Online rate adjustment for adaptive random access compressed sensing of time-varying fields

Naveen Kumar[1*], Fatemeh Fazel[2]
, Milica Stojanovic[2]
 and Shrikanth S. Naryanan[1]

[*]Correspondence:
komathnk@usc.edu
[1]Signal Analysis and Interpretation
Laboratory (SAIL),
Electrical Engineering,
University of Southern California,
Los Angeles, CA,
Full list of author information is
available at the end of the article

Abstract

We develop an adaptive sensing framework for tracking time-varying fields using a wireless sensor network. The sensing rate is iteratively adjusted in an online fashion using a scheme that relies on an integrated sensing and communication architecture. As a result, this scheme allows for an implementation that is both energy-efficient and robust. The objective is to promote an "active" framework which uses the information extracted from the network data and iteratively adjusts the monitoring process to capture the temporal variations in the monitored field. We propose two metrics based on target detection/tracking for this feedback scheme that seek to trade off between energy-efficiency and accuracy of the detection/ tracking tasks. Our simulation results suggest that tying target detection with the rate adjustment algorithm, ensures that the robustness to changes in the field can be achieved simultaneously with the end goal of accurate target detection. Compared to a baseline method that uses the correlation of the acquired field over time, our method exhibits better performance when the targets of interest have a smaller spatial spread.

Keywords: sensor networks; adaptive sensing; detection; random access; compressed sensing; joint communication and detection

1 Introduction

The emergence of compressed sensing framework presents significant potential for efficient sensing and sampling systems [1, 2, 3] by helping to reduce sample complexity under realistic communication constraints. Sensor network technology greatly benefits from the compressed sensing paradigm [4, 5, 6, 7, 8, 9, 10, 11, 12]. Consider for example large-scale networks that are deployed for long-term monitoring of dynamic fields such as the ocean bed that typically need to account for power consumption considerations. An efficient scheme in such cases requires performance optimization that jointly considers both the sensing and communication constraints.

To address this issue, Fazel et al. proposed a Random Access Compressed Sensing (RACS) scheme in [13, 14] for energy-efficient reconstruction of sensing fields. The proposed sensing scheme depends on integrating information from the communication and channel access modules into the data acquisition process. The method is simple to implement under realistic communication constraints and requires minimal assumptions on the field. However, the proposed RACS scheme is designed for

stationary[1] sensing fields, where the field being monitored is assumed to remain static during sensing. In order to monitor time-varying fields, in [15] the authors employed low-rank matrix recovery to reconstruct the space-time map of the field. However this approach is offline and entails considerable delays since full recovery can be attained only after the data have been collected over multiple time segments. Moreover, it is assumed that the coherence properties of the underlying field are known a-priori and remain unchanged throughout the full sensing duration. This assumption might be justified for the monitoring of natural phenomena, where the field is assumed to be either stationary or changing at a fixed rate. It is also common to find similar assumptions of stationarity in related works in object detection or classification in underwater fields [16, 17]. However, when the field being monitored undergoes a varying rate of change, (e.g., when the process is impacted by a target that is moving at an unknown or variable speed), such assumptions may not hold.

In a more recent work, Kerse et.al. [18] addressed this problem by proposing to unify target detection with reconstruction using a standard sparse identification technique. Targets are tracked from frame to frame and the authors further suggest that the tracking error could be used as a measure to adjust the sensing rate in turn. While their method does not require any aprioi knowledge of the number of targets or coherence properties of the underlying field, it relies on knowledge of the exact target signatures for both target localization and tracking. In this paper, we propose a rate adjustment method that does not require explicit knowledge of the target signature. Rather knowledge of the family of models to which the field might belong is adequate as the field model parameters can be jointly estimated.

Similar to the work in [18], we consider that the end-goal in sensing is to detect or track targets, and incorporate these data processing aspects into the joint sensing and communication scheme. We propose a framework to adjust the sensing rate by estimating different attributes of the field to make an informed decision. Our adaptive rate adjustment procedure for compressed sensing iteratively adjusts the per-node sensing rate to capture the variations in the underlying field. First, we treat the field as piecewise stationary and apply random access compressed sensing within each sensing period. Second, we compute two heuristic metrics that seek to tie in the end goal of target detection/tracking with the rate adjustment scheme. Using the data collected in each segment, the fusion center (FC) relies on a detection algorithm to first determine the current state it is in. Finally, depending on the current state a control algorithm instructs the FC to change the sensing rate if required.

A high rate of sensing would typically bode well for target detection/tracking, but it is not energy efficient. On the other hand a low rate of sensing may not necessarily lead to poor performance in detection. Thus there is a possible trade-off between energy efficiency and the target detection accuracy which the rate adjustment seeks to exploit. We perform simulation experiments using the proposed method and present results that suggest that the proposed sensing rate adjustment method exhibits better performance compared to the baseline method when compared on the following evaluation criteria: a) mean-squared error of tracking the underlying coherence time and b) F-score of the target detection accuracy.

---

[1]We use the term *stationary* in this paper to refer to fields that are temporally static during the sensing duration.

The paper is organized as follows. In Sec. 2 we explain the basic sensing model for the stationary case and in Sec. 3 we relax the stationarity assumptions. In Sec. 4 we describe the sample field model for simulation. In Sec. 5 and 6 we describe an adaptive strategy for sensing the time-varying field. Finally, in Sec. 7 and 8 we provide results for the simulation.

2 Random Sensing Network Over Stationary Fields

Consider a grid network consisting of $N = P \times Q$ sensors, with P and Q sensors in the x and y directions, respectively. The underlying assumption is that most signals of interest (natural or man-made) vary smoothly spatially and hence are compressible in the spatial Discrete Fourier Transform (DFT) basis. We denote the sparsity of the signal by S. The data from the distributed sensors is conveyed to the FC, where a full map of the sensing field is reconstructed. This map can be used for target detection as will be shown in Sec. 5.
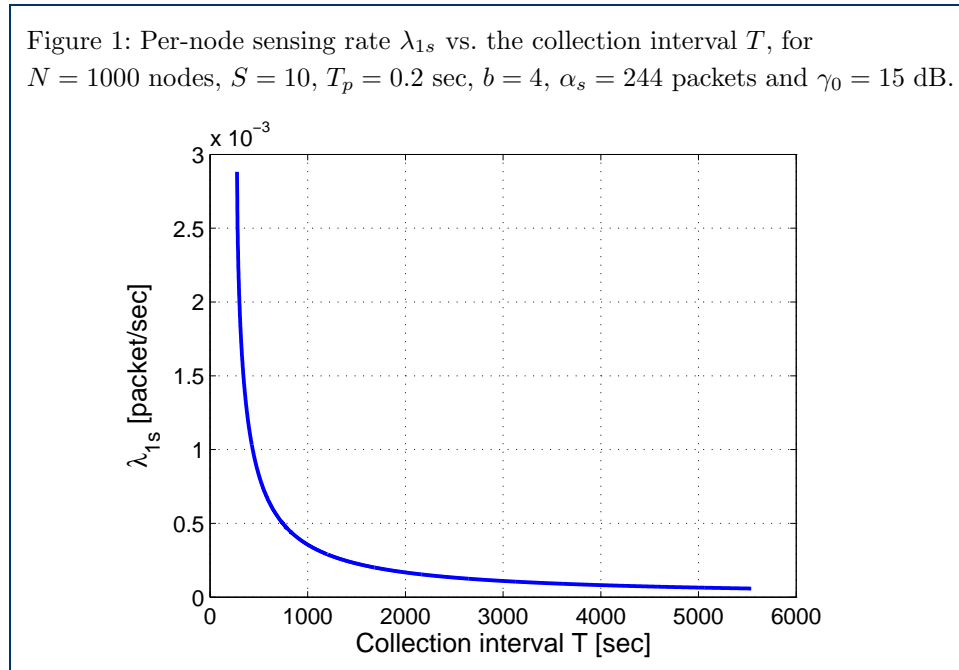
Inspired by the theory of compressed sensing, the architecture proposed in [13, 14] employs *random sensing*, i.e., transmission of sensor data from only a random subset of all the nodes. For a stationary field, each sensor node measures the signal of interest at random time instants —independently of the other nodes— at a rate of λ_1 measurements per second. It then encodes each measurement along with the node's location tag into a packet, which is digitally modulated and transmitted to the FC in a random access fashion. Owing to the random nature of channel access, packets from different nodes may collide, creating interference at the FC, or they may be distorted as a result of the communication noise. A packet is declared erroneous if it does not pass the cyclic redundancy check or a similar verification procedure. Since the recovery is achieved using a randomly selected subset of all the nodes' measurements, we let the FC discard the erroneous packets as long as there are sufficiently many packets remaining to allow for the reconstruction of the field.

