Computer Communications 78 (2016) 45–55

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

TDL: Two-dimensional localization for mobile targets using compressive sensing in wireless sensor networks

Baoming Sun [a,*], Yan Guo [a], Ning Li [a], Laixian Peng [a], Dagang Fang [b]

[a] College of Communications Engineering, PLA University of Science and Technology, China
[b] School of Electronic and Optical Engineering, Nanjing University of Science and Technology, China

ARTICLE INFO

Article history:
Received 10 January 2015
Revised 19 September 2015
Accepted 17 October 2015
Available online 30 October 2015

Keywords:
Wireless sensor networks
Two-dimensional localization
Mobile targets
Compressive sensing

ABSTRACT

Many applications in wireless sensor networks (WSNs) (*e.g.*, traffic monitoring, environment surveillance and intruder tracking) rely heavily on the availability and accuracy of targets' locations. Compressive sensing (CS) has been widely applied to localization as it asserts that a small number of samples will suffice for sparse or compressible signal recovery. Despite much progress in CS-based localization, existing solutions mainly consider static targets and often perform poorly for mobile targets.

In this paper, we develop a novel two-dimensional localization (TDL) framework for mobile targets using compressive sensing. TDL is composed of two modules: (i) spatial localization module (SLM) that first achieves localization at sampling times by exploiting the sparse nature of Received Signal Strength (RSS) vector in space domain, and (ii) temporal localization module (TLM) that then achieves localization at all times by exploiting the compressible nature of location vector in time domain. Furthermore, two practicable measurement matrices are constructed to conduct linear measurements. We analyze the flexibility and effectiveness of TDL in theory. Extensive numerical evaluations with real mobility traces further confirm the superior performance of our localization framework.

1. Introduction

1.1. Motivation

Wireless sensor networks (WSNs) attract much attention as they promise an ability to monitor the physical world *via* a lot of small and inexpensive sensors. Location awareness is highly critical to many applications in WSNs, such as geographic routing [1], disaster response [2], environment surveillance [3], and vehicle tracking [4]. The Global Positioning System (GPS) [5] is widely used to obtain location information in WSNs. However, there are several situations (*e.g.*, indoors, under the ground, or in urban environments) where GPS does not work well due to the lack of line of sight to multiple satellites. Moreover, due to the constraints on hardware cost, it is undesirable and unfeasible to equip each target with a GPS.

The limitations of GPS have motivated researchers to develop a large body of literatures on localization. However, most of these localization schemes fail to localize mobile targets as they are designed for static targets. For static targets, it is not a problem because their positions are unlikely to change once determined. However, for mobile targets, it is a significant challenge since they may change their positions at any time.

As a matter of fact, most targets in WSNs are mobile. For example, rescuers move in a disaster area, soldiers move in a battlefield, animals move in a habitat, and vehicles move in a city. There are only a limited number of researches that consider the problem of mobile target localization. Moreover, these researches only use a maximum speed to constrain the distance between a target's positions during two consecutive time intervals while they do not fully exploit the hidden compressible nature of the target's positions during all time intervals.

Another simple approach to mobile target localization is to divide time into several time intervals. In each interval, a target can be considered static and localized using static localization methods. However, the localization accuracy of this approach depends heavily on the resolution of time division. To achieve accurate localization, high resolution of time division is needed, posing great challenges to hardware equipments in terms of sampling, storage and calculation.

Compressive sensing (CS) technique [6,7] provides a new solution to the problem of mobile target localization. As a novel signal sampling paradigm, CS asserts that a small number of samples will suffice for original signal recovery. To achieve this, CS relies on two key components: sparsity and incoherence.

* Corresponding author. Yan Guo. Room 701, Block 30, Rong Yuan Shan Zhuang, 240 Mu Xu Yuan Street, Qin Huai District, Nanjing, Jiangsu, 210007, P.R. China.
E-mail addresses: baomings1988@sina.com (B. Sun), guoyan_2000@sina.com (Y. Guo).

http://dx.doi.org/10.1016/j.comcom.2015.10.006
0140-3664/

- *Sparsity* expresses the idea that many real-world signals are sparse or compressible in the sense that they have concise representations when expressed in some representation basis.
- *Incoherence* expresses the idea that the sparse signals in some representation basis must be spread out in another domain where they are sampled. In other words, incoherence indicates that unlike the signals of interest, the sampling waveforms should have extremely dense representations in the representation basis.

We observe that the spatial signal (*e.g.*, RSS vector) and the temporal signal (*e.g.*, location vector) are sparse or compressible in appropriate representation bases. Motivated by the observation, we investigate the solution to mobile target localization using compressive sensing.

1.2. Our work and contributions

There are two key challenges in applying CS technique to our problem. (1) How to find proper representation bases in which the original signals can be sparsely represented. (2) The measurement matrix in our context is restricted by physical constraints. In primal literature, the measurement matrix is usually specified by a dense matrix, *e.g.*, Gaussian matrix, as it exhibits very low coherence with any representation basis. However, it should be noted that the dense measurement matrix is not feasible in practice. As a matter of fact, each measurement is a linear combination of multiple samples of the underlying signal [8]. It almost requires all samples of the signal since there are nearly no empty columns in the dense measurement matrix. This obviously goes against compressive sensing theory.

In this paper, we leverage compressive sensing to develop a two-dimensional localization framework for mobile targets. The localization framework takes full advantage of the sparse or compressible nature of signals to highly reduce the data collection needed for accurate localization in both space and time domains. The main contributions of this paper can be summarized as follows.

- We propose a novel two-dimensional localization framework for mobile targets. The framework is composed of a spatial localization module (SLM) and a temporal localization module (TLM).
- We design appropriate representation bases by exploiting the sparse or compressive nature of signals in both space and time domains. It is validated that these representation bases can sufficiently sparsify the underlying signals.
- We develop two simple and practical measurement matrices to conduct linear measurements. We demonstrate that they are highly incoherent with designed representation bases.
- We perform extensive simulations to evaluate the performance of TDL with various parameter settings. The superiority of TDL compared with other approaches is validated by the simulation results.

The remainder of this paper is organized as follows. A brief review of related work is presented in Section 2. We introduce background of compressive sensing and mathematically formulate the problem in Section 3. Section 4 provides detailed descriptions on our localization framework. The matrix design and performance analysis are given in Section 5. Section 6 demonstrates the performance of our localization framework through extensive numerical evaluations. Section 7 gives a discussion about the localization framework. Finally, we conclude the paper in Section 8.

Notations: we use bold uppercase (lowercase) letters to denote matrices (vectors). $(\cdot)^{\mathrm{T}}$ denotes the transpose, $(\cdot)^{-1}$ denotes the inverse, $\|\cdot\|_p$ denotes the p-normal, $\langle\cdot\rangle$ denotes the inner product, $min(\cdot)$ denotes the minimization operator, and $max(\cdot)$ denotes the maximization operator.

2. Related work

In this section, we first review the applications of CS in WSNs, and then summarize the existing researches on target localization.

2.1. CS applications in WSNs

Compressive sensing has been widely applied to data collection in WSNs because of its ability to reduce signal samplings significantly and balance energy consumption across sensors [9]. In single-hop WSNs, Compressive Wireless Sensing (CWS) [10] considers the spatial correlation among sensor readings and reduces the latency of data gathering by delivering the linear projections of sensor readings. Distributed Compressive Sensing (DCS) [11] extends CWS to time domain by considering both spatial and temporal correlations among sensor readings. DCS studies joint sparsity models and joint data recovery algorithms without considering multi-hop communication and in-network data processing. Compressive Data Gathering (CDG) [12,13] addresses the large-scale data gathering problem in multi-hop WSNs. CDG uses dense measurement matrices for CS projections, not achieving as much energy reduction as sparse matrices. Mahmudimanesh et al. [14] significantly enhance CDG by balancing computation and communication loads over all sensors. There are fundamental differences between the aforementioned applications and the application in this paper. On one hand, we are interested in using a few sensor readings to estimate mobile targets' locations while the aforementioned applications aim at reconstructing all sensor readings. On the other hand, the aforementioned applications consider the spatial or temporal correlation of the same signal while we explore the spatial or temporal correlation of different signals, *i.e.* the spatial correlation of RSS readings and temporal correlation of targets' locations.

2.2. Target localization

We classify existing target localization researches into four categories: (i) non-CS based static target localization, (ii) CS based static target localization, (iii) non-CS based mobile target localization, and (iv) CS based mobile target localization.

(i) *Non-CS based static target localization:* These methods are either range-based or range-free. Range-based methods [15–18] first use *Received Signal Strength* (RSS), *Time of Arrival* (ToA), *Time Difference of Arrival* (TDoA), or *Angle of Arrival* (AoA) to measure the distances or angles between unknown nodes and anchors with known positions, and then use trilateration, triangulation, or maximum likelihood to determine the positions of unknown nodes. These methods are simple but have two significant drawbacks: (1) they are sensitive to fading, noise and non-line of sight, and (2) it is often not affordable to equip all nodes with ranging capability. On the contrary, range-free methods [19–21] do not require the hardware support for measuring distances or angles. Instead, they exploit the network connectivity or proximity relationship between nodes. Range-free methods are easy to implement in WSNs, but they only achieve a low level of accuracy in most situations.

