceleration. Hence, its new value is

$$\mathbf{v} \leftarrow \mathbf{r}(\mathbf{v}, -\boldsymbol{\omega}\delta) + \big(\mathbf{a} - \mathbf{r}(\mathbf{g}, -\boldsymbol{\omega}\delta)\big)\delta, \tag{4}$$

with $\mathbf{g}$ from the beginning of the period.

B. State of inertial sensor

State of the sensor

$$\text{state} = \begin{bmatrix} \mathbf{v} \\ \mathbf{g} \\ \bar{\mathbf{a}} \\ \bar{\boldsymbol{\omega}} \end{bmatrix}, \tag{5}$$

encompasses the following elements, each of them being a vector in R^3:

- $\mathbf{v}$ — linear velocity of IMU in relation to the ground, expressed in IMU frame,
- $\mathbf{g}$ — vector of acceleration due to gravity, expressed in IMU frame,
- $\bar{\mathbf{a}}$ — bias in accelerometer measurements,
- $\bar{\boldsymbol{\omega}}$ — bias in gyroscope measurements.

IMUs are usually burdened with significant biases which change over time. However, they can be handled by introducing them to the filter state [7].

C. Model of dynamics

Defining evolution of the state we consider IMU measurements, $\boldsymbol{\omega}^G$ and $\mathbf{a}^A$, as inputs to the dynamical system. The measurement $\boldsymbol{\omega}^G$ is assumed to be a sum of true angular velocity, $\boldsymbol{\omega}$, the gyroscope bias, $\bar{\boldsymbol{\omega}}$, and zero-mean noise, $\widetilde{\boldsymbol{\omega}}$. Similarly, the acceleration measurement $\mathbf{a}^A$ is assumed to be a sum of true acceleration, $\mathbf{a}$, accelerometer bias, $\bar{\mathbf{a}}$, and zero-mean noise, $\widetilde{\mathbf{a}}$. Then the equations of dynamics (4) and (3) take the following form

$$\begin{aligned} \mathbf{v} \leftarrow\ & \mathbf{r}(\mathbf{v}, -(\boldsymbol{\omega}^G - \bar{\boldsymbol{\omega}} - \widetilde{\boldsymbol{\omega}})\delta) \\ & + \big(\mathbf{a}^A - \bar{\mathbf{a}} - \widetilde{\mathbf{a}} - \mathbf{r}(\mathbf{g}, -(\boldsymbol{\omega}^G - \bar{\boldsymbol{\omega}} - \widetilde{\boldsymbol{\omega}})\delta)\big)\delta, \end{aligned} \tag{6}$$

$$\mathbf{g} \leftarrow \mathbf{r}(\mathbf{g}, -(\boldsymbol{\omega}^G - \bar{\boldsymbol{\omega}} - \widetilde{\boldsymbol{\omega}})\delta). \tag{7}$$

In order to apply Extended Kalman Filter, the above model needs to be linearised. We notice that $\boldsymbol{\beta} = (\boldsymbol{\omega}^G - \bar{\boldsymbol{\omega}} - \widetilde{\boldsymbol{\omega}})\delta$ is a vector of angles by which IMU rotates within time δ. These angles must be small (that is, δ must be small enough), otherwise it can not be assumed that $\mathbf{a}$ and $\boldsymbol{\omega}$ are approximately constant within that time. Therefore, the above model is linearised around $\boldsymbol{\beta} = \mathbf{0}$. To this end let us denote

$$\mathbf{D}(\mathbf{x}) = \left.\frac{\partial \mathbf{r}(\mathbf{x},\boldsymbol{\beta})}{\partial \boldsymbol{\beta}}\right|_{\boldsymbol{\beta}=\mathbf{0}} = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix}, \tag{8}$$

and assume

$$\mathbf{r}(\mathbf{x},\boldsymbol{\beta}) \cong \mathbf{x} + \mathbf{D}(\mathbf{x})\boldsymbol{\beta}. \tag{9}$$

Application of (9) to eliminate $\mathbf{r}$ from (6) and (7) transforms these equations to the following linear form

$$\begin{aligned} \mathbf{v} \leftarrow\ & \mathbf{v} - \mathbf{D}(\mathbf{v})\boldsymbol{\omega}^G\delta + \mathbf{D}(\mathbf{v})\bar{\boldsymbol{\omega}}\delta + \mathbf{D}(\mathbf{v})\widetilde{\boldsymbol{\omega}}\delta \\ & + \mathbf{a}^A\delta - \bar{\mathbf{a}}\delta - \widetilde{\mathbf{a}}\delta - \mathbf{g}\delta \\ & + \mathbf{D}(\mathbf{g})\boldsymbol{\omega}^G\delta^2 - \mathbf{D}(\mathbf{g})\bar{\boldsymbol{\omega}}\delta^2 - \mathbf{D}(\mathbf{g})\widetilde{\boldsymbol{\omega}}\delta^2, \end{aligned} \tag{10}$$

$$\mathbf{g} \leftarrow \mathbf{g} - \mathbf{D}(\mathbf{g})\boldsymbol{\omega}^G\delta + \mathbf{D}(\mathbf{g})\bar{\boldsymbol{\omega}}\delta + \mathbf{D}(\mathbf{g})\widetilde{\boldsymbol{\omega}}\delta. \tag{11}$$

