A Comparative Study of UWB-based True-Range Positioning Algorithms using Experimental Data

Cung Lian Sang, Michael Adams, Marc Hesse, Timm Hörmann, Timo Korthals, Ulrich Rückert
Cognitronics and Sensor Systems Group
CITEC, Bielefeld Univeristy, Bielefeld, Germany
{csang, madams, mhesse, thoerman, tkorthals, rueckert}@techfak.uni-bielefeld.de

***Abstract*—In this paper, we analyze five true-range positioning algorithms for UWB-based localization systems. The evaluated algorithms are: (i) trilateration using a geometric method, (ii) a closed-form multilateration solution using least squares, (iii) an iterative approach using first-order Taylor series, a recursive solution based on (iv) the *Extended Kalman Filter* (EKF), and (v) the *Unscented Kalman Filter* (UKF). In contrast to the existing comparative studies in literature, which are solely based on simulation results, our analysis is based on experimental evaluations. The evaluated algorithms are strictly chosen for a scenario, where a true-range multilateration method is applicable. True-range means the accuracy of the measured ranges is not influenced by the clock drift errors. The performance comparison of the five algorithms is examined and discussed in the paper.**

***Index Terms*—comparative study, comparative review, true-range, positioning algorithm, performance comparison, UWB, trilateration, multilateration, Taylor series, EKF, UKF, Kalman**

I. Introduction

Ultra-Wideband (UWB) has been regarded as a promising technology for precise wireless localization systems, especially in global navigation satellite system (GNSS)-denied environments [1]–[3]. Example applications include logistics, medical services, a self-localized robot, players' statistics in indoor sports, search and rescue missions and others [1]. In general, the location determination in wireless communications is typically composed of two phases: (i) ranging phase, during which the distance between two transceivers is determined, and (ii) positioning phase, during which the location of the unknown device is calculated using a positioning algorithm. In this paper, we focus on the positioning phase. The presented positioning algorithms are examined for a scenario, where multiple true-range measurements are available in the ranging phase. True-range means that the accuracy of the measured ranges is not influenced by clock drift errors [2], i.e. in contrast to pseudo-range. There are several methods to achieve a true-range measured distance in literature [4], [5]. We use the alternative double-sided two-way ranging method (AltDS-TWR) for true-range distance measurement in this paper, which has been approved in [4], [5] as the most reliable true-range method among different available schemes in literature at several tested conditions.

The positioning algorithm is a backbone of all localization systems in any environments because it is the main element in the system to compute the unknown position of the mobile node (hereinafter referred to as a tag) based on the known references (hereinafter referred to as anchors). There have been multiple studies, especially for UWB-based tracking and navigation systems, on the performance comparison of different positioning algorithms in literature [3], [6]–[10]. However, the majority of the studies are solely based on computer simulations. In fact, the simulation results are important. However, they often do not reflect realistic conditions. For instance, a misleading conclusion can occur because of the impractical parameter choice, motion model, and other factors in the simulation environments. Besides, an unfair judgment between two totally different methods is frequently performed in the simulation (e.g. trilateration vs. EKF or particle filter), which is usually not matched with real-world system implementations [7]. In practice, geometric and closed-form methods are typically coupled with a filter or smoother [2].

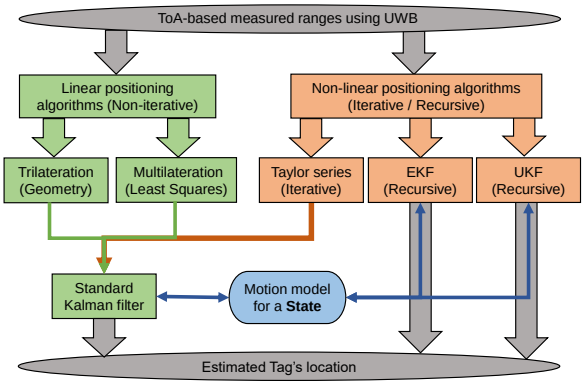

Fig. 1. Overview process of the evaluated true-range positioning algorithms. The methods are categorized as linear (light green) vs. non-linear (light orange). The motion model corresponds to the Kalman-based filters (section IV).

In this paper, the evaluated positioning algorithms are designed to reflect the practical real-world system implementation. Besides, the performance comparison is conducted upon the final outcomes of each method. Fig. 1 shows the overview process of the analyzed positioning algorithms. In general, UWB-based positioning algorithms have been divided into categories when a comparative analysis was examined in literature such as parametric vs. non-parametric [7], iterative vs. optimization-based [9], [10], and geometric-based vs. Bayesian-based approaches [6]. For simplicity, we categorized the evaluated positioning algorithms under the linear vs. non-linear approach in this paper (Fig. 1).

