Pushing towards the Limit of Sampling Rate: Adaptive Chasing Sampling

Ying Li, Kun Xie, Xin Wang
Department of Electrical and Computer Engineering
Stony Brook University
Email: {yingli, xwang}@ece.sunysb.edu   kun.xie@stonybrook.edu

Abstract—**Measurement samples are often taken in various monitoring applications. To reduce the sensing cost, it is desirable to achieve better sensing quality while using fewer samples. Compressive Sensing (CS) technique finds its role when the signal to be sampled meets certain sparsity requirements. In this paper we investigate the possibility and basic techniques that could further reduce the number of samples involved in conventional CS theory by exploiting learning-based non-uniform adaptive sampling. Based on a typical signal sensing application, we illustrate and evaluate the performance of two of our algorithms, Individual Chasing and Centroid Chasing, for signals of different distribution features. Our proposed learning-based adaptive sampling schemes complement existing efforts in CS fields and do not depend on any specific signal reconstruction technique. Compared to conventional sparse sampling methods, the simulation results demonstrate that our algorithms allow 46% less number of samples for accurate signal reconstruction and achieve up to 57% smaller signal reconstruction error under the same noise condition.**

I. Introduction

Efficient information collection is critical for many applications, such as medical imaging, radar detection and spectrum sensing. Practical signals are generally continuous and can be sampled into digital form for more efficient storage, processing and communications. Obviously, a higher number of samples would lead to larger resource consumption and higher processing complexity. The fundamental challenge is to achieve the desired degree of signal fidelity in an often noisy environment with the minimum number of samples.

Compressive Sensing (CS) [2]–[4], [12] has attracted a lot of recent attention, with its capability of reconstructing sparse signals with the number of samples much lower than that of the Nyquist rate. The fundamental works of CS include the introduction of the l_1-minimization to reconstruct the signal, and more recently the greedy recovering of signal components gradually [28] [13] [27] [32]. Attempts have been made to directly apply CS in different application areas, including the reduction of traffic volume during signal acquisition [24] [1], and finding target locations and numbers [10] [32] [17].

Given the limitation of sensor resources and battery energy, it is often desirable to minimize the number of sensors involved in a single sensing task. This will in turn contribute to a lower average duty cycle of all the sensors and the extension of the reliable working life of the sensor network. In applications that require the participation of surrounding devices, for example the fast growing research field of crowd sensing, certain rewards can be given to the contributors to motivate their involvement

and cooperation in sensing. In these cases, minimizing the number of involved sensors helps reduce the extra sensing cost besides power consumption.

Conventional compressive sensing schemes take samples randomly and uniformly in a sampling space. This inevitably makes it difficult to determine the minimum number of samples to take. If the samples are not sufficient, the signal cannot be accurately recovered. Thus to be safe, usually more than enough samples are taken to guarantee the accurate reconstruction at higher cost. Instead of taking samples all at once, we would like to find out if the samples can be taken sequentially, with subsequent number of samples determined based on the estimation results from previous samples. More specifically, we would like to find ways of adapting the sensing matrix that defines the sensing behaviors based on the information learnt from previous observations to guide subsequent measurements to "focus" on the areas detected with possible existence of signals, so that the overall number of samples can be greatly reduced. Although it may incur some delay in collecting samples over multiple rounds, this method may help significantly reduce the sensing resources when there is a need to continuously monitor some targets or signals, while the chance of appearance of the targets or signals is very small.

Fundamentally different from some existing CS algorithms [3], [5], [14], [28] whose aims are to improve the quality of reconstructed signals with a given number of samples, our proposed algorithms aims to reduce the number of samples needed for the whole sensing process through the adaptation of sampling distribution, taking advantage of the learning process to achieve similar or even better signal recovery accuracy. Different from conventional schemes that search for signals in random locations which may miss the signals, sensors selected in our scheme first detect the potential existence of signals although their received values may be low, and the knowledge is then applied to guide efficient signal sensing later.

To illustrate the principle of our proposed learning-based adaptive CS schemes, we construct a sensing matrix according to a practical application, which allows us to map the change of the sensing matrix in theory to the choice of sensor measurement locations in practice. Specifically, we consider the detection and estimation of the strengths and locations of sparsely distributed signal sources using sensors embedded in commonly accessible wireless devices such as cell phones, which have been considered for use in crowd sensing. Detection of the signal strength on different locations finds its use in many

applications, including the emerging cognitive radio field where cell phones can be used collaboratively to form the spectrum sensing maps on the activities of primary users, localizing the sound sources for detecting events such as gunshot or riots, and detection of radiation sources. Our adaptive sampling scheme can be applied to broad categories of energy signal sensing.

We propose two learning-based adaptive sensing algorithms for different application scenarios to significantly improve the sensing efficiency and quality. Our algorithms do not depend on the underlying CS reconstruction methods, thus can be applied in many CS-suitable applications. The features of our work can be summarized as follows:

- Compared to conventional schemes which employ completely random sampling, our adaptive algorithms wisely choose where to sample based on the knowledge from previous sampling process, thus requiring much fewer samples to estimate the signals at desired accuracy level.
- Compared to a few existing adaptive sensing schemes [6], [30] in the literature:
  - The literature studies take samples uniformly although the sampling rate is adapted in the temporal or spatial domain to improve the sensing quality, while our schemes adapt the sample distribution based on the learning from previous estimation results to greatly reduce the total number of samples needed.
  - Our unique sensing matrix structure allows convenient reuse of samples previously taken along with later samples for higher sensing accuracy while avoiding tasking same sensors multiple times to conserve sensing resources and reduce the overhead in transmitting sensing signals in the network.

As we will theoretically show later, if a sensor in our schemes is tasked once, it only takes measurement in one round in the whole adaptive sensing process, thus "the number of sensors used" equals "the number of samples" in our work, and we refer to them inter-changeably in the paper. Therefore, minimizing the number of samples is equivalent to reducing the number of involved sensors, which helps to conserve the sensing cost involved for both sampling and communications. In addition, our schemes achieve much lower reconstruction error compared to conventional CS schemes under the same level of noise for the same number of sensors used.

The rest of the paper is organized as follows. In Section II, some related works are discussed. Section III gives the fundamentals of CS. Our system model and formulation are described in Section IV. Section V introduces the two algorithms of adaptive sensing. We evaluate the performance in Section VI, and conclude the work in Section VII.

