Beacon-based Proximity Detection using Compressive Sensing for Sparse Deployment

Pai Chet Ng†, Li Zhu‡, James She†, Rong Ran‡, Soochang Park†
†Department of Electronic and Computer Engineering,
Hong Kong University of Science and Technology; Email: {pcng, eewinter, eejames}@ust.hk
‡School of Electrical and Computer Engineering, Ajou University; Email: {sunnyran, julie}@ajou.ac.kr

***Abstract*—A proximity-based service (PBS) leverages the estimated proximity to provide users the accessibility to object or location restricted service. This paper exploits the interaction between Bluetooth Low Energy (BLE) Beacon and smartphone to set forth the fundamental building block of a beacon-based PBS system. In real-world scenarios, a beacon-based PBS system might suffer from sparse conditions when some beacons malfunction or beacons can only be deployed in a few specific positions. Motivated by such limitations, a similarity filter extended with compressive sampling matching pursuit (*SF-CoSaMP*) is proposed to ensure the reliability of proximity detection under such sparse conditions before smartphone proceed to retrieve the corresponding PBS. An extensive simulation with large volume of collected data has been conducted and the results prove the reliability of the proposed algorithm with high detection accuracy in an environment with sparse deployment.**

I. Introduction

Over the years, various indoor location-based services (LBSs) have been developed to provide users an accessibility to the available service in an indoor environment [1]. Most often, this type of service is location restricted in which users need to provide their location information before they can access to the service. Localization techniques (e.g., trilateration [2] and fingerprinting [3]) have been implemented to derive indoor location information such that diverse LBS applications can be delivered with accurate location estimation, if possible. However, accurate location estimation is still an elusive research issue due to the multipath effects in indoor environments [4]. Despite many notable attempts, most localization techniques are only capable of delivering coarse-grained location estimation up to sub-meter accuracy [5]. Instead of struggling with the accuracy in connection with indoor localization, proximity-based service (PBS) has been employed as a substitution to LBS when the exact geographical location is out of concern [6].

As an alternative, PBS uses the proximity information of a smartphone to deliver diverse services ranging from access authentication [7] [8], on-spot transaction [9] [10], multimedia sharing [11] [12], etc., to users. Besides using RFID as the driving tool for proximity detection [13], the introduction of Bluetooth Low Energy (BLE) beacon has also escalated the development of PBS [14]. Step on the shoulders of emergence BLE Beacon, this paper explores the potential of a BLE Beacon-based PBS system.

978-1-5386-2723-5/17/$31.00

As illustrated in Fig. 1, the core elements of a beacon-based PBS system are a BLE Beacon network, smartphone and server. Here, a beacon-based PBS region defines a region that serves same PBS, this region might consist of a number of proximities of interest (*PoI*s) that deliver same PBS with different contents. For example, a music exhibition, which delivers proximity-based music listening service is a PBS region that consists of a number of *PoI*s, where each *PoI* contains different music genres. In general, we define the notion of *PoI* as an ambient object or location associated with at least one BLE Beacon such that each *PoI* can continuously broadcast their signals to implicitly announce their presence. These signals can be measured by an encountered smartphone to identify the target *PoI* (i.e., the *PoI* that smartphone is closer to). Since PBS involves human in the loop, quality of service (QoS) is the main evaluation criterion for a PBS system. That is, a reliable proximity detection is expected such that the corresponding service can be delivered to the encountered smartphone through correct target *PoI* selection.

The fundamental requirement to achieve a PBS system with high QoS is that the smartphone must be able to see a clear picture regarding the encountered PBS region. This means smartphone must have a complete and comprehensive knowledge about all the signals contributed by all *PoI*s residing in a given region. This is attainable provided each beacon associated with each *PoI* works flawlessly and smartphone can capture all the signals broadcast by all beacons. However, the likelihood to achieve these ideal conditions is rare because some beacons might fail to function due to dead batteries or faulty circuit components as time goes by. Furthermore, it is unrealistic to have a long scanning duration when service response time is of major concern. Lastly, some *PoI*s might need to compromise with no beacon association owing to physical deployment constraint. For example, some buildings might be unable to provide a suitable location to confine the beacon association to designed *PoI*s layout. In this case, these *PoI*s are assumed to announce their presence implicitly by manipulating the beacon information from adjacent *PoI*s. In this situation, smartphone might fail to correctly decide the target *PoI* due to incomplete signals information.

