Ultra-wideband-based real-time positioning with cascaded wireless clock synchronization method

Article

Accepted Version

Ultra Wideband Based Real-time Positioning with Cascaded Wireless Clock Synchronization Method

Fengyun Zhang, *Member, IEEE,* Shengguang Hong, Yulong Ding, Shuang-Hua Yang⋆, *Senior Member, IEEE*

***Abstract*—Positioning services are often required in industrial and public applications. The requirements for positioning systems include sufficient positioning accuracy, appropriate anchor deployment in complex scenarios, stability, and precise clock synchronization between anchors. Therefore, algorithms designed for single-room and simple laboratory scenarios are not suitable. To address these challenges, this paper proposes a comprehensive solution that combines ultra-wideband (UWB) technology, time difference of arrival (TDoA)-based positioning, and a cascaded wireless clock synchronization algorithm. First, we introduce the cascaded wireless clock synchronization algorithm. Second, we discuss the control algorithm for transmitting clock calibration packets (CCP) through wireless broadcast. Finally, we define the time of arrival selection strategy for time difference of arrival calculation. For multi-room scenarios, we present a positioning boundary optimization method based on received signal strength power and the first path power. This method performs well in real-world experiments.**

***Index Terms*—Ultra wideband (UWB), time difference of arrival (TDoA), cascaded wireless clock synchronization (CWCS), time-base selection strategy**

I. Introduction

In recent years, location-based services have become highly attractive due to the rapid development of wireless communication technology and the Internet of Things [1], [2]. In these scenarios, accurately and swiftly obtaining the location information of mobile terminals is a crucial problem that researchers are focusing on [3]. When it comes to indoor positioning, ultra-wideband technology offers significant advantages in terms of positioning accuracy, system capacity, and power consumption control compared to other technologies such as BLE, ZigBee, and Wi-Fi [4].

UWB is a radio technology that uses pulses instead of carriers to transmit data, ensuring low power consumption. The Federal Communications Commission (FCC) has specified that UWB pulses should occupy a wide frequency bandwidth (>500 MHz) or a relative bandwidth (>20%) within the restricted frequency band of 3.1 to 10.6 GHz, with a power density of −41.3 dBm/MHz [5]. According to the data provided in the manufacturer's technical documentation, the ranging accuracy based on UWB technology can reach up to 10 cm, and the frequency band of UWB allows for data transmission rates of up to 500 Mbit/s [6].

UWB-based indoor positioning schemes include Time of Flight (ToF) [7], Two-Way Ranging (TWR) [8], Time of Arrival (ToA) [9], Time Difference of Arrival (TDoA) [10], Angle of Arrival (AoA) [11], and Phase Difference of Arrival (PDoA) [12]. Among these schemes, positioning methods based on TWR or ToF are constrained by the bidirectional communication mechanism between anchors and tags, requiring more communication resources to achieve centimeter-level positioning accuracy. AoA and PDoA positioning modes are limited by UWB antenna technical specifications, which impose higher requirements for line-of-sight transmission. Therefore, TDoA-based positioning, supporting higher tag densities and suitable for cascaded anchor deployment, has garnered more attention from researchers. High-precision real-time indoor positioning based on UWB demonstrates a strong correlation with mobile computing. UWB technology provides accurate location data, facilitating mobile computing applications, including indoor navigation, location awareness, and augmented reality. This symbiotic relationship strengthens both indoor positioning and mobile computing, advancing them through seamless data interaction and real-time capabilities.

To satisfy the deployment requirements of a real-time positioning system, UWB-based TDoA positioning and wireless clock synchronization must fulfill three core requirements:

1) **Support cascaded anchor deployment:** UWB and TDoA-based positioning methods should support the deployment of anchors in complex scenarios. Since all anchors require tight wireless clock synchronization, effective schemes to develop a wireless clock synchronization algorithm that supports a theoretically abundant number of anchors are necessary.
2) **Enable higher tag densities:** UWB-based positioning systems should accommodate higher tag densities in real-world industrial and public spaces. To ensure the success rate of wireless signal reception and transmission under limited channel capacity, a control algorithm for clock calibration packets (CCP) transmission through wireless broadcast is required.
3) **Maintain positioning accuracy:** UWB-based TDoA positioning should ensure accurate tracking in complex environments. To reduce positioning errors in such scenarios, a time-base selection algorithm is needed to find a valid clock synchronization link for TDoA calculations. However, the presence of ambiguous positioning boundaries in

(Corresponding author: Shuang-Hua Yang.)

Fengyun Zhang is with the College of Electronic and Information Engineering, Southwest University, Chongqing, China; Yulong Ding is with the Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen, China; Shuang-Hua Yang is with the Department of Computer Science, University of Reading, UK; Fengyun Zhang, Shengguang Hong, Yulong Ding, and Shuang-Hua Yang are also with the Shenzhen Key Laboratory of Safety and Security for Next Generation of Industrial Internet, Shenzhen, China (email: zhangfy2019@mail.sustech.edu.cn; 11949001@mail.sustech.edu.cn, dingyl@sustech.edu.cn, yangsh@sustech.edu.cn).

transition regions between rooms introduces non-negligible positioning errors.

To address these challenges, this paper presents a solution that combines UWB-based TDoA positioning with the cascaded wireless clock synchronization algorithm. The main contributions of this paper can be summarized as follows:

- We design a cascaded wireless clock synchronization (CWCS) algorithm with improved performance to support multiple master anchors.
- We propose a control algorithm for transmitting clock calibration packets through wireless broadcast to avoid signal collisions.
- We define a time-based ToA selection strategy for TDoA calculations to ensure accurate tag tracking.
- We propose a positioning boundary optimization method based on received signal strength and first path power for multiple space scenarios.

The remainder of the paper is organized as follows: Section II provides a literature review on TDoA positioning and wireless clock synchronization. Section III presents a comprehensive solution combining the cascaded wireless clock synchronization algorithm with the control algorithm for transmitting clock calibration packets. Section IV proposes the time-based ToA selection strategy for TDoA calculation and the positioning boundary optimization method. Section V discusses the evaluation and experiments. Finally, Section VI concludes and discusses the findings.

II. Related Work

For indoor positioning using UWB-based TDoA positioning, precise clock synchronization between anchors is essential. It is necessary to synchronize all the anchors to ensure a uniform reference time. Previous studies have summarized the latest clock synchronization methods for WSNs [13]. These studies have discussed the primary clock model and various clock synchronization protocols. They have also presented and compared different wireless clock synchronization algorithms, including linear interpolation (LI), proportional-integral (PI) control, proportional-integral differential (PID) control, proportional-integral integral (PII) control, and a Kalman filter (KF) [14]. Another study has designed and implemented a wireless synchronization method with clock drift compensation, suitable for time division multiple access (TDMA) scenarios [15]. In the context of TDoA positioning, a novel wireless clock synchronization algorithm similar to linear interpolation has been proposed and validated for multi-user scenarios [16]. Additionally, an efficient clock synchronization algorithm based on a two-state Kalman filter has been proposed and performed well in indoor positioning scenarios [17]. Moreover, a Kalman filter-based protocol for wireless clock synchronization of multiple UWB anchors, aiming to achieve accurate positioning for a mobile robot, has been presented [18]. Another study has proposed a new wireless synchronization method that corrects the TDoA by using a pair of packets and a known recorded timestamp [19].

However, most of the wireless clock synchronization algorithms mentioned above focus on small-scale deployments in a single positioning area or simple laboratory scenarios. In the case of UWB-based TDoA positioning in actual deployments, support for multiple anchors is required, necessitating precise synchronization of the clocks of all the anchors. To address this, a novel wireless clock synchronization scheme called TALLA has been proposed for TDoA positioning, covering a larger area while ensuring positioning accuracy [20]. Based on the TDMA scheduling mode, this method supports continuous multi-hop operations between anchors. A novel approach to UWB positioning is introduced by [21], wherein the time difference of arrival (TDOA) of signals sent by known anchor points is calculated by target devices, enabling support for multiple targets. Sub-meter positioning accuracy is achieved in real-world experiments. Another study has presented a wireless broadcast relative positioning and clock synchronization system called BLAS, addressing the challenges in multi-agent systems (MAS) [22]. Experimental and simulation results have demonstrated that the BLAS system performs well in MAS scenarios with a large number of agents. Moreover, novel methods for precise multi-hop wireless clock synchronization and a positioning zone selection scheme have been proposed, achieving significantly higher positioning accuracy in complex scenarios with multiple separated spaces compared to existing approaches [23].

A novel passive ranging scheme is introduced by [24] to address the challenges of indoor localization. By leveraging wireless communication between infrastructure nodes, clients can determine their relative location without the need to transmit signals. A UWB-based indoor localization system (Snaploc) is proposed by [25], which allows an unlimited number of tags to self-localize at a theoretical upper bound of 2.3 kHz. This is achieved by allowing tags to passively receive signals from multiple anchors and estimate their position based on the time difference of arrival. Similarly, a novel virtual Two-Way Ranging method for passive reception-based localization without time synchronization (VULoc) is introduced by [26]. Redundant ranging packets are utilized for auto-calibration, and an adaptive anchor scheduling algorithm is employed to mitigate NLOS effects. A wireless high-accuracy clock synchronization solution for multi-node distributed cooperative systems (Wicsync) is proposed by [27]. It is implemented using an optimized wireless two-way clock synchronization and mutual calibration protocol, along with a UWB synchronization hardware architecture, to achieve non-GNSS-aided, high-accuracy, multi-node, low-cost wireless clock synchronization. Time synchronization in WSN using low-cost UWB transceivers is explored by [28]. Precise synchronization of 0.14 ns between nodes is achieved in the research using commercial UWB hardware, enabling support for applications with stringent synchronization requirements. The introduction of a novel Gradient Clock Synchronization (GCS) algorithm for UWB ranging networks, addressing the issue of chaotic global clock rate, is made by [29]. The algorithm guarantees the convergence of the global clock rate to the average of individual clock rates, and asymptotic stability in clock rate errors is achieved. The use of UWB technology for indoor real-time location systems (I-RTLS) in IoT applications is discussed by [30], with various methods for device localization

being explored, emphasizing time synchronization for anchors. A high-precision wireless clock synchronization method using UWB technology is introduced by [31], [32]. Sub-nanosecond accuracy is achieved through distance measurements and multi-agent consistency theory, as validated in hardware-in-the-loop experiments. Novel message-based TDOA equations for hyperbolic localization that eliminate the need for clock synchronization are introduced by [33], and these equations outperform existing methods. The proposed approach utilizes anchor nodes and innovative TDOA equation reformulations to minimize clock drift errors, validated through experiments with eight anchor nodes and one localizing node in a 128 m^3 volume.