II. RELATED WORK

The research problems on CS can be roughly divided into two categories. One is to study and improve the reconstruction technique itself in order to reduce the computational complexity or increase the recovery accuracy against noises. Another is the application of CS to solving practical problems.

For signal reconstruction, the l_1-minimization algorithm [3], [5], [14], also called basis pursuit, transforms the original reconstruction problem into a linear programming problem to solve with convex optimization; Another family of methods is greedy-based, including Orthogonal Matching Pursuit (OMP) [28], Stagewise OMP (StOMP) [13] and Regularized OMP (ROMP) [27], which tackles the reconstruction problem by gradually recovering the components of the signal in each iteration. The l_1-minimization is generally believed to give better reconstruction performance. While the greedy-based methods have the advantages of simpler implementation and lighter computational overhead, they require more measurements for the same reconstruction quality. Other reconstruction algorithms include iterative thresholding methods [9], [20] and various Bayesian methods [22], [31].

CS has been directly applied in various network and sensing applications. In the network area, some schemes have been proposed to reduce the total number of network messages by aggregating and forwarding the linearly weighted summation of the messages on the path towards the destination [24] or a neighbor [1], taking advantage of the capability of CS to recover the original messages. The Greedy Matching Pursuit (GMP) algorithm for target counting in [32] assumes that the nonzero values can only take several possible integers and exhaustively try out each position of the vector. This method is too computational costly for signal vectors of large dimensions, and nearly impossible for vectors with arbitrary values. [17] straightforwardly adopts CS into the Access Point received signal strength based localization problem. CS also finds its use in the field of Cognitive Radio recently. The work in [23] exploits CS to estimate the occupied spectrum channels and the locations of Primary Radios.

Iterative CS algorithms have also been proposed in the literature. In the field of image/video processing where the computing overhead is not a constraint, multiple rounds of CS reconstructions are sequenced to achieve better quality of recovered signal accuracy [29] [15] at the cost of higher number of samples and computation overhead. There are also attempts to adaptively reconstruct components of the signals for an overall better performance [19], [21], [25]. Despite the increase of tolerance to noise, these adaptive CS algorithms cannot be applied to effectively reduce the number of samples.

In the existing CS theory, the necessary number of samples to take is given and derived for basic l_1 minimization based on random sampling. In this work, we investigate the possibility and ways of reducing the number of samples needed as well as the computational complexity while maintaining the sensing accuracy. Rather than taking random samples as in conventional CS schemes, to reduce the number of samples and improve the recovery quality, we propose to adapt the sensing matrix taking advantage of the iterative learning process.

III. FUNDAMENTALS OF COMPRESSIVE SENSING

Conventional information theory mandates a sampling rate to be at least twice the bandwidth of the signal being sampled. Recent research shows that a sparse signal can be reconstructed through Compressive Sensing with a high probability at much lower sampling rate. Moreover, most signals that are not sparse enough can also be projected to other domains to achieve the desired sparsity.

Let vector $x \in \mathbb{R}^N$ be a signal not sparse enough. Given an $N \times N$ orthogonal basis $\Psi = [\Psi_1, \Psi_2, ..., \Psi_N]$ with each Ψ_i being a column vector, we have:

$$x = \Psi s = \sum_{i=1}^{N} s_i \Psi_i \qquad (1)$$

where s is the coefficients of x in the transformed domain Ψ. s is said to be k-sparse if it has at most k nonzero entries and $k \ll N$. The samples are then

$$y = \Phi x = \Phi \Psi s = As \qquad (2)$$

where Φ is an $M \times N$ measurement matrix to be defined later with $k \ll M \ll N$, the combined $M \times N$ matrix A is called the sensing matrix, and y is the sample vector of $M \times 1$.

Under the condition of l_0 minimization: $\min \|s\|_{l_0}$, instead of acquiring N samples of s, only $M = 2k$ of noise-free measurements are needed to reconstruct s [8]. However, this problem is NP-hard. If A meets the Restricted Isometry Property (RIP) condition, it is much easier to solve the l_1-minimization problem below

$$\min \|\mathbf{s}\|_{l_1} \qquad (3)$$
$$\text{s.t.} \quad \|\mathbf{Ax} - \mathbf{y}\|_{l_2} \leq \varepsilon \qquad (4)$$

where the parameter ε is the bound of the error, as long as $M \geq c \cdot k \cdot log(N/k)$ [2].

Nonetheless, the lower bound requirement of $M = 2k$ measurements is rarely achieved using conventional CS methods. The major contribution of this paper is to provide some insights on the possibility and potential strategies that can be exploited to push the sampling rate towards the limit.

IV. PROBLEM FORMULATION

To investigate the possibility and methods of improving the sampling efficiency and accuracy, in this paper, we instantiate a specific problem of using cognitive cell phones which can switch to other spectrum bands to detect the strength and locations of primary signal sources for the potential cognitive use of the unoccupied spectrum of the measured bands. This setting helps to picture the mapping of theoretical changes on the sensing matrix into the adjustment of sensor positions/sampling points in practice.

We consider a general scenario where some cell phone users are willing to participate in estimating the strength and locations of aggregate primary radio signals (e.g., TV or radar signals) which are generally located sparsely in the sensing field. Since we utilize cell phones to detect the signal strength, we consider cell phones as spectrum "**sensors**" in this paper. To provide a location reference and facilitate the scalable monitoring, the sensing domain is partitioned into N grids with the size determined based on the resolution requirement of a specific application, and each grid could have no or several signal sources inside it. Similarly, a grid could have no or several cell phones, each can update the base station its grid location upon grid crossing by piggying back the information with other uplink messages.

While some literature work [32] assume that signal sources have identical transmitting power to count the number of targets, we do not have this restriction or to differentiate individual signals within a grid. Instead, we consider signals within a grid as an **aggregate signal source** located at the grid center. Generally, within the short sensing period, we don't expect a significant change on the strength of the signals to be measured and a large number of signal sources or sensors to move across grids.

