# Compressive Wireless Data Transmissions under Channel Perturbation

Jie Zhao and Xin Wang
The Department of Electrical and Computer Engineering
State University of New York at Stony Brook
Stony Brook, New York 11794
{jiezhao, xwang}@ece.sunysb.edu

*Abstract*—**Compressed sensing (CS) technique has attracted a lot of recent research interests in mathematics and signal processing fields. Literature studies often exploit CS at the receiver side to sub-sample the receiving signals to reduce the sampling rate and processing overhead. It would be of great benefit if it is possible to exploit CS at the transmitter side to reduce the redundancy of the data before transmission to conserve precious wireless bandwidth. Different from receiver-side sub-sampling, the sub-sampled transmitting data may be perturbed by the dynamics of wireless channels and experience higher overall noise.**

**In this paper, we propose a set of mechanisms to enable *compressive wireless data transmissions*. Specifically, we investigate the impacts of imperfect channel equalization on the data reconstruction, and propose a comprehensive signal recovery algorithm to cope with the perturbations introduced by wireless channels. Simulation results demonstrate that our proposed schemes can effectively reduce the effects of dynamic wireless channels on the data reconstruction and maintain the performance comparable to that of traditional communication scheme which does not apply CS to compress data. This indicates that it is promising to exploit CS to reduce the communication data thus bandwidth requirement. Transmission data reduction can complement existing efforts of improving wireless channel capacity to support the quick growth of wireless applications.**

*Index Terms*—**compressed sensing; imperfect channel estimation; adaptive measurement; robust data transmission; reconstruction algorithm.**

## I. Introduction

Compressed sensing (CS) [1], or compressive sampling [2], has gained increased interests over the past few years. The conventional Nyquist sampling theory requires the analog to digital converter (ADC) to sample the Radio Frequency (RF) signals with the rate at least twice the signal bandwidth. For high-frequency applications, this brings great challenge and cost to the ADC design. The recent CS techniques provide a promising venue to reduce the need for high speed ADCs.

The essence of CS is to exploit the sparsity within signals to significantly reduce the sampling rate while still capturing the information at similar quality. CS theory presents that if an $N$-dimensional signal is $K$-sparse in a certain domain, then with an overwhelming probability, one can fully recover the signal by taking measurements at the order of $K \log N$. This indicates that one can reconstruct the sparse signals with very few samples (much smaller than what Nyquist rate suggests).

Compressed sensing, a novel paradigm, has been successfully applied to various signal processing fields. In imaging processing, specifically, CS has achieved a level of maturity. Though not as widely investigated as in signal processing, CS has also been applied to several realms in communications, such as signal detection, sparse channel estimation [3] [4], channel-source coding [5] and data gathering [6].

The vast amount of existing work mainly focus on the application of CS at the receiver side and studying its impact on signal detection and recovery. Wireless transmission bandwidth is well known to be limited and many efforts have been made in the past several decades to improve the wireless channel capacity. As a *complementary* technique, CS can fundamentally reduce the data rate from the source, thus requiring much lower bandwidth to transmit the user signals. Although promising, there are very limited studies to investigate the possibility and potential problems to apply CS at the source.

Conventionally, to reduce bandwidth need, signals may be compressed after they are sampled at the cost of higher computation overhead at the encoder. Signal compressions are commonly used for reducing multimedia data, especially for their transmissions over wireless networks. Due to the complexity of conventional signal compression techniques, they are not commonly used for many other types of data, and the quick growth of data have created big pressure for every field. For example, sensor data are well known to have a lot of redundancy. Due to the limited computation capability of low cost sensors, raw data are often transmitted. By reducing data through simple random sampling at the source and exploiting advanced decoder to recover the data, CS is a promising technique to apply to data reduction in sensor networks. Additionally, CS can be applied to sub-sample/compress wide-band signals to reduce transmission cost at the transmitter and ADC speed at the receiver.

As many signals have redundancy, reducing the data through

978-1-4799-4657-0/14/$31.00 © 2014 IEEE

CS at the source will greatly reduce the transmission rate thus the burden to the wireless networks. This has the potential to fundamentally relieve the problem due to the constant increase of data and the limitation in wireless bandwidth. However, this also faces many challenges. The transmission of compressed information will make data bits of the signals more significant. On the other hand, the wireless channel is well known to be unstable and could introduce loss and errors to the transmitted signals.

In order to maintain the quality of information, a reliable and robust transmission scheme is helpful. Retransmission schemes have been widely used to improve the delivery rate, however, simply relying on retransmissions is not efficient. Overhead such as round trip latency and Automatic Repeat Request (ARQ) traffic may reduce the throughput significantly. Instead, the random sampling feature of CS can be exploited to mitigate loss by simply increasing the sampling rate. In [9], the authors show that for the Fourier random sampling scheme, over sampling is much less expensive than competing erasure coding methods and performs just as well. However, the authors assume the loss to be random without considering the actual channel impact.

Besides random loss, the variation of channel quality and states could lead to channel estimation error, which will in turn lead to difficulty in signal recovery and consequent the increase of recovery errors. The CS recovery error grows as the noise of the sampled signal becomes larger. This motivates many studies on the impact of noise on sparse recovery. However, existing work mostly only present the recovery quality as a function of the noise level without providing a scheme to better reconstruct the signal. Although this may be enough in the presence of bounded small noise, the high total noise contributed by the channel estimation error could significantly compromise the quality of CS signal recovery. Simply increasing the sampling rate, e.g. oversampling in [9], may help to alleviate the problem, but cannot completely eliminate the channel impacts. The increase of sampling rate will also reduce the efficiency of compressive data communications. Different from simple signal recovery in signal processing domain, the delay of the feedback from the receiver thus the delay for the source to respond to the recovery quality change. Finally, sampling rate cannot be adapted too frequently to avoid system oscillation.

