A New Solution for UWB Localization: Online Algorithms, Implementation and Testbed

Hassan Nahas, Asfandyar Sirhindi and Nikolaos M. Freris

***Abstract*— In several cyberphysical systems applications such as in robotics, transportation and sensor networks, it is imperative to devise low-power and low-complexity solutions for positioning and tracking. In this paper, we present a custom testbed in which agents equipped with Arduino microcontrollers and UWB transceivers are organized in a wireless ad hoc network. Ranging estimates are continually obtained and processed in real time to provide accurate position estimates via numerical schemes such as multilateration, Recursive Least Squares and Kalman Filtering. We expose illustrative experimental findings.**

***Index Terms*— Localization, Ultra-Wide Band (UWB) communication, Recursive Least Squares, Kalman Filtering.**

I. Introduction

The explosive proliferation of microprocessors, sensors, actuators and cloud computing has given rise to *Cyber-physical Systems* (CPS) such as large-scale sensor networks [1], transportation networks, multi-agent robotics systems and smart grids. There is a rising demand for cheap and reliable solutions for accurate positioning of smart devices equipped with on-board processing and wireless communication capabilities, and dynamically organized in a network. In such settings, power limitations impose developing protocols that are efficient in terms of computation and communication requirements, while maintaining fast responsiveness and adaptability needed for smooth real-time operation [2]. Traditionally, the Global Positioning System (GPS) has been considered a panacea for localization, offering a consistent service at a global level. However, GPS is limited by its ability to function indoors and in conditions where line-of-sight (LoS) communication with satellites is impossible. There are numerous fledgling applications where GPS is not an option: indoors (e.g., in public areas like malls, museums, schools and in industrial settings such as assembly lines and warehouses), inside tunnels as well as in dense downtown areas with tall buildings, and in underwater applications such as sensornets for geophysical exploration and oil extraction.

The problem of localization has been extensively studied [3], [4]. Solutions require distance measurements, angle measurements or combinations of both, between the entities that need to be localized (*tags*) and other reference nodes (*anchors*) of known position. Angle measurements can be obtained using antenna arrays but these methods remain unreliable and commercially unavailable. Distance measurements, on the other hand, can be readily embedded in a network through a range of techniques including Received Signal Strength Index (RSSI), Time of Flight (ToF) and Time Difference of Arrival (TDoA). By using a two-way ranging scheme, ToF can avoid the inaccuracies of clock synchronization [5] inherent in TDoA and, in general, outperforms RSSI in terms of accuracy [6]. This is the approach we adopt here. Our proposed solution features Ultra-Wide Band (UWB) communication due to its low-power requirements, and its ability to obtain high-quality Time of Flight (ToF) measurements in cluttered indoor environments [7].

Once the distance measurements are obtained, various algorithms can be used to obtain the position estimates. In [3], the authors apply Kalman Filtering (KF) to RSSI values and use multilateration to localize a single 3D target. Similarly, in [8], the authors have studied KF and variants using TDoA ranging methods.

This paper seeks to expand on the aforementioned research and investigate real-time localization using our custom testbed. In specific, we focus on the online processing of range measurements so as to boost the accuracy of position estimates, and track moving objects. For this purpose, we develop a Recursive Least Squares (RLS) estimator, as well as a Kalman Filter and discuss the implementation challenges alongside our experimental findings.

II. Problem Setup

Our study investigates the problem of obtaining optimal online position estimates of devices with variable unknown positions (tags) given distance measurements to devices with known positions (anchors). This is a well studied problem referred to as *multilateration*, that we briefly recap next.

A. Multilateration

Let $i = 1, 2, \ldots, n$, be an enumeration of the set of n anchor nodes. The squared Euclidean distance of a tag located at point (x, y) to the i-th anchor is given by:

$$d_i^2 = (x - x_i)^2 + (y - y_i)^2. \tag{1}$$

Choosing the first anchor (without loss of generality) as the reference, we obtain a set of $n-1$ linear equations:

$$(x_j - x_1)x + (y_j - y_1)y = \frac{1}{2}[(x_j^2 + y_j^2) - (x_1^2 + y_1^2) + d_1^2 - d_j^2],$$

for $2, 3, \ldots, n$, by subtracting (1) applied to $i = j$ and $i = 1$. In matrix form,

$$A\mathrm{x} = b, \tag{2}$$

New York University Abu Dhabi, Division of Engineering, Saadiyat Island, P.O. Box 129188, Abu Dhabi, UAE. E-mails: {hhn211,as7584,nf47}@nyu.edu.