(ii) *CS based static target localization:* Localization Via Spatial Sparsity (LVSS) [22] is the first work to apply compressive sensing to localization problem. LVSS discretizes the area of interest into a grid so that the localization problem is formulated as a sparse signal recovery problem. The authors also propose a Bayesian framework [23] for the localization problem and provide sparse approximation to its optimal solution. The drawback of these methods is that a localization dictionary is needed at each sensor. By formulating multiple targets' locations as a sparse matrix, Feng et al. [24] propose a CS based indoor localization approach. The approach is only able to localize single target though it is designed for multiple targets. A clustering method

is further introduced to reduce computational complexity in [25,26]. The localization accuracy greatly depends on the clustering performance and online cluster matching algorithms. However, it is usually challenging to determine the number of clusters and choose an effective cluster matching algorithm. Zhang et al. [27] formulate multiple targets' locations as a sparse vector and leverage compressive sensing for localization. Guyen et al. [28] propose a RSS-based WLAN indoor positioning system where K-SVD is used to decompose the radio map into a dictionary and a sparse representation matrix. The support of sparse representation of real-time RSS vector is matched with the supports of sparse representation matrix to determine which reference point is closest to the target. Liu et al. [29] propose a range-free multiple target localization algorithm using compressive sensing. Instead of measuring RSSs, the information whether targets are detected by sensors are used to achieve localization. Nguyen et al. [30] propose a multiple target localization algorithm based on compressive sensing, where deterministic sensing matrices rather than random sensing matrices are used to take RSS measurements. However, the aforementioned localization methods are only designed for static targets and not suitable for mobile targets.

(iii) *Non-CS based mobile target localization:* Monte Carlo Localization (MCL) [31] is the first localization scheme for mobile targets. In MCL, a target uses its previous positions and maximum speed to generate possible current positions, then filters out infeasible positions using the current connectivity information from neighbor anchors. Mobile and Static Localization (MSL) [32] improves MCL by using the information from not only neighbor anchors but also neighbor unknown-nodes. This modification results in faster convergence speed and better location estimation. However, a common disadvantage for both MCL and MSL is that a lot of communication cost is needed to achieve accurate localization in high mobility environment.

(iv) *CS based mobile target localization:* A CS-based positioning system for single mobile target is introduced in [33], which consists of a coarse localization step and a fine localization step. The coarse localization step is executed to estimate the approximate position of the target, and then the accurate location is estimated using CS in the fine localization step. Deng et al. [34] propose a location-fingerprint based indoor positioning approach, which includes an offline stage and an online stage. At the offline stage, RSS measurements are taken to establish a fingerprint database. During the online stage, the realtime RSSs are collected and compared with the RSSs stored in the fingerprint database for location estimation. However, building the fingerprint database is tedious and time-consuming. By exploiting its hidden structure, a new fingerprint database building approach [35] based on compressive sensing is provided to recover absent fingerprints using a few fingerprints. However, all of these approaches only utilize the sparse nature of signals in space domain and neglect the compressible nature of signals in time domain.

3. Compressive sensing and problem formulation

In this section, we first provide a brief background on compressive sensing, and then focus on the problem formulation.

3.1. Compressive sensing

Consider a discrete signal given by vector $\mathbf{x} \in \mathbb{R}^N$. It is referred to as a sparse vector if $\mathbf{x}$ has only a few non-zero elements, *i.e.*, $\|\mathbf{x}\|_0 \ll N$. More generally, $\mathbf{x}$ is called compressible in the sense that it has many small entries, and only a few large entries carrying most of its information. Results in CS have shown that if $\mathbf{x}$ is sparse or compressible, then it is possible to reconstruct $\mathbf{x}$ from M measurements produced by a proper linear transform $\mathbf{\Phi}$:

$$\mathbf{y} = \mathbf{\Phi}\mathbf{x}, \tag{1}$$

where $M \ll N$. The matrix $\mathbf{\Phi} \in \mathbb{R}^{M\times N}$ is usually termed as *measurement matrix*. It should be noted that most signals in practice are not truly sparse or compressible. However, they can be sparsely represented in an alternative domain. Specifically, $\mathbf{x}$ may be linearly represented as $\mathbf{x} = \mathbf{\Psi}\mathbf{s}$, for some matrix $\mathbf{\Psi} \in \mathbb{R}^{N\times N}$, where $\mathbf{s}$ is the sparse or compressible coefficient vector in $\mathbf{\Psi}$-domain. The matrix $\mathbf{\Psi}$ is referred to as *representation basis*. Therefore the measurement vector can be expressed as:

$$\mathbf{y} = \mathbf{\Phi}\mathbf{\Psi}\mathbf{s}. \tag{2}$$

Obviously, it is an under-determined linear system in that the number of measurements M is much smaller than the number of unknowns N. This ill-conditioned system can be solved by minimizing the ℓ_0 norm:

$$\ell_0 : \ \hat{\mathbf{s}} = \arg\min_{\mathbf{s}\in\mathbb{R}^N} \|\mathbf{s}\|_0 \text{ s.t. } \mathbf{y} = \mathbf{\Phi}\mathbf{\Psi}\mathbf{s}. \tag{3}$$

Directly solving the problem is NP-hard. However, it is feasible when $\mathbf{\Phi}\mathbf{\Psi}$ satisfies the restricted isometric property (RIP) [36]:

Definition 1. A matrix $\mathbf{A}$ satisfies the restricted isometry property (RIP) of order k if there exists a $\delta_k \in (0, 1)$ such that

$$(1-\delta_k)\|\mathbf{s}\|_2^2 \le \|\mathbf{A}\mathbf{s}\| \le (1+\delta_k)\|\mathbf{s}\|_2^2 \tag{4}$$

for all k-sparse vectors $\mathbf{s} \in \mathbb{R}^N$.

When $\mathbf{\Phi}\mathbf{\Psi}$ satisfies the RIP, the solution can be obtained by solving the following ℓ_1-minimization problem:

$$\ell_1 : \ \hat{\mathbf{s}} = \arg\min_{\mathbf{s}\in\mathbb{R}^N} \|\mathbf{s}\|_1 \text{ s.t. } \mathbf{y} = \mathbf{\Phi}\mathbf{\Psi}\mathbf{s}. \tag{5}$$

The problem can be easily solved using existing recovery algorithms (also named as solvers later), such as Basis Pursuit (BP) [36] and Orthogonal Matching Pursuit (OMP) [37]. Once the coefficient vector $\hat{\mathbf{s}}$ is determined, the original signal can be reconstructed as $\hat{\mathbf{x}} = \mathbf{\Psi}\hat{\mathbf{s}}$.

No matter which solver is used, the measurement number M should satisfy the following condition [38]:

$$M \geqslant C\mu^2(\mathbf{\Phi}, \mathbf{\Psi})k\log N, \tag{6}$$

where C is some positive constant, $\mu(\mathbf{\Phi}, \mathbf{\Psi})$ is the coherence between measurement matrix $\mathbf{\Phi}$ and representation basis $\mathbf{\Psi}$. The coherence $\mu(\mathbf{\Phi}, \mathbf{\Psi})$, for a given pair $(\mathbf{\Phi}, \mathbf{\Psi})$ in $\mathbb{R}^N$, can be defined as

$$\mu(\mathbf{\Phi}, \mathbf{\Psi}) = \sqrt{N} \max_{1\leqslant i,j\leqslant N} |\langle \varphi_i, \psi_j \rangle| \in [1, \sqrt{N}], \tag{7}$$

where $\boldsymbol{\varphi}_i$ is the ith row of $\mathbf{\Phi}$ and $\boldsymbol{\psi}_j$ is the jth column of $\mathbf{\Psi}$. Therefore given $\mathbf{x}$, the matrix pair $(\mathbf{\Phi}, \mathbf{\Psi})$ should be chosen carefully: it is desirable to represent $\mathbf{x}$ in $\mathbf{\Psi}$-domain sparsely at the same time to make $\mu(\mathbf{\Phi}, \mathbf{\Psi})$ as small as possible.

3.2. Problem formulation

We consider a wireless sensor network with k mobile targets as seen in Fig. 1. Each target is equipped with an emitter which broadcasts beacons periodically. Some sensors are deployed in the monitored area to measure the RSSs. Then these RSSs are transmitted to a distant Fusion Center (FC) according to the compressive data gathering method introduced in [39]. Finally, a recovery algorithm is executed to estimate these targets' locations in the FC. In this manner, we transfer computing load from sensors to the FC, significantly reducing the energy cost of sensors. Our objective is to determine where to deploy sensors and when these sensors collect RSSs.