The aim of this work is to investigate the possibility and techniques to support robust compressive transmissions over dynamic wireless channels. Our proposed scheme has three important components: adaptive sampling in response to estimated reconstruction quality, channel equalization to minimize the side impacts due to channel dynamics, and robust compressive signal recovery under large noise. The three components work interactively to improve the overall transmission quality. Different from the literature work which mainly target for theoretical analysis of the benefits and limitation of CS, we pay special attention to the difficulty resulted from the practical wireless transmissions. Our schemes are designed to be efficient to run in the practical system, and our contributions can be summarized as follows:

1) We propose a framework for robust transmission of compressive data over dynamic wireless channels. We expect the reduction of source data at the transmitter (rather than receiver) through simple compressive sampling helps conserve the expensive spectrum resources.
2) We investigate the possibility and methodologies in improving the signal reconstruction quality in the presence of large noise as a result of wireless transmissions. This is important because the sub-sampled data are more sensitive to wireless channel dynamics.
3) We propose adaptive compressed sampling algorithms to combat the signal recovery errors due to channel dynamics, channel equalization inaccuracy, and reconstruction errors.
4) We perform extensive simulations to evaluate the efficiency of our algorithm in supporting robust compressed wireless communications.

We would like this study to serve as a basis for future research on enabling compressive wireless network communications.

The remainder of the paper is organized as follows. Section II discusses related work. Section III gives some background of compressed sensing. We present our system model in Section IV, and describe our proposed CS-based data transmission methodology in Section V. Section VI shows the simulation results and Section VII concludes the work.

## II. Related Work

CS theory has been widely applied to signal processing fields (for applications such as detection and estimation) to reduce the signal sampling rate directly. However, there are very limited studies to apply CS to reduce data before transmission.

In [9], Z. Charbiwala *et al.* observe that the stochastic nature of wireless link losses and short-term sensor malfunctions do not disturb the performances of reconstruction schemes at the decoder, and that random losses are indistinguishable from an *a priori* lower measurement rate, and then propose a oversampling framework to ensure robust data transmission in sensor networks. Similarly, S. Kadhe et. al in [10] propose to integrate the CS framework with real expander codes (RECs), coined as CS-REC, for robust data transmission. Simulations show that CS-REC can achieve the recovery performance close to the case where there is no data loss.

However, as we mentioned in Section I, besides random loss, the channel dynamics and estimation error could significantly impact the transmission quality. A large increase of transmission rate will compromise the benefit of transmission of compressive data in wireless networks.

In [11], the potential of the compressed sensing (CS) paradigm for video streaming in Wireless Multimedia Sensor Networks is investigated. The authors only studied the effect of key video parameters (i.e., quantization, CS samples per frame, and channel encoding rate) on the received video quality when transmitting CS images through a wireless channel, but didn't provide concrete schemes to maintain or increase the transmission quality under severe channel conditions.

Different from the literature work which targets to address a specific issue, in this work, we systematically investigate the problems in different phases of wireless communications and propose a set of schemes to ensure the reliable transmission of compressive signals in wireless networks. Particularly, to combat channel equalization inaccuracy and consequently the increase of symbol errors and noise in compressed samples, we propose an a novel recovery methodology to more accurately reconstruct the signal in the presence of large noise and an adaptive compressive sampling scheme to respond to the recovery error over a longer time scale.

## III. Preliminaries

In this section, we will present some compressed sensing background. And we will use all the notations consistently throughout this paper.

In general, one needs $N$ measurements to fully recover an $N$-dimensional signal. However, in the recently emerged field of compressed sensing, the compressed sensing theory states a rather surprising result: if an $N$-dimensional signal is sparse in certain domain, one can fully recover the signal by using only $\Omega(\log N)$ linear measurements.

The main idea of compressed sensing is to take advantage of the sparsity within the signal to significantly reduce the sampling rate. An $N$-dimensional signal $\mathbf{d}$ is considered to be $K$-sparse in a domain (also called a dictionary matrix) $\mathbf{\Psi} \in \mathbb{C}^{N\times N}$ if there exists an $N$-dimensional vector $\mathbf{x} \in \mathbb{R}^{N\times 1}$ so that $\mathbf{d} = \mathbf{\Psi}\mathbf{x}$ and $\mathbf{x}$ has at most $K$ non-zero entries ($K \ll N$).

If one performs linear measurements of the signal $\mathbf{d}$ with a measurement matrix $\mathbf{\Phi}$, then one can consider the obtained linear measurements $\mathbf{y}$, possibly affected by noise as:

$$\mathbf{y} = \mathbf{\Phi}\mathbf{\Psi}\mathbf{x} + \mathbf{n} = \mathbf{A}\mathbf{x} + \mathbf{n}, \tag{1}$$

where the measurements are $\mathbf{y} \in \mathbb{R}^{M\times 1}$, the sparse vector $\mathbf{x} \in \mathbb{R}^{N\times 1}$, the additive noise $\mathbf{n} \in \mathbb{R}^{M\times 1}$, the sensing matrix $\mathbf{A} \in \mathbb{R}^{M\times N}$, and $M < N$. $\mathbf{A}$ is essentially the product of the measurement matrix and the dictionary matrix: $\mathbf{A} = \mathbf{\Phi}\mathbf{\Psi}$, where $\mathbf{\Phi} \in \mathbb{C}^{M\times N}$, $\mathbf{\Psi} \in \mathbb{C}^{N\times N}$. We notice that different from the notations above, a small number existing works call $\mathbf{\Phi}$ the sensing matrix. In order to avoid inconsistency and misunderstanding, we will consistently regard $\mathbf{\Phi}$ as the measurement matrix, and $\mathbf{A} = \mathbf{\Phi}\mathbf{\Psi}$ the sensing matrix.

Obviously, the number of measurements is smaller than the number of variables in Equation 1, and this is an under-determined equation system. Candès *et al.* show in [12] that the under-determined equation system can be solved provided that:

1) The vector $\mathbf{x}$ is sparse, i.e., only few ($K$) elements in $\mathbf{x}$ are non-zero.

$$K = |\{x_i | x_i \neq 0, i = 1, \ldots, N\}| \tag{2}$$

$\mathbf{x}$ can also be approximated sparsely if it is compressible, meaning that its coefficients sorted by magnitude decay rapidly to zero.