Figure 1 shows an example system of 16 grids in a monitoring region. Signal sources with different level of energy are indicated by the dot of different sizes. For grids that have several phones inside, we can always pick only one cell phone to do the task in order to preserve the sensing resources. Thus it is safe to assume each grid has at most one sensor for a specific sensing task at a given time. Cell phones tasked for sensing the spectrum will send their samples to the base station to be fused.
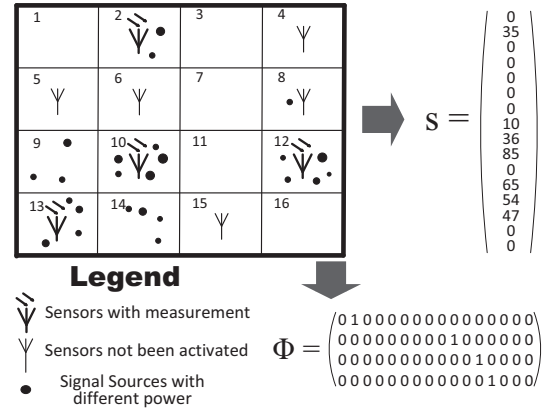

Fig. 1. A demo of the system with the corresponding signal vector s and measurement matrix Φ.

To monitor the energy of signals, traditionally, a large number of sensors are placed across the whole monitoring domain and kept active to maintain the coverage [18] [16]. In reality, for a given sensing resolution, i.e. grid size in this setup, only a small number of grids will contain signal sources at a given time. This spatial domain sparsity makes it possible to apply the compressive sensing technique to reconstruct the locations and strength of the signal sources.

Instead of tasking an adequate number of sensors [32] randomly based on the CS recovery requirement, to increase the recovery accuracy while reducing the total number of sensors involved, we would like to start with a small number of sensors at random locations. We then adaptively task additional sensors by learning from past measurement data. Next we will show mathematically how our matrix structure helps to facilitate this process.

The strength or energy of many real-world signals decays over the distance, and could also be impacted by the environment. Our proposed schemes do not rely on any specific channel models, therefore we denote the channel factor between the radio transmitter in grid i and receiver in grid j as C_{ij}. The channel fading and environment impact often add in random factors into C_{ij}, rendering the channel to be Gaussian.

We abstract the distribution of signal sources into vector presentation. Let $s = [s_1, s_2, ..., s_N]^T$ be an $N \times 1$ column vector, where the ith entry s_i is the aggregate signal strength of grid i. s is k-sparse with $k \ll N$, which means at most k grids out of N actually have signal sources.

Let Ψ be an $N \times N$ transformation matrix which embodies the signal energy decaying process over the radio channel:

$$\Psi = \begin{pmatrix} C_{11} & C_{21} & \cdots & C_{N1} \\ C_{12} & C_{22} & \cdots & C_{N2} \\ \vdots & \vdots & \ddots & \vdots \\ C_{1N} & C_{2N} & \cdots & C_{NN} \end{pmatrix} \quad (5)$$

Then $x = \Psi s$ is the received signal strength vector with $x(j)$ denoting the aggregate signal strength received by the sensor at grid j from all signal sources.

As illustrated in Figure 1, the sensors tasked to take samples can be "selected" by an $M \times N$ measurement matrix Φ. The mth row of the matrix is a $1 \times N$ row vector with all elements equal to zero except $\Phi(m, j) = 1$, where j is the index of the grid at which the sensor m is located. Each entry $x(j)$ denotes the signal strength received at the sensor in grid j, so the effect of left-multiplying Φ with the vector x in Equation(2) is to select M out of N rows of x, or equivalently choose the M sensors at specific grid positions to take samples.

Under this formulation, a set of sensors can be flexibly and conveniently specified with matrix Φ to take samples in each round of the adaptive sensing process. We can take advantage of the samples acquired in the previous iterations of the adaptive process and combine them with the new samples to form a more informative sample vector y, where

$$y = \begin{pmatrix} y' \\ y'' \end{pmatrix} = \Phi \Psi s = \begin{pmatrix} \Phi' \\ \Phi'' \end{pmatrix} \Psi s \quad (6)$$

to recover the data using the new Φ matrix. In Equation 6, y' is the vector of samples already collected by sensors previously selected by Φ', and y'' contains the new samples taken by sensors newly specified in Φ''. A selected sensor only needs to take sample once during the whole sensing process, with its sample saved and used with later samples to reconstruct the signal. Therefore, the total number of samples taken in the whole sensing period equals the number of sensors used, thus we do not differentiate the number of samples from the number of sensors thereafter. The spectrum map can be updated periodically, and for each period, a new sequence of sensing can be initiated. The reuse of samples from previous rounds is enabled by our unique sensing matrix structure, which helps to greatly reduce the sensing resource consumption and serves as the basis of our algorithms to be proposed.

The sensing matrix $A = \Phi \Psi$ has been proven in [12], [32] to obey RIP condition as long as matrix Φ and Ψ are constructed as defined above. Therefore in each adaptive iteration, we can safely apply the l_1-minimization method to perform the reconstruction. Although the small number of sensors used in the beginning phase of the adaptive process would result in inaccurate reconstruction, in the next section, we will show that by learning the result from each previous iteration, the process

eventually converges to an accurate reconstruction, while the overall number of sensors used is significantly reduced.

V. ADAPTATION OF SENSING MATRIX AND PROCESS FOR IMPROVED SENSING QUALITY

In this work, we propose two learning-based methods in which a small set of sensors considered to be better will be tasked to take samples in each round of sensing based on the reconstruction results from the previous round, then the new samples together with the existing samples will generate an improved reconstruction, and so forth. In this section, we first present our basic setting in Section V-A, and then introduce in details of our sensing algorithms in Section V-B and V-C, respectively. Finally, we discuss our strategy in avoiding the local optimal solution in Section V-D.

A. Basic Setting and Design Consideration

In this paper, we use terms such as "task", "turn on" or "add" when selecting a sensor to take measurement. The locations of sensors in terms of which grid are denoted by the set $\mathbb{L}$. For example, in Figure 1 sensors are installed at grids 2, 4, 5, 6, etc., then $\mathbb{L}$ will have the corresponding elements.