The final published version of the paper is available on IEEE Xplore under the DOI: 10.1109/WPNC47567.2019.8970249

II. Related Work

In literature, a general overview of UWB-based indoor positioning technologies can be found in [3]. Mathematical methods for indoor positioning algorithms were addressed in [6]. However, the algorithms are expressed in a generic way intending to be helpful for new researchers in the field as a general guideline. In [7], a comparative analysis of EKF, particle filter, trilateration, and least squares approach were evaluated in a simulation. The results showed that the particle filter has the best performance while the EKF performs worst (especially when the target speed is increased) in most of the conducted scenarios. The latter case is somewhat misleading. Moreover, the parameters used in the simulation (e.g. $1\,\mathrm{s}$ position update rate, and $15\,\mathrm{m}$ coverage range) are outdated compared to the current state-of-the-art UWB systems.

In [8], different UWB-based positioning algorithms for both *Time-of-Arrival* (ToA)- and *Time-Difference-of-Arrival* (TDoA)-based approaches were evaluated. The authors performed five positioning techniques (analytical method, least squares, *Taylor Series* (TS), maximum likelihood, and genetic algorithm) for each ToA and TDoA approaches. However, the simulated environments are specifically designed only for a *Line-of-Sight* (LOS) scenario. The simulation results showed that the TS method has been dominantly preferred in most of the evaluated conditions with minimal average error and failure rate. In practice, this is not always true in LOS scenario as our experimental results reveal in section V.

On the contrary, we performed the comparative analysis of the positioning algorithms based on the experimental evaluations in this paper. This, in turn, closely reflects to the realistic system performance of real-world scenarios.

III. True-range Positioning Algorithms in UWB

This section reviews the five true-range positioning algorithms expressed in Section I (Fig. 1) and additionally, reviews the standard *Kalman Filter* (KF) in section III-D.

A. Trilateration Algorithm using Geometric Technique

The fundamental and straightforward positioning algorithm in UWB is trilateration, which is a conventional surveying method based on geometric technique as illustrated in Fig. 2 (a). The term trilateration is usually used interchangeably with true-range multilateration in literature [11]. In this paper, trilateration specifically refers to the positioning algorithm that uses three circles to compute the location of an unknown mobile node (tag) using an analytic geometry in 2D.

By assuming the three known anchors be $A1$, $A2$ and $A3$, and their corresponding Cartesian locations in 3D be $A1(0,0,0)$, $A2(U,0,0)$, $A3(V_x,V_y,0)$ as depicted in Fig. 2 (a), the location of an unknown tag ($T(x_t,y_t,z_t)$) can be computed by trilateration, i.e Pythagoras theorem [11], [12]:

$$d_1^2 = x_t^2 + y_t^2 + z_t^2 \tag{1a}$$

$$d_2^2 = (U - x_t)^2 + y_t^2 + z_t^2 \tag{1b}$$

$$d_3^2 = (x_t - V_x)^2 + (y_t - V_y)^2 + z_t^2 \tag{1c}$$

where, d_i is the measured distance (radius of a circle or sphere) between the ith anchor and the tag, (x_t, y_t, z_t) is the interested unknown location of the tag.

By subtracting (1b) from (1a) yields

$$x_t = \frac{d_1^2 - d_2^2 + U^2}{2 \cdot U}$$

Extract z_t^2 from (1a) and substitute it into (1c) achieves

$$y_t = \frac{d_1^2 - d_3^2 + V_x^2 + V_y^2 - 2 \cdot x_t \cdot V_x}{2 \cdot V_y}$$

Substituting the computed x_t and y_t into (1a) gives

$$z_t = \pm\sqrt{d_1^2 - x_t^2 - y_t^2} \tag{4}$$

To determine the ambiguous solutions from (4), the information from the fourth anchor is necessary in 3D trilateration. There are three constraints in the geometry of trilateration (Fig. 2 (a)): (i) the first anchor ($A1$) should be located in the origin of a coordinate system, i.e. $(0,0,0)$ in 3D Cartesian coordinate, (ii) the second anchor should be located on the X-axis, and (iii) the height of the anchors (Z-value) should be the same for all anchors. In an arbitrary system set-up, the first constrain can be accomplished by subtracting the value of the first anchor ($A1$) from all the three available known anchors including itself. The second constrain can be accomplished by projecting the second anchor's value ($A2$) onto the X-axis.

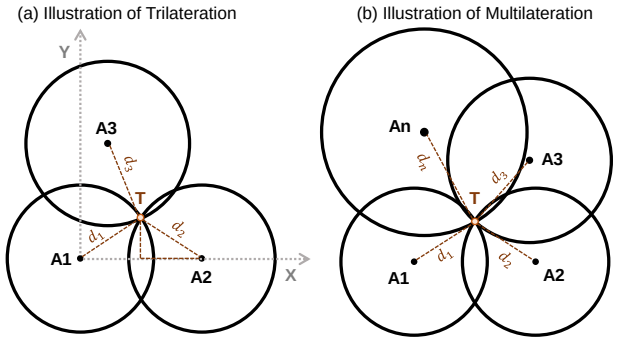

Fig. 2. Illustration of trilateration and multilateration algorithms in 2D.