2) The sensing matrix $\mathbf{A}$ obeys the Restricted Isometry Property (RIP) with isometry constant $\delta_K > 0$, defined as follows:

$$(1 - \delta_K) \|\mathbf{x}\|_{\ell_2}^2 \leq \|\mathbf{A}\mathbf{x}\|_{\ell_2}^2 \leq (1 + \delta_K) \|\mathbf{x}\|_{\ell_2}^2, \tag{3}$$

for any at most $K$-sparse vector $\mathbf{x}$ such that:

$$\delta_K + \delta_{2K} + \delta_{3K} < 1. \tag{4}$$

3) The measurement process captures a sufficient amount of measurements $M$ :

$$M \geq cK \log\left(\frac{N}{K}\right), \tag{5}$$

where $c$ is a fairly small constant. Further details can be found in [18].

Given the measurements $\mathbf{y}$, the unknown sparse vector $\mathbf{x}$ can be reconstructed by solving the following convex optimization problem:

$$\min \quad \|\mathbf{x}\|_{l_1} \tag{6}$$

$$\text{s.t.} \quad \|\mathbf{\Phi}\mathbf{d} - \mathbf{y}\|_{l_2} \leq \epsilon \tag{7}$$

$$\mathbf{d} = \mathbf{\Psi}\mathbf{x} \tag{8}$$

where the parameter $\epsilon$ is the bound of the error caused by noise $\mathbf{n}$, $l_p$ means the $l_p$-norm ($p = 1, 2, ...$). The solution can also be expressed as:

$$\hat{\mathbf{x}} = \underset{\mathbf{u}:\ \|\mathbf{y} - \mathbf{A}\mathbf{u}\|_{l_2} \leq \epsilon}{\operatorname{argmin}} \|\mathbf{u}\|_{l_1}, \tag{9}$$

The signal $\mathbf{d} = \mathbf{\Psi}\mathbf{x}$ can then be recovered as $\hat{\mathbf{d}} = \mathbf{\Psi}\hat{\mathbf{x}}$.

The form of the optimization problem in (9) is known as LASSO [13] or BPDN [14] and also some other variations such as the Dantzig selector [15]. In addition to the convex optimization approach to reconstruction in compressed sensing, there exist several iterative/greedy algorithms such as IHT [16] and Cosamp [8]. Such convex or greedy approaches are generally called reconstruction algorithms.

## IV. Problem Statement and System Model

We consider integrating CS into data transmission where a transmitter will send sub-sampled signals to a receiver through a wireless link. We expect the reduction of data at the source to help greatly relieve the transmission burden on the bandwidth-limited wireless networks. Different from conventional work on applying CS at the receivers, the lossy and dynamic wireless links have big impact on the information transmitted and the data recovery. Sepcifically, the inaccurate channel estimation as a result of channel dynamics could significantly reduce the signal recovery quality and this is often not considered by the literature work. The goal of our work is to propose various strategies to ensure reliable transmission of compressive data.

### A. Compressed Sensing Elements

We first introduce the CS elements used in our framework.

*1) Sparse Dictionary:* $\mathbf{\Psi}$

Wavelet Transform and Fourier Transform are commonly used as dictionary matrix in the literature work of compressive sensing. As many natural signals of interest are sparse in frequency domain (Fourier Domain), throughout this work, we adopt inverse Fourier Transform as the sparse dictionary $\mathbf{\Psi}$. Another benefit of choosing Fourier Transform is that we can take advantage of the symmetry of non-zero elements in $\mathbf{x}$ due to Fourier Transform characteristics to improve the signal recovery quality, which will be explained further in Section V.

*2) Measurement Matrix:* $\mathbf{\Phi}$

In traditional CS applications where a sparse vector $\mathbf{x}$ is measured/sampled, instead of $\mathbf{d}$, the design of sensing matrix $\mathbf{A}$ is an important theoretical problem because $\mathbf{A}$ needs to meet the requirements describe in Section III in order to enable successful recovery.

Similarly, in the circumstances that the signal is not sampled/measured in its sparse domain, we will need to design carefully the measurement matrix $\mathbf{\Phi}$ (recall that $\mathbf{A} = \mathbf{\Phi\Psi}$). In this case, one generally needs to design the measurement matrix $\mathbf{\Phi}$ from $\mathbf{A}$ and $\mathbf{\Psi}$.

It is shown in [9] that if the signal's spectrum vector $\mathbf{x} = \mathbf{\Psi}^{-1}\mathbf{d}$ is sparse ($\mathbf{\Psi}^{-1}$ is the Discrete Fourier Transform), then $\mathbf{\Phi} = \mathbf{A\Psi}$ is essentially an $M \times N$ random sampling matrix constructed by selecting $M$ rows independently and uniformly from an $N \times N$ identity matrix $\mathbf{I}$. This measurement matrix $\mathbf{\Phi}$ can be trivially implemented by pseudo-randomly sub-sampling the original signal $\mathbf{d}$. As we can adopt inverse DFT matrix as the sparse dictionary $\mathbf{\Psi}$, in our framework, the measurement matrix will be reflected by sub-Nyquist sampling. For time domain signals with length $N$, this measurement process corresponds smaller sampling numbers $M < N$, thus compressed sensing enables the source node to transmit sub-sampled/compressed data. As pointed out in Section I, in practical applications sub-sampling (compression) may be very useful for supporting wide-band signals which incurs high cost to transmit the data at the transmitter and has a high cost for ADC at the receiver.

### B. Compressed-Sensing-Based Data Transmission

We will now introduce our CS-based data transmission framework.

