24th Mediterranean Conference on Control and Automation (MED)
June 21-24, 2016, Athens, Greece

Quadruped Robot Roll and Pitch Estimation using an Unscented Kalman Filter*

Sotirios Nousias and Evangelos Papadopoulos, *Senior Member, IEEE*

Abstract—**We present a novel algorithm for estimating a quadruped robot's pitch and roll angles. Assuming even terrain and an ideal bounding gait, we compute the roll and pitch angles to be fused with on-board Inertial Measurement Unit (IMU) measurements in an unscented Kalman filter (UKF). Simulation results illustrate the validity of the methodology developed. It is shown that the error in the estimation of both angles is much smaller compared to those in the literature.**

I. Introduction

Quadruped robots have been proven to provide a promising locomotion system, capable of performing extraordinary tasks compared to conventional wheeled vehicles. Their advantage stems from the fact that they use isolated footholds that optimize support in contrast to the continuous path of support required by wheel vehicles. To achieve the desired performance, the controllers of such robots require both precise and real-time knowledge of the controlled states. Of all the states that can be chosen, pitch and roll are critical for dynamically stable gaits. The most common technique to obtain precise estimation is sensor fusion. To this end, a number of research groups use various kinds of available information, from leg kinematics to vision based techniques.

One of the earliest navigation systems based on leg-kinematics was presented by Roston et al. [1]. Several years later Lin et al., working with the RHex hexapod, and assuming that three of its feet are co-linear at every time, they implemented a leg-odometer technique based on the kinematics [2]. As their method was affected by drift, they fused that information with the data provided by an on-board Inertial Measurement Unit (IMU) and the results were improved [3]. Along these lines, Bloesch et al. fused IMU feedback and leg kinematics information within an Extended Kalman Filter (EKF) [4]. Their algorithm estimated within 0.5 deg. the roll and pitch angles of a quadruped, as well as its Center of Mass (CoM) velocity. The same group proposed an improved estimator for handling rough terrain [5].

Reinstein and Hoffman focused on velocity-aided estimators [6]. Their novel leg odometer was based on a combination of joint and pressure sensors, yielding the stride length in each gait period. The estimated velocity was used to update the Inertial Navigation System (INS) algorithm within an EKF. The main drawback of this method is that the estimator requires training, as the legged odometer is based on a data-driven model that relates sensory information to stride length. Chilian developed a technique for the fusion of leg-odometry, visual odometry and IMU data [7]. The multisensor data fusion resulted in robust and accurate pose estimation of the DLR crawler, even in poor lighting conditions. Singh considered optical flow based estimation, but the pitch angle experimental error was about one degree [8]. Additionally, optical flow adds significant delay, degrading control performance. Alternate feedback sources include Motion Capture systems (MoCap) and Global Positioning Systems (GPS). Despite the cost of the former, it is still limited to confined workspaces, while the latter lacks signal robustness, accuracy and availability in interior spaces.

In this paper, we develop a novel algorithm for estimating the orientation of a quadruped robot in bounding. Inspired by [2], we employ the quadruped full stance phase to calculate its absolute pitch and roll angles. The resulting estimates are used to improve the predicted states estimated using the on-board IMU measurements. It is assumed that measurements are affected by Gaussian noise. While results on estimating the rotation around an axis using more complex probabilistic distributions are encouraging, still a recursive estimation algorithm of 3D rotations is missing [9]. Our implementation is achieved with an UKF. Although the resulting computational costs in UKF are slightly higher than those in an EKF, the minimal selection of the states yields fast and precise information when requested by the controller. Our method results in an error of 0.03 deg. for both roll and pitch, outperforming previous referenced estimation techniques.

II. Gyroscope Model and Roll Pitch Angles

A. Gyroscope Model

As part of the Inertial Measurement Unit (IMU), a three-axis gyro provides measurements of the angular velocity expressed in the IMU coordinate frame. We use a simple model to relate the gyro measurement $\tilde{\boldsymbol{\omega}}$ to the real angular rate $\boldsymbol{\omega}$, both expressed in the IMU body coordinate frame:

$$\tilde{\boldsymbol{\omega}} = \boldsymbol{\omega} + \mathbf{b}_{\omega} + \boldsymbol{\eta}_{\omega} \tag{1}$$

where the tilde is used to denote measured quantities. The measurement noise $\boldsymbol{\eta}_{\omega}$ is modeled as additive white Gaussian noise and the term $\mathbf{b}_{\omega}$ represents the non-static bias of the gyro, considered as a Brownian process, i.e. its derivative with respect to time is modeled as white noise:

$$\dot{\mathbf{b}}_{\omega} = \boldsymbol{\eta}_{b\omega} \tag{2}$$

Both white noise processes are assumed to be of zero-mean,

$$\mathrm{E}[\boldsymbol{\eta}_{\omega}] = \mathrm{E}[\boldsymbol{\eta}_{b\omega}] = \mathbf{0} \tag{3}$$

The angular velocity and the corresponding biases covariance matrices are $\mathbf{Q}_{\omega}$, $\mathbf{Q}_{b\omega}$. The covariance parameters can be derived by examining the measured IMU Allan plots [10].

---

* This research has been financed by the European Union (European Social Fund-ESF) and Greek funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) - Research Program: ARISTEIA: Reinforcement of the interdisciplinary and/or inter-institutional research and innovation.

The authors are with the Department of Mechanical Engineering, National Technical University, 15780 Athens, Greece. (e-mail: sotirisno@gmail.com; egpapado@central.ntua.gr; ph: +30-210-772-1440).