Let each component of $\widetilde{\boldsymbol{\omega}}$ and $\widetilde{\mathbf{a}}$ be stochastically independent and have standard deviations, respectively

$$\sigma_{\boldsymbol{\omega}} > 0, \quad \sigma_{\mathbf{a}} > 0. \tag{12}$$

For gyroscope and accelerometer bias we assume that they drift in random walk fashion, i.e.,

$$\bar{\mathbf{a}} \leftarrow \bar{\mathbf{a}} + \boldsymbol{\xi}_{\mathbf{a}} \tag{13}$$

$$\bar{\boldsymbol{\omega}} \leftarrow \bar{\boldsymbol{\omega}} + \boldsymbol{\xi}_{\boldsymbol{\omega}} \tag{14}$$

for $\boldsymbol{\xi}_{\mathbf{a}}$ and $\boldsymbol{\xi}_{\boldsymbol{\omega}}$ being zero-mean uncorrelated noise vectors with standard deviations, respectively

$$\sigma_{\boldsymbol{\xi}_{\boldsymbol{\omega}}} > 0, \quad \sigma_{\boldsymbol{\xi}_{\mathbf{a}}} > 0. \tag{15}$$

Then, eqs. (10), (11), (13), and (14) may be written in the following matrix form

$$\begin{bmatrix}\mathbf{v}\\ \mathbf{g}\\ \bar{\mathbf{a}}\\ \bar{\boldsymbol{\omega}}\end{bmatrix} \leftarrow \mathbf{F}\begin{bmatrix}\mathbf{v}\\ \mathbf{g}\\ \bar{\mathbf{a}}\\ \bar{\boldsymbol{\omega}}\end{bmatrix} - \mathbf{C}\begin{bmatrix}\mathbf{a}^A/\sigma_{\mathbf{a}}\\ \boldsymbol{\omega}^G/\sigma_{\boldsymbol{\omega}}\\ \mathbf{0}\\ \mathbf{0}\end{bmatrix} + \mathbf{C}\begin{bmatrix}\widetilde{\mathbf{a}}/\sigma_{\mathbf{a}}\\ \widetilde{\boldsymbol{\omega}}/\sigma_{\boldsymbol{\omega}}\\ \boldsymbol{\xi}_{\mathbf{a}}/\sigma_{\boldsymbol{\xi}_{\mathbf{a}}}\\ \boldsymbol{\xi}_{\boldsymbol{\omega}}/\sigma_{\boldsymbol{\xi}_{\boldsymbol{\omega}}}\end{bmatrix} \tag{16}$$

where

$$\mathbf{F} = \begin{bmatrix}\mathbf{I} & -\mathbf{I}\delta & -\mathbf{I}\delta & \mathbf{D}(\mathbf{v})\delta - \mathbf{D}(\mathbf{g})\delta^2\\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{D}(\mathbf{g})\delta\\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}\end{bmatrix}, \tag{17}$$

$$\mathbf{C} = \begin{bmatrix}-\mathbf{I}\delta\sigma_{\mathbf{a}} & \mathbf{D}(\mathbf{v})\delta\sigma_{\boldsymbol{\omega}} - \mathbf{D}(\mathbf{g})\delta^2\sigma_{\boldsymbol{\omega}} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{D}(\mathbf{g})\delta\sigma_{\boldsymbol{\omega}} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{I}\sigma_{\boldsymbol{\xi}_{\mathbf{a}}} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}\sigma_{\boldsymbol{\xi}_{\boldsymbol{\omega}}}\end{bmatrix}. \tag{18}$$

D. Model of observation

The observation model takes the form

$$\mathbf{v}^O = \mathbf{v} + \widetilde{\mathbf{v}}^O \tag{19}$$

where $\widetilde{\mathbf{v}}^O$ is the error of velocity observation from the external measurement system (based on kinematics or GPS). We assume that components of $\widetilde{\mathbf{v}}^O$ are stochastically independent and have standard deviations equal to

$$\sigma_{\mathbf{v}} > 0. \tag{20}$$

E. Integration

In the integration phase, Extended Kalman Filter is used to estimate the state of the inertial sensor. The following notation is applied in the algorithm below:

$\widehat{\mathbf{v}}, \widehat{\mathbf{g}}, \widehat{\bar{\mathbf{a}}}, \widehat{\bar{\boldsymbol{\omega}}}$ — state estimate composed of, respectively, the estimates of velocity, gravity vector, accelerometer bias, and gyroscope bias,

$\mathbf{P} \in R^{12\times 12}$ — covariance matrix of state estimates,

$\mathbf{K} \in R^{12\times 3}$— Kalman gain.

Initialization:

It is assumed that the robot is initially immobile. The variables are initialized as:

$$\widehat{\mathbf{v}} \leftarrow \mathbf{0}, \tag{21}$$

$$\widehat{\mathbf{g}} \leftarrow \mathbf{a}^A g_0/\|\mathbf{a}^A\|, \tag{22}$$

$$\widehat{\bar{\mathbf{a}}} \leftarrow \mathbf{a}^A - \widehat{\mathbf{g}}, \tag{23}$$

$$\widehat{\bar{\boldsymbol{\omega}}} \leftarrow \widehat{\boldsymbol{\omega}}, \tag{24}$$

$$\mathbf{P} \leftarrow \begin{bmatrix}\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \sigma_A^2\mathbf{I} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \sigma_A^2\mathbf{I} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \sigma_{\omega}^2\mathbf{I}\end{bmatrix}, \tag{25}$$

where $g_0 = 9.81[ms^{-2}]$ is the Earth gravity acceleration, and σ_A^2 is the variance of error of bare accelerometer readout.

(If the assumption about robot's immobility is not valid, the above initialization should be replaced with one based on the state estimation from the kinematic model.)