Suppose the transmitter has a $N \times 1$ data array $\mathbf{d}$ to transmit (typically time-domain signal), where $\mathbf{d}$ is $K$-sparse in a dictionary $\mathbf{\Psi}$ ($\mathbf{d} = \mathbf{\Psi x}$). In order to perform compressive signal transmissions, instead of transmitting $\mathbf{d}$ ($N \times 1$), the transmitter will send compressed data $\mathbf{\Phi d}$ ($M \times 1$, $M < N$).

The measurement matrix $\mathbf{\Phi}$ and signal dictionary matrix $\mathbf{\Psi}$ can be known to both the sender and receiver. One way is to attach the matrix information with the header of the data packets. Another way is that the transmitter side and receiver side can have some common pseudo-random matrix generating mechanisms, and the transmitter may only need to transmit a pseudo-random seed.

In wireless communications, to recover the transmitted symbols, channel information needs to be estimated at the receiver to enable channel equalization [17] and reduce Inter Symbol Interference (ISI). There are various kinds of equalizers in digital communications, among them Linear Equalizers such as Minimum Mean Square Error (MMSE) equalizer and Zero Forcing Equalizer are commonly used. The performance of channel equalization is typically characterized by Symbol Error Rate (SER).

In the circumstances of low SNR and fast-varying channel, the channel equalization performance may be affected. When channel is not equalized well, inaccuracies will impact the data reconstruction. Due to space limitations we will not talk about how to improve channel equalization but we will investigate how it will affect the data reconstruction at the receiver.

Suppose channel estimation indicates the Channel State Information (CSI) is $\tilde{\mathbf{H}}$, then after channel equalization and demodulation, what is received at the receiver will be

$$\mathbf{y} = \mathbf{H\Phi d} + \mathbf{n} = \tilde{\mathbf{H}}^{-1}\mathbf{HAx} + \mathbf{n} = \mathbf{I_e Ax} + \mathbf{n}, \qquad (10)$$

where the $M \times M$ matrix $\mathbf{H}$ is the channel matrix (Channel State Information, CSI), $M \times 1$ vector $\mathbf{n}$ is an additive noise brought by the channel, $\mathbf{I_e}$ is the identity matrix plus possible error in every entry. In the perfect channel equalization case, $\mathbf{I_e} = \mathbf{I}$.

The actual received $\mathbf{y}$ is subject to noise and channel equalization error. If we simply recover the original data array by solving Equation (9) without considering the effects of channel information inaccuracy, significant errors might occur. This will inevitably harm the quality of data transmission.

### C. Channel Effects on Compressed-Sensing-Based Data Transmission

In our framework, the quality of data transmission can be characterized by an error, which is the difference between the original signal vector $\mathbf{d}$ (recall $\mathbf{d} = \mathbf{\Psi}\mathbf{x}$) at the transmitter and the reconstructed signal vector $\mathbf{\Psi}\hat{\mathbf{x}}$ at the receiver. We call this the Data Transmission Error throughout the paper. The Data Transmission Error is impacted by at least two factors:

*1) Symbol Error Rate:* SER

Before data are sent to the channel, they are modulated and need to be demodulated at the receiver, i.e., the receiver will demodulate the received symbols. If the channel equalization is not well performed, it will cause Inter Symbol Interference and consequently the symbol errors at the receiver.

The performance of channel equalization is typically characterized by Symbol Error Rate (SER), which will impact the accuracy of received samples at the receiver. Since the receiver reconstructs the signal from the received signal samples, it is easy to see that the overall error between the transmitter signal and reconstructed receiving signal at may grow with SER. Our simulation tests in Figure 1 indicate how channel situation influences SER and thus imposes a significant impact on data transmission quality.

In order to reduce the SER effect, we will later develop adaptive compressive sampling schemes that can adapt the sample rate to compbat the channel compensation inaccuracy.

*2) CS Reconstruction Error:* CSRE

Error can be resulted after the channel equalization and during the process of the signal recovery from the sub-sampled data, and we call it CS Reconstruction Error. CS Reconstruction Error is generally impacted by the effectiveness of compressive sensing reconstruction algorithm mentioned in Section III. It is intuitive that even with perfect channel equalization, the overall error between the original signal without sub-sampling and the reconstructed signal at the receiver will grow with CS reconstruction error.

The inaccuracy of channel equalization as a result of the channel dynamics could lead to large noise added to the sub-samples. As CSRE is impacted mostly by the noise in the received samples, in order to reduce CS reconstruction error in the presence of possibly large noise, we will develop an efficient reconstruction schemes that can help reduce the noise effect and achieve better reconstruction quality.

## V. Robust CS-Based Data Transmission

As discussed in previous sections, CS-based data transmission can be perturbed by the dynamics of wireless channels and is more sensitive to the noise. This can be seen more clearly from our simulation results shown in Figure 5 in Section VI, where under the same SNR condition, CS-based schemes (e.g. L1-magic, Cosamp) don't perform as well as the traditional non-CS data transmission. This motivates us to propose error resilient and adaptive schemes to make CS-based data transmission more robust.

We will present our CS-Based Data Transmission methodology in this section. We first discuss how to improve the CS reconstruction performance under large noise, and then how

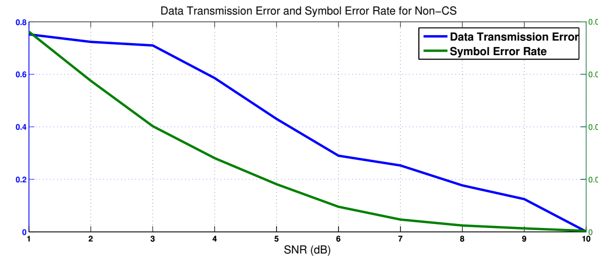


Fig. 1. Non-CS case: Data Transmission Error and Symbol Error Rate under different SNRs.

to combat channel impacts. We first introduce our strategies for improving the CS reconstruction performance under large noise, and then our schemes applied to combat channel impacts and various errors.

