RESEARCH
Open Access

A real-time device-free localization system using correlated RSS measurements

Zhiyong Yang, Kaide Huang, Xuemei Guo and Guoli Wang[*]

Abstract

Device-free localization (DFL) with wireless sensor networks (WSN) is an emerging technology for target localization, which has received much attention in the area of Internet of Things. Received signal strength (RSS) measurements are the key to realize DFL and mainly affects the localization performance. Most existing approaches need to measure the RSS of all the wireless links in WSN, which take much time on measurement process and localization algorithm due to the large amounts of RSS data, thus they are inefficient, especially in the case of target tracking. In this paper, by making full use of the consecutiveness of motion, we present an efficient measurement strategy based on a small set of correlated wireless links. Furthermore, a lightweight compressed maximum matching select (CMMS) algorithm is proposed to localize target, which only needs a small-scale matrix-vector product operating for one estimation. The proposed approach can significantly reduce the number of RSS measurements and improve the real-time capability of the DFL system. Experimental results demonstrate the superior performance of the proposed method in the context of target localization and tracking.

Keywords: Internet of Things; Wireless sensor networks; Device-free localization; Compressed maximum matching select; Efficient measurement

1 Introduction

Internet of Things (IoT) concerns about the seamless interaction of objects, sensors, and computing devices [1]. With the integration of wireless sensor networks (WSN) and the Internet, the IoT is fast becoming a reality. IoT is applicable to various areas, including business logistics, home automation, and healthcare [2]. Tracking is an important aspect of the healthcare domain [3]. Device-free localization (DFL) [4] is an emerging method for localizing and tracking target with WSN, which does not need equipping the target with any wireless device. Hence, the DFL technology would not inconvenience the target or make it uncomfortable. The DFL also can be used in other applications such as intrusion detection, nighttime security monitoring, and emergency rescue, where the traditional localization scheme that target needs to equip with a wireless device to transmit or receive wireless signals will become invalid. The location information is extremely useful in these applications, as it may provide

life-saving benefits for the emergency responders. Therefore, the study on realizing efficient real-time DFL with WSN is necessary and significative.

There are lots of wireless links within the deployment area of the WSN. When an object moves into the area, it may shadow some of the links and reflect, absorb, diffract, or scatter some of the transmitted power, which will change the received signal strength (RSS) of the shadowed links. The object locating at a different location will shadow different links, so we can realize DFL based on RSS measurements. Since there are too many wireless links in a WSN, the measurements of all the wireless links will cause some disadvantages in resources consumption, system latency, and estimation processing. This article focuses on using efficient measurement strategy and lightweight algorithm to realize real-time DFL.

Wilson and Patwari [4-7] formulated the DFL as a radio tomography imaging (RTI) problem and utilized regularization method to solve it. Moussa and Youssef [8,9] modeled the DFL as a machine learning problem and adopted fingerprint-matching method to solve the problem. Zhang et al. [10-14] proposed a signal dynamic model and used the geometric method and probabilistic

*Correspondence: isswgl@mail.sysu.edu.cn
School of Information Science and Technology, Sun Yat-sen University, Guangzhou 510006, China

cover algorithm based on dynamic clustering to localize targets. All these works require sufficient RSS measurements of wireless links, which are inefficient, have high-resource consumption, and sometimes are not even possible. In this paper, based on the consecutiveness of motion, we propose an efficient measurement strategy that only needs to measure a few wireless links. Furthermore, we consider the information of target locations as a sparse signal and reconstruct it via compressive sensing (CS) method. CS has been applied to realize DFL in [15]; however, the convex optimization approach was adopted to reconstruct the sparse signal, which is computationally expensive. Meanwhile, the reconstruction is based on a randomly selected set of links, and this leads to inefficiency and low consistency. Wang et al. [16] presented a novel Bayesian greedy matching pursuit (BGMP) algorithm to solve the DFL problem based on the enumeration region built on prior information. However, the algorithm is still computationally expensive.

In this article, for realizing efficient real-time DFL system with wireless network, we first provide the radio tomography imaging model for relating the variances of RSS measurements of wireless links to the spatial locations of the targets. Based on the RTI model, we formulate the DFL issue as a sparse signal reconstruction problem. Then we propose a novel and efficient measurement strategy based on the correlated links which are determined with possible region built on the previous reconstruction. We also propose a compressed maximum matching select (CMMS) algorithm for fast reconstruction of the signal. It only utilizes RSS measurements of the correlated links and reconstructs the signal within a restricted subspace. Hence, the running time of the algorithm is reduced and the reconstruction performance is improved simultaneously, which completely meet the need for real-time DFL applications.

The remainder of this paper is organized as follows. In the next section, we will discuss some related works. Section 3 introduces the RTI model and formulates the DFL issue as a compressed RTI (CRTI) problem with CS theory. Section 4 presents the efficient measurement strategy and the CMMS algorithm. The experiments and results are showed in Section 5. Finally, we conclude the paper in Section 6.

2 Related works

Localization of targets based on received signal strength in WSN is a promising technique which has received extensive attention [17-21]. However, little attention has been given to the real-time CS-based DFL and efficient measurement method. In this section, we briefly summarize the most relevant research on the DFL and the CS-based DFL.

Device-free localization was first introduced by Zhang et al. [10] and Youssef et al. [9]. Zhang et al. [10-13] presented a dynamic model to describe the relation between the RSS variance and the target location, then utilized geometric method and the dynamic cluster-based probabilistic cover algorithm to solve the DFL problem. They also proposed a real-time DFL system [14]. They divided the tracking area into distinct sub-regions, with each region assigned with a separate radio channel, and used the support vector regression model to locate the target in each sub-region. Youssef et al. [8,9] adopted the fingerprint-matching method to realize DFL. The target's location was estimated by comparing the current RSS measurements with the trained database. Although the above methods achieve reasonable performance, they need to build a separate training measurement database before realizing the DFL. Training measurements increase exponentially with the increase of the number of wireless links and targets. Moreover, the database will be unavailable when the environment changed. Wilson and Patwari [5,6,22,23] firstly modeled the DFL as a RTI problem, then they carried out in-depth research on relating the temporal link signature with the target's location. They utilized the regularization method to solve the ill-posed inverse problem in the reconstruction of the radio tomographic image. Their studies laid the foundation for future research on the radio tomographic imaging and encouraged other researchers to start to work in this direction. Chen et al. [24] adopted an auxiliary particle filter to realize tracking of device-free target based on the RSS measurements. These works require that there should be sufficient number of wireless links be measured to guarantee the reconstruction performance, otherwise, the reconstruction performance will drop significantly. On the basis of these works, we formulate the DFL as a CRTI problem and propose an efficient measurement strategy and a lightweight algorithm to realize real-time tracking.

To the best of our knowledge, Kanso and Rabbat [15] adopted convex optimization algorithm to reconstruct the sparse image, which is the first work that adopts CS theory to solve the DFL problem. However, the computation complexity of the ℓ_1 minimization algorithm is too high and not suitable for wireless network. Wang et al. [16] also utilized the CS theory to solve the DFL problem. They limited the region where the target may be located with prior information of last reconstruction, then used the proposed BGMP algorithm to solve the simplified DFL problem. The BGMP algorithm iteratively seeks the contribution of each pixel in the restricted region and finally locates the target on the pixel which has the biggest contribution value. BGMP essentially is a fusion of the orthogonal matching pursuit (OMP) [25] algorithm and back-projection algorithm. However, the algorithm is also computationally expensive and not very suitable

for real-time localization and tracking of targets. Furthermore, the above works choose a set of wireless links to reconstruct the sparse signal and they still need to measure all of the wireless links.

In this paper, we proposed a novel efficient measurement strategy, which only needs to measure a small set of correlated links so it is efficient and energy saving. To our best knowledge, this is the first work which realizes efficient measurement in real-time DFL system. We also proposed a lightweight CMMS algorithm to localize target, which only needs a small-scale matrix-vector product and a sorting operation. Hence it runs fast and is suitable for real-time system.