B. Closed-form Multilateration using Least Squares

The generic spherical equation for true-range multilateration (Fig. 2 (b)) [2] can be represented in 3D as:

$$d_i^2 = (x_i - x_t)^2 + (y_i - y_t)^2 + (z_i - z_t)^2 \tag{5}$$

where, d_i is the distance (range or radius of a sphere) between the coordinates of the ith anchor and the tag.

By subtracting the first one (d_1 when $i = 1$) from the result of other equation in (5) and simplifying it, the set of equations can be written in matrix notation [2] as follows:

$$Ax = b \tag{6}$$

where,

$$A = \begin{bmatrix} x_2 - x_1 & y_2 - y_1 & z_2 - z_1 \\ x_3 - x_1 & y_3 - y_1 & z_3 - z_1 \\ & \cdots & \\ x_n - x_1 & y_n - y_1 & z_n - z_1 \end{bmatrix}, \quad x = \begin{bmatrix} x_t \\ y_t \\ z_t \end{bmatrix},$$

The final published version of the paper is available on IEEE Xplore under the DOI: 10.1109/WPNC47567.2019.8970249

$$b = \frac{1}{2} \cdot \begin{bmatrix} d_1^2 - d_2^2 + (x_2^2 + y_2^2 + z_2^2) - (x_1^2 + y_1^2 + z_1^2) \\ d_1^2 - d_3^2 + (x_3^2 + y_3^2 + z_3^2) - (x_1^2 + y_1^2 + z_1^2) \\ \dots \\ d_1^2 - d_n^2 + (x_n^2 + y_n^2 + z_n^2) - (x_1^2 + y_1^2 + z_1^2) \end{bmatrix}$$

The detailed implementation and derivation of the method are available in our previous work [2]. The solution for (6) can be achieved using over-determined least square method [2] as:

$$x = (A^T A)^{-1} A^T b \tag{7}$$

C. Non-linear Iterative Solution using first-order Taylor Series

From (5), the actual ith measured range between the ith anchor and tag can be defined as a function [13] (Fig. 2 (b)):

$$\begin{aligned} f_i(x, y, z) &= \sqrt{(x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2} \\ &= d_i + \varepsilon_i \qquad (i = 1, 2, ..., n) \end{aligned} \tag{8}$$

where, ε_i is the range estimation error between a tag and the ith anchor. The errors (ε) are statically distributed, and the elements of it are independent using zero-mean Gaussian random variables [13]. The error covariance matrix becomes

$$R = E[\varepsilon\varepsilon^T] = diag[\sigma^2 ... \sigma^2] \tag{9}$$

where, σ is the range estimation error.

Assuming that (x_v, y_v, z_v) is an initial guess of the true tag's location (x_t, y_t, z_t), it can be written as:

$$x_t = x_v + \delta_x, y_t = y_v + \delta_y, z_t = z_v + \delta_z \tag{10}$$

where, δ_x, δ_y and δ_z are the location errors of a tag to be determined, which is the incremental error between the guess and the true positions.

Expanding (8) into TS and keeping first order term yields:

$$f_{i,v} + a_{i,1} \cdot \delta_x + a_{i,2} \cdot \delta_y + a_{i,3} \cdot \delta_z \approx d_i + \varepsilon_i \tag{11}$$

where,

$$f_{i,v} = h_i(x_v, y_v, z_v), \quad a_{i,1} = \frac{\partial f_i}{\partial x}\Big|_{x_v, y_v, z_v} = \frac{x_v - x_i}{r_i},$$

$$a_{i,2} = \frac{\partial f_i}{\partial y}\Big|_{x_v, y_v, z_v} = \frac{y_v - y_i}{r_i}, a_{i,3} = \frac{\partial f_i}{\partial z}\Big|_{x_v, y_v, z_v} = \frac{z_v - z_i}{r_i},$$

$$r_i = \sqrt{(x_i - x_v)^2 + (y_i - y_v)^2 + (z_i - z_v)^2}$$

Equation (11) can be written in matrix notation as

$$H\delta = \Delta d + \varepsilon \tag{15}$$

where, $\Delta d = d_i - f_{i,v} = d_i - r_i$,

$$H = \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \\ & \dots & \\ a_{n,1} & a_{n,2} & a_{n,3} \end{bmatrix}, \quad \delta = \begin{bmatrix} \delta_x \\ \delta_y \\ \delta_z \end{bmatrix}, \quad \varepsilon = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \dots \\ \varepsilon_n \end{bmatrix}$$

Using the covariance of measurement error (R) in (9) as a weight, (15) can be solved using the over-determined weighted least square method [13] as:

$$\delta = (H^T R^{-1} H)^{-1} H^T R^{-1} \Delta d \tag{16}$$

By substituting the initial guess (x_v, y_v, z_v) and the computed incremental error (δ) from (16) to (10), the total value of location estimation (x_t, y_t, z_t) for the tag device can be continually refined by iterative procedure.

