Convex Combination of Compressed Sensing Algorithms

Ketan Atul Bapat[1] and Mrityunjoy Chakraborty[2]
Dept. of Electronics and Electrical Communication Engg.
Indian Institute of Technology Kharagpur
Kharagpur, India
E. Mail : [1]ketan9318@gmail.com, [2]mrityun@ece.iitkgp.ac.in

***Abstract*—In this paper, we propose a new framework for compressed sensing (CS) based on data fusion principles, where several CS algorithms work in parallel to recover a K-sparse signal, and their outputs are combined convexly using a set of combiner coefficients drawn randomly, followed by a $2K$ level hard-thresholding, a pursuit step and final pruning by another K level hard-thresholding. A rigorous convergence analysis is presented and sufficient conditions for convergence are derived. Unlike other existing algorithms which use data fusion concepts but do not employ convex combination of estimates generated by the participating algorithms and instead, use a union of support of individual estimates, the proposed method is free from the restrictions on the number of participating algorithms. Empirically, it is observed that the proposed framework allows us to recover the sparse signal of interest using fewer measurements than required by any individual algorithm. It is further seen that even when some of the participating algorithms do not perform well, the proposed framework is still able to recover the K-sparse signal maintaining the same level of performance. Simulations also reveal that performance of the proposed framework is either at par or better than that of existing algorithms that apply data fusion.**

***Index Terms*—Data fusion, Compressed Sensing, Sparse recovery, Convex combination.**

I. Introduction

In recent years, the field of compressed sensing (CS) has attracted a lot of attention from the research community due to its vast applications. In CS, the problem is to recover a vector $\mathbf{x} \in \mathbb{R}^N$ with number of nonzero entries no larger than K for the linear system with the measurements $\mathbf{y} \in \mathbb{R}^m$ given by $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e}$, where $\mathbf{A} \in \mathbb{R}^{m \times N}, m < N$ is the so-called sensing matrix and $\mathbf{e}$ denotes the measurement error. This can be solved by using linear programming (LP) methods such as the basis pursuit, which aims at solving the following problem

$$\min \|\mathbf{z}\|_1 \quad \text{subject to } \mathbf{y} = \mathbf{A}\mathbf{z}. \tag{1}$$

However, the LP methods are computationally heavy and thus impractical in many applications [1]. In recent years, numerous greedy and thresholding-based algorithms that provide a good tradeoff between computational cost and reconstruction performance have been proposed and analyzed. These include subspace pursuit [1], orthogonal matching pursuit (OMP) [2], iterative hard thresholding (IHT) [3], compressive sampling matching pursuit (CoSaMP) [4], hard thresholding pursuit (HTP) [5], [6], Newton step (NS) based iterative hard thresholding (NSIHT), NS based hard thresholding pursuit (NSHTP) [7] algorithms etc. Success of reconstruction by any of these algorithms depends on parameters like as m, N, K, as well as on the underlying statistical distribution of $\mathbf{x}$, and no sparse recovery algorithm outperforms other algorithms in all respects [8]–[10]. In many cases, statistical distribution of the underlying data is not known and thus we are unable to judiciously choose the best algorithm for the problem under consideration [11].

The concept of merging various estimators in order to form a better estimator has been employed in various applications such as image processing [12], [13], sparse representation [14] and machine learning [15]. For CS, the use of multiple algorithms for a better recovery performance was proposed in [16] and was termed "Fusion of Algorithms for Compressed Sensing (FACS)". Based on FACS, generalized FACS (gFACS) was proposed in [11] wherein, at each iteration, the estimates arising from different algorithms were used for finding a better K sparse estimate. In both FACS and gFACS, a union of support sets of the estimates arising from different algorithms was taken and a least squares (LS) problem was solved with restriction to this unified support. However, the underlying assumption was that the cardinality of this unified support is smaller than m, which in practice may not hold good if the number of participating algorithms is large enough.