While the research results mentioned above demonstrate impressive clock synchronization and positioning accuracy, it is important to note that these clock synchronization algorithms are primarily tailored for ranging mode or involve the integration of ranging information into TDoA positioning mode. Consequently, they may not be well-suited for purely TDoA-based positioning systems. In real-world industrial deployments, anchors are typically stationary and can be powered through wired connections or equipped with higher-capacity batteries. In contrast, tags are commonly implemented as bracelets or work cards. Minimizing communication frequency is crucial to optimize tag battery life. Our approach adheres to this principle, enabling tags to operate with maximum energy efficiency. Compared to alternative approaches, our scheme excels in scalability, positioning accuracy, deployment simplicity, and reduction of power consumption for tags. Remarkably, our solution also extends to the downlink TDoA scheme, offering enhanced versatility. Consequently, this paper intentionally avoids distinguishing between uplink and downlink TDoA, aligning with the broader scope of our research.

III. Solution Design

A. Cascaded Wireless Clock Synchronization

1) Clock Modeling: The clock function of an anchor is described as Equation (1), [34] and [35]:

$$t_i = \theta_k + \delta_k t + \frac{1}{2}\phi_k t^2 + \omega_k(t) \tag{1}$$

where t_i denotes the local time of anchor i, and k indicates the moment when the anchor receives the synchronization message. θ_k is the clock offset, δ_k denotes the clock drift, t is the reference time, $\omega_k(t)$ is the random frequency noise, and ϕ_k denotes the frequency drift.

The clock model is described in Figure 1. The slope of the ideal reference clock is 1, and the clock offset is 0. In a real clock system, the slope of the clock drift is greater than 0 and fluctuates around 1.

Without considering the influence of frequency drift and noise, we could obtain a simplified linear version of the clock model as Equation (2):

$$t_i = \theta_k + \delta_k t \tag{2}$$

Figure 2 illustrates a typical and minimum UWB-based positioning network (UPN) comprising four anchors, one tag, one switch, one DHCP server, and a central localization engine

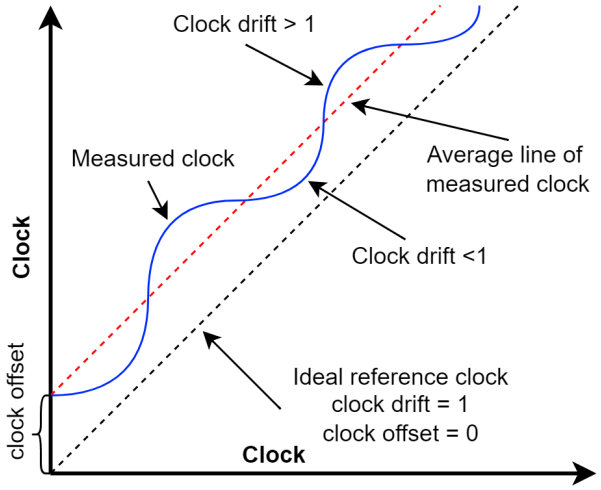


Fig. 1: Clock model.

(CLE). The communication between the anchors and the CLE is facilitated using the TCP/IP protocol. The master anchor (MA) communicates with the slave anchors (SA2, SA3, and SA4), while the tag (T1) communicates with all the anchors. The CLE operates on an upper computer, which serves as the central processing unit for the positioning network. The MA transmits a synchronized frame (Sync message) periodically, which is received and timestamped at the slave anchors. The tag (T1) transmits a blink frame (Blink message) periodically, which is received and timestamped at all the anchors, and the timestamps represent the raw time of arrivals (ToAs). Subsequently, each anchor sends the ToA reports to the CLE via a wired or wireless network. The CLE then utilizes the ToAs to estimate the tag's location. During the real-time positioning system operation, the master anchors and tags broadcast UWB packets periodically.

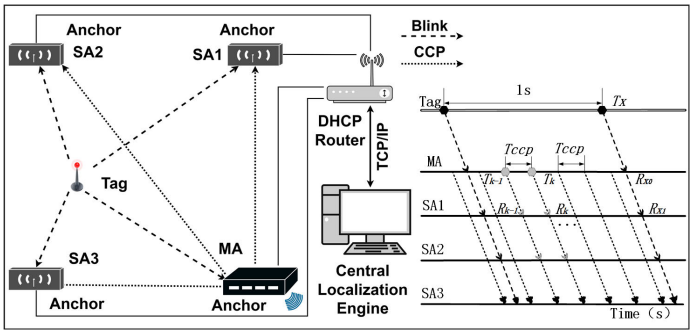


Fig. 2: UWB-based Wireless Positioning Network.

The communication within the wireless clock synchronization (WCS) algorithm is also illustrated in Figure 2, depicting the reception and transmission of positioning and synchronization messages over time. The master anchor is denoted as **MA**, and the slave anchors as **SA2-SA4**. The slave anchors (**SAs**) are deployed exclusively for positioning and do not participate in time synchronization. The figure represents five timelines, with the top one depicting the tag and the bottom four corresponding to individual anchors. The dark dashed

lines represent positioning packets (**blinks**) transmitted by the tag at a frequency of one second, while the light dashed lines represent the synchronous packet (**CCP**) sent by the master anchor at a 150 ms interval. R_{x1} and R_{x0} symbolize the timestamps recorded by SA and MA when they receive a blink, respectively, and can be considered as the raw Time of Arrivals (ToAs). *SN* denotes the sequence number. The one-way synchronization messages determine the clock drift and clock offset. The parameters are calculated as follows:

$$\delta_k = \frac{T_k - T_{k-1}}{R_k - R_{k-1}},\ \theta_k = T_k - R_k$$

where R_k and R_{k-1} are the timestamps recorded by SA when it receives the CCP, and T_k and T_{k-1} are the timestamps recorded by MA when it sends the CCP.

2) Wireless Clock Synchronization Algorithm: Each UWB device, including the anchor, is equipped with an independent high-resolution timer. Over time, the oscillation frequency of the timer will drift, and this drift must be corrected through clock calibration. Synchronization is crucial for all anchors, as precise timestamping is essential for accurate location estimation. There are three main issues that need to be addressed in synchronization:

- **Clock Offset Synchronization** - Ensure that recorded timestamps are synchronized between anchors using the same reference time.
- **Clock Drift Calibration** - Calibrate the clock drift that arises due to the circuit board and RF components.
- **Frequency Drift Elimination** - Eliminate the deviation in oscillation frequency caused by factors such as temperature and other environmental conditions.

The linear interpolation (LI) algorithm, which utilizes a 150 ms synchronization interval, has been shown to achieve decimeter-level synchronization accuracy [15]. However, a drawback of this algorithm is that the latency in retrieving the corrected time of arrival (ToA) is at least as long as the synchronization period, making it unsuitable for real-time positioning systems. On the other hand, the algorithm based on a two-state Kalman filter can eliminate clock drift within the synchronization period and ensure the continuity of clock synchronization [17], but it is unable to handle frequency drift. To address synchronization errors caused by linear clock offset, nonlinear clock drift, and frequency drift during the synchronization period, we propose a clock synchronization algorithm that combines linear interpolation with a three-state Kalman filter (LI-KF). The LI-KF algorithm consists of three main steps:

- **Step 1:** Linear interpolation process.
- **Step 2:** Raw ToA correction process.
- **Step 3:** Kalman filtering process.

Linear interpolation process: The clock model for the linear interpolation algorithm is described by Equation (2). When the master anchor transmits the synchronization message, the timestamp T_k is recorded. Similarly, when the slave anchor receives the synchronization message from the master base station, the receiving timestamp R_k is recorded. Once the Central Localization Engine (CLE) receives the timestamps of the master anchor and slave anchor at time instants k and $k-1$, the linear interpolation process can be initiated.

Raw ToA correction process: By using $T_{k-1} = T_k$ and $R_{k-1} = R_k$, the sent and received timestamps can be updated. Assuming that the raw ToA is greater than R_k for all received timestamps, the synchronized timestamps (ToA_k) of anchor i can be calculated using the following equation:

$$\begin{cases} \Delta = ToA - R_k \\ ToA_k = T_k + \Delta/\delta_k \end{cases}$$

Kalman filtering process: As the time interval error is not linear, a three-state Kalman filter is implemented to eliminate the nonlinear clock drift and frequency drift during the synchronization period [36]. The state vector X_k is expressed as $X_k = [ToA_k, \delta_k, \phi_k]^T$, where $\delta_k = \frac{T_k - T_{k-1}}{R_k - R_{k-1}}$ and ϕ_k represents the minimum resolution of the crystal oscillator, such as 0.1 ppm. The 3×3 state transition matrix F, and the time difference of T_{ccp}, which denotes the time elapsed between the previous estimation time T_{k-1} and the current time T_k, are used in the following equation:

$$\begin{bmatrix} ToA_{k+1} \\ \delta_{k+1} \\ \phi_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & T_{ccp} & \frac{T_{ccp}^2}{2} \\ 0 & 1 & T_{ccp} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} ToA_k \\ \delta_k \\ \phi_k \end{bmatrix}$$

Based on the three-state Kalman filter model, the estimating process of clock synchronization can be expressed as:

$$\hat{X}_{k+1}^{\breve{}} = F\hat{X}_k + w_k \tag{3}$$

$$Z_{k+1} = H\hat{X}_{k+1}^{\breve{}} + v_k \tag{4}$$

where $\hat{X}_{k+1}^{\breve{}}$ is a prior estimated state vector, and $\hat{X}_k$ expresses a posteriori estimate. The observation vector Z_{k+1} is the combination of the prior estimate $\hat{X}_{k+1}^{\breve{}}$ and $H\hat{X}_k^{\breve{}}$, and the coefficient matrix can be expressed as $H = [1\ 0\ 0]$. w_k follows Gaussian distribution: $w_k \backsim \mathcal{N}(0, Q)$. The noise vector obeys the zero mean Gaussian distribution, which can be expressed as a 3×3 constant matrix Q. v_k is the the observation noise, and v_k follows $v_k \backsim \mathcal{N}(0, R)$. According to the variance of the measurements taken by the UWB manufacturer, $R = \sigma^2 = 1.5 \cdot 10^{-20} s^2$.

