Chapter 2
Distributed Compressed Sensing

This chapter first introduces CS in the conventional setting where one device acquires one signal and sends it to a receiver, and then extends it to the distributed framework in which multiple devices acquire multiple signals. In particular, we focus on two key problems related to the distributed setting. The former is the definition of sparsity models for an ensemble of signals, as opposed to just one signal. The second is the structure of the corresponding recovery algorithm, which can be centralized or distributed; each solution entails specific advantages and drawbacks that are preliminarily discussed in this chapter, whereas a detailed description of the corresponding recovery algorithms is given in Chaps. 4 and 5.

2.1 Compressed Sensing for Single Sources

Before starting, we define some notations. We denote column vectors with small letters, and matrices with capital letters. Given a matrix A, $A^\top$ denotes its transpose. We consider $\mathbb{R}^n$ as an Euclidean space endowed with the following norms:

$$\|x\|_p = \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p}$$

with $p = 1, 2$. Given $x \in \mathbb{R}^n$, we denote the ℓ_0 pseudo-norm as

$$\|x\|_0 = \sum_{i=1}^{n} |x_i|^0,$$

where we use the convention $0^0 = 0$. For a rectangular matrix $M \in \mathbb{R}^{m \times n}$, we consider the Frobenius norm, which is defined as follows:

G. Coluccia et al., *Compressed Sensing for Distributed Systems*,
SpringerBriefs in Signal Processing, DOI 10.1007/978-981-287-390-3_2

$$\|M\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} M_{ij}^2}$$

and the operator norm

$$\|M\|_2 = \sup_{z \neq 0} \frac{\|Mz\|_2}{\|z\|_2}.$$

We denote the sign function as

$$\operatorname{sgn}(x) = \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{if } x = 0 \\ -1 & \text{otherwise.} \end{cases}$$

If x is a vector in $\mathbb{R}^n$, $\operatorname{sgn}(x)$ is intended as a function to be applied elementwise.

2.1.1 Sensing Model

The basic CS problem can be stated as follows. An unknown signal represented by a column vector $x \in \mathbb{R}^n$ is sensed by taking a number m of its linear projections, i.e.,

$$y = \Phi x, \tag{2.1}$$

where $\Phi \in \mathbb{R}^{m \times n}$ is a given sensing matrix, and $y \in \mathbb{R}^m$, with $m < n$, is the measurements vector. According to this process, x is repeatedly sensed by taking its scalar product with every row of Φ, yielding a measurements vector y. Since $m < n$, the signal representation provided by y is more "compact" than the original representation x, hence the term *compressed* sensing.

The CS reconstruction problem can be stated as follows: given y and Φ, one wishes to reconstruct the original signal x. With $m < n$, this is clearly an underdetermined problem that may have infinitely many valid solutions satisfying $y = \Phi x$. As is the case of most regularization problems, in order to recover x, it is therefore necessary to add some prior knowledge about the signal in order to constrain the solution set of (2.1). In CS, this is done using the concept of sparsity. In plain terms, a signal is said to be sparse if it has a low number of nonzero entries with respect to its length. In particular, x is said to be k-sparse if it has at most k nonzero entries, or equivalently $\|x\|_0 \leq k$. The set of all k-sparse signals in $\mathbb{R}^n$ is denoted as Σ_k, i.e., $\Sigma_k = \{x \in \mathbb{R}^n : \|x\|_0 \leq k\}$.

2.1.2 Sparse Recovery

Armed with the notion of sparsity, one can attempt to recover x solving the following problem, which aims at finding the sparsest possible signal that satisfies the sensing equation (2.1):

$$\hat{x} = \arg\min_{x} \|x\|_0 \quad \text{such that} \quad y = \Phi x. \tag{2.2}$$