Main Contributions : In this paper, a new framework based on data fusion for reconstructing the sparse signal is proposed. This framework is termed as Convex Combination of Compressed Sensing Algorithms (CCCSA). In the CCCSA framework, rather than taking union of the support sets as in the case of gFACS, at each iteration, we take a convex combination of the estimates arising from different algorithms and this is then followed by a hard thresholding (HT) operation. The next steps are similar to that of gFACS, where a LS problem is solved and then pruning is done to ensure that the estimate is K sparse. Unlike FACS and gFACS, the proposed framework, however, does not require a constraint on the size of the unified support, thus permitting any number of algorithms to participate in the CCCSA framework without any modifications. Also, the dimension of the LS problem being solved in the proposed

This work is funded partially by a research grant from the SERB, Govt. of India.

CCCSA framework (at the most $2K \times 2K$) is less than in the case of gFACS, resulting in distinct computational advantage. Though the proposed CCCSA framework can be applied to any set of sparse recovery algorithms, for simplicity, we restrict ourselves in this paper to the algorithms which have been analyzed using restricted isometry property (RIP) and which satisfy the following recursive inequality :

$$\|\mathbf{x}^{k+1} - \mathbf{x}\|_2 \leq \alpha \|\mathbf{x}^k - \mathbf{x}\|_2 + \beta \|\mathbf{e}\|_2, \qquad (2)$$

where $\mathbf{x}^k$ denotes the estimate at k^{th} iteration and α, β are constants that depend on other parameters of the algorithm like stepsize μ in the case IHT and HTP, and stepsize μ and perturbation parameter ϵ in case of the NSIHT and NSHTP. We carry out a convergence analysis of the proposed algorithm using RIP and derive sufficient conditions for convergence. For simulation-based comparative assessment of the proposed CCCSA framework, we evaluate and compare the performance of the proposed scheme with FACS and gFACS in terms of the probability of successful reconstruction and probability of support reconstruction in noiseless and noisy settings, respectively. The following algorithms : IHT, CoSaMP, HTP, NSIHT, and NSHTP are chosen as the participating partners, all of which satisfy (2). Simulation results demonstrate that the proposed CCCSA framework outperforms all the chosen competing algorithms and also its performance is superior to FACS and gFACS.

II. Notations

We use lowercase and uppercase bold letters for vectors and matrices respectively. The number of nonzero entries in $\mathbf{z}$ is denoted by $\|\mathbf{z}\|_0$. A vector $\mathbf{z}$ is said to be K sparse if $\|\mathbf{z}\|_0 \leq K$. For $k \in \mathbb{Z}$, by $H_k(\mathbf{z})$ we refer to the operation of hard thresholding, which retains top k elements (in magnitude) of $\mathbf{z}$ and sets the other elements to zero. The support of a vector $\mathbf{z}$ is defined as $S = \{i : z_i \neq 0\}$ where z_i is the i th element of $\mathbf{z}$. If T is some indexing set then $\mathbf{A}_T$ is a submatrix formed by columns of $\mathbf{A}$ indexed by T and $\mathbf{z}_T$ denotes a subvector of $\mathbf{z}$ with entries indexed by T. The complement of the set T is denoted by $\overline{T}$. The notation $\mathbf{s}_i^n = Algo_i(\mathbf{y}, \mathbf{A}, \mathbf{x}^n)$ denotes the output of one iteration of the i^{th} CS algorithm when presented with the input $\mathbf{x}^n$.

III. Proposed CCCSA Algorithm

The proposed CCCSA framework is a three stage procedure. At the $(n+1)$-th iteration, in the first stage, combining coefficients are generated and a convex combination of the estimates arising from L participating algorithms is taken. These coefficients are allowed to be random, in an attempt that the proposed framework is not biased towards any particular algorithm. This is then followed by a HT operation to retain top $2K$ elements to form $\mathbf{u}^n$. In the second stage, $\tilde{\mathbf{x}}^{n+1}$ is obtained by solving the LS problem given by (3). In the last stage, pruning is done by HT the signal $\tilde{\mathbf{x}}^{n+1}$ to retain top K elements to form $\mathbf{x}^{n+1}$. This resulting estimate $\mathbf{x}^{n+1}$ is then fed to each participating algorithm which completes one iteration of the procedure. The pseudo-code for the proposed framework is presented in Table 1.