As an iterative scheme, the algorithm needs an initial value to start with. Specifically, at the beginning of our adaptive sensing process, we need to decide the number of sensors to take samples for the first round, then the adaptive algorithm will handle the following iterations automatically. In order to determine the optimal number of samples to take, conventional CS techniques often assume the prior knowledge of the sparsity value k or estimate k, and their performances are highly impacted by the accuracy of k value taken. In contrast, our schemes take samples sequentially and determine the number of additional samples needed based on the previous estimation results, and the initial k is only used as a reference. As mentioned in Section III, $2k$ (k is the sparsity) is believed to be the theoretical extreme of the number of necessary samples [8] for under-sampled signal recovery. Our algorithms thus start the first iteration by activating $M^{(0)} = 2k$ sensors at locations randomly chosen from the set $\mathbb{L}$ to take samples. Our simulation study on the impact of the starting number of sensors at the first iteration also reveals that $2k$ is the optimal choice in terms of the final total sensing resources needed.

In our proposed adaptive schemes, instead of distributing the sensing resources evenly to randomly sample the signal vector and repeating the same sensing process over time, in each following iteration, we propose to just take a few new samples around the vector entries detected to have a higher chance of being nonzero from the reconstruction result of the previous iteration. Then the new samples combined with existing samples, as in Equation (6), are applied to reconstruct the signal vector.

We use $\hat{s}^{(i)}$ to denote the reconstructed signal after the ith round of sensing. As we start from sensing with the number of sensors smaller than necessary for CS recovery, there exists inaccuracy in getting each intermediate $\hat{s}^{(i)}$. Although neither the positions nor the values of the nonzero entries of $\hat{s}^{(i)}$ may be accurate, our preliminary studies indicate that the actual nonzero entries of the original vector are close to the

nonzero positions indicated by $\hat{s}^{(i)}$. In our signal detection example, the signal sources are possibly located in the region close to the grids corresponding to the nonzero positions of $\hat{s}^{(i)}$. So by "moving" sensing focus (i.e., selecting sensors at the desired locations) towards the estimated locations of the signal sources step by step, the algorithms will improve the reconstruction results until the positions and the values of the nonzero entries no longer change. This way we can find the accurate positions and energy levels of the signal sources. This also helps to increase the sensing efficiency and reduce the number of sensors needed. This is the fundamental principle of our adaptive algorithms.

The learning process iterates until the recovered result has reached a desired quality or cannot be further improved. Our results in Section VI will show that the total number of sensors used in the overall adaptive process is much less than that needed for a single-time conventional CS recovery for achieving the same level of accuracy.

One thing to note is that $\hat{s}^{(i)}$ may contain many nonzero entries with very small values, which are insignificant and tedious to consider as the possible positions of targets. It has to be modified before each next iteration. To be specific, first all the negative entries of $\hat{s}^{(i-1)}$ will be set to zero. Then for the positive entries, all those with values below $\alpha\%$ that of the largest positive entry value in $\hat{s}^{(i-1)}$ will be set to zeros. The impact of α on performance is inspected in the simulations.

Based on the same adaptive principle, we propose two different algorithms on choosing new sampling locations in the next round for the signal strength detection problem, depending on signal source location distributions.

B. Individual Chasing

If there is no knowledge on the locations or distribution features of signal sources, it is reasonable to assume they are randomly distributed in the monitoring area. In this case, we will adapt the positions for sampling towards the estimated location of each individual signal source given by the last iteration.

In the ith iteration, according to the previous reconstruction vector $\hat{s}^{(i-1)}$, for each of its nonzero entry $\hat{s}^{(i-1)}(n)$, a sensor in set $\mathbb{L}$ whose location is closest to grid n is selected to take a sample if a measurement is not already taken there before. It is best to choose the sensor right inside grid n for sampling. However if there is no sensor located in the grid n, another sensor in the sensor location set $\mathbb{L}$ with the smallest euclidian distance to grid n will be selected. After each non-zero position n is ensured to have one sampling in the corresponding grid, the l_1-minimization process is invoked to get the reconstruction $\hat{s}^{(i)}$ based on the combined samples y and combined Φ in Equation (6). The reconstruction result $\hat{s}^{(i)}$ is fed into Algorithm 2 for termination condition check to determine whether the algorithm should end or continue with more iterations. Algorithm 1 outlines the details of Individual Chasing in each iteration.

Figure 2-(a) shows the sensor locations when the Individual Chasing algorithm terminates. It can be observed that sampling measurements have been taken at each grid with signal source that also has a sensor inside. For the grids with signal sources
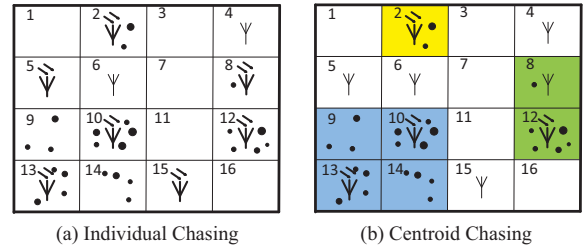

(a) Individual Chasing (b) Centroid Chasing

Fig. 2. Illustration of Individual Chasing and Centroid Chasing.

but without sensors deployed, samples are taken at the closest grids, i.e. samples have been taken by sensors in grid 5 and 15 for nearby signals in grid 9 and 14.

The Individual Chasing scheme adapts well when signal sources are uniformly distributed in the monitoring field. More importantly, the simulations in Section VI demonstrate that the Individual Chasing algorithm converges fast with superior recovery accuracy regardless of how the signal source locations are distributed.

Algorithm 1 Individual Chasing

1: In the ith iteration:
2: **for** each nonzero position n of $\hat{s}^{(i-1)}$ **do**
3:    find a grid position p in $\mathbb{L}$ with the smallest euclidian distance to grid n.
4:    **if** no measurement has previously been taken at grid p **then**
5:       take sampling at grid p's sensor.
6:    **end if**
7: **end for**
8: combine new samples with existing ones for y.
9: do l_1-minimization on y and Φ to get $\hat{s}^{(i)}$.
10: call Algorithm 2 to check the termination condition.
11: **if** algorithm does not terminate in this iteration **then**
12:    $i = i + 1$, go back to Line 1 and start the next iteration.
13: **end if**

Algorithm 2 Termination Condition Check

1: **if** the nonzero positions of $\hat{s}^{(i)}$ are all the same as $\hat{s}^{(i-1)}$ **then**
2:    **if** the numeric difference of each nonzero value between $\hat{s}^{(i)}$ and $\hat{s}^{(i-1)}$ is smaller than a percentage threshold Δ^* of $\hat{s}^{(i-1)}$ **then**
3:       reconstruction process terminates in this iteration.
4:    **else**
5:       continue with the next iteration of reconstruction.
6:    **end if**
7: **else**
8:    continue with the next iteration of reconstruction.
9: **end if**
10: $*$ *the choice of Δ is given in SectionVI*

C. Centroid Chasing

In some scenarios, the signal sources may locate closely and form clusters. The clustering patterns of signal sources may be exploited to guide the sampling locations to facilitate the finding of signal sources with even fewer sensor measurements. For example, in an urban area, a region with high noise often indicates the existence of some events, such as street concert, parade, or riot. In this case, many sound sources may stay closely, and there is a need to detect the high noise areas to identify the events and protect against potential harm to the community. The microphones of cell phones can be exploited to collaboratively detect the strength and location of these sound sources.