Problem (2.2) has very strong guarantees of success. In [1, Chap. 1] it is shown that, under some mild conditions on the independence of the columns of Φ, if $m \geq 2k$, i.e., the number of linear measurements is at least twice as the sparsity of x, then there exists at most one signal $x \in \Sigma_k$ such that $y = \Phi x$, and (2.2) will yield the correct solution for any $x \in \Sigma_k$. However, despite its attractiveness, solving (2.2) is not a viable way to recover x, because this problem has combinatorial complexity. The so-called "oracle" receiver simply assumes to know in advance the set $\mathscr{S} = \text{supp}(x) = \{i \in \{1, \ldots, n\} | x_i \neq 0\}$ identifying the indexes of the nonzero entries of x. Given $\mathscr{S}$, one can construct the reduced matrix $\Phi_{\mathscr{S}}$ which is obtained removing from Φ the columns ϕ_i whose index does not belong to $\mathscr{S}$. Then, the nonzero components of x are readily obtained as $x_{\mathscr{S}} = \Phi_{\mathscr{S}}^{\dagger} y$, where $A^{\dagger}$ denotes the pseudoinverse of A, i.e., $A^{\dagger} = (A^{\top} A)^{-1} A^{\top}$. The oracle receiver is very useful to derive theoretical properties of CS systems. In practice, however, $\mathscr{S}$ is not known, so that in order to solve (2.2) one should consider all possible sets of k out of n index positions, i.e., the sparsity supports of x, and test each of them for correctness. This is an NP-hard problem that is computationally infeasible even for small values of n.

To address this issue, it is possible to solve a slightly different problem where the function to be minimized is convex, namely

$$\hat{x} = \arg\min_{x} \|x\|_1 \quad \text{such that} \quad y = \Phi x. \tag{2.3}$$

Replacing the ℓ_0 with the ℓ_1 norm makes the problem convex, and essentially reduces it to a linear program that can be solved using convex optimization tools, e.g., quadratic programming [2], such as interior-point methods.
The complexity is polynomial and depends on the specific solver employed for (2.3), e.g., $O(n^3)$. The algorithm in (2.3), also known as basis pursuit (BP), has very good performance in terms of success of reconstruction, since the ℓ_1 norm tends to promote a sparse solution. Moreover, it also has some interesting performance guarantees, which are easily described via the Restricted Isometry Property (RIP, see e.g., [3]). In particular, a matrix Φ satisfies the RIP of order k if there exists $\delta \in [0, 1)$ such that the following relation holds for all $x \in \Sigma_k$:

$$(1 - \delta)\|x\|_2^2 \leq \|\Phi x\|_2^2 \leq (1 + \delta)\|x\|_2^2. \tag{2.4}$$

We define the RIP-constant $\delta_k := \inf\{\delta \in [0, 1)$: Φ satisfies the RIP of order $k\}$. Basically, the RIP ensures that the columns of Φ are nearly orthonormal, at least when operating on sparse vectors. Moreover, Φ is an approximately norm-preserving (and hence distance-preserving) mapping for k-sparse signals, all the more so as δ_k approaches zero. It can be shown (see [4]) that, if x is k-sparse and Φ satisfies the RIP of order $2k$ with RIP-constant $\delta_{2k} < \sqrt{2} - 1$, then the solution to (2.3) is the same as the solution to (2.2).

Verifying whether a given matrix Φ satisfies the RIP of order k is also an NP-hard problem. However, it has been shown [5] that some classes of random matrices with $m = O(k \log \frac{n}{k})$, and particularly those with independent and identically distributed (i.i.d.) entries drawn from a sub-Gaussian distribution, satisfy the RIP with very high probability. This is the main reason behind the popularity of random sensing matrices with Gaussian, Bernoulli or Rademacher distributions, so that it is also common to refer to linear measurements as "random projections."

In practice, in most cases the sparsity model describes well the signal in a transformed domain, rather than in its natural domain. The formulation above can be easily modified to accommodate this. In particular, we let $x = \Psi\theta$, with $\Psi \in \mathbb{R}^{n\times n}$. Matrix Ψ represents the linear inverse transform of a representation θ that is indeed sparse. Putting this definition into (2.1) yields $y = \Phi x = \Phi\Psi\theta = A\theta$, where $A = \Phi\Psi$ and θ is a sparse vector. This new problem can be solved in exactly the same way as the original one, solving for θ and considering A as the new "sensing" matrix, and eventually recovering $x = \Psi\theta$.

Another limitation of the sensing model (2.1) lies in the fact that the acquired linear measurements are typically affected by noise, leading to the following more accurate model:

$$y = \Phi x + e, \tag{2.5}$$

where e is some unknown perturbation bounded by $\|e\|_2 \leq \varepsilon$.