B. Roll and Pitch Estimation

The attitude of the quadruped can be described by means of the quaternion parameterization defined as:

$$\mathbf{q} = \begin{bmatrix} \hat{\mathbf{j}}\sin(\rho/2) \\ \cos(\rho/2) \end{bmatrix} = \begin{bmatrix} [\ q_1 \quad q_2 \quad q_3\ ]^T \\ q_4 \end{bmatrix} \tag{4}$$

where $\hat{\mathbf{j}}$ is a unit vector corresponding to the rotation axis and ρ is the angle of rotation. A rotation quaternion must always satisfy the following constraint:

$$\mathbf{q}^T\mathbf{q} = q_1^2 + q_2^2 + q_3^2 + q_4^2 = 1 \tag{5}$$

In addition, the inverse quaternion is:

$$\mathbf{q}^{-1} = [\ -q_1 \quad -q_2 \quad -q_3 \quad q_4\ ] \tag{6}$$

Due to the complexity of the problem, a few assumptions are made. Firstly, the ground is considered even, which is realistic since high-speed gaits are can be achieved on even terrain. The slope of the ground can be nonzero; in such a case, it is assumed that this slope is known via some other system, such as vision. Additionally, the bounding gait is assumed to include a full-stance phase.

We now focus our attention to the quadruped shown in Fig. 1. We denote by **I** and **B** the inertial and body frame respectively, while the subscripts $(\cdot)_{br}$ and $(\cdot)_{fl}$ refer to back right leg and front left leg respectively. Moreover, a pre-superscript refers to the frame in which a vector is expressed.

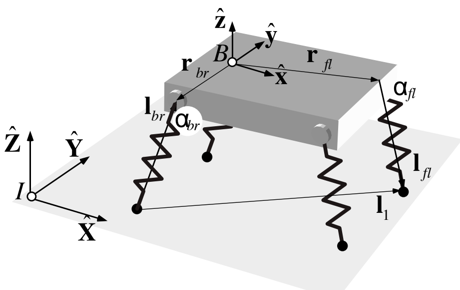

Fig. 1. 3D model of the quadruped at double stance during a bounding gait.

Assuming that a leg pair such as the (br, fl) is in contact with the ground, then vectors $\mathbf{l}_{br}$ and $\mathbf{l}_{fl}$ can be expressed as functions of the encoder angles α_{br} and α_{fl} as follows:

$${}^{B}\mathbf{l}_{br} = \mathbf{Rot}_y(\alpha_{br})\, l_{br}\, \hat{\mathbf{z}} \tag{7}$$

$${}^{B}\mathbf{l}_{fl} = \mathbf{Rot}_y(\alpha_{fl})\, l_{fl}\, (-\hat{\mathbf{z}}) \tag{8}$$

where l_i is the length of the foot i, which is provided by the leg spring encoders. The term $\mathbf{Rot}_y(\alpha)$ is a rotation matrix corresponding to a rotation of angle α around a y-axis:

$$\mathbf{Rot}_y(\alpha) = \begin{bmatrix} \cos\alpha & 0 & -\sin\alpha \\ 0 & 1 & 0 \\ \sin\alpha & 0 & \cos\alpha \end{bmatrix} \tag{9}$$

Since the vectors $\mathbf{r}_{br}$ and $\mathbf{r}_{fl}$, which express the position of the i-th hip of the quadruped in the body frame, are constant in the body frame, the computation of the vector ${}^{B}\mathbf{l}_1$ is trivial:

$${}^{B}\mathbf{l}_1 = {}^{B}\mathbf{l}_{br} + {}^{B}\mathbf{l}_{fl} + {}^{B}\mathbf{r}_{fl} - {}^{B}\mathbf{r}_{br} \tag{10}$$

Additionally, the vector $\mathbf{l}_1$ in the inertial frame is given by:

$${}^{I}\mathbf{l}_1 = {}^{I}\mathbf{C}_B(\psi,\theta,\varphi)\cdot {}^{B}\mathbf{l}_1 \tag{11}$$

where ${}^{I}\mathbf{C}_B$ represents the rotation matrix from **B** to **I**, parameterized by the yaw, ψ, pitch, θ, and roll, φ, angles.

In the case of ideal bounding, the yaw angle is zero while due to the assumption of even terrain, the third component of ${}^{I}\mathbf{l}_1$ lies in the X-Y plane. The assumption of zero yaw is also realistic on a treadmill, where we can constrain the rotation of the platform with respect to the gravity vector. The mathematical expression of the above observations is:

$${}^{I}\mathbf{C}_B(0,\theta,\varphi)\, {}^{B}\mathbf{l}_1 \cdot \hat{Z} = 0 \tag{12}$$

Considering the full stance phase, (12) holds for any set of two feet. Denoting ${}^{B}\mathbf{l}_2$ the distance between the back right leg and front right leg, pitch and roll can be computed as:

$$\varphi = \tan^{-1}\left(\frac{{}^{B}l_{2,z}\cdot{}^{B}l_{1,x} - {}^{B}l_{2,x}\cdot{}^{B}l_{1,z}}{{}^{B}l_{2,x}\cdot{}^{B}l_{1,y} - {}^{B}l_{2,y}\cdot{}^{B}l_{1,x}}\right) \tag{13}$$

$$\theta = \tan^{-1}\left(\frac{\sin(\varphi)\cdot{}^{B}l_{1,y} + \cos(\varphi)\cdot{}^{B}l_{1,z}}{{}^{B}l_{1,x}}\right) \tag{14}$$

Finally, given the roll and pitch, we compute the Direction Cosine Matrix (DCM) and the corresponding quaternion. The quaternion that expresses the rotation from the inertial to the body frame is a function, namely **m**, of roll and pitch:

$$\mathbf{q}_{BI,m} = \mathbf{m}(\theta,\varphi) \tag{15}$$

Next, an UKF is developed to improve the IMU feedback.