Prediction:

Prediction takes place every δ sec. It starts with taking the measurements of acceleration, $\mathbf{a}^A$, and angular velocity, $\boldsymbol{\omega}^G$.

State estimate update:

$$\widehat{\mathbf{v}} \leftarrow \mathbf{r}(\widehat{\mathbf{v}}, -(\boldsymbol{\omega}^G - \widehat{\bar{\boldsymbol{\omega}}})\delta) \tag{26}$$

$$+ \big(\mathbf{a}^A - \widehat{\bar{\mathbf{a}}} - \mathbf{r}(\widehat{\mathbf{g}}, -(\boldsymbol{\omega}^G - \widehat{\bar{\boldsymbol{\omega}}})\delta)\big)\delta, \tag{27}$$

$$\widehat{\mathbf{g}} \leftarrow \mathbf{r}(\widehat{\mathbf{g}}, -(\boldsymbol{\omega}^G - \widehat{\bar{\boldsymbol{\omega}}})\delta). \tag{28}$$

State estimate covariance matrix update with $\mathbf{F}$ (17) and $\mathbf{C}$ (18):

$$\mathbf{P} \leftarrow \mathbf{F}\mathbf{P}\mathbf{F}^T + \mathbf{C}\mathbf{C}^T. \tag{29}$$

Correction:

Correction only takes place when it is possible to take an observation of velocity, $\mathbf{v}^O$.

Velocity error:

$$\widetilde{\mathbf{v}} = \mathbf{v}^O - \widehat{\mathbf{v}} \tag{30}$$

Auxiliary matrices:

$$\mathbf{H} = [\mathbf{I}\;\mathbf{0}\;\mathbf{0}\;\mathbf{0}] \tag{31}$$

$$\mathbf{S} = \mathbf{H}\mathbf{P}\mathbf{H}^T + \mathbf{I}\sigma_{\mathbf{v}}^2, \tag{32}$$

where $\mathbf{0}$ and $\mathbf{I}$ are appropriate 3×3 matrices.

Kalman gain:

$$\mathbf{K} = \mathbf{P}\mathbf{H}^T\mathbf{S}^{-1}. \tag{33}$$

Correction:

$$\begin{bmatrix}\widehat{\mathbf{v}}\\ \widehat{\mathbf{g}}\\ \widehat{\bar{\mathbf{a}}}\\ \widehat{\bar{\boldsymbol{\omega}}}\end{bmatrix} \leftarrow \begin{bmatrix}\widehat{\mathbf{v}}\\ \widehat{\mathbf{g}}\\ \widehat{\bar{\mathbf{a}}}\\ \widehat{\bar{\boldsymbol{\omega}}}\end{bmatrix} + \mathbf{K}\widetilde{\mathbf{v}}. \tag{34}$$

State estimate covariance matrix update:

$$\mathbf{P} \leftarrow (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{P}. \tag{35}$$

IV. Simplified model

The algorithm presented above combines available measurements to compute accurate estimates of velocity, tilt, and IMU biases. Its main area of application is legged robotics, where available computational power is sufficient for on-line calculation of 12×12 matrices. However, the same estimates may be required in small mobile robots, nanorobots, or just sensors, where computing power may be insufficient. The number of computations (along with implementation burden) may be significantly reduced at the cost of limited accuracy deterioration. The idea may be summarized in two points:

- The filter is transformed such that all its calculations are performed with the use of matrices that are composed of 3×3 diagonal matrices.
- The calculations may be performed with 3-fold smaller matrices in which each element represents the diagonal of the submatrix in the original matrix. E.g., matrix summation is replaced by scalar summation: let $a, b \in \Re$, then $(a\mathbf{I}) + (b\mathbf{I}) = (a+b)\mathbf{I}$.

The filter is transformed in two steps:

- In order for the $\mathbf{F}$ matrix (17) to be composed of diagonal ones, some of the state variables of the system are redefined.
- Another matrix in the original Kalman Filter operations that is not composed of 3×3 diagonal ones, is $\mathbf{CC}^T$ in (29). It represents a covariance matrix of the noise that is added to state estimates due to errors in measurements. It encompasses gyroscope error which rotates $\widehat{\mathbf{g}}$ and $\widehat{\mathbf{v}}$. The matrix $\mathbf{CC}^T$ in (29) is replaced by a matrix that is composed of two 3×3 diagonal matrices. This new matrix represents a covariance matrix of a noise that would modify $\widehat{\mathbf{g}}$ and $\widehat{\mathbf{v}}$ additively and cover influence of the actual noise in the gyroscope readout.

In order to replace $\mathbf{D}(\mathbf{g})$ and $\mathbf{D}(\mathbf{v})$ in $\mathbf{F}$ (17) and in $\mathbf{C}$ (18) with diagonal matrices, new state variables are defined, namely

$$\boldsymbol{\mu} = \mathbf{D}(\mathbf{v})\bar{\boldsymbol{\omega}} \quad \text{and} \quad \boldsymbol{\rho} = \mathbf{D}(\mathbf{g})\bar{\boldsymbol{\omega}}, \tag{36}$$

and applied together instead of $\bar{\boldsymbol{\omega}}$. However, once their adjustments have been computed with the use of EKF equations, $\bar{\boldsymbol{\omega}}$ will be adjusted instead. The new variables drift in time according to

$$\boldsymbol{\mu} \leftarrow \boldsymbol{\mu} + \mathbf{D}(\mathbf{v})\boldsymbol{\xi}_{\boldsymbol{\omega}}, \quad \boldsymbol{\rho} \leftarrow \boldsymbol{\rho} + \mathbf{D}(\mathbf{g})\boldsymbol{\xi}_{\boldsymbol{\omega}}. \tag{37}$$