Recently, compressive sensing (CS) has attracted a lot of attention as it can provide a novel framework for recovering signals that are sparse [15]. Their success in indoor positioning systems (cf., in wireless local area network[16] [17], in wire-

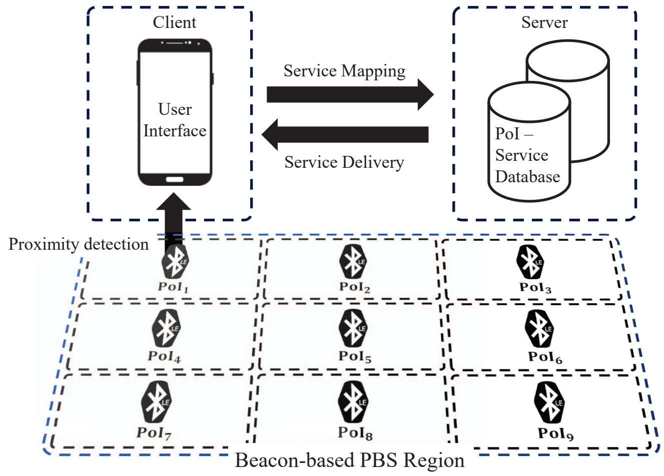

Fig. 1. The fundamental building block for a beacon-based PBS system: a beacon network, a client app, and a server.

less sensor network [18] [19], in RFID network [20] [21]) motivate us to exploit CS algorithms for beacon-based PBS system. To the best of our knowledge, none of the research has leverage CS algorithms for proximity detection, particularly in an indoor environment with sparse deployment. To this end, a CS-based proximity detection algorithm is proposed such that the smartphone is still able to perform a reliable proximity detection without an explicit knowledge of complete signals information. The main contributions of this paper are summarized as follows:

- A beacon-based PBS system model is set forth to describe its fundamental building block and related notions.
- A CS-based proximity detection is proposed to mitigate incomplete signals challenges under sparse deployment.
- A similarity filter is integrated with the CS-based proximity detection to improve the detection reliability.

The rest of the paper is organized as follows. Section II introduces the beacon-based PBS model and formulates the proximity detection problem. Section III presents the similarity filter extended with CS-based proximity detection. Section IV describes the simulation with real data and presents the simulation results. Section V concludes the paper.

II. System Model

This section describes the beacon-based proximity-based service (PBS) system model and follows with a discussion on problem formulation.

A. Beacon-based PBS

The basic building block of a beacon-based PBS system model comprises a region that has been deployed with multiple BLE Beacons and a server to facilitate the service mapping. Any Bluetooth compatible receiver can be used to acquire the signals and estimate its proximity, for simplicity, this paper considers smartphone as the receiver since it is widely available. The role of BLE Beacons is to broadcast their signals to the surrounding such that encountered smartphone can know the proximity of interest (*PoI*) they are closer to after processing these signals. We define an area which serves the same PBS purpose as a region, which might consist of multiple *PoI*s delivering the same service but different content. For example, in a restaurant, each dining table is associated with a beacon (i.e., each dining table represents a *PoI*), the PBS is intended to deliver food menu to customers and each *PoI* has its own meta attributes such as the table number, bill, etc.

Mathematically, these *PoI*s in a same region can be described as a set of vectors,

$$\Omega = \{\Omega_b : b = \{1, 2, ..., n\}\} \tag{1}$$

where b is a set of indexes and n is an integer denoting the total number of *PoI*s for a given beacon-based PBS region. Each vector Ω_b can be characterized by its features such as received signal strength (RSS) or time of arrival (ToA).

In this paper, RSS is used to characterize each $\Omega_b \in \mathbb{R}^n$. That is, given a scanning interval T_s, each Ω_b can be represented with a list of time average RSS contributed by all the associated beacons from all the *PoI*s in a beacon-based PBS region.

$$\Omega_b = \begin{bmatrix} \phi_{\Omega_1} \\ \vdots \\ \phi_{\Omega_n} \end{bmatrix}, \quad \forall b \tag{2}$$

where the time average RSS is defined as $\phi = \frac{1}{N(t_k)} \sum_{t=t_1}^{t_k} R(t)$ with $R(t)$ denotes the RSS measured at discrete time t for all $t_k \leq T_s$. The ϕ_{Ω_b} means the time average RSS contributed by bth *PoI*.

Intuitively the target *PoI* can be identified by examining ϕ in an observed vector Ω_y, where $y \in b$. Since RSS is a measurement in dBm in which its strength decreases logarithmically with increasing distance, in other words, we can roughly estimate the closest *PoI* by simply selecting the strongest ϕ. However, such straight forward selection might fail to return the target *PoI* correctly, especially when those beacons associated with some designated *PoI*s stop functioning.