III. Unscented Kalman Filter

We use the following notation: a subscript k is added to discretized quantities, while a hat represents estimated quantities. Two kind of errors are used, $\delta[\cdot]$ and $d[\cdot]$. The former represents a rotation error in the Euclidean space, while the latter represents a rotation error in the quaternion manifold. Furthermore, the superscripts $(\cdot)^-$ and $(\cdot)^+$ refer to *a priori* and *a posteriori* estimates.

A. State Definition

The state vector **x** consists of the quaternion describing the rotation from the **I** to the **B** frame $\mathbf{q}_{BI,}$ and the gyro biases $\mathbf{b}_\omega$:

$$\mathbf{x} = \begin{bmatrix} \mathbf{q}_{BI}^T & \mathbf{b}_\omega^T \end{bmatrix}^T \tag{16}$$

The error of the states, which is tracked by the covariance matrix $\mathbf{P}_{xx}$, is denoted by $\delta\mathbf{x}$.

Since the quaternion represents three degrees of freedom, the corresponding covariance matrix also has to be represented as a three dimensional matrix. The constraint of the unit quaternion influences the positive definiteness of the state covariance. To this end, we employ the quaternion error $d\mathbf{q}$ corresponding to a small rotation between the estimated quaternion $\hat{\mathbf{q}}_{BI}$ and the true quaternion $\mathbf{q}_{BI}$:

$$\mathbf{q}_{BI} = d\mathbf{q} \otimes \hat{\mathbf{q}}_{BI} \tag{17}$$

where $\otimes$ stands for the quaternion multiplication. In contrast to the additive property of the gyro biases' error, the quaternion error is multiplicative, and is derived from (17) as:

$$d\mathbf{q} = \mathbf{q}_{BI} \otimes \hat{\mathbf{q}}_{BI}^{-1} \tag{18}$$

Given $d\mathbf{q}$, we can extract the rotation error $\delta\boldsymbol{\zeta}$. The error $\delta\boldsymbol{\zeta}$ is considered as the vector part of $d\mathbf{q}$.

Having all state errors expressed in the Euclidean space, the derivation of the state covariance $\mathbf{P}_{xx}$, where we make use of the covariance operator Cov(*), is trivial:

$$\mathbf{P}_{xx} = \mathrm{Cov}(\delta\mathbf{x}) \tag{19}$$

where,

$$\delta\mathbf{x} = \begin{bmatrix} \delta\boldsymbol{\zeta}^T & \delta\mathbf{b}_\omega^T \end{bmatrix}^T \tag{20}$$

In the above equation the term $\delta\mathbf{b}_\omega$ denotes the error related to the bias states. On the other hand given $\delta\boldsymbol{\zeta}$, the corresponding quaternion error $d\mathbf{q}$ can be computed using the unit norm constraint. As there will be both positive and negative values of the scalar part of the quaternion that satisfy the quaternion normalization constraint, we choose the positive value. As a consequence the quaternion error is:

$$d\mathbf{q} = [\ \delta\boldsymbol{\zeta}^T \quad \sqrt{1-\delta\boldsymbol{\zeta}^T\delta\boldsymbol{\zeta}}\ ]^T \tag{21}$$

For the subsequent discretization, we use the function $\boldsymbol{\varepsilon}(\cdot)$ which maps an arbitrary three dimensional rotation vector $\mathbf{v}$ to its corresponding quaternion:

$$\boldsymbol{\varepsilon}: \mathbf{v} \to \boldsymbol{\varepsilon}(\mathbf{v}) = \begin{bmatrix} \mathbf{v}/\|\mathbf{v}\|\sin\left(\|\mathbf{v}\|/2\right) \\ \cos\left(\|\mathbf{v}\|/2\right) \end{bmatrix} \tag{22}$$

where $\|\cdot\|$ stands for the Euclidean norm.

B. Prediction Model

The propagation of state $\mathbf{x}$ is achieved by the following equations:

$$\dot{\mathbf{q}}_{BI} = \frac{1}{2}\begin{bmatrix} {}^B\tilde{\boldsymbol{\omega}} - \mathbf{b}_\omega - \boldsymbol{\eta}_\omega \\ 0 \end{bmatrix} \otimes \mathbf{q}_{BI} \tag{23}$$

$$\dot{\mathbf{b}}_\omega = \boldsymbol{\eta}_{b\omega} \tag{24}$$

To discretize the above stochastic differential equations, we integrate the deterministic part separately. Assuming a zero-order hold for the gyroscope measurements, and denoting the time between two subsequent measurements as Δt, we obtain:

$$\hat{\mathbf{q}}_{k+1}^- = \boldsymbol{\varepsilon}\left(\Delta t \cdot \left(\tilde{\boldsymbol{\omega}}_k - \hat{\mathbf{b}}_{\omega,k}^+\right)\right) \otimes \hat{\mathbf{q}}_k^+ \tag{25}$$

$$\hat{\mathbf{b}}_{\omega,k+1}^- = \hat{\mathbf{b}}_{\omega,k}^+ \tag{26}$$

Furthermore, the discretized process noise covariance matrix is:

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}_\omega \cdot \Delta t & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{Q}_{b\omega} \cdot \Delta t \end{bmatrix} \tag{27}$$

In (27), the diagonal terms represent the discretized quaternion $\mathbf{q}_{BI}$ and bias $\mathbf{b}_\omega$ covariance matrices respectively.