The new linearised model follows

$$\begin{bmatrix}\mathbf{v}\\ \mathbf{g}\\ \bar{\mathbf{a}}\\ \boldsymbol{\mu}\\ \boldsymbol{\rho}\end{bmatrix} \leftarrow \mathbf{F}\begin{bmatrix}\mathbf{v}\\ \mathbf{g}\\ \bar{\mathbf{a}}\\ \boldsymbol{\mu}\\ \boldsymbol{\rho}\end{bmatrix} - \mathbf{B}\begin{bmatrix}\mathbf{a}^A/\sigma_{\mathbf{a}}\\ \boldsymbol{\omega}^G/\sigma_{\boldsymbol{\omega}}\end{bmatrix} + \mathbf{C}\begin{bmatrix}\widetilde{\mathbf{a}}/\sigma_{\mathbf{a}}\\ \widetilde{\boldsymbol{\omega}}/\sigma_{\boldsymbol{\omega}}\\ \boldsymbol{\xi}_{\mathbf{a}}/\sigma_{\boldsymbol{\xi}_{\mathbf{a}}}\\ \boldsymbol{\xi}_{\boldsymbol{\omega}}/\sigma_{\boldsymbol{\xi}_{\boldsymbol{\omega}}}\end{bmatrix} \tag{38}$$

for

$$\mathbf{F} = \begin{bmatrix}\mathbf{I} & -\mathbf{I}\delta & -\mathbf{I}\delta & \mathbf{I}\delta & -\mathbf{I}\delta^2\\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{I}\delta\\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}\end{bmatrix}, \tag{39}$$

$$\mathbf{B} = \begin{bmatrix}-\mathbf{I}\delta\sigma_{\mathbf{a}} & \mathbf{D}(\mathbf{v})\delta\sigma_{\boldsymbol{\omega}} - \mathbf{D}(\mathbf{g})\delta^2\sigma_{\boldsymbol{\omega}}\\ \mathbf{0} & \mathbf{D}(\mathbf{g})\delta\sigma_{\boldsymbol{\omega}}\\ \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{bmatrix}, \tag{40}$$

$$\mathbf{C} = \begin{bmatrix}-\mathbf{I}\delta\sigma_{\mathbf{a}} & \mathbf{D}(\mathbf{v})\delta\sigma_{\boldsymbol{\omega}} - \mathbf{D}(\mathbf{g})\delta^2\sigma_{\boldsymbol{\omega}} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{D}(\mathbf{g})\delta\sigma_{\boldsymbol{\omega}} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{I}\sigma_{\boldsymbol{\xi}_{\mathbf{a}}} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{D}(\mathbf{v})\sigma_{\boldsymbol{\xi}_{\boldsymbol{\omega}}}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{D}(\mathbf{g})\sigma_{\boldsymbol{\xi}_{\boldsymbol{\omega}}}\end{bmatrix} \tag{41}$$

It is seen that $\mathbf{F}$ (39) is composed of diagonal matrices. While $\mathbf{B}$ (40) does not have such property, it is not a problem because this matrix does not take part in EKF computations. The matrix that may replace $\mathbf{CC}^T$ in the Kalman prediction step (29) takes the form

$$\boldsymbol{\Omega} = \begin{bmatrix}(e^2 + 2\|\mathbf{p}\|^2)\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & 2\|\mathbf{q}\|^2\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \sigma^2_{\boldsymbol{\xi}_{\mathbf{a}}}\mathbf{I} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & 2\|\mathbf{v}\|^2\sigma^2_{\boldsymbol{\xi}_{\boldsymbol{\omega}}}\mathbf{I} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & 2\|\mathbf{g}\|^2\sigma^2_{\boldsymbol{\xi}_{\boldsymbol{\omega}}}\mathbf{I}\end{bmatrix} \tag{42}$$

with

$$e = -\sigma_{\mathbf{a}}\delta \tag{43}$$

$$\mathbf{p} = \widehat{\mathbf{v}}\sigma_{\boldsymbol{\omega}}\delta - \widehat{\mathbf{g}}\sigma_{\boldsymbol{\omega}}\delta^2 \tag{44}$$

$$\mathbf{q} = \widehat{\mathbf{g}}\sigma_{\boldsymbol{\omega}}\delta. \tag{45}$$

Noise with covariance matrix $\boldsymbol{\Omega}$ covers the noise with covariance matrix $\mathbf{CC}^T$ in the sense that

$$\boldsymbol{\Omega} \geq \mathbf{CC}^T. \tag{46}$$

(The above inequality formally means that $\boldsymbol{\Omega} - \mathbf{CC}^T$ is a non-negatively defined matrix i.e., $\forall \mathbf{z}\ \mathbf{z}^T(\boldsymbol{\Omega} - \mathbf{CC}^T)\mathbf{z} \geq 0$.) The second of the following lemmas proves (46).