Table 1: Proposed CCCSA Algorithm

Input: Initial estimates $\mathbf{x}^0, \mathbf{s}_i^0, i = 1, \ldots, L, \mathbf{A}, \mathbf{y}$. Typically $\mathbf{x}^0 = \mathbf{0}, \mathbf{s}_i^0 = \mathbf{0}$.

While stopping criteria not met

- Generate combining coefficients c_i^n randomly, satisfying $\sum_{i=1}^L c_i^n = 1, c_i^n \geq 0, i = 1, \ldots, L$.
- $\mathbf{u}^n = H_{2K}\left[\sum_{i=1}^L c_i^n \mathbf{s}_i^n\right], supp(\mathbf{u}^n) = U^n$
- $$\tilde{\mathbf{x}}^{n+1} = \arg \min_{\substack{\mathbf{z} \in \mathbb{R}^N \\ supp(\mathbf{z}) \subseteq U^n}} \|\mathbf{y} - \mathbf{A}\mathbf{z}\|_2^2 \qquad (3)$$
- $\mathbf{x}^{n+1} = H_K[\tilde{\mathbf{x}}^{n+1}]$
- $\mathbf{s}_i^{n+1} = Algo_i(\mathbf{y}, \mathbf{A}, \mathbf{x}^{n+1}), i = 1, \ldots, L$.
- $n \leftarrow n + 1$.

Output: Final estimate $\mathbf{x}^*, S^* = supp(\mathbf{x}^*)$.

IV. Convergence Analysis

In this section, we carry out a convergence analysis of the proposed CCCSA algorithm and derive sufficient conditions for convergence. For this, we first recall some of the useful definitions and lemmas.

Definition 4.1. *[17] The Kth order restricted isometry constant (RIC) δ_K for the matrix $\mathbf{A}$ is defined to be the smallest δ such that*

$$(1 - \delta)\|\mathbf{z}\|_2^2 \leq \|\mathbf{A}\mathbf{z}\|_2^2 \leq (1 + \delta)\|\mathbf{z}\|_2^2,$$

holds for all $\mathbf{z} \in \mathbb{R}^N$ satisfying $\|\mathbf{z}\|_0 \leq K$.

Lemma 4.1. *[5] [7] Let $\mathbf{u}, \mathbf{v} \in \mathbb{R}^N$ with $\mathbf{u}$ being k sparse. Then following holds:*

$$\|H_k(\mathbf{v}) - \mathbf{u}\|_2 \leq \sqrt{3}\|(\mathbf{v} - \mathbf{u})_{T \cup V}\|_2, \qquad (4)$$

where $T = supp(\mathbf{u}), V = supp(H_k(\mathbf{v}))$.

Lemma 4.2. *[6] Let $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e}$ denote the noisy measurements of K sparse vector $\mathbf{x}$ and T denote any indexing set of size t. If the sensing matrix satisfies RIP of order $K + t$ with RIC $\delta_{K+t} < 1$, then the solution of pursuit step*

$$\mathbf{z}^* = \arg \min_{\substack{\mathbf{z} \in \mathbb{R}^N \\ supp(\mathbf{z}) \subseteq T}} \|\mathbf{y} - \mathbf{A}\mathbf{z}\|_2^2, \qquad (5)$$

satisfies the following:

$$\|\mathbf{z}^* - \mathbf{x}\|_2 \leq \rho_{K+t}\|\mathbf{x}_{\overline{T}}\|_2 + \gamma_{K,t}\|\mathbf{e}\|_2, \qquad (6)$$

where $\rho_{K+t} = \frac{1}{\sqrt{1 - (\delta_{K+t})^2}}$ and $\gamma_{K,t} = \frac{\sqrt{1+\delta_t}}{1 - \delta_{K+t}}$.

Using the above, it is then possible to state and prove the following :

Theorem 4.1. *Let the i^{th} participating algorithm satisfy (2) with constants α_i, β_i and the sensing matrix $\mathbf{A}$ satisfy $3K^{th}$ order RIP with $\delta_{3K} < 1$. Then for any iteration index $n \geq 0$, the following relation holds:*

$$\left\|\mathbf{x}^{n+1} - \mathbf{x}\right\|_2 \leq \tau^{n+1}\left\|\mathbf{x}^0 - \mathbf{x}\right\|_2 + \gamma' \frac{1 - \tau^{n+1}}{1 - \tau}\|\mathbf{e}\|_2, \quad (7)$$

where $\tau = 3\rho_{3K}\max_i \alpha_i$ *and* $\gamma' = 3\rho_{3K}\max_i \beta_i + \gamma\sqrt{3}$.