The Centroid Chasing scheme initializes similarly as Individual Chasing by randomly activating $M^{(0)} = 2k$ sensors from the set $\mathbb{L}$ to start the first iteration. Algorithm 3 details the actions for each consecutive iteration. Let $\mathbb{T}^{(i-1)}$ denotes the set of grid numbers that correspond to the non-zero positions of the reconstructed $\hat{s}^{(i-1)}$, which indicate the possible grid positions where signal sources may reside estimated in the ith round. In the ith iteration, grid numbers in $\mathbb{T}^{(i-1)}$ are grouped into C clusters based on their mutual euclidian distances, with $\mathbb{T}_t^{(i-1)}$ denoting the set of grids belonging to the tth ($t = 1, ..., C$) cluster. The optimal clustering criteria is given later. The minimum rectangular region that covers all the grids of the tth cluster is called the cluster region of the tth cluster, and its size $R_t^{(i-1)}$ is the number of grids it covers. For each cluster, based on its signal target density defined as $|\mathbb{T}_t^{(i-1)}|/R_t^{(i-1)}$, $M_t^{(i-1)}$ closest sensors (these sensors could be either within or outside of the rectangular cluster region) will be selected for samplings at this iteration, where

$$M_t^{(i-1)} = |\mathbb{T}_t^{(i-1)}| \cdot \left(1 - \frac{|\mathbb{T}_t^{(i-1)}|}{R_t^{(i-1)}}\right). \qquad (7)$$

Equation (7) reflects the tendency to put less sensing resource in a cluster region when the target sources are denser. The number of sampling positions to be considered for adding at each iteration is controlled to be less than that of the Individual Chasing method, which is $|\mathbb{T}_t^{(i-1)}|$, by multiplying it with the density which is a value smaller than 1. This feature, however, may result in more iterations needed thus more sensors involved than the Individual Chasing algorithm when the signals are sparsely distributed and poorly clustered. The performances of two algorithms are compared under different scenarios in the simulation. Again, the new measurements obtained at the current iteration will be combined with existing samples for CS reconstruction, which will be checked by termination condition afterwards.

Specifically, Line 2 of Algorithm 3 clusters the possible signal sources whose positions are estimated by the nonzero positions of $\hat{s}^{(i)}$. This procedure starts at an initial empirical number of clusters $C = \sqrt{|\mathbb{T}^{(i-1)}|/2}$, where $|\mathbb{T}^{(i-1)}|$ is the total number of elements in the set to be clustered, as given in [26]. The optimal number of clusters to form is found by varying the number of clusters C and picking the one under which the average distance between each grid position to the

---

Algorithm 3 Centroid Chasing

1: In the ith iteration:
2: group the grids in $\mathbb{T}^{(i-1)}$ into C clusters.
3: **for** each of the cluster $\mathbb{T}_t^{(i-1)}$ **do**
4:      find the $M_t^{(i-1)}$ sensors in $\mathbb{L}$ that are closest in euclidian distance to the center of the tth cluster region.
5:      **if** any of these $M_t^{(i-1)}$ chosen sensor grids has not sampled before **then**
6:          activate the sensors at these grids.
7:      **end if**
8: **end for**
9: combine new samples with existing ones for y.
10: do l_1-minimization on y and Φ to get $\hat{s}^{(i)}$.
11: call Algorithm 2 to check the termination condition.
12: **if** algorithm does not terminate in this iteration **then**
13:     $i = i + 1$, go back to Line 1 and start the next iteration.
14: **end if**

---

centroid of cluster it belongs to is minimal. After clustering, a smaller number of sensors can be tasked in the signal concentrated area to achieve even higher sensing efficiency.

Figure 2-(b) shows the sensing condition at the end of the Centroid Chasing algorithm. Grids with signals are grouped into 3 clusters. Within each cluster, a portion of the closest sensors are activated for sampling. Compared with Individual Chasing scheme on the left of the figure, fewer number of sensors are used given the signal sources have a nice clustering feature.

The Centroid Chasing algorithm is specially suitable for applications with knowledge of clustered distribution, which, nonetheless, does not prevent its applicability in general cases. The simulations in section VI-D show that Centroid Chasing algorithm outperforms the Individual Chasing algorithm when the signal sources are denser in clusters, and works almost equally efficient in general scenarios where the signal sources are sparsely distributed.

D. Local Optimum Avoidance - Random Exploration

Very unlikely but possible, adaptive algorithms could converge at the local optimum. In our problem, local optimum does not give accurate reconstruction at the algorithm termination time. In an iterative scenario, one can only look at the intermediate results to decide whether the iteration should stop. The Individual Chasing and Centroid Chasing algorithms will stop when consecutive iterations present no more change in the results. In order to avoid possible local optimum, we introduce an extra step called Random Exploration. To be specific, for both algorithms, when the Termination Condition Alg.2 is satisfied at certain iteration, we do not terminate the program immediately. Instead, we randomly pick up some sensors that have not been activated before to take samples, then re-enter the adaptive process and let it converge again. This random walk procedure is proved to work extremely well in our simulations.

VI. Simulation and Performance Evaluation

We evaluate the performance of our two proposed learning-based compressive sensing schemes through extensive simula-

tions. Before showing our simulation results, we first introduce our performance metrics and simulation set-up.