Under certain assumptions on the sensing matrix and for a sufficiently low level of the signal sparsity [6], *robust signal recovery* is achieved by solving

$$\hat{x} = \arg\min_x \|x\|_0 \quad \text{s.t.} \quad \|y - \Phi x\|_2 \leq \varepsilon. \tag{2.6}$$

This means that the solution $\hat{x}$ of (2.6) obeys $\|\widehat{x} - x\| \leq \kappa\varepsilon$ where κ is a positive constant. Alternatively, an estimation can be provided by the following estimator

$$\hat{x} = \arg\min_x \|y - \Phi x\|_2^2 \quad \text{s.t.} \quad x \in \Sigma_k\ . \tag{2.7}$$

As in the noise-free scenario, (2.6) and (2.7) are known to be NP-hard problems. However, an attractive alternative is given by considering (2.7) and taking the convex relaxation of the ℓ_0 pseudonorm. This problem is also known as basis pursuit with denoise (BPDN) and consists in selecting the element x with residual norm below the tolerance ε which has minimal ℓ_1-norm:

$$\hat{x} = \arg\min_{x} \|x\|_1 \quad \text{s.t.} \quad \|y - \Phi x\|_2 \leq \varepsilon, \tag{2.8}$$

It can be shown (see [4]) that, if x is k-sparse and Φ satisfies the RIP of order $2k$ with RIP-constant $\delta_{2k} < \sqrt{2} - 1$, then the solution to (2.8) is such that $\|\widehat{x} - x\| \leq c\varepsilon$ where c is a positive constant.

Another take on the same problem is the least-absolute shrinkage and selection operator (Lasso) [7], which recasts the problem (2.8) using an unconstrained formulation:

$$\hat{x} = \arg\min_{x} \frac{1}{2} \left[\|y - \Phi x\|_2^2 + 2\lambda \|x\|_1 \right], \tag{2.9}$$

where the parameter $\lambda > 0$ weights the sparsity term of the cost function. It is well known that for problems in which the number of variables n exceeds the number of observations m the cost function in (2.9) is not strictly convex, and hence it may not have a unique minimum. Sufficient conditions guaranteeing the uniqueness of the solution of (2.9) are derived in [8]. The solution x_{Lasso} provides an approximation of (2.6) with a bounded error, which is controlled by λ (see [9, 10]).

2.1.3 Iterative Thresholding Algorithms

Despite their provably good performance, however, solvers of BPDN and Lasso problems have a rather high computational complexity, which may be excessive for certain applications, especially when the signal length n is large. Therefore, alongside these solvers, the literature describes a large number of algorithms for sparse recovery. The main approaches can be classified as optimization-based methods [11], pursuit strategies [12–15], coding-theoretic tools [16, 17], and Bayesian methods (see [18] and reference therein).

For example, the orthogonal matching pursuit algorithm [13] is a very popular solution; this algorithm attempts to estimate the k nonzero components one by one, starting from the strongest one. At the same time, iterative hard thresholding (IHT) [19, 20], and iterative soft thresholding (IST) algorithms [21] have been proposed. These algorithms have lower computational complexity per iteration and lower storage requirements than interior-point methods, and are amenable to a theoretical performance analysis. We briefly review IHT and IST, which will also be employed in later sections of this book. They approximate the solution to (2.6) and (2.9).

The solution is obtained through an iterative technique that alternatively applies a gradient descent step to minimize $\|y - \Phi x\|_2^2$, followed by a scalar thresholding step that enforces sparse estimations. The procedure is iterated until a stopping criterion is met. Generally, if the algorithm is analytically proved to converge, one can stop it when numerical convergence is achieved, that is, when the relative distance between the estimates of two successive iterates is below a fixed threshold. Alternatively, one can fix a maximum number of iterations.

The thresholding operators can be hard or soft and are defined as follows:

$$\sigma_k[x] = \underset{z \in \Sigma_k}{\operatorname{argmin}} \|x - z\|_2 \tag{2.10}$$

and

$$\eta_\lambda[x] = \begin{cases} \operatorname{sgn}(x)(|x| - \lambda) & \text{if } |x| > \lambda \\ 0 & \text{otherwise.} \end{cases} \tag{2.11}$$

It should be noticed that the hard thresholding operator takes a best-k term approximation for some value of k. This is equivalent to selecting a certain number of components that have the largest magnitude and sets the remaining ones to zero at each iteration. The soft thresholding operator is also called the "proximity operator" of the ℓ_1-norm, and it acts component-wise by taking the components with magnitude above a certain threshold and shrinks remaining ones. IHT and IST are described in Algorithm 1 and in Algorithm 2, respectively.