For the sake of simplicity, we suppose that these mobile targets' locations are discrete in both space and time domains. Space discretization can be realized by dividing the monitored area into a grid with N cells while time discretization can be realized by dividing the time scale into T time intervals. It is obviously not true, but

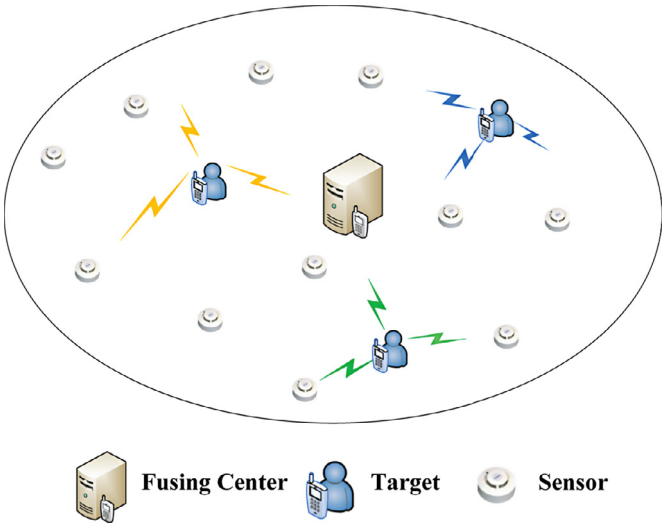

Fig. 1. Network scenario.

a quite good representation of actual targets' locations if the cell size and time interval are small enough with respect to the grid size and time scale. More importantly, it is worth noting that since RSSs are measured discretely no matter how high the frequency is, the measurements are inevitably discrete in both space and time domains. Therefore adopting a discrete model allows us to precisely evaluate the performance of our approach.

4. Two-dimensional localization framework

In this section, we first present the overview of two-dimensional localization (TDL)framework, and then give the detailed description of its two modules: spatial localization module (SLM) and temporal localization module (TLM).

4.1. Overview of TDL

An overview of proposed two-dimensional localization framework is illustrated in Fig. 2.

In spatial localization module, at sampling times specified by temporal measurement matrix, RSSs are collected at sampling cells specified by spatial measurement matrix, and then these RSSs are transmitted to the FC; at last the sparse representation of RSS vector is estimated by solving a ℓ_1-norm minimization problem. After this, the locations of all targets at all sampling times are known.

In temporal localization module, for each target, the sparse representation of its position vector is estimated with its locations at sampling times by solving a ℓ_1-norm minimization problem. Then the location vector can be recovered by multiplying the sparse representation with temporal representation basis.

4.2. Spatial localization module

In space domain, we consider a case where k targets exist in a monitored area, which is divided into a grid with N cells. The positions of these cells and target number k are known a priori. Since a target's location is unique in the discrete space at a certain time, by assuming that a cell only holds one target [24], we can represent these targets' positions as an ideal k-sparse vector $\mathbf{c} = [c_1, c_2, \cdots, c_N]^{\mathrm{T}}$. The support of $\mathbf{c}$ encodes the locations of multiple targets, one nonzero element represents a target and its index represents the location of the target. At a certain time, we denote the RSSs at N cells by a vector $\mathbf{r} = [r_1, r_2, \cdots, r_N]^{\mathrm{T}}$. Then the vector $\mathbf{r}$ can be expressed as:

$$\mathbf{r} = \mathbf{\Psi}_D \mathbf{c}, \tag{8}$$

where $\mathbf{\Psi}_D$ is a $N \times N$ energy decay matrix, whose element $\mathbf{\Psi}_D(i, j) = RSS(d_{ij})$ denotes the RSS reading at cell i from the target at cell j.

To localize these targets, a traditional approach is to place one sensor at each cell to take RSS measurements, and then obtain the sparse vector $\mathbf{c}$ according to Eq. (8). However, in this approach, a large number of sensors are needed to take RSS measurements, bringing high cost. it should be noted that the problem has a sparse nature as the RSS vector $\mathbf{r}$ is fully sparse in the representation basis $\mathbf{\Psi}_D$. The sparse nature motivates us to reconstruct the sparse vector $\mathbf{c}$ by collecting a few RSS samples.

4.3. Temporal localization module

As for time domain, without loss of generality, we only consider single mobile target in one-dimensional Euclidean space, as the same method can be applied to multiple targets and two-dimensional Euclidean space. To discretize the mobile target's locations, we divide time into T intervals. In each interval, the target can be considered static.

We denote by $\mathbf{x} = [x_1, x_2, \cdots, x_T]^{\mathrm{T}}$ an actual realization of a mobile target's locations during T time intervals. x_i represents the target's location at the i-th time interval. After SLM, the target's locations at sampling times are known. CS provides us a solution to recovering the location vector $\mathbf{x}$ with the locations at sampling times as long as $\mathbf{x}$ is sparse or compressible. An important point is that the location vector in practice is not truly sparse or compressible. However, it can be sparsely represented in an alternative representation basis.

5. Matrix design and performance analysis

The performance of signal recovery is determined by two key components: measurement matrix $\mathbf{\Phi}$ and representation basis $\mathbf{\Psi}$. The matrix $\mathbf{\Phi}$ directly corresponds to a measurement policy, whereas $\mathbf{\Psi}$ is used not only to sparsify the original signal but also to recover it once its coefficient vector is determined. In this section, we first present how to design $\mathbf{\Psi}$ and $\mathbf{\Phi}$, and then check the performance of designed matrices in terms of sparsity and coherence.

5.1. Representation basis design

(1) *Spatial representation basis:* As we pointed out early, the RSS vector $\mathbf{r}$ can be fully sparsely represented in basis $\mathbf{\Psi}_D$. Therefore we construct spatial representation basis as $\mathbf{\Psi}_D$ produced by the radio propagation model [40], which follows:

$$RSS(d) = P_t + K_e - 10\eta \log_{10}\left(\frac{d}{d_0}\right) + X_\sigma, \tag{9}$$

where P_t denotes the transmitting power; K_e is a constant that depends on the environment; d is the target-sensor distance, and d_0 is a reference distance; η is the path loss coefficient; X_σ is a random variable that follows the Gaussian distribution with zero mean and variance of σ^2.

(2) *Temporal representation basis:* In fact, a mobile target's location vector $\mathbf{x}$ is not sparse or compressible itself. However, we observe that the target changes its speed slowly. Motivated by this observation, we sparsely represent the location vector with a *speed-difference* matrix $\mathbf{M}_S = Toeplitz(0, 1, -2, 1)$, which denotes the matrix with central diagonal given by "1", the first upper diagonal given by "−2", and the

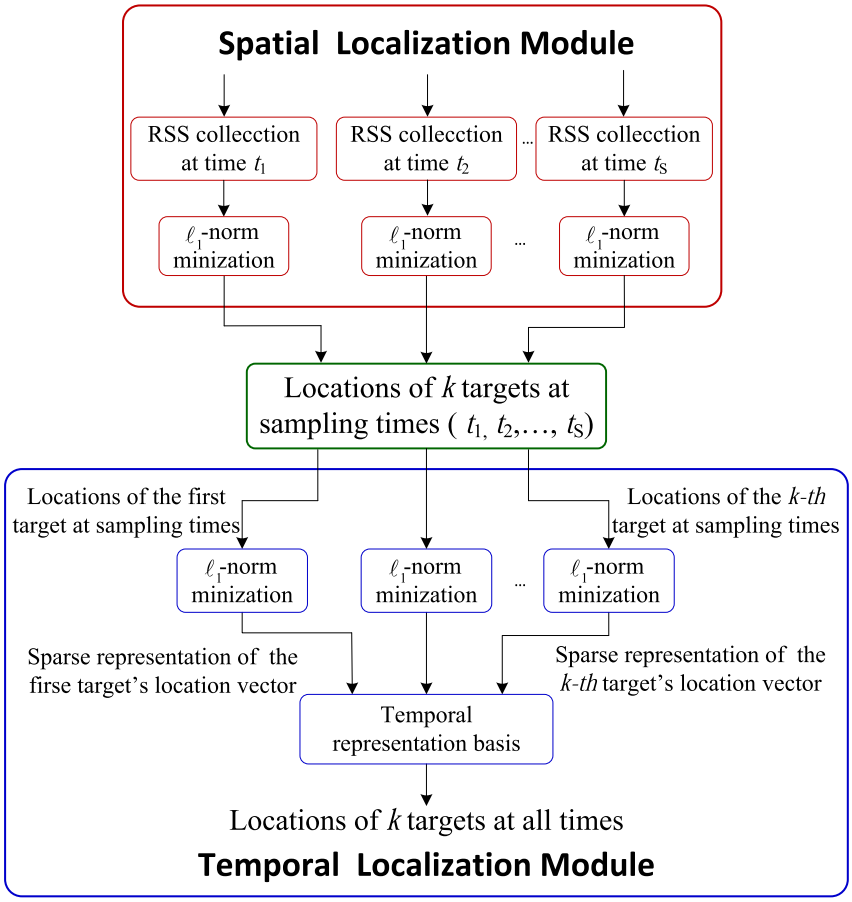

Fig. 2. Block diagram of proposed mobile target localization framework.

second upper diagonal given by "1", *i.e.*,

$$\mathbf{M}_S = \begin{bmatrix} 1 & -2 & 1 & 0 & \cdots \\ 0 & 1 & -2 & 1 & \ddots \\ 0 & 0 & 1 & -2 & \ddots \\ \vdots & \ddots & \ddots & \ddots & \ddots \end{bmatrix}. \tag{10}$$

We project $\mathbf{x}$ onto $\mathbf{M}_S$ and obtain the coefficient vector, where each term $(x_{i+1} - x_i) - (x_i - x_{i-1})$ represents the difference between speeds of the target during two consecutive time intervals. The more slowly the target changes its speed, the smaller the terms are, *i.e.*, the more compressible the coefficient vector is. Therefore we define the representation basis as $\mathbf{\Psi}_S = \mathbf{M}_S^{-1}$.