Consider every possible scenario which might lead to a sparse representation (i.e., the Ω_y can only be represented by far few samples of ϕ), this paper proposes a reliable proximity detection algorithm by extending a similarity filter with conventional compressive sensing such that sufficient information can be recovered before smartphone performs the proximity detection to identify the target *PoI*.

B. Problem Formulation

Given n number of *PoI*s are established in a PBS region, but only the *PoI*s at four corners are associated with beacons, then the proximity detection problem has a sparse nature, as the number of measurements m (i.e., ϕ in the observed vector Ω_y) is smaller than the total number of *PoI*s n. In other words, the observed vector Ω_y has been compressed from size n to size m. Assume that a user carrying his/her smartphone is in close proximity to yth *PoI*, then user's proximity to each

PoI can be formulated as a 1-sparse vector, $\mathbf{b} \in \mathbb{R}^n$, with all elements equal to zero except $\mathbf{b}_y = 1$, where $y \in b$ represents the index of target *PoI*,

$$\mathbf{b} = [0, 0, \cdots, 1, 0, \cdots, 0]^T \tag{3}$$

Since only ϕ from four *PoI*s are measured/observed due to the deployment constraint, the relation between the compressed observed vector $\Omega_y \in \mathbb{R}^m$ and the user's proximity to each *PoI* $\mathbf{b} \in \mathbb{R}^n$ can be described as follows:

$$\Omega_y = \Psi\Phi\mathbf{b} + \mathbf{v} \tag{4}$$

where $\mathbf{v}$ represents the measurement noise. Φ is a $n \times n$ is a radio map as follows:

$$\Phi = \begin{pmatrix} \phi_{1,1} & \phi_{1,2} & \cdots & \phi_{1,n} \\ \phi_{2,1} & \phi_{2,2} & \cdots & \phi_{2,n} \\ \vdots & \vdots & \cdots & \vdots \\ \phi_{n,1} & \phi_{2,2} & \cdots & \phi_{n,n} \end{pmatrix} \tag{5}$$

where $\phi_{i,j}$ is the time average RSS in jth *PoI* contributed from the beacon associated with ith *PoI*. Ψ is a $m \times n$ measurement matrix. Each row of Ψ contains zeros and unit one which indicates the index of *PoI* associated with a functioning beacon.

Note that Eq. (4) is an underdetermined system ($m \leq n$) with 1-sparse vector $\mathbf{b}$. Therefore, the proximity detection is converted into following sparse vector reconstruction problem:

$$\hat{\mathbf{b}} = \min_{\mathbf{b}}(\|\Omega_y - \Psi\Phi\mathbf{b}\|^2 + \lambda\|\mathbf{b}\|_1) \tag{6}$$

where λ is a tuning parameter of CS, and the index of non-zero entry in $\hat{\mathbf{b}}$ refers to the index of target *PoI*.

Remark 1: The CS theory [15] states that as long as the number of observations is greater than $CS \log n$ (i.e., $m \geq CS \log n$), where C is a constant and S is the degree of sparsity, then with very high probability that the solution to the optimization problem given in Eq. 6 is unique and is approximately equivalent to $\mathbf{b}$. In our system model, since $s = 1$, as long as the number of *PoI*s is less than 10000, $m = 4$ is enough to guarantee a reliable signal recovery.

Remark 2: T_s must be long enough to capture a sufficient number of signals from all the *PoI*s in the PBS region. However, it is unrealistic to have a very long T_s when quality of service (QoS) is of concern. Furthermore, long T_s will drain the battery at the smartphone and slow down the receivers processing memory. Therefore, it is necessary to select only a few *PoI*s associated with beacons in order to achieve an elegant trade-off between scanning interval and QoS.

III. Proximity Detection based on Similarity Filter and Compressive Sensing

This paper proposes a similarity filter incorporated with compressive sampling matching pursuit (*SF-CoSaMP*) which first, removes part of interference *PoI*s in advance by comparing the similarities and thus refine the performance of conventional *CoSaMP*. This section describes 3 main components of the proposed algorithm: similarity filter. signal pre-processing and *CoSaMP*.