A. Performance Metrics

- **Reconstruction Error**: defined as the Sum of Absolute Difference (SAD) between the recovered and the original signal vector:

$$SAD = \|\hat{s} - s\|_{l_1} = \sum_{i=1}^{N} |\hat{s}_i - s_i| \qquad (8)$$

This measurement metric evaluates the accuracy of vector reconstruction. It reflects not only the degree of error due to the position mismatch of nonzero entries, but also the difference in magnitude for each unmatched entry.

- **Number of Sensors Needed** (M): in our adaptive algorithms, new sensors may be added for sampling in each new iteration. The number of sensors needed for our algorithms are defined as the total number of sensors used during the whole sensing process, which is the least number of sensors needed in our performance studies to give 100% reconstruction accuracy.

B. Simulation Set-Up

We simulate the general problem on forming the energy measurement map and compare the performances of our proposed schemes with peer works. dBm is adopted as the measurement unit of signal strength in our simulation. One grid could have multiple signal sources, and the overall signal strength for a grid is the aggregate of these signals. Being aware that the numeric scale of the nonzero entries of signal vector is not critical to the problem of compressive sensing recovery but actually the sparsity is, we assume a range of 30-500 dBm for the possible aggregated signal strength inside any single grid. For every simulation run, signal sources are generated at random locations across the sensing region with the aggregate signal strength of a grid to be a random value selected from the 30-500 range.

The simulation is carried out with MATLAB. Our schemes improve the sensing performance benefitting from the adaptive chasing process, but do not rely on any adopted specific channel model as previously pointed out. In the simulation, we would like to use a specific radio channel model to test our schemes. We define the channel as the following. The strength at location j for a signal source at location i is roughly approximated as:

$$\mathbb{P}_{ij} = \frac{P_i \mathbb{G}_{ij}}{d_{ij}^{\beta}}, \; where \; \mathbb{G}_{ij} = \mathbb{X}_{ij} + \mathbb{Y}_{ij} \cdot \mathrm{i} \qquad (9)$$

P_i is the signal strength at its source location i, which is essentially the aggregated strength of signal sources at grid i, i.e. s_i The denominator represents the path loss due to the distance d_{ij} between locations i and j. β is the decaying factor with possible value in $[2.0, 5.0]$, depending on the environment. 'i' is the imaginary sign. $\mathbb{G}_{ij}$ is a complex random Gaussian variable with real and imaginary components both being independent and identically distributed zero-mean Gaussian variables $\mathbb{X}_{ij}, \mathbb{Y}_{ij} \sim \mathcal{N}(0, \sigma_0^2)$, which captures the Raleigh distribution for multi-path fast fading of the signal [7], [11].

The variance σ_0 is set to 0.5 as in [32] for fair comparison that follows. $\frac{\mathbb{G}_{ij}}{d_{ij}^{\beta}}$ corresponds to C_{ij} in the channel model of Equation (5).

In the short duration of performing the sensing task, sensors (essentially cell phones in our application) will not have significant moving distance or move across grids. For each sensing task, 400 sensors are randomly deployed in an area of $N = 30 \times 30 = 900$ grids with at most one sensor inside one grid, as we can always select one sensor to participate in the sensing task when a grid has multiple sensors inside as discussed previously. The size of each square grid is set to 30 meter which can reasonably reflect the distance effect in signal propagation. Although the grid size determines the signal monitoring resolution, it does not have significant impact on the performance of our algorithms once the resolution requirement is given. There are $k(k << N)$ grids with signal sources at a given time instant. k is the sparsity value, which is varied in different simulation studies. The termination condition threshold Δ in Alg.2 is set to 5% which will generally guarantee the recovery result to be accurate at the algorithm termination time with the recovery error in the order of 10^{-4} even for real valued signal vectors based on our preliminary studies. To compare our two schemes, the positions of k grids with signal sources can be either distributed in a clustered fashion, or uniformly distributed. To examine the reliability performance of our schemes, Gaussian White noise $N(0, \sigma^2)$ is added to the observed sample vector y in some of the simulation runs, and SNR measure is exploited to quantify the noise strength. Each presented result is the average of many runs.

Our proposed two algorithms Individual Chasing and Centroid Chasing, also referred to as IC and CC from now on, do not depend on any specific CS reconstruction technique. Thus we chose two fundamental and most prevalent types of work for performance evaluations-l_1 minimization based CS and greedy based CS. GMP [32] provides a greedy based reconstruction algorithm for CS, and also exploits the received signal strength at different grid positions to help solve the target localization and counting problem. l_1-magic is a concise and dominant realization of l_1 minimization based CS scheme, which can be directly applied to and is thus worth comparing with our simulation scenario of signal strength vector reconstruction.

C. Parameter Study

For our adaptive algorithms, the total number of sensors used increases after each iteration. Naturally, the number of sensors taken in the first iteration would have impact on the final total number of sensors used. In Figure3-(a), we study the optimal starting number of sensors for our Individual Chasing algorithm. For each sparsity k, the optimal choice on the number of sensors to start with is clearly around $2k$. Intuitively, if we start with too few sensors, the intermediate recovered vectors will be very inaccurate and result in much more iterations thus more sensors involved in total; however starting with too many sensors at the beginning would provide more sensing resources than needed which again causes more sensors in total. Therefore the number of sensors to start with for both Individual Chasing and Centroid Chasing algorithms

will be set to twice the value of k in the following simulations if not otherwise specified.
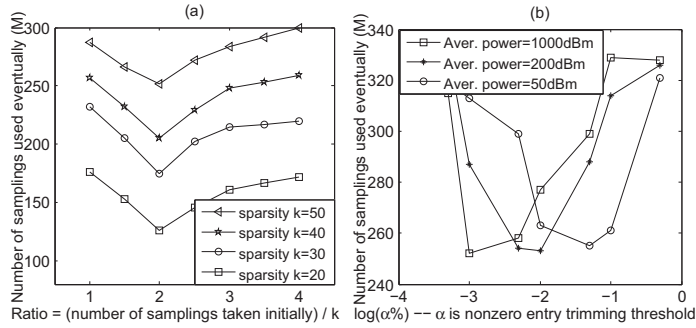

Fig. 3. (a) The number of sensors ultimately used vs. the number of sensors started with. (b) The number of sensors ultimately used vs. the trimming threshold α

In Section V, we mentioned the trimming threshold α, based on which a portion of insignificant nonzero entries of the recovered vector are set to '0' before the next iteration in order to shrink the target range of nonzero entries to work on. In Figure 3-(b), we fix k at 50, but vary the average signal source transmitting power (i.e. magnitude of each vector entry) at three distinct levels-1000, 200 and 50 dBm, and investigate the number of sensors needed for accurate recovery using each different trimming threshold. Since the range of α picked spans over several orders of magnitude, we apply base-10 logarithm on $\alpha\%$ as the x-axis. It can be clearly observed that for each setting of signal vector magnitude, there is an optimal range of α that gives the minimum number of sensors needed. Out of this range, the performance deteriorates and quickly reach a steady level. This is because only when the $\alpha\%$ is appropriately chosen that not too many components of the vector are wiped out nor too many insignificant ones are preserved will the reconstruction process requires the least number of sensors.