The necessary steps of correction and prediction of Kalman filter can be estimated as the following formulas.

$$P_{k+1}^{\breve{}} = FP_kF^T + Q \tag{5}$$

$$K_k = P_{k+1}^{\breve{}} H^T \left(HP_{k+1}^{\breve{}} H^T + R\right)^{-1} \tag{6}$$

$$\hat{X}_{k+1} = \hat{X}_{k+1}^{\breve{}} + K_k \left(Z_{k+1} - H\hat{X}_{k+1}^{\breve{}}\right) \tag{7}$$

$$P_{k+1} = (I - K_kH)\, P_{k+1}^{\breve{}} \tag{8}$$

where P_{k+1} is the estimated covariance matrix, which can be recursively derived from the initial matrix P_0. The covariance matrix P_0 is related to the initial state vector. $P_{k+1}^{\breve{}}$ is a prior estimate P_{k+1}, and K_k is the Kalman gain.

$$P_0 = \begin{bmatrix} \sigma_{ToA}^2 & 0 & 0 \\ 0 & \sigma_\delta^2 & 0 \\ 0 & 0 & \sigma_\phi^2 \end{bmatrix}$$

where σ_{ToA}^2, σ_{δ}^2, and σ_{ϕ}^2 represent the initial variances of the state vector components. We could summarize the proposed LI-KF algorithm in **Algorithm 1**.

Algorithm 1 LI-KF algorithm

Input:

T_k, R_k; T_{ccp}, X_0, Q, R, P_0

Output:

The corrected array of ToA, $ToA[]$

Calculate the clock drift δ_k and clock offset θ_k
for the i_{th} ToA_k in $ToA[]$ **do**
  $\Delta = ToA - R_k$
  $ToA_k = T_k + \Delta/\delta_k$
end for
$T_{k-1} = T_k$
$R_{k-1} = R_k$
Initialize X_k and Z_k
Calculate the transmission matrices F and H
Perform the predict process:
  $\hat{X}_{k+1}^{\check{}} \leftarrow F\hat{X}_k + W_k$
  $P_{k+1}^{\check{}} \leftarrow FP_kF^T + Q$
Perform the correct process:
  $K_k \leftarrow P_{k+1}^{\check{}} H^T \left(HP_{k+1}^{\check{}} H^T + R\right)^{-1}$
  $\hat{X}_{k+1} \leftarrow \hat{X}_{k+1}^{\check{}} + K_k \left(Z_{k+1} - H\hat{X}_{k+1}^{\check{}}\right)$
  $P_{k+1} \leftarrow (I - K_kH)\, P_{k+1}^{\check{}}$
Return $ToA[]$

3) Complexity Analysis of WCS: The proposed wireless clock synchronization algorithm, based on the three-state Kalman filter (LI-KF), is a linear and theoretically optimal unbiased filter with a complexity of $O(m^{2.376} + n^2)$, where m is the observation dimension and n is the number of states. In the Kalman filter-based wireless clock synchronization algorithm, the meticulous adjustment of parameters R and Q holds paramount significance. The value of R is determined based on the amplitude of process noise, and Q is assigned relative to R. Choosing a smaller Q indicates higher confidence in the observation, while a larger Q is selected when the observation is less reliable. A larger Q facilitates quicker convergence but results in a slower response to observation changes. Conversely, a smaller Q boosts confidence, enabling a faster response but potentially compromising system state convergence.

4) Cascaded Wireless Clock Synchronization: To meet the requirements of actual deployments, it is necessary to employ more than one master anchor (**MA**) and many slave anchors (**SA**). One of the master anchors is designated as the primary master anchor, while the others are secondary master anchors. The secondary master anchors follow the primary master anchor and transmit its clock calibration packet (CCP) at a specified lag time after receiving the CCP from the primary master anchor. If there are multiple secondary master anchors, their CCP transmit times should be staggered to avoid collision. We establish a multi-level cascaded topological structure of the master anchors to coordinate the order of sending CCPs between the primary master anchors and the secondary master anchors for complete clock synchronization. Figure 3 illustrates this cascaded structure, where **MA1** is the primary master anchor. Any **MA** receiving the CCPs sent by **MA1** is a secondary master anchor. Additionally, the **MA** receiving the CCPs from the secondary master anchor is a level-3 master anchor (a tertiary master anchor). Therefore, **MA2**, **MA3**, and **MA4** are secondary master anchors, while **MA5** and **MA6** are tertiary master anchors, as shown in Figure 4. This cascading model of master anchors follows the order: primary master anchor $->$ secondary master anchor $->$ tertiary master anchor $->$ level-4 master anchor $->$ ...$->$ level-N master anchor. When a lower-level master anchor receives the **CCP** from its upper-level master anchor, it starts to send the **CCP** after a specified interval (Lag).

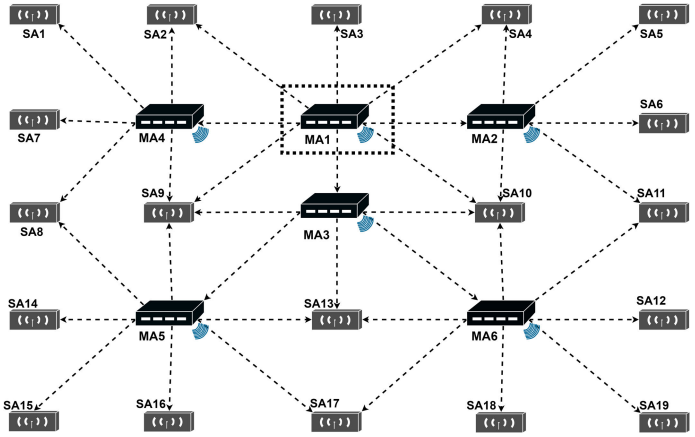


Fig. 3: Diagram of WCS with multiple master anchors.

B. Clock Calibration Packets Transmission Control

UWB-based TDoA positioning requires cascaded wireless clock synchronization to synchronize the timestamps of the anchors. We have designed the Clock Calibration Packet (CCP) transmission control algorithm to track the clock offset and clock drift between the master anchors and slave anchors. Ensuring that all slave anchors are within the coverage range of the master anchor is challenging due to the limited transmission distance of wireless signals in actual deployment scenarios. Therefore, multiple cascaded master anchors need to be arranged. To prevent CCPs sent by the primary master anchor from overlapping during transmission, we propose the use of the CCP transmission control algorithm. This algorithm ensures that wireless clock-synchronized packets do not overlap in the cascaded system of multiple masters, thus avoiding CCP collisions. The primary master anchor sends CCPs at regular intervals, and the CCP transmissions of all other master anchors are scheduled based on the initial master's CCP or the CCP transmissions of other secondary master anchors.

Figure 4 illustrates the concept of the transmission control algorithm of CCP. Each horizontal line represents the CCP transmission activity of a master anchor. The curved arrows indicate three CCP transmissions from the master anchor MA1 to three secondary master anchors (MA2, MA3, and MA4), and two CCP transmissions from MA3 to MA5 and MA6.

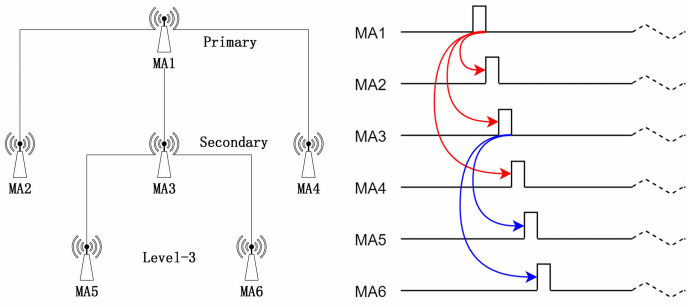


Fig. 4: Staggered CCP control transmission for 6 master anchors.

The directions of the CCP transmissions between the master anchors and the CCP sending intervals (lag time) can be defined through the CLE's configuration file. Each master anchor utilizes its local clock as the sending timestamp. The time lags are also measured using the local anchor's clock. The CCPs sent through the anchors are arranged by the CCP transmission control algorithm, which ensures that no two CCP transmissions will occur simultaneously from the anchors within the receiving range. This prevents collisions among CCPs sent by the anchors. For a secondary master, the anchor sends a time-lagged CCP: the received CCP is time-stamped, and the time-lag delay is added to determine the transmission time of the CCP. All CCPs are sent at a precisely defined CCP repetition rate. The scheme is configured with a redundancy mechanism, so if a secondary master anchor encounters a signal reception error, the next CCP will still be sent at the same interval as the last transmission to ensure the normal operation of the system.

The details of the CCP transmission control algorithm are presented in **Algorithm 2**. The algorithm consists of four main parameters: a nominal period (NP), master ID to follow ($Master_to_follow$), configured lag time (Lag), adjusted time period (TP), and the latency from the timer interrupt to restarting the timer with a new timestamp ($T1$). In the UWB-based embedded system, Timer_IRQ represents the system clock interrupt, and DW_IRQ represents the RF clock interrupt of UWB. As shown in Figure 4, the number of $Lags$ can vary based on the number and sequence of the current cascaded anchors. For example, there can be one, two, or three lags when there are three secondary master anchors following the primary master anchor. The protocol scheme described below adjusts the timing of the CCP transmission to lag the reception time of a CCP frame from another master. This allows the follow-on Master to send its CCP at a later, non-conflicting time.

Algorithm 2 CCP transmission control algorithm

Input:
Lag, NP
Output:
Adjusted TP
Initialize Timer_IRQ and DW_IRQ
Set the level of the master anchors
Select the primary master anchor
Determine the cascaded relationship between anchors, obtain the $Master\_to\_follow$
Calculate the number of $Lags$ based on the $Master\_to\_follow$
Start RTLS
Record the CCP sending time of primary master anchor, CCP_TX
Start Timer_IRQ, record the timestamp, $T1$
Record the CCP receiving time of the secondary master anchor, CCP_RX
Start DW_IRQ, record current time period, TP
Calculate adjusted TP
if *CCP is received in last awaiting period* **then**
  $lag\_error = CPP\_TX - CPP\_RX - Lags$
  $TP = NP - Lag\_error$
else if *CCP is not received in last awaiting period* **then**
  $TP = NP$ *or* $(last\ adjusted\ TP)$
else
  $TP = TP - T1$
end
Return Adjusted TP

IV. TDoA-based Real-time Positioning

A. Time-base ToA Selection Algorithm

The CLE requires four or more times of arrival (ToAs) to estimate the locations of tags, also known as targeted objects or tags. In a real-time positioning system, more than four anchors can capture the blink of a tag. Hence, the CLE collects all the ToAs from the slave anchors and invokes the positioning algorithm at a specified interval. In a multi-master system, the received ToA of any anchor must be synchronized with the time-base of one of the designated master anchors. Before calling the positioning algorithm, the CLE needs to select all or some of the ToAs and corrected ToAs for TDoA calculation. We define the corresponding rule and choose the anchor with the maximum number of masters as the time-base anchor (TBA). Based on the time-base ToA selection algorithm, the actual calculation of TDoA can be summarized in four cases.

