An Embedded Underground Navigation System

Khonzi Hlophe
Novel Mining Methods Research Group
CSIR: Centre for Mining Innovations
Johannesburg, South Africa
Email: hlophek@gmail.com

***Abstract*—Platform pose (localization and orientation) information is a key requirement for autonomous mobile systems. The severe natural conditions and complex terrain of underground mines diminish the capability of most pose estimation systems, especially GPS. Our research interest is focused on using a low-cost off-the-shelf inertial measurement unit (IMU) to improve the Active Beacon Positioning System (ABPS) developed here at the CSIR. This paper proposes a novel pose estimator, for underground mines, that fuses together data from the ABPS and a low-cost MEMS based inertial navigation system (INS). This pose estimator uses an unscented Kalman filter (UKF) to fuse the data together. The method is evaluated by building a complete system in a lab.**

I. Introduction

South Africa plays a major role in the international mining fraternity. Mining employs 495 000 workers directly and a similar amount indirectly, providing a daily subsistence for approximately 5 million South Africans [1]. Robotics could aid the mining industry to achieve government safety standards. Some of the main challenges were discussed by Green and Vogt in [2].

Navigation systems that integrate measurements from Inertial Navigation Systems (INS) and the Global Positioning Systems (GPS) are widely used in outdoor navigation systems [3]. INS provides accurate navigation data (position, velocity and attitude) over very short time intervals. INS based on low-cost inertial measurement units (IMUs) suffers from accuracy degradation over time, due to the combined effect of its sensor errors causing drift problems. GPS provides absolute data which is used to restrict the error of the INS.

Navigation in GPS deprived environments requires new absolute localization systems to be developed. This paper provides an introduction to a statistical positioning algorithm of an ultrasonic time-of-flight (TOF) based system. It further discusses the fusion of this system with an INS using an unscented Kalman filter (UKF).

II. Previous Work

A vital operation performed in all Kalman type filters is the propagation of a Gaussian random variable (GRV) through the system dynamics. In the extended Kalman filter (EKF), the system state distribution and all relevant noise densities are approximated by GRV. These GRV are propagated analytically through a first-order linearization, or sometimes second-order estimation, of the nonlinear system. The estimation can introduce large errors in the true posterior mean and covariance of the transformed GRV, which may lead to sub-optimal performance and occasionally divergence of the filter.

Van der Merwe and Wan [4] presented a dynamic process model for a loosely coupled GPS/INS integrated system which includes *time varying bias terms*. They begin by pointing out the inherent shortcomings in using the EKF for data fusion and present, as an alternative, a family of improved derivativeless nonlinear Kalman filters called the *sigma-point Kalman filters* (SPKF). They demonstrate the improved navigation state (position, velocity and attitude) estimation performance of the SPKF by applying it to navigate a small-scale helicopter. Performance metrics indicate an approximate 30% error reduction in both attitude and position estimates relative to the baseline EKF implementation.

Zhang et al [5] focused on using some low-cost off-the-shelf sensors, such as an IMU and inexpensive single GPS receiver, for autonomous navigation. They present an autonomous vehicle navigation method by fusing data measurements from an IMU, a GPS, and a digital compass. Two key steps were adopted to overcome the low precision of the sensors. The first was to establish sophisticated dynamics models which consider Earth self rotation, measurement bias, and system noise. The second was to use a sigma-point Kalman filter for the system state estimation, which has higher accuracy compared with the extended Kalman filter, as shown in [4]. The method was evaluated by experimenting on a land vehicle equipped with IMU, GPS, and digital compass.

The contribution of this paper includes the use of the active beacon positioning system (ABPS), designed and build within the CSIR, and fusion of this system with an inertial measurement unit. The UKF is used to fuse the data from the ABPS and the IMU.

The next section discusses the square-root unscented Kalman filter algorithm. Section IV describes the ABPS. It further describes the process of statistical estimation of position, given the distances between receivers and beacons and the positions of the relative beacons. Section V describes the state vector, process model and measurement model. The final two section discuss the results and provides the conclusion.

III. The Square-Root UKF

In this document we used a sigma-point Kalman filter called the square-root UKF as the main data fusion algorithm [6]. Sigma-point filters pass a set of points representing the input distribution through the non-linear functions, and then

approximate the output statistics. The square-root UKF was first introduced by van der Merwe and Wan [4] and uses a Cholesky factor updating instead of the actual covariance matrix.

Algorithm 1 Square-Root UKF [4]

Initialization:

$$\hat{\mathbf{x}}_0 = \mathbf{E}[\mathbf{x}_0] \tag{1}$$

$$\mathbf{S}_0 = chol\left\{\mathbf{E}[(\mathbf{x}_0 - \hat{\mathbf{x}_0})(\mathbf{x}_0 - \hat{\mathbf{x}_0})^T]\right\} \tag{2}$$

The Cholesky factorization decomposes a symmetric, positive-definite matrix into the product of a lower-triangular matrix and its transpose.