Lemma 1: For each $\mathbf{x} \in R^3$, it is true that

$$\mathbf{D}(\mathbf{x})\mathbf{D}(\mathbf{x})^T \leq \|\mathbf{x}\|^2\mathbf{I}. \tag{47}$$

Proof: Let $\mathbf{y} \in R^3$. Eq. (8) is applied to obtain

$$\begin{aligned}
&\mathbf{y}^T\left(\|\mathbf{x}\|^2 I - \mathbf{D}(\mathbf{x})\mathbf{D}(\mathbf{x})^T\right)\mathbf{y}\\
&= \mathbf{y}^T\left(\begin{bmatrix}\|\mathbf{x}\|^2 & 0 & 0\\ 0 & \|\mathbf{x}\|^2 & 0\\ 0 & 0 & \|\mathbf{x}\|^2\end{bmatrix} - \begin{bmatrix}x_2^2 + x_3^2 & -x_1x_2 & -x_1x_3\\ -x_1x_2 & x_1^2 + x_3^2 & -x_2x_3\\ -x_1x_3 & -x_2x_3 & x_1^2 + x_2^2\end{bmatrix}\right)\mathbf{y}\\
&= \mathbf{y}^T\begin{bmatrix}x_1^2 & x_1x_2 & x_1x_3\\ x_1x_2 & x_2^2 & x_2x_3\\ x_1x_3 & x_2x_3 & x_3^2\end{bmatrix}\mathbf{y}\\
&= \mathbf{y}^T(\mathbf{x}\mathbf{x}^T)\mathbf{y} = (\mathbf{y}^T\mathbf{x})^2\\
&\geq 0.
\end{aligned}$$

■

Lemma 2: For $\mathbf{C}$ (41) and $\boldsymbol{\Omega}$ (42),

$$\boldsymbol{\Omega} \geq \mathbf{CC}^T. \tag{48}$$

Proof: Let us decompose $\mathbf{C}$ (41) into

$$\mathbf{C} = \begin{bmatrix}\mathbf{C}_1 & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{I}\sigma_{\boldsymbol{\xi}_{\mathbf{a}}} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{C}_3\end{bmatrix}$$

with $\mathbf{C}_1$ and $\mathbf{C}_3$ such that

$$\mathbf{C}_1 = \begin{bmatrix}\mathbf{I}e & \mathbf{P}\\ \mathbf{0} & \mathbf{Q}\end{bmatrix}, \quad \mathbf{C}_3 = \begin{bmatrix}\mathbf{D}_1\\ \mathbf{D}_2\end{bmatrix}$$

where $\mathbf{P}, \mathbf{Q}, \mathbf{D}_1, \mathbf{D}_2 \in R^{3\times 3}$ and $e \in R$. Let $\mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_3, \mathbf{z}_4, \mathbf{z}_5 \in R^3$, and $\mathbf{z} = [\mathbf{z}_1^T, \mathbf{z}_2^T, \mathbf{z}_3^T, \mathbf{z}_4^T, \mathbf{z}_5^T]^T$. We look for a matrix not smaller than $\mathbf{CC}^T$. To this end, we notice that

$$\mathbf{z}^T\mathbf{CC}^T\mathbf{z} = \begin{bmatrix}\mathbf{z}_1^T\mathbf{z}_2^T\end{bmatrix}\mathbf{C}_1\mathbf{C}_1^T\begin{bmatrix}\mathbf{z}_1\\ \mathbf{z}_2\end{bmatrix} + \|\mathbf{z}_3\|^2\sigma^2_{\boldsymbol{\xi}_{\mathbf{a}}} + \begin{bmatrix}\mathbf{z}_4^T\mathbf{z}_5^T\end{bmatrix}\mathbf{C}_3\mathbf{C}_3^T\begin{bmatrix}\mathbf{z}_4\\ \mathbf{z}_5\end{bmatrix}. \tag{49}$$

Let us analyse the elements of the above sum in order:

$$\begin{aligned}
&[\mathbf{z}_1^T\mathbf{z}_2^T]\,\mathbf{C}_1\mathbf{C}_1^T\begin{bmatrix}\mathbf{z}_1\\ \mathbf{z}_2\end{bmatrix} = [\mathbf{z}_1^T\mathbf{z}_2^T]\begin{bmatrix}\mathbf{I}e^2+\mathbf{P}\mathbf{P}^T & \mathbf{P}\mathbf{Q}^T\\ \mathbf{Q}\mathbf{P}^T & \mathbf{Q}\mathbf{Q}^T\end{bmatrix}\begin{bmatrix}\mathbf{z}_1\\ \mathbf{z}_2\end{bmatrix}\\
&= \mathbf{z}_1^T(e^2\mathbf{I}+\mathbf{P}\mathbf{P}^T)\mathbf{z}_1 + \mathbf{z}_1^T\mathbf{P}\mathbf{Q}^T\mathbf{z}_2 + \mathbf{z}_2^T\mathbf{Q}\mathbf{P}^T\mathbf{z}_1 + \mathbf{z}_2^T\mathbf{Q}\mathbf{Q}^T\mathbf{z}_2\\
&\le \mathbf{z}_1^T(e^2\mathbf{I}+\mathbf{P}\mathbf{P}^T)\mathbf{z}_1 + \mathbf{z}_1^T\mathbf{P}\mathbf{Q}^T\mathbf{z}_2 + \mathbf{z}_2^T\mathbf{Q}\mathbf{P}^T\mathbf{z}_1 + \mathbf{z}_2^T\mathbf{Q}\mathbf{Q}^T\mathbf{z}_2\\
&\quad + \|\mathbf{P}\mathbf{z}_1-\mathbf{Q}\mathbf{z}_2\|^2\\
&= \mathbf{z}_1^T(e^2\mathbf{I}+\mathbf{P}\mathbf{P}^T)\mathbf{z}_1 + (\mathbf{z}_1^T\mathbf{P}\mathbf{P}^T\mathbf{z}_1+\mathbf{z}_2^T\mathbf{Q}\mathbf{Q}^T\mathbf{z}_2) + \mathbf{z}_2^T\mathbf{Q}\mathbf{Q}^T\mathbf{z}_2\\
&= [\mathbf{z}_1^T\mathbf{z}_2^T]\begin{bmatrix}e^2\mathbf{I}+2\mathbf{P}\mathbf{P}^T & \mathbf{0}\\ \mathbf{0} & 2\mathbf{Q}\mathbf{Q}^T\end{bmatrix}\begin{bmatrix}\mathbf{z}_1\\ \mathbf{z}_2\end{bmatrix}\\
&= [\mathbf{z}_1^T\mathbf{z}_2^T]\,\mathbf{\Omega}_1\begin{bmatrix}\mathbf{z}_1\\ \mathbf{z}_2\end{bmatrix}.
\end{aligned}$$