A. Similarity filter (SF)

Consider only four beacons are associated at four corners, for a $a \times a$ ($a^2 = n$) grid system, we selectively compute the similarities for PoIs in the index subset $\mathcal{H} = \{1, a, 1+(a-1)*a, a*a\}$. Define $s_{\mathcal{H}} = ||\Omega_b - \Psi\phi_{\mathcal{H}}||^2$ as the similarity between observed vector and all $\mathcal{H}$ column vector in Φ, then the filtered index set $\mathcal{S}$ can be constructed accordingly by taking the K elements which meets the similarity threshold s_γ,

$$\mathcal{S} = \{K : s_K < s_\gamma, \quad \forall k \in \mathcal{H}\} \tag{7}$$

The *PoI*s which are belong to $\mathcal{S}$ are filtered out in advance. The resulted subset $\mathcal{C} = b \backslash \mathcal{S}$, which refers to the collection of indexes included by b but excluded by $\mathcal{S}$, is utilized to refine Φ to

$$\hat{\Phi} = \{\Phi_j : \forall j \in \mathcal{C}\} \tag{8}$$

where Φ_j is the column vector of radio map in Eq. (5).

B. Signal Pre-processing

To ensure $\mathbf{b}$ can be recovered from observed vector Ω_y with very high probability, the achieved radio map is transformed into an orthonormal basis $Q \in \mathbb{R}^{n \times n}$ with following operation,

$$Q = orth((\Psi\Phi)^T)^T \tag{9}$$

Besides that, for each observed vector Ω_y, a signal pre-processing operator $T = Q(\Psi\Phi)^\dagger$ is introduced to perform a corresponding transformation on Ω_y,

$$\Omega'_y = T\Omega_y \tag{10}$$

By substituting Eq. (9) and Eq. (10) into Eq. (6), the sparse vector reconstruction problem is further refined to

$$\hat{\mathbf{b}} = \min_{\mathbf{b}}(\|\Omega'_y - Q\mathbf{b}\|^2 + \lambda\|\mathbf{b}\|_1) \tag{11}$$

C. Compressive Sampling Matching Pursuit (CoSaMP)

Before the smartphone can select the target *PoI*, a recovery process is performed by integrating the conventional compressive sampling matching pursuit (CoSaMP) with the previous two operations (i.e., similarity filter and signal preprocessing). In general, *CoSaMP* is a greedy algorithm which iteratively computes the proxy, merges the strongest support, estimate the signal, prunes the least entries and updates the residual until a stopping threshold is reached. Assume that the signal proxy at current iteration is $\rho^{(i)} = Q \times r^{(i-1)}$, where r is the residual from previous iteration and $r^{(0)} = \Omega'_y$, then the merging operation of the strongest support sets is,

$$\nu^{(i)} = supp(\rho_{2s}^{(i)}) \cup supp(\tilde{\mathbf{b}}^{(i-1)}) \tag{12}$$

where $\tilde{\mathbf{b}}^{(i-1)}$ is the estimated 1-sparse vector at $(i-1)$th iteration and s is the degree of sparsity.

Least square is used to estimate the best possible $\tilde{\mathbf{b}}^{(i)}$ at current iteration:

