NONCONVEX COMPRESSIVE SENSING RECONSTRUCTION FOR TENSOR USING STRUCTURES IN MODES

Xin Ding† Xin Ding⋆ Wei Chen†⋆ Ian Wassell⋆

⋆ Computer Laboratory, University of Cambridge, UK
† State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, China

ABSTRACT
This paper focuses on the reconstruction of a tensor captured using Compressive Sensing (CS). Instead of processing the signals via vectorization as is done in conventional CS, in tensor CS high dimensional signals are kept in their original formats, which benefits hardware implementation and eases memory requirements. In addition, more structures exist in a tensor along its various dimensions than in its vectorized format. Utilizing these various structures, this paper proposes a general reconstruction algorithm for tensor CS. Employing the proximity operator of a nonconvex norm function, a special case for a tensor with low rank and sparse structures is elaborated, which is shown to outperform the state-of-art tensor CS reconstruction methods when applied to magnetic resonance imaging and hyper-spectral imaging.

Index Terms—Compressive sensing, tensor reconstruction, sparse and low rank reconstruction.

1. INTRODUCTION
Compressive Sensing (CS) [1, 2, 3], which achieves sensing and compression at the same time, is drawing increasing attention for data acquisition. It relies upon the fact that most practical signals can be represented with only a few non-zero coefficients (i.e., the signal is sparse) in a suitable basis and guarantees the reconstruction of such signals from far fewer samples than that required by the Nyquist sampling theorem.

However, it is challenging to apply CS in modern applications, which often generate signals with multidimensional structures, e.g., Magnetic Resonance Imaging (MRI) and Hyper-Spectral Imaging (HSI). In conventional CS, a multidimensional signal, i.e., a tensor, is mapped in vector format for sensing and reconstruction. Such a vectorized tensor in real-world applications requires an enormous CS sensing matrix, which imposes large demands in terms of processing power and memory size. In addition, sensing the vectorized signal is equivalent to simultaneously multiplexing along all data dimensions, which is difficult to achieve in hardware. Besides, a tensor usually possesses structures along its various dimensions; while the vectorization results in a loss of structure.

In the past few years, extension of CS to 2D signals, i.e., matrices, has been made [4, 5, 6, 7], where a rank deficient model is involved rather than simply assuming a sparse model. However, research concerning CS for tensors is still quite rare. In [8], Kronecker product matrices are proposed to form the CS sensing matrix, thereby the sensing procedure is partitioned along dimensions. The theory presented in this paper paves the way to tensor CS. Using the canonical polyadic decomposition model of a tensor, in [9, 10], a framework for recovering tensors that have sparse factor matrices is proposed. A different approach, based on the Tucker decomposition model that is more intuitively related to practical CS, multidimensional CS is proposed in [11, 12]. The authors assume a sparse Tucker core tensor and propose the Kronecker Orthogonal Matching Pursuit (KOMP) algorithm to reconstruct a tensor from the Kronecker CS samples. It is also extended to exploit the block sparse structure and a Block-based KOMP algorithm is proposed. Other related work lies in the field of tensor completion, where a low rank tensor model is assumed [13, 14]. However, all the aforementioned work utilizes either a sparse or a low rank model for the whole tensor, which does not fully exploit the various structures present in a high dimensional signal.

In this paper, we propose an approach to CS tensor reconstruction, in which diverse structures along each dimension (i.e., mode) of a tensor are exploited. Even though simultaneous structures have been considered for matrices, e.g., the matrix signal is both rank deficient and sparse or group sparse [7, 15, 16], their approaches are not applicable to our problem because they do not separately sense and reconstruct the signals along dimensions using the Kronecker structures, which means that their implementations are limited by the large data size of a tensor. In some other work [17, 18], dictionary learning techniques are extended to tensors to take advantage of structures in modes. However, to the best of our knowledge, there is no prior work focused on exploiting various structures in the tensor modes in the reconstruction process of tensor CS. We solve the proposed multi-structure optimization problem using an approach based upon the Alternating Di-
rection Method of Multipliers (ADMM) [19], in which we employ nonconvex reconstruction because the convex relaxation commonly used in CS is shown to not perform well in the case of simultaneous structures [20]. The derived algorithm overcomes the difficulty of high memory requirements and computation loads encountered for tensor reconstruction in conventional CS and ADMM, and it provides the flexibility of involving various structures, thereby improving the reconstruction accuracy.