For the sake of comparison, we also consider a *location-difference* matrix $\mathbf{M}_L = Toeplitz(0, 1, -1, 0)$ introduced in [41] that captures the temporal smoothness of the soil moisture process. $\mathbf{M}_L$ denotes the matrix with central diagonal given by "1", the first upper diagonal given by "−1", *i.e.*,

$$\mathbf{M}_L = \begin{bmatrix} 1 & -1 & 0 & 0 & \cdots \\ 0 & 1 & -1 & 0 & \ddots \\ 0 & 0 & 1 & -1 & \ddots \\ \vdots & \ddots & \ddots & \ddots & \ddots \end{bmatrix}. \tag{11}$$

Similarly, we project $\mathbf{x}$ onto $\mathbf{M}_L$ and get the coefficient vector, where each term $(x_i - x_{i-1})$ represents the difference between locations of the target during two consecutive time intervals. If the target's trace is smooth enough, the coefficient vector is compressible. Therefore we define the representation basis as $\mathbf{\Psi}_L = \mathbf{M}_L^{-1}$.

5.2. Measurement matrix design

(1) *Spatial measurement matrix:* The spatial measurement matrix $\mathbf{\Phi}$ specifies a measurement policy: its entry $\mathbf{\Phi}(m, n) = 1$ if the mth RSS measurement is taken at the nth cell; otherwise $\mathbf{\Phi}(m, n) = 0$. It is obvious that at most one measurement can be conducted at any cell. This implies, regardless of the measurement policy, $\mathbf{\Phi}$ contains one and only one "1" in any row, at most one "1" in any column, and "0" everywhere else. As a result, the measurement matrix $\mathbf{\Phi}$ will be extremely sparse. This is quite different from existing literatures where the measurement matrices are usually dense. To address this problem, we consider the following two measurement policies.

- *Uniform* measurement policy: measurements are taken periodically. The corresponding measurement matrix is referred to as $\mathbf{\Phi}_U$.
- *Random* measurement policy: measurements are taken randomly. The corresponding measurement matrix is referred to as $\mathbf{\Phi}_R$.

Note that since N is not always an integral multiple of M, in uniform measurement policy, the first measurement point is randomly selected within $[1, \lfloor N/M \rfloor]$, and subsequence measurements are taken every $\lfloor N/M \rfloor$ time intervals till N is exhausted.

(2) *Temporal measurement matrix:* Similar to spatial measurement matrix, temporal measurement matrix is also confronted with physical constraints. Namely, at most one measurement can be conducted at any time interval. It contains one and only one "1" in any row, at most one "1" in any column, and "0" everywhere else. Its entry $\mathbf{\Phi}(s, t) = 1$ implies that the sth measurement is taken at the tth time interval. Therefore, in temporal localization module, we consider the same measurement matrices: $\mathbf{\Phi}_U$ and $\mathbf{\Phi}_R$.

Table 1
Real traces used in mobility characterization.

Trace name	Description	Time interval (s)
KAIST Trace	20 students' movement in KAIST	10
NCSU Trace	32 students' movement in NCSU	10
NewYork Trace	12 volunteers' movement in NewYork	10
Orlando Trace	8 volunteers' movement in Orlando	10

5.3. Performance analysis

As we discussed early, there are two main criteria in selecting representation basis Ψ and measurement matrix Φ: (1) the coefficient vector obtained by representing the original signal in Ψ should be as sparse or compressible as possible, and (2) the coherence between Ψ and Φ should be as low as possible. In this section, we first evaluate the capabilities of designed representation bases to sparsify signals, and then check the coherence between representation bases and measurement matrices.

(1) *Sparsity:* We first evaluate the capabilities of temporal representation bases to sparsify signals. The reason why we ignore spatial representation basis lies in the fact that it can sparsify the RSS vector fully and the sparsity level equals the number of targets. It turns out that the coefficient vectors yielded by temporal representation bases are not precisely sparse but compressible, *i.e.*, most of the elements are non-zero but small enough. Accordingly, we can approximate these coefficient vectors by neglecting the small coefficients. We introduce a metric to evaluate the capabilities of the representation bases to sparsify signals. The metric is defined as the fraction of total energy captured by the top K coefficients, *i.e.*, $(\sum_{i=1}^{K} s_i^2)/(\sum_{i=1}^{N} s_i^2)$, where s_i is the ith largest coefficient and $\sum_{i=1}^{N} s_i^2$ denotes total energy of the coefficient vector. The larger the metric is, the better the representation bases sparsify signals.

The data sets we use are obtained from CRAWDAD [42], including a number of publicly available traces as listed in Table 1. The KAIST (Korea Advanced Institute of Science and Technology) traces were collected by 20 students who carried the GPS receivers in the computer science department. The NCSU (North Carolina State University) traces were taken by 32 students who lived in a campus dormitory. The New York traces were obtained from 12 volunteers living in Manhattan or its vicinity. The Orlando traces were collected from 8 volunteers who visited a local state fair. In addition, it is worth mentioning that all of these traces are obtained from a two-dimensional Euclidean space. However, for the sake of space, we abstract and only consider x-coordinates as the same method can be applied to y-coordinates.

Fig. 3 plots the capabilities of Ψ_S and Ψ_L to sparsify these mobility traces. As we can see, most energy is captured by the top few coefficients in all traces. For example, the top 20 coefficients capture 62.5–85.2% energy when Ψ_S is used. These results clearly suggest that these mobility traces can be sparsely approximated well. Another important observation is that Ψ_S is much more effective than Ψ_L at sparsifying these traces as we can expect.

(2) *Coherence:* Now we check the coherence between temporal representation bases and measurement matrices. Since the concept of coherence is not defined for non-orthogonal matrices, we will use its dual-incoherence [43] to indirectly evaluate coherence. The incoherence between Ψ and Φ is calculated as follows:

$$I(\Phi, \Psi) = \min_{1 \le i \le M} \|\zeta_i\|_0, \tag{12}$$

where $\boldsymbol{\zeta}_i$ is the coefficient vector by projecting the ith row of Φ onto the space spanned by the columns of Ψ:

$$\zeta_i = (\Psi^{\mathrm{T}}\Psi)^{-1}\Psi^{\mathrm{T}}\varphi_i^{\mathrm{T}}. \tag{13}$$

The larger this metric is, the more incoherent the two matrices are.

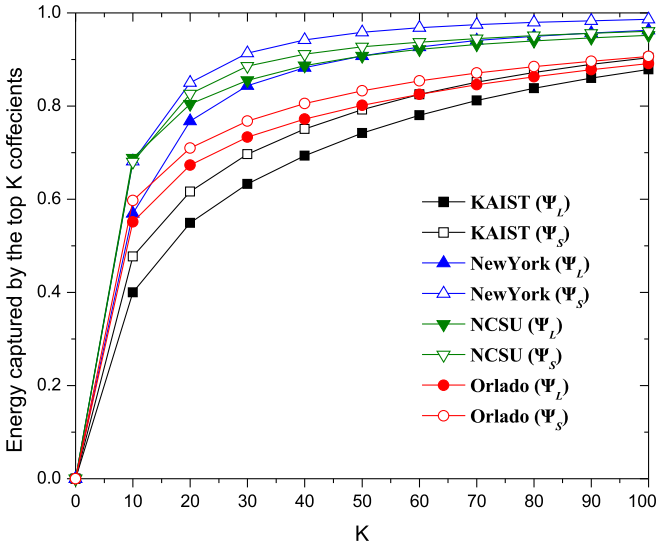

Fig. 3. Sparsity levels of traces in different temporal representation bases.

Table 2
Incoherence between Ψ and Φ.

N	$I(\Psi_L, \Phi_U)$	$I(\Psi_S, \Phi_U)$	$I(\Psi_L, \Phi_R)$	$I(\Psi_S, \Phi_R)$
200	188	199	184	199
400	386	399	385	400
600	555	599	561	600
800	764	800	765	800
1000	973	999	979	1000

Table 2 lists the incoherence of four combinations of temporal representation bases and measurement matrices at different scales. As we can see, when the measurement matrix is fixed, Ψ_S performs much better than Ψ_L. However, when the representation basis is fixed, Φ_U and Φ_R show nearly same incoherence. In the next section we will conduct evaluations to validate these results.

6. Numerical evaluations

6.1. Performance of SLM

In this section, we examine the performance of spatial localization module. In our simulation, the energy decay matrix is produced according to the empirical model defined by the IEEE 802.15.4 standard [44]:

$$RSS(d) = \begin{cases} P_t - 40.2 - 20\log d, & d \leqslant 8 \\ P_t - 58.5 - 33\log d, & d > 8 \end{cases} \tag{14}$$

where P_t denotes transmitting power, and is set to be 30 dBm.