1) If the time-base anchor is a master anchor and the current anchor is the slave of the TBA, TDoA data can be calculated by subtracting the ToA of the TBA from the syncToA of the current anchor.
2) If the TBA is not a master anchor and the current anchor is the master anchor, TDoA data can be calculated by subtracting the syncToA of the current anchor from the ToA of the TBA.
3) If neither the TBA nor the current anchor is a master, but they can be synchronized with a common master anchor, TDoA data can be calculated by subtracting the syncToA of the TBA from the syncToA of the current anchor.
4) If the current anchor or TBA have a common synchronized link, TDoA data can be calculated by subtracting the ToA of the TBA from the syncToA of the current anchor's master anchor.

The details of the time-based ToA selection algorithm are demonstrated in **Algorithm 3**. We will use a typical case to

illustrate how our proposed approach calculates TDoA based on collected clock-synchronized ToAs in the cascaded anchor coverage area (Figure 5). When the tag is in the area marked by the red dotted line (a) or the green dotted line (b), the first case of Algorithm 3 is satisfied. Hence, we can calculate the corresponding TDoA according to the predefined rules. If the tag moves to the area marked by the blue dotted line (c), we could use the second or third cases of Algorithm 3 to calculate the TDoA. In cases where the tag is within the blue dotted line area, but direct TDoA calculation is not possible (d), we can compute the TDoA using the cascaded synchronous link, the fourth case of Algorithm 3.

Algorithm 3 Time-base ToA selection algorithm

Input:
$Blinks(bl)$, $ToAs$
Output:
Calculated $TDoA^*[]$

Calculate the number of masters for each anchor
Choose the anchor with maximum masters as time-base anchor (TBA)
Set TBA with $TDoA[a] = 0.0$ and $a = 0$
Find and calculate $TDoA$ for all anchors w.r.t the TBA
Deal with the typical cases of each anchor w.r.t the TBA:
if *TBA is a master anchor and current anchor is the slave of TBA* **then**
  $TDoA = bl.Anchor.syncToA - bl.TBA.ToA$
else if *TBA is not a master and current anchor is the master anchor* **then**
  $TDoA = bl.TBA.ToA - bl.Anchor.syncToA$
else if *Neither the TBA or current anchor are masters, sync to a common master* **then**
  $TDoA = bl.Anchor.syncToA - bl.TBA.syncToA$
else if *Current anchor or TBA have a common synchronized link* **then**
  $TDoA = bl.Anchor.syncMasterToA - bl.TBA.ToA$
else
  Could not calculate $TDoA$
end
for $TDoA$ in $TDoA[]$ **do**
  $TDoA[a] = TDoA$
  $a++$
end for
Return $TDoA^*$

B. Positioning Boundary Optimization Method

In real-world positioning scenarios, multiple independent positioning areas are separated by walls that can be penetrated by UWB signals, such as gypsum board. Two main situations need to be addressed in positioning. First, when the tag moves from one area to another, the positioning packets sent by the tag will be captured by the anchors that belong to both positioning areas. If the positioning packets received by the anchors in both regions meet the position resolution requirements, the anchors from both regions will be used to calculate the tag's position. However, this can cause inconsistent positioning in the transition region of the positioning

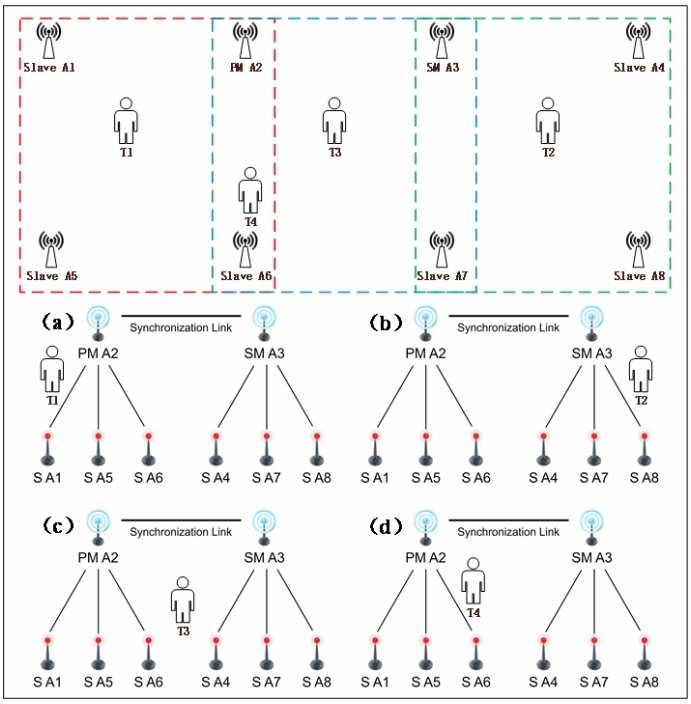


Fig. 5: Time-base selection algorithm for TDoA calculation.

boundary. Secondly, even within the same positioning region, the tag's positioning packet may be received by multiple anchors simultaneously. However, not all cases where more anchors are involved in the solution are better. When the number of anchors in the TDoA solution is greater than four, which is the minimum number of anchors required by TDoA, we select the four anchors that meet the solution requirements and have the smallest syncToA to solve the position. The goal of this section is to select four anchors that can optimize the positioning solution.

In a UWB-based positioning system, the transmitted signals between the transmitter and the receiver can be categorized into direct path signals in line-of-sight (LoS) situations and indirect path signals in non-line-of-sight (NLoS) and multipath (MP) situations, influenced by the ground and obstacles. This is illustrated in Figure 6.

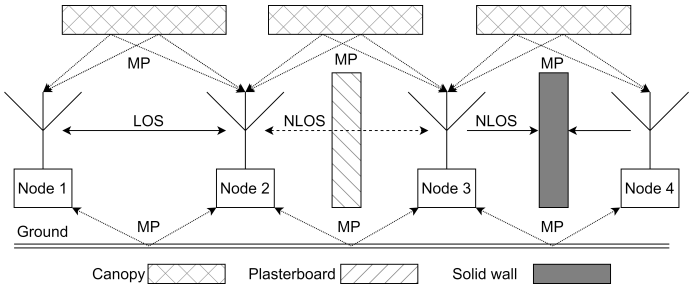


Fig. 6: Direct and indirect path signals in UWB.

We define the power received by the receiver from the transmitter as the 'received signal strength power' (P_{RSS}) and the power received by the receiver from the transmitter for the first path signal as the 'first path power' (P_{FP}). The P_{FP} and P_{RSS} values provided in the data manual are utilized to

TABLE I: Parameters' definitions of received UWB signals.

Parameter	Parameters's definitions in the register
F_1	First path amplitude (Point 1) magnitude value
F_2	First path amplitude (Point 2) magnitude value
F_3	First path amplitude (Point 3) magnitude value
N	Number of preamble symbols accumulated, or accumulated STS length
C	Channel Impulse Response Power value
D	DGC_DECISION, treated as an unsigned integer in range 0 to 7
A	Constant:
•	113.8 for a PRF of 16 MHz
•	121.7 for a PRF of 64 MHz Ipatov preamble
•	120.7 for a PRF of 64 MHz STS

determine whether the received signal meets the requirements for position calculation, based on the characteristics of the UWB RF chip. The definitions of P_{FP} and P_{RSS} are expressed as follows (in dBm) [6]:

$$P_{FP} = 10 \times log_{10}(\frac{F_1^2 + F_2^2 + F_3^2}{N^2}) + (6 \times D) - A \ dBm$$

$$P_{RSS} = 10 \times log_{10}(\frac{C \times 2^{21}}{N^2}) + (6 \times D) - A \ dBm$$

where the meanings of the parameters are given in Table I.

We clarify the receiving signal into LoS, NLoS, and the power attenuation according to the difference ϕ_{FR} between the P_{FP} and P_{RSS} [37].

$$\phi_{FR} = P_{FP} - P_{RSS}$$

$$Type = \begin{cases} LoS, & \text{for } \phi_{FR} > -6dB \\ Attenuation, & \text{for } -6dB > \phi_{FR} > -10dB \\ NLoS, & \text{for } -10dB > \phi_{FR} \end{cases}$$

In this work, we propose a positioning boundary optimization algorithm, detailed in Algorithm 4. This algorithm serves as the foundation for assessing whether the received signal is suitable for position calculation. It tallies the number of positioning packets when the tag is positioned at the border of two designated areas. The algorithm assesses the tag's location based on the value of ϕ_{FR}, subsequently determining the accurate location information of the tag.

To demonstrate the capability of our approach in calculating the correct location of the tag, we consider a use case scenario as shown in Figure 7. In this scenario, the tag is located at position T1. The tag sends blinks that can be received by all anchors in Cell-1 and Cell-2. However, a thin wall made of gypsum board exists between Cell-1 and Cell-2, causing signal attenuation when the tag's signal passes through the wall. As a result, the anchor in Cell-2 receives a weaker signal compared to the anchor in Cell-1. Consequently, the position of T2 calculated in Cell-2 will be discarded. After wireless clock synchronization, six anchors in Cell-1 receive the blinks and obtain six ToAs. Among these ToAs, the ToAs of anchors A2, A3, A4, and A5 are used to calculate the position of the tag. As expected, the final output result is the position of T1.