Algorithm 1 IHT

Input: Sensing matrix Φ, measurement y, sparsity level k
  Set $x^{(0)} = 0$, iterate
  for $t = 1$ to StopIter **do**
    $x^{(t)} \leftarrow \sigma_k[x^{(t-1)} + \Phi^T(y - \Phi x^{(t-1)})]$
  end for

Algorithm 2 IST

Input: Sensing matrix Φ, measurement y, sparsity parameter λ
  Set $x^{(0)} = 0$, iterate
  for $t = 1$ to StopIter **do**
    $x^{(t)} \leftarrow \eta_\lambda[x^{(t-1)} + \Phi^T(y - \Phi x^{(t-1)})]$
  end for

The convergence of these algorithms was proved under the assumption that $\|\Phi\|_2^2 < 1$ in [21] (for ISTA) and [20] (for IHTA). A dissertation about the convergence results can be found in [11].

2.2 Compressed Sensing for Distributed Systems

The problem addressed in the previous section refers to the sensing and reconstruction process of a single signal. In many cases, however, it is of interest to consider a set of signals sensed by independent nodes. The sensing setup can be extended accordingly, and suitable recovery algorithms can be derived.

2.2.1 Distributed Setup

In the following, we refer to a scenario where CS is applied to a distributed system in which several nodes independently sense a signal according to (2.5), as in Fig. 2.1, where each black dot represents a node and dashed lines represent available communication links between pairs of nodes. Let $\mathscr{V}$ be the set of sensor nodes of size $J = |\mathscr{V}|$. The vth node acquires a signal $x_v \in \mathbb{R}^n$ via linear measurements of the form $y_v = \Phi_v x_v + e_v$. According to this model, each node senses a different signal x_v, using an individual sensing matrix Φ_v, the sensing process being affected by an individual noise signal e_v. Nodes are represented as vertexes of a graph $\mathscr{G}$ where the set $\mathscr{E}$ of edges between pairs of nodes identify the communications links, so that $(\mathscr{G}, \mathscr{V}, \mathscr{E})$ defines the network structure.

The distributed setup poses particular challenges for the sensing and reconstruction process.

- If the signals x_v are independent, then the reconstruction problem is essentially equivalent to J individual problems, one at each node. In other terms, each node will have to acquire a number of linear measurements that is sufficiently large to enable reconstruction of y_v from x_v and Φ_v. No collaboration among nodes is needed or useful at all, because a node does not have any information that may help in reconstructing the signal measured by another node.
- The more interesting case is when the signals x_v are correlated among each other. This typically occurs when the nodes sense a physical phenomenon that exhibits a spatially smooth behavior, e.g., a temperature or pressure field. In this scenario, there are two types of sparsity to be exploited. Namely, intranode sparsity describes the degree of dependency among samples of the same signal at different time instants, while internode sparsity describes the dependency among samples acquired by different nodes at the same time instant. Internode sparsity is a specific

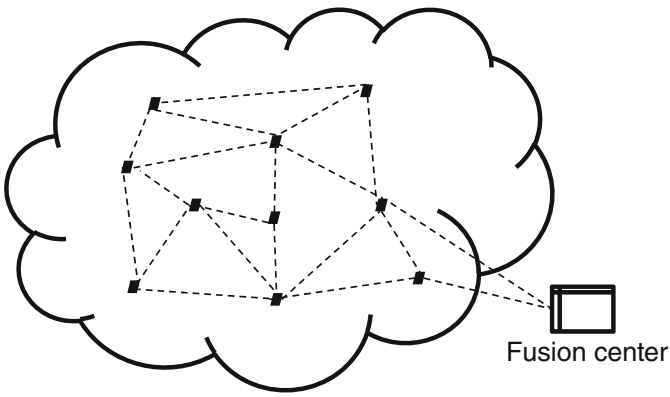

Fig. 2.1 Distributed system setting

aspect of distributed systems, which will be addressed in the remaining part of this chapter and in the next chapters.