C. Measurement Model

The measurement equation is:

$$\hat{\mathbf{q}}_{BI,m} = \mathbf{m}\left(\theta, \varphi, \boldsymbol{\eta}_m\right) \tag{28}$$

where the term $\boldsymbol{\eta}_m$ expresses the standard measurement noise of the incremental encoders, assumed to be white Gaussian noise, as well as model imperfections and numerical errors. The measurement covariance matrix, which will be used in the UKF, is $\mathbf{R}$ and is the main tuning parameter of the filter.

D. Unscented Kalman Filter Equations

The UKF addresses the linear approximation issues of the EKF. The concept is based on the observation that it is easier to approximate a probability distribution than a nonlinear function. The state distribution is represented by a Gaussian Random Variable (GRV), but is now specified using a minimal set of carefully selected perturbations about the current state estimate, namely sigma points. These sigma points completely capture the true mean and covariance of the GRV, and after being propagated through the nonlinear system, they capture the posterior mean and covariance accurately to the second order (Taylor series expansion). The main tool of the selection of the sigma points is the Unscented Transformation (UT). For a more in depth study of the UT as well as the UKF, please refer to [10].

From a mathematical point of view, to implement a UKF, our system has to be in the form of discrete time equations:

$$\mathbf{x}_{k+1} = \mathbf{f}\left(\mathbf{x}_k, \boldsymbol{\eta}_{x,k}\right) \tag{29}$$

$$\mathbf{y}_{k+1} = \mathbf{h}\left(\mathbf{x}_k, \boldsymbol{\eta}_{y,k}\right) \tag{30}$$

where $\mathbf{x}_k \in \mathbb{R}^{n\times1}$, $\mathbf{y}_k \in \mathbb{R}^{m\times1}$ are the n-dimensional state and m-dimensional measurement vectors respectively, while $\mathbf{f}$ and $\mathbf{h}$ correspond to nonlinear functions which represent the process model and the measurement model. In addition, the terms $\boldsymbol{\eta}_{x,k}$ and $\boldsymbol{\eta}_{y,k}$ are related to the process noise and measurement noise. Here, the process model is given by (25)-(26), whereas the measurement model is described by (28).

Since the quaternion lies on the SO(3) manifold, the quaternion mean cannot be computed as a weighted sum of sample points. As a consequence, the UKF implementation in quaternion space needs special care. Although the UKF algorithm is known, there is need for its adaptation in the unit quaternion manifold. This is presented next, based on [11].

Given the $n\times n$ covariance matrix $\mathbf{P}_{xx,k}$ and the process covariance matrix $\mathbf{Q}_k$ at the timestamp k, a set of (2n+1) sigma points, namely $\mathbf{X}_k \in \mathbb{R}^{n\times(2n+1)}$, is calculated:

$$\mathbf{X}_k = \begin{bmatrix} \mathbf{X}_{0,k} & \mathbf{X}_{1,k} & \cdots & \mathbf{X}_{2n,k} & \mathbf{X}_{2n+1,k} \end{bmatrix} \tag{31}$$

where the columns of $\mathbf{X}_k$ are calculated as:

$$\mathbf{X}_{i,k} = \hat{\mathbf{x}}_k \qquad i = 0 \tag{32}$$

$$\mathbf{X}_{i,k} = \hat{\mathbf{x}}_k + \mathbf{u}_i \qquad i = 1,\ldots,n \tag{33}$$

$$\mathbf{X}_{n+i,k} = \hat{\mathbf{x}}_k - \mathbf{u}_i \qquad i = 1,\ldots,n \tag{34}$$

The term $\mathbf{u}_i$ stands for the i^{th} column of the matrix $\mathbf{U}$; it is computed as the square root of the sum of the matrices $\mathbf{P}_{xx,k}$ and $\mathbf{Q}_k$ scaled by γ, and derived by employing a lower Cholesky decomposition:

$$\mathbf{U} = \sqrt{\gamma\left(\mathbf{P}_{xx,k} + \mathbf{Q}_k\right)} \tag{35}$$

The factor γ is given by:

$$\gamma = \sqrt{n+\lambda} \tag{36}$$

$$\lambda = w^2\left(n+\kappa\right) - n \tag{37}$$

In (36) and (37), w determines the sigma points spread around the state $\hat{\mathbf{x}}_k$ mean, and is usually set to a small positive value. The constant κ is set between 3-n and zero, to improve the capture of distribution higher order moments.

While the selection of sigma points is trivial, when the states are expressed in Euclidean space, using (32)-(34), the quaternion states needs special care since addition is not closed in the SO(3) manifold. Based on this idea, we distinguish between sigma points related with quaternion states, denoted by $\mathbf{X}_k^q$, and sigma points referring to bias states, namely $\mathbf{X}_k^b$. Furthermore, we denote $\mathbf{u}_i$, at the timestamp k, in (33)-(34) as:

$$\mathbf{u}_i = \left[\boldsymbol{\xi}_{i,k}^{q\ T}, \boldsymbol{\xi}_{i,k}^{b\ T} \right]^T \tag{38}$$

where $\boldsymbol{\xi}_{i,k}^q$ are the first three elements of the ith column of matrix $\mathbf{U}$ which represent the vector part of the quaternion and thus they are a three dimensional error. The $\boldsymbol{\xi}_{i,k}^b$ are the last three elements of the ith column of $\mathbf{U}$ related to the error in the gyro bias $\mathbf{b}_\omega$.