3 Model and problem formulation

In this section, we provide a RTI model for relating the variance of the measured RSS value of each wireless links to the location of the target, and then we introduce the CS theory and formulate the DFL problem as a compressive RTI question.

3.1 System model

In a WSN, if the number of nodes is K, then the number of the unique two-way wireless links is $M = (K^2 - K)/2$. For simplicity, we illustrate a WSN in Figure 1 with all the 20 wireless nodes uniformly distributed in a square perimeter. When wireless nodes communicate, the RSS $y_i(t)$ of a particular link i at time t is denoted as

$$y_i(t) = P_i - S_i(t) - F_i(t) - L_i - v_i(t), \qquad (1)$$

where

- P_i is the transmitted power in decibel,
- $S_i(t)$ is the shadowing loss in decibel caused by the targets which attenuate the signal,
- $F_i(t)$ is the fading loss in decibel due to constructive and destructive interference of narrow-band signals in multi-path communication,
- L_i is the static loss in decibel due to antenna patterns, distance, and device inconsistencies,
- $v_i(t)$ is the measurement noise.

For two time instants t_a and t_b, the change of the RSS measurement $\triangle y_i$ is

$$\begin{aligned} \triangle y_i &= y_i(t_b) - y_i(t_a) \\ &= S_i(t_b) - S_i(t_a) + F_i(t_b) - F_i(t_a) + v_i(t_b) - v_i(t_a) \end{aligned} \qquad (2)$$

which can be rewritten as

$$\triangle y_i = \triangle S_i + n_i, \qquad (3)$$

where the noise is the grouping of fading and measurements

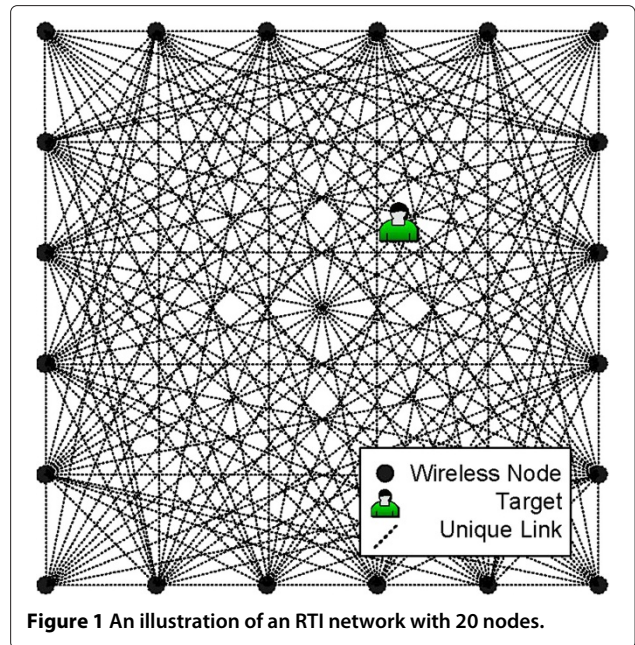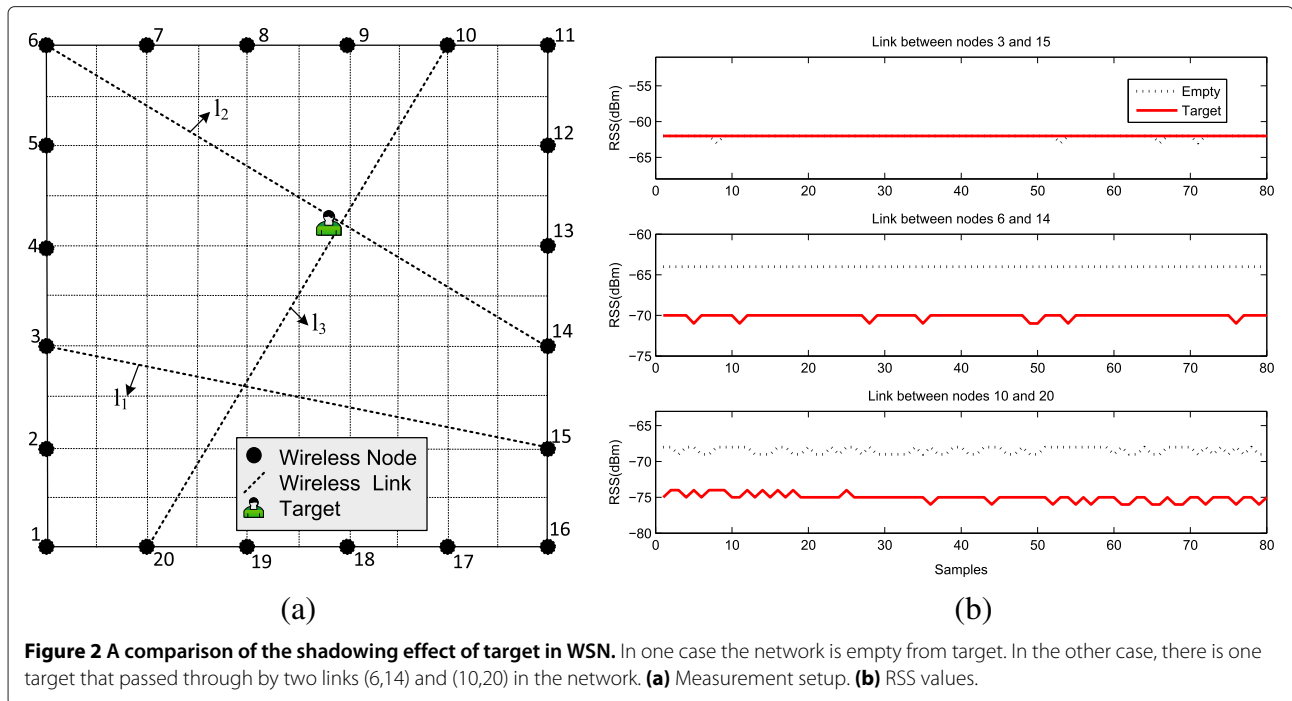$$n_i = F_i(t_b) - F_i(t_a) + v_i(t_b) - v_i(t_a). \qquad (4)$$

Figure 1 An illustration of an RTI network with 20 nodes.

As described in (3), we can see that $\triangle y_i$ is primarily determined by the shadowing loss difference of the two time instants. We examined the shadowing effect of the target on single link measurement. As seen in Figure 2, a target was located in the WSN, and we took the links l_1, l_2, and l_3 for comparison. Figure 2b shows the values of RSS measurement when a human stood in the position as showed in Figure 2a and when without any human in the deployment WSN area. We can see that if a wireless link is shadowed by a target, its RSS value will significantly change from when it is not shadowed. Hence we can use the shadowing model to realize DFL.

In the shadowing model, which is the most widely adopted model, the noise n_i is caused by time-varying measurements miscalibration of the receiver, by the contribution of thermal noise, and by the variations in the multipath channels. The statistics of the noise n_i has been examined in [5], which is constant with time. Hence, the calibration (when no moving targets existed in the wireless network field) could be able to establish it as the baseline. Then one can use the changes of RSS measurements to realize DFL.