© IPIN2017

$$A = \begin{bmatrix} x_2 - x_1 & y_2 - y_1 \\ x_3 - x_1 & y_3 - y_1 \\ \vdots & \vdots \\ x_n - x_1 & y_n - y_1 \end{bmatrix} \quad \mathrm{x} = \begin{bmatrix} x \\ y \end{bmatrix},$$

$$b = \frac{1}{2} \begin{bmatrix} (x_2^2 + y_2^2) - (x_1^2 + y_1^2) + d_1^2 - d_2^2 \\ (x_3^2 + y_3^2) - (x_1^2 + y_1^2) + d_1^2 - d_3^2 \\ \vdots \\ (x_n^2 + y_n^2) - (x_1^2 + y_1^2) + d_1^2 - d_n^2 \end{bmatrix}.$$

To tackle noisy measurements, we treat multilateration in the least-squares sense:

$$C(\mathrm{x}) := (A\mathrm{x} - b)^T (A\mathrm{x} - b), \tag{3}$$

$$\hat{\mathrm{x}} = \arg\min_{\mathrm{x}} C(\mathrm{x}) = A^\dagger b, \tag{4}$$

where $A^\dagger$ denotes the Moore-Penrose pseudoinverse of A [9]. In the case that anchors are not collinear, $A^T A$ has full-rank, and $A^\dagger = (A^T A)^{-1} A^T$.

B. Recursive Least Squares

Multilateration is sensitive to noise in the distance measurements. For applications where range measurements are acquired on a constant basis and/or agents are mobile, it is crucial to design low-complexity recursive estimators to filter data in the real-time. To account for historical data, the model (2) is re-written to reflect dependency on time instance k:

$$Ax_k = b_k.$$

The cost function at instance k is then defined as:

$$C_k(\mathrm{x}) := \sum_{i=0}^{k} \lambda^{k-i} (A\mathrm{x} - b_i)^T (A\mathrm{x} - b_i),$$

$$\hat{\mathrm{x}}_k = \arg\min_{\mathrm{x}} C_k(\mathrm{x}) = A^\dagger \frac{1-\lambda}{1-\lambda^{k+1}} \sum_{i=0}^{k} \lambda^{k-i} b_i,$$

where $\lambda \in (0, 1)$ is a "forgetting" coefficient, used to weigh the importance of past measurements. A recursive means of computing the optimal solution at the $k-$th instant with minimum overhead is derived:

$$\hat{\mathrm{x}}_k = \lambda \frac{1-\lambda^k}{1-\lambda^{k+1}} \hat{\mathrm{x}}_{k-1} + \frac{1-\lambda}{1-\lambda^{k+1}} A^\dagger b_k. \tag{5}$$

C. Kalman Filtering

The KF comprises two equations: a state equation and an observation equation [10].

$$\mathrm{x}_{k+1} = F_k \mathrm{x}_k + B_k u_k + w_k, \tag{6}$$

$$\mathrm{y}_k = H_k \mathrm{x}_k + v_k. \tag{7}$$

The state model we adopt captures Newtonian dynamics and the state x_k includes the two dimensional position p, velocity v and acceleration a of the tag.

$$\mathrm{x}_k = \begin{bmatrix} p_x & p_y & v_x & v_y & a_x & a_y \end{bmatrix}_k^T.$$

The transition matrix F_k is given by:

$$F_k := \begin{bmatrix} 1 & 0 & \Delta t_k & 0 & \frac{1}{2}\Delta t_k^2 & 0 \\ 0 & 1 & 0 & \Delta t_k & 0 & \frac{1}{2}\Delta t_k^2 \\ 0 & 0 & 1 & 0 & \Delta t_k & 0 \\ 0 & 0 & 0 & 1 & 0 & \Delta t_k \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

where Δt_k represents the sampling time, that is taken time-varying to capture for asynchronous packet exchanges. In this model, $B_k u_k$ accounts for change in acceleration. Due to the inability to accurately measured such changes, we will assume $u_k = 0$ in the sequel, but show how to measure acceleration via an Inertial Measurement Unit (IMU).