Observing (33)-(34), one can see that sigma points are scattered around the current mean based on the uncertainty of the current estimate, which can be considered as a small perturbation to the current mean. To scatter the quaternion sigma points around the current quaternion estimate $\hat{\mathbf{q}}_k$, the error $\boldsymbol{\xi}_{i,k}^q$ has to be represented in quaternion space. To this end, we map $\boldsymbol{\xi}_{i,k}^q$ to the quaternion error $d\boldsymbol{\xi}_{i,k}^q$ by means of the quaternion normalization constraint:

$$d\boldsymbol{\xi}_{i,k}^q = \begin{bmatrix} \left(\boldsymbol{\xi}_{i,k}^q\right)^T & \sqrt{1-\left(\boldsymbol{\xi}_{i,k}^q\right)^T \boldsymbol{\xi}_{i,k}^q} \end{bmatrix}^T \tag{39}$$

Given the quaternion error $d\boldsymbol{\xi}_{i,k}^q$ the quaternion sigma points are:

$$\mathbf{X}_k^q = \begin{bmatrix} \mathbf{X}_{0,k}^q & \mathbf{X}_{i,k}^q & \mathbf{X}_{n+i,k}^q \end{bmatrix} \quad i = 1,...,n \tag{40}$$

where:

$$\mathbf{X}_{i,k}^q = \hat{\mathbf{q}}_k \qquad i = 0 \tag{41}$$

$$\mathbf{X}_{i,k}^q = d\boldsymbol{\xi}_{i,k}^q \otimes \hat{\mathbf{q}}_k \quad i = 1,...,n \tag{42}$$

$$\mathbf{X}_{n+i,k}^q = \left(d\boldsymbol{\xi}_{i,k}^q\right)^{-1} \otimes \hat{\mathbf{q}}_k \quad i = 1,...,n \tag{43}$$

From a mathematical point of view, the definitions of addition and subtraction in the quaternion manifold stand out from the above equations. In other words, addition is considered as the pre-multiplication between the two quaternions, namely $d\boldsymbol{\xi}_{i,k}^q$ and $\hat{\mathbf{q}}_k$, while subtraction is considered as the pre-multiplication with the inverse quaternion $\left(d\boldsymbol{\xi}_{i,k}^q\right)^{-1}$ and $\hat{\mathbf{q}}_k$.

The rest $\mathbf{X}_k^b$ sigma points are selected based in (32)-(34):

$$\mathbf{X}_k^b = \begin{bmatrix} \mathbf{X}_{0,k}^b & \mathbf{X}_{i,k}^b & \mathbf{X}_{n+i,k}^b \end{bmatrix} \quad i = 1,...,n \tag{44}$$

where:

$$\mathbf{X}_{i,k}^b = \hat{\mathbf{b}}_{\omega,k} \qquad i = 0 \tag{45}$$

$$\mathbf{X}_{i,k}^b = \hat{\mathbf{b}}_{\omega,k} + \boldsymbol{\xi}_{i,k}^b \quad i = 1,...,n \tag{46}$$

$$\mathbf{X}_{n+i,k}^b = \hat{\mathbf{b}}_{\omega,k} - \boldsymbol{\xi}_{i,k}^b \quad i = 1,...,n \tag{47}$$

Then, the set of sigma points is:

$$\mathbf{X}_k = \left[\mathbf{X}_k^{qT}, \mathbf{X}_k^{bT} \right]^T \tag{48}$$

Additionally, the UKF weights are chosen as:

$$W_0^{(m)} = \lambda / (n + \lambda) \tag{49}$$

$$W_0^{(c)} = \lambda / (n + \lambda) + (1 - w^2 + \beta) \tag{50}$$

$$W_i^{(m)} = W_i^{(c)} = 1 / \left[2(n + \lambda) \right] \quad i = 1,...,2n \tag{51}$$

where the superscripts m and c stand for mean and covariance respectively. The term β is used to improve the estimates of higher order moments of the distribution; its optimal value is 2 for Gaussian distributions [10].

The set of sigma points in (48) is then propagated based on (25)-(26) to form the newly predicted sigma points $\mathbf{X}_{k+1}$. Furthermore, the predicted sigma points of the measurements are computed by:

$$\mathbf{Y}_{k+1} = \mathbf{H}\mathbf{X}_{k+1} \tag{52}$$

$$\mathbf{H} = \begin{bmatrix} \mathbf{I}_{4x4} & \mathbf{0}_{4x3} \end{bmatrix} \tag{53}$$

The mean of the state as well as the mean of predicted measurements at timestamp k+1, are calculated as:

$$\hat{\mathbf{x}}_{k+1}^- = \sum_{i=0}^{2n} W_i^{(m)} \mathbf{X}_{i,k+1} \tag{54}$$

$$\hat{\mathbf{y}}_{k+1} = \sum_{i=0}^{2n} W_i^{(m)} \mathbf{Y}_{i,k+1} \tag{55}$$

Although, the above equations hold in the case of the states that are expressed in the Euclidean manifold, the quaternion mean, part of the state mean, cannot be computed in this way. The computed quaternion mean has to lie in the unit sphere and as a consequence it requires a different metric. The metric used was the Euclidean norm [11]. As a result, the quaternion mean was computed as:

$$\hat{\mathbf{q}}_{k+1}^- = \sum_{i=0}^{2n} W_i^{(m)} \mathbf{X}_{i,k+1}^q \Bigg/ \left| \sum_{i=0}^{2n} W_i^{(m)} \mathbf{X}_{i,k+1}^q \right| \tag{56}$$