D. Brief Review of Standard KF

In general, KF gives a recursive solution to a linear filtering problem for estimating the state of a process governed by the difference equation and its measurement as follows [14]:

$$x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1}, w_k \sim N(0, Q) \tag{17}$$

$$z_k = Hx_k + v_k, v_k \sim N(0, R) \tag{18}$$

where, x_k is the state vector, A is the $n \times n$ state transition matrix, u_k is the input vector, B is the $n \times l$ input control matrix, z_k is the measurement vector, H is the $m \times n$ state to measurement transition matrix, w_k and v_k are the process and measurement noise respectively, and Q and R represent the process and measurement noise covariances respectively.

For the sake of brevity, we refer the theory of KF to [14]. The set of equations for time update phase in KF is:

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_{k-1} \tag{19}$$

$$P_k^- = AP_{k-1}A^T + Q \tag{20}$$

where, P_k^- and P_k are the priori and posteriori estimate of error covariance matrices.

The set of equations for measurement update phase is:

$$K_k = P_k^- H^T (HP_k^- H^T + R)^{-1} \tag{21}$$

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-) \tag{22}$$

$$P_k = (I - K_k H)P_k^- \tag{23}$$

where, K_k is the $n \times m$ matrix Kalman gain.

E. Recursive Solution using Extended Kalman Filter

EKF [14] is somewhat akin to the combination of TS (section III-C) and standard KF (section III-D). The exception is that the EKF keeps track of the state variables of a total quantities, while the incremental ones are kept track in the TS [15]. Therefore, the non-linear functions are usually linearized by keeping either the first-order or the second-order TS. The former conduct is typically known as the Jacobian approximation, and the later as Hessian.

In general, the state of a dynamic process and its measurement relationship in non-linear cases are as follows [14], [15]:

$$x_k = f(x_{k-1}, u_{k-1}, w_{k-1}) \tag{24}$$

$$z_k = h(x_k, v_k) \tag{25}$$

where, f and h are the known non-linear functions of the state and measurement model respectively.

Similar to the standard KF, the time update equations are:

$$\hat{x}_k^- = f(\hat{x_{k-1}}, u_{k-1}, 0) \tag{26}$$

$$P_k^- = A_k P_{k-1} A_k^T + W_k Q_{k-1} W_k^T \tag{27}$$

The set of measurement update equations in EKF is:

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + V_k R_k V_k^T)^{-1} \tag{28}$$

$$\hat{x_k} = \hat{x_k}^- K_k(z_k - h(\hat{x_k}^-, 0)) \tag{29}$$

$$P_k = (I - K_k H_k)P_k^- \tag{30}$$

The final published version of the paper is available on IEEE Xplore under the DOI: 10.1109/WPNC47567.2019.8970249

F. Recursive Solution using Unscented Kalman Filter

The UKF addresses the non-linear problem by using carefully chosen deterministic sample points called sigma points [16], [17]. The method is known as the unscented transformation. Similar to EKF, the distribution of the state and measurement function is also approximated by a Gaussian random variable (GRV). It has been claimed that the UKF can accurately capture the posterior mean and covariance of the GRV up to 3rd order TS expansion of any non-linearity [17]. Moreover, the computational complexity of the UKF and the EKF are in the same order [16], [17]. In general, the major advantage of the UKF over EKF is a scenario, where a non-linear function cannot be easily differentiated analytically as it is necessary to be done via Jacobian or Hessian in EKF.

For the sake of brevity, we refer the theory of UKF to [17] and its original paper [16] since the method is well established and has been widely used across multiple fields.

IV. Implementation of the Kalman-based Filters

In this paper, the standard KF is used for refining the outcome achieved from three positioning algorithms namely trilateration, multilateration and TS approaches. This means that the input (observed value) of the standard KF is the estimated location data from the mentioned methods. In contrast, the EKF and UKF act as standalone positioning algorithms.