For $k = 1, \cdots, \infty$**:**

1) Calculate sigma-points: The Cholesky triangular matrix is used directly to calculate the sigma points as follows:

$$\chi_{k-1} = \begin{bmatrix}\hat{\mathbf{x}}_{k-1} & \mathbf{x}_{k-1} + \eta\mathbf{S}_{k-1} & \mathbf{x}_{k-1} - \eta\mathbf{S}_{k-1}\end{bmatrix}. \tag{3}$$

2) Time-update equations:

$$\chi_{k|k-1} = \mathbf{f}(\chi_{k-1}) \tag{4}$$

$$\hat{\mathbf{x}}_k^- = \sum_{i=0}^{2L} w_i^m \chi_{i,k|k-1} \tag{5}$$

$$\mathbf{S}_{x_k}^- = \mathrm{qr}\left\{[\sqrt{w_1^c}(\chi_{1:2L,k|k-1} - \hat{\mathbf{x}}_k^-)]\right\} \tag{6}$$

$$\mathbf{S}_{x_k}^- = \mathrm{cholupdate}\cdots \left\{\mathbf{S}_{x_k}^-, \chi_{0,k|k-1} - \hat{\mathbf{x}}_k^-, w_0^{(c)}\right\} \tag{7}$$

$$\mathcal{Y}_{k|k-1} = \mathbf{h}(\chi_{i,k|k-1}) \tag{8}$$

$$\hat{\mathbf{y}} = \sum_{i=0}^{2L} \beth_i^m \mathbf{y}_{i,k|k-1} \tag{9}$$

3) Measurement-update:

$$\mathbf{S}_{\hat{y}_k} = \mathrm{qr}\left\{[\sqrt{w_1^c}(\mathcal{Y}_{1:2L,k|k-1} - \hat{\mathbf{y}}_k^-)]\right\} \tag{10}$$

$$\mathbf{S}_{\hat{y}_k}^- = \mathrm{cholupdate}\cdots \left\{\mathbf{S}_{\hat{y}_k}^-, \mathcal{Y}_{0,k|k-1} - \hat{\mathbf{y}}_k^-, w_0^{(c)}\right\} \tag{11}$$

$$\mathbf{P}_{x_k,y_k} = \sum_{i=0}^{2L} (\chi_{i,k|k-1} - \hat{\mathbf{x}_k^-})(\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{y}_k^-})^T \tag{12}$$

$$\mathbf{K}_k = (\mathbf{P}_{x_k,y_k}/\mathbf{S}_{\hat{y}_k}^T)/\mathbf{S}_{\hat{y}_k} \tag{13}$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k(\hat{\mathbf{x}}_k - \hat{\mathbf{x}}_k^-) \tag{14}$$

$$\mathbf{U} = \mathbf{K}_k\mathbf{S}_{\hat{y}_k} \tag{15}$$

$$\mathbf{S}_{x_k} = \mathrm{cholupdate}\left\{\mathbf{S}_{x_k}^-, \mathbf{U}, -1\right\} \tag{16}$$

Algorithm 1 provides the generic square-root UKF. It is important to note the following:

- *Weights & parameters:* $\eta = \sqrt{L+\lambda}$, $w_0^m = \lambda/(L+\lambda)$, $w_0^c = w_0^m + (1-\alpha^2+\beta)$, $w_i^c = w_i^m = 1/[2(L+\lambda)]$ for $i = 1, \cdots, 2L$. $\lambda = \alpha^2(L+\kappa) - L$ is a compound scaling parameter, L is the dimension of the augmented state-vector, $0 < \alpha \leq 1$ is the primary scaling factor determining the extent of the spread of the sigma-points around the prior mean. β is a secondary scaling factor used to emphasize the weighting on the 0^{th} sigma-point for the posterior covariance calculation. β can be used to minimize certain higher-order error terms based on known moments of the prior random variable. For Gaussian priors, $\beta = 2$ is optimal. κ is a tertiary scaling factor and is usually set to zero. For more detail on the selection procedure can be seen in [6].
- *Linear-algebra operators:*
  - $\sqrt{\cdot}$:matrix square-root using lower triangular Cholesky decomposition.
  - $\mathrm{qr}(\mathbf{A})$: lower-triangular part of $\mathbf{R}$ matrix resulting from economy qr decomposition of data-matrix $\mathbf{A}$.
  - $\mathrm{cholupdate}\{\mathbf{R}, \mathbf{U}, \pm\nu\}$: N consecutive rank-1 Cholesky up(down)dates of the lower-triangular Cholesky factor $\mathbf{R}$ by the N columns of $\sqrt{\nu}\mathbf{U}$.
  - $/$: Efficient least-squares pseudo inverse implemented using triangular QR decomposition with pivoting.