The FC thus collects the useful packets over a collection interval of duration T. The interval T is assumed to be much shorter than the coherence time of the process, such that the process can be approximated as fixed during one such interval. Let $R_{xx}(\tau)$ denote the temporal autocorrelation of the process, which quantifies the average correlation between two samples of the process separated by time τ. The coherence time T_{coh} is then defined as the time lag during which the samples of the signal are sufficiently correlated, i.e., $R_{xx}(T_{coh}) = qR_{xx}(0)$, where q is the desired level of the correlation (e.g., $q = 98\%$). We borrow the above random sensing strategy from Fazel et. al. [13, 14]. This strategy in addition to the compressibility assumption mentioned earlier we can reconstruct back the field using Eq. (1)

$$\mathbf{y} = \mathbf{R}\mathbf{\Psi}\mathbf{v} + \mathbf{z} \tag{1}$$

where $\mathbf{z}$ represents the *sensing noise*, $\mathbf{\Psi}$ is the inverse Discrete Fourier Transform matrix, $\mathbf{v}$ is the sparse vector of Fourier coefficients and $\mathbf{R}$ is a $K \times N$ matrix–with K corresponding to the number of useful packets collected during T–which models the selection of correct packets. Each row consists of a single 1 in the position corresponding to the sensor contributing the useful packet. The FC can form $\mathbf{R}$

from the correctly received packets, since they carry the location tag. We emphasize the distinction between the sensing noise $\mathbf{z}$, which arises due to the limitations in the sensing devices, and the communication noise, which is a characteristic of the transmission system. The sensing noise appears as an additive term in Eq. 1, whereas the communication noise results in packet errors and its effect is captured in the matrix $\mathbf{R}$.

Figure 1: Per-node sensing rate λ_{1s} vs. the collection interval T, for $N = 1000$ nodes, $S = 10$, $T_p = 0.2$ sec, $b = 4$, $\alpha_s = 244$ packets and $\gamma_0 = 15$ dB.
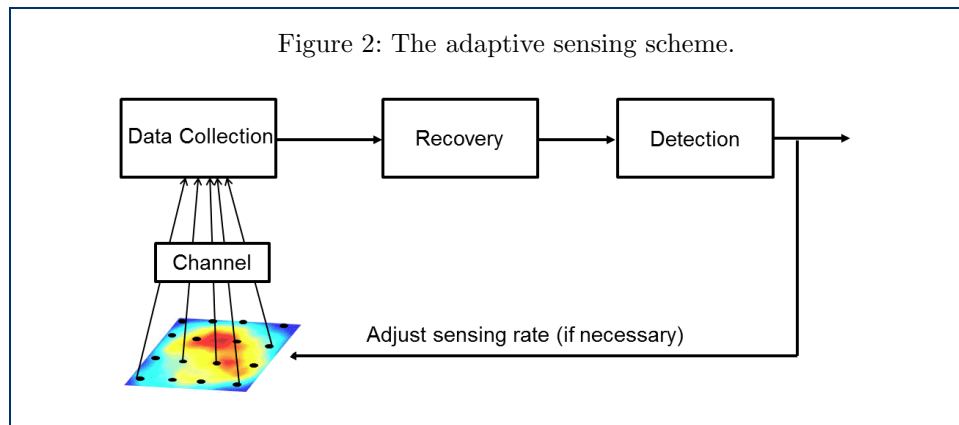

The FC then recovers the map of the field using sparse approximation algorithms [19]. It suffices to ensure that the FC collects a minimum number of packets, $N_s = \mathcal{O}(S \log N)$, picked uniformly at random, from different sensors, to guarantee accurate reconstruction of the field with very high probability. The random nature of the system architecture necessitates a probabilistic approach to system design using the notion of sufficient sensing probability [13] denoted by P_s. This is the probability with which full field reconstruction is guaranteed at the FC. Setting this probability to a desired target value, system optimization under a minimum-energy criterion yields the necessary design parameter, i.e., the per-node sensing rate λ_1. The minimum per-node sensing rate can be expressed in terms of the system parameters as shown in [14]

$$\lambda_{1s} = \frac{-1}{2NT_p \frac{b}{b+1}} \cdot W_0 \left(\frac{2NT_p \frac{b}{b+1} e^{\frac{b}{\gamma_0}}}{T} \log \left(1 - \frac{\alpha_s}{N} \right) \right) \tag{2}$$

where T_p is the packet duration, b is the packet detection threshold, α_s is the average number of packets that need to be collected in one observation interval T to meet the sufficient sensing probability, γ_0 is the nominal received SNR, and $W_0(\cdot)$ is the principal branch of the Lambert W function. (More details can be found in [14]). Note that as shown in Fig. 1, λ_{1s} depends on the collection interval T, which in turn must be lower than T_{coh}, for the stationarity assumption to hold.

3 Adaptive Sensing

In the above framework, the minimum per node sensing rate λ_{1s} can be determined based on the properties of the field and is then kept fixed throughout the entire sensing process. However, most fields of interest are seldom stationary and the coherence properties of a non-stationary dynamic field usually vary significantly over time. This calls for the design of an adaptive random sensing network for the monitoring of temporally varying fields. The objective is to transition from passive monitoring where only the map of the sensing field is reconstructed, to an active framework where the relevant information is extracted (e.g., target detection/tracking), and exploited to instruct the sensor nodes to adjust their sensing rates. In practice, we assume the coherence properties of a non-stationary field to be piecewise constant over time, such that given prior information about the field at a particular time instant it can be assumed to be stationary within a single coherence time interval. To this end, we employ a detection method, which uses the collected data to determine the attributes of the underlying field. The FC can then use this information to determine any appropriate modifications to the per-node sensing rate, i.e., increase, maintain or decrease the sensing rate. This cycle of sensing-decision-adjustment is illustrated in Fig. 4.

Figure 2: The adaptive sensing scheme.

For a given collection interval T, the corresponding per-node sensing rate can be determined using Eq. (2), as shown in Fig. 1. The proper choice of T however depends on the rate of variations in the field and is adaptively tuned. In particular, we use an approach based on target detection, where we assume that an object/target model of interest is known beforehand. This is a common assumption in most supervised pattern recognition tasks. Given a reconstructed map, the location of targets is first estimated. Using this knowledge, we then estimate the parameters of the object model from the map. These parameters describe the detection system's understanding of the field, and can be used to generate the map of the field. Comparing this model-based map with the observed one reconstructed using sparse approximation algorithms, provides us with a reliability metric for reconstruction. In other words, if there is a difference in what the algorithm expects to see and what it sees, it indicates an error in either the acquisition of the field or the algorithm's understanding of the map. In either case, the FC needs to adapt its sensing rate. We specifically discuss our methods in the context of the following two cases:

3.1 *Oversensing*

This situation corresponds to the case when there occurs redundant sensing because the per node sensing rate is much larger than the rate of change of the field i.e. $T << T_{coh}$. Although this case favors reconstruction using the RACS architecture, it leads to a wastage of communication resources. Thus we seek to lower the sensing rate in this case to an optimal point such that the accuracy of our end goal is not affected. Fig. 7 shows the result for of *oversensing* for the example discussed in Section 5. In this case, we devise a scheme to estimate the motion of targets using multiple frames. We use the term frame here and elsewhere in the paper to refer to the field reconstructed from samples acquired within a single sensing time duration.

3.2 *Undersensing*

In this scenario, the rate of sensing per node is insufficient and the field changes within one collection interval since $T > T_{coh}$. Thus the per-node sensing rate λ_{1s} needs to be increased. The targets are no longer steady within the duration of a collection interval T leading to blurring, which makes target detection challenging. The estimation task in this case is further complicated by the fact that the packets from different frames may have been collected during the interval T. This scenario leads to a violation of the stationarity assumptions made initially. The reconstructed map is thus blurred because of different packets originating from different frames (Fig. 3). In this case, we use a specific error metric based on model-fitting to estimate the reliability of reconstruction and object detection.