### A. Error Resilient CS Reconstruction

CS reconstruction error is impacted mostly by the noise in the received samples. Due to the channel equalization error, this noise will increase. In order to reduce the CS reconstruction error, our reconstruction scheme applies various strategies to reduce the noise effect for better reconstruction quality.

*1) Exploiting Support Feature to Improve Recovery Quality:* The support of a sparse vector $\mathbf{x}$ is defined to be the set of locations where the elements are non-zero. In CS reconstruction algorithms, finding the correct support is very important for signal recovery and ensuring the reconstruction performance. There are many existing works to address the issue of support recovery [20].

Since we take Fourier Transform as the sparse dictionary $\mathbf{\Psi}$, we find that for a time-domain signal with a certain frequency, there are two non-zero elements with symmetric frequency locations in the Fourier domain sparse vector. Generally, for a length-$N$ signal with frequencies, after Fourier Transform, a length-$N$ sparse vector will be obtained. One frequency in the original signal will contribute to two non-zero elements in the sparse vector, and the locations of the two elements in the vector is symmetric, i.e., the sum of the two indices are $N + 1$. Other types of applications may also have special features in their support besides symmetry. These features can be exploited to improve the signal reconstruction performance in the presence of noise.

This can be tested through some simple simulations. In Figure 2, for a signal that contains 3 frequency components, we observe six non-zero elements in the Fourier Basis. In order to improve the CS reconstruction performance, we propose to take advantage of this support information. For other CS applications, it will also help greatly if one can discover the underlying informative characteristics of the supports.

We can easily integrate this feature-based recovery into many CS reconstruction algorithms that involve iterations in the recovery process. In this work, we study the benefit of exploiting support feature in recovery by taking Cosamp [8]

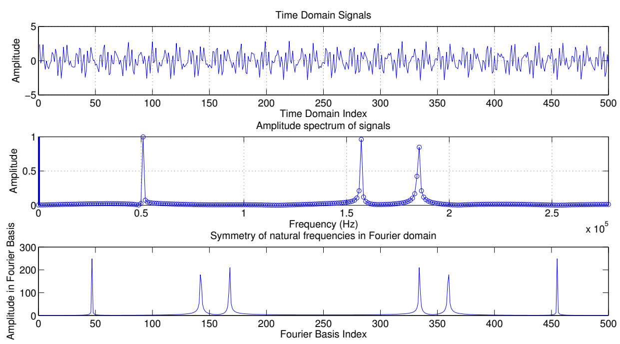


Fig. 2. Symmetry of natural frequencies in Fourier Basis.

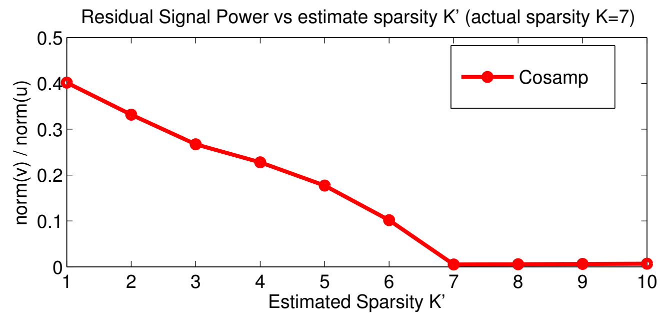


Fig. 3. Residual Signal Power vs estimated sparsity $K'$.

as an example, as it has inherent iteration feature, and we call our algorithm Symmetry-Cosamp. The main idea is briefly described as follows: 1) At the end of every iteration in Symmetry-Cosamp, if there are non-zero elements in symmetric locations in the estimated sparse vector $\hat{\mathbf{x}}$, keep them in the candidate support list; otherwise throw the non-zero elements out. 2) Calculate the residual signal to use in the next iteration.

In order to better illustrate our idea, we present the symmetry Cosamp reconstruction in Algorithm 1. During each iteration, Symmetric-Cosamp performs six major steps:

1) Identificiation. The algorithm forms a proxy of the residual from the current samples and locates the largest $2K$ components of the proxy, denoted as $\Omega$.
2) Support Merger. The set of newly identified components $\Omega$ is united with the set of those appear in the current approximation. Denote the union set as $T$.
3) Estimation. Solve a least-square problem to approximate the target signal on the merged set of components $T$. Denote the estimation as $\mathbf{b}|_T$.
4) Symmetry Check. Check symmetry of components and keep only the symmetric components in the least-squares estimation $\mathbf{b}|_T$.
5) Pruning. Produce a new approximation by retaining only the largest $K$ entries in the updated estimation from last step.
6) Sample Update. Update current samples that reflect the residual signal, the part of signals that are not approximated yet. Go on to the next iteration.

At each iteration, the current approximation introduces a residual, the part of the target signal that has not be approximated yet. As the algorithm continues, samples are updated to reflect the residual signal used to construct a proxy for the residual in order to identify the large components in the residual. This step provides a tentative support for the next approximation. Through iterations in the algorithm, the support information and recovery get more and more accurate.

*2) Estimation of Sparsity :* Existing iterative greedy algorithms for CS reconstruction such as [8] generally require the knowledge of the sparsity level $K$ in the signal. However, in practical applications, the sparsity level may not always be precisely known because the uncertainty of signals. In data transmission scenarios, the sparsity within the signals at the transmitter is even more difficult for the receiver to know. For CS signal recovery, the sparsity level $K$ that is input to run the algorithm will significantly impact the reconstruction result. So the problem is, with unknown or inaccurate $K$, how to ensure the reconstruction performance?

Through studies in Figure 3 based on Cosamp, we find that if the estimated $K'$ input is smaller than the actual $K$, the residual signal which is regarded as noise at the end of the algorithm (for further details of Cosamp please see [8]) will be relatively large and change rapidly with the variation of $K'$; whereas if the estimated $K'$ input to Cosamp is larger than actual $K$, the residual signal is relatively small and changes slowly with the change of $K'$. From this observation, we propose an algorithm to adjust the $K'$ based on the recovery result to further improve the reconstruction quality.