Now we provide the RTI model. The monitoring area of the WSN is divided into N square cells which can be seen as pixels, as indicated in Figure 3. We use the word pixel to represent the cell in the following sections, and the location of each pixel is represented by the center point of the corresponding cell. The shadowing loss $\triangle y_i$ of link i can be approximated as a sum of attenuation that occurs in each pixel. Since the contribution of each pixel to the attenuation of each link is different, a weighting is

Figure 2 A comparison of the shadowing effect of target in WSN. In one case the network is empty from target. In the other case, there is one target that passed through by two links (6,14) and (10,20) in the network. **(a)** Measurement setup. **(b)** RSS values.

applied. Mathematically, this is described for single link i as

$$\triangle y_i = \triangle S_i + n_i = \sum_{j=1}^{N} w_{ij} x_j + n_i \tag{5}$$

where x_j is the attenuation occurring in pixel j, w_{ij} is the weighting of pixel j for link i, and N is the total number of pixels. The weighting w_{ij} can be calculated with ellipse model mathematically described as

$$w_{ij} = \frac{1}{\sqrt{d}} \times \begin{cases} 1, & \text{if } d_{ij}(1) + d_{ij}(2) \le d + \lambda, \\ 0, & \text{otherwise,} \end{cases} \tag{6}$$

where d is the distance between the two nodes of the link i, $d_{ij}(1)$ and $d_{ij}(2)$ are the distance from the center of pixel j to the two nodes, and λ is a tunable parameter describing the width of the ellipse. The width parameter λ is typically set very low in RTI, such that it is essentially the same as using the line-of-sight (LOS) model, as depicted in Figure 3.

Supposing that M links are adopted to realize RTI, the changes of RSS measurements can be expressed in matrix form as

$$\mathbf{y} = \mathbf{W}\mathbf{x} + \mathbf{n} \tag{7}$$

where the vector $\mathbf{y} = [\triangle y_1, \triangle y_2, \ldots, \triangle y_M]^T$ is a $M \times 1$ vector that represents the changes of the RSS measurements, $\mathbf{W} = \{w_{ij}, i = 1, 2, \ldots, M, j = 1, 2, \ldots, N\}$ is the $M \times N$ weighting matrix, $\mathbf{x} = [x_1, x_2, \ldots, x_N]^T$ is the unknown $N \times 1$ pixel vector to be reconstructed, and $\mathbf{n} = [n_1, n_2, \ldots, n_M]^T$ is the $M \times 1$ noise vector.

The weighting matrix can be calculated with (6). With sufficient link RSS measurements, we can reconstruct an image vector by solving the inverse problem in (7). The image vector describes the amount of radio power attenuation occurring due to the targets within the pixels of the WSN region. Since the pixel locations are known, RTI allows us to know where the attenuations in a WSN are occurring and, therefore, where the targets are located.

Figure 3 Illustration of the RTI elliptical weight model.
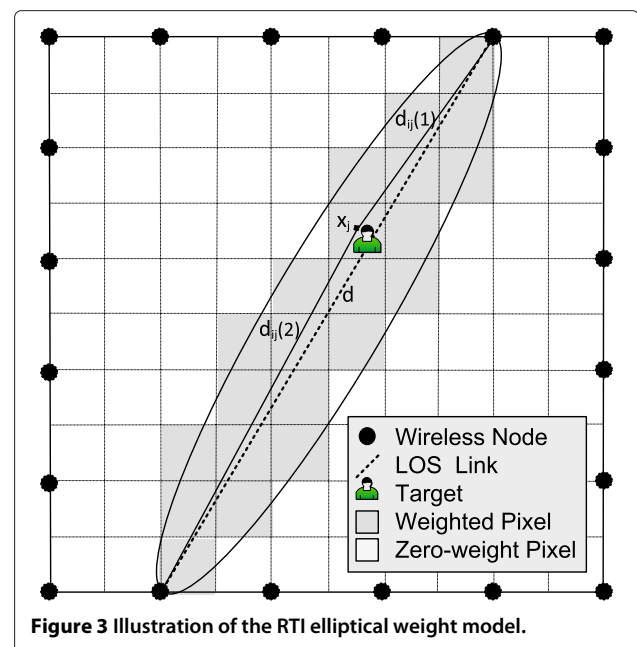
3.2 Formulation as CRTI problem

CS is an emerging theory for reconstructing sparse signals from a much lower sampling rate than Shannon/Nyquist theorem. In the deployment area of a RTI system, when a target moves into a pixel j, the pixel value x_j will be non-zero, otherwise, x_j will be zero. Note that after sufficient dense gridding, each target can be guaranteed to have a unique location in 1 pixel. In general DFL application, the number of targets K is considerably less than the number of pixels N. Hence, the image vector $\mathbf{x}$ will be a sparse signal, and it will be possible to reconstruct the $\mathbf{x}$ from a few measurements. This motivates us to utilize the CS theory to reconstruct the sparse signal based on (7). We will present a lightweight algorithm to solve the sparse signal reconstruction problem in the next section.

4 Efficient measurement strategy and CMMS algorithm

In this section, we describe the detailed implementation of the proposed real-time DFL system, including the efficient measurement strategy, the lightweight reconstruction algorithm CMMS, and the system scheme.

In a RTI system, if one directly use (7) to find the pixel where the target is located, almost all the links should be measured, which is time and power consuming, especially when N is large. As we know, the WSN generally is power-limited. To solve this problem, we propose utilizing prior information of last reconstruction to restrict the range of the pixels where the target may locate and to guide the next measurements. Once we know which links need to be measured, the radio of the other nodes which does not need to participate in the measurement can be turned off, hence, we can both reduce the latency of the system and save the power of the nodes.

As moving is consecutive, the target's current location must be around the last location. As illustrated in Figure 4, supposing that the target is located on x_j at time instant $t-1$, then at the next time instant t, the target must be located in a adjacent pixel of x_j (the shaded pixels in Figure 4), which is called as possible region in this paper. If the distance d_{ij} between pixel i and pixel j is less than the threshold l, then the pixel i is in the possible region. The parameter l should be set as $l = V_{\max} \times t_{\text{int}}$, where $V_{\max}$ is the maximum speed of the target, and t_{int} is the time interval between two successive running of the estimation algorithm. The wireless links that pass through the possible region are called as correlated links. We can only use the correlated links to localize the target in the possible region. Hence, we only need to measure the correlated links in the DFL system. Algorithm 1 presents the construction process of the possible region $\mathbf{R}$ and the correlated links of set $\mathbf{L}$.

Algorithm 1 Possible region and correlated links construction algorithm

---

Require:
    Index set $\mathbf{R}^{t-1}$ of pixels where targets located of the last time instant,
    Number of targets K,
   Measurement matrix $\mathbf{W}$,
    The threshold l

Ensure:
    The possible region $\mathbf{R}$,
    The correlated links set $\mathbf{L}$

1: **for** $j = 1$ to N **do**
2:     **for** $i = 1$ to K **do**
3:       **if** $d_{jp_i^{t-1}} < l$ **then**
4:         $\mathbf{R_i} \leftarrow \mathbf{R_i} \cup \{j\}$;
5:       **end if**
6:     **end for**
7: **end for**
8: **for** $i = 1$ to K **do**
9:     **for** $m = 1$ to M **do**
10:     **if** $\sum_{j \in \mathbf{R}_i} w_{ij} > 0$ **then**
11:       $\mathbf{L}_i \leftarrow \mathbf{L}_i \cup \{m\}$
12:     **end if**
13:     **end for**
14: **end for**

After calculating the set of links that needs to be measured, the host of the algorithm (generally a laptop) sends command messages to the wireless nodes through the base station node to tell them when to participate in the measurement and when to go to sleep. Therefore, in
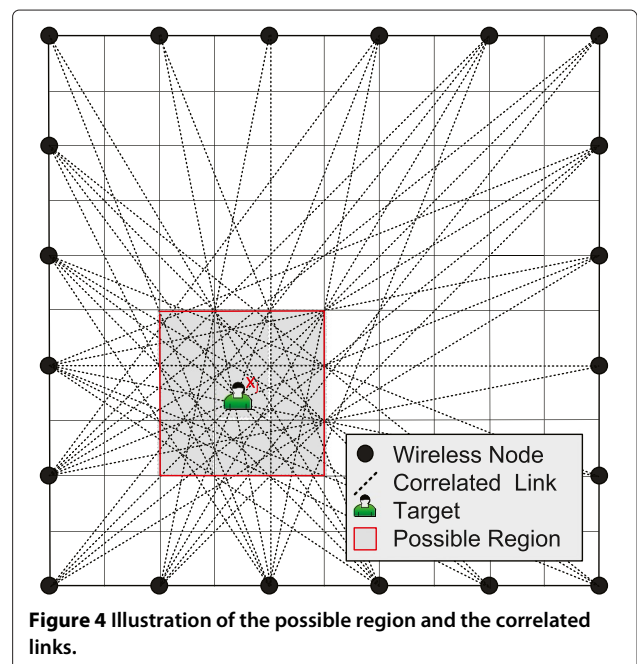
Figure 4 Illustration of the possible region and the correlated links.