1.1. Multi-linear Algebra and Notations
A multidimensional array \( \mathbf{X} \in \mathbb{R}^{N_1 \times \ldots \times N_n} \) is known as a mode-\( n \) tensor, where the dimension of its \( i \)-th mode is \( N_i \). The mode-\( i \) vectors are determined by fixing every index except the one in the mode \( i \). The mode-\( i \) unfolding matrix \( \mathbf{X}(i) \in \mathbb{R}^{N_i \times N_1 \cdots N_{i-1} N_{i+1} \cdots N_n} \) is obtained by arranging all the mode-\( i \) vectors as columns of a matrix. Given a matrix \( \mathbf{A} \in \mathbb{R}^{J \times K} \), the mode-\( k \) tensor by matrix product is defined as \( \mathbf{Z} = \mathbf{X} \circ \mathbf{A} \). The \( \mathbf{Z} \) is obtained by arranging all the \( k \)-th mode and \( \mathbf{A} \circ \mathbf{B} \) as columns of a matrix. Specifically, in conventional CS and ADMM, and it provides the flexibility of exploiting various structures is formulated as:

\[
\begin{align*}
\min_{\mathbf{X}} & \quad \sum_{i=1}^{n} \alpha_i \| \mathbf{X}(i) \|_{(i-\text{th norm})} \\
\text{s.t.} & \quad \mathbf{Y} = \mathbf{S} \times_1 \mathbf{A}_1 \times_2 \mathbf{A}_2 \cdots \times_n \mathbf{A}_n,
\end{align*}
\]

which is NP-hard and is normally solved by replacing the \( l_0 \) norm with a \( l_1 \) norm. The reconstruction is guaranteed when the sensing matrix \( \mathbf{A} \) obeys the Restricted Isometry Property (RIP) [1]. The relationship between the RIP of \( \mathbf{A} \) and each component \( \mathbf{A}_i \) has been analyzed in [8]. Nonconvex optimization, where the \( l_0 \) norm in (5) is replaced by a \( l_p \) \((0 < p < 1)\) quasi-norm, has been shown [21, 22] to be capable of further reducing the required number of samples for reconstruction, while still being tractable.

Tensor CS enlarges the dimension of the signals that can be processed in a conventional CS system. Specifically, in conventional CS, an \( n \)-mode tensor \( \mathbf{X} \in \mathbb{R}^{N_1 \times \ldots \times N_n} \) is vectorized as \( \mathbf{x} \in \mathbb{R}^{N_1 N_2 \cdots N_n} \), which requires a global multiplexing with a sensing matrix of size \( M_n \times N_n \); while in tensor CS, such a sensing matrix is partitioned as \( n \) smaller sensing matrices of size \( M \times N \), which reduces the memory storage requirements, and also eases hardware implementation. In addition, extending CS reconstruction algorithms to utilize the Kronecker structure reduces the computational complexity. This is primarily because the complexity involved in operations such as \( \mathbf{y} - \mathbf{A} \mathbf{x} \) is \( O(M^n N^n + M^n) \); while that for \( \mathbf{y} - \mathbf{S} \times_1 \mathbf{A}_1 \times_n \mathbf{A}_n \) is \( O([M N^n (1 - M^n/N^n)](1 - M/N) + M^n) \), which is always smaller than the former.