The observation model from (7) includes the multilateration model. Additionally, *local* acceleration (i.e., acceleration in a local coordinate system), is measured using an Inertial Measurement Unit (IMU). The global acceleration can be computed using a rotational transformation using the heading θ_k of the tag. In brief, the measurement y_k can be defined as:

$$\mathrm{y}_k := \begin{bmatrix} b & a_{\mathrm{loc}} \end{bmatrix}_k^T.$$

We define H_k as follows:

$$H_k := \begin{bmatrix} A & 0 & 0 \\ 0 & 0 & \Psi(\theta_k) \end{bmatrix}, \qquad \Psi(\theta_k) := \begin{bmatrix} \cos\theta_k & -\sin\theta_k \\ \sin\theta_k & \cos\theta_k \end{bmatrix},$$

where $\Psi(\theta_k)$ is used to transform the acceleration data to the global coordinate system. One of the formidable properties of our KF is complete observability as defined in [10]; we skip the analysis due to length considerations.

III. Implementation and Experimental Setup

A custom solution was designed for the problem of indoor localization as described in the prequel. This design includes a *hardware layer* based on the DWM1000 UWB transceiver and the Arduino Pro Mini microcontroller, a *network layer* that was specifically designed to deploy an ad-hoc network along with an integrated ranging protocol, and a *computational layer* that processes the distance measurements to yield accurate position estimates in real time.

A. Hardware Design

The key hardware components of the system are the Arduino Pro Mini and the Decawave DWM1000 transceiver modules. The Arduino Pro Mini was used for its low cost, light weight and easy availability especially with open-source platforms. The 5.0 V Arduino Pro Mini was integrated with the 3.3 V DWM1000 module through level shifting circuitry, cf. [11]. The tags and the anchors were homogeneous in terms of hardware. To perform controlled testing of stationary and mobile performance, the tag Arduino-DWM1000 circuits were attached to ground robots.

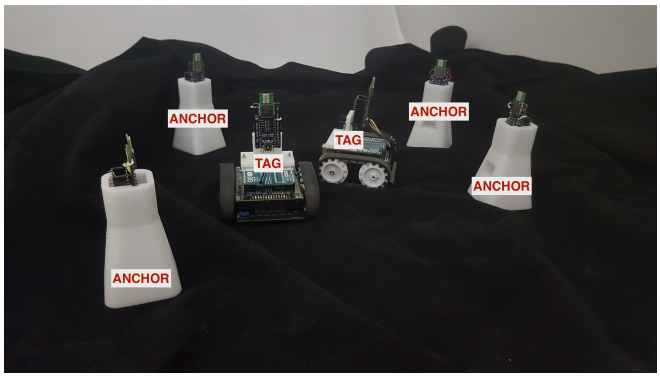

Fig. 1: System modules: custom motes arranged in anchors (on top of 3D printed white towers) and tags (aboard mobile robots).

B. Ranging Protocol

As discussed, a two-way ranging ToF protocol was used to calculate the distance between anchor and tag devices. This was calculated via an asymmetric two-way ranging scheme using an existing library [12]. In order to boost the measurement fidelity, a calibration mechanism was developed so as to remove the bias in the measurements. The final ranging protocol consists of two messages that include timestamps to estimate Time-of-Flight (ToF) measurements.

C. Network Management

The network configuration involves tags communicating using broadcast messages with anchors that respond in a sequential order based on predefined MAC addresses. Tags were programmed to self-organize autonomously to alleviate interference and crosstalk. Tags are allowed to enter or leave the network at any time. Although this process makes the time of each update cycle proportional to the number of devices in the network, it is more robust than a random order communication as originally implemented in [12].

D. Implementation

The multilateration and RLS algorithms defined in equations (2)-(4) and (5) respectively were programmed to the Arduino Pro Mini on each tag device. Moreover, KF was implemented according to the state and observation equations (6) and (7). However, instead of on-board runs, KF was emulated offline using post processing of the raw data collected from the motes; this was due to the incapability of the robot IMU to produce reliable heading estimates.

E. System Evaluation

A Bluetooth module was used to transmit the distance, acceleration, heading, and real-time position estimates for offline analysis and documentation. An overview of the system is illustrated in Fig. 1. To produce controlled mobile behavior, the robot was made to follow black lines that define a desired trajectory on a pale surface using an open-source library developed by Pololu [13].