the real-time system, the nodes have two phases: one for measurement and the other for receiving command messages from the host computer. In the phase of measurement, the nodes are synchronized and time slots are assigned by the following scheme. In 1 cycle of measurement, the total number of slots is the same as the number of the nodes. Each node is assigned to transmit only in the slot in which the serial number is equal to its node ID number; in other slots, the node either receives messages from another node for measurement the RSS of the link or keeps radio in sleep phase to save power. The packet transmitted by a node includes the ID of the node and the RSS values that it has already measured. In the measurement phase, the base station node receives all the data packets and sends them to the host computer. Hence, when the last node has sent its data packet, the host computer has all the RSS values of the correlated links. Then the wireless node will turn to the second phase. All of them will keep alive and receive command messages from the base station node. Simultaneously, the host computer will use the measured RSS vector to estimate the target's location with Algorithm 2.

Now, we only need to reconstruct the signal $\mathbf{x_R}$, which contains far less elements than the source signal $\mathbf{x}$. In addition, the magnitude of the measured vector $\mathbf{y_L}$ is also far
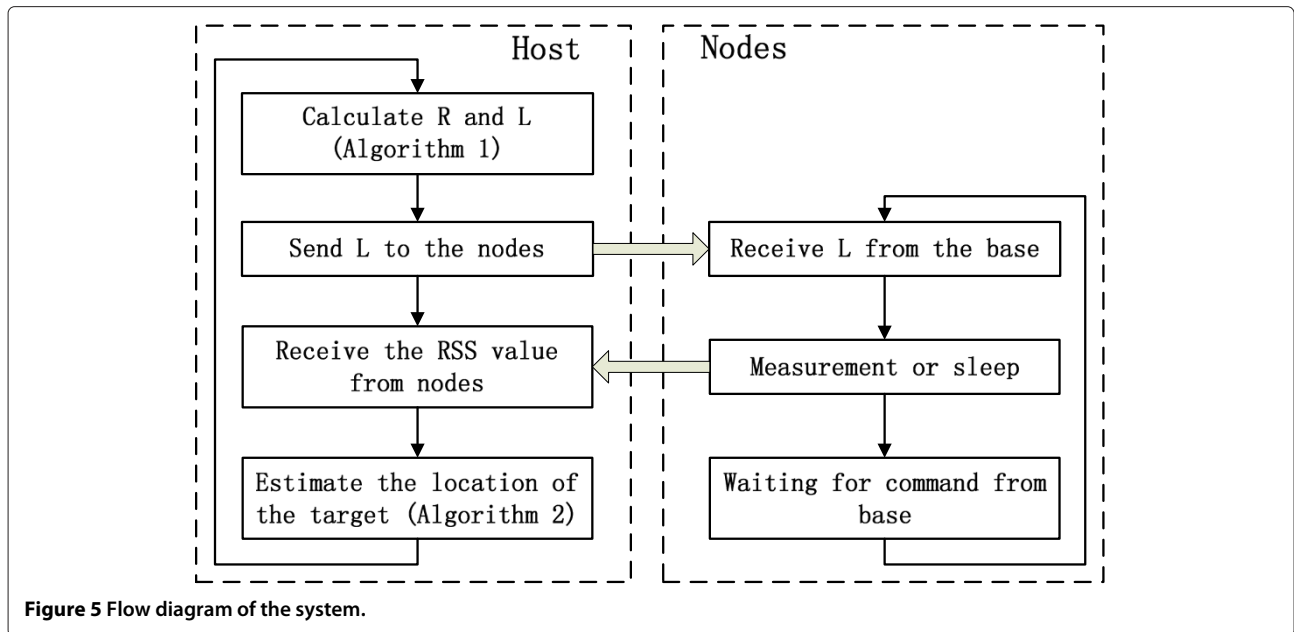

Algorithm 2 CMMS algorithm

Require:
  Measurement matrix $\mathbf{W}$,   Number of targets K
  RSS Measurement vector $\mathbf{y_L}$,   Possible region $\mathbf{R}$
Ensure:
  Index set $\mathbf{p}^t$ of pixels where targets located of the current time instant,
 Reconstructed signal $\mathbf{x}$
1: Initialization:
2:   $x_j \leftarrow 0, \ \forall j \in \{1, 2, \ldots, N\}$;
3: **for** $i = 1$ to K **do**
4:   $\mathbf{W_{L_iR_i}} \leftarrow \{w_{mj}, \forall m \in \mathbf{L}_i, \forall j \in \mathbf{R}_i\}$;
5:   $\mathbf{c} = \mathbf{W_{L_iR_i}}^T \times \mathbf{y_{L_i}}$;
6:   $p_i^t \leftarrow \ \arg\max_j |c_j|, j \in R_i, j \notin \{p_1^t, p_2^t, \ldots, p_{i-1}^t, p_{i+1}^t, \ldots, p_K^t\}$;
7: **end for**
8: **for** $i = 1$ to $K - 1$ **do**
9:   **for** $j = i + 1$ to K **do**
10:    **if** $p_i^t \in \mathbf{R}_j$ & $p_j^t \in \mathbf{R}_i$ **then**
11:     **if** $\cos(\overrightarrow{p_i^{t-2}p_i^{t-1}}, \overrightarrow{p_i^{t-1}p_i^t}) < \cos(\overrightarrow{p_i^{t-2}p_i^{t-1}}, \overrightarrow{p_i^{t-1}p_j^t})$ **then**
12:       $m \leftarrow p_i^t, \ p_i^t \leftarrow p_j^t, \ p_j^t \leftarrow m$;
13:     **end if**
14:    **end if**
15:   **end for**
16: **end for**
17: $x_{p_i^t} \leftarrow 1, \ \forall i \in \{1, 2, \ldots, K\}$

---

less than in the source vector $\mathbf{y}$. Hence, the reconstruction problem in (7) is compressed to

$$\mathbf{y_L} = \mathbf{W_{LR}} \times \mathbf{x_R} + \mathbf{n_L}. \tag{8}$$

We propose the CMMS algorithm to solve the above compressed problem. The pseudocode of the CMMS algorithm is summarized in Algorithm 2. For estimating the next location of target i, the CMMS algorithm only needs to find out the pixel j that could maximize $|\mathbf{W_{L_iR_i}}^T \times \mathbf{y_{L_i}}|$, where $\mathbf{R}_i$ and $\mathbf{L}_i$ are the possible region and the correlated links of the target i, respectively; $\mathbf{W_{L_iR_i}}$ is a sub-matrix of $\mathbf{W}$ that only includes the rows in $\mathbf{L}_i$ and columns in $\mathbf{R}_i$; and $\mathbf{y_{L_i}}$ is the measured RSS values of the correlated links $\mathbf{L}_i$. Since it only needs a small-scale matrix-vector product and a sorting operation, the CMMS algorithm is lightweight and has low complexity, which makes it meet the requirements of real-time systems. When two targets move too close together that it is difficult to distinguish from each other, the algorithm will make their trajectories as smoother as possible.

For clarity, the outline of the proposed real-time DFL method is summarized in Figure 5.

5 Experimental results

To evaluate the performance of the real-time DFL system, we conduct the real-time measurement and tracking experiment. In this section, we first describe our experimental setup. Then the tracking performance of moving targets is provided. Lastly, we will give some analyses and discussions.