3. TENSOR CS RECONSTRUCTION USING STRUCTURES IN MODES
High dimensional signals often possess various structures along its dimensions, e.g., sparse, low rank, row sparse, column sparse. Utilizing these structures simultaneously can reduce the degrees of freedom of the reconstruction problem and consequently improve the reconstruction performance [20]. The reconstruction problem that provides the flexibility of exploiting various structures is formulated as:

\[
\begin{align*}
\min_{\mathbf{X}} & \quad \sum_{i=1}^{n} \alpha_i \| \mathbf{X}(i) \|_{(i-\text{th norm})} \\
\text{s.t.} & \quad \mathbf{Y} = \mathbf{S} \times_1 \mathbf{A}_1 \times_2 \mathbf{A}_2 \cdots \times_n \mathbf{A}_n,
\end{align*}
\]

where \( \Omega_i \) denotes an orthogonal transform for the \( i \)-th mode, \( \| \cdot \|_{(i-\text{th norm})} \) is a general form of norms that can be defined specifically according to the structure of the \( i \)-th mode and \( \alpha_i \) \((i = 1, \ldots, n)\) are the tuning parameters. This problem can be rewritten as:

\[
\begin{align*}
\min_{\mathbf{X}} & \quad \sum_{i=1}^{n} \alpha_i \| \mathbf{X}(i) \|_{(i-\text{th norm})} \\
\text{s.t.} & \quad \mathbf{Y} = \mathbf{S} \times_1 \mathbf{A}_1 \times_2 \mathbf{A}_2 \cdots \times_n \mathbf{A}_n,
\end{align*}
\]

where \( \mathbf{Z}_i \) \((i = 1, \ldots, n)\) are intermediate variables.

We then propose to solve this problem using the Alternating Direction Method of Multipliers (ADMM) [19]. However, since large dimensions are involved in the problem, the original ADMM becomes impractical as it leads to high storage requirements and a significant computational burden. We thus derive efficient calculations in the ADMM steps to enable implementation. The augmented Lagrangian function of
(7) is defined as:

\[ L_i(x, z_i, \ldots, z_n, p_1, \ldots, p_n, \Omega) = \sum_{i=1}^{n} (\alpha_i |\Omega_i z_i(i)|_{\text{1-th norm}}) + \langle \Omega_i(z_i - x_i), \Omega_i p_i \rangle + \frac{\rho}{2} |\Omega_i(z_i - x_i)|_2^2 + \langle y - x, \Phi_i \times_n \Phi_n \rangle, \Omega \succ 0, \]

in which \( \rho > 0 \), and \( \Omega \) and \( Q \) are dual variables. The ADMM then proceeds by iteratively updating \( z_i, x, p_i \) and \( Q \) with the others fixed.

Updating each \( z_i \) for \( i = 1, \ldots, n \) yields:

\[ \begin{aligned}
    z_i^{t+1} &= \min_{z_i} \alpha_i |\Omega_i z_i(i)|_{\text{1-th norm}} \\
    &\quad + \frac{\rho}{2} |\Omega_i z_i(i) - \Omega_i x_i - \rho p_i |_2^2 \\
    &= \Omega_i^T \text{prox}_{\frac{\rho}{\alpha_i}}(\Omega_i x_i + \frac{1}{\rho} \Omega_i p_i),
\end{aligned} \]

where \( \text{prox}_{\frac{\rho}{\alpha_i}}(z) \) denotes the proximal operator [23] and is defined for function \( f \) as:

\[ \text{prox}_{\frac{\rho}{\alpha_i}} f(x) = \arg \min_{y} \frac{1}{\rho} f(y) + \frac{1}{2\rho} \|x - y\|_2^2. \]

The convergence of convex ADMM has been studied in [19] and a recent work also analyzes the nonconvex and smooth case [26]. The convergence issue for a nonconvex and nonsmooth case is still an open question, but it has been empirically shown to converge reliably for certain cases [6].

4. NONCONVEX CS RECONSTRUCTION FOR TENSORS WITH LOW RANK AND SPARSE MODES

In this section, we consider a special case when a tensor has low rank and sparse modes, i.e., \( \text{rank}(X(i)) < r_i \) for some of the modes \( i \) and \( \text{supp}(X(i)) < K \) for the others. Such models widely exist in practical applications, e.g., MRI, HSI, video processing and sub-wavelength optical imaging [27].