The SR-UKF is used to fuse data from two sensors, an inertial navigation system (INS) and a time-of-flight based beacon localization system. The described application is for a loosely coupled configuration of the navigation systems. The INS is developed using a Analog Devices ADIS16364 inertial measurement unit. The data from the IMU is transformed to the inertial frame position, velocity and attitude using the process described in Figure 1. The equations are described in Section V-A. The ABPS was developed at the CSIR in South Africa. The ABPS is described in the next section.

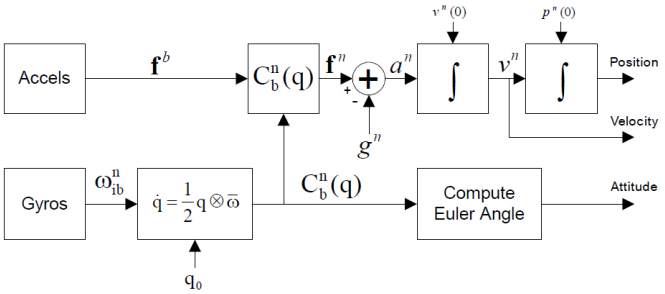

Fig. 1: The strapdown INS in the inertial reference frame

IV. Active Beacon Position System

The localization system described is a 2D positioning system and mainly finds its application within excavations of the order of $30m \times 3m \times 1m$ in tabular ore bodies. The vertical height in the application environment is limited to $1m$ and no significant value is added by determining this coordinate.

The system consists of a number of beacons with known locations and a number of receivers that need to localize themselves within a shared reference grid. An receiver can only localize itself within coverage of a sufficient number of beacons, in this case at least three. The system is required to

resolve coordinates with $10cm$ accuracy for useful application in geo-stamping measurement data for sensors designed at the CSIR, such as the autonomous sounding device defined in [7], and navigation of a mobile platform.

Localization is achieved by implementing a trilateration algorithm with an ordinary least squares (OLS) estimator [8]. With this approach, the receiver only requires the distance between itself and the beacons, and know the associated beacons locations in order to localize itself. The OLS estimator is used (as opposed to analytical methods), because only approximate distances can be measured as a result of noise inherent in the system.

Receivers determine the distance to a given beacon by measuring the time-of-flight (TOF) of an ultrasonic signal emitted by a beacon. The TOF is related to the distance between the beacon and the receiver by the speed at which the wave travels. In order to measure TOF, the receiver has to know when the signal was transmitted. To accomplish this, a beacon periodically emits the ultrasonic signal, together with an electromagnetic signal at the same instant, so that the TOF upon reception by the receiver is given by the difference between the arrival times of the two signals. The technique assumes the TOF of the electromagnetic signal to be zero, allowing the electromagnetic signal to act as a synchronization mechanism. Over time, an receiver acquires distance measurements from different beacons covering it and localizes itself after the sufficient number of distances has been measured.

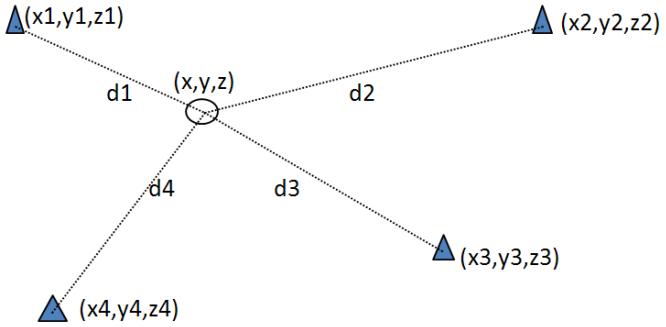

Fig. 2: Example of a localization network

A. Building a linear model

Consider a system of four beacons and a single receiver, as shown in Figure 2. Let n denote the total number of measurements taken at all beacons combined. Let $\boldsymbol{\delta} = (x, y, z)$ denote the spatial coordinates of the target point. Let $\boldsymbol{B}_i = (x_i, y_i, z_i)$ be the exact location of the beacon at which the i^{th} measurement is taken. Also let r_i be the measured distance from the i^{th} beacon to the target point and $r_i = d_i(\delta) + \epsilon_i$. Define the regression equations as

$$\begin{aligned} d_i(\delta) &= E(r_i|x, y, z) \\ &= \sqrt{(x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2} \end{aligned} \tag{17}$$

as the true distance from the i^{th} beacon to the target point which are non-linear in the unknowns x, y, z.