We randomly deploy $k = 10$ targets in a two-dimensional region with the size of 20 m × 20 m. The region is uniformly divided into a grid with 20 × 20 cells. $M = 80$ sensors are deployed to collect RSSs. All localization results are averaged over 100 random trials. Once sparse vector $\mathbf{c}$ is determined, we choose indexes of the top k elements in $\mathbf{c}$ as the numbers of cells which contain targets. The targets' locations are estimated as the positions of these cells. Spatial localization error (*SpaLocError*) is used to evaluate localization performance,

$$SpaLocError = \frac{1}{k}\sum_{i=1}^{k}\sqrt{(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2} \tag{15}$$

where (x_i, y_i) and $(\hat{x}_i, \hat{y}_i)$ are true and estimated locations of the ith target, respectively.

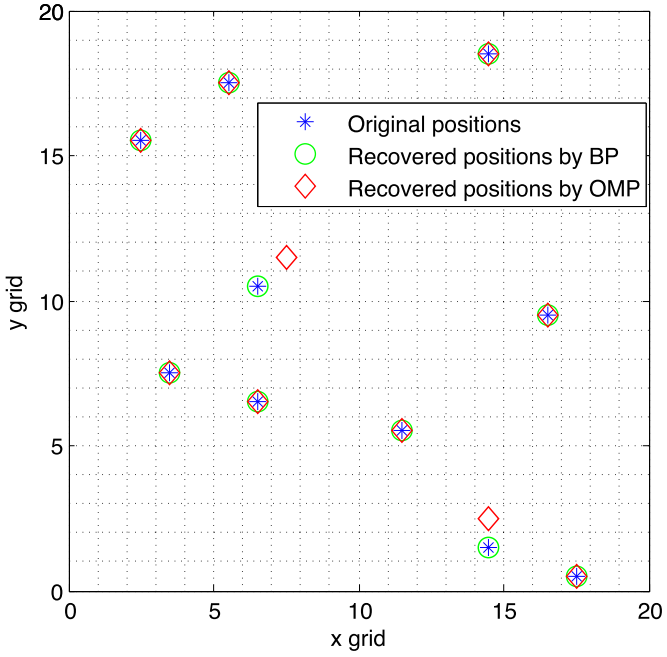

Fig. 4. Spatial localization results of different recovery algorithms.

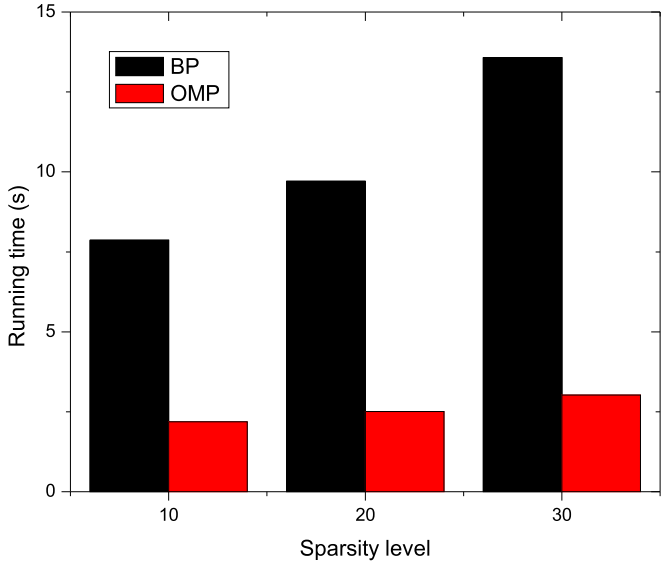

Fig. 5. Running times over 100 random trials of different recovery algorithms.

We first investigate which solver works better when spatial measurement matrix is chosen as $\mathbf{\Phi}_U$. The solvers we consider include BP and OMP. The Matlab codes of two solvers can be obtained from Sparse Lab [42].

The results in Fig. 4 show that BP performs better than OMP. BP achieves accurate localizations for all targets while OMP only localizes partial targets. The reason lies in the fact that BP applies linear programming to the optimization problem while OMP iteratively identifies a cell which contributes most to the measurements.

The running times over 100 random trials of two solvers with different sparsity levels (target numbers) are shown in Fig. 5. As we can expect, for both solvers, running times increase with the increasing of sparsity levels. Another observation is that the running time of BP is higher than OMP while BP achieves higher localization accuracy.

The effect of measurement policy on localization performance is shown in Fig. 6. In the figure, BPU means BP solver with uniform

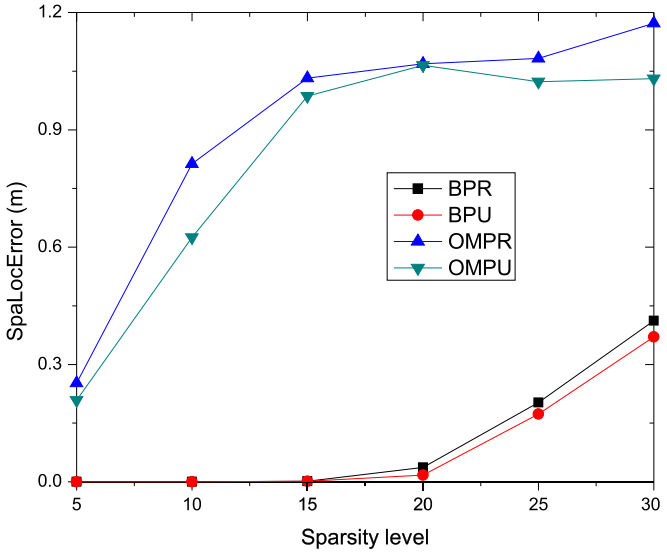

Fig. 6. Spatial localization errors with different measurement policies.

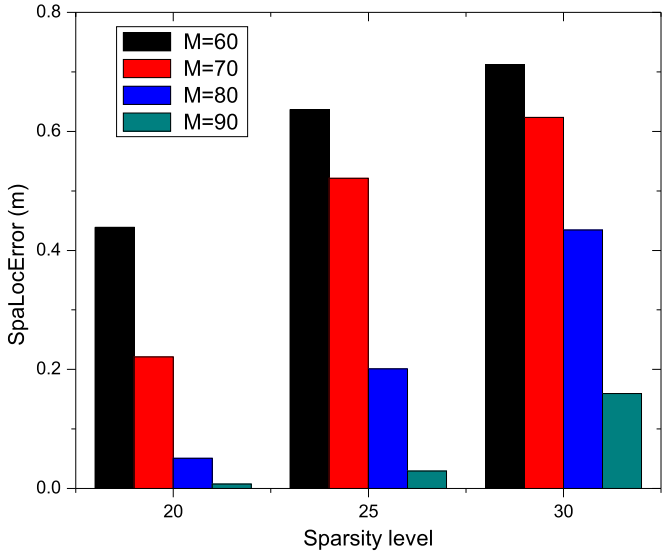

Fig. 7. Spatial localization errors with different measurement numbers.

measurement policy, and the others are similar. As we can see, when the same solver is used, the localization errors with uniform measurement policy are smaller than those with random measurement policy. Therefore, unless otherwise specified, we adopt uniform measurement policy in the later simulation. Furthermore, for both solvers, the localization errors increase with the increasing of sparsity level. The results further confirm the superior performance of BP over OMP. For the sake of space, we will limit our attention to BP in the rest of our numerical evaluations.

Fig. 7 reports the localization errors when measurement number varies from 60 to 90 at a step of 10 under different sparsity levels. The localization error increases as sparsity level increases and decreases with the increasing of measurement number. This is reasonable as higher sampling rate leads to higher accuracy in sparse signal recovery. Specially, when $k = 30$ and $M = 60$, a maximum localization error of 0.74m is observed, which is smaller than the resolution of grid. Therefore, we conclude that SLM is an effective localization algorithm.

It is inevitable for measurements to be corrupted with environmental noise. In order to check the robustness of spatial localization

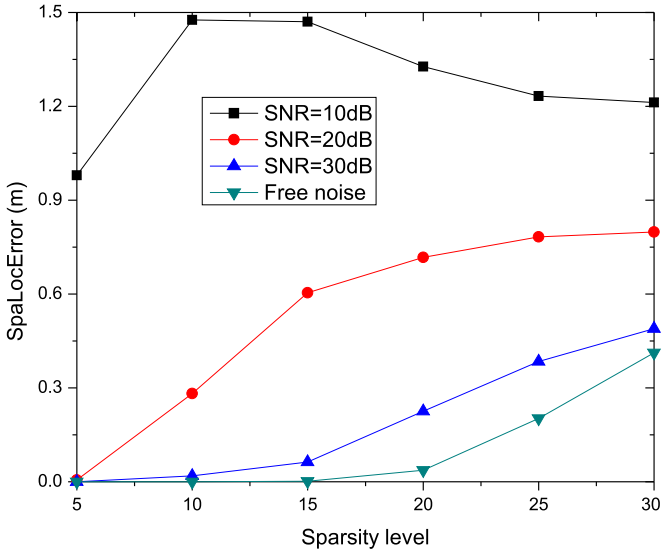

Fig. 8. Spatial localization errors with different noise levels.

module, we intentionally add Gaussian white noise $N(0, \sigma^2)$ to each measurement. We define Signal Noise Ratio (SNR) as $SNR = 20\log(||\mathbf{r}||_2/\sigma)$, which is set to be 10 dB, 20 dB and 30 dB respectively in our simulations. Fig. 8 reports the localization results under different noise levels when $M = 80$ and k varies from 5 to 30 at a step size of 5. The main observations are as follows.