5.1 Physical description of experiment

A wireless network containing 20 nodes was deployed in a laboratory. Each node is placed 1.0 m apart along the perimeter of 5×5 m^2 and 1.0 m off the ground on a tripod. A photograph of the experimental setup is shown in Figure 6. The network comprises MICAz wireless nodes [26] made by MEMSIC (MEMSIC Inc., Andover, MA, USA). Each node operates in the 2.4G frequency band and runs the IEEE 802.15.4 standard protocol for communication. A base station node listens all network traffic then feeds the data to a laptop computer via a USB port for processing. It also sends command messages to the wireless nodes from the laptop computer.

To avoid network transmission collisions, a simple token passing protocol is used. Each node is assigned an ID number from 1 to 20 and programmed with a known order of transmission. When a node transmits its packet, the other nodes that need to measure the RSS of the link will wake up to receive the message for acquiring the RSS value and then put it into its send buffer. When its turn to transmit arrives, the measured RSS values will be sent

Figure 5 Flow diagram of the system.

out. Since the base station node always receives all the packages after the last node transmitted its packet, the real-time tracking program running on the laptop will get all the needed RSS values.

When the base station node sends a message to the wireless nodes, its package follows a certain format. The base station node needs to send two kinds of messages to the wireless nodes: the first kind of messages tells the nodes when they need to wake up for receiving packages; we

call this kind of messages as *L message*. Packages of the *L message* follow the format as indicated in Figure 7. The second kind of message is a command that tells the nodes start to measure; we call this kind of message as *start message*, which is set as a simple 3 bytes packet '11 22 33'. Once the nodes receive the *start message*, they will start their timer and the node with ID number 1 will transmit first. The *start message* can also synchronize the wireless nodes.

Figure 6 The setup of the experiment.
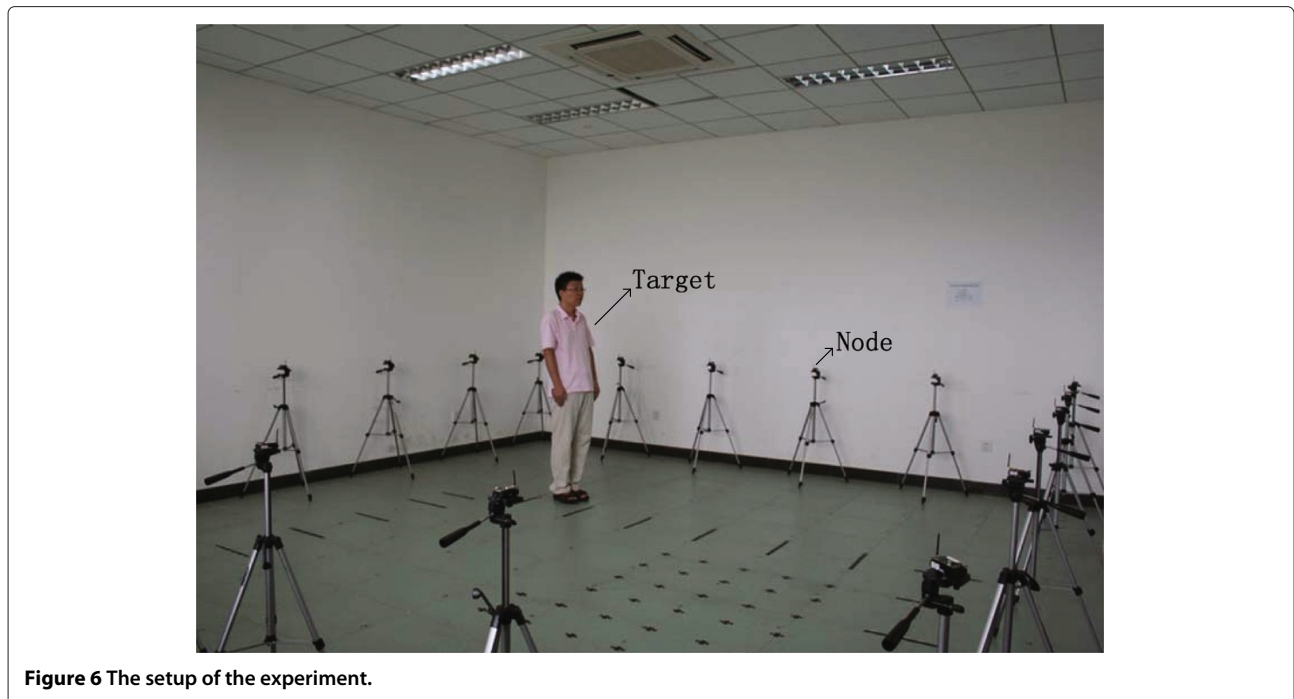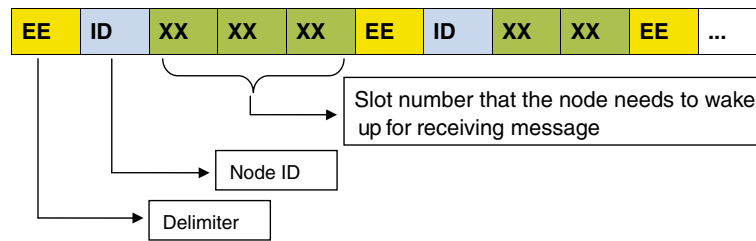
Figure 7 The packet format of the L message.

In the experiment, the system is calibrated by taking RSS measurements while the network is vacant from moving targets. The RSS vector is averaged over a 120-second period, which results in approximately 1,240 samples from each link. The calibrated RSS vector is saved in the laptop computer and provides a baseline against which all other RSS measurements are differenced. This process has to be done off-line.

The default parameters are as follows: the pixel size is 0.5×0.5 m, and the possible region threshold is $l = 0.75$ m; hence, the possible region includes 9 pixels. In the experiments, the parameters were set as given in Table 1.

5.2 Tracking performance

In order to verify the proposed method, three experiments were conducted: one for one target tracking, the others for two targets tracking. Each experiment was repeated two times: one experiment for real-time tracking with the CMMS algorithm and one that measured the RSS of all the wireless links for off-line processing with the other algorithms. In the experiments, the experimenter moves at a typical walking pace on a predefined path at a normal walking speed of 1.2 m/s. A metronome and uniformly placed markings on the floor help the experimenter to take constant-sized steps at a regular time interval. The actual location of the experimenter is interpolated using the start and stop time and the known marker positions.

In the first experiment, a target moved around the square is shown in Figure 8. The figure compares the true trace with the estimated ones obtained by CMMS, ℓ_1, regularization, and BGMP. In the experiment, we assume that the starting location of the target is known, which is reasonable in target tracking. It should be pointed out that

there were two consecutive locations estimated by the algorithms for each pixel that the target traveled. It is obvious that the proposed CMMS algorithm could achieve better tracking performance than the other algorithms.

The detailed statistical characters of the localization errors of different algorithms are summarized in Table 2. We can see that the average error of the proposed CMMS is 0.09 m which is smaller than the errors of the regularization (0.27 m) and BGMP (0.15 m) and even smaller than the error of the ℓ_1 (0.12 m). The maximum error of the CMMS is also smaller than the ones of the other algorithms. We additionally tested the average running time of different algorithms for once estimation on a dual-core 2.6 GHz PC. The proposed CMMS algorithm consumed about 0.06 ms, which is far less than 1 ms the regularization algorithm consumed 0.3 ms; the BGMP algorithm consumed approximately 5 ms; and ℓ_1 consumed much more about 28 ms. We can see that the CMMS needs considerably less time than the other algorithms.

Based on the proposed efficient measurement strategy, each node needs to receive 2.3 packets on average from the other nodes and additional 3 packets from the base station node for one estimation. Without utilizing the efficient measurement strategy, each node needs to receive 19 packets from all the other nodes. Each node needs to send one packet in the two situations. Therefore, one can save nearly 88% of the RSS measurements and about 72% of the radio communications using the proposed efficient measurement strategy.