In the last equality, the matrix $\mathbf{\Omega}_1$ is defined such that $\mathbf{\Omega}_1 \ge \mathbf{C}_1\mathbf{C}_1^T$.

$$\begin{aligned}
&[\mathbf{z}_4^T\mathbf{z}_5^T]\,\mathbf{C}_3\mathbf{C}_3^T\begin{bmatrix}\mathbf{z}_4\\ \mathbf{z}_5\end{bmatrix} = [\mathbf{z}_4^T\mathbf{z}_5^T]\begin{bmatrix}\mathbf{D}_1\mathbf{D}_1^T & \mathbf{D}_1\mathbf{D}_2^T\\ \mathbf{D}_2\mathbf{D}_1^T & \mathbf{D}_2\mathbf{D}_2^T\end{bmatrix}\begin{bmatrix}\mathbf{z}_4\\ \mathbf{z}_5\end{bmatrix}\\
&= \mathbf{z}_4^T\mathbf{D}_1\mathbf{D}_1^T\mathbf{z}_4 + \mathbf{z}_4^T\mathbf{D}_1\mathbf{D}_2^T\mathbf{z}_5 + \mathbf{z}_5^T\mathbf{D}_2\mathbf{D}_1^T\mathbf{z}_4 + \mathbf{z}_5^T\mathbf{D}_2\mathbf{D}_2^T\mathbf{z}_5\\
&\le \mathbf{z}_4^T\mathbf{D}_1\mathbf{D}_1^T\mathbf{z}_4 + \mathbf{z}_4^T\mathbf{D}_1\mathbf{D}_2^T\mathbf{z}_5 + \mathbf{z}_5^T\mathbf{D}_2\mathbf{D}_1^T\mathbf{z}_4 + \mathbf{z}_5^T\mathbf{D}_2\mathbf{D}_2^T\mathbf{z}_5\\
&\quad + \|\mathbf{D}_1\mathbf{z}_4-\mathbf{D}_2\mathbf{z}_5\|^2\\
&= \mathbf{z}_4^T 2\mathbf{D}_1\mathbf{D}_1^T\mathbf{z}_4 + \mathbf{z}_5^T 2\mathbf{D}_2\mathbf{D}_2^T\mathbf{z}_5\\
&= [\mathbf{z}_4^T\mathbf{z}_5^T]\begin{bmatrix}2\mathbf{D}_1\mathbf{D}_1^T & \mathbf{0}\\ \mathbf{0} & 2\mathbf{D}_2\mathbf{D}_2^T\end{bmatrix}\begin{bmatrix}\mathbf{z}_4\\ \mathbf{z}_5\end{bmatrix}\\
&= [\mathbf{z}_4^T\mathbf{z}_5^T]\,\mathbf{\Omega}_3\begin{bmatrix}\mathbf{z}_4\\ \mathbf{z}_5\end{bmatrix}.
\end{aligned}$$

The last equality defines the $\mathbf{\Omega}_3$ matrix. Inequality (47) is applied to submatrices in $\mathbf{\Omega}_1$ and $\mathbf{\Omega}_3$. To this end, we notice that $\mathbf{P}$, $\mathbf{Q}$, $\mathbf{D}_1$, and $\mathbf{D}_2$ are of the form $\mathbf{D}(\mathbf{x})$ (8) with $\mathbf{x}$ equal to, respectively, $\mathbf{p}$ (44), $\mathbf{q}$ (45), $\mathbf{v}\sigma_{\boldsymbol{\xi}_\omega}$, and $\mathbf{g}\sigma_{\boldsymbol{\xi}_\omega}$. Therefore, we obtain

$$\mathbf{C}\mathbf{C}^T \le \begin{bmatrix}\mathbf{\Omega}_1 & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{I}\sigma^2_{\boldsymbol{\xi}_\mathbf{a}} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{\Omega}_3\end{bmatrix} \le \mathbf{\Omega}$$

which completes the proof. ■

The last thing required to present EKF equations for the redefined model is adjustment of gyroscope bias, $\bar{\boldsymbol{\omega}}$, on the basis of vectors that are to modify $\boldsymbol{\mu}$ and $\boldsymbol{\rho}$ (36). The following fact is helpful in this regard. Let $\mathbf{z} \in R^3$ be a direction in which $\mathbf{D}(\mathbf{x})\mathbf{y}$ needs to be pushed by adjusting $\mathbf{y}$. The lemma below states that the adjustment should be equal to $\|\mathbf{x}\|^{-2}\mathbf{D}(\mathbf{x})^T\mathbf{z}$.

Lemma 3: Let $\mathbf{x},\mathbf{y},\mathbf{z} \in R^3$. The function

$$L(\mathbf{y}) = \|\mathbf{D}(\mathbf{x})\mathbf{y}-\mathbf{z}\|^2 \tag{50}$$

is minimized for $\mathbf{y} = \|\mathbf{x}\|^{-2}\mathbf{D}(\mathbf{x})^T\mathbf{z}$.