4 Simulation of a sample dynamic field

In this paper, we demonstrate the adaptive monitoring procedure using a sample model for the field. We demonstrate our adaptive scheme to adjust the per node sensing rate in RACS using a simulated example field in this paper. Using a realistic example allows us to control the coherence parameter T_{coh} and monitor the effect of changing the collection interval T in accordance with the adaptive algorithm.
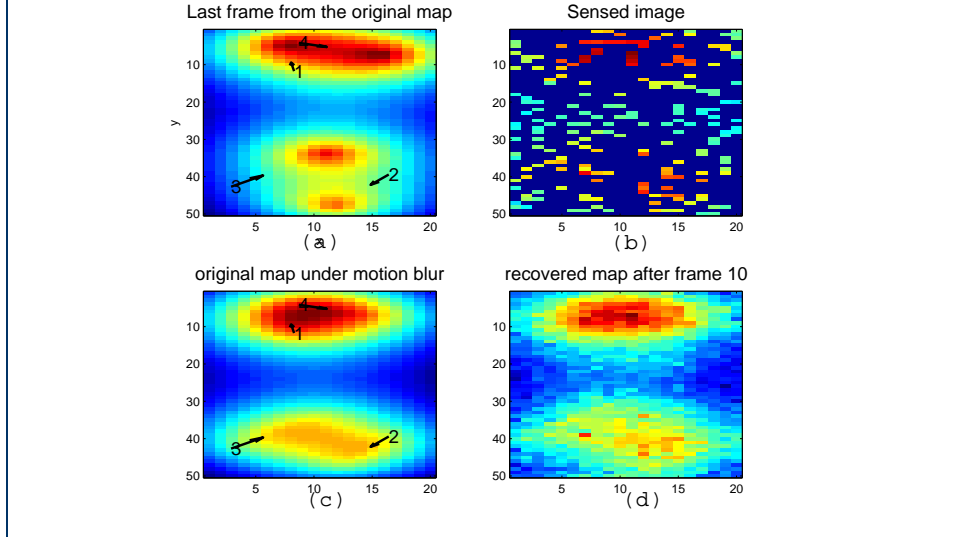
We first describe the example field that serves as our test case. Suppose, a field with M targets is being observed. Each target in the field is assumed to generate a signature (e.g., heat, sound, etc) decaying exponentially with distance from its location. At time t, the process observed by sensor node i at coordinate (x_i, y_i) is given by Eqn.(3)

$$u_i(t) = \sum_{m=1}^{M} A_m e^{-p\sqrt{(x_i-a_m(t))^2+(y_i-b_m(t))^2}} \tag{3}$$

where $a_m(t), b_m(t)$ are the coordinates of the mth target at time t, A_m is its strength, and p is the decay rate of the sources. The process then evolves over time as the sources move along random trajectories. Similar models are commonly found in the energy-based localization literature for static sensor networks [20, 21, 22] when the targets being detected are not moving.

Initially, the FC has no knowledge of the location of targets or the rate of variation in the field, i.e., the speed at which the targets are moving. It thus instructs the

Figure 3: Figures (a)-(d) shows the effect of the field changing at rate faster than the sensing rate. (a) is the original undistorted map that was sensed at T_{coh} while (c) shows the same field obtained through temporal blurring after an observation interval $T = 10 \times T_{coh}$. (d) is the field obtained as a result of sensing at a rate 10 times slower than the field's underlying rate of change. Notice how (d) is more similar to the blurred map (c) than (a). This distortion in the sensed field is an example of undersensing.
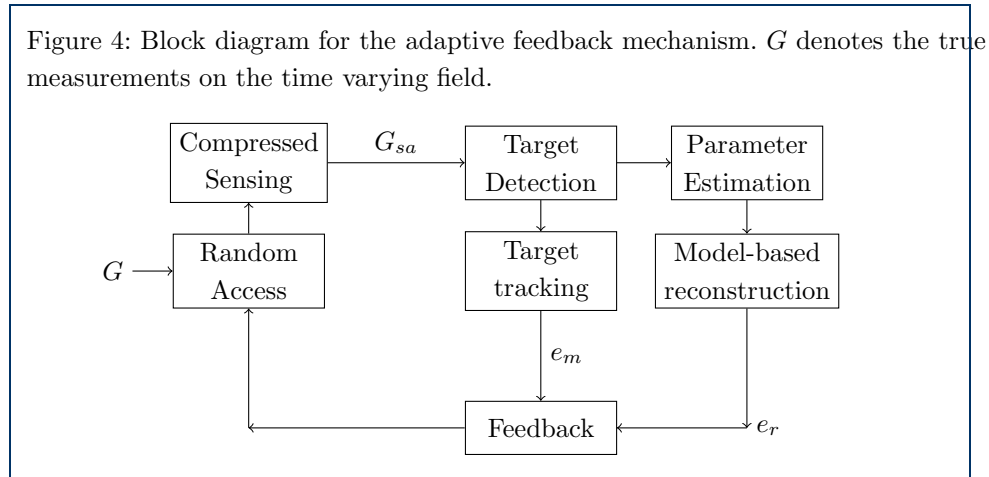

data collection to begin with an initial sensing rate $\lambda_{1s}^{\text{init}}$. The initial sensing rate is determined using historical data by setting the desired parameters in Eq. (2). Once the map of the field G^{sa} is recovered using sparse approximation techniques, the FC may now use the rate-adjustment algorithm described later to decide if the sensing duration T needs to be adjusted. The sensors employ the adjusted sensing rate in the ensuing sensing duration. In this paper, we discuss metrics for the proposed rate adjustment algorithm for this family of field models, although the framework itself is quite general.

For simulating a map of the field as defined above, we start with a set of randomly chosen parameters. In addition each target is assigned a random velocity and direction of movement. To simulate the *undersensing* case, we consider that collection occurs over N_b coherence time intervals (referred to as frames) where $T \approx N_b T_{coh}$. The randomly sampled packets are then collected uniformly over the last N_b frames (Fig. 3). This effectively leads to motion blurring of the targets as mentioned before. To simulate the *oversensing* case the collection interval is reduced to $T' = T_{coh}$ while each target's velocity is scaled by T/T_{coh} to make them appear to be moving slower. To deal with the issue of a finite field size, targets moving out of the field are replaced by new targets starting from the same location assigned a new random velocity and direction.

5 Adaptive Field Monitoring Scheme

In this section, we discuss specific algorithms and metrics extracted using the reconstructed map, which are used for the adaptive rate control of per node sensing in RACS. Although the algorithms discussed below have been adapted to this particular family of models used in the simulation example, we have attempted to layout broad steps whenever possible and provide a general scheme to be used in such cases. These algorithm stages have been presented in the block diagram shown in Fig.4.

Figure 4: Block diagram for the adaptive feedback mechanism. G denotes the true measurements on the time varying field.

5.1 Target Localization

Before taking any decisions about the current sensing state, we first detect and localize the targets in the image. Similar to [18], by involving target detection in the feedback process, we would like to ensure that the rate of sensing is optimized for the end goal of target detection.

Traditionally, most energy-based localization methods [20, 21, 22] assume that the number of targets is known in advance. In addition, they assume a field model for target signature decay allowing for parameter estimation by model-fitting. More recently, Kerse et al. proposed a method for direct target localization based on a standard sparse identification technique [18]. The advantage in their method is that target localization can be performed in a single step without first having to reconstruct the field. The number of targets can also be jointly estimated in this process using a sparsity constraint. However, the method still depends on exact knowledge of the field model.

To overcome this drawback, we propose a target localization algorithm based on local gradient ascent. The proposed method only requires that the target signatures be monotonically decaying away from the target. This technique is adapted from the mode seeking mean shift algorithm [23] which is frequently used for unsupervised clustering of data. Consequently, it can be interpreted as searching local peaks in the data histogram. In the task at hand, we are instead interested in finding local peaks in field intensity. We also modify the algorithm slightly to adapt to the discrete search space for this problem. Specifically, we start from a random initial point X on the map. A mean-shift based gradient ascent technique is then used to update

the location of this point in each iteration till a peak is found. Given the current location X we estimate an intensity weighted mean X_c for locations around X in a window of size W as shown in Eqn.(4). Note that the direction from X to X_c gives the direction of gradient ascent, along which X should be updated in the next iteration.

However in the current problem we are working on a space of discrete sensor locations. To make the algorithm better suited to such a scenario, we quantize the gradient direction to 8 angular bins. Depending on the direction of the gradient, X is then shifted to one of its 8 connected points. This process is explained using a schematic in Fig. 5. The iterations are repeated until X converges to a local maximum or reaches a point that has already been visited. The entire procedure is repeated multiple times with new random initial points till the peaks have been discovered. In practice, it is not necessary to traverse all points in the field to discover all the peaks (Fig.6) and we restrict the algorithm to a fixed number of iterations. The parameter W in this algorithm serves as a mask over which the gradient can be estimated more robustly. The target detection procedure is explained in Algorithm 1.