Given the newly predicted mean $\hat{\mathbf{x}}_{k+1}^-$, the predicted state covariance is:

$$\hat{\mathbf{P}}_{xx,k+1}^- = \sum_{i=0}^{2n} W_i^{(c)} (\mathbf{X}_{i,k+1} - \hat{\mathbf{x}}_{k+1}^-)(\mathbf{X}_{i,k+1} - \hat{\mathbf{x}}_{k+1}^-)^T \tag{57}$$

In (57), when the states are expressed in quaternion terms, to derive the quaternion error at the timestamp k+1, the subtraction in the quaternion manifold is employed as:

$$d\mathbf{q}_{i,k+1} = \mathbf{X}_{i,k+1}^q \otimes (\hat{\mathbf{q}}_{k+1}^-)^{-1} \tag{58}$$

In addition, the above quaternion error has to be represented in three dimensions; this is achieved using the error δζ we defined earlier. Given δζ, the predicted state covariance matrix, at the timestamp k+1, $\hat{\mathbf{P}}_{xx,k+1}^-$ is computed. The measurement mean $\hat{\mathbf{y}}_{k+1}$ and measurement covariance matrix $\mathbf{P}_{yy,k}$ are computed similarly as presented above:

$$\hat{\mathbf{P}}_{yy,k+1} = \sum_{i=0}^{2n} (\mathbf{Y}_{i,k+1} - \hat{\mathbf{y}}_{k+1}^-)(\mathbf{Y}_{i,k+1} - \hat{\mathbf{y}}_{k+1}^-)^T + \mathbf{R} \tag{59}$$

The cross correlation matrix $\mathbf{P}_{xy,k+1}$ between the states and the measurements is computed as:

$$\hat{\mathbf{P}}_{xy,k+1} = \sum_{i=0}^{2n} (\mathbf{X}_{i,k+1} - \hat{\mathbf{x}}_{k+1}^-)(\mathbf{Y}_{i,k+1} - \hat{\mathbf{y}}_{k+1}^-)^T \tag{60}$$

where subtraction as defined in the quaternion space, is used when necessary. Then, the Kalman Gain can be calculated as:

$$\hat{\mathbf{K}}_{k+1} = \hat{\mathbf{P}}_{xy,k+1} \hat{\mathbf{P}}_{yy,k+1}^{-1} \tag{61}$$

The innovation, which is the difference between the actual measurements and the predicted ones, is expressed as the quaternion error between $\hat{\mathbf{q}}_{m,k+1}$ and $\hat{\mathbf{y}}_{k+1}$:

$$\mathrm{d}\mathbf{q}_{k+1} = \hat{\mathbf{q}}_{m,k+1} \otimes \left(\hat{\mathbf{y}}_{k+1}\right)^{-1} \tag{62}$$

Finally, the correction step of the UKF is:

$$\hat{\mathbf{P}}^{+}_{xx,k+1} = \hat{\mathbf{P}}^{-}_{xx,k+1} - \hat{\mathbf{K}}_{k+1}\hat{\mathbf{P}}_{yy,k+1}\hat{\mathbf{K}}^{-1}_{k+1} \tag{63}$$

$$\delta\mathbf{x}_{k+1} = \hat{\mathbf{K}}_{k+1}\begin{bmatrix} \mathrm{d}q_{1,k+1} & \mathrm{d}q_{2,k+1} & \mathrm{d}q_{3,k+1} \end{bmatrix}^{T} \tag{64}$$

$$\hat{\mathbf{x}}^{+}_{k+1} = \hat{\mathbf{x}}^{-}_{k+1} + \delta\mathbf{x}_{k+1} \tag{65}$$

In (65), addition as defined in quaternion space was used in order to calculate the *a posteriori* estimate of the quaternion $\hat{\mathbf{q}}^{+}_{k+1}$ as well as (21) when the correction vector has to be mapped from the Euclidean Space to the SO(3) manifold.

IV. Simulation Results

Experiments were performed in the Webots^{TM} simulation environment, see Fig. 2. The quadruped has four legs, each with an actuated rotational hip joint and a compliant passive prismatic joint. The IMU is positioned at the CoM of the quadruped robot. The implemented controller can set both the *robot speed* and its *apex height*, despite the single per leg actuated hip joint [12]. The robot is running in a bounding gait, and achieves an average speed of 1 m/s along the x-axis.

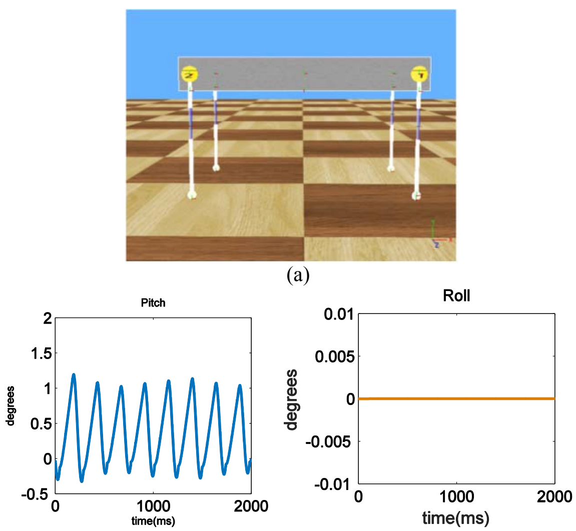

Fig. 2. (a) The quadruped robot in the Webots^{TM} simulation environment, (b) body pitch angle response, and (c) body roll angle response.