Algorithm 4 Positioning boundary optimization algorithm

Input:
P_{FP}, P_{RSS}, synchronized $ToAs$
Output:
Locations of the tag, $Coordinates(x, y)$

Collect the received $ToAs$
Calculate the P_{FP} of each ToA:
if *the value of* P_{FP} *satisfy FP rejection* **then**
  Deal with the received $ToAs$
if $ToAs$ *belong to the same positioning cell* **then**
  Count the number of $ToAs$
else if *the number of* $ToAs >= 4$ **then**
  Select 4 smallest synchronized $ToAs$
  Calculate the $TDoAs$ and estimate the locations
if $ToAs$ *belong to the different positioning cells* **then**
  Analyze the P_{FP} and P_{RSS}
  Divide $ToAs$ into different positioning cells
  Calculate each cell's maximal P_{RSS}
  Compare each cell's maximal P_{RSS}
  Choose the cell with maximal P_{RSS}
  Count the number of $ToAs$
else if *the number of* $ToAs >= 4$ **then**
  Select 4 smallest synchronized $ToAs$
  Calculate the $TDoAs$ and estimate the locations
else
  Continue and output the locations using $coordinate(x, y)$
end
$i = 0$
for $coordinate(x, y)$ in $coordinates(x, y)[]$ **do**
  $coordinates(x, y)[i] = coordinate(x, y)$
  $i++$
  if $i == 10$ **then**
    Using smoothing algorithms
    $i = 0$
end for
Return the $coordinates(x, y)^*$ after smoothing

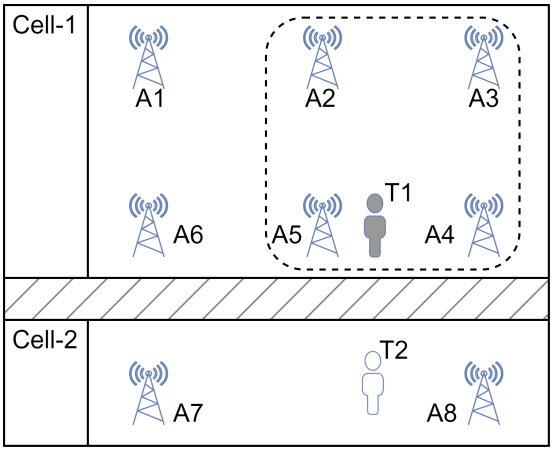


Fig. 7: Positioning boundary optimization method.

C. Complexity Analysis of TDoA-based Positioning

The mathematical model of TDoA is illustrated in Figure 8. We could get the timestamp τ_i when the anchor receives the signal, we could further assume that the measurement of $\hat{\tau}_i$ satisfies $\hat{\tau}_i \sim N(\tau_i, \delta_i^2)$. We set $d_i = \|p - p_i\|$ are the distances of anchors from the target. The core formula of TDoA is

$$\hat{d}_{ij} \triangleq \hat{d}_i - \hat{d}_j = c(\hat{\tau}_i - \hat{\tau}_j) := c\hat{\tau}_{ij}, \quad \forall i, j = 1, 2, \ldots, n, \quad (9)$$

where c is the velocity of the electromagnetic wave. In this way, we represent range differences (RD) by the TDoA measurements, the distance $\hat{d}_{ij}$ of here satisfy $\sum_{i,j} \hat{d}_{ij} \equiv 0$.

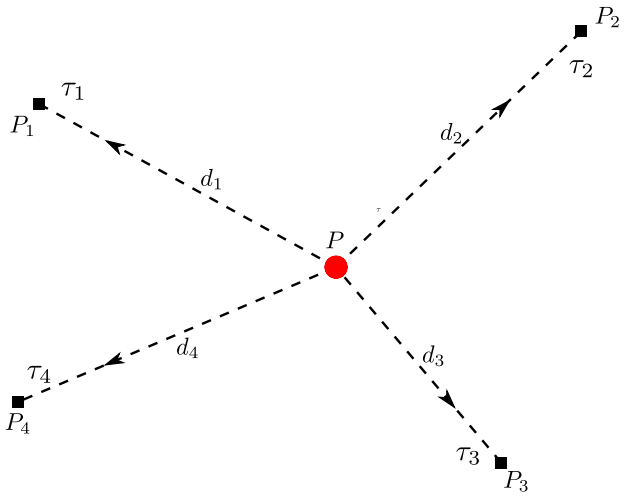


Fig. 8: The mathematical model of TDoA.

TDoA algorithms aim to find p, such that

$$d_{ij}(p) = \hat{d}_{ij}, \qquad \forall\ i, j = 1, 2, \ldots, n, \quad (10)$$

where $d_{ij}(p) = d_i(p) - d_j(p) = \|p - p_i\| - \|p - p_j\|$.

All above, the inputs of TDoA positioning algorithms are the anchors' coordinates p_i and the measured TDoA $\hat{\tau}_{ij}$, and the output p is the tag's coordinates. In practice, due to the measurement error, the equal sign is usually not satisfied in the previous formula. So the mathematical model for solving the non-linear estimator is as follows,

$$\underset{p}{argmin}\ \frac{1}{2}\|r\|^2, \quad r = [d_{ij} - \hat{d}_{ij}]_{ij}$$

When the iterative algorithm is adopted, the Jacobi matrix $\mathbf{J}$ of the residual term r has an important effect on the error and stability of the algorithm, namely

$$\text{Error of } p = \sigma^{-1}(\mathbf{J}) \cdot \text{Error of } f.$$

In general, we want the minimum eigenvalue of $\mathbf{J}$ to have a lower bound, so ensuring it's not rank deficient. Notice the form $\mathbf{J}(p)$,

$$\mathbf{J}(p) = \left[\frac{x - x_i}{d_i} - \frac{x - x_j}{d_j} \quad \frac{y - y_i}{d_i} - \frac{y - y_j}{d_j} \right]_{1 \le i < j \le n}$$

So we could get that,

$$\sigma_{\min}(\mathbf{J}(p)) = k(P(p))$$

Where, $P(p) = \left[p - p_1, p - p_2, \ldots, p - p_n\right]^T$. Note that, $\kappa(P(p))$ is the degree of dispersion with the center p and the coordinates $p_1, p_2, \ldots, p_n$. In this paper, we employ the extended Kalman filter (EKF) algorithm to determine the final position. The complexity analysis of the Kalman filter has been elucidated in Section III-A3; hence, it will not be reiterated here.

V. Evaluation and Experiments

A. Experimental Setup

1) UWB Platform: We have developed a UWB platform for a real-time positioning system using the DW1000 RF chip from DecaWave (now Qorvo) and an MCU from ST Microelectronics. This platform includes embedded hardware, embedded software, a central localization engine, and application software. The embedded hardware, comprising anchors and tags, was independently designed and prototyped, surpassing existing solutions in terms of signal transmission power and receiving sensitivity. It is important to note that we independently designed and developed all the necessary hardware, software, platforms, and algorithms for creating the real-time positioning system. Throughout the development process, from the initial prototype demonstration to the final product, we underwent several iterations to refine the system.

Each UWB anchor can be configured as a primary master anchor, a secondary master anchor, or a slave anchor based on the deployment situation. The embedded software follows the IEEE 802.15.4-2011 protocol standard for UWB signal transmission, and the software architecture, developed based on FreeRTOS, makes it easier to dispatch tasks. Additionally, the CCP sending control algorithm is integrated into the embedded software. The CLE integrates a database, algorithm processing unit, radio frequency (RF) parameter configuration unit, and log management unit. The algorithm unit implements the wireless clock synchronization algorithm and real-time position estimation algorithm. The application software allows for the configuration of the anchors' RF parameters and displays the tag's location and running track in real-time. To facilitate debugging, an efficient log system with different levels was designed, providing rich log information for developers or users according to their needs.

2) Experimental Devices: For testing in real-world scenarios, we have prepared a Wi-Fi router, a laptop running Windows 11, sixteen tripods, sixteen UWB anchors, several power banks, and multiple UWB tags. The anchors can transmit messages using a wired Ethernet or Wi-Fi infrastructure, simplifying the deployment process by minimizing the requirements for internet connectivity and power supply considerations.

3) Ground Truth: We have equipped a laser rangefinder and two tape measures: one with an effective length of 50 m and the other with a length of 10 m. After conducting initial measurements and calibration, we position anchors at specified locations. Simultaneously, we use the rangefinder to calibrate multiple reference positions within the coverage area of the real-time positioning system, establishing them as ground truth. Additionally, the intelligent vehicle used in

our experiment, equipped with the Robot Operating System (ROS) and LiDAR, possesses centimeter-level positioning and mapping capabilities. This enables us to obtain real-time location information from the intelligent vehicle, serving as ground truth for tag tracking.

B. Evaluation of Cascaded Wireless Clock Synchronization

To validate the performance of cascaded wireless clock synchronization and real-time positioning, we devised an experimental setup within an underground parking slot (Figure 9). In the case of multi-master cascading, it is crucial to ensure seamless communication between the cascaded master anchors and the slave anchors encompassed by them. Additionally, anchors are strategically deployed based on the dilution of precision (DOP) [38] along both axes. The parking slot exemplifies typical scenarios for indoor positioning, rendering it suitable for the assessment of our system and algorithms.

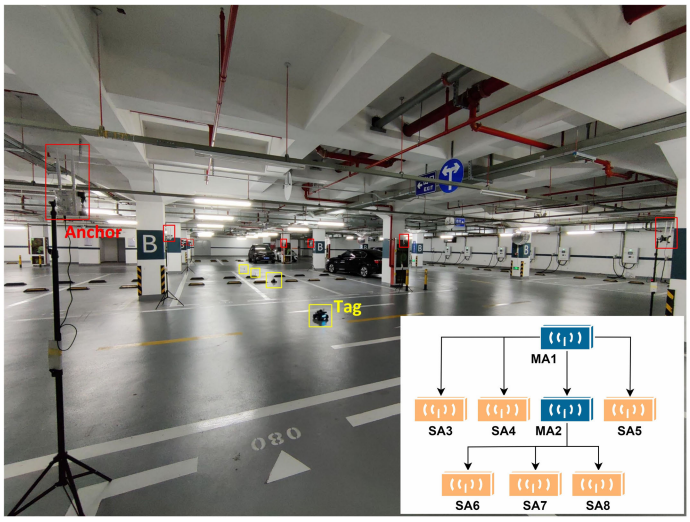


Fig. 9: Anchor deployment in the parking slot.

TDoA positioning is sensitive to clock drift, and the rates of clock synchronization and positioning updates significantly impact the system's performance. We conducted experiments in an area with dimensions of $10 \times 20\ m^2$ using an intelligent vehicle based on the ROS platform. The vehicle was equipped with a UWB tag on top, transmitting UWB signals to the anchors strategically placed around the deployment area.

It is imperative to elucidate the mechanism through which the UWB anchor attains sub-nanosecond accuracy. In the wireless clock synchronization algorithm delineated in this paper, a master anchor and several slave anchors play pivotal roles. The primary objective of clock synchronization is to align the local clock of each slave anchor with that of the master anchor. We conduct statistical analyses based on timestamps documented in the log file, with the outcomes presented in Table II. The timestamps recorded by the master anchor serve as the reference value; the accuracy of clock synchronization can be assessed by calculating the difference between the measured TDoAs recorded by the master and slave anchors and the corresponding true acquired TDoAs through distance difference extrapolation. This comparison enables an evaluation of the precision of clock synchronization. In conclusion, the proposed wireless clock synchronization algorithm demonstrates the capability to achieve clock synchronization accuracy at the sub-nanosecond level.