- While the nodes individually acquire linear measurements of the signals, two approaches are available for the reconstruction, namely *centralized* and *distributed*. In the centralized approach, the nodes offload the reconstruction process to a fusion center, which receives the linear measurements $\{y_v\}_{v=1}^{J}$ acquired by all nodes and recovers the corresponding set of signals $\{x_v\}_{v=1}^{J}$. Whether centralized reconstruction is possible or even useful depends on many aspects, including the energy cost incurred by transmitting the measurements to the fusion center; this latter also depends on several factors, including distance and the existence and willingness of neighboring nodes to serve as relays. Moreover, the nodes do not get to know the reconstructed signal x_v, which is known only at the fusion center, unless it is transmitted back to each node.
- In the distributed approach, the network does away with a fusion center. This has many advantages, as it avoids the need to transfer all the data to the fusion center, thereby saving a lot of energy. Moreover, the network can function even in the case that the fusion center suddenly becomes unavailable, e.g., because of a hardware failure. In addition, the information does not travel long distances, avoiding the danger of eavesdropping or other security threats. Conversely, distributed reconstruction is based on *local* short-range communications of each node with its neighbors. Short-range communications are very convenient in terms of privacy and energy consumption, and the failure of few nodes will generally not break down the operation of the whole network. On the other hand, local communications decrease the speed at which the information spreads through the network, calling for iterative reconstruction techniques to allow time for each node to contribute to the reconstruction of all other nodes in the network. Since each iteration has an energy cost, the design of a distributed reconstruction algorithm must be done carefully, in order to avoid that too many iterations outweigh the energy benefits of local communications.

2.3 Joint Sparsity Models

This section extends the signal model for one source to the case of distributed signals. Several joint sparsity models are considered, in which all or parts of the signal model are assumed to be sparse. Joint sparsity models entail signals having a twofold source of correlation:

- Intracorrelation, denoting how samples of the same signal are correlated to each other (typically, correlation in time).
- Intercorrelation, denoting how samples of different signals are correlated to each other at the same time instant (correlation in space).

These models represent a good fit for physical signals acquired during time by a sensor network in different points of space. In particular, the original signal sensed by the vth node can be written as

$$x_v = x_{\mathsf{C}} + x_{\mathsf{I},v} \,, \tag{2.12}$$

and node v acquires linear projections of this signal as

$$y_v = \Phi_v x_v = \Phi_v \left(x_{\mathsf{C}} + x_{\mathsf{I},v} \right) \,. \tag{2.13}$$

According to this model, each signal is composed of a common part x_{C}, which is the same for all sensors and is referred to as the *common* component, and an individual part $x_{\mathsf{I},v}$, called *innovation* component, which is specific to each individual sensor. We also assume that x_{C} has k_{C} nonzero entries, and each $x_{\mathsf{I},v}$ has $k_{\mathsf{I},v}$ nonzero entries, so that x_v has at most $k_v = k_{\mathsf{C}} + k_{\mathsf{I},v}$ nonzero entries. If each node had to recover x_v from y_v without receiving help from other nodes, it should acquire a number of linear measurements m_v proportionally larger than k_v, i.e., $m_v \simeq C k_v$, with C sufficiently large depending on the recovery algorithm employed. If each node acted like this, the total number of measurements acquired by the network would be equal to

$$\sum_v C k_v = C \sum_v \left(k_{\mathsf{C}} + k_{\mathsf{I},v} \right) = C \left(J k_{\mathsf{C}} + \sum_v k_{\mathsf{I},v} \right) .$$

What is clear from this analysis is that, while intrasensor sparsity is exploited at each node, the common component is measured J times individually. This is a clear waste of resource that is caused by the lack of exploitation of intersensor sparsity.

The model described in (2.12) can be further detailed according to the structure of x_{C} and $x_{\mathsf{I},v}$. In particular, in [22, 23] the following cases are identified:

- Both the common and innovation components are sparse, namely $k_{\mathsf{C}} \ll n$ and $k_{\mathsf{I},v} \ll n$ for all v. This model is also referred to as JSM-1 in [23]. The common and innovation components need not necessarily be sparse in the same basis. This is a very general model that encompasses many cases of practical interest. For example, a physical phenomenon that is spatially smooth over the coverage area of the sensor network will typically yield a sparse common component that represents the large-scale behavior of the phenomenon, and an innovation component that accounts for the local behavior.
- The common and innovation components have the same sparsity support for all signals (model JSM-2 in [23]). This model is relevant e.g., when a signal described by few frequency components undergoes a frequency-selective attenuation such as the multipath effect, whereby each component is multiplied by a different coefficient, but no new component is created.