Proof: L is a quadratic function with the gradient equal to

$$\nabla L(\mathbf{y}) = 2\mathbf{D}(\mathbf{x})^T\mathbf{D}(\mathbf{x})\mathbf{y} - 2\mathbf{D}(\mathbf{x})^T\mathbf{z}. \tag{51}$$

L attains its minimum for $\mathbf{y}$ such that $\nabla L(\mathbf{y}) = \mathbf{0}$. Straightforward calculus reveals that

$$\mathbf{D}(\mathbf{x})^T\mathbf{D}(\mathbf{x})\mathbf{D}(\mathbf{x})^T\mathbf{z} = \|\mathbf{x}\|^2\mathbf{D}(\mathbf{x})^T\mathbf{z}. \tag{52}$$

Now let us take $\mathbf{y} = \|\mathbf{x}\|^{-2}\mathbf{D}(\mathbf{x})^T\mathbf{z}$ and compute $\nabla L(\mathbf{y})$. Using the above equation we get $\nabla J(\mathbf{y}) = 0$. ■

Replacing $\mathbf{C}\mathbf{C}^T$ in the Kalman prediction with $\mathbf{\Omega}$ (42), Kalman Filter equations may be rewritten in the following new form that involves smaller matrices.

Initialization:

It is assumed that the robot is initially immobile.

$$\widehat{\mathbf{v}} \leftarrow \mathbf{0}, \tag{53}$$
$$\widehat{\mathbf{g}} \leftarrow \mathbf{a}^A g_0/\|\mathbf{a}^A\|, \tag{54}$$
$$\widehat{\bar{\mathbf{a}}} \leftarrow \mathbf{a}^A - \widehat{\mathbf{g}}, \tag{55}$$
$$\widehat{\bar{\boldsymbol{\omega}}} \leftarrow \widehat{\boldsymbol{\omega}}, \tag{56}$$
$$\mathbf{P} \leftarrow \begin{bmatrix}0&0&0&0&0\\0&\sigma_A^2&0&0&0\\0&0&\sigma_A^2&0&0\\0&0&0&0&0\\0&0&0&0&g_0^2\sigma^2_{\boldsymbol{\omega}}\end{bmatrix}. \tag{57}$$

Prediction:

State estimate update:

$$\begin{aligned}\widehat{\mathbf{v}} \leftarrow\ & \mathbf{r}(\widehat{\mathbf{v}}, -(\boldsymbol{\omega}^G-\widehat{\bar{\boldsymbol{\omega}}})\delta)\\ &+ \big(\mathbf{a}^A - \widehat{\bar{\mathbf{a}}} - \mathbf{r}(\widehat{\mathbf{g}}, -(\boldsymbol{\omega}^G-\widehat{\bar{\boldsymbol{\omega}}})\delta)\big)\delta,\end{aligned} \tag{58}$$
$$\widehat{\mathbf{g}} \leftarrow \mathbf{r}(\widehat{\mathbf{g}}, -(\boldsymbol{\omega}^G-\widehat{\bar{\boldsymbol{\omega}}})\delta). \tag{59}$$

State estimate covariance matrix update:

$$\mathbf{P} \leftarrow \mathbf{F}\mathbf{P}\mathbf{F}^T + \mathbf{\Omega} \tag{60}$$

for

$$\mathbf{F} = \begin{bmatrix}1&-\delta&-\delta&\delta&-\delta^2\\0&1&0&0&\delta\\0&0&1&0&0\\0&0&0&1&0\\0&0&0&0&1\end{bmatrix}, \tag{61}$$

$$\mathbf{\Omega} = \begin{bmatrix}e^2+2\|\mathbf{p}\|^2&0&0&0&0\\0&2\|\mathbf{q}\|^2&0&0&0\\0&0&\sigma^2_{\boldsymbol{\xi}_\mathbf{a}}&0&0\\0&0&0&2\|\widehat{\mathbf{v}}\|^2\sigma^2_{\boldsymbol{\xi}_\omega}&0\\0&0&0&0&2\|\widehat{\mathbf{g}}\|^2\sigma^2_{\boldsymbol{\xi}_\omega}\end{bmatrix} \tag{62}$$

Correction:

Correction only takes place when it is possible to take an observation of velocity, $\mathbf{v}^O$, from the external measurement system (based on kinematics or GPS).

Velocity error:

$$\widetilde{\mathbf{v}} = \mathbf{v}^O - \widehat{\mathbf{v}} \tag{63}$$

Auxiliary matrices:

$$\mathbf{H} = [1\ 0\ 0\ 0\ 0] \tag{64}$$
$$\mathbf{S} = \mathbf{H}\mathbf{P}\mathbf{H}^T + \sigma^2_{\mathbf{v}}. \tag{65}$$

Kalman gain:

$$\mathbf{K} = \mathbf{P}\mathbf{H}^T/S. \tag{66}$$

Correction (the term ϵ is a small constant that prevents division by zero):

$$\widehat{\mathbf{v}} \leftarrow \widehat{\mathbf{v}} + K_1\widetilde{\mathbf{v}} \quad (67)$$

$$\widehat{\mathbf{g}} \leftarrow \widehat{\mathbf{g}} + K_2\widetilde{\mathbf{v}} \quad (68)$$

$$\widehat{\mathbf{a}} \leftarrow \widehat{\mathbf{a}} + K_3\widetilde{\mathbf{v}} \quad (69)$$

$$\widehat{\boldsymbol{\omega}} \leftarrow \widehat{\boldsymbol{\omega}} + (\epsilon + \|\widehat{\mathbf{v}}\|)^{-2}\mathbf{D}(\widehat{\mathbf{v}})K_4\widetilde{\mathbf{v}} \quad (70)$$

$$+ \|\widehat{\mathbf{g}}\|^{-2}\mathbf{D}(\widehat{\mathbf{g}})K_5\widetilde{\mathbf{v}} \quad (71)$$