The results of the CWCS scheme with multiple master anchors are depicted in Figure 10. The Time of Arrival (ToA) recorded by each anchor was synchronized using the CWCS algorithm, and the difference between the synchronized ToAs was computed to derive the Time Difference of Arrival (TDoA) through the time-base selection algorithm. Figures 10 (a), (b), (c), and (d) illustrate the clock synchronization results obtained by the PID algorithm, LI algorithm, KF algorithm, and LI-KF algorithm, respectively. In comparison to the other three algorithms, the LI-KF algorithm more effectively compensates for synchronization errors during the synchronization period and eliminates the influence of outliers. This ensures a more accurate estimation of the tag positions.

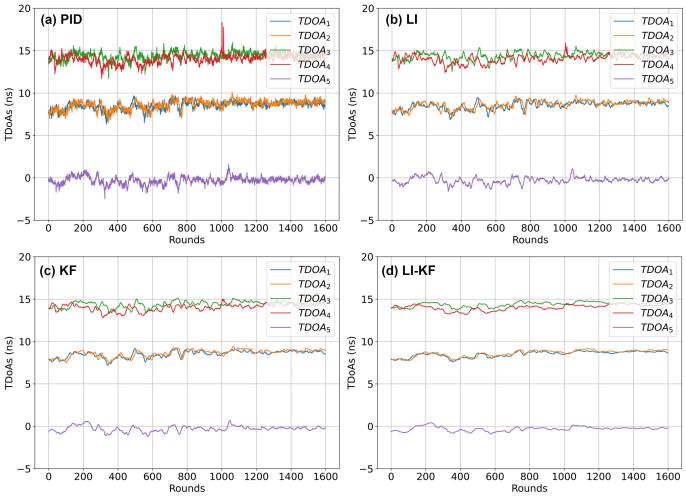


Fig. 10: Performance of CWCS: (a) PID-algorithm, (b) LI-algorithm, (c) KF-algorithm, (d) LI-KF algorithm.

To offer a more comprehensive comparison of the performance across different synchronization intervals and wireless clock synchronization algorithms, we conducted additional analyses. Upon reviewing the data presented in Table V, it becomes apparent that an increase in the clock synchronization rate corresponds to a decrease in synchronization error. This trend holds true for various clock synchronization algorithms. The underlying reason is that, as the clock synchronization intervals become shorter, the offset and drift of the local clock decrease, leading to higher precision in clock synchronization. It is worth noting that a higher synchronization rate implies an increased communication frequency between anchors. In our proposed TDOA-based positioning system, each anchor must simultaneously receive and process signals from both the tag and the main anchor. Handling a greater number of clock synchronization signals within a given time slot inevitably reduces the capacity to process positioning signals. To strike a balance between tag capacity and clock synchronization accuracy, we have chosen 150 ms as the interval of clock synchronization.

TABLE II: Sub-nanosecond accuracy of synchronized UWB Anchors

Anchor	Raw-timestamp (s)	Sync-timestamp (s)	Measured TDoA with MA1	Real TDoA with MA1	Accuracy
MA1	1.698929664123535e+01	1.6989296641235352325e+01	0 ns	0 ns	0 ns
SA2	1.709112450334598e+01	1.6989296651647475044e+01	10.41212 ns	10.57234 ns	0.16022 ns
SA3	1.718919632014348e+01	1.6989296654589647772e+01	13.35429 ns	13.18324 ns	0.17105 ns
SA4	1.697634507344564e+01	1.6989296647589974043e+01	6.35462 ns	6.53217 ns	0.17755 ns
SA5	1.688929664123535e+01	1.6989296657678647632e+01	16.44329 ns	16.26317 ns	0.18012 ns
SA6	1.709275641462315e+01	1.6989296649599873042e+01	8.36451 ns	8.20135 ns	0.16315 ns

TABLE III: RMSE of different synchronization algorithms at different synchronization intervals

Sync interval	LI	PID	KF	LI-KF
50 ms	0.22 ns	0.25 ns	0.19 ns	0.15 ns
150 ms	0.25 ns	0.30 ns	0.21 ns	0.17 ns
200 ms	0.31 ns	0.38 ns	0.26 ns	0.21 ns
300 ms	0.39 ns	0.48 ns	0.33 ns	0.27 ns
500 ms	0.50 ns	0.61 ns	0.42 ns	0.36 ns

C. Evaluation of tag densities

The solution presented in this paper is applicable to scenarios involving multiple tags. We will assess the tag densities of the real-time positioning system in terms of positioning success rate and positioning frequency. We positioned eight tags within the laboratory, as illustrated in Figure 11. The real-time positioning results for the eight tags in the deployed scenario are displayed in Figure 12. We conducted five sets of positioning experiments on these tags, configuring the localization frequency for each group of experiments to be 5 Hz, 10 Hz, 20 Hz, 50 Hz, and 100 Hz, respectively. We observed and recorded the success rates of tag positioning in each experimental group within one hour, and the corresponding statistical data are presented in Table IV. The data in the table clearly illustrate that the real-time positioning system, built upon the proposed solution, consistently attains a nearly perfect positioning success rate of almost 100% when the tag positioning frequency is set to 5 Hz. Even at a higher positioning frequency of 20 Hz, the system maintains a comprehensive positioning success rate of over 95%. Remarkably, even with a positioning frequency as high as 100 Hz, the system still achieves a comprehensive positioning success rate exceeding 80%.

Fig. 11: Tags in the laboratory.

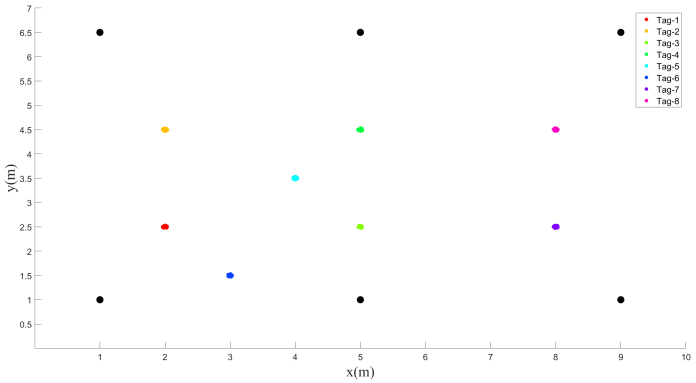


Fig. 12: Real-time positioning of tags.

TABLE IV: Success rates of tag positioning at different localization frequencies

Label	5 Hz	10 Hz	20 Hz	50 Hz	100 Hz
Tag-1	100%	97.6%	94.7%	90.5%	81.2%
Tag-2	99.7%	98.5%	95.1%	91.1%	79.8%
Tag-3	100%	96.4%	95.7%	89.8%	79.9%
Tag-4	99.8%	95.8%	95.3%	91.3%	81.0%
Tag-5	100%	96.7%	94.5%	90.6%	82.1%
Tag-6	100%	97.2%	94.9%	89.7%	80.5%
Tag-7	99.9%	98.3%	95.8%	91.8%	81.3%
Tag-8	100%	98.1%	96.1%	92.2%	82.2%

D. Evaluation of Positioning Boundary Optimization Algorithm and Real-time Positioning

We use typical examples to verify the performance of the cascaded wireless clock synchronization method, positioning boundary optimization algorithm, and real-time positioning accuracy. The ROS platform used in these experiments consists of two four-wheel intelligent vehicles. A remote controller is used to control the vehicles via Bluetooth communication. The extended Kalman filter (EKF) algorithm is employed to achieve car tracking and static tag positioning. The detailed implementation process of the algorithm is described in the author's previous work [39]. In the experiments, the original Time of Arrivals (ToAs) are recorded by all anchors. Firstly, we use the wireless clock synchronization algorithm to calibrate the original ToAs. Secondly, we calculate the corresponding Time Differences of Arrival (TDoAs) according to the time-base selection strategy. Thirdly, we enhance the positioning system by incorporating the cascaded wireless clock synchronization method and the localization boundary optimization algorithm. Finally, we evaluate the accuracy of

the real-time positioning system using the extended Kalman filter.

We have conducted validations in three distinct test scenarios. The first scenario involves a stringent Line of Sight (LoS) case between anchors. In the second scenario, anchors in different areas operate within a complete Non-Line of Sight (NLoS) situation. The third scenario entails a mixed case where the anchors experience a combination of LoS and NLoS conditions. In the experiments presented in this paper, the influence of tag movement speed on positioning accuracy will be discussed in Section V-D1, and therefore, this aspect will not be revisited in other experiments.

1) LoS case: We first consider the stringent Line-of-Sight (LoS) case, the experimental scenario shown in Figure 9 (which has already been mentioned when verifying the cascaded wireless clock synchronization algorithm in Section V-B). We remotely control the intelligent vehicle to move along a preset route within the positioning area, acquiring real-time location information of the tag in the process. Additionally, this intelligent vehicle can adjust its driving speed, allowing us to assess the impact of the tag's movement speed on positioning accuracy. To conduct this verification, we programmed the intelligent vehicle to travel at speeds of 1m/s, 1.5m/s, 2m/s, and 2.5m/s. We then recorded the real-time positions of the tag as obtained by the vehicle at each of these varying speeds. The resulting trajectory of the vehicle, depicted in Figure 13, illustrates the path taken based on the tag's position data.

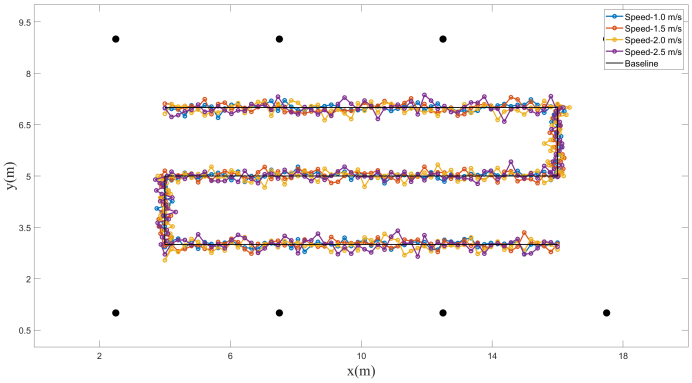


Fig. 13: Actual positioning result.

In this experiment, the cumulative distribution function (CDF) of the positioning error at different movement speeds is calculated and illustrated in Figure 14. During this test, the intelligent vehicle runs at a speed of 1 m/s, with 90% of the observed data errors being less than 6 cm. At a speed of 1.5 m/s, this value increases to 8 cm, at 2 m/s, it extends to 10 cm, and at 2.5 m/s, it reaches 12 cm. The maximum recorded observation error is 20 cm.