Let (x_r, y_r, z_r) be the coordinates of any point in real space $\mathbf{R}^3$, which we will refer to as the reference point. We can rewrite Eq. 17, to include this reference point, as

$$\begin{aligned} d_i(\delta)^2 = (x_i - x_r + x_r - x)^2 + (y_i - y_r + y_r - y)^2 \\ + (z_i - z_r + z_r - z)^2 \end{aligned} \tag{18}$$

Let the true distance between the reference point and the location of the beacon at which the i^{th} measurement was taken be

$$d_{ir} = \sqrt{(x_i - x_r)^2 + (y_i - y_r)^2 + (z_i - z_r)^2} \tag{19}$$

and also let

$$d_r(\delta) = \sqrt{(x - x_r)^2 + (y - y_r)^2 + (z - z_r)^2} \tag{20}$$

be the true distance between the reference point and the target point (x, y, z). Expanding and regrouping terms in Eq. 18, using the cosine rule [8],

$$d_i(\delta)^2 = (x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2 \tag{21}$$

$$\begin{aligned} &= (x_i - x_r + x_r - x)^2 + (y_i - y_r + y_r - y)^2 \\ &+ (z_i - z_r + z_r - z)^2 \end{aligned} \tag{22}$$

$$\begin{aligned} &= (x - x_r)^2 + 2(x_i - x_r)(x - x_r) + (x_i - x_r)^2 \\ &+ (y - y_r)^2 + 2(y_i - y_r)(y - y_r) + (y_i - y_r)^2 \\ &+ (z - z_r)^2 + 2(z_i - z_r)(z - z_r) + (z_i - z_r)^2 \end{aligned} \tag{23}$$

A closer look at the final double product terms above gives

$$\begin{aligned} &2((x_i - x_r)(x - x_r) + (y_i - y_r)(y - y_r) + (z_i - z_r)(z - z_r)) \\ &= (x - x_r)^2 + (y - y_r)^2 + (z - z_r)^2 \\ &+ (x_i - x_r)^2 + (y_i - y_r)^2 + (z_i - z_r)^2 \\ &= d_r(\delta)^2 + d_{ir}^2 - d_i(\delta)^2 \end{aligned} \tag{24}$$

where $i = 1, 2, \cdots, n$ and $n \geq 4$.

If one of the beacons (say B_1) is chosen as a reference point, the exact distance can be replaced by a measured distance. This special case yields,

$$\begin{aligned} &(x_2 - x_1)(x - x_1) + (y_2 - y_1)(y - y_1) + (z_2 - z_1)(z - z_1) \\ &\approx \frac{1}{2}[r_1^2 - r_2^2 + r_{21}^2] = b_{21} \\ &(x_3 - x_1)(x - x_1) + (y_3 - y_1)(y - y_1) + (z_3 - z_1)(z - z_1) \\ &\approx \frac{1}{2}[r_1^2 - r_3^2 + r_{31}^2] = b_{31} \\ &\vdots \end{aligned} \tag{25}$$

$$\begin{aligned} &(x_n - x_1)(x - x_1) + (y_n - y_1)(y - y_1) + (z_n - z_1)(z - z_1) \\ &\approx \frac{1}{2}[r_1^2 - r_n^2 + r_{n1}^2] = b_{n1} \end{aligned} \tag{26}$$

$$\tag{27}$$

where b_{i1} is the measured distance. This is a linear system of $(n - 1)$ equations with 3 unknowns.

B. The ordinary least-squares estimator

The linear system of equations given by Eq. 27 can be written in matrix form as $\boldsymbol{A}\boldsymbol{x} \approx \boldsymbol{b}$, with

$$\boldsymbol{A} = \begin{bmatrix} 2(x_1 - x_1) & 2(y_1 - y_1) & 2(z_1 - z_1) \\ 2(x_2 - x_1) & 2(y_2 - y_1) & 2(z_2 - z_1) \\ \vdots & \vdots & \vdots \\ 2(x_n - x_1) & 2(y_n - y_1) & 2(z_n - z_1) \end{bmatrix} \tag{28}$$

consisting of the beacon positions from beacon 1 to n and the parameter vector

$$\boldsymbol{x} = \begin{bmatrix} x - x_1 \\ y - y_1 \\ z - z_1 \end{bmatrix} \tag{29}$$

and $\boldsymbol{b}$ as

$$\boldsymbol{b} = \begin{bmatrix} b_{21} \\ b_{31} \\ \vdots \\ b_{n1} \end{bmatrix} \tag{30}$$

the matrix of measured distances.