The main idea is briefly described as follows:

1) Input the estimated parameter $K'$ to Symmetry-Cosamp, run Symmetry-Cosamp and obtain residual signal $v'_K$
2) Input the estimated parameter $K'+1$ to Symmetry-Cosamp, run Symmetry-Cosamp and obtain residual signal $v_{K'+1}$.
3) If $\|v_{K'}\|_{l_2} - \|v_{K'+1}\|_{l_2} > T_v$, where $T_v$ is a positive small threshold, update $K' \leftarrow K'+1$.
   Else stop, claim reconstruction completed.

This algorithm is conservative, to make sure $K'$ is large enough for better recovery. Some other variations of this idea may be adjusting $K'$ at a different rate. The adaptation of K will help the algorithm to gradually find a proper estimated $K'$ and thus improve the reconstruction performance.

### *B. Adaptive Measurements*

We mentioned that in wireless communications, the data transmission quality will be impacted by both CS reconstruction and channel compensation error. We have previously presented how to improve CS reconstruction performance. Although it helps to combat channel compensation error, it may not be enough to combat big channel estimation error due to channel perturbations.

**Algorithm 1** Symmetric-Cosamp

**Initialization:**

sparsity level $K$, measurement matrix $\mathbf{\Phi}$, noisy received vector $\mathbf{u}$, residual signal $\mathbf{v}$

$j \leftarrow 0$

$\mathbf{v} \leftarrow \mathbf{u}$

$\mathbf{a}^0 \leftarrow 0$

**Iteration:**

$j \leftarrow j + 1$

$\mathbf{y} \leftarrow \mathbf{\Phi} * \mathbf{v}$ (Form signal proxy)

$\Omega \leftarrow supp(\mathbf{y}_{2K})$ (Identify large compenents)

$T \leftarrow \Omega \bigcup supp(\mathbf{a}^{j-1})$ (Merge supprts)

$\mathbf{b}|_T \leftarrow \mathbf{\Phi}_T^{\dagger}\mathbf{u}$ (Signal estimation)

1: **Repeat in** $T$

If an element in $T$ does not satisfy symmetry, remove it from $T$.

Else if an element in $T$ does satisfy symmetry with another element, keep both elements in $T$.

2: **Continue**

$\mathbf{b}|_{T^c} \leftarrow \mathbf{0}$

$\mathbf{a}^j \leftarrow \mathbf{b}_K$ (Prune to next approximation)

$\mathbf{v} \leftarrow \mathbf{\Phi}\mathbf{a}^j$ (Update current samples)

**Algorithm 2** Dynamic Cosamp (D-Cosamp)

**Initialization:**

estimated sparsity level $K'$, measurement matrix $\mathbf{\Phi}$, noisy received vector $\mathbf{u}$, residual signal $\mathbf{v}_{K'}$, $T_v$ a positive small threshold

**Iteration:**

Input sparsity $K' + 1$ to Symmetry-Cosamp, run Symmetry-Cosamp and obtain residual signal $\mathbf{v}_{K'+1}$.

1: **If** $\|\mathbf{v}_{K'}\|_{l_2} - \|\mathbf{v}_{K'+1}\|_{l_2} > T_v$

$K' \leftarrow K' + 1$

2: **Else**

Stop program. Claim reconstruction completed.

We know from CS theory that adapting the number of measurement $M$ may help improve the CS recovery quality. The problem is how to determine the number of samples that needs to adapt to. On one hand, we want to adjust $M$ to a sufficient large value so that the signal can be recovered well; on the other hand, too large an $M$ will introduce high communication cost and compromise the benefit of using CS to conserve bandwidth.

An intuitive guide for adapting $M$ is the Data Transmission Error. For a certain $M$, if the quality of data transmission error is far worse than satisfactory, then $M$ needs to be increased according to how bad the reconstruction is. However, in practical scenarios, since the receiver doesn't know the original data at the transmitter, it is often impossible for them to know the actual Data Transmission Error. Therefore, we need to consider other schemes that do not rely on information from the transmitter.

In this paper, we propose to adapt measurement number $M$ according to the difference between recovered results. Once the receiver acquires $M$ measurements and recovers $\hat{\mathbf{x}}_M$, it first compares $\hat{\mathbf{x}}_M$ with its last normally recovered result and check if the difference $D$ is within a threshold $T_D$. This is reasonable because many natural signals will not change dramatically in a short time period. A typical example is temperature. Another example is that the sparsity level within a time-varying signal may also not change rapidly. One approach is: consider $\hat{\mathbf{x}}_M$ a successful recovery if the absolute value of $D$ is within $T_D$ and keep $M$ the same; if the absolute value of $D$ is beyond threshold $T_D$, increase $M$ at a ”proper” rate. It remains a problem how to design a ”proper” rate.

In order to ensure stable performances, we propose a control-based $M$ adaptation scheme, whose main idea can be summarized as follows:

1) The receiver acquires $M$ measurements and recovers $\hat{\mathbf{x}}_M$, then it compares $\hat{\mathbf{x}}_M$ with its last normally recovered result.
2) If the absolute value of $D$ is beyond threshold $T_D$, update $M$ as :$M \leftarrow M + \delta\frac{D-D_T}{D_T}$ for a predefined target small difference $D_T$ and step factor $\delta$.
   Otherwise make no change to $M$.

The adaptive idea can be implemented on top of any CS reconstruction algorithm in order to facilitate an comprehensive CS framework. The algorithms using the adaptive measurements with Cosamp and D-Cosamp are shown in Algorithm 3 and Algorithm 4, respectively. The only difference between the two is the reconstruction algorithm.

**Algorithm 3** Cosamp with Adaptive Measurements (A-Cosamp)

**Initialization:**

$M$, $D$, $D_T$, $\delta$, $\hat{\mathbf{x}}_{Mprev}$

**Iteration:**

Recover $\hat{\mathbf{x}}_M$ by Cosamp.