Proof. The estimate $\mathbf{x}^{n+1}$ is given by $\mathbf{x}^{n+1} = H_K[\tilde{\mathbf{x}}^{n+1}]$. Applying lemma 4.1, we obtain

$$\begin{aligned}\left\|\mathbf{x}^{n+1}-\mathbf{x}\right\|_2 &\leq \sqrt{3}\left\|\left(\tilde{\mathbf{x}}^{n+1}-\mathbf{x}\right)_{S^{n+1}\cup S}\right\|_2 \\ &\leq \sqrt{3}\left\|\tilde{\mathbf{x}}^{n+1}-\mathbf{x}\right\|_2, \end{aligned} \tag{8}$$

where $S^{n+1} = supp(\mathbf{x}^{n+1})$ and $S = supp(\mathbf{x})$. Now using the update procedure for $\tilde{\mathbf{x}}^{n+1}$ given by (3), it follows from (6) that

$$\begin{aligned}\left\|\tilde{\mathbf{x}}^{n+1}-\mathbf{x}\right\|_2 &\leq \rho_{3K}\left\|\mathbf{x}_{\overline{U^n}}\right\|_2 + \gamma_{K,2K}\|\mathbf{e}\|_2 \\ &= \rho_{3K}\left\|(\mathbf{x}-\mathbf{u}^n)_{\overline{U^n}}\right\|_2 + \gamma_{K,2K}\|\mathbf{e}\|_2,\end{aligned}$$

where, from (6) and the fact that $|U^n|$ is at most $2K$, we have, $\rho_{3K} = \frac{1}{\sqrt{1-(\delta_{3K})^2}}$ and $\gamma_{K,2K} = \frac{\sqrt{1+\delta_K}}{1-\delta_{2K}}$. Since $\|(\mathbf{x}-\mathbf{u}^n)_{\overline{U^n}}\|_2 \leq \|\mathbf{x}-\mathbf{u}^n\|_2$, we obtain from above,

$$\left\|\tilde{\mathbf{x}}^{n+1}-\mathbf{x}\right\|_2 \leq \rho_{3K}\left\|\mathbf{x}-\mathbf{u}^n\right\|_2 + \gamma_{K,2K}\|\mathbf{e}\|_2. \tag{9}$$

Now, from the first stage of the CCCSA framework $\mathbf{u}^n = H_{2K}[\sum_{i=1}^{L} c_i^n \mathbf{s}_i^n]$. Applying Lemma 4.1 with $k = 2K$, we obtain

$$\begin{aligned}\|\mathbf{u}^n-\mathbf{x}\|_2 &\leq \sqrt{3}\left\|\left(\mathbf{x}-\sum_{i=1}^{L} c_i^n\mathbf{s}_i^n\right)_{S\cup U^n}\right\|_2 \\ &= \sqrt{3}\left\|\left(\sum_{i=1}^{L} c_i^n\left(\mathbf{x}-\mathbf{s}_i^n\right)\right)_{S\cup U^n}\right\|_2 \\ &\leq \sqrt{3}\sum_{i=1}^{L} c_i^n\left\|(\mathbf{x}-\mathbf{s}_i^n)_{S\cup U^n}\right\|_2 \\ &\leq \sqrt{3}\sum_{i=1}^{L} c_i^n\left\|\mathbf{x}-\mathbf{s}_i^n\right\|_2.\end{aligned} \tag{10}$$

Here, in the second step, we have used the fact that the combining coefficients satisfy $\sum_{i=1}^{L} c_i^n = 1$. The third step follows from repeated use of triangle inequality while noting that c_i^n are non-negative. Combining (8), (9) and (10) and defining $\gamma \coloneqq \gamma_{K,2K}$ we have,

$$\left\|\mathbf{x}^{n+1}-\mathbf{x}\right\|_2 \leq 3\rho_{3K}\sum_{i=1}^{L} c_i^n\left\|\mathbf{x}-\mathbf{s}_i^n\right\|_2 + \gamma\sqrt{3}\|\mathbf{e}\|_2. \tag{11}$$