Minimizing the sum of the squares of the residuals can be written as

$$S = (\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x})^T(\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}) \tag{31}$$

This requires the solution of the normal equation $\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{x} = \boldsymbol{A}^T b$. The solution method depends on the condition number $\boldsymbol{A}^T\boldsymbol{A}$. If $\boldsymbol{A}^T\boldsymbol{A}$ is non-singular and well-conditioned then the linear/ordinary least squares (OLS) is given by

$$\boldsymbol{x} = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T b \tag{32}$$

If $\boldsymbol{A}^T\boldsymbol{A}$ is nearly-singular (which means that it is poorly conditioned) then:

- It is necessary to compute $\boldsymbol{A} = \boldsymbol{Q}\boldsymbol{R}$. Where
  - $\boldsymbol{Q}$ is the orthonormal matrix, and
  - $\boldsymbol{R}$ is the upper-triangular matrix.
- It is then possible to solve for $\boldsymbol{R}\boldsymbol{x} = \boldsymbol{Q}^T \overrightarrow{b}$ by back substitution when $\boldsymbol{A}$ is full rank.

V. SR-UKF based ABPS/INS intergration

An IMU driven kinematic process model formulation, that comprises of an INS mechanization component [9] and an IMU error model component, will be used. The IMU sensor error model components are added to the state vector because low-cost IMU have large bias and scale factor errors which can lead the filter to diverge. The estimated values of these errors are used to correct the raw IMU acceleration and gyro-rate measurements before they are used inside the INS mechanization of the process model [6], which is sufficient to model the combined effect of the bias and scale error terms. The 16 dimensional state vector, x, of our system is defined as follows:

$$\mathbf{x} = \begin{bmatrix} \mathbf{p} & \mathbf{v} & \mathbf{q} & \mathbf{b}_a & \mathbf{b}_\omega \end{bmatrix}, \tag{33}$$

where $\mathbf{p} = [x, y, z]^T$, $\mathbf{v} = [v_n, v_e, v_d]^T$, and $\mathbf{q} = [q_0, q_1, q_2, q_3]^T$ represents the position, velocity, and attitude in quaternion of the navigation vehicle in the navigation frame (n-frame). $\mathbf{b}_a$ is the IMU acceleration biases, and $\mathbf{b}_\omega$ is the IMU gyro rate biases. van der Merwe [6] states that a time-varying bias term is sufficient to model the combined effect of the bias and scale error terms and therefore eliminates the need to include the scale factor in the state vector.

A. Process Model

The continuous time kinematic navigation equations, INS mechanization equations and error model discussed above, oparating on this state vecor and driven by the error corrected IMU measurements can be established as follows:

$$\begin{bmatrix} \dot{\mathbf{p}} \\ \dot{\mathbf{v}} \\ \dot{\mathbf{q}} \\ \dot{\mathbf{b}}_a \\ \dot{\mathbf{b}}_\omega \end{bmatrix} = \begin{pmatrix} \mathbf{v} \\ \mathbf{C}_b^n(\bar{\mathbf{a}} - \mathbf{a}_{\bar{\mathbf{r}}_{imu}}) + \mathbf{g}^n \\ -\frac{1}{2}\tilde{\boldsymbol{\Omega}}_{\bar{\omega}}\mathbf{q} \\ \mathbf{W}_{b_a} \\ \mathbf{W}_{b_\omega} \end{pmatrix}. \tag{34}$$

In Equation 34, the $\mathbf{C}_b^n$ is the direction cosine matrix (DCM) transforming vectors from the b-frame to the n-frame. The DCM is a nonlinear function of the current attitude quaternion and is given by

$$\begin{aligned} \mathbf{C}_b^n &= (\mathbf{C}_n^b)^T \\ &= 2\begin{pmatrix} 0.5 - q_2^2 - q_3^2 & q_1q_2 - q_0q_3 & q_1q_3 + q_0q_2 \\ q_1q_2 + q_0q_3 & 0.5 - q_1^2 - q_3^2 & q_2q_3 - q_0q_1 \\ q_1q_3 - q_0q_2 & q_2q_3 + q_0q_1 & 0.5 - q_1^2 - q_2^2 \end{pmatrix}. \end{aligned} \tag{35}$$

The $\mathbf{g}^n$ is the gravity vector in the n-frame, which is expressed by

$$\mathbf{g}^n = \begin{pmatrix} 0 \\ 0 \\ g \end{pmatrix}, \tag{36}$$

where g is the local gravity, which is decided by the co-ordinates in the geodetic coordinate system [10]. The IMU readings in equation 34 is defined as

$$\bar{\mathbf{a}} = \tilde{\mathbf{a}} - \mathbf{b}_a - \mathbf{n}_a, \tag{37}$$

$$\bar{\omega} = \tilde{\omega} - \mathbf{b}_\omega - \mathbf{C}_n^b\omega_c - \mathbf{n}_\omega. \tag{38}$$

In the above equations, $\bar{\mathbf{a}}$ and $\bar{\omega}$ are the raw measurements of acceleration and gyro-rates coming from the IMU, and $\mathbf{n}_a$ and $\mathbf{n}_\omega$ are the IMU acceleration and gyro-rate measurement noise, and ω_c is the rotation rate of the earth as measured in the navigation frame (Coriolis effect) relative to the earth frame and hence is time-varying [6] as the navigation frame moves relative to the earth frame. For terrestrial navigation, we assume the navigation frame is stationary relative to the earth frame resulting in a constant ω_c for a given origin location, latitude and longitude, of the navigation frame.