$$\mathbf{b}^{(i)} = Q^\dagger_{|\nu} r^{(i-1)} \tag{13}$$

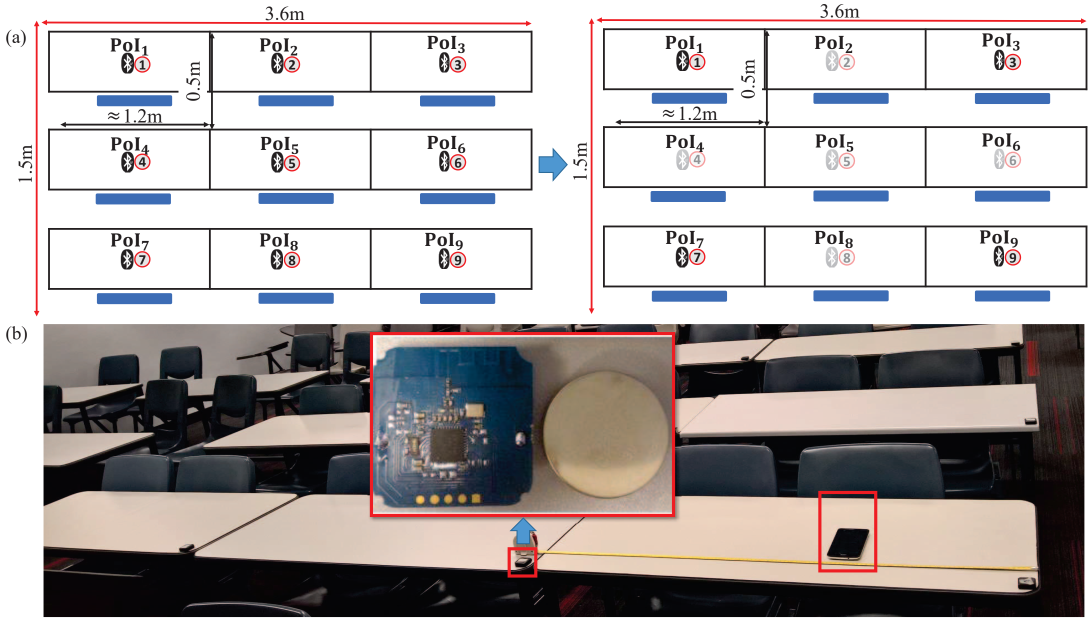

Fig. 2. A classroom testbed with 9 tables, where each table represents one *PoI* with one associated beacon: (a) The battery of 5 beacons (shaded in grey) were taken out to emulate a sparse environment, (b) Signal collection process on the smartphone, and (c) the beacon and its battery

where $Q^{\dagger}_{|\nu}$ is the pseudo-inverse matrix of $Q_{|\nu}$ which consists only the column vector at the index identified by $\nu^{(i)}$

The resultant $\mathbf{b}^{(i)}$ is pruned according to the degree of sparsity s, such that the current estimation is $\tilde{\mathbf{b}}^{(i)} = \mathbf{b}_s$, and then the residual is updated for next iteration, i.e.,

$$r^{(i)} = r^{(i-1)} - Q\tilde{\mathbf{b}}^{(i)} \tag{14}$$

In general, the proximity detection algorithm based on the integrated similarity filter and compressive sampling matching pursuit (*SF-CoSaMP*) can be summarized as follows:

Algorithm 1 *SF-CoSaMP*

Input: sparsity level s, radio map Φ, index set b, similarity threshold s_γ and observed vector Ω_y

Output: filtered index set $\mathcal{C}$, estimated 1-sparse vector $\tilde{\mathbf{b}}$, and estimated target index y

Initialization : $i = 1$, $\tilde{\mathbf{b}}^{(i-1)} = 0$, $r^{(i-1)} = \Omega_y$

1: Compute similarity set $\mathcal{S}$ using Eq. (7)

2: Obtain filtered index set $\mathcal{C} = b \setminus \mathcal{S}$

3: Refine radio map according to Eq. (8)

4: Transform radio map to Q using Eq. (9)

5: Transform observed vector to Ω'_y using Eq. (10)

while Stopping threshold is not met **do**

  6: Compute current signal proxy, $\rho^{(i)} = Q \times r^{(i-1)}$

  7: Identify top $2s$ components $supp(\rho^{(i)}_{2s})$

  8: Merge the strongest support sets using Eq. (12)

  9: Estimate signal using least square Eq. (13)

  10: Prune $\tilde{\mathbf{b}}^{(i)}$

  11: Update $r^{(i)}$ using Eq. (14) for next iteration

  12: Increase iteration by 1

end while

IV. Simulations and Results

The reliability of the proposed *SF-CoSaMP* was analyzed through simulations. First, a classroom with total 9 *PoI*s was established in which each *PoI* is associated with a beacon, all *PoI*s are arranged in an equal rectangular grid as illustrated in Fig. 2(a). All beacons are built with CC2541 BLE chips and powered by CR2450 $3V$ lithium coin cell battery, as shown in Fig. 2(b). A smartphone was used to collect the signals during an offline phase to construct and calibrate the radio map, as depicted in Fig. 3. Next, the batteries in beacons associated with PoI_2, PoI_4, PoI_5, PoI_6 and PoI_7 were purposely taken out to emulate the sparse environment, and similar signal acquisition process was repeated. The following subsection describes the baseline and performance metrics used to compare the simulation results.

A. Baseline

Two baselines were used for comparison; one was the conventional method provided as open source by most BLE Beacon manufacturers such as Altbeacon [1] and Estimote [2], this method has been used in most commercial applications. Another baseline is the classical Compressive Sampling Matching Pursuit (*CoSaMP*) [22]. The conventional method simply examines observed vector Ω_y and selects the *PoI* which contributes the strongest RSS value; whereas *CoSaMP* is a greedy algorithm that iteratively recover the signal coefficient which has the strongest correlation with the sparse measurement.