A. Implementation of Standard KF for localization systems

For a kinematic motion model of the state in standard KF, a *Position-Velocity* (PV) [15], [18], a.k.a *Constant Velocity*, is used in our implementation. PV is applied as the motion model for all Kalman filters (standard KF, EKF and UKF) in this paper. The parameters used in the standard KF are:

$$x_k = \begin{bmatrix} x & y & z & \dot{x} & \dot{y} & \dot{z} \end{bmatrix}^T, R = diag(\begin{bmatrix} \sigma_{vx}^2 & \sigma_{vy}^2 & \sigma_{vz}^2 \end{bmatrix})$$

$$Q = diag(\begin{bmatrix} \frac{T^4}{4} & \frac{T^4}{4} & \frac{T^4}{4} & \frac{T^3}{2} & \frac{T^3}{2} & \frac{T^3}{2} \end{bmatrix}) \cdot \sigma_w^2$$

$$A = \begin{bmatrix} 1 & 0 & 0 & T & 0 & 0 \\ 0 & 1 & 0 & 0 & T & 0 \\ 0 & 0 & 1 & 0 & 0 & T \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

where x, y and z are the position coordinates of the state in 3D, $\dot{x}, \dot{y}$ and $\dot{z}$ are their corresponding velocities (derivation of the position), and T is the update rate of the system.

In our implementation, the system update rate (T) is $10\,\mathrm{Hz}$, σ_w^2 in the process noise (Q) is 0.01, and $[\sigma_{vx}^2, \sigma_{vy}^2, \sigma_{vz}^2]$ in the measurement noise (R) is $[0.015, 0.015, 0.025]$ (the value is drawn from our previous experimental measurement results).

B. Implementation of EKF and UKF for localization systems

The dynamic model of the state in navigation and tracking systems are conventionally assumed as linear [15], [18] according to Newton's second law of motion (section IV-A). Therefore, the non-linearity occurs only in the measurement function according to (8). This means the state model in EKF stays exactly the same as a standard KF.

In our implementation of EKF, the observation vector (z) for the measurement [19] update phase is:

$$z_{i,k} = [d_{1,k} \quad d_{2,k} \quad ... \quad d_{n,k}]^T \tag{31}$$

where, $d_{i,k}$ represents the measured distance between a tag and the ith anchor at the current estimation time t_k.

The $h(x_{k/\hat{k-1}})$ is a vector of measurement function from (8), i.e. the process distances between the tag and ith anchors at the estimation time t_k. This can be expressed as:

$$h(\hat{x}_{k/k-1}) = [f_1(t, A_1) \quad f_2(t, A_2) \quad ... \quad f_n(t, A_n)]^T \tag{32}$$

where, $f_i(t, A_i) = \sqrt{(x_i - x_t)^2 + (y_i - y_t)^2 + (z_i - z_t)^2}$

The Jacobian matrix (H_k) in EKF becomes:

$$H_k = \begin{bmatrix} \frac{x_t - x_1}{f_1(t, A_1)} & \frac{y_t - y_1}{f_1(t, A_1)} & \frac{z_t - z_1}{f_1(t, A_1)} & 0 & 0 & 0 \\ ... & ... & ... & ... & ... & ... \\ \frac{x_t - x_n}{f_n(t, A_n)} & \frac{y_t - y_n}{f_n(t, A_n)} & \frac{z_t - z_n}{f_n(t, A_n)} & 0 & 0 & 0 \end{bmatrix}$$

Therefore, the error covariance matrix (R_k) for the measurement function in EKF now becomes:

$$R_k = diag([\sigma_{1,k}^2 \quad \sigma_{2,k}^2 \quad ... \quad \sigma_{n,k}^2]) \tag{33}$$

Similar to EKF, the dynamic state function for UKF is again linear (section IV-A). Therefore, the unscented transform is applied only in the measurement function achieved from (8).

V. Experimental Evaluation Results

In this section, the comparative result of five true-range position algorithms for a static scenario is reported in section V-B, a LOS scenario in section V-C, and a *None-Line-of-Sight* (NLOS) scenario in section V-D.

A. Experimental Setup

The experiments were conducted in the Cognitive Interaction Tracking (CITrack) laboratory [20] and at the university's sports hall. CITrack allows the integration and evaluation of current tracking technologies in a controlled environment of $6\,\mathrm{m} \times 6\,\mathrm{m} \times 3.7\,\mathrm{m}$. The VICON's motion capturing system in the CITrack, which has an accuracy of millimeter range, supports the evaluation with a reference trajectory. In the experiment, the commercially available TREK1000 system [21] from Decawave was used as an UWB hardware for collecting the measurement data. In each experimental trial, four anchors and a tag were used to track a moving tag. The anchors were deployed in each of the four corners of the test environments (lab. and sports hall). During the experiment, the four measured ranges from each anchor were logged into a computer. Then, the logged data were applied afterward in each of the presented algorithms to compare their performance using Matlab. For evaluation, the point cloud registration method Iterative Closest Point (ICP) algorithm was applied for registering the UWB with the reference. The remaining root-mean-square error (RMSE) quantifies the performance.