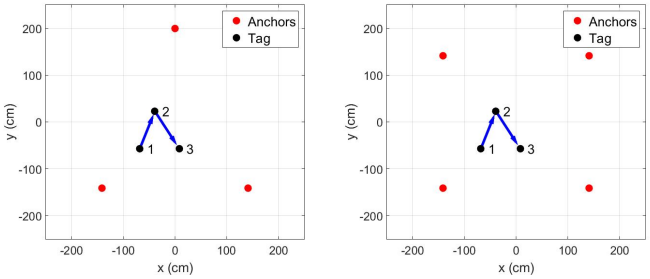

Fig. 2: Configurations for testing impact of number of anchors on localization performance. Left: 3 anchors; right: 4 anchors; one tag was used in both cases, placed in three different positions marked as 1,2,3.

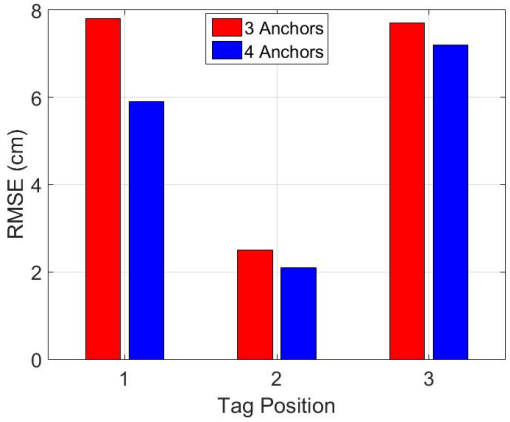

Fig. 3: RMSE evaluation for localization system with 3 and 4 anchors.

IV. Experimental Results

To evaluate the impact of the number of anchors on localization performance, two configurations were tested. Anchors were distributed around a circle with radius 2 m and the tag was placed in random positions around the center of the circle. The configurations of anchors and tag positions are shown in Fig. 2. We have gathered 100 localization samples produced by RLS, using $\lambda = 0.9$, for each tag position. We plot the Root Mean Square Error (RMSE) for the three configurations under study in Fig. 3; it is evident that increasing the number of available anchors improves localization accuracy.

To evaluate stationary performance for each algorithm (multilateration, RLS and KF), the tag was placed on the origin of the setup shown in Fig. 4 and 900 samples were collected for Root Mean Squared Error (RMSE) and Standard Deviation (SD) analysis. For mobile evaluation, 1000 position samples were collected as the robot followed the square track shown in Fig. 4. In order to measure the RMSE and SD, data from each experiment was divided into four parts where each part corresponds to one of the straight line segments in the square track. The position samples in the left and right segments were used to measure the RMSE and SD in the x-dimension. The top and bottom segment samples

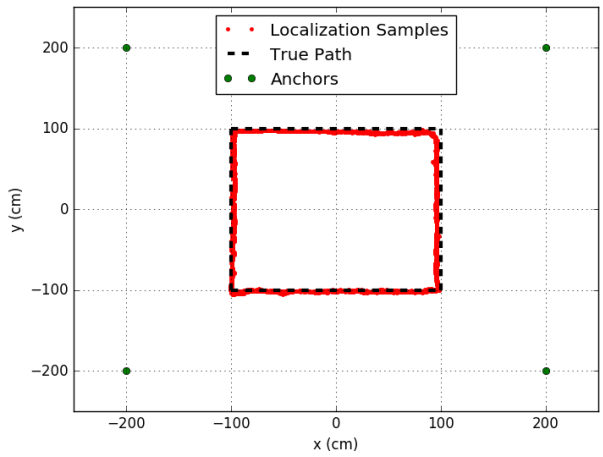

Fig. 4: Experimental setup: the Zumo robot is moving along the square trajectory marked as "true path." A path reconstruction based on RLS applied on collected measurements is also plotted in red.

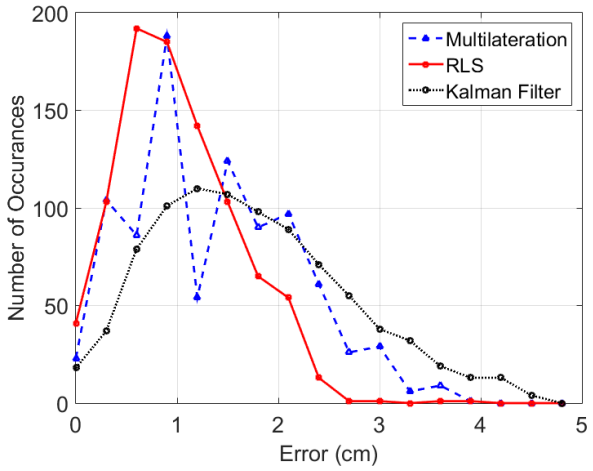

Fig. 5: Distribution of error for stationary position estimates.

were used for the y-dimension. One of the experimental runs involving the RLS algorithm is shown in Fig. 4 for reference. We have used $\lambda = 0.75$ for this experiment; a decrease in λ is appropriate to account for mobility by substantially reducing the impact of past measurements in position estimates. We have also implemented finite-horizon processing, but do not expose the results here due to length limitations.