$\tilde{\mathbf{\Omega}}_{\bar{\omega}}$ is a 4×4 skew-symmetric matrix composed of the error corrected IMU gyro-rate measurements, i.e.,

$$\tilde{\mathbf{\Omega}}_{\bar{\omega}} = \begin{bmatrix} 0 & \bar{\omega}_p & \bar{\omega}_q & \bar{\omega}_r \\ -\bar{\omega}_p & 0 & -\bar{\omega}_r & \bar{\omega}_q \\ -\bar{\omega}_q & \bar{\omega}_r & 0 & -\bar{\omega}_p \\ -\bar{\omega}_p & -\bar{\omega}_q & \bar{\omega}_p & 0 \end{bmatrix} \tag{39}$$

In Equation 34, $\mathbf{a}_{\bar{\mathbf{r}}_{imu}}$ is the IMU-lever-arm coupling component due to the IMU not being located at the center of gravity of the vehicle. This component can be ignored if the navigation filter computes the state estimate at the IMU location [6]. This IMU centric navigation solution can then simply be transformed to the center of gravity location after the fact as needed by the control system.

One of the most common properties of IMU, despite their quality, is that the acceleration and gyro-rate output are known to be in error by an unknown slowly time-varying bias [5]. The turn-on bias can be modelled as a zero-mean, stationary, first-order Gauss-Markov process [11].

Since the bias and scale factor of low cost MEMS based IMU sensors exhibit non-zero mean and non-stationary behaviour, the residual time-varying bias error is modelled as a *random-walk* process [10] in order to improve the tracking of these time-varying errors by the navigation filter. This does, however, require that the effect of these errors be *observable* through the specific choice of measurement model. Therefore, $\mathbf{W}_{b_a}$, and $\mathbf{W}_{b_\omega}$ in Equation 34 are white noise of acceleration and gyro rate respectively in the IMU.

Equation 34 has to be linearized for implementation in the embedded system. The position and velocity discrete-time updates are calculated by the following simple first-order Euler updates [6]

$$\mathbf{p}_{k+1} = \mathbf{p}_k + \dot{\mathbf{p}}_k \cdot dt \tag{40}$$

$$\mathbf{v}_{k+1} = \mathbf{v}_k + \dot{\mathbf{v}}_k \cdot dt, \tag{41}$$

where $\dot{\mathbf{p}}_k$ and $\dot{\mathbf{v}}_k$ are calculated using Equation 34 and dt is the integration time-step of the system, usually dictated by the IMU data rate. The quaternion propagation equation can be discretized with an analytical calculation of the exponent of the skew-symmetric matrix given by Horn [12]. The discrete-time update can be written as

$$\mathbf{q}_{k+1} = exp(-\frac{1}{2}\tilde{\mathbf{\Omega}} \cdot dt)\mathbf{q}_k. \tag{42}$$

If we further denote

$$\Delta\phi = \bar{\omega_p} \cdot dt \tag{43}$$

$$\Delta\theta = \bar{\omega_q} \cdot dt \tag{44}$$

$$\Delta\psi = \bar{\omega_r} \cdot dt, \tag{45}$$

as the effective rotations around the body frame or roll, pitch and yaw axes undergone by the vehicle during the time period dt, assuming that the gyro-rates $\bar{\omega_p}$, $\bar{\omega_q}$ and $\bar{\omega_r}$ remained constant during that interval, we can re-introduce the 4×4 skew-symmetric matrix

$$\mathbf{\Phi}_\Delta = \tilde{\mathbf{\Omega}} \cdot dt \tag{46}$$

$$= \begin{bmatrix} 0 & \Delta\phi & \Delta\theta & \Delta\psi \\ -\Delta\phi & 0 & -\Delta\psi & \Delta\theta \\ -\Delta\theta & \Delta\psi & 0 & -\Delta\phi \\ -\Delta\psi & -\Delta\theta & \Delta\phi & 0 \end{bmatrix}. \tag{47}$$