Here, we employ nonconvex functions to model the sparsity and rank deficiency. Specifically, we adopt the proximal \( l_p \)-norm (denoted by \( G_{p,r} \)) as defined in [6], of which the proximity has been derived as a \( p \)-shrinkage operator that functions element-wise as follows:

\[ \text{shrink}_{p}(t, \mu) = \max(0, |t| - \mu |t|^{p-1}) \frac{t}{|t|}. \]

The optimization problem then becomes:

\[ \min_X \sum_{i \in \mathcal{S}} \alpha_i(G_{p,r}(\Omega_i X(i))) + \sum_{i \in \mathcal{L}} \alpha_i(G_{p,r}(\sigma(X(i)))). \]

We observe that in equation (10), the term \((nI + \Phi^T \Phi)\) has the size \( N_1 N_2 \ldots N_n \times N_1 N_2 \ldots N_n \), which is not feasible to be processed by a conventional computing device and worse, its inverse is even more difficult to calculate having a complexity of \( O(N_1 N_2 \ldots N_n^3) \).

To keep the advantage of low memory requirement in tensor CS, we rewrite the term \((nI + \Phi^T \Phi)\) as:

\[ (U_1 \otimes \cdots \otimes U_n)(nI + V_n \otimes \cdots \otimes V_1)^{-1}(U_n^T \otimes \cdots \otimes U_1^T), \]

where \( U_i \) and \( V_i \) come from the eigendecompositions of \( \Phi_i^T \Phi_i \), i.e., \( U_i \times_n \Phi_i = \Phi_i \times_n \Phi_i \) and the matrices \( V_i \) (\( i = 1, \ldots, n \)) are diagonal. Therefore the term \((nI + V_n \otimes \cdots \otimes V_1)\) is also diagonal and its inverse can be easily obtained. The computational complexity is now reduced to that of the eigendecompositions, i.e., \( O(N_1^3) \) and they only need to be calculated once before the iterations start. To avoid the storage of the large matrix in (11), we partition it into small matrices and update \( X \) by:

\[ X_i^{t+1} = [\text{ten}(v) \circ (G_i \times_1 U_i^T \times_n U_n^T)] \times_1 U_1 \times_n U_n, \]

where \( v = \text{diag}((nI + V_n \otimes \cdots \otimes V_1)^{-1}) \), \( G = \sum_{i=1}^{n} (Z_i - \frac{1}{\rho} P_i) + (\frac{1}{\rho} Q + \frac{1}{\rho} Y) \times_1 \Phi_i \times_n \Phi_n \). In this way, the complexity of the matrix-vector multiplication in (10) is also reduced, as discussed in Section 2.

In the final steps, the dual variables are updated by:

\[ \begin{aligned}
    P_i^{t+1} &= P_i^t - \rho (Z_i^{t+1} - X_i^{t+1}) \\
    Q_i^{t+1} &= Q_i^t - \rho (X_i^{t+1} + \frac{1}{\rho} \Phi_i \times_n \Phi_n - Y).
\end{aligned} \]

5. SIMULATION RESULTS

In this section, we evaluate the proposed approach, i.e., TADMM, on 3-mode tensors with low rank and sparse modes in both MRI and HSI applications. In both experiments, we compare TADMM with the state-of-the-art tensor CS algorithms, which include the sparsity based methods: Kronecker CS
(KCS) [8] and Block sparse KOMP (BKOMP) [11]; the low rank based method: Low Rank Tensor Completion (LRTC) [13]; and the sparse and low rank based method: Split Bregman algorithm (SB) [15]. For the SB algorithm, as it was originally designed for vectorized signals, we modify it to utilize the tensor Kronecker product so that it can be applied to large dimensional signals. We also extend it to utilize the proximal $p$-norm so that it becomes a nonconvex method, denoted by SB-$p$. We employ the SPGL1 solver [28] for the KCS method.