Firstly, Fig. 8 presents a trend that *SpaLocError* increases when measurement noise increases. This is reasonable as signal recovery error is proportional to measurement noise.

Secondly, SLM can tolerate a certain level of measurement noise. In the case of $SNR = 20$ dB, as the sparsity level increase up to 30, the localization error is less than 0.8 m. Therefore SLM is robust to slight measurement noise.

Thirdly, when $SNR = 10$ dB, the error first increases and then decreases with the increasing of sparsity level. The reason might be that we localize targets as the positions of cells corresponding to the top k largest elements in $\mathbf{c}$. When sparsity level is small, signal recovery error increases with the increasing of sparsity level, leading to the increasing of *SpaLocError*. However, when sparsity level is too large, though signal recovery error increases, it is possible that more targets are localized successfully, resulting in the decrease of *SpaLocError* which is averaged over all targets in our paper.

6.2. Performance of TLM

In this section we conduct simulations to evaluate effectiveness of temporal localization module using the matrices designed in previous section.

We consider the real mobility traces listed in Table 1. To keep consistent with the size of grid in spatial localization module, we scale down the real area to be a square of length 20 m. In addition, for the sake of simplicity, we fix the signal length as 1000 by preprocessing these traces as follows: (1) remove the traces with length less than 1000; (2) abstract and only regard the first 1000 samples as original signals. Temporal localization error (*TemLocError*) is used to evaluate the performance of temporal localization module.

$$TemLocError = \frac{1}{k \cdot T}\sum_{i=1}^{T}\sum_{j=1}^{k}\sqrt{\left(x_{ij}-\hat{x}_{ij}\right)^2+\left(y_{ij}-\hat{y}_{ij}\right)^2} \tag{16}$$

where k is the number of targets; T is the number of time intervals; (x_{ij}, y_{ij}) and $(\hat{x}_{ij}, \hat{y}_{ij})$ are the true and estimated locations of the jth target at the i-th time interval, respectively.

We first study the localization performance of BP solver with different combinations of representation basis and measurement matrix. Fig. 9 summarizes the results with four real mobility traces. In the figure, LU means the localization approach where $\mathbf{\Psi}_L$ is chosen as representation basis and $\mathbf{\Phi}_U$ is chosen as measurement matrix, the others are similar. The main observations are as follows.

Firstly, as we can see, the localization errors of all traces are acceptable compared to the area which is scaled down to be a square of length 20 m. Among these traces, NewYork traces achieve minimum localization error. The main reason for the performance difference lies in the fact that the coefficient vectors of NewYork traces are the sparsest among all traces as can be seen in Fig. 3. For the sake of space, we will only consider NewYork traces when mentioning real mobility traces in the rest of our numerical evaluations.

Secondly, when measurement matrix is fixed, representation basis $\mathbf{\Psi}_S$ shows a slight advantage over $\mathbf{\Psi}_L$ regardless of the traces used. The reason for this is twofold. On one hand, compared with $\mathbf{\Psi}_L$, $\mathbf{\Psi}_S$ performs better in sparsifying the traces as shown in Fig. 2. On the other hand, $\mathbf{\Psi}_S$ shows much higher incoherence with measurement matrix than $\mathbf{\Psi}_L$ as shown in Table 2.

Thirdly, when representation basis is fixed, measurement matrix $\mathbf{\Phi}_U$ outperforms $\mathbf{\Phi}_R$ regardless of the traces used. This seems very surprising since they have almost same incoherence with representation basis as shown in Table 2. It is possibly due to the fact that the sampling points in uniform measurement policy spread fully around the signals and nearly capture all information of the signals while the sampling points in random measurement policy may focus on one segment and only capture partial information of the signals.

Then, we investigate the performance of temporal localization module for single target. Fig. 10 illustrates a comparison between true trace and estimated trace of a randomly chosen target. In this simulation, representation basis is chosen as $\mathbf{\Psi}_S$ and measurement matrix is chosen as $\mathbf{\Phi}_U$. The temporal measurement number $S = 100$. As we can see form Fig. 10, temporal localization module reconstructs the target's trace to very high accuracy.

In order to check the robustness of temporal localization module, we intentionally add Gaussian white noise $N(0, \sigma^2)$ to each measurement. The SNR is set to be 10 dB, 20 dB and 30 dB respectively. Fig. 11 reports the localization errors of BP solver under different measurement noise levels. Representation basis is chosen as $\mathbf{\Psi}_S$ and measurement matrix is chosen as $\mathbf{\Phi}_U$. The measurement number varies from 80 to 200 at a step size of 20. The main observations are as follows.

Firstly, Fig. 11 presents a trend that the *SpaLocError* increases as measurement noise increases. This is reasonable as the signal recovery error is proportional to measurement noise.

Secondly, when noise is slight or free, the *SpaLocError* decreases as measurement number increases. However, it goes against the trend when noise is high, for example when SNR = 10 dB. The reason accounting for this phenomenon is that the measurements are quite inaccurate when noise level is high.

Thirdly, we can see that TLM can tolerate a certain level of measurement noise. When SNR is more than 20 dB, the localization errors are less than 0.1 m. The localization errors are less than 0.5m even when SNR = 10 dB. Therefore TLM is very robust to measurement noise.

6.3. Performance of TDL

In this section, we conduct simulations to demonstrate the effectiveness of our two dimensional localization framework. The metric to evaluate its performance is average localization error (*AvgLocError*) which is the localization error averaged over all time intervals and all targets.

We first investigate the effect of spatial measurement number M and temporal measurement number S on the localization performance of TDL. As we can see from Fig. 12, *AvgLocError* sharply

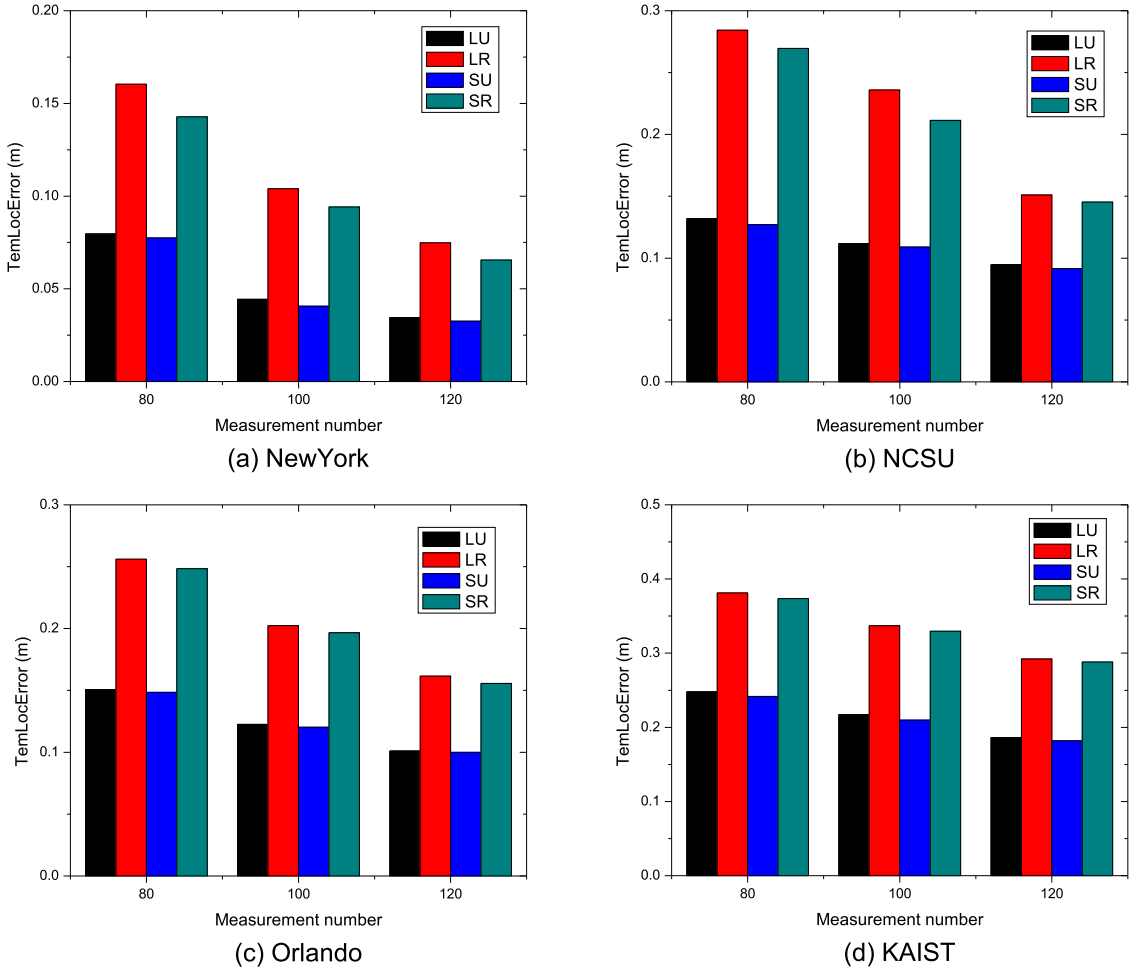

Fig. 9. Temporal localization errors with different matrix combines.