To evaluate the performance of the UKF, the noise levels of the gyroscope measurements as well as of the leg encoders must be realistic. In the Webots^{TM} environment, this was achieved by including actual sensor measurement noise, obtained from static experiments. The robot has variable pitch angle; the yaw and roll angles were commanded by the controller to remain zero. During the simulation, the robot was executing a bounding gait for approximately one minute. The terrain was set to be even. The sensor sampling frequency was set at 1ms, and the w, κ and β filter parameters were chosen as 0.03, -2.0 and 2.0 respectively. The time response of the robot pitch, roll, and yaw angles is shown in Fig. 2 for reference. The pitch angle error which is of high significance for control, oscillates between 0.08° and -0.08°, see Fig. 3. Some peaks around 0.1° are related to numerical errors in the Webots^{TM} environment. The absolute of the mean of the error in the pitch angle is approximately 0.03°. In Fig. 4, the error in roll angle is plotted; the corresponding error is similar to that for the pitch angle.

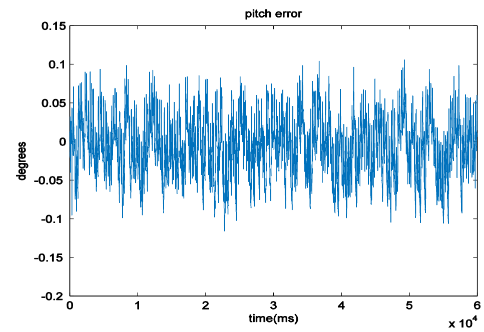

Fig. 3. Pitch Error.

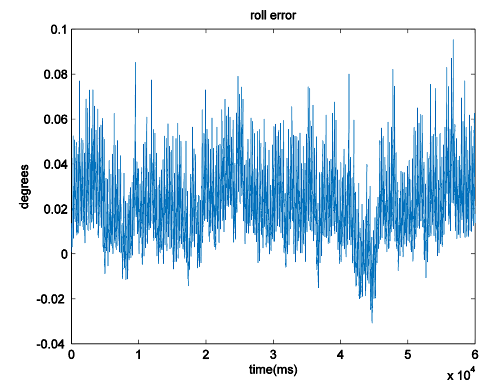

Fig. 4. Roll Error.

To compare our estimation results with results in the literature, we chose the estimation technique proposed in [4]. While we could choose to fuse the information of the leg kinematics within a UKF, we implemented the EKF due to the computational effort a computer needed to handle a 21-state UKF. Figure 5 depicts the pitch errors for the developed estimation methodology and the referenced one, whereas in Fig. 6 the corresponding roll errors are plotted.

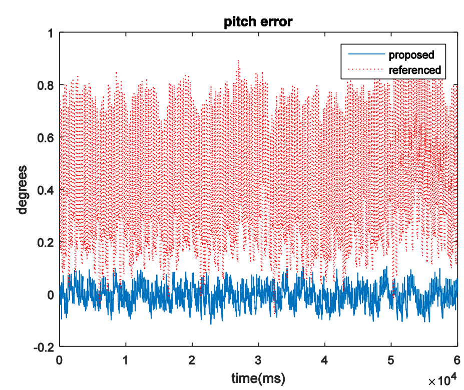

Fig. 5. Pitch error for different estimation techniques.

Comparing the two responses, we can observe that the mean of the pitch error in the developed estimator is close to zero, (0.029°), whereas in the referenced one, the mean is approximately 0.33°, see Fig. 5. Regarding the roll angle error, despite the fact that the means of the errors are close, namely 0.02° and 0.09°, the difference in the magnitude of the oscillations, as illustrated in Fig. 6, is significant.

The pitch and roll errors corresponding to the developed UKF estimator and the simplest sensor fusion filter, i.e. the complementary filter are given in Figs. 7 and 8. The resulting pitch and roll angle estimation error means, considering the complementary filter, are 0.16^{o} and 0.079^{o} respectively; i.e. three times larger than the error obtained from the UKF.

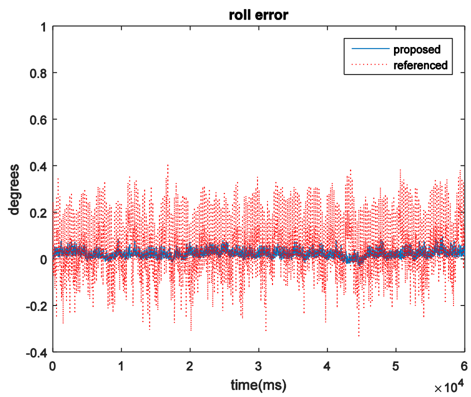

Fig. 6. Roll error for different estimation techniques.

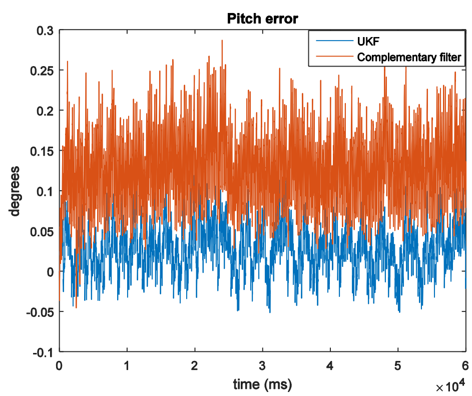

Fig. 7. Pitch errors for the developed UKF and the complementary filter.