- The common component is not sparse, but the innovation components $x_{\mathrm{I},v}$ are sparse (model JSM-3). This is a more general version of JSM-1.

Similar sparsity models were proposed in [24], along with respective reconstruction algorithms.

2.4 Reconstruction for Distributed Systems

It should be noted that, given the number of possible setups, there is no one-size-fits-all reconstruction algorithm that can be used in all cases. In the next chapters we will overview a few algorithms which have been developed for specific scenarios, i.e., centralized (Chap. 4) versus distributed (Chap. 5) reconstruction for a specific joint sparsity model.

The centralized problem entails that all linear measurements $\{y_v\}_{v=1}^{J}$ are available at the fusion center, which attempts to take advantage of the correlation among the signals x_v. Early work on this reconstruction problem goes back to the multiple-measurement vectors problem [25, 26]. Indeed, [25] showed that the availability of multiple measurements improves recovery performance. Later, [26] extended the equivalence between ℓ_0 and ℓ_1 reconstruction to the multiple-measurement vectors case. A few practical algorithms have also been proposed, e.g., M-FOCUSS and M-OMP. Typically, these algorithms are based on a sensing model $\tilde{Y} = \tilde{\Phi}\tilde{X}$, with $\tilde{Y} = [y_1, \ldots, y_v] \in \mathbb{R}^{m\times|\mathcal{V}|}$, $\tilde{X} = [x_1, \ldots, x_v] \in \mathbb{R}^{n\times|\mathcal{V}|}$, and $\tilde{\Phi} \in \mathbb{R}^{m\times n}$. Such algorithms are convenient extensions of the corresponding algorithm in the single-sensor case. However, they have significant limitations. First, the dictionary $\tilde{\Phi}$ must be the same for all vectors. Second, they work well when all signals x_v have the *same sparsity support*, e.g., in the case of model JSM-2. This is because a common sparsity support leads to an unknown vector $\tilde{X}$ that is row-sparse, facilitating the recovery task as well as the derivation of theoretical recovery guarantees. In Chap. 4 we will show more general application scenarios.

The distributed reconstruction problem is more challenging, because at any stage no node has a complete knowledge of the measurements sensed by all other nodes. This information spreads in the network over time, and nodes make greedy decisions based on limited knowledge of the information circulating in the network. Distributed reconstruction algorithms raise the following questions.

- Given the design of a specific distributed recovery algorithm, does the algorithm converge at all?
- If it does converge, does it converge to the global or local minimum of some given cost function?
- Is this functional a sensible one, e.g., the same functional solved by a corresponding centralized reconstruction algorithm?
- Do all nodes individually converge to a sensible solution?

As will be seen in Chap. 5, and as is the case of many single-sensor recovery algorithms, these questions can be answered for some classes of algorithms, but

sometimes only partial responses can be obtained. In particular, Chap. 5 will describe a distributed generalization of thresholding algorithms, for which strong guarantees can be obtained, and some extensions aimed at minimizing communication cost, which are very interesting from the practical standpoint, but less amenable to a complete analytical characterization.