Invoking the recursive inequality satisfied by the i-th algorithm with constants α_i and β_i and noting that at any n-th iteration, all the participating algorithms are provided with the same input $\mathbf{x}^n$, (11) can be written as

$$\begin{aligned}\left\|\mathbf{x}^{n+1}-\mathbf{x}\right\|_2 &\leq 3\rho_{3K}\sum_{i=1}^{L} c_i^n\left[\alpha_i\left\|\mathbf{x}-\mathbf{x}^n\right\|_2 + \beta_i\|\mathbf{e}\|_2\right] \\ &\quad + \gamma\sqrt{3}\|\mathbf{e}\|_2 \\ &= 3\rho_{3K}\left(\sum_{i=1}^{L} c_i^n\alpha_i\right)\left\|\mathbf{x}-\mathbf{x}^n\right\|_2 \\ &\quad + \left(3\rho_{3K}\sum_{i=1}^{L} c_i^n\beta_i + \gamma\sqrt{3}\right)\|\mathbf{e}\|_2 \\ &\leq 3\rho_{3K}\alpha_{\max}\left\|\mathbf{x}-\mathbf{x}^n\right\|_2 + \left(3\rho_{3K}\beta_{\max} + \gamma\sqrt{3}\right)\|\mathbf{e}\|_2,\end{aligned}$$

where $\alpha_{\max} \coloneqq \max_i \alpha_i$ and $\beta_{\max} \coloneqq \max_i \beta_i$. Finally, applying above inequality recursively backwards we obtain (7) where τ and γ' are defined as earlier. ■

Corollary 4.1. *For the noiseless measurements i.e.* $\mathbf{e} = \mathbf{0}$*, if* $\max_i \alpha_i < \sqrt{\frac{1-\delta_{3K}^2}{9}}$*, where* $\max_i \alpha_i$ *is defined in Theorem 4.1, then the estimate* $\mathbf{x}^n$ *converges to the target signal* $\mathbf{x}$*.*

Proof. From (7) it readily follows that, if $\max_i \alpha_i < \sqrt{\frac{1-\delta_{3K}^2}{9}}$ then $\tau < 1$, and with $\mathbf{e} = \mathbf{0}$, clearly $\lim_{n\to\infty} \mathbf{x}^n = \mathbf{x}$. ■

Remark 4.1. *The proposed CCCSA framework requires the sensing matrix to satisfy the RIP of order* $3K$ *with small RIC values, which is a relaxed condition than the gFACS model, which requires the sensing matrix to satisfy the RIP of order* $(L+1)K$ *with small RIC values.*

V. Simulations

We consider a problem with dimension $N = 200$. The mean square deviation (MSD) is defined to be $MSD(n) = \frac{\|\mathbf{x}-\mathbf{x}^n\|_2^2}{\|\mathbf{x}\|_2^2}$. The entries of the sensing matrix $\mathbf{A}$ are $i.i.d$ with $a_{ij} \sim \mathcal{N}(0, 1/m)$. For generating the combining coefficients $c_i, i = 1, \ldots, L$, we first generate L random numbers from uniform distribution $\mathcal{U}(0, 1]$ and then normalize them by their sum. The support of $\mathbf{x}$ is chosen randomly from $\{1, \ldots, N\}$ without replacement. The nonzero values of $\mathbf{x}$ are generated from Gaussian distribution $\mathcal{N}(0, 1)$. The values of the stepsize parameter are set to 10 and 1 for the IHT and HTP algorithms respectively. For both NSIHT and NSHTP algorithms, the stepsize parameter μ and perturbation parameter ϵ are set to $\mu = \epsilon = 5$. We allow all the algorithms to run for a maximum of 100 iterations. For each result, ensemble averaging is done by taking 100 instances of the problem. We present our results for both noiseless as well as noisy scenario. In the noiseless setting, we say that reconstruction is successful if the MSD falls below 10^{-3}. For the noisy case, we consider the probability of support recovery (PSR) for performance evaluation. The PSR is defined as $PSR = \frac{|S\cap S^*|}{|S|}$. In the noisy scenario, $\mathbf{e}$ is drawn from $\mathcal{N}(0, \sigma^2\mathbf{I})$ and signal to noise ratio (SNR) is set to $20dB$, where $SNR \coloneqq \|\mathbf{x}\|_2^2/\|\mathbf{e}\|_2^2$. All the simulations are carried out on a core $i5-9500$ CPU equipped with 8 GB of memory. Lastly, all the data fusion schemes i.e. FACS, gFACS and CCCSA use all the five algorithms in each of the result.