Algorithm 1: Modified gradient ascent algorithm for target detection on the acquired map. (See Fig.6)

Input: $P \times Q$ map G^{sa} reconstructed by RACS

Output: Location of targets $\mu_1, \mu_2, \ldots$ where the number of targets is initially unknown

Set maximum number of iterations to M, size of window to W

Cluster index $K \leftarrow 1$

for $iter \leftarrow 1$ **to** M **do**

   Randomly select an initial location X on the map.

   while X hasn't already been visited **do**

      Mark X as visited and belonging to cluster K

      //assuming we're going to find a new target

      Compute the new center of mass X_c within a window of size W around X.

      Compute the direction from X **to** X_c and quantize it into 8 bins between $\{-\pi/8, \pi/8, 3\pi/8 \ldots\}$ (Fig. 5)

      Update the position of X to one of the 8 adjacent positions based on the quantized direction (Fig. 5)

   end while

   Mark all points in this trail leading upto X as belonging to the same cluster as X

   if X belongs to the new cluster K **then**

      Append X as μ_K to the set of targets *//new target found*

      $K \leftarrow K + 1$

   end if

end

$$X_c = \frac{\sum_s I(s)K(s-X)s}{\sum_s I(s)K(s-X)}, s \in P \times Q \ grid$$

$$(4)$$

$$K(s-x) = \begin{cases} 1 & \text{if } ||s-x||_c \leq W \\ 0 & \text{if } otherwise \end{cases}$$
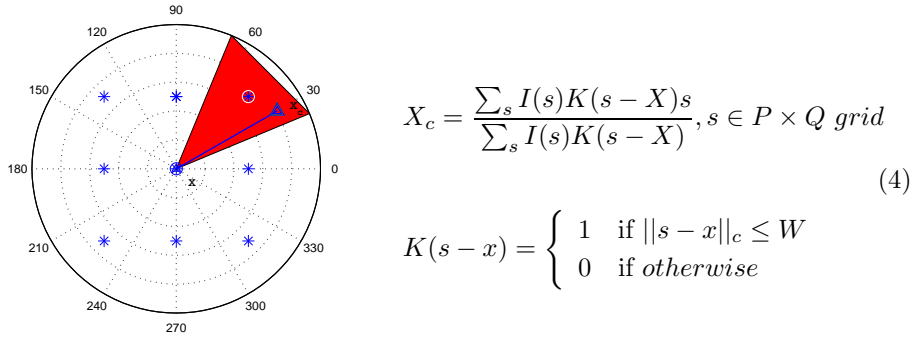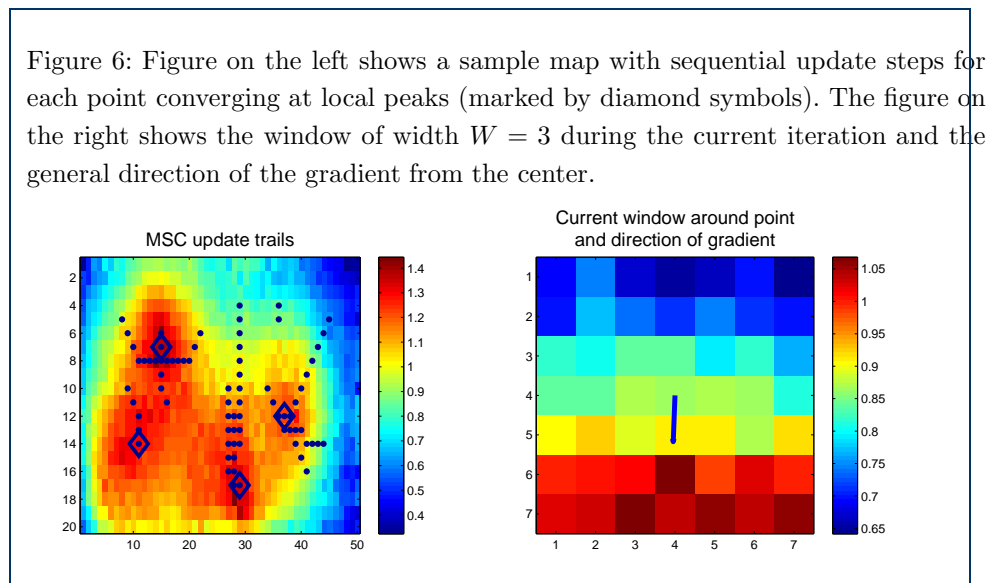
Figure 5: Modified discrete step update scheme using quantized directions. Asterisk(*) indicates points on the grid. I denotes the $P \times Q$ grid of intensity values and $K(.)$ denotes our particular choice of kernel function used for computing mean.
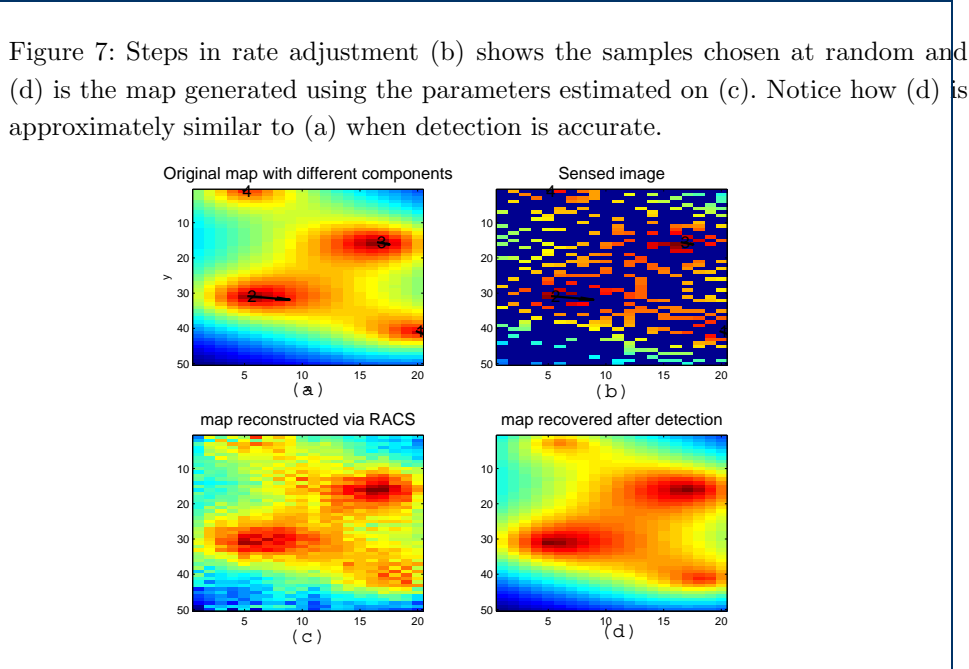
5.2 Parameter estimation

Once the positions a_m, b_m for the targets have been identified using the target localization algorithm described above, we try to obtain an MMSE estimate of the field parameters defined in Eq. (3) using a method of comparison by synthesis to match the acquired map G^{sa} against a model. The parameters of model denoted by G^{model} comprise the target locations (a_m, b_m), their respective strengths A_m and the decay parameters of the target signature (p).

We estimate the parameters using a nonlinear regression [24] technique to optimize an MMSE formulation as shown in Eqn.(5). We use the *nlinfit* function in MATLAB for solving this optimization problem. Since the optimization is not convex, the choice of initial points is important. We use the target localizations estimated earlier and a reasonable value for the decay as our initial point. The estimated parameters are then used to generate a map of the field G^{det} (Fig. 7(d)). G^{det} represents what the control algorithm expects the field to look like based on the target localizations and its knowledge of the model.