References

1. Eldar, Y.C., Kutyniok, G. (eds.): Compressed sensing: theory and applications. Cambridge University Press, Cambridge (2012)
2. Boyd, S., Vandenberghe, L.: Convex optimization. Cambridge University Press, New York (2004)
3. Candès, E.J., Tao, T.: Decoding by linear programming. IEEE Trans. Inf. Theor. **51**(12), 4203–4215 (2005)
4. Candès, E.J., Romberg, J.K., Tao, T.: Stable signal recovery from incomplete and inaccurate measurements. Commun. Pure Appl. Math. **59**(8), 1207–1223 (2006)
5. Devore, R., Petrova, G., Wojtaszczyk, P.: Instance-optimality in probability with an l1-minimization decoder. Appl. Comput. Harmon. Anal **27**(3), 275–288 (2009)
6. Donoho, D.L., Elad, M., Temlyakov, V.N.: Stable recovery of sparse overcomplete representations in the presence of noise. IEEE Trans. Inf. Theor. **52**(1), 6–18 (2006)
7. Hastie, T., Tibshirani, R., Friedman, J., Hastie, T., Friedman, J., Tibshirani, R.: The elements of statistical learning, vol. 2. Springer, Heidelberg (2009)
8. Tibshirani, R.J.: The Lasso problem and uniqueness. Electron. J. Stat. **7**, 1456–1490 (2013)
9. Candès, E.J.: The restricted isometry property and its implications for compressed sensing. Compte Rendus de l'Academie des Sciences. Paris, France, ser. I, vol. 346, pp. 589–592 (2008)
10. Donoho, D.L.: Compressed sensing. IEEE Trans. Inf. Theor. **52**(4), 1289–1306 (2006)
11. Fornasier, M.: Theoretical foundations and numerical methods for sparse recovery. Radon Series on Computational and Applied Mathematics (2010)
12. Tropp, J.A., Gilbert, A.C., Strauss, M.J.: Algorithms for simultaneous sparse approximation: part i: Greedy pursuit. Signal Process. **86**(3), 572–588 (2006)
13. Tropp, J.A., Gilbert, A.C.: Signal recovery from random measurements via orthogonal matching pursuit. IEEE Trans. Inf. Theor. **53**, 4655–4666 (2007)
14. Needell, D., Vershynin, R.: Signal recovery from incomplete and inaccurate measurements via regularized orthogonal matching pursuit. IEEE J. Sel. Top. Signal Process. **4**(2), 310–316 (2010)
15. Needell, D., Tropp, J.A.: CoSaMP: iterative signal recovery from incomplete and inaccurate samples. Appl. Comput. Harmonic Anal. **26**(3), 301–321 (2008)
16. Baron, D., Sarvotham, S., Baraniuk, R.: Bayesian compressive sensing via belief propagation. IEEE Trans. Signal Process. **58**(1), 269–280 (2010)
17. Jafarpour, S., Xu, W., Hassibi, B., Calderbank, R.: Efficient and robust compressed sensing using optimized expander graphs. IEEE Trans. Inf. Theor. **55**(9), 4299–4308 (2009)
18. Ji, S., Xue, Y., Carin, L.: Bayesian compressive sensing. IEEE Trans. Signal Process. **56**(6), 2346–2356 (2008)
19. Blumensath, T., Davies, M.E.: Iterative thresholding for sparse approximations. J. Fourier Anal. Appl. **14**(5), 629–654 (2004)
20. Blumensath, T., Davies, M.E.: Iterative hard thresholding for compressed sensing. Appl. Comput. Harmonic Anal. **27**(3), 265–274 (2009)
21. Daubechies, I., Defrise, M., De Mol, C.: An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. Comm. Pure Appl. Math. **57**(11), 1413–1457 (2004)

22. Duarte, M.F., Wakin, M.B., Baron, D., Sarvotham, S., Baraniuk, R.G.: Measurement bounds for sparse signal ensembles via graphical models. IEEE Trans. Inf. Theor. **59**(7), 4280–4289 (2013)
23. Baron, D., Duarte, M.F., Wakin, M.B., Sarvotham, S., Baraniuk, R.G.: Distributed compressive sensing. arXiv preprint arXiv:0901.3403 (2009)
24. Hormati, A., Vetterli, M.: Distributed compressed sensing: sparsity models and reconstruction algorithms using annihilating filter. In: IEEE International Conference on Acoustics, Speech and Signal Processing, 2008, ICASSP 2008, pp. 5141–5144 (2008)
25. Cotter, S., Rao, B., Engan, K., Kreutz-Delgado, K.: Sparse solutions to linear inverse problems with multiple measurement vectors. IEEE Trans. Signal Process. **53**(7), 2477–2488 (2005)
26. Chen, J., Huo, X.: Theoretical results on sparse representations of multiple-measurement vectors. IEEE Trans. Signal Process. **54**(12), 4634–4643 (2006)

Springer

http://www.springer.com/978-981-287-389-7

Compressed Sensing for Distributed Systems

Coluccia, G.; Ravazzi, C.; Magli, E.

2015, X, 97 p. 23 illus., 14 illus. in color., Softcover

ISBN: 978-981-287-389-7