The final published version of the paper is available on IEEE Xplore under the DOI: 10.1109/WPNC47567.2019.8970249

B. Performance Comparison in a Static Scenario

In a static condition, all the evaluated methods gave very accurate results with RMSE value less than $0.1\,\mathrm{cm}$ when VICON system was used as a reference (Fig. 3 and first column in Fig. 4). Moreover, the results show that both EKF and UKF had very similar performances. Their differences can be spotted only in the initial condition when the two filters were not completely converged yet (Fig. 3). This occurs in all of our evaluated conditions. Therefore, we can conclude that EKF and UKF have the same performance, at least, for the non-linear measurement function using (8). The TS method has the worst performance in a static scenario (Fig. 3 and Fig. 4). In general, a few outliers were evident in all of the evaluated algorithms at their initial conditions (Fig. 3), which is very typical in Kalman-based filters.

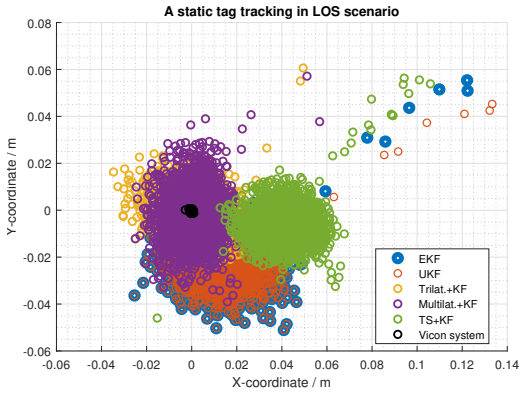

Fig. 3. Performance comparison for static measurement results in the CITrack.

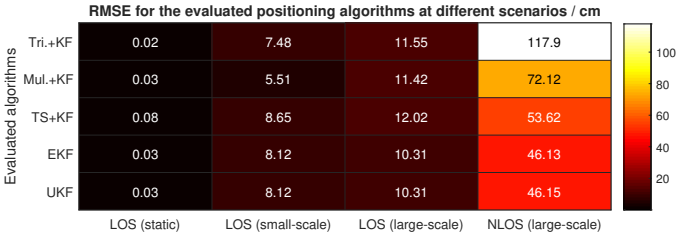

Fig. 4. RMSE for the evaluated algorithms in different scenarios

C. Performance Comparison in a pure LOS Scenario

LOS condition was conducted in two scenarios, the CITrack (Fig. 5) and the university's sports hall (Fig. 6). In a small-scale scenario at CITrack, it is evident in the result that multilateration follows the trajectory from the VICON system significantly closer than the other four methods (Fig. 5). This can be further confirmed in the RMSE provided in the second column of Fig. 4 and the best-case scenario of the five algorithms in this particular measurement illustrated in Fig. 5 (a). Moreover, the TS method gave comparable results to EKF and UKF. In this particular experiment, an unusual measurement result has occurred at roughly between $2.7\,\mathrm{m}$ and $3.7\,\mathrm{m}$ in X-axis (Fig. 5). This has an impact on all of the evaluated methods. However, the linear methods are more sensitive to this occurrence than the non-linear ones.

In a large-scale scenario, a person carried the tag on top of his head while running along on the borderline of a basketball field (Fig. 6). Overall, the evaluated five algorithms correctly

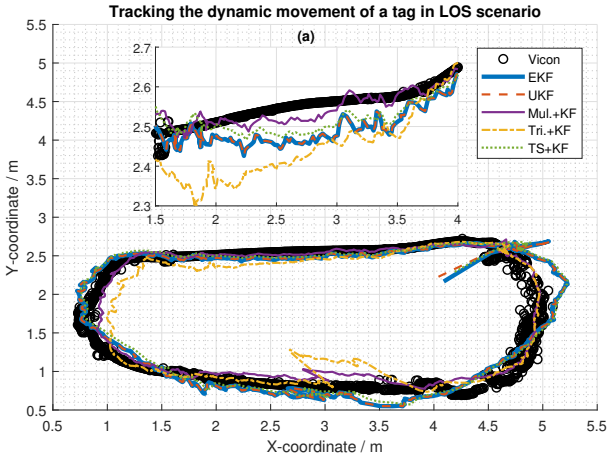

Fig. 5. Performance comparison of five positioning algorithms for tracking the movement of a tag in the CITrack.