Figure 6: Figure on the left shows a sample map with sequential update steps for each point converging at local peaks (marked by diamond symbols). The figure on the right shows the window of width $W = 3$ during the current iteration and the general direction of the gradient from the center.

$$\mathbf{a}, \widehat{\mathbf{b}, \mathbf{A}}, p = \underset{a_m, b_m, A_m, p}{\arg\min} \sum_{i=1}^{P} \sum_{j=1}^{Q} \left(\sum_{m=1}^{M} G_{ij}^{model}(a_m, b_m, A_m, p) - G_{ij}^{sa} \right)^2 \tag{5}$$

$$G^{det} = G^{model}(\mathbf{a}, \widehat{\mathbf{b}, \mathbf{A}}, p) \tag{6}$$

Figure 7: Steps in rate adjustment (b) shows the samples chosen at random and (d) is the map generated using the parameters estimated on (c). Notice how (d) is approximately similar to (a) when detection is accurate.
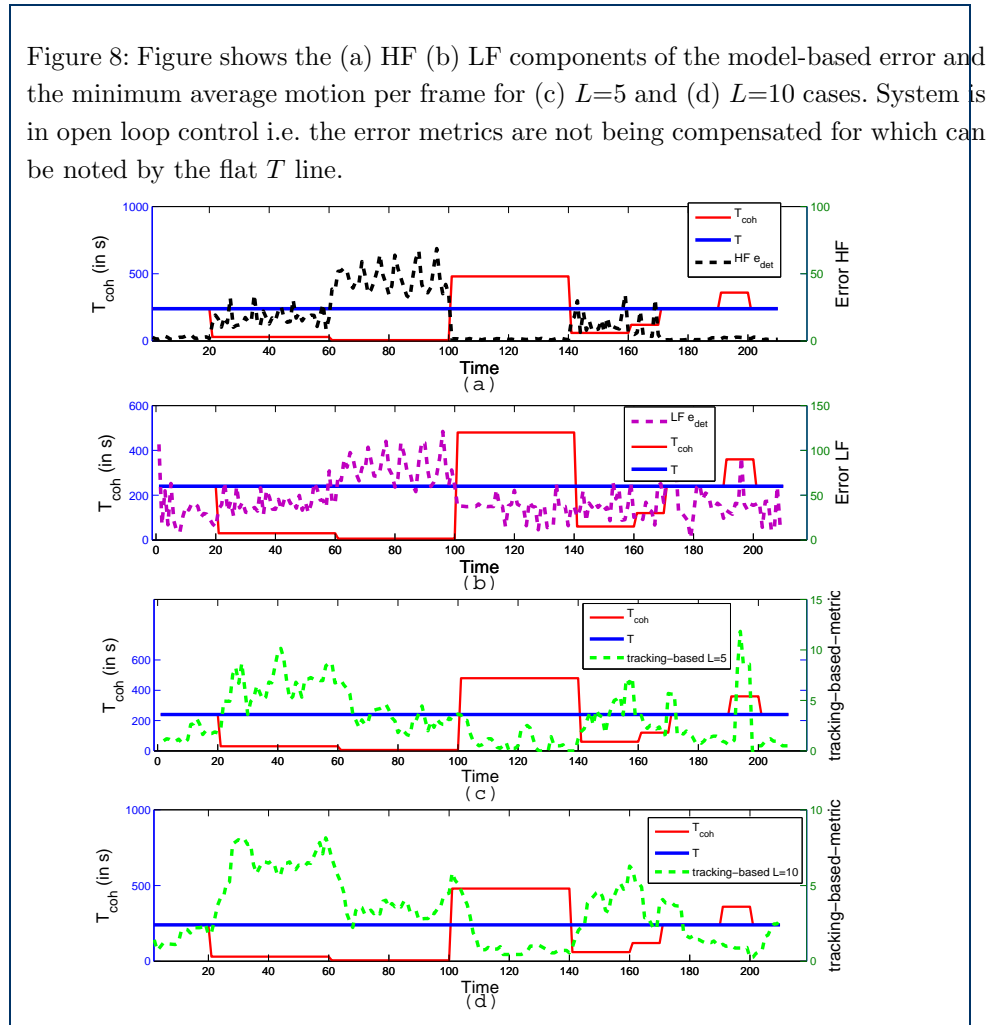

5.3 Reliability metric for reconstruction

We obtain a reliability metric for reconstruction based on the error in the model-fitting described above. We compare the model-based map G^{det} against the non-model-based map G^{sa} recovered via sparse approximation by projecting both of them on the DFT matrix, and decomposing the error into a high frequency (HF) and low frequency (LF) term. The intuition is that G^{sa} should be similar to G^{det} but for any errors resulting from issues in either sensing or localization. A gross mismatch between the coefficients will be captured by the LF term indicating an inaccurate detection while a large value of the HF term (denoted by e_r) is characteristic of a poor reconstruction due to violation of the assumptions in RACS (Fig. 8 (a) & (b)). The definition of this error term is shown in Eqn.(7) where $\mathcal{F}_{\mathcal{H}}$ denotes a 2D high pass filter and e_r is obtained by convolving it with the difference between the images G^{det} and G^{sa}. The cutoff frequency for the filters can be chosen from a previous estimate of the field. The error metric e_r thus comes in handy in the *undersensing* case when the acquired map G^{sa} is either blurred or poorly reconstructed.

$$e_r = \mathcal{F}_{\mathcal{H}} * (G^{det} - G^{sa}) \tag{7}$$

5.4 Measuring motion by target tracking

In addition to compensating for the dynamic nature of the field in the *undersensing* case, we would also like to eliminate any redundant sensing. Recall, that in the *oversensing* case, the field changes very slowly, yielding perfect reconstruction and detection. This makes it necessary to monitor the field over multiple frames for detecting any signs of *oversensing.*

To quantify motion in the field, we track the location of detected targets over the last L frames. Note that this is not trivial since the number of targets detected is not guaranteed to be consistent from one frame to another. Moreover, the target indices assigned by the detection algorithm are non-unique. To deal with this issue, we use a tracking approach based on dynamic programming that ensures tracking even if the object is not detected in some of the intermediate frames. More specifically, we recursively minimize the total distance moved by a target over L frames. If the targets are spaced sufficiently apart, this ensures tracking of the slowest moving target over multiple frames.

Figure 8: Figure shows the (a) HF (b) LF components of the model-based error and the minimum average motion per frame for (c) L=5 and (d) L=10 cases. System is in open loop control i.e. the error metrics are not being compensated for which can be noted by the flat T line.

Suppose $(\hat{a}_i^t, \hat{b}_i^t)$ denote the location of the ith target detected in frame t and let d_{ij}^t be defined as the distance between the ith target detected in the frame t and the jth

target detected in the frame $t + 1$. Then the net distance $D(n_1, n_2, \ldots, n_L)$ moved by a target over L frames and detected at the indices $\{n_1, n_2, \ldots, n_L\}$ is given by Eqn.(8).

$$D(n_1, n_2, \ldots, n_L) = \sum_{t=1}^{L-1} d_{n_t n_{t+1}}^t; \quad 1 \le n_t \le M_t \tag{8}$$

$$d_{ij}^t = \sqrt{(a_i^t - a_j^{t+1})^2 + (b_i^t - b_j^{t+1})^2} \tag{9}$$

where M_t is the total number of targets detected at each frame. The optimization problem then boils down to finding the sequence of object location indices $\{n_1, n_2, \ldots, n_L\}$ that minimizes the sum of total point to point distances over L frames. To normalize over the length of the temporal window, we use the minimum average distance moved as a tracking-based-metric as shown in Eqn.(10). This metric (denoted by e_m) is expected to be low when the targets move slowly in the *oversensing* case while it is expected to be much higher when the objects get blurred in the *undersensing* case.
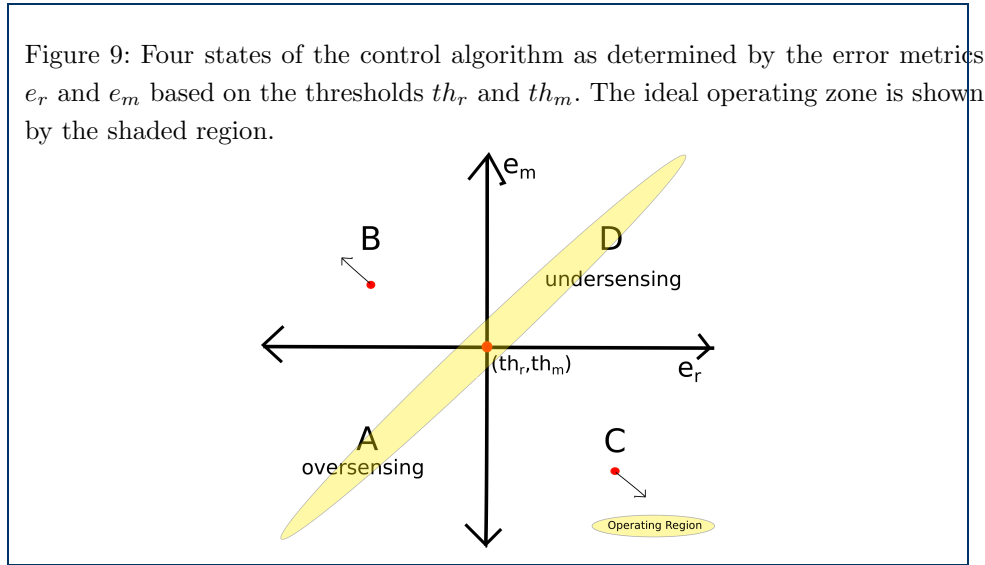
$$e_m = \frac{1}{L} \min_{n_1, \ldots, n_L} D(n_1, \ldots, n_L) \tag{10}$$

6 Feedback Algorithm for Rate Adjustment

After computing the error metrics e_r and e_m, a control algorithm is used to determine if the current sensing rate needs to be changed. This information is fed back to the FC which makes any necessary changes to the sensing rate, thereby establishing a closed-loop control. The objective of control is to minimize the reconstruction error based metric e_r at the same time ensuring that the targets move significantly from frame to frame as indicated by the tracking error metric e_r. This ensures that the system is in a sweet spot between *undersensing* and *oversensing*.