With the aggregate signal strength ranging between 30-500 dBm, the average signal strength in our simulation is close to 200 dBm. According to the figure, 1% is adopted as the default trimming threshold in the simulations that follow.

D. *Number of Sensors Needed*

Pursuing a minimum number of sensors needed for an accurate signal reconstruction is the major challenge and research focus in the closely related research fields. We evaluate the minimum number of sensors needed for accurate signal vector reconstruction (zero reconstruction error) under different levels of signal sparsity for each scheme in Figure 4. As expected, the number of sensors needed increases as k grows for all the algorithms. Particularly, Figure 4-(a) is under the scenario where the signal sources are randomly uniformly distributed across the network grids. IC performs slightly better than CC as expected. The clustering function of CC is not effective when the signal sources are uniformly distributed, which leads to more iterations to converge and more sensors needed. Compared to GMP, IC requires 45% fewer sensors when k is small, and about 23% fewer sensors when k gets bigger. IC

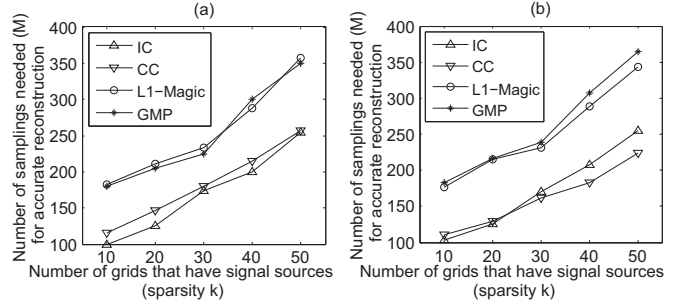requires $46\% - 25\%$ fewer sensors than l_1-magic for different k.

Fig. 4. (a) scattered signal sources. (b) clustered signal sources.

In Figure 4-(b), the signal sources are distributed in clustered fashion across grids, which benefits the clustering process of CC algorithm. Thus CC is observed to require fewer sensors than IC as the number of signal sources exceeds certain value. While the performance difference between IC and CC is small, CC requires $40\% - 33\%$ fewer sensors than GMP, and $40\% - 30\%$ fewer than l_1-magic, which are big improvements. In general, both IC and CC work extremely well no matter the signal sources are concentrated or scattered, and far outperform the other two schemes under the same k.

Given the performance difference between IC and CC is small and both algorithms follow the same principle, we only study and compare IC with the other two schemes and assume the signal sources are randomly and uniformly distributed in the simulations that follow.

E. *Reconstruction Error*

1) Convergency Study: We study the convergency of IC in Figure 5-(a). It is clear that under all k, IC is able to converge within 3-6 iterations to get accurate reconstruction with 0 error, and it exhibits a rather steady (approximate-linear) improvement in reducing the reconstruction error in each iteration. It converges faster for larger k. This is due to the fact that we initialize $2k$ number of sensors for the optimal performance. With a larger k, there are more samples taken at the beginning, therefore it needs fewer iterations to get enough overall sensor samples for accurate recovery.
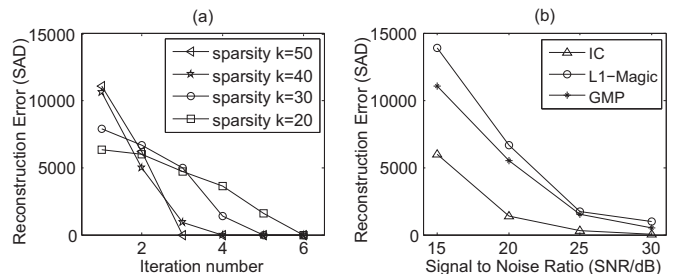

Fig. 5. (a) Convergency study of Individual Chasing algorithm. (b) The reconstruction error comparison due to noise under the same $k = 50$ and $M = 250$.

2) Performance Under Noise: The adaptive algorithms can always find the accurate reconstruction given enough iterations and proper handling with local optimum avoidance. However in a noisy environment where the sample y is contaminated, the final reconstructed result could be different from the actual signal vector.

The reconstruction errors due to sampling noise for different algorithms are compared in Figure 5-(b). The reconstruction error reduces as signal-to-noise ratio increases for all three schemes. Under the same sensing condition of $k = 50$ and $M = 250$, at each SNR level, IC gives much more accurate result than the other two. In the worst scenario with the strongest noise at 15dB SNR in our test setting, the reconstruction error for IC is approximately $(14000 - 6000)/14000 = 57\%$ smaller than that of l_1-magic. GMP is slightly more accurate than l_1-magic under the same sensing setting, because it enumerates all possible values for each possible nonzero position of the vector at the cost of higher computational overhead.

VII. Conclusion

Rather than using conventional random sampling in compressive sensing, we observe and theoretically prove that by learning from the sensing results and adjusting the structure of the sensing matrix adaptively, the number of samples needed for high-quality signal recovery in compressive sensing can be significantly lower than that required by the basic l_1 minimization solution. We propose two learning-based adaptive algorithms, Individual Chasing and Centroid Chasing, for different signal source distribution scenarios. Both schemes adaptively concentrate sensing resources to proper signal subspace towards better acquisition of signals, and do not depend on any specific CS reconstruction methods. The instantiation of our algorithms to solve the general signal strength sensing problem with distance fading can be conveniently generalized to various similar applications. Extensive simulations demonstrate that our algorithms can achieve as much as 57% more accurate signal recovery under noisy conditions, and require up to 46% fewer sensors than state-of-the-art related works.