In experiments 2 and 3, two targets walked along the polyline shown in Figure 9a,b. The target A moved from A1 to A2; simultaneously, target B moved from B1 to B2. In Figure 9a, the distance between target A and target B was always longer than 3 pixels. We can see that the trajectories estimated by the CMMS algorithm is similar to the real ones. The average error of this experiment is 0.11 m. In Figure 9b, target A and target B moved into two adjacent pixels $(1.25, 2.25)$ and $(1.25, 2.75)$. We can see that the tracking performance is getting worse when two targets moved too close to each other. The average error of this experiment increases to 0.23 m. In summary, the total average error of the experiments of two targets is 0.17 m, thus the experimental tracking performance is acceptable.

Table 1 Relevant parameter settings

Parameter	Value	Description
w	0.5	Pixel width (m)
λ	0.035	Width of weighting ellipse (m)
α	4	Regularization parameter
l	0.75	Threshold for calculate possible region (m)
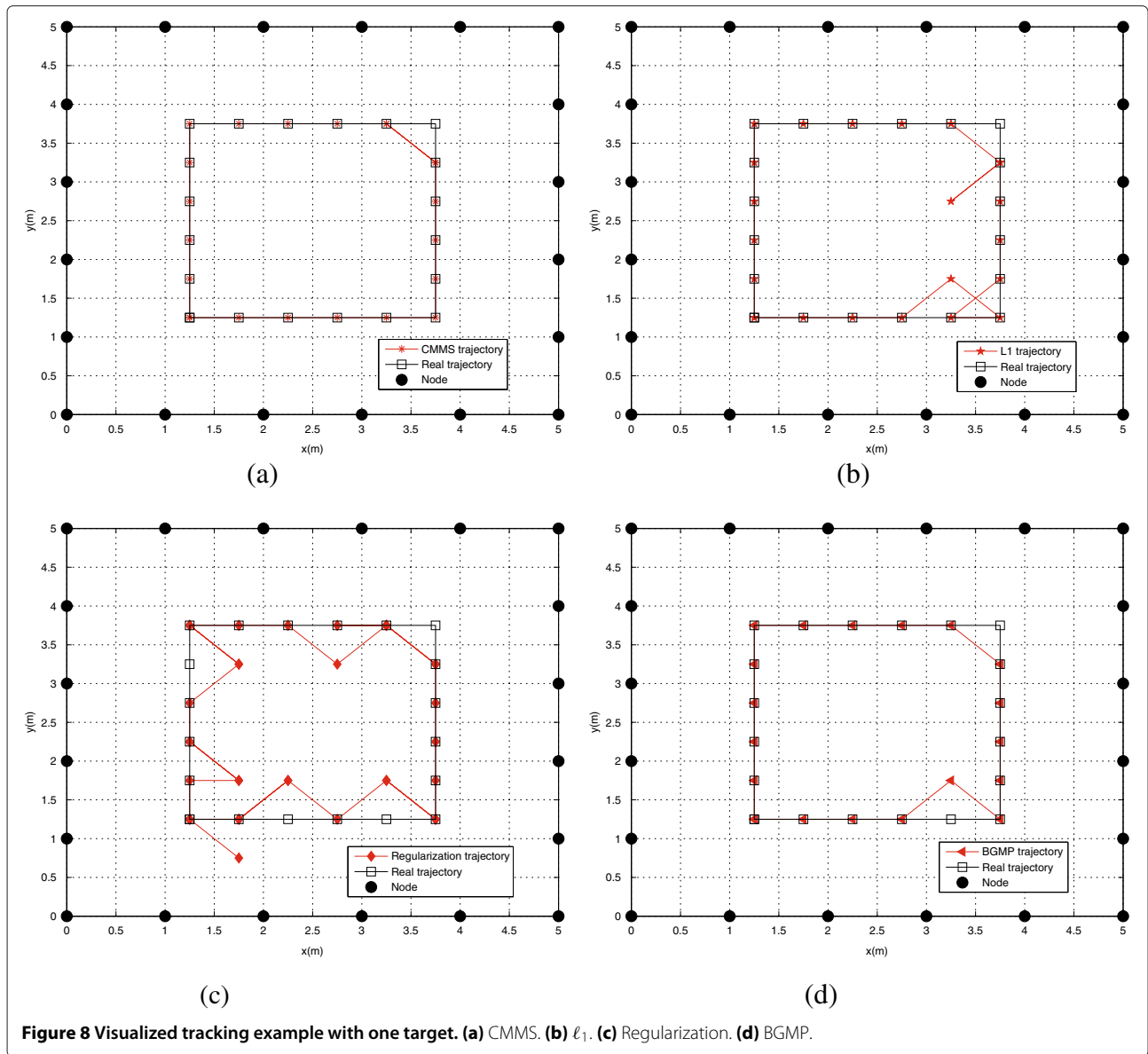
Figure 8 Visualized tracking example with one target. (a) CMMS. **(b)** ℓ_1. **(c)** Regularization. **(d)** BGMP.
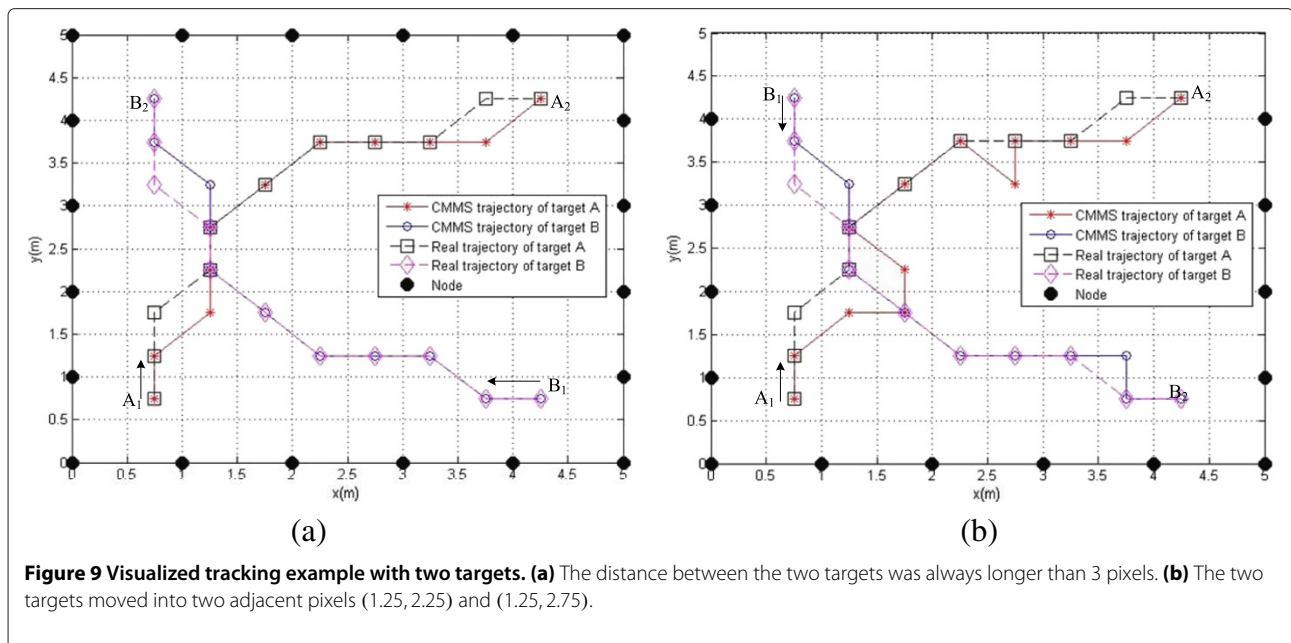
5.3 Analyses and discussions

We first analyze the latency of the system. Since the total number of the nodes is 20, there 20 time slots in the measurement phase. A MICAz node takes 7 ms on average to transmit a packet with 51 bytes, so we assigned each time slot with 8 ms. The measurement phase needs $20 \times 8 = 160$ ms $= 0.16$ s. In the first experiment, the number of links each node needs to measure on average is 2.3, following the packet format in Figure 7; each node needs up to 5 bytes on average in the packet of the *L message*. The maximum length of the payload in TinyOS packet is over 51 bytes, so the *L message* can be sent in