In Fig. 1, we plot the probability of recovery with the number of measurements (m). The sparsity level (K) is fixed to $K = 20$ in this case. It is observed that even if the performance of some of the participating algorithms is poor (IHT in this case), the proposed CCCSA framework can

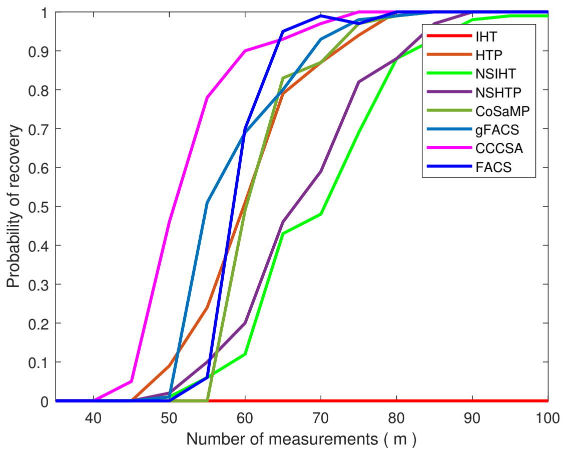

Fig. 1: Variation of recovery performance with number of measurements for $K = 20$ (noiseless measurements).

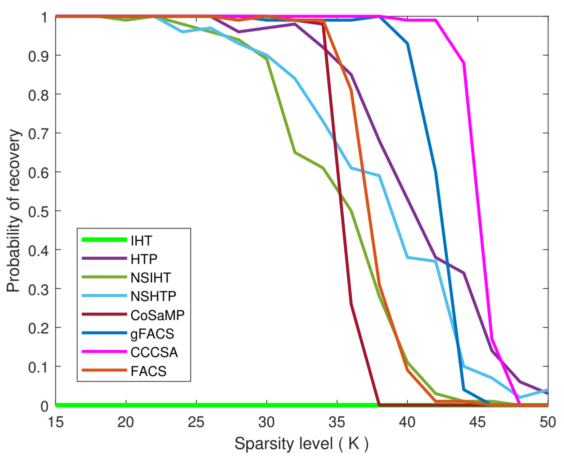

Fig. 2: Variation of recovery performance with level of sparsity (K) for $m = 100$ (noiseless measurements).

reconstruct the signal with less measurements than any other method under consideration including gFACS.

In Fig. 2, the recovery performance against the level of sparsity (K) for fixed number of measurements $(m = 100)$ is shown. In this case also, CCCSA framework is able to recover the signal for a higher sparsity value K than any other method under consideration.

In Fig. 3, we plot PSR against number of measurements for fixed value of $K = 20$ in the noisy scenario. It is observed that while HTP and NSHTP algorithms are able to offer better recovery results when m is small, the CCCSA framework outperforms each method as soon as the number of measurements increase beyond a certain threshold.

In Fig. 4, we plot PSR against the level of sparsity (K), with the number of measurements fixed at $m = 100$ for the noisy scenario. It can be clearly seen that the CCCSA framework offers a better support recovery performance than other methods under consideration.