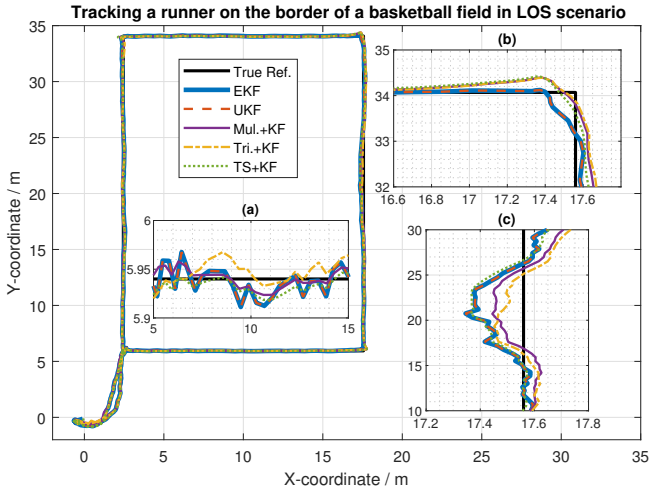

Fig. 6. Performance comparison of positioning algorithms for tracking a moving tag in a large-scale scenario (University's sport arena) at LOS.

tracked the trajectory of the runner and all have comparable performance (Fig. 6 and the third column in Fig. 4). In the best case scenario, all evaluated algorithms gave an error of less than $5\,\mathrm{cm}$ (Fig. 6 (a)). In the worst case scenario, the non-linear methods gave an error of up to $20\,\mathrm{cm}$ while the linear methods reached up to $12\,\mathrm{cm}$ (Fig. 6 (c)). Moreover, the abrupt changes in the corner of the field have two separate effects on the evaluated algorithms (Fig. 6 (b)). This is due to the applied dynamic model and process noise parameters in the filters (standard KF vs. EKF or UKF). In either case, all algorithms suffer from the abrupt changes in the corner of the field (Fig. 6 (b)), which can be improved by using a different dynamic model other than the PV model (section IV-A).

D. Performance Comparison in a NLOS scenario

A soft-NLOS scenario was evaluated in the sports hall, where a person carried a tag right in front of his chest to block the direct LOS communication between some anchors and the tag (Fig. 7). The device was at $0.3\,\mathrm{m}$ distance from the runner's body. The runner took two paths consequently one after another (on the border of a square and across the four corners). An intentional drift was made at the center when

The final published version of the paper is available on IEEE Xplore under the DOI: 10.1109/WPNC47567.2019.8970249

crossing from the bottom left corner to the upper right corner to examine the detection rate of the algorithms (Fig. 7 (a)).

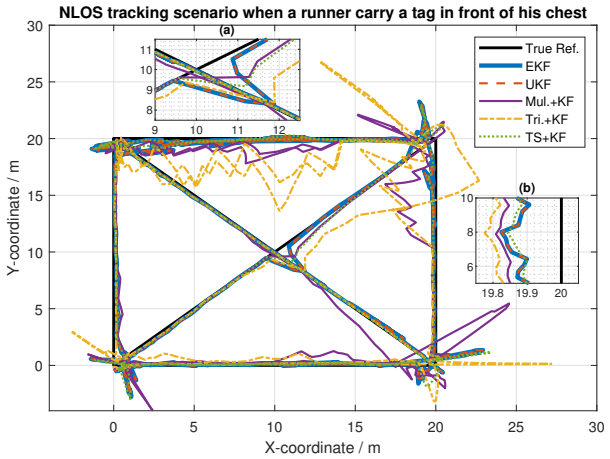

Fig. 7. Comparison of positioning algorithms when a moving tag is tracked in university's sports hall ($20\,\mathrm{m} \times 20\,\mathrm{m}$ arena) at a NLOS scenario (a runner hold the tag right in front of his chest at $0.3\,\mathrm{m}$ distance from the body).

In general, the two linear methods performed poorly in the NLOS scenario while the three non-linear methods had impressive performances (Fig. 7 and fourth column in Fig. 4). This confirms that the incremental correction in the positioning algorithm using the iterative and/or recursive approach has a significant benefit in a scenario where abrupt disturbances can occur in the measurement function. Additionally, the NLOS condition degraded the accuracy of all algorithms significantly (the fourth column in Fig. 4). The deviation can be visualized even in the best case scenarios (Fig. 7 (b)). Moreover, the intentional drift in the middle was accurately detected by all algorithms (Fig. 7 (a)), in which EKF and UKF have the best reaction while trilateration has the worst.

VI. Conclusion and Future Work

In literature, the linear positioning algorithms are less appreciated and their performances based on simulation results are usually regarded as poor. However, the quality of the ranging process (true-range) is neglected in simulation environments when the merits of performance comparison are conducted [6]–[10]. In this paper, we performed a comparative analysis of UWB-based true-range positioning algorithms based on experimental evaluations. The results show that the linear positioning algorithms, especially the multilateration, have superior performance compared to non-linear approaches in LOS condition (section V-C) when the measured ranges are very accurate. On the contrary, the non-linear approaches have better resistance in abrupt changes and significantly better performance in a NLOS scenario. In general, the non-linear techniques require a good approximation of the initial value, and they are computationally more expensive than the linear approaches. It should be noted that the presented results were based on the measurements without using any NLOS identification and mitigation techniques.