In this paper, we propose a dual threshold feedback scheme to keep the system in this "optimal" state. We define two parameters: a lower threshold th_m for the metric e_m and an upper threshold th_r for e_r respectively. In the proposed scheme, the value of these two parameters is tuned adaptively. The control algorithm itself consists of two modes: an "adjust" mode and a "calibrate" mode. The system's current mode is decided based on which of the four system states (A-D) it currently is in. These system states are shown in Fig.9.

In the "adjust" mode (States A and D in Table 1) the collection interval T is incremented or decremented by a scalar update ($\kappa > 1$) depending on the value of the metrics. In state A, the condition $e_m \le th_m$ indicates *oversensing* and hence the algorithm would decide to increase T, thereby decreasing sensing rate. In State D, the condition $e_r \ge th_r$ indicates that the system is *undersensing* which would result in the control decision to decrease T thereby increasing the sensing rate. The mutually opposing nature of these two conditions gives rise to a "stable" or buffer region for T (see Table 1) where it is not updated. In this mode we only adapt the width of the buffer region by adjusting the threshold values such that for the current value of T the control is just within the "stable" zone. Note, that it might not feasible to guess a value for th_m or th_r making this adaptive approach necessary.

Figure 9: Four states of the control algorithm as determined by the error metrics e_r and e_m based on the thresholds th_r and th_m. The ideal operating zone is shown by the shaded region.
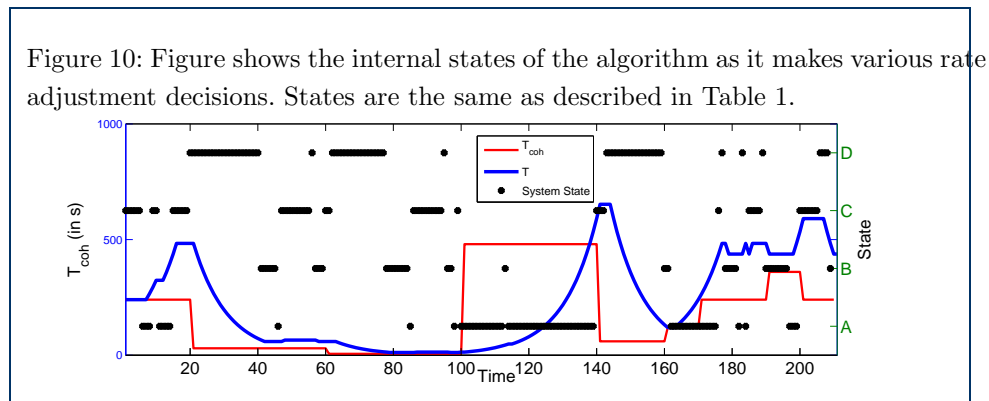

In practice, the thresholds can be initialized to extreme values such that the buffer region is wide enough and the system is guaranteed to start in State B. In subsequent intervals the thresholds are updated in the "calibration" mode to shrink this buffer region. The ideal operating zone, centered at (th_r, th_m), is shown by a shaded region in Fig. 9. If the system steps out of this operating zone, the thresholds are adjusted in a direction such that the current system state is contained within the operating zone. In the "calibration" mode, the thresholds are adjusted by predefined steps α, β such that the system stays within the operating zone. In other words, the aim of this adaptive control is to obtain the tightest stable region of control for T.

Table 1: Control Feedback Rules ("adjust" mode: A,D ; "calibrate" mode: B,C)

	$e_m < th_m$		$e_r > th_r$	
	Oversensing	**Stable Region**	**Undersensing**	

State	**A**	**B**	**C**	**D**
Condition	$e_m < th_m$ $e_r < th_r$	$e_m > th_m$ $e_r < th_r$	$e_m < th_m$ $e_r > th_r$	$e_r > th_r$ $e_m > th_m$
Decision	$T \uparrow$	*Narrow*	*Widen*	$T \downarrow$
Update	$T = \kappa T$	$th_r = th_r - \alpha$ $th_m = th_m + \beta$	$th_r = th_r + \alpha$ $th_m = th_m - \beta$	$T = T/\kappa$

To provide an intuition of how the scheme works, consider the graphical illustration of the proposed feedback algorithm in Fig. 9. At any time instant, the system state can be represented as a point on the $e_r e_m$ plane. The relative position of this point with respect to the thresholds th_r and th_m is then used to decide the next course of action. In general, the algorithm tries to maintain a narrow buffer region of control by keeping the operating point (e_r, e_m) close to (th_r, th_m). This can be achieved by either updating the sensing time period T (states A and D) or by adapting the thresholds (states B and C).
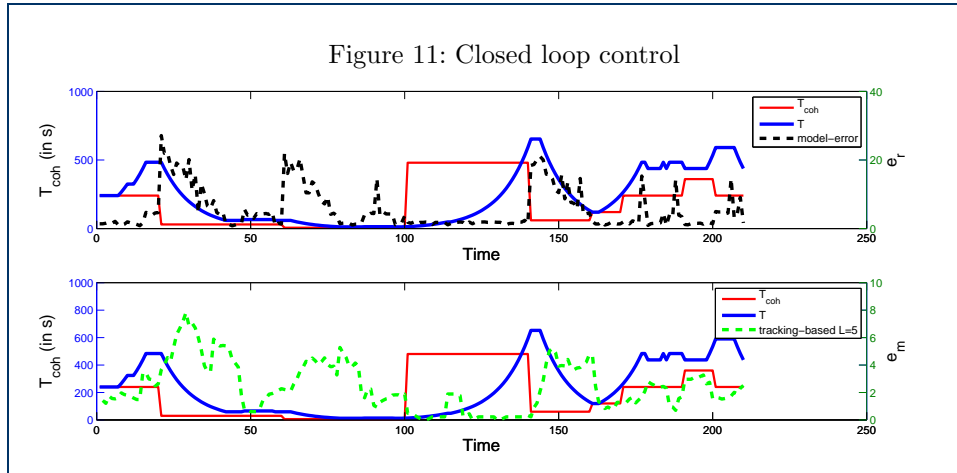
By observing how the states of the algorithm transition, we present an argument to show that the proposed algorithm is Bounded-Input-Bounded-Output (BIBO) stable. This means that for a finite coherence time T_{coh} the controlled variable viz. the sensing time period T is always bounded. This can be shown by considering each of the four states of operation A, B, C, D of the proposed algorithm above. Suppose, we are currently in state A (oversensing) and the feedback algorithm responds by increasing T. This leads to an increase in both e_m and e_r since fewer samples are sensed per unit time. This, in turn, prevents the system from staying in state A and is most likely to cause a transition to states B or C. Similarly, if the system is in state D (undersensing), the algorithm responds by decreasing T. This leads to the net effect that both e_r and e_m decrease, taking the system out of state D. When the system is in states B or C the thresholds are varied so that operating point is maintained close to the adapted thresholds. This ensures that the sensing time period T and in turn the error metrics e_r and e_m cannot grow unbounded for a given finite T_{coh}. As an example, note the state transitions in Fig. 10 which shows the internal states of the algorithm for a simulated control scenario.

Figure 10: Figure shows the internal states of the algorithm as it makes various rate adjustment decisions. States are the same as described in Table 1.