[1]"The Open and Interoperable Proximity Beacon Specification", "http://altbeacon.org/"

[2]"Estimote Beacons Real World Context for your Apps", "http://estimote.com/"

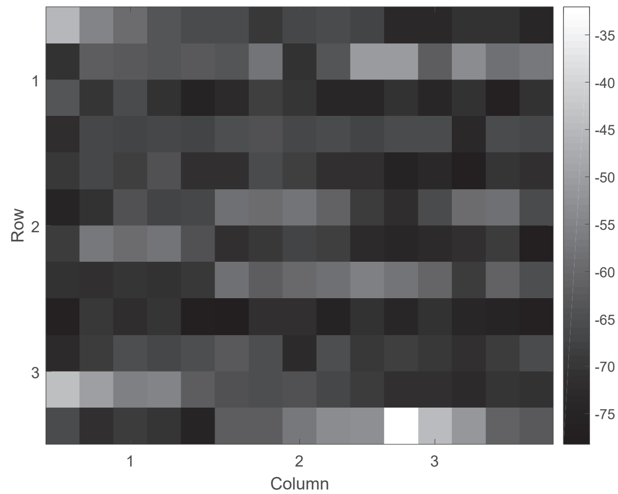

Fig. 3. Radio map for the deployed testbed with row 1 column 1 indicates PoI_1, row 2 column 1 indicates PoI_4 and so on.

1) Maximum RSS Selection: Given the number of RSS values collected from different beacons during a scanning interval T_s, the mean RSS values contributed by ith *PoI* can be expressed as follows:

$$\overline{RSS_i} = \frac{1}{T_s} \sum_{t=t_1}^{t_k} RSS_i(t) \tag{15}$$

where t_k is the time to receive kth signal and $t_k \leq T_s$.

Since a PBS region often consists multiple *PoI*s, the contribution from all *PoI*s can be represented by a vector, i.e., $\overline{\mathbf{RSS}} = [\overline{RSS_1}, \overline{RSS_2}, ..., \overline{RSS_n}]$. With this vector, smartphone can select the target *PoI* which contributes the maximum mean RSS value,

$$\tilde{i} := \arg\max_{i \in n} \{\overline{\mathbf{RSS}}\} \tag{16}$$

where $\tilde{i}$ indicates the index of *PoI* selected by the smartphone. To facilitate the following discussion, we named this conventional method as *maxRSS*.

2) Compressive Sampling Matching Pursuit: Compressive Sampling Matching Pursuit (*CoSaMP*), on the other hand, identifies the locations of the largest influence from the residual induced by the signal approximation at current iteration. The 5 steps of *CoSaMP* (i.e., computes the proxy, merges the strongest support, estimate the signal, prunes the least entries and updates the residual) are repeated until a stopping threshold is reached. The process is similar to the algorithm described in Algorithm 1 but without the similarity filter.

B. Performance Metrics

Given the total number of *PoI*s established in a PBS region, detection accuracy is used to evaluate the number of times where the smartphone selects the target *PoI* correctly,

$$A_{D,PoI_i} = \frac{1}{N} \sum_{n=1}^{N} O_n, \quad \forall i = \{1, 2, ..., n\} \tag{17}$$