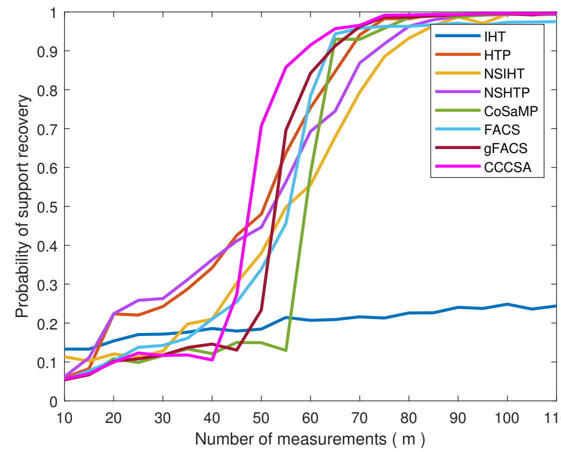

Fig. 3: Variation of probability of support recovery with number of measurements for $K = 20$ (SNR = $20dB$).

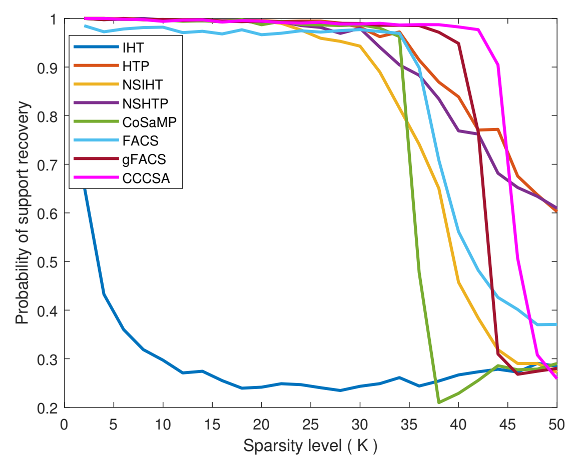

Fig. 4: Variation of probability of support recovery with level of sparsity (K) for fixed $m = 100$ (SNR = $20dB$).

We also compare the run-time of the proposed framework against the gFACS model for a variety of distributions for the target signal $\mathbf{x}$ and the sensing matrix $\mathbf{A}$. For simplicity, we restrict ourselves to the noiseless setting for this particular set of experiments. All the quantities except $\mathbf{x}$ and $\mathbf{A}$ are generated/set as earlier. Three types of sensing matrices are employed which are as follows:

1) Gaussian sensing matrix, as described in the beginning of this section.
2) Gaussian sensing matrix with ℓ_2 normalized columns: We generate the matrix $\mathbf{A}$ as in above and then normalize it so that each column has unit ℓ_2 norm.
3) Bernoulli sensing matrix: The entries of matrix $\mathbf{A}$ are $i.i.d$ with $a_{ij} = \pm 1/\sqrt{m}$, equiprobable.

The support of target signal $\mathbf{x}$ is generated as described earlier at the beginning of this section. Two distributions are used for generating the nonzero values of the target signal $\mathbf{x}$. The distributions are as follows:

Table 2: Experimental results for a fixed number of measurements (m) for various distributions.

S.No	m	K	Dist. of $\mathbf{A}$	Dist. of $\mathbf{x}$	PR - gFACS	Runtime gFACS	PR - CCCSA	Runtime CCCSA
1	100	40	Gaussian	Gaussian	0.92	3.95	**1**	**1.72**
2	100	42	Gaussian	Gaussian	0.75	6.57	**0.99**	**2.76**
3	100	30	Gaussian	Rademacher	0.95	1.49	**0.98**	**0.96**
4	100	32	Gaussian	Rademacher	0.88	2.47	**0.92**	**2.05**
5	100	30	Bernoulli	Rademacher	0.92	1.82	**0.97**	**1.33**
6	100	32	Bernoulli	Rademacher	0.77	3.46	**0.95**	**1.78**
7	100	40	Bernoulli	Gaussian	0.90	4.25	**0.98**	**1.98**
8	100	42	Bernoulli	Gaussian	0.70	6.27	**0.98**	**2.86**
9	100	38	Gaussian, Normalized columns	Gaussian	0.94	2.91	**1**	**1.49**
10	100	42	Gaussian, Normalized columns	Gaussian	0.68	7.46	**0.99**	**2.47**
11	100	30	Gaussian, Normalized columns	Rademacher	0.93	1.78	**0.96**	**1.44**
12	100	42	Gaussian, Normalized columns	Rademacher	0.84	2.88	**0.96**	**1.77**

1) Gaussian distribution, as described in the beginning of this section.
2) Rademacher distribution, with the entries being ± 1 equiprobable.