7 System operation

We presented an example of the control system in an open-loop operation earlier in Figure. 8 which showed the value of metrics e_r and e_m for a particular T_{coh} profile. Fig. 8 (c) and (d) suggest that a larger L, might be related to a slower response time of e_m as can be seen around $t=80,100$ when the field's T_{coh} changes. The error metrics based on model fitting (Fig. 8 (a) and (b)) are computed from the current frame and thus respond instantaneously to the changes in field coherence. It is also worth noting here that the HF errors (spatial high-frequency component) typically correspond to the reconstruction error in the *undersensing* case, while the LF error serves as a sanity check for detection.

In this section, we present an example of closed loop control by using the feedback mechanism (Table 1) discussed earlier to compensate for the error metrics as shown in Figure 11. Note that the collection time interval T now attempts to trace the underlying hidden parameter T_{coh} as the system tries to keep both e_r and e_m within bounds. Figure 10 provides a better insight into the system by showing the feedback algorithm's underlying internal state. Recall that States A and D correspond to oversensing and undersensing respectively while in states B and C the thresholds are adapted to achieve a tight buffer region of stability.

Figure 11: Closed loop control

8 Simulation Results

We run simulations with $M = 4$ targets and different velocity and decay parameters, using the proposed feedback algorithm for sensing rate adjustment. We evaluate the performance of the algorithm using two different criteria. First, average mean squared error (MSE) between the estimated sampling duration T (blue line in Fig.11) and the coherence time ground truth T_{coh} (red line in Fig.11) is computed as a measure of the algorithm's ability to track and adapt to changes in coherence time. Secondly, we calculate the F-score measure for target detection in each frame. This is a reasonable criterion for our task since the end goal of such an application would be target detection and tracking. To calculate the F-score of target detection in a frame, we check how many of the localized targets correspond to actual target locations. F-score weights both the precision and recall of target detection as follows

$$Precision = \frac{\# \text{ targets that were correctly localized}}{\# \text{ targets that were localized}}$$

$$Recall = \frac{\# \text{ targets that were correctly localized}}{\# \text{ targets actually present}}$$

$$F = \frac{2.Precision.Recall}{Precision + Recall} \tag{11}$$

8.1 Baseline algorithm

We compare the results against a baseline method based on correlation of the currently acquired field G^{sa} with the field acquired in the previous cycle as suggested in [15]. Control decisions are taken depending on two *fixed* high (δ_h) and low (δ_l) thresholds on the correlation value. T is incremented if correlation exceeds δ_h and decremented if correlation falls below δ_l. T is not updated if correlation lies between these two thresholds. The sampling duration T is changed using a scaling parameter κ as earlier.

Figure 12 shows the results of our simulation for the baseline and proposed methods. The results, averaged for each run, are shown for different parameter settings of velocity and decay p as defined in Eqn.(3). Remember that a high F-score indicates accurate target detection while a low MSE denotes accurate tracking of the underlying coherence time (T_{coh}).

Figure 12: Figure shows the values for averaged MSE and F-score evaluation criteria for **a)** correlation baseline mechanism and the **b)** proposed feedback algorithm at different velocity and decay parameter settings.
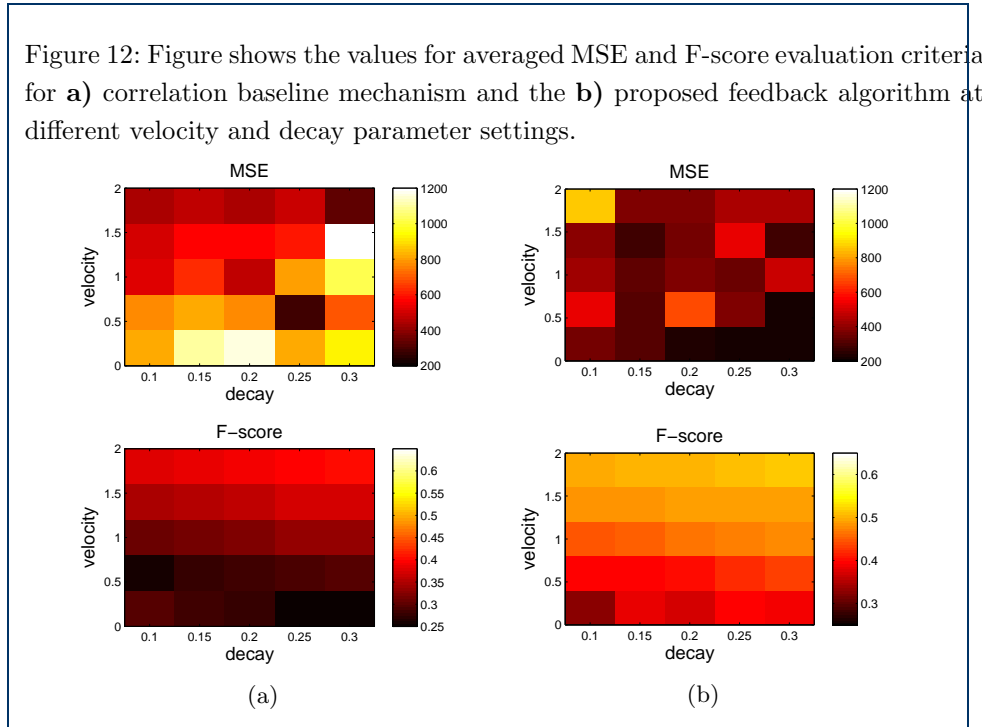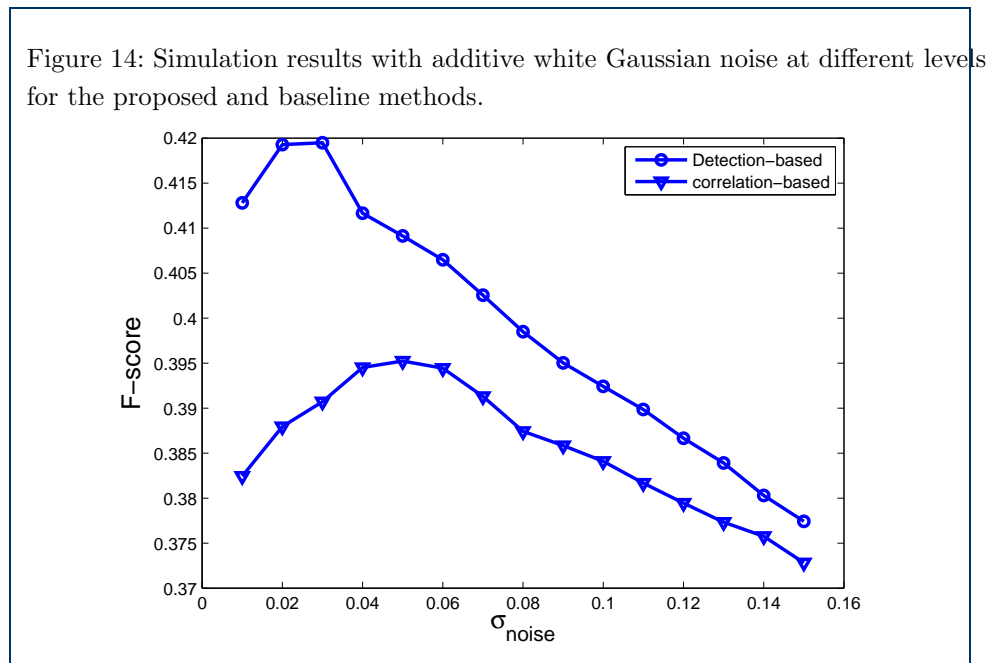
Figure 13a shows the results for the baseline method using correlation as a feedback metric. From the MSE and F-score plots it is evident that the baseline algorithm performance drops with increase in decay parameter p. This is expected since the targets with a lesser spread do not affect the correlation metric significantly. In contrast, for our proposed feedback algorithm the performance improves with increase in the decay parameter p since targets with lesser spread are easier to localize and hence field reconstruction is more accurate (Figure 13b). A similar trend can be seen for average MSE which is related inversely to the target detection F-score. In other words, this might indicate that poor tracking of the underlying T_{coh} is related to poor target detection, thereby making a case for the proposed sensing rate adjustment in this paper.

Since velocity for each target is assigned uniformly at random in the simulation, we control the velocity range that can be assigned to each target as a parameter instead. We note that increase in velocity has a positive influence on the performance of both the algorithms. This can be easily explained by considering that an increase in velocity of the targets makes it easier to discriminate between the oversensing, adequate sensing and undersensing states. This holds for both the algorithms and hence we notice a general increase in F-score and decrease in MSE with increase in velocity. Finally, we also note that our proposed detection-based feedback mechanism outperforms the baseline correlation-based method in terms of the accuracy of target detection as indicated by the higher F-score. The average target detection F-score obtained for the baseline algorithm was 0.32 while our proposed method achieved an average F-score of 0.45.