Calculate difference between current reading and last normal recovery result $D = \|\hat{\mathbf{x}}_M - \hat{\mathbf{x}}_{Mprev}\|_{l_1}$.

1: **If** $\|D\|_{l_1} > D_T$

$M \leftarrow M + \delta\frac{D-D_T}{D_T}$

$\hat{\mathbf{x}}_{Mprev} \leftarrow \hat{\mathbf{x}}_M$

2: **Else**

Make no change to $M$

$\hat{\mathbf{x}}_{Mprev} \leftarrow \hat{\mathbf{x}}_M$

## VI. Simulations and Results

In this section we will present our simulations in MATLAB and show that our scheme is robust in wireless data transmission.

**Algorithm 4** Dynamic Cosamp with Adaptive Measurments (AD-Cosamp)

**Initialization:**

$M$, $D$, $D_T$, $\delta$, $\hat{\mathbf{x}}_{Mprev}$

**Iteration:**

Recover $\hat{\mathbf{x}}_M$ by D-Cosamp.

Calculate difference between current reading and last normal recovery result $D = \|\hat{\mathbf{x}}_M - \hat{\mathbf{x}}_{Mprev}\|_{l_1}$.

1: **If** $\|D\|_{l_1} > D_T$

$M \leftarrow M + \delta \frac{D - D_T}{D_T}$

$\hat{\mathbf{x}}_{Mprev} \leftarrow \hat{\mathbf{x}}_M$

2: **Else**

Make no change to $M$

$\hat{\mathbf{x}}_{Mprev} \leftarrow \hat{\mathbf{x}}_M$

### A. Simulation Settings

We will compare the performances of different schemes, which are listed as follows:

1) L1-magic: A convex programming reconstruction algorithm in [7].
2) Cosamp: A greedy Matching Pursuit algorithm proposed in [8].
3) D-Cosamp: Dynamic Cosamp proposed in Section V Algorithm 2.
4) A-Cosamp: Cosamp with Adaptive Measurements proposed in Section V Algorithm 3.
5) AD-Cosamp: Dynamic Cosamp with Adaptive Measurements proposed in Section V Algorithm 4.
6) Non-CS: Traditional approach that allows the transmitter transmit all the data without compression measurements.

In our simulations, the original signal is simulated to be sampled from aggregated sine-waves with different frequencies.

We fixed the number of original data samples to be $M = 1000$. We simulate 100 time frames, each consisting of 100 periods. The signal sparsity level $K$ is set to range uniformly from 10 to 50. In each period, one CS reconstruction is performed. For each time frame, we keep the sparsity level $K$ the same, and a random value within a small range of $\Delta K$ from the previous frame $K = K_{prev}$, i.e., $K \in [K_{prev} - \Delta K, K_{prev} + \Delta K]$. In our settings, we set $\Delta K$ to 5. In this way, we can make the time-varying signal change at a proper speed and also average the performances.

In order to observe the effectiveness of our techniques, we first test adaptive schemes and find a common average $M$ value and then use this value for all the other non-adaptive schemes. In this way we can make the average compression rate $\frac{M}{N}$ the same for each scheme. And then we will see how the Data Transmission Error changes for each scheme under different SNRs. In adaptive schemes, we initialize the number of measurements $M$ as $M = cK \log\left(\frac{N}{K}\right)$, where $c$ is a fairly small constant and $K$ is an estimate of the sparsity level (typically can be the average value of sparsity level calculated from its range). In our simulation we set $c = 2$, an empirical value from many other references.

The channel is simulated to be multi-path fading. And we use an Minimum Mean Square Error (MMSE) equalizer to perform channel equalization.

We will adopt average Data Transmission Error ($DTE$) as the metric to evaluate the performances of data transmission. DTE in one data transmission is defined as the normalized difference between the original signal at the transmitter and recovered signal at the receiver, which can be expressed as:

$$DTE = \frac{\left\|\hat{\mathbf{d}} - \mathbf{d}\right\|_{l_2}}{\|\mathbf{d}\|_{l_2}} \tag{11}$$

where $\hat{\mathbf{d}}$ is the recovered data and $\mathbf{d}$ is the original data.

We will first show some results of preliminary simulations and then show the performance of our proposed scheme with comparisons with other schemes.

### B. Robust CS-Based Data Transmission

*1) Comparisons of the proposed schemes:* D-Cosamp, A-Cosamp and AD-Cosamp

In Figure 4, we can see that for all schemes, Data Transmission Error decreases with the growth of channel SNR. We can also observe that A-Cosamp usually works better than D-Cosamp, which is reasonable to understand because D-cosamp is usually limited by its capability of handling large noise, whereas increasing the number of measurements can significantly increase the effectiveness of CS. Since AD-Cosamp takes advantage of the two, it outperforms the two, as we can see in the Figure.

In our simulations we find that AD-Cosamp is very effective in combatting noise and channel dynamics. Even when the SNR is low, it can approach the performance of the traditional non-CS case where the transmitters send out all the data. The effectiveness of AD-Cosamp is due to its capability of handling large noise in CS reconstruction and adapting measurements according to the environment. When SNR is low, more measurements may be needed, also with the help of efficient CS reconstruction algorithm, AD-Cosamp can approximate the non-CS case while still maintaining certain compression gain.

CS-based scheme can compress the data to be transmitted over the channel and to be processed by the receiver, and AD-Cosamp is shown to achieve compression gain while maintaining a comparable performance to non-CS schemes. This indicates that CS-based data transmission scheme is a promising candidate to ensure low-overhead transmission.