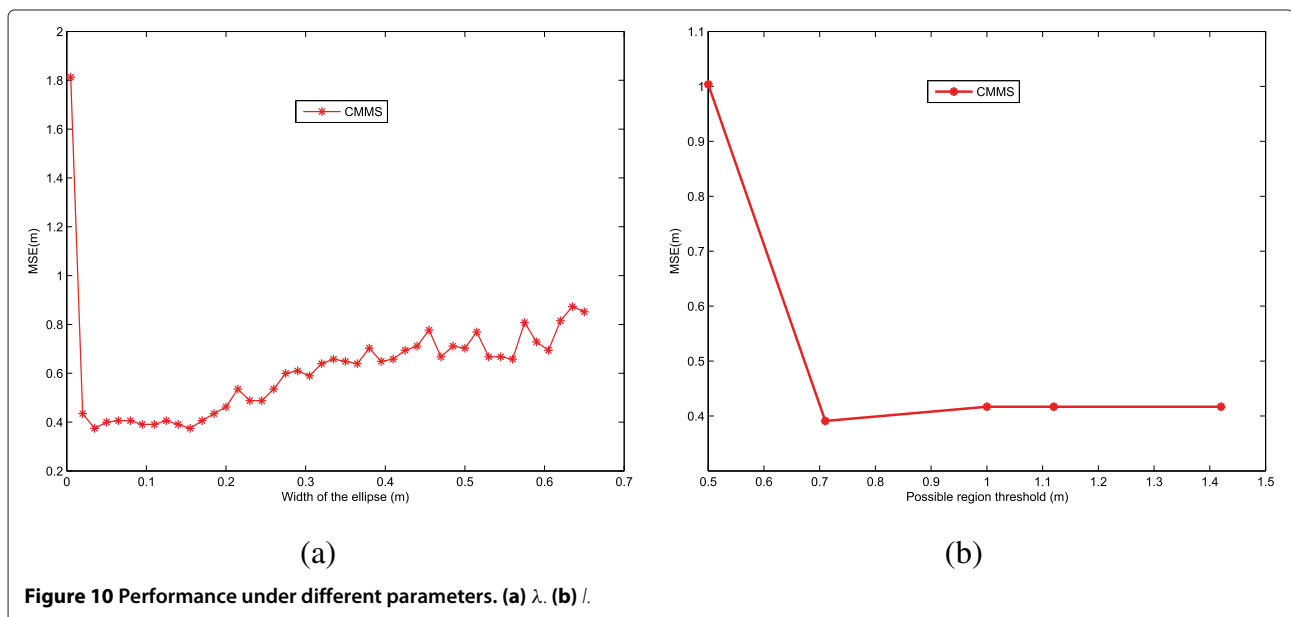**Table 2 Comparison of localization error and execution time**

Algorithm	Median (m)	Average (m)	Stand deviation (m)	90% (m)	Max (m)	CPU time (ms)
CMMS	0	0.09	0.19	0.50	0.50	0.06
ℓ_1	0	0.12	0.23	0.50	0.71	28
Regularization	0	0.27	0.29	0.71	0.71	0.3
BGMP	0	0.15	0.24	0.50	0.71	5

Figure 9 Visualized tracking example with two targets. (a) The distance between the two targets was always longer than 3 pixels. **(b)** The two targets moved into two adjacent pixels (1.25, 2.25) and (1.25, 2.75).
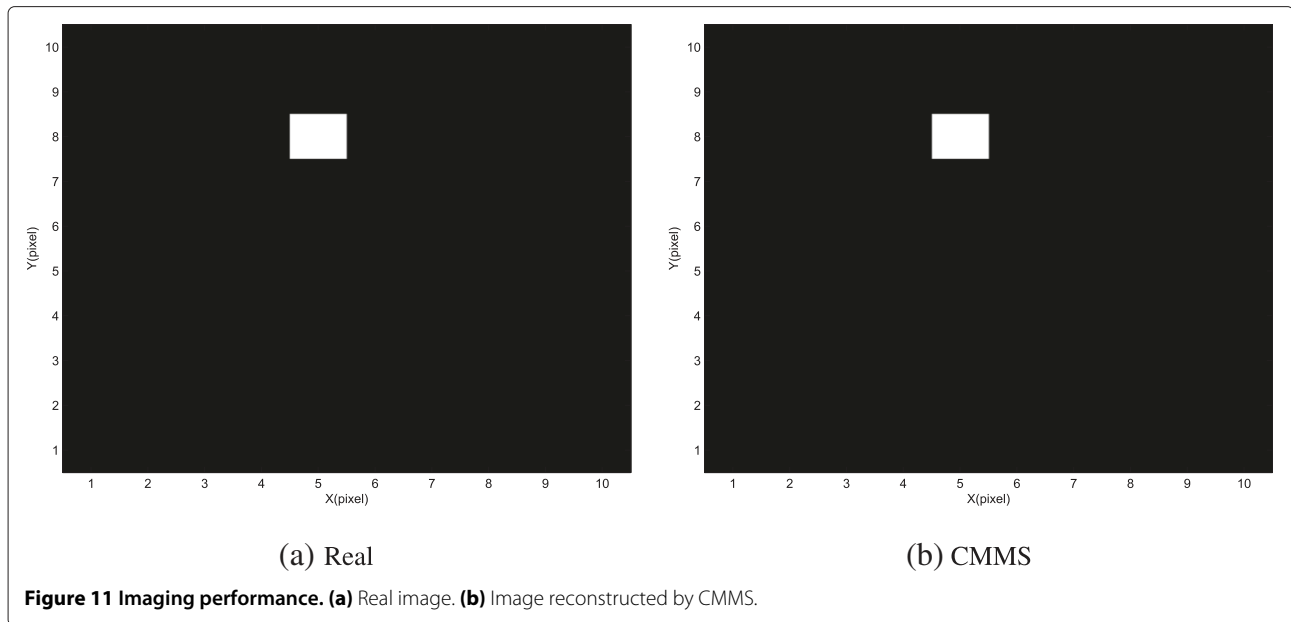
two packets. The time required by the base station node sending *L message* and *start message* to the nodes is up to $(2+1) \times 7 = 21$ ms. The time needed for the host running the reconstruction algorithm is less than 1 ms. In total, once estimation of the location of the target needs up to $160 + 21 + 1 = 182$ ms ≈ 0.2 s. In summary, our system can reach the real-time tracking with the latency of about 0.2 s, which significantly outperforms previous tracking systems [10,11,14]. The latency will increase as enlargement of the deployment area with more nodes. Based on the above analysis, we can see that the system latency will

increase at less than 10 ms with each additional node. In general application, two estimation in 1 s is enough. On this condition, the system can be extended up to 50 nodes. The system can also be extended by clustering nodes into different regions which are assigned with separate channels and share timeslots as introduced in [11]. Hence, the system is scalable.

The width of the pixel is an important parameter in RTI problem, which correlates highly with the resolution. We believe that it is mainly dependent on the application. In the experiments, we set it at 0.5 m, which can

Figure 10 Performance under different parameters. (a) λ. **(b)** l.

(a) Real   (b) CMMS

Figure 11 Imaging performance. (a) Real image. **(b)** Image reconstructed by CMMS.

meet the requirement of general applications. In addition, increasing or decreasing of the width of the pixel will not significantly change the latency of the system.