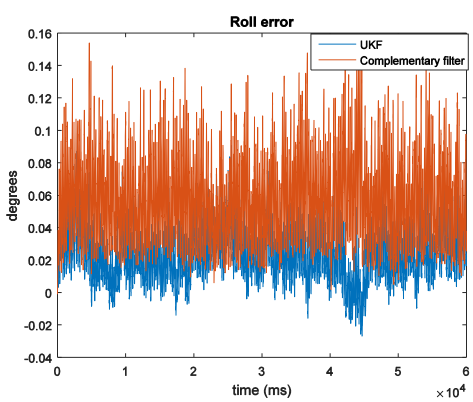

Fig. 8. Roll errors for the developed UKF and the complementary filter.

To conclude, the developed estimator yields angle errors which are *ten times smaller* compared to the referenced technique. The main reason for these superior results is that in our measurement model, the actual states are being observed, whereas in the referenced estimator the measurement model consists of a combination of all states; as a consequence, an error even in one state affects all its states. From an implementational point of view, since the referenced estimator consists of more states, it is prone to more numerical errors than the proposed one.

V. Conclusion and Future Work

In this paper, a novel approach for the estimation of roll and pitch angles of a bounding quadruped was proposed. Assuming even terrain and an ideal bounding gait, the roll and pitch angles were computed, to be fused with on-board IMU measurements in an UKF. The simulation results were very encouraging as the error, even for durations up to a minute, was not drifting but instead was staying below 0.07^{o}. The main bottleneck of the UKF is the computation of the square root of the covariance matrix, which has complexity $O(n^3)$. This was overcome via a minimal state selection.

In the near future, full experimental results will be obtained after our quadruped robot under development is operational. Although, the simulation environment can model many realistic situations, we did not simulate slippage. We anticipate that our estimator will handle slippage, since no assumption of zero velocity at the foot points is made. Thus we aim at developing algorithms for slippage detection. Finally, the use of sensors such as the Kinect™ will be considered in order to improve position relative estimates.

VI. References

[1] G. P. Roston and E. P. Krotkov, “Dead Reckoning Navigation For Walking Robots,” *IEEE/RSJ Int. Conference on Intelligent Robots and Systems,* July 1992, pp. 607-612.

[2] P.-C. Lin, H. Komsuoglu and D. E. Koditschek, “A Leg Configuration Measurement System for Full-Body Pose Estimates in a Hexapod Robot,” *IEEE Trans. Robotics,* vol. 21, no. 3, June 2005, pp. 411-422.

[3] P.-C. Lin, H. Komsuoglu and D. E. Koditschek, “Sensor data fusion for body state estimation in a hexapod robot with dynamical gaits,” *IEEE Trans. Robotics,* vol. 22, no. 5, October 2006, pp. 932 - 943.

[4] M. Bloesch, M. Hutter, M. A. Hoepflinger, S. Leutenegger, C. Gehring, D. C. Remy and R. Siegwart, “State Estimation for Legged Robots - Consistent Fusion of Leg Kinematics and IMU,” *Proceedings of Robotics: Science and Systems,* July 2012.

[5] M. Bloesch, M. Hutter, M. A. Hoepflinger, S. Leutenegger, C. Gehring, D. C. Remy and R. Siegwart, “State Estimation for Legged Robots on Unstable and Slippery Terrain,” *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems (IROS),* 3-7 Nov. 2013, pp. 6058 - 6064.

[6] M. Reinstein and M. Hoffmann, “Dead reckoning in a dynamic quadruped robot: inertial navigation system aided by a legged odometer,” *IEEE Int. Conference on Robotics and Automation (ICRA)*, 9-13 May 2011, pp. 617 - 624.

[7] A. Chilian and H. Hirschmuller, “Multisensor Data Fusion for Robust Pose Estimation of a Six-Legged Walking Robot,” *IEEE/RSJ Int. Conference on Intelligent Robots and Systems (IROS)*, 25-30 September 2011, pp. 2497 - 2504.

[8] S. P. Singh and K. J. Waldron, “A Hybrid Motion Model for Aiding State Estimation in Dynamic Quadrupedal Locomotion,” *IEEE Int. Conf. on Robotics and Automation,* 10-14 April 2007, pp. 4337-4342.

[9] G. Kurz, I. Gilitschenski, S. Julier and U. D. Hanebeck, Recursive Estimation of Orientation Based on the Bingham Distribution, *16th Int. Conference on Information Fusion (FUSION)*, 2013, pp. 1487-1494.

[10] N. El-Sheimy, H. Hou and X. Niu, “Analysis and Modeling of Inertial Sensors Using Allan Variance,” *IEEE Trans. Instrumentation and Measurement,* vol. 57, no. 1, January 2008., pp. 140-149.

[11] S. J. Julier and J. K. Uhlmann, “Unscented Filtering and Nonlinear Estimation,” *Proc. IEEE,* vol. 92, no. 3, pp. 401-422, March 2004.

[12] Y.-J. Cheon and J.-H. Kim, “Unscented Filtering in a Unit Quaternion Space for Spacecraft Attitude Estimation,” *IEEE Int. Symposium on Industrial Electronics*, 4-7 June 2007, pp. 66 - 71.

[13] N. Cherouvim and E. Papadopoulos, “Speed and Height Control for a Special Class of Running Quadruped Robots,” *IEEE Int. Conference on Robotics and Automation (ICRA),* 19-23 May 2008, pp. 825-830.