State estimate covariance matrix update:

$$\mathbf{P} \leftarrow (\mathbf{I} - \mathbf{KH})\mathbf{P}. \quad (72)$$

Discussion

As the original algorithm discussed in Sec. III the simplified version attributes the error $\widetilde{\mathbf{v}}$ (63) to noise and errors in the state estimates. The errors in $\widehat{\mathbf{v}}$, $\widehat{\mathbf{g}}$, and $\widehat{\mathbf{a}}$ influence $\widetilde{\mathbf{v}}$ additively. The simplified filter just pushes these estimates proportionally to $\widetilde{\mathbf{v}}$; the original filter basically does the same. However, influence of $\widehat{\boldsymbol{\omega}}$ inflicted on $\widetilde{\mathbf{v}}$ is more complex. Namely, $\widehat{\boldsymbol{\omega}}$ rotates $\widehat{\mathbf{v}}$ and $\widehat{\mathbf{g}}$ in filter prediction, (58) and (59). The simplified filter attributes appropriate part of $\widetilde{\mathbf{v}}$ to erratic rotation of $\widehat{\mathbf{v}}$ (through $\boldsymbol{\mu}$) and erratic rotation of $\widehat{\mathbf{g}}$ (through $\boldsymbol{\rho}$). Then, these errors in $\boldsymbol{\mu}$ and $\boldsymbol{\rho}$ are translated into the error in $\widehat{\boldsymbol{\omega}}$ and corrected.

The dominating operation in the above algorithm is multiplication in the matrix operation involved in prediction. This requires $2 \times 5^3 = 250$ multiplications while in the original version of the algorithm presented in Sec. III there are $2 \times 12^3 = 3456$ multiplications. The simplified version is thus about 14 times more economical.

V. Experimental results

In order to verify the presented method the following experiment is performed:

- The same Bioloid is taken as was used in [6]. It is a 35 cm tall humanoid robot, additionally equipped with IMU (ADIS 16365), touch sensors in the feet, and a small PC in its “backpack” as a controller.
- The robot takes a one minute walk. Its velocity, tilt, and inertial sensors biases are being estimated with the use of the original method presented in [6] and the simplified method presented here.

In order to compare the estimates, the relative measure of discrepancy is applied

$$e = \frac{\sum_{t=1}^{T} \|\widehat{\mathbf{x}}_t^1 - \widehat{\mathbf{x}}_t^2\|^2}{\sum_{t=1}^{T} \|0.5(\widehat{\mathbf{x}}_t^1 + \widehat{\mathbf{x}}_t^2)\|^2}$$

where $\widehat{x}_t^1$ is an estimate taken at time t with one method, and $\widehat{x}_t^2$ is taken with the other one. The results are presented in Tab. I.

It is seen that the relative discrepancies are very small, especially those of velocity and tilt, which are actually useful. The relative discrepancies in biases estimates are noticeably larger, but still sufficiently small.

TABLE I
DISCREPANCIES BETWEEN THE ESTIMATES

value	discrepancy
velocity	$5.3 \cdot 10^{-4}$
tilt	$3.8 \cdot 10^{-8}$
acc bias	0.01
gyr bias	0.08

VI. Conclusions

In this paper a method was proposed for velocity and tilt estimation based on IMU and a velocity measurement system whose readouts are possibly erratic and temporarily unavailable. The method achieves similar accuracy as the method introduced in [6] but it is 14 times computationally more economical, which makes this approach readily implementable in sensors and low power controllers.

References

[1] J. Borenstein, H. R. Everett, L. Feng, and D. K. Wehe, “Mobile robot positioning: Sensors and techniques.” *Journal of Robotic Systems, Special Issue on Mobile Robots*, vol. 14, no. 4, pp. 231–249, 1997.

[2] J. Z. Sasiadek and Q. Wang, “Low cost automation using ins/gps data fusion for accurate positioning.” *Robotica*, vol. 21, no. 3, pp. 255–260, 2003.

[3] B. Gassmann, F. Zacharias, J. Zollner, and R. Dillmann, “Localization of walking robots,” in *2005 IEEE International Conference on Robotics and Automation (ICRA ’05)*, 2005, pp. 1471–1476.

[4] J. Cobano, P. Estremera, and J. Gonzalez de Santos, “Location of legged robots in outdoor environments,” *Robotics and Autonomous Systems*, vol. 56, no. 9, pp. 751–761, 2008.

[5] S.-H. Liu, T.-S. Huang, and J.-Y. Yen, “Comparison of sensor fusion methods for an sma-based hexapod biomimetic robot,” *Robotics and Autonomous Systems*, vol. 58, no. 5, pp. 737–744, May 2010.

[6] P. Wawrzyński, J. Możaryn, and K. Klimaszewski, “Robust estimation of walking robots velocity and tilt using proprioceptive sensors data fusion,” *Robotics and Autonomous Systems*, vol. 66, pp. 44–54, 2015.

[7] M. Bloesch, M. Hutter, M. Hoepflinger, S. Leutenegger, C. Gehring, C. D. Remy, and R. Siegwart, “State estimation for legged robots - consistent fusion of leg kinematics and IMU,” in *Proceedings of Robotics: Science and Systems*, Sydney, Australia, July 2012.