As future work, a maximum likelihood-based positioning algorithm and particle filter-based variants can be included in the comparative analysis of true-range positioning algorithms.

Acknowledgment

This work was supported by the Cluster of Excellence Cognitive Interaction Technology 'CITEC'(EXC 277) at Bielefeld University, which is funded by the German Research Foundation (DFG). Author Cung Lian Sang was supported by the German Academic Exchange Service (DAAD). The authors are responsible for the contents of this publication.

References

[1] M. Z. Win, D. Dardari, A. F. Molisch, W. Wiesbeck and J. Zhang, "History and Applications of UWB," in Proceedings of the IEEE, vol. 97, no. 2, pp. 198-204, Feb. 2009.

[2] C. L. Sang, M. Adams, T. Korthals, T. Hörmann, M. Hesse, and U. Rückert, "A bidirectional object tracking and navigation system using a true-range multilateration method," 2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Pisa, 2019, in press.

[3] A. Alarifi, et al., "Ultra Wideband indoor positioning technologies: analysis and recent advances," Sensors, vol. 16, no. 5, p. 707, 2016.

[4] C. L. Sang, M. Adams, T. Hörmann, M. Hesse, M. Porrmann and U. Rückert, "An analytical study of time of flight error estimation in two-way ranging methods," 2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Nantes, 2018, pp. 1-8.

[5] C. Lian Sang, M. Adams, T. Hörmann, M. Hesse, M. Porrmann, and U. Rückert, "Numerical and experimental evaluation of error estimation for two-way ranging methods," Sensors, vol. 19, no. 3, p. 616, 2019.

[6] F. Seco, A. R. Jimenez, C. Prieto, J. Roa and K. Koutsou, "A survey of mathematical methods for indoor localization," 2009 IEEE International Symposium on Intelligent Signal Processing, Budapest, 2009, pp. 9-14.

[7] J. Choliz, M. Eguizabal, A. Hernandez-Solana and A. Valdovinos, "Comparison of Algorithms for UWB Indoor Location and Tracking Systems," 2011 IEEE 73rd Vehicular Technology Conference (VTC Spring), Yokohama, 2011, pp. 1-5.

[8] G. Shen, R. Zetik and R. S. Thomä, "Performance comparison of TOA and TDOA based location estimation algorithms in LOS environment," 2008 5th Workshop on Positioning, Navigation and Communication, Hannover, 2008, pp. 71-78.

[9] K. Yu and I. Oppermann, "UWB positioning for wireless embedded networks," Proceedings. 2004 IEEE Radio and Wireless Conference (IEEE Cat. No.04TH8746), Atlanta, GA, 2004, pp. 459-462.

[10] K. Yu, J. Montillet, A. Rabbachin, P. Cheong and I. Oppermann, "UWB location and tracking for wireless embedded networks," in Signal Processing, vol. 86, no. 9, pt. 2153-2171, 2006.

[11] Z. Sahinoglu, S. Gezici, and I. Guvenc, "Ultra-wideband Positioning Systems," Cambridge University Press, 2008.

[12] B. T. Fang, "Trilateration and extension to Global Positioning System navigation," Journal of Guidance, Control, and Dynamics, vol. 9, no. 6, p. 715-717, 1986.

[13] W. H. FOY, "Position-location solutions by taylor-series estimation," in IEEE Transactions on Aerospace and Electronic Systems, vol. AES-12, no. 2, pp. 187-194, 1976.

[14] G. Welch and G. Bishop, "An introduction to the Kalman filter," Proc of SIGGRAPH, Course 8.27599-3175, 59, 2001.

[15] R. G. Brown and P. Y. Hwang, "Introduction to random signals and applied Kalman filtering," 4th Edition, John Wiley & Sons, 2012.

[16] S. J. Julier and J. K. Uhlmann, "New extension of the Kalman filter to nonlinear systems," in Signal processing, sensor fusion, and target recognition VI, vol. 3068, p. 182-194, 1997.

[17] E. A. Wan and R. Van Der Merwe, "The unscented Kalman filter for nonlinear estimation," Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373), Lake Louise, Alberta, Canada, 2000, pp. 153-158.

[18] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan," Estimation with applications to tracking and navigation: theory, algorithms and software," John Wiley & Sons, 2004.

[19] J, Yin, C. Park, J. Joo and S. Jeong,"Extended Kalman filter for wireless LAN based indoor positioning," Decision Support Systems 45, no. 4, p.960-971, 2008.

[20] T. Korthals, D. Wolf, D. Rudolph and U. Rückert, "Fiducial marker based extrinsic camera calibration for robot experiment platforms," 2019 European Conference on Mobile Robots (ECMR), 2019, in press.

[21] Decawave, "TREK1000 user manual," ver. 1.08, 2016.

The final published version of the paper is available on IEEE Xplore under the DOI: 10.1109/WPNC47567.2019.8970249