8.2 Simulations with measurement noise

Next, we test the noise robustness of our proposed method by adding synthetic sensor or measurement noise to the field. We simulate measurement noise by adding zero mean white Gaussian noise at different amplitudes (σ_{noise}) while keeping the original field's intensity same. Results are averaged on full runs for different noise level and the F-score for target detection accuracy is shown in Figure 14. Other simulation factors such as velocity-scale and decay were held constant at 1 and 0.3 respectively for the purposes of this simulation.

The results show a clear degradation in F-score for both the proposed detection-based and baseline correlation-based methods, as the signal-to-noise ratio decreases. However, we note that for the baseline algorithm the F-score initially increases with decrease in SNR hinting at suboptimality in processing. On further investigation we found this peculiar trait to be an artifact of the control algorithm and the choice of metric used in the baseline scheme. As noise is added to the field value of the correlation metric drops, which leads the FC to confuse the current state as undersensing. This inference leads to a monotonic decrease in sampling duration T, temporarily increasing performance at the cost of energy efficiency. The target detection F-score peaks around $\sigma_{noise} = 0.05$ beyond which the accuracy of target-detection algorithm is significantly affected. On the other hand, this artifact doesn't affect the proposed method significantly, since the metric used for our control algorithm is based directly on target detection. As a result, we note that our proposed method exhibits a more graceful degradation in performance with increase in noise without compromising on energy efficiency.

Figure 14: Simulation results with additive white Gaussian noise at different levels for the proposed and baseline methods.

9 Conclusion

In this work, we present a rate adjustment scheme for Random Access Compress Sensing (RACS) to monitor and compensate for the rate of change in time-varying

fields. Adjustment of sensing rate for RACS is motivated by the trade-off between energy efficiency and the accuracy of tracking targets of interest. Although direct estimation of the coherence time might seem to be the best approach for sensing a time varying field, we note that it is not directly related to our end goal of target detection in the field. We also observe that the reconstruction error in RACS has a complex relation to the current coherence time of the field and factors like the position of targets, their velocity etc. Thus by making the algorithm depend on the detection and localization of targets, we ensure that the rate adjustment is tied in to our actual objective.

In this paper, we propose two unsupervised metrics to inform the rate adjustment scheme. A model-based field reconstruction is done after target localization for each acquired frame, and the model fit error e_r is used as a metric to detect cases when the FC is undersensing. To account for oversensing in operation we define a measure for the motion of detected targets in the past L frames. This motion based metric e_m indicates any redundancy in sensing and can be used to decrease the collection time interval T if needed.

We proposed a dual threshold based feedback mechanism using these error metrics for rate sensing adjustments. The technique assumes minimum prior knowledge and adapts the threshold in an online fashion using reasonable assumptions about the field. We also show that the proposed control mechanism is BIBO stable. In addition, we compare our results against a baseline algorithm that uses temporal correlation of the acquired fields and show that the proposed rate adjustment mechanism performs better on an average in terms of target detection accuracy even in the presence of noise.

Author details
[1]Signal Analysis and Interpretation Laboratory (SAIL),
Electrical Engineering,
University of Southern California, Los Angeles, CA,. [2]Department of Electrical and Computer Engineering,
Northeastern University, Boston, MA, USA.

References
 1. Candes, E.J., Romberg, J., Tao, T.: Stable signal recovery from incomplete and inaccurate measurements. Comm. Pure Appl. Math. **59**, 1207–1223 (2005)
 2. Candes, E.J., Romberg, J., Tao, T.: Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. IEEE Transactions on Information Theory **52**, 489–509 (2006)
 3. Candes, E., Romberg, J.: Sparsity and incoherence in compressive sampling. Inverse Problems **23**, 969–985 (2006)
 4. Bajwa, W.U., Haupt, J., Sayeed, A.M., Nowak, R.: Compressive Wireless Sensing. In: 5th Int. Conf. Information Processing in Sensor Networks (IPSN'06), pp. 134–142 (2006)
 5. Bajwa, W.U., Haupt, J., Sayeed, A.M., Nowak, R.: Joint source-channel communication for distributed estimation in sensor networks. IEEE Transactions on Information Theory **53**(10), 3629–3653 (2007)
 6. Chou, C.T., Rana, R., Hu, W.: Energy Efficient Information Collection in Wireless Sensor Networks Using Adaptive Compressive Sensing. In: IEEE 34th Conference on Local Computer Networks (LCN), pp. 443–450 (2009)
 7. Luo, C., Wu, F., Sun, J., Chen, C.W.: Compressive data gathering for large-scale wireless sensor networks. In: Proceedings of the 15th Annual International Conference on Mobile Computing and Networking. MobiCom, pp. 145–156 (2009)
 8. Meng, J., Li, H., Han, Z.: Sparse Event Detection in Wireless Sensor Networks Using Compressive Sensing. In: 43rd Annual Conference on Information Sciences and Systems (CISS), pp. 181–185 (2009)
 9. Masiero, R., Quer, G., Munaretto, D., Rossi, M., Widmer, J., Zorzi, M.: Data acquisition through joint compressive sensing and principal component analysis. In: Proceedings of the 28th IEEE Conference on Global Telecommunications. GLOBECOM'09, pp. 1271–1276 (2009)

10. Masiero, R., Quer, G., Rossi, M., Zorzi, M.: A bayesian analysis of compressive sensing data recovery in wireless sensor networks. In: ICUMT'09, pp. 1–6 (2009)
11. Lee, S., Pattem, S., Sathiamoorthy, M., Krishnamachari, B., Ortega, A.: Spatially-localized compressed sensing and routing in multi-hop sensor networks. In: GeoSensor Networks. Lecture Notes in Computer Science, vol. 5659, pp. 11–20. Springer, ??? (2009)
12. Motamedvaziri, D., Saligrama, V., Castanon, D.: Decentralized compressive sensing. In: Communication, Control, and Computing (Allerton), 2010 48th Annual Allerton Conference On, pp. 607–614 (2010)
13. Fazel, F., Fazel, M., Stojanovic, M.: Random access compressed sensing for energy-efficient underwater sensor networks. IEEE Journal on Selected Areas in Communications (JSAC) **29**(8) (2011)
14. Fazel, F., Fazel, M., Stojanovic, M.: Random access compressed sensing over fading and noisy communication channels. Wireless Communications, IEEE Transactions on **12**(5), 2114–2125 (2013)
15. Fazel, F., Fazel, M., Stojanovic, M.: Random access sensor networks: Field reconstruction from incomplete data. In: Information Theory and Applications Workshop (ITA), 2012, pp. 300–305 (2012). IEEE
16. Reed, S., Petillot, Y., Bell, J.: An automatic approach to the detection and extraction of mine features in sidescan sonar. Oceanic Engineering, IEEE Journal of **28**(1), 90–105 (2003)
17. Kumar, N., Tan, Q.F., Narayanan, S.S.: Object classification in sidescan sonar images with sparse representation techniques. In: Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference On, pp. 1333–1336 (2012). IEEE
18. Kerse, K., Fazel, F., Stojanovic, M.: Target localization and tracking in a random access sensor network. In: Signals, Systems and Computers, 2013 Asilomar Conference On, pp. 103–107 (2013). IEEE
19. Tropp, J.A., Wright, S.J.: Computational methods for sparse solution of linear inverse problems. Proceedings of the IEEE, special issue, Applications of sparse representation and compressive sensing **98**(6), 948–958 (2010)
20. Li, D., Hu, Y.H.: Energy-based collaborative source localization using acoustic microsensor array. EURASIP Journal on Applied Signal Processing **2003**, 321–337 (2003)
21. Sheng, X., Hu, Y.-H.: Maximum likelihood multiple-source localization using acoustic energy measurements with wireless sensor networks. Signal Processing, IEEE Transactions on **53**(1), 44–53 (2005)
22. Meng, W., Xiao, W., Xie, L.: An efficient em algorithm for energy-based multisource localization in wireless sensor networks. Instrumentation and Measurement, IEEE Transactions on **60**(3), 1017–1027 (2011)
23. Cheng, Y.: Mean shift, mode seeking, and clustering. Pattern Analysis and Machine Intelligence, IEEE Transactions on **17**(8), 790–799 (1995)
24. Holland, P.W., Welsch, R.E.: Robust regression using iteratively reweighted least-squares. Communications in Statistics-Theory and Methods **6**(9), 813–827 (1977)