Using the definition of the matrix exponent and the skew symmetric property of $\mathbf{\Phi}_\Delta$, we can write down the following closed-form solution:

$$exp(-\frac{1}{2}\mathbf{\Phi}_\Delta) = \mathbf{I}\cos(s) - \frac{1}{2}\mathbf{\Phi}_\Delta\frac{\sin(s)}{s}, \tag{48}$$

where

$$s = \frac{1}{2}\left\|\begin{bmatrix}\Delta\phi & \Delta\theta & \Delta\psi\end{bmatrix}\right\| = \frac{1}{2}\sqrt{(\Delta\phi)^2 + (\Delta\theta)^2 + (\Delta\psi)^2}. \tag{49}$$

Proof of this closed-form can be found in [6]. Theoretically, Equations 42 and 48 ensure that the updated quaternion $\mathbf{q}_{k+1}$ has a unit norm. It is common to add a small Lagrange multiplier term to the first component of Equation 48 to further maintain numerical stability and the unity norm of the resulting quaternion [6]. The resulting final solution for the time-update of the quaternion vector is given by

$$\mathbf{q}_{k+1} = [\mathbf{I}(\cos(s) + \eta \cdot dt \cdot \lambda) - \frac{1}{2}\mathbf{\Phi}_\Delta\frac{\sin(s)}{s}]\mathbf{q}_k, \tag{50}$$

where $\lambda = 1 - \|\mathbf{q}_k\|^2$ is the deviation of the square of the quaternion norm from unity due to numerical integration errors, and η is the factor that determines the convergence speed of the numerical error. These factors serve the role of the above mentioned Lagrange multiplier that ensures that the norm of the quaternion remains close to unity [13]. The constraint on the speed of convergence for the stability of the numerical solution is $\eta \cdot dt < 1$ [14]

Finally, the discrete time random-walk process for the IMU sensor error terms are given by

$$\mathbf{b}_{a_{k+1}} = \mathbf{b}_{a_k} + dt \cdot \mathbf{W}_{b_a} \tag{51}$$

$$\mathbf{b}_{\omega_{k+1}} = \mathbf{b}_{\omega_k} + dt \cdot \mathbf{W}_{\omega_a}, \tag{52}$$

where $\mathbf{W}_{b_a}$ and $\mathbf{W}_{\omega_a}$ are zero-mean Gaussian random variables.

B. Measurement Model

The position obtained from ABPS is considered as the measurement set in a Kalman filter. This can be expressed in the following formulation.

$$\mathbf{z}_k = \mathbf{p}_k + \mathbf{C}_b^n\mathbf{r}_{ABPS} + \nu_{ABPS} \tag{53}$$

where $\mathbf{r}_{ABPS}$ is the vector from the origin of the b-frame to the ABPS mounting position. ν_{ABPS} is the ABPS measurement noise, which is white with normal probability distribution $p(\nu) \approx N(0, R)$. The measurement noise matrix $R = diag(\sigma_x^2, \sigma_y^2, \sigma_z^2)$ can be calculated by statistics from a set of data which is obtained from the ABPS in a fixed point in its working environment.

VI. Experimental Results

A single-board computer was used to collect data from the IMU and the ABPS, and processing the data. The configuration is shown in Figure 3. The filter was implemented in MATLAB. Sensor data was streamed from the single-board computer to a PC over a RS232 serial cable. In the current configuration the data is read from the COM port and saved in a text file. A number of visualizations were developed to aid in the debugging process.

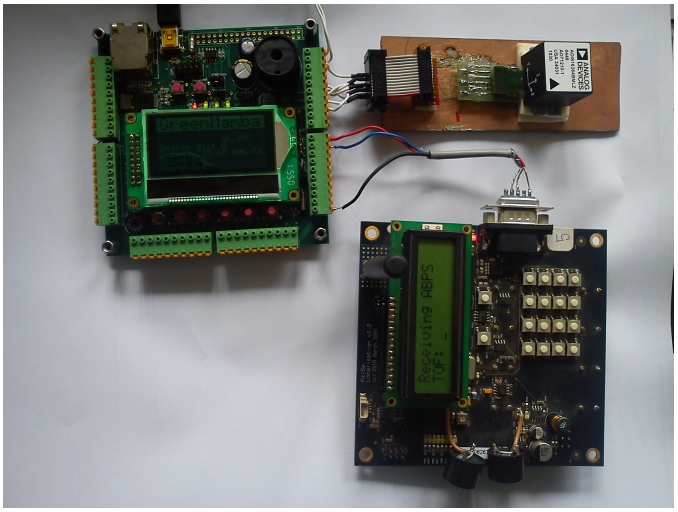

Fig. 3: Experimental Setup with IMU, ABPS and a single board computer (Green Mamba from CSIR)

The MATLAB implementation was tested by comparing simulated data to the result of $\hat{\mathbf{y}}_k^-$ of the measurement prediction step of the SR-UKF. The position results shown in Figure 4. The estimate closely resemble the measured data. Further work still has to be done in filter initialization to have a more accurate initial estimation.