We fix the number of measurements m to be 100 in this case. The probability of recovery (PR) for the gFACS model and the proposed CCCSA framework along with the corresponding run-time (in seconds) for 100 problem instances are given in Table 2 for different values of sparsity level K. It can be clearly seen from Table 2 that the proposed CCCSA framework is faster and also provides better probability of recovery as compared to the gFACS model. Such a gain in computation is observed due to the fact that pursuit step in the proposed CCCSA framework is solved on a set with a smaller cardinality vis-a-vis the gFACS model, especially in the high sparsity regime. Finally, we would like to mention that our implementation for the algorithms/models used in this paper might not be optimal and further improvements are still possible.

VI. Conclusion

We have proposed and analyzed a new framework based on the data fusion principles for sparse signal recovery from compressed measurements, which takes into account the values of estimates along with the support of the estimate. The numerical experiments carried out on random linear systems indicate that the proposed CCCSA outperforms the existing data fusion based methods in recovering the sparse signal of interest with lower computation.

References

[1] W. Dai and O. Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *IEEE Transactions on Information Theory*, vol. 55, no. 5, pp. 2230–2249, 2009.

[2] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655–4666, 2007.

[3] T. Blumensath and M. E. Davies, "Iterative hard thresholding for compressed sensing," *Applied and Computational Harmonic Analysis*, vol. 27, no. 3, pp. 265–274, 2009.

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

[5] S. Foucart, "Hard thresholding pursuit: An algorithm for compressive sensing," *SIAM J. Numer. Anal.*, vol. 49, no. 6, pp. 2543–2563, 2011.

[6] J.-L. Bouchot, S. Foucart, and P. Hitczenko, "Hard thresholding pursuit algorithms: Number of iterations," *Applied and Computational Harmonic analysis*, vol. 41, no. 2, pp. 412–435, 2016.

[7] N. Meng and Y.-B. Zhao, "Newton-step-based hard thresholding algorithms for sparse signal recovery," *IEEE Transactions on Signal Processing*, vol. 68, pp. 6594–6606, 2020.

[8] A. Maleki and D. L. Donoho, "Optimally tuned iterative reconstruction algorithms for compressed sensing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 330–341, 2010.

[9] S. Chatterjee, D. Sundman, M. Vehkapera, and M. Skoglund, "Projection-based and look-ahead strategies for atom selection," *IEEE Transactions on Signal Processing*, vol. 60, no. 2, pp. 634–647, 2011.

[10] B. L. Sturm, "Sparse vector distributions and recovery from compressed sensing," *arXiv preprint arXiv:1103.6246*, 2011.

[11] A. Zaki, S. Chatterjee, and L. K. Rasmussen, "Generalized fusion algorithm for compressive sampling reconstruction and RIP-based analysis," *Signal Processing*, vol. 139, pp. 36–48, 2017.

[12] M. J. Fadili, J. -L. Starck and L. Boubchir, "Morphological Diversity and Sparse Image Denoising," IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07, 2007, pp. I-589-I-592.

[13] J.-L. Starck, D. L. Donoho, and E. J. Candes, "Very high quality image restoration by combining wavelets and curvelets," in *Wavelets: Appl. in Signal and Image Processing IX*, vol. 4478. SPIE, 2001, pp. 9–19.

[14] M. Elad and I. Yavneh, "A plurality of sparse representations is better than the sparsest one alone," *IEEE Transactions on Information Theory*, vol. 55, no. 10, pp. 4701–4714, 2009.

[15] A. Dalalyan and A. B. Tsybakov, "Aggregation by exponential weighting, sharp PAC-bayesian bounds and sparsity," *Machine Learning*, vol. 72, no. 1, pp. 39–61, 2008.

[16] S. K. Ambat, S. Chatterjee, and K. Hari, "Fusion of algorithms for compressed sensing," *IEEE Transactions on Signal Processing*, vol. 61, no. 14, pp. 3699–3704, 2013.

[17] S. Foucart and H. Rauhut, *A mathematical introduction to compressive sensing*. Springer, 2013.