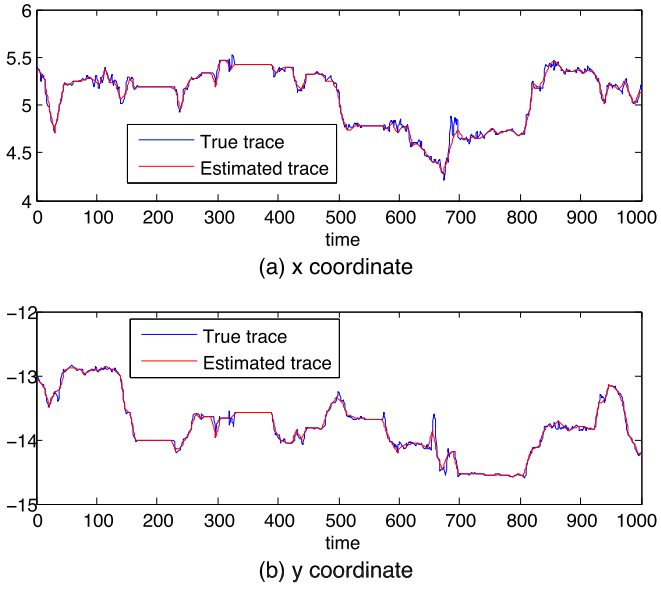

Fig. 10. Comparison between true and estimated traces.

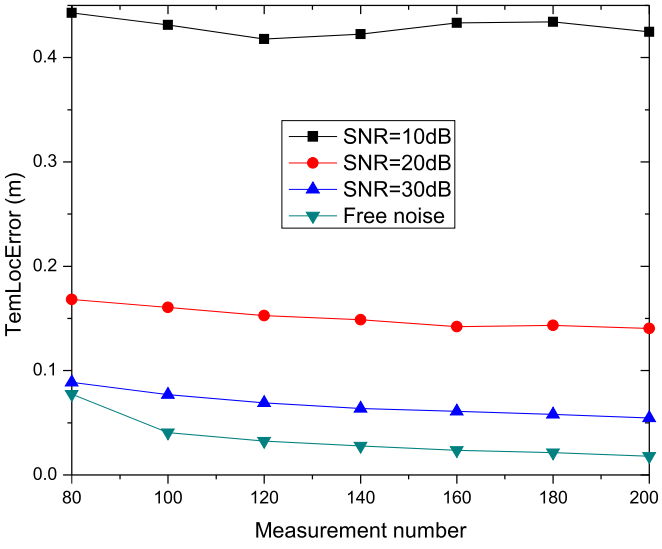

Fig. 11. Temporal localization errors with different noise level.

decreases with the increasing of spatial measurement number and slowly decreases with the increasing of temporal measurement number. Thus, to achieve high localization accuracy at low cost, we should appropriately increase spatial measurement number and decrease temporal measurement number as far as possible.

To further validate the effectiveness of TDL, we compare its performance with the performance yielded by spline interpolation

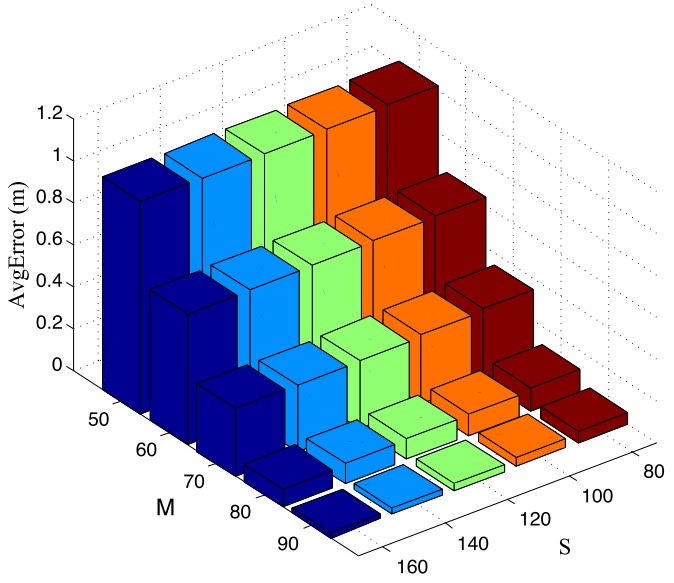

Fig. 12. Average localization errors with different measurement number.

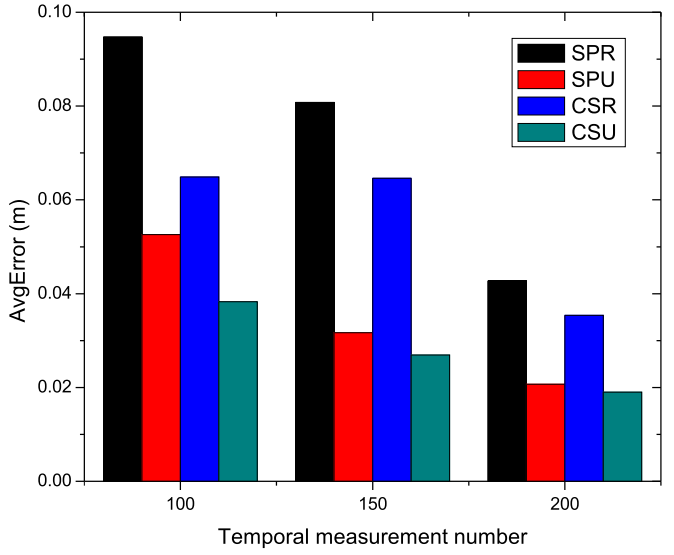

Fig. 13. Average localization error comparison between TDL and SP.

(referred to as SP later), a simple mathematic tool to reconstruct original signals using a small set of samples.

Fig. 13 demonstrates the performance gap between TDL and SP when the spatial measurement number is fixed as 80. The temporal representation basis is chosen as Ψ_S. In the figure, SPR means spline interpolation with random measurement policy, and the others are similar. The key observation is that TDL performs significantly better than SP for small temporal measurement number S. When S is large, the performance of TDL is slightly better than SP. The reason for the performance difference lies in the fact that TDL exploits the sparse nature of original signals while SP does not.

7. Discussion

Improvements on existing work: Energy is an important consideration for wireless sensor networks. We improve on existing work on mobile target localization by reducing energy cost. Signal recovery could consume a significant amount of energy, therefore our localization framework transfers it to the FC. Furthermore, spatial localization module effectively reduces the number of sensors that need to be deployed while temporal localization module extremely reduces the frequency of signal sampling for each sensor, significantly prolonging the lifetime of WSNs.

Limitations of our work: On one hand, our approach is limited by unpredictable environments because RSSs are used for location estimation. For example, in highly dynamic environments, our approach performs poorly as the radio propagation model used to produce spatial representation basis cannot accurately describe the radio environments any longer. On the other hand, our evaluations are based on human traces. How does TDL perform with fast moving targets, such as buses or taxi? The bad compressible nature of traces may lead TDL to perform poorly with fast moving targets. To address this problem, in the future, we will attempt to design better temporal representation bases to explore the compressible natures of the traces.

8. Conclusion

In this paper, we consider the problem of mobile target localization and develop a novel two-dimensional localization framework using compressive sensing. We design representation bases by exploiting the sparse or compressible nature hidden in the spatial signal (*e.g.*, RSS vector) and the temporal signal (*e.g.*, location vector). We investigate what types of measurement matrices are consistent with the physical constraints and sufficiently incoherent with the representation bases. TDL can significantly reduce spatial and temporal measurement numbers at the cost of small localization error. The effectiveness and robustness of TDL are confirmed through extensive numerical evaluations.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (61571463, 61201217, 61232018, 61272487, 61472445 and 61371124), and in part by Jiangsu Province Natural Science Foundation (BK2011118).