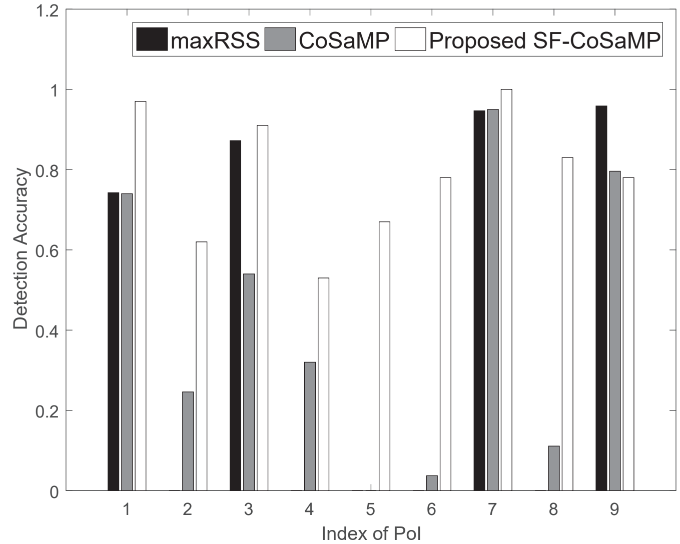

Fig. 4. Detection accuracy at each *PoI*

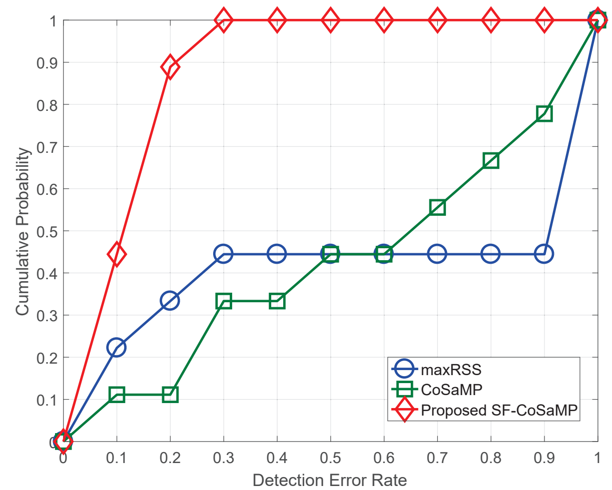

Fig. 5. The cumulative probability for detection error rate of all the implemented algorithms

where N is the total number of detections and O_n is the nth detection output with,

$$O = \begin{cases} 1, & \text{if } \hat{b} = \text{index of target PoI} \\ 0, & \text{if } \hat{b} \neq \text{index of target PoI} \end{cases} \tag{18}$$

Furthermore, the overall error rate resulted given each *PoI* is calculated as follows:

$$P(e|PoI_i) = \frac{1 - A_{D,PoI_i}}{\sum_{i=1}^{n}(1 - A_{D,PoI_i})} \tag{19}$$

where A_{D,PoI_i} is the detection accuracy for ith *PoI* and $\sum_{i=1}^{n} P(e|PoI_i) = 1$.

C. Simulation Result

The following sections discuss the simulation results based on the two performance metrics described in Section B.

1) Detection Accuracy: Referring to Fig. 5, all methods are able to maintenance a high detection accuracy in the *PoI*s where the beacons are still working. And for *PoI*s without beacons (i.e., PoI_2, PoI_4, PoI_5, PoI_6 and PoI_7), the detection accuracy degrades for both *maxRSS* and *CoSaMP*. In general, *maxRSS* is unable to produce even one correct detection at these *PoI*s; whereas *CoSaMP* still can produce a few correct detections. However, with the proposed *SF-CoSaMP*, a significant improvement in the detection performance is observed. Fig. 4 shows that *SF-CoSaMP* achieves an average detection accuracy up to 60% - 70% at these *PoI*s where beacons are not working compared to *CoSaMP* with average detection accuracy up to 20% and *maxRSS* 0%.

2) Detection Error Rate: The detection error rates for all the implemented algorithms are compared and presented in Fig. 5. It is clear that after applying the similarity filter to remove some interference *PoI*s, a much lower error rate is guaranteed. Even though *SF-CoSaMP* is proposed for the current grid system configuration with four beacons being set at the four corners respectively, a general similarity filtering algorithm, however, is needed for arbitrary system configurations. Besides, the similarity is measured according to Euclidean Distance, and may not be the optimal. All of these should be studied in future work. Overall, *SF-CoSaMP* shows a superior performance with a low error rate compared to conventional *maxRSS* and classical *CoSaMP* algorithms under a sparse condition.

V. Conclusions

The synergy between smartphone and BLE Beacons is a key enabler to proximity-based service (PBS). PBS ensures a secure service through a reliable proximity detection in an environment with sparse beacon deployment, which is an unavoidable condition when some beacons fail to function or smartphone is unable to acquire explicit signals information due to deployment constraint. Under this condition, the likelihood to achieve high detection accuracy is low, especially in the proximity of interest (*PoI*) where the associated beacon is not working. To address the sparsity challenge, this paper proposes a proximity detection algorithm by adopting a similarity filter into Compressive Sampling Matching Pursuit (*SF-CoSaMP*). Through extensive simulations with real data, the proposed algorithm achieves high detection accuracy (i.e., approximately 70%) compared to conventional proximity detection.

Acknowledgment

This work was supported by HKUST-NIE Social Media Laboratory. And all the beacon equipment are sponsored by CyPhy Media Limited.

References

[1] M. Werner, "Indoor location-based services," *Prerequisites and foundations. Cham: Springer*, 2014.

[2] Z. Li, T. Braun, and D. C. Dimitrova, "A passive wifi source localization system based on fine-grained power-based trilateration," in *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2015 IEEE 16th International Symposium on a*. IEEE, 2015, pp. 1–9.

[3] Q. Jiang, Y. Ma, K. Liu, and Z. Dou, "A probabilistic radio map construction scheme for crowdsourcing-based fingerprinting localization," *IEEE Sensors Journal*, vol. 16, no. 10, pp. 3764–3774, 2016.