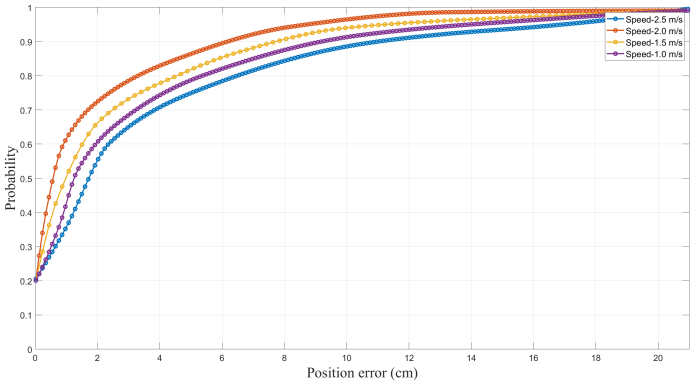


Fig. 14: CDF of positioning error.

2) NLoS case: In NLoS scenarios within an office space, we strategically deploy a total of 15 anchors. This anchor deployment effectively covers three distinct areas, including independent rooms and connecting corridors. Specifically, each room accommodates four anchors, while three anchors are strategically positioned within the corridors. It is noteworthy that these regions are deliberately chosen to be non-line-of-sight from each other, as illustrated in Figure 15. Simultaneously, we position three tags within the corridors, distributing the remaining tags across the designated rooms. Notably, within one of the rooms, the tags are affixed to an intelligent vehicle. Following the execution of the positioning system, it captures and records the real-time locations of the tags, as exemplified in Figure 16.

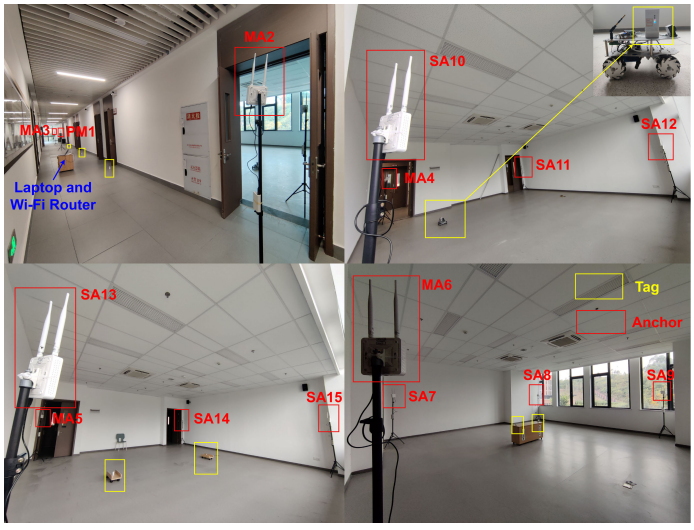


Fig. 15: Anchor deployment in the office space.

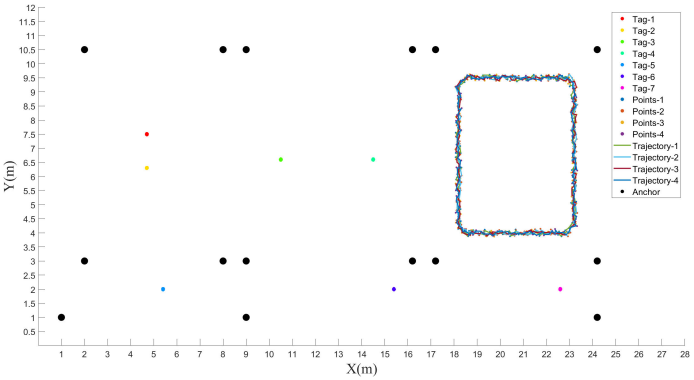


Fig. 16: Results of real-time positioning in NLoS case.

3) The mixed case of LoS and NLoS: We conducted experiments in a real-world location scenario depicted in Figure 17, which involved a mixed case of Line of Sight (LoS) and Non-Line of Sight (NLoS). We positioned two anchors in the corridor just outside the laboratory and placed an additional six anchors within the experimental area. The intelligent vehicle completed four circuits around the table within the laboratory and then traveled a distance within the corridor at a speed of 1.5 m/s. The outcomes of the experiments are presented in Figure 18. These experimental results clearly demonstrate the capability of our algorithm to effectively manage location boundaries.

Fig. 17: Anchor deployment in the laboratory.

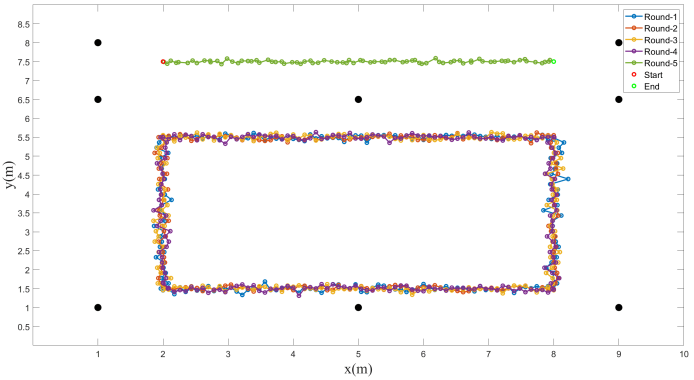


Fig. 18: Results of real-time positioning in mixed case of LoS and NLoS.

4) Comparison of positioning accuracy in different scenarios.: We assessed tag positioning accuracy under static conditions and tag tracking across three experimental scenarios, with the statistical results presented in Table V. In static conditions, the tag's positioning accuracy ranged from 5 to 10 cm, while for tag tracking, it varied between 10-20 cm, 12-20 cm, and 10-18 cm across different scenarios. It is crucial to emphasize that when implementing the real-time positioning system based on the algorithm proposed in this paper, ensuring an adequate number of anchors in each independent area that meets the minimum position calculation requirements is imperative. Consequently, anchors within each independent room are maintained under Line of Sight (LoS) conditions. Notably, three states exist: LoS, Non-Line of Sight (NLoS), and a mixed state of LoS and NLoS between different rooms, explaining the reasons for consistent positioning accuracy across diverse scenes. Our work addresses fundamental issues related to clock synchronization among anchors in distinct rooms and the optimization of localization boundaries between these rooms.

TABLE V: Comparison of positioning accuracy in different scenarios.

Status	LoS	NLoS	Los and NLoS
Static	5 – 10 cm	5 – 10 cm	5 – 10 cm
Tracking	10 – 20 cm	12 – 20 cm	10 – 18 cm

E. Comparison Between Different Positioning Systems

When compared with other state-of-the-art wireless TDoA systems, our overall results are put into perspective by Table VI. The TDoA positioning system proposed in PnPLoc and TALLA boasts the broadest coverage, spanning approximately 2000 m^2. In contrast, the positioning systems delineated in CHORUS and SnapLoc are confined to areas of less than 100 m^2 and fail to achieve superior positioning accuracy in scenarios involving LoS transmission. PnPLoc excels in achieving high accuracy when anchors transmit signals through NLOS conditions, with scalability to enhance accuracy by incorporating additional anchors. Meanwhile, TALLA attains an impressive accuracy range of 69cm-89cm within the anchor deployment area, specifically in corridors.

In comparison to the aforementioned positioning systems, the TDoA system implemented in this study ensures that each room is equipped with a sufficient number of anchors (meeting the condition of LoS between anchors) according to the characteristics of the positioning scenario. Addressing both LoS and NLoS conditions between different rooms, the proposed algorithm and optimized anchor deployment scheme synergistically contribute to guarantee positioning accuracy.

TABLE VI: Comparison of positioning error between our system and other state-of-art solutions.

Solution	75%	90%	Test area
Our system	5 – 10 cm	12 – 20 cm	90 – 336 m^2
PnPLoc [24]	28.9 – 110 cm	40 – 251 cm	695 – 2241 m^2
SnapLoc [25]	55.7 – 74 cm	NA	30.36 – 60.5 m^2
CHORUS [21]	$\approx 80cm$	$\approx 100cm$	42.0 – 83.2 m^2
TALLA [20]	69 cm	89 cm	1875 m^2

VI. Conclusion and Discussion

We presented a comprehensive solution that combines UWB and TDoA-based positioning with the Cascaded Wireless Clock Synchronization (CWCS) algorithm to address the challenges of actual deployment. Our contributions include designing an improved CWCS algorithm to support multiple master anchors in cascaded anchor deployment scenarios. We proposed a control algorithm for transmitting clock calibration packets through wireless broadcast to avoid signal collisions. Additionally, we defined a time-based ToA selection strategy for TDoA calculation to ensure the accuracy of tag tracking. We also introduced a positioning boundary optimization method based on received signal strength power and first path power in the selection of tags involved in positioning in multi-room scenarios. The effectiveness of these findings was demonstrated through simulations and real experiments.

While the real-time positioning system presented in this paper demonstrates exceptional positioning performance in tested scenarios, certain limitations persist. Primarily, UWB signal transmission experiences attenuation in absolute NLoS environments, significantly impacting positioning accuracy. Relying solely on UWB technology proves insufficient for effective positioning, necessitating the integration of complementary technologies to enhance positioning capabilities. Additionally, challenges arise in large-scale deployment scenarios, where the installation of UWB anchors and positioning system deployment encounters obstacles. Addressing the intricacies of cascaded clock synchronization in massive deployment scenarios constitutes an ongoing focus of our research efforts.

The DW1000 chip utilized in this paper adheres to the standards of IEEE 802.15.4-2011 and IEEE 802.15.4-2015. The recently established IEEE 802.15.4z-2020 standard introduces new features built upon the original standard, offering enhanced security, lower power consumption, and increased transmission distance. Notably, the new standard, which incorporates TDoA, AoA, and PDoA positioning methods, merits careful consideration for future research. Moreover, in our upcoming studies, we intend to delve into the intricate relationship between the cascaded wireless clock synchronization model and synchronization rates in large-scale deployment scenarios.

Acknowledgment

This research is supported in part by the National Natural Science Foundation of China (Grant No. 62211530106), and in part by Shenzhen Science and Technology Program (Grant No. ZDSYS20210623092007023, GJHZ20210705141808024).

References

[1] F. Zafari, A. Gkelias, and K. K. Leung, "A survey of indoor localization systems and technologies," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2568–2599, 2019.

[2] M. Ridolfi, A. Kaya, R. Berkvens, M. Weyn, W. Joseph, and E. D. Poorter, "Self-calibration and collaborative localization for uwb positioning systems: a survey and future research directions," *ACM Computing Surveys (CSUR)*, vol. 54, no. 4, pp. 1–27, 2021.