The experimental results with the stationary tags are summarized in Table I below. The distribution of absolute error is also plotted for the case of a stationary tag in Fig. 5. The error and standard deviation are also depicted for mobile tag localization, cf. Table II.

V. Conclusion

We have developed a custom solution for indoor localization based on UWB ranging. The motes that we built possess an Arduino microprocessor and an UWB transceiver, and run protocols for network management and range acquisition. We have devised online algorithms to process range measurements based on Recursive Least Squares that substantially boost the performance over multilateration while maintaining low complexity. Besides, we have leveraged IMU data to provide a tracking mechanism of mobile tags based on Kalman Filtering. Our experimental findings showcase online localization and tracking within a few centimeters, which supports a low-cost solution for cyberphysical systems.

Algorithm	RMSE (cm)		SD (cm)	
	x	y	x	y
Multilateration	1.1	1.4	1.1	1.1
RLS	0.8	1.1	0.7	0.8
KF	1.4	1.7	1.1	1.1

TABLE I: RMSE and SD evaluation of each algorithm for stationary tag localization.

Algorithm	RMSE (cm)		SD (cm)	
	x	y	x	y
Multilateration	3.9	4.3	1.8	2.5
RLS	3.5	3.7	1.0	1.5
KF	3.7	4.0	2.4	3.5

TABLE II: RMSE and SD evaluation of each algorithm for mobile tag localization.

References

[1] N. Freris, H. Kowshik, and P. R. Kumar, "Fundamentals of large sensor networks: Connectivity, capacity, clocks, and computation," *Proceedings of the IEEE*, vol. 98, no. 11, pp. 1828–1846, 2010.

[2] P. Bahl and V. Padmanabhan, "Radar: an in-building RF-based user location and tracking system," in *Proceedings of IEEE INFOCOM*, vol. 2, 2000, pp. 775–784.

[3] J. Yang, H. Lee, and K. Moessner, "Multilateration localization based on Singular Value Decomposition for 3D indoor positioning," in *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, Oct 2016, pp. 1–8.

[4] D. Dardari, A. Conti, J. Lien, and M. Win, "The effect of cooperation on UWB-based positioning systems using experimental data," *EURASIP Journal of Advanced Signal Processing*, pp. 1110–8657, Jan. 2008.

[5] N. Freris, S. Graham, and P. R. Kumar, "Fundamental limits on synchronizing clocks over networks," *IEEE Transactions on Automatic Control*, vol. 56, no. 6, pp. 1352–1364, 2011.

[6] Q. Dong and W. Dargie, "Evaluation of the reliability of RSSI for indoor localization," in *2012 International Conference on Wireless Communications in Underground and Confined Areas*, Aug 2012, pp. 1–6.

[7] D. Jourdan, D. Dardari, and M. Win, "Position error bound for UWB localization in dense cluttered environments," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 44, no. 2, pp. 613–628, 2008.

[8] G. Binazzi, L. Chisci, F. Chiti, F. Fantacci, and S. Menci, "Localization of a swarm of mobile agents via unscented Kalman filtering," in *IEEE International Conference on Communications*, June 2009, pp. 1–5.

[9] R. Horn and C. Johnson, *Matrix analysis*. Cambridge University Press, 1985.

[10] A. Jazwinski, *Stochastic processes and filtering theory*. Courier Corporation, 2007.

[11] W. Holder, "UWB Ranging with the DecaWave DWM1000," https://sites.google.com/site/wayneholder/uwb-ranging-with-the-decawave-dwm1000—part-ii, 2016.

[12] T. Trojer, "Arduino-DW1000," https://github.com/thotro/arduino-dw1000, 2015.

[13] P. Corporation, "LineFollower," https://github.com/pololu/zumo-shield/tree/master/ZumoExamples/examples/LineFollower, 2013.