[4] X. Li, "Rss-based location estimation with unknown pathloss model," *IEEE Transactions on Wireless Communications*, vol. 5, no. 12, pp. 3626–3633, 2006.

[5] S. Sorour, Y. Lostanlen, S. Valaee, and K. Majeed, "Joint indoor localization and radio map construction with limited deployment load," *Mobile Computing, IEEE Transactions on*, vol. 14, no. 5, pp. 1031–1043, 2015.

[6] M. Papandrea, S. Giordano, S. Vanini, and P. Cremonese, "Proximity marketing solution tailored to user needs," in *World of Wireless Mobile and Multimedia Networks (WoWMoM), 2010 IEEE International Symposium on a*. IEEE, 2010, pp. 1–3.

[7] M. Portnoi and C.-C. Shen, "Location-aware sign-on and key exchange using attribute-based encryption and bluetooth beacons," in *Communications and Network Security (CNS), 2013 IEEE Conference on*. IEEE, 2013, pp. 405–406.

[8] T. Higuchi, P. Martin, S. Chakraborty, and M. Srivastava, "Anonycast: privacy-preserving location distribution for anonymous crowd tracking systems," in *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 2015, pp. 1119–1130.

[9] T. Halevi, D. Ma, N. Saxena, and T. Xiang, "Secure proximity detection for nfc devices based on ambient sensor data," in *European Symposium on Research in Computer Security*. Springer, 2012, pp. 379–396.

[10] M. Choi, J. Lee, S. Kim, Y.-S. Jeong, and J.-H. Park, "Location based authentication scheme using ble for high performance digital content management system," *Neurocomputing*, vol. 209, pp. 25–38, 2016.

[11] J. Visca, R. Fuentes, A. R. Cavalli, and J. Baliosian, "Opportunistic media sharing for mobile networks," in *Network Operations and Management Symposium (NOMS), 2016 IEEE/IFIP*. IEEE, 2016, pp. 799–803.

[12] B. Xing, K. Seada, and N. Venkatasubramanian, "Proximiter: Enabling mobile proximity-based content sharing on portable devices," in *Pervasive Computing and Communications, 2009. PerCom 2009. IEEE International Conference on*. IEEE, 2009, pp. 1–3.

[13] M. Bolic, M. Rostamian, and P. M. Djuric, "Proximity detection with rfid: A step toward the internet of things," *IEEE Pervasive Computing*, vol. 14, no. 2, pp. 70–76, 2015.

[14] R. Tabata, A. Hayashi, S. Tokunaga, S. Saiki, M. Nakamura, and S. Matsumoto, "Implementation and evaluation of ble proximity detection mechanism for pass-by framework," in *Computer and Information Science (ICIS), 2016 IEEE/ACIS 15th International Conference on*. IEEE, 2016, pp. 1–6.

[15] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE signal processing magazine*, vol. 25, no. 2, pp. 21–30, 2008.

[16] C. Feng, W. S. A. Au, S. Valaee, and Z. Tan, "Received-signal-strength-based indoor positioning using compressive sensing," *Mobile Computing, IEEE Transactions on*, vol. 11, no. 12, pp. 1983–1993, 2012.

[17] A. Khalajmehrabadi, N. Gatsis, D. Pack, and D. Akopian, "A joint indoor wlan localization and outlier detection scheme using lasso and elastic-net optimization techniques," *IEEE Transactions on Mobile Computing*, 2016.

[18] K. Xin, P. Cheng, and J. Chen, "Multi-target localization in wireless sensor networks: a compressive sampling-based approach," *Wireless Communications and Mobile Computing*, vol. 15, no. 5, pp. 801–811, 2015.

[19] S. Hwang, R. Ran, J. Yang, and D. K. Kim, "Multivariated bayesian compressive sensing in wireless sensor networks," *IEEE Sensors Journal*, vol. 16, no. 7, pp. 2196–2206, 2016.

[20] B. Yang, S. He, and S.-H. G. Chan, "Updating wireless signal map with bayesian compressive sensing," in *Proceedings of the 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*. ACM, 2016, pp. 310–317.

[21] M. Brandl, T. Posnicek, and K. Kellner, "Position estimation of rfid-based sensors using saw compressive receivers," *Sensors and Actuators A: Physical*, vol. 244, pp. 277–284, 2016.

[22] M. A. Davenport, D. Needell, and M. B. Wakin, "Signal space cosamp for sparse recovery with redundant dictionaries," *IEEE Transactions on Information Theory*, vol. 59, no. 10, pp. 6820–6829, 2013.