[3] A. Pughat and V. Sharma, "Performance analysis of an improved dynamic power management model in wireless sensor node," *Digital Communications and Networks*, vol. 3, no. 1, pp. 19–29, 2017.

[4] T. Yang, A. Cabani, and H. Chafouk, "A survey of recent indoor localization scenarios and methodologies," *Sensors*, vol. 21, no. 23, p. 8086, 2021.

[5] F. C. Commission, "Revision of part 15 of the commission's rules regarding ultra wideband transmission systems," vol. 65, no. 115, 2002. [Online]. Available: https://www.fcc.gov

[6] Decawave, "Dw1000 user manual—dw1000 ieee802.15.4-2011 uwb transceiver (user manual for dw1000)," 2017. [Online]. Available: https://www.decawave.com/dw1000/usermanual/

[7] F. Dong, C. Shen, J. Zhang, and S. Zhou, "A tof and kalman filtering joint algorithm for ieee802.15.4a uwb locating," in *2016 IEEE Information Technology, Networking, Electronic and Automation Control Conference*, 2016, pp. 948–951.

[8] Y. Jiang and V. C. Leung, "An asymmetric double sided two-way ranging for crystal offset," in *2007 International Symposium on Signals, Systems and Electronics*, 2007, pp. 525–528.

[9] F. Sun, H. Yin, and W. Wang, "Finite-resolution digital receiver for uwb toa estimation," *IEEE Communications Letters*, vol. 16, no. 1, pp. 76–79, 2012.

[10] M. Bocquet, C. Loyez, and A. Benlarbi-Delai, "Using enhanced-tdoa measurement for indoor positioning," *IEEE Microwave and Wireless Components Letters*, vol. 15, no. 10, pp. 612–614, 2005.

[11] N. Smaoui, M. Heydariaan, and O. Gnawail, "Single-antenna aoa estimation with uwb radios," in *2021 IEEE Wireless Communications and Networking Conference (WCNC)*, 2021, pp. 1–7.

[12] F. Ge and Y. Shen, "Single-anchor ultra-wideband localization system using wrapped pdoa," *IEEE Transactions on Mobile Computing*, pp. 1–1, 2021.

[13] Y.-C. Wu, Q. Chaudhari, and E. Serpedin, "Clock synchronization of wireless sensor networks," *IEEE Signal Processing Magazine*, vol. 28, no. 1, pp. 124–138, 2011.

[14] C. McElroy, D. Neirynck, and M. McLaughlin, "Comparison of wireless clock synchronization algorithms for indoor location systems," in *IEEE International Conference on Communications Workshops (ICC)*, 2014, pp. 157–162.

[15] S. Leugner, M. Pelka, and H. Hellbrück, "Comparison of wired and wireless synchronization with clock drift compensation suited for u-tdoa localization," in *2016 13th Workshop on Positioning, Navigation and Communications (WPNC)*, 2016, pp. 1–4.

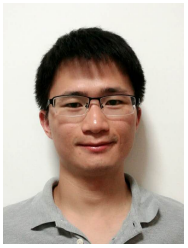
[16] J. Tiemann, F. Eckermann, and C. Wietfeld, "Multi-user interference and wireless clock synchronization in tdoa-based uwb localization," in *2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2016, pp. 1–6.

[17] B. You, X. Li, and W. Liu, "A kalman-filter-based wireless clock synchronization method in indoor localization," in *Eighth International Conference on Digital Image Processing (ICDIP 2016)*, vol. 10033. SPIE, 2016, pp. 1185–1190.

[18] J. Cano, S. Chidami, and J. L. Ny, "A kalman filter-based algorithm for simultaneous time synchronization and localization in uwb networks," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 1431–1437.

[19] V. Djaja-Josko and J. Kolakowski, "A new method for wireless synchronization and tdoa error reduction in uwb positioning system," in *21st International Conference on Microwave, Radar and Wireless Communications (MIKON)*, 2016, pp. 1–4.
[20] D. Vecchia, P. Corbalán, T. Istomin, and G. P. Picco, "Talla: Large-scale tdoa localization with ultra-wideband radios," in *2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2019, pp. 1–8.
[21] P. Corbalán, G. P. Picco, and S. Palipana, "Chorus: Uwb concurrent transmissions for gps-like passive localization of countless targets," in *2019 18th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, 2019, pp. 133–144.
[22] Q. Shi, X. Cui, S. Zhao, S. Xu, and M. Lu, "Blas: Broadcast relative localization and clock synchronization for dynamic dense multiagent systems," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 56, no. 5, pp. 3822–3839, 2020.
[23] J. Friedrich, J. Tiemann, and C. Wietfeld, "Accurate multi-zone uwb tdoa localization utilizing cascaded wireless clock synchronization," in *2021 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2021, pp. 1–8.
[24] H. Chen and A. Dhekne, "Pnploc: Uwb based plug & play indoor localization," in *2022 IEEE 12th International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE, 2022, pp. 1–8.
[25] B. Großwindhager, M. Stocker, M. Rath, C. A. Boano, and K. Römer, "Snaploc: An ultra-fast uwb-based indoor localization system for an unlimited number of tags," in *Proceedings of the 18th International Conference on Information Processing in Sensor Networks*, 2019, pp. 61–72.
[26] J. Yang, B. Dong, and J. Wang, "Vuloc: Accurate uwb localization for countless targets without synchronization," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 6, no. 3, pp. 1–25, 2022.
[27] B. Xue, Z. Li, P. Lei, Y. Wang, and X. Zou, "Wicsync: a wireless multi-node clock synchronization solution based on optimized uwb two-way clock synchronization protocol," *Measurement*, vol. 183, p. 109760, 2021.
[28] J. J. Pérez-Solano, S. Felici-Castell, A. Soriano-Asensi, and J. Segura-Garcia, "Time synchronization enhancements in wireless networks with ultra wide band communications," *Computer Communications*, vol. 186, pp. 80–89, 2022.
[29] N. M. Senevirathna, O. De Silva, G. K. I. Mann, and R. G. Gosine, "Asymptotic gradient clock synchronization in wireless sensor networks for uwb localization," *IEEE Sensors Journal*, vol. 22, no. 24, pp. 24 578–24 592, 2022.
[30] S. Rinaldi, A. Musatti, A. Depari, P. Ferrari, A. Flammini, and E. Sisinni, "An experimental characterization of chain of plls for wired clock synchronization of uwb anchors for indoor location," in *2022 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, 2022, pp. 1–6.
[31] R. Zou, N. Wu, Z. Qu, and J. Chen, "Design and implementation of a high-precision wireless clock synchronization system based on uwb," in *2022 4th International Conference on Intelligent Control, Measurement and Signal Processing (ICMSP)*, 2022, pp. 1094–1099.
[32] S. Rinaldi, A. Depari, P. Ferrari, A. Flammini, E. Mondini, and E. Sisinni, "An experimental characterization of time synchronization in multiple uwb location cells," in *2022 IEEE International Symposium on Precision Clock Synchronization for Measurement, Control, and Communication (ISPCS)*, 2022, pp. 1–6.
[33] D. Chiasson, Y. Lin, M. Kok, and P. B. Shull, "Asynchronous hyperbolic uwb source-localization and self-localization for indoor tracking and navigation," *IEEE Internet of Things Journal*, vol. 10, no. 13, pp. 11 655–11 668, 2023.
[34] D. Allan, "Time and frequency (time-domain) characterization, estimation, and prediction of precision clocks and oscillators," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 34, no. 6, pp. 647–654, 1987.
[35] D. W. Allan, "Tutorial: Clock and clock systems performance measures," 1995.
[36] F. Zhang, S. Duan, and J. Tan, "Uwb-based wireless clock synchronization and real-time positioning in smart parking," *IEEE Sensors Journal*, 2023.
[37] C. L. Sang, B. Steinhagen, J. D. Homburg, M. Adams, M. Hesse, and U. Rückert, "Identification of nlos and multi-path conditions in uwb localization using machine learning methods," *Applied Sciences*, vol. 10, no. 11, 2020. [Online]. Available: https://www.mdpi.com/2076-3417/10/11/3980
[38] F. Zhang, H. Li, Y. Ding, S. Yang, and L. Yang, "Dilution of precision for time difference of arrival with station deployment," *Iet Signal Processing*, vol. 15, pp. 353–364, 2021.
[39] F. Zhang, L. Yang, Y. Liu, Y. Ding, S.-H. Yang, and H. Li, "Design and implementation of real-time localization system (rtls) based on uwb and tdoa algorithm," *Sensors*, vol. 22, no. 12, p. 4353, 2022.

Fengyun Zhang received a B.S. degree in electronic and information engineering and an M.S. degree in signal and information processing from Southwest University, Chongqing, China, in 2014 and 2017, respectively, and the Ph.D. degree in mechanics (intelligent manufacturing) from Southern University of Science and Technology, Shenzhen, China, in 2023. He is currently a senior lecturer at the College of Electronic and Information Engineering, Southwest University. His research interests include UWB-based indoor localization, wireless sensor networks, and industrial control network protocol reverse engineering.



Shengguang Hong is a PhD student in the Department of Computer Science and Engineering at Harbin Institute of Technology, Harbin, China. His research interests include wireless sensor network (WSN) and LoRa technology. He received his B.E. degree in 2012 with the major of mechanical engineering and automation from Chongqing Technology and Business University.

Yulong Ding received the B.A.Sc. and M.A.Sc. degrees in Chemical Engineering from Tsinghua University, China, in 2005 and 2008, respectively and the Ph.D. in Chemical Engineering from University of British Columbia, Canada, in 2012. He is currently a Research Assistant Professor with the Academy for Advanced Interdisciplinary Studies of Southern University of Science and Technology. His main interests are Industrial Internet of Things and Low-Power Wide Area Networks (LPWAN).

Shuanghua Yang received the B.S. degree in instrument and automation and the M.S. degree in process control from the China University of Petroleum (Huadong), Beijing, China, in 1983 and 1986, respectively, and the Ph.D. degree in intelligent systems from Zhejiang University, Hangzhou, China, in 1991. He was awarded DSc from Loughborough University in 2014 to recognize his academic contribution to wireless monitoring research. He is currently a chair professor of computer science with Southern University of Science and Technology (SUSTech), Shenzhen, China. Before joined SUSTech in 2016 he had spent over two decades in Loughborough University, as a professor in computer science and head of department. His current research interests include cyber-physical system safety and security, Internet of Things, wireless network-based monitoring and control. He is a Fellow of IET and a Fellow of InstMC, U.K. He is an Associate Editor of the IET Journal Cyber-Physical Systems Theory and applications, the InstMC Journal Measurement and Control, and the International Journal of Computing and Automation.