*2) Comparisons with other schemes:* L1-magic, Cosamp and AD-Cosamp

In Figure 5, we compare our proposed schemes with other CS techniques to see the effectiveness of our schemes. We can

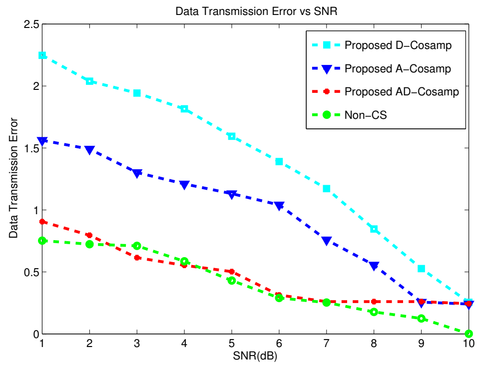


Fig. 4. Comparisons of proposed schemes.

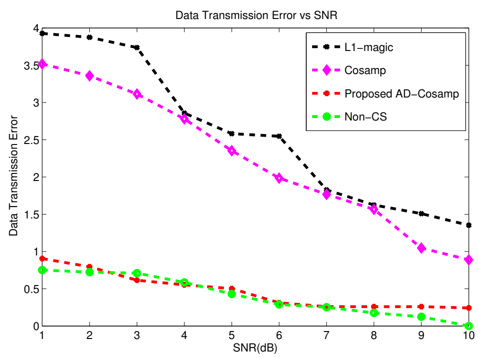


Fig. 5. Comparisons with other schemes.

see from the Figure that among all CS-based schemes, AD-Cosamp works the best, because it takes advantage of both A-Cosamp and D-Cosamp.

Compared with L1-magic and Cosamp, we improve from two aspects: one is to handle larger noise in CS reconstruction by exploiting information from sparse basis and estimating sparsity level more accurately; another is to adapt the number of measurements according to the impacts of the environment to combat the channel errors.

## VII. Conclusions

In this paper we investigate the possibility and methodologies to enable robust transmission of compressive wireless data in the presence of channel dynamics and large noise. Different from literature work which normally apply CS at the receiver, the transmitter exploits CS to sub-sample the data to reduce the information redundancy thus transmission load and bandwidth consumption before sending the data over the communication channel, and the receiver will reconstruct the signals from the received data. With less information redundancy, CS-based data transmission is more sensitive to the environment dynamics. We observe the significant impacts brought by channel perturbations in wireless communications. We propose various strategies to ensure higher quality signal reconstruction under large noise which is partly contributed by the channel equalization error, and adaptive compressed sampling techniques to combat the accumulative signal recovery errors due to channel dynamics, channel equalization inaccuracy, and reconstruction errors. Our performance study demonstrate the efficiency of our design in supporting robust compressive data transmission. We hope our work can serve as a base for future research in enabling practical compressive wireless network communications.

## References


[1] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol.52, no.4, pp.1289-1306, April 2006.

[2] E. J. Candès, "Compressive sampling," *Proc. International Congress of Mathematicians*, Madrid, Spain, 2006.

[3] C. R. Berger; Z. Wang; J. Huang and S. Zhou, "Application of compressive sensing to sparse channel estimation," *IEEE Communications Magazine*, vol.48, no.11, pp.164-174, November 2010.

[4] W. U. Bajwa; J. Haupt; A.M. Sayeed; R. Nowak, "Compressed Channel Sensing: A New Approach to Estimating Sparse Multipath Channels," *Proc. of the IEEE*, vol.98, no.6, pp.1058-1076, June 2010.

[5] S. Feizi and M. Medard, "A power efficient sensing/communication scheme: Joint source-channel-network coding by using compressive sensing," *Proc. Allerton Conf. on Communication, Control, and Computing*, pp. 1048-1054, 2011.

[6] C. Luo, F. Wu, J. Sun, and C. W. Chen, "Compressive data gathering for large-scale wireless sensor networks," *Proc. ACM Mobicom09*, pp.145-156, Sep. 2009.

[7] E. J. Candès, and J. Romberg, "$\ell_1$ -MAGIC: Recovery of sparse signals via convex programming", 2005.

[8] D. Needell and J. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301-321, May 2009.

[9] Z. M. Charbiwala, S. Chakraborty, S. Zahedi , Y. Kim, T. He , C. Bisdikian and M. B. Srivastav "Compressive oversampling for robust data transmission in sensor networks", *Proc. the 29th IEEE INFOCOM*, pp.1-9, 2010.

[10] S. Kadhe, S. Thaskani, M. G. Chandra and B. S. Adiga, "Reliable data transmission in sensor networks using compressive sensing and real expander codes," *Proc. National Conference on Communications (NCC)*, pp.1,5, Feb. 2012.

[11] S. Pudlewski and T. Melodia, "On the Performance of Compressive Video Streaming for Wireless Multimedia Sensor Networks", *Proc. IEEE International Conference on Communications (ICC)*, 2010.

[12] E. J. Candès , J. Romberg and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements", *Comm. Pure Appl. Math.*, vol. 59, no. 8, pp.1207-1223, 2006.

[13] R. Tibshirani. "Regression shrinkage and selection via the lasso, *Journal Royal Statistical Society B*, vol. 58, pp. 267-288, 1996.

[14] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal of Scientic Computation*, vol. 20, pp. 33-61, 1998.

[15] E. J. Candès and T. Tao, "The Dantzig selector: Statistical estimation when $p$ is much larger than $n$," *Annals Of Statistics*, vol. 35, pp. 2313, 2007.

[16] T. Blumensath and M. E. Davies, "Iterative hard thresholding for compressed sensing, *Appl. Comput. Harmon. Anal.*, vol. 27, no. 3, pp. 265-274, 2009.

[17] B. Widrow, M. A. Lehr, "Noise canceling and channel equalization," *Conf. Rec. IEEE Int. Conf. Communications*, pp. 648-650, 1995.

[18] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 2130, Mar. 2008.

[19] H. Rauhut, "Compressive sensing and structured random matrices, " *Radon Series Comp. Appl. Math. 9, de Gruyter,* pp.1-92, 2012.

[20] Z. Weng and X. Wang, "Support recovery in compressive sensing for estimation of direction-of-arrival." *Conf. Rec. 45th IEEE Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*, pp.1491-1495, 2011.