To evaluate the performance of the proposed method under different conditions, we evaluate the algorithm with different width of the ellipse λ and possible region threshold l. The weighting parameter plays an important role in generating accurate RTI images. If the ellipse is too wide, the pixels of where attenuation is not occurring may be included. If the ellipse is too narrow, pixels that do, in fact, attenuate a link's signal may not be captured by the model. This may result in a loss of information that degrades the final reconstruction quality. Figure 10a illustrates the mean squared error (MSE) with respect to different λ. It is obvious that the CMMS algorithm can achieve better results when λ in the neighborhood of the point 0.03. And the MSE with respect to different possible region threshold l is indicated in Figure 10b. We can see that the MSE is very large (over 1.8 m) when $l = 0.5$ m, because the target may move out of the possible region which only comprises 5 pixels. The MSE does not significantly increase with the threshold l increasing, but a very large value of l will increase both the computational complexity and the system latency. Essentially, the threshold l is determined by V_{max} and t_{int} ($l \geq V_{max} \times t_{int}$). If the movement speed of the target is very slow, one can set t_{int} with a big value (such as 1 or 2 s) for conserving system resources. Since once estimation of the target's location needs less than 0.2 s, the value of t_{int} can down to 0.2 s. On the precondition of $l = 0.75$ m and $w = 0.5$ m, the value of V_{max} can up to 3.75 m/s, which is much larger than the normal walking speed. In short, one should carefully select the value of l according to specific application for

reaching good tracking accuracy and conserving system resources.

Essentially, our proposed real-time DFL is based on the RTI theory, hence, the localization performance is depending on the imaging performance. We provide an imaging result in Figure 11. It can be seen that the CMMS algorithm could achieve reasonable imaging performance.

6 Conclusion

In this article, we designed and implemented a real-time DFL system, which is based on efficient measurement strategy and lightweight reconstruction algorithm. The measurement strategy makes use of the last localization result to predict a possible region of the target, then finds the wireless links which travel through the possible region to establish the set of correlated links. The system only needs to measure the RSS of the correlated links via cooperating with the bastion station node. As far as we know, we are the first to realize real-time DFL based on measurements of correlated links. Furthermore, the proposed CMMS algorithm only needs a small-scale matrix-vector product operation to reconstruct the signal and localize the target. In summary, we realized an efficient and energy-saving real-time DFL wireless network system. Experimental results demonstrate the effectiveness of our approach and confirm that the CMMS algorithm could achieve satisfactory localization and tracking results.

References

1. The Internet of Things, ITU Internet Reports (2005). http://www.itu.int/internetofthings/. Accessed 15 May 2013
2. C Fok, C Julien, G Roman, C Lu, in *Proceedings of the 2nd workshop on Software Engineering for Sensor Network Applications, SESENA'11,* Waikiki, Honolulu. Challenges of satisfying multiple stakeholders: quality of service in the internet of things (ACM New York, 2011), pp. 55–60
3. L Atzori, A Iera, G Morabito, The internet of things: a survey. Comput. Netw. **54**(15), 2787–2805 (2010)
4. N Patwari, J Wilson, RF sensor networks for device-free localization: measurements, models, and algorithms. Proc. IEEE. **98**(11), 1961–1973 (2010)
5. J Wilson, N Patwari, Radio tomographic imaging with wireless networks. IEEE Trans. Mob. Comput. **9**(5), 621–632 (2010)
6. J Wilson, N Patwari, F Vasquez, in *Virginia Tech Symposium on Wireless Personal Communications*. Regularization methods for radio tomographic imaging. The Inn, 3–5 June 2009
7. J Wilson, N Patwari, See-through walls: motion tracking using variance-based radio tomography networks. IEEE Trans. Mob. Comput. **10**(5), 612–621 (2011)
8. M Moussa, M Youssef, in *Proceedings of the 2009 IEEE International Conference on Pervasive Computing and Communications (PerCom)*. Smart devices for smart environments: device-free passive detection in real environments. Galveston, TX, 9–13 March 2009
9. M Youssef, M Mah, A Agrawala, in *Proceedings of the 13th annual ACM International Conference on Mobile Computing and Networking (MobiCom),* Montreal, 9–14 September. Challenges: device-free passive localization for wireless environments (ACM New York, 2007), pp. 222–229
10. D Zhang, J Ma, Q Chen, LM Ni, in *Proceedings of the Fifth IEEE International Conference on Pervasive Computing and Communications (PerCom)* White Plains, NY, 19–23 March. An RF-based system for tracking transceiver-free objects (IEEE Computer Society Press Los Alamitos, 2007), pp. 135–144
11. D Zhang, LM Ni, in *Proceedings of the 2009 IEEE International Conference on Pervasive Computing and Communications (PerCom)* Galveston, 9–13 March. Dynamic clustering for tracking multiple transceiver-free objects (IEEE Computer Society Press Los Alamitos, 2009), pp. 1–8
12. D Zhang, Y Liu, LM Ni, in *Proceedings of the 30th International Conference on Distributed Computing Systems (ICDCS)* Genova, 21–25 June. Link-centric probabilistic coverage model for transceiver-free object detection in wireless networks (IEEE Press, Piscataway, 2010), pp. 116–125
13. D Zhang, Y Yang, D Cheng, S Liu, LM Ni, in *Proceedings of the 2010 IEEE International Conference on Communications (ICC),* Cape Town, 23–27 May. Cocktail: an RF-based hybrid approach for indoor localization (IEEE Press, Piscataway, 2010), pp. 1–5
14. D Zhang, Y Liu, LM Ni, in *Proceedings of the 2011 IEEE International Conference on Pervasive Computing and Communications (PerCom)* Seattle, 21–25 March. RASS: a real-time, accurate and scalable system for tracking transceiver-free objects (IEEE Computer Society Press, Los Alamitos, 2011), pp. 197–204
15. M Kanso, M Rabbat, in *Distributed Computing in Sensor Systems*. Compressed RF tomography for wireless sensor networks: centralized and decentralized approaches (Springer, New York, 2009), pp. 173–186
16. J Wang, Q Gao, H Chen, X Zhang, H Wang, Device-free localisation with wireless networks based on compressive sensing. IET Commun. **6**(15), 2395–2403 (2012)
17. V Seshadri, G Zaruba, M Huber, in *Proceedings of the Third IEEE International Conference on Pervasive Computing and Communications (PerCom)* Kauai Island, HI , 8-12 March. A Bayesian sampling approach to in-door localization of wireless devices using received signal strength indication (IEEE Computer Society Press, Los Alamitos, 2005), pp. 75–84
18. R Ouyang, AS Wong, CT Lea, Received signal strength-based wireless localization via semidefinite programming: noncooperative and cooperative schemes. IEEE Trans. Veh. Technol. **59**(3), 1307–1318 (2010)
19. X Li, Collaborative localization with received-signal strength in wireless sensor networks. IEEE Trans. Veh. Technol. **56**(6), 3807–3817 (2007)
20. Q Shi, C He, H Chen, L Jiang, Distributed wireless sensor network localization via sequential greedy optimization algorithm. IEEE Trans. Signal Process. **58**(6), 3328–3340 (2010)
21. X Pi, H Yu, in *Proceedings of the Sixth International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT),* Dalian, 5–8 December. A distributed and cooperative target localization algorithm in wireless sensor networks (IEEE Computer Society Press, Los Alamitos, 2005), pp. 887–889
22. N Patwari, SK Kasera, Temporal link signature measurements for location distinction. IEEE Trans. Mob. Comput. **10**(3), 449–462 (2011)
23. J Wilson, N Patwari, A fade level Skew-Laplace signal strength model for device-free localization with wireless networks. IEEE Trans. Mob. Comput. **6**(99), 947–958 (2012)
24. X Chen, A Edelstein, Y Li, M Coates, M Rabbat, A Men, in *Proceedings of the 10th International Conference on Information Processing in Sensor Networks (IPSN),* Chicago, 12–14 April. Sequential Monte Carlo for simultaneous passive device-free tracking and sensor localization using received signal strength measurements (IEEE Computer Society Press, Los Alamitos, 2011), pp. 342–353
25. J Tropp, A Gilbert, Signal recovery from random measurements via orthogonal matching pursuit. IEEE Trans. Inf. Theory. **53**(12), 4655–4666 (2007)
26. MICAz, MEMSIC, Wireless sensor networks. http://www.memsic.com/wireless-sensor-networks/. Accessed 15 May 2013