In both MRI and HSI experiments, for the TADMM algorithm, mode 1 and 2 of the tensor are sparse in a 4 level DWT basis and mode 3 is low rank in the original domain. A 3D 4 level DWT is utilized for the sparsity based methods. The sensing matrices $\Phi_1, \Phi_2$ are set as i.i.d Gaussian sensing matrices with various sample ratios and $\Phi_3$ is an identity matrix. To initialize the ADMM algorithm, we set $X^0 = P_i^0 = Z_i^0 = 0$ ($i = 1,...,n$) and $Q^0 = Y$. The parameters are chosen empirically as $\alpha_1 = \alpha_2 = \alpha_3 = 1$, $\rho^0 = 1e - 4$, $\rho^{i+1} = 1.15\rho^i$ [13] (increasing $\rho$ gradually improves the rate of convergence). The stopping criteria for all the methods are as follows: 1) the normalized consecutive change on $X$ is small; 2) the normalized residue $R = Y - X \times_1 \Phi_1 \times_2 \Phi_2 \times_3 \Phi_3$ is small; 3) the normalized consecutive change on $R$ is small.

### Table 1. Accuracy and Running Time Comparison when sampling ratio is 0.2.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>PSNR (dB)</th>
<th>Time (h)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TADMM ($p = 0.2$)</td>
<td>56.22</td>
<td>1.73</td>
</tr>
<tr>
<td>TADMM ($p = 0.7$)</td>
<td>55.64</td>
<td>1.69</td>
</tr>
<tr>
<td>TADMM ($p = 1$)</td>
<td>53.95</td>
<td>1.39</td>
</tr>
<tr>
<td>SB ($p = 0.2$)</td>
<td>53.51</td>
<td>1.74</td>
</tr>
<tr>
<td>SB ($p = 0.7$)</td>
<td>53.84</td>
<td>1.70</td>
</tr>
<tr>
<td>SB ($p = 1$)</td>
<td>52.73</td>
<td>1.35</td>
</tr>
<tr>
<td>KCS</td>
<td>51.13</td>
<td>25.1</td>
</tr>
<tr>
<td>BKOMP</td>
<td>37.13</td>
<td>0.77</td>
</tr>
<tr>
<td>LRTC</td>
<td>36.82</td>
<td>0.54</td>
</tr>
</tbody>
</table>

The Peak Signal to Noise Ratio (PSNR) is employed for the evaluations. The experiments are conducted on a Macbook pro with a 2.6GHz Intel Core i5 CPU and 8GB RAM.

We employ a cardiac sequence [29] with size $128 \times 128 \times 128$ for the MRI experiments and Fig. 1(a) shows the PSNR results when various sampling ratios (i.e., $M_1M_2/(N_1N_2)$) are used. The superiority of the proposed approach is clear. The BKOMP and LRTC methods perform poorly because of model mismatch. The reconstructed 100-th slice of the best four approaches when the sampling ratio is 0.3 is shown in Fig. 2(a) for visual comparison.

For the HSI experiments, Scene 4 in the natural scenes HSI database [30] with size $1024 \times 1024 \times 32$ (i.e., 32 frequency channels) is employed. The PSNR results using various sampling ratios are shown in Fig. 1(b) and the reconstruction results when the sampling ratio is 0.1 are displayed in RGB in Fig. 2(b). Our approach is shown to outperform the others in the HSI application. We then vary the value of $p$ and compare the accuracy and running time in Table 1, where it can be observed that smaller $p$ leads to more accurate reconstruction and slower execution time. Note that $p=1$ corresponds to the case of convex relaxation. The TADMM algorithm has the highest accuracy during all the tests and a similar execution time to the SB method. In addition, we observe that the sparse based basis pursuit algorithm KCS runs significantly slower than the others; while the BKOMP and LRTC methods converge faster but lead to inaccurate solutions as their assumed models do not closely match the application.

### 6. CONCLUSION

In this paper, an ADMM based tensor CS reconstruction approach utilizing various structures in tensor modes is proposed. The general method can be applied to any structures as long as the proximity operators for the associated norms are derivable. In addition, efficient calculations that ease memory requirements and reduce computational complexity for large data are derived. In particular, the TADMM algorithm for a tensor with sparse and low rank modes is presented in detail, in which the nonconvex shrinkage operators are used. The experiments for MRI and HSI applications demonstrate the superiority of the proposed method compared to the state-of-art tensor CS approaches.
7. REFERENCES