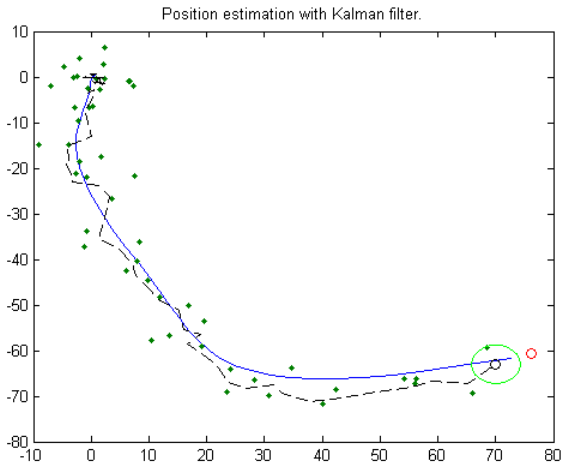

Fig. 4: Position Estimation in Matlab

VII. Conclusion and Recommendations

We present a navigation method for GPS deprived environments by integrating the measurements of IMU and ABPS. The measurement of an IMU is based on the inertial frame (i-frame) while the measurements of the ABPS is based on the n-frame. So the standard IMU driven kinematic model used for the system include the associated transformations from the i-frame to the n-frame. The system state estimation is implemented using the SR-UKF for its computational advantage against the UKF.

This system was evaluated in a lab at the CSIR. It was shown that the estimated state can generate measurement predictions that closely match the position data. The next phase of the research is to implement the SR-UKF on the single-board computer. It has not been evaluated if the single board computer can handle the processing speed necessary to complete a SR-UKF cycle which is less than $10ms$.

Acknowledgment

The authors would like to thank the Centre for Mining Innovations (CMI) of the Council for Scientific and Industrial Research (CSIR) in South Africa for funding this work.

References

[1] J. J. Green, P. Bosscha, L. Candy, K. Hlophe, S. Coetzee, and S. Brink, "Can a robot improve mine safety?" in *International conference of CAD/CAM, Robotics & Factories of the Future*, vol. 25, July 2010.

[2] J. J. Green and D. Vogt, "Robot miner for low grade narrow tabular ore bodies: the potential and the challenge," in *3rd Robotics and Mechatronics Symposium (ROBMECH 2009)*, November 2009.

[3] B. Barshan and H. F. Durrant-Whyte, "An inertial navigation system for mobile robot," in *Proceedings of the 1993 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE/RSJ, July 1993, pp. 2243–2248.

[4] R. van der Merwe and E. Wan, "Efficient derivative-free kalman filters for online learning." in *Proceedings of the 9th European Symposium on Artificial Neural Networks (ESANN) (Bruges, Belgium)*. IEEE, April 2001, pp. 205–210.

[5] P. Zhang, J. Gu, E. E. Milios, and P. Huynh, "Navigation with imu/gps/digital compass wth unscented kalman filter," in *International Conference on Mechatronics & Automation*, IEEE. IEEE press, July 2005 2005, pp. 1497–1502.

[6] R. van der Merwe, "Sigma-point kalman filters for probabilistic inference in dynamic state-space models," Ph.D. dissertation, OGI School of Science & Engineering at Oregon Health & Science University, April 2004.

[7] R. Teleka, J. Green, and S. Brink, "The automation of the 'making safe' process in south african hard-rock underground mines," in *CARs & FOF 2011*, 2011.

[8] W. Navidi, W. S. M. Jr, and W. Hereman, "Statistical methods in surveying by trilateration," in *Computational Statistics & Data Analysis*, vol. 27, 1998, pp. 209–227.

[9] P. G. Savage, "Strapdown inertial navigation intergration algorithm design part 1: Attitude algorithms," in *Journal of Guidance, Control, and Dynamics*, January 1998, ch. 21(1), pp. 19–28.

[10] J. A. Farrell and M. Barth, *The global positioning system & inertial navigation*. McGraw-Hill, 1998.

[11] S. Nassar, K. Schwarz, and N. El-Sheimy, "Ins and ins/gps accuracy improvement using autoregressive (ar) modeling of ins sensor errors," in *In Proceedings of ION-NTM*, San Diego, Jan 2004, pp. 936–944.

[12] G. M. Horn, "Attitude estimation for a low-cost uav," Thesis, University of California, Santa Cruz, March 2009.

[13] J. M. Rolfe and K. J. Staples, *Flight Simulation*. Cambridge University Press, 1986.

[14] V. Gavrilets, "Autonomous aerobatic maneuvering of miniature helicopters: Modelling and control," Ph.D. dissertation, Massachusetts Institute of Technology, 2003.