References

[1] B. Karp, H.-T. Kung, GPSR: greedy perimeter stateless routing for wireless networks, in: Proceedings of the 6th Annual International Conference on Mobile Computing and Networking, ACM, 2000, pp. 243–254.
[2] S.M. George, W. Zhou, H. Chenji, M. Won, Y.O. Lee, A. Pazarloglou, R. Stoleru, P. Barooah, Distressnet: a wireless ad hoc and sensor network architecture for situation management in disaster response, IEEE Commun. Mag. 48 (2010) 128–136.
[3] D.C. Steere, A. Baptista, D. McNamee, C. Pu, J. Walpole, Research challenges in environmental observation and forecasting systems, in: Proceedings of the 6th Annual International Conference on Mobile Computing and Networking, ACM, 2000, pp. 292–299.
[4] C. Sharp, S. Schaffert, A. Woo, N. Sastry, C. Karlof, S. Sastry, D. Culler, Design and implementation of a sensor network system for vehicle tracking and autonomous interception, in: Proceeedings of the Second European Workshop on Wireless Sensor Networks, IEEE, 2005, pp. 93–107.
[5] P.S. A ANNEX, Global positioning system standard positioning service signal specification, United States Coast Guard Navigation Center, 1995.
[6] D.L. Donoho, Compressed sensing, IEEE Trans. Inf. Theory 52 (2006) 1289–1306.
[7] R.G. Baraniuk, Compressive sensing, IEEE Signal Process. Mag. 24 (2007) 128–133.
[8] D. Baron, M.B. Wakin, M.F. Duarte, S. Sarvotham, R.G. Baraniuk, Distributed compressed sensing, IEEE, preprint 52 (2006) 5406–5425.
[9] W.U. Bajwa, J.D. Haupt, A.M. Sayeed, R.D. Nowak, Joint source–channel communication for distributed estimation in sensor networks, IEEE Trans. Inf. Theory 53 (2007) 3629–3653.
[10] W. Bajwa, J. Haupt, A. Sayeed, R. Nowak, Compressive wireless sensing, in: Proceedings of the 5th International Conference on Information Processing in Sensor Networks, ACM, 2006, pp. 134–142.
[11] M.F. Duarte, M.B. Wakin, D. Baron, R.G. Baraniuk, Universal distributed sensing via random projections, in: Proceedings of the 5th International Conference on Information Processing in Sensor Networks, ACM, 2006, pp. 177–185.
[12] C. Luo, F. Wu, J. Sun, C.W. Chen, Compressive data gathering for large-scale wireless sensor networks, in: Proceedings of the 15th Annual International Conference on Mobile Computing and Networking, ACM, 2009, pp. 145–156.
[13] C. Luo, F. Wu, J. Sun, C.W. Chen, Efficient measurement generation and pervasive sparsity for compressive data gathering, IEEE Trans. Wirel. Commun. 9 (2010) 3728–3738.

[14] M. Mahmudimanesh, A. Khelil, N. Suri, Balanced spatio-temporal compressive sensing for multi-hop wireless sensor networks, in: Proceedings of the IEEE 9th International Conference on Mobile Adhoc and Sensor Systems (MASS), IEEE, 2012, pp. 389–397.
[15] P. Bahl, V.N. Padmanabhan, Radar: an in-building RF-based user location and tracking system, in: Proceedings of IEEE INFOCOM, 2000, pp. 775–784.
[16] D. Niculescu, B. Nath, Ad hoc positioning system (APS) using AOA, in: Proceedings of IEEE INFOCOM, 2003, pp. 1734–1743.
[17] A. Savvides, C.-C. Han, M.B. Strivastava, Dynamic fine-grained localization in ad-hoc networks of sensors, in: Proceedings of the 7th Annual International Conference on Mobile Computing and Networking, ACM, 2001, pp. 166–179.
[18] K. Yedavalli, B. Krishnamachari, S. Ravula, B. Srinivasan, Ecolocation: a sequence based technique for RF localization in wireless sensor networks, in: Proceedings of the 4th International Symposium on Information Processing in Sensor Networks, IEEE, 2005, pp. 285–292.
[19] N. Bulusu, J. Heidemann, D. Estrin, GPS-less low-cost outdoor localization for very small devices, IEEE Person. Commun. 7 (2000) 28–34.
[20] T. He, C. Huang, B.M. Blum, J.A. Stankovic, T. Abdelzaher, Range-free localization schemes for large scale sensor networks, in: Proceedings of the 9th Annual International Conference on Mobile Computing and Networking, ACM, 2003, pp. 81–95.
[21] Y. Shang, W. Ruml, Y. Zhang, M.P. Fromherz, Localization from mere connectivity, in: Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking & Computing, ACM, 2003, pp. 201–212.
[22] V. Cevher, M. Duarte, R.G. Baraniuk, Distributed target localization via spatial sparsity, in: Proceedings of the European Signal Processing Conference (EUSIPCO), 2008, pp. 1–5.
[23] V. Cevher, P. Boufounos, R.G. Baraniuk, A.C. Gilbert, M.J. Strauss, Near-optimal Bayesian localization via incoherence and sparsity, in: Proceedings of the 8th International Conference on Information Processing in Sensor Networks, IEEE, 2009, pp. 205–216.
[24] C. Feng, S. Valaee, Z. Tan, Multiple target localization using compressive sensing, in: Proceedings of the Global Telecommunications Conference (GLOBECOM), IEEE, 2009, pp. 1–6.
[25] C. Feng, W. Au, S. Valaee, Z. Tan, Compressive sensing based positioning using RSS of WLAN access points, in: Proceedings of IEEE INFOCOM, 2010, pp. 1–9.
[26] C. Feng, W.S.A. Au, S. Valaee, Z. Tan, Received-signal-strength-based indoor positioning using compressive sensing, IEEE Trans. Mob. Comput. 11 (2012) 1983–1993.
[27] B. Zhang, X. Cheng, N. Zhang, Y. Cui, Y. Li, Q. Liang, Sparse target counting and localization in sensor networks based on compressive sensing, in: Proceedings of IEEE INFOCOM, 2011, pp. 2255–2263.
[28] G.K. Guyen, T. Van Nguyen, H. Shin, Learning dictionary and compressive sensing for WLAN localization, in: Proceedings of the Wireless Communications and Networking Conference (WCNC), IEEE, 2014, pp. 2910–2915.
[29] L. Liu, T. Cui, W. Lv, A range-free multiple target localization algorithm using compressive sensing theory in wireless sensor networks, in: IEEE 11th International Conference on Mobile Ad Hoc and Sensor Systems (MASS), IEEE, 2014, pp. 690–695.
[30] T.L. Nguyen, Y. Shin, Multiple target localization in WSNS based on compressive sensing using deterministic sensing matrices, Int. J. Distrib. Sens. Netw. (2015) 1–9.
[31] L. Hu, D. Evans, Localization for mobile sensor networks, in: Proceedings of the 10th Annual International Conference on Mobile Computing and Networking, ACM, 2004, pp. 45–57.
[32] M. Rudafshani, S. Datta, Localization in wireless sensor networks, in: Proceedings of the 6th International Symposium on Information Processing in Sensor Networks, IEEE, 2007, pp. 51–60.
[33] A.W.S. Au, C. Feng, S. Valaee, S. Reyes, S. Sorour, S.N. Markowitz, D. Gold, K. Gordon, M. Eizenman, Indoor tracking and navigation using received signal strength and compressive sensing on a mobile device, IEEE Trans. Mob. Comput. 12 (2013) 2050–2062.
[34] J. Deng, Q. Cui, X. Zhang, Data pre-processing in compressive sensing based indoor fingerprinting positioning, Int. J. Wireless Inf. Netw. 20 (2013) 256–267.
[35] Y. Zhang, Y. Zhu, M. Lu, A. Chen, Using compressive sensing to reduce fingerprint collection for indoor localization, in: Proceedings of IEEE Wireless Communications and Networking Conference (WCNC), IEEE, 2013, pp. 4540–4545.
[36] E.J. Candès, T. Tao, Decoding by linear programming, IEEE Trans. Inf. Theory 51 (2005) 4203–4215.
[37] J.A. Tropp, A.C. Gilbert, Signal recovery from random measurements via orthogonal matching pursuit, IEEE Trans. Inf. Theory 53 (2007) 4655–4666.
[38] E.J. Candès, J. Romberg, T. Tao, Robust uncertainty principles exact signal reconstruction from highly incomplete frequency information, IEEE Trans. Inf. Theory 52 (2006) 489–509.
[39] J. Wang, S. Tang, B. Yin, X.-Y. Li, Data gathering in wireless sensor networks through intelligent compressive sensing, in: Proceedings of IEEE INFOCOM, 2012, pp. 603–611.
[40] G. Zanca, F. Zorzi, A. Zanella, M. Zorzi, Experimental comparison of RSSI-based localization algorithms for indoor wireless sensor networks, in: Proceedings of the Workshop on Real-world Wireless Sensor Networks, ACM, 2008, pp. 1–5.
[41] X. Wu, M. Liu, In-situ soil moisture sensing: measurement scheduling and estimation using compressive sensing, in: Proceedings of the 11th International Conference on Information Processing in Sensor Networks, ACM, 2012, pp. 1–12.
[42] I. Rhee, M. Shin, S. Hong, K. Lee, S. Kim, S. Chong. CRAWDAD dataset ncsu/mobilitymodels (v. 2009-07-23), doi:10.15783/C7X302, 7, 2009. http://crawdad.org/ncsu/mobilitymodels/.
[43] G. Quer, R. Masiero, D. Munaretto, M. Rossi, J. Widmer, M. Zorzi, On the interplay between routing and signal representation for compressive sensing in wireless sensor networks, in: Proceedings of Information Theory and Applications Workshop, 2009, IEEE, 2009, pp. 206–215.
[44] IEEE standard online resource provided by IEEE 802.15 WPAN, 2, 2009. http://www.ieee802.org/15/pub/TG4.html.