References

[1] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah. Randomized gossip algorithms. *IEEE/ACM Trans. Netw.*, 14(SI):2508–2530, June 2006.

[2] E. Candes, J. Romberg, and T. Tao. Stable signal recovery from incomplete and inaccurate measurements. *Comm. Pure Appl. Math.*, 59(8):1207–1223, 2006.

[3] E. Candes and T. Tao. Decoding by linear programming. *Information Theory, IEEE Transactions on*, 51(12):4203–4215, 2005.

[4] E. J. Candes and J. Romberg. Practical signal recovery from random projections. In *SPIE Computational Imaging*, volume 5674, pages 76–86, 2005.

[5] S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM Rev.*, 43(1):129–159, 2001.

[6] C. T. Chou, R. Rana, and W. Hu. Energy efficient information collection in wireless sensor networks using adaptive compressive sensing. In *IEEE 34th Conference on Local Computer Networks*, pages 443–450, Oct 2009.

[7] T. Clouqueur, K. Saluja, and P. Ramanathan. Fault tolerance in collaborative sensor networks for target detection. *Computers, IEEE Transactions on*, 53(3):320–333, Mar 2004.

[8] W. Dai and O. Milenkovic. Subspace pursuit for compressive sensing signal reconstruction. *IEEE Trans. Inf. Theor.*, 55(5):2230–2249, May 2009.

[9] I. Daubechies, M. Defrise, and C. De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Comm. Pure Appl. Math.*, 57(11):1413–1457, 2004.

[10] M. Ding, F. Liu, A. Thaeler, D. Chen, and X. Cheng. Fault-tolerant target localization in sensor networks. *EURASIP J. Wirel. Commun. Netw.*, 2007(1):19–19, Jan. 2007.

[11] M. Ding, F. Liu, A. Thaeler, D. Chen, and X. Cheng. Fault-tolerant target localization in sensor networks. *EURASIP J. Wirel. Commun. Netw.*, 2007(1):19–19, Jan. 2007.

[12] D. Donoho. Compressed sensing. *Information Theory, IEEE Transactions on*, 52(4):1289–1306, 2006.

[13] D. Donoho, Y. Tsaig, I. Drori, and J.-L. Starck. Sparse solution of under-determined systems of linear equations by stagewise orthogonal matching pursuit. *Information Theory, IEEE Transactions on*, 58(2):1094–1121, 2012.

[14] D. L. Donoho and Y. Tsaig. Fast solution of l1-norm minimization problems when the solution may be sparse. *Information Theory, IEEE Transactions on*, 54(11):4789–4812, 2008.

[15] D. duan Liu, D. Liang, X. Liu, and Y. ting Zhang. Under-sampling trajectory design for compressed sensing mri. In *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*, pages 73–76, Aug 2012.

[16] Q. Fang, F. Zhao, and L. Guibas. Lightweight sensing and communication protocols for target enumeration and aggregation. In *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*, MobiHoc '03, pages 165–176.

[17] C. Feng, W. Au, S. Valaee, and Z. Tan. Compressive sensing based positioning using rss of wlan access points. In *INFOCOM, 2010 Proceedings IEEE*, pages 1–9, 2010.

[18] Y. Guo, B. Hua, and L. Yue. Energy-based target numeration in wireless sensor networks. In *Future Generation Communication and Networking (FGCN 2007)*, volume 2, pages 380–385, 2007.

[19] J. Haupt, R. Nowak, and R. Castro. Adaptive sensing for sparse signal recovery. In *Digital Signal Processing Workshop and 5th IEEE Signal Processing Education Workshop, 2009. DSP/SPE 2009. IEEE 13th*, pages 702–707, Jan 2009.

[20] K. Herrity, A. Gilbert, and J. Tropp. Sparse approximation via iterative thresholding. In *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, volume 3, 2006.

[21] M. Iwen and A. Tewfik. Adaptive compressed sensing for sparse signals in noise. In *Signals, Systems and Computers (ASILOMAR), 2011 Conference Record of the Forty Fifth Asilomar Conference on*, pages 1240–1244, Nov 2011.

[22] S. Ji, Y. Xue, and L. Carin. Bayesian compressive sensing. *Signal Processing, IEEE Transactions on*, 56(6):2346–2356, 2008.

[23] L. Liu, Z. Han, Z. Wu, and L. Qian. Collaborative compressive sensing based dynamic spectrum sensing and mobile primary user localization in cognitive radio networks. In *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE*, pages 1–5, Dec 2011.

[24] C. Luo, F. Wu, J. Sun, and 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*, MobiCom '09, pages 145–156.

[25] M. Malloy and R. D. Nowak. Near-optimal adaptive compressed sensing. *CoRR*, abs/1306.6239, 2013.

[26] K. Mardia. *Multivariate Analysis*. Academic Press, 1979.

[27] D. Needell and R. Vershynin. Signal recovery from incomplete and inaccurate measurements via regularized orthogonal matching pursuit. *Selected Topics in Signal Processing, IEEE Journal of*, 4(2):310–316, 2010.

[28] Y. Pati, R. Rezaiifar, and P. S. Krishnaprasad. Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition. In *Signals, Systems and Computers, 1993. 1993 Conference Record of The Twenty-Seventh Asilomar Conference on*, volume 1, pages 40–44, 1993.

[29] S. Ravishankar and Y. Bresler. Adaptive sampling design for compressed sensing mri. In *Engineering in Medicine and Biology Society,EMBC, 2011 Annual International Conference of the IEEE*, pages 3751–3755, Aug 2011.

[30] J. Wang, S. Tang, B. Yin, and X.-Y. Li. Data gathering in wireless sensor networks through intelligent compressive sensing. In *Proceedings IEEE INFOCOM*, pages 603–611, March 2012.

[31] D. Wipf and B. Rao. Sparse bayesian learning for basis selection. *Signal Processing, IEEE Transactions on*, 52(8):2153–2164, 2004.

[32] B. Zhang, X. Cheng, N. Zhang, Y. Cui, Y. Li, and Q. Liang. Sparse target counting and localization in sensor networks based on compressive sensing. In *INFOCOM, 2011 Proceedings IEEE*, pages 2255–2263, 2011.