DETECTION OF ROW-SPARSE MATRICES WITH ROW-STRUCTURE CONSTRAINTS

David Gregoratti, Carlos Buelga, and Xavier Mestre

Centre Tecnològic de Telecomunicacions de Catalunya (CTTC/CERCA)
Parc Mediterrani de la Tecnologia – 08860 – Castelldefels – Barcelona (Spain)
email: first.last@cttc.es

ABSTRACT

An underdetermined multi-measurement vector linear regression problem is considered where the parameter matrix is row-sparse and where an additional constraint fixes the number of nonzero elements in the active rows. Even if this additional constraint offers side structure information that could be exploited to improve the estimation accuracy, it is highly nonconvex and must be dealt with with caution. A detection algorithm is proposed that capitalizes on compressed sensing results and on the generalized distributive law (message passing on factor graphs).

Index Terms— Structured sparsity, compressed sensing, factor graphs, message passing.

1. INTRODUCTION

Mathematical models involving a huge number of parameters demand significant amounts of storage and computation capabilities, and extracting useful information from them may be a challenging task. This is especially true when practical issues (e.g., time constraints) limit the number of observations and lead to underdetermined problems.

In this context, the sparsity hypothesis has proven to be a powerful ally [1, 2]: by assuming a sparse parameter vector (that is, most parameters are negligible) one can easily reduce computation/storage requirements and, most importantly, can tackle ill-posed problems with more degrees of freedom than observations, which otherwise present no meaningful solution. Note that there exist problems that are inherently sparse like, for instance, massive machine-type communications systems with a huge number of registered devices but only a reduced number of them transmitting at a time. However, the sparsity assumption may be helpful also in those cases where the model is deliberately overcomplete to compensate for an inadequate understanding of the phenomenon: Examples include medical image processing and genetic analysis (see, e.g., [3, 4, 5]).

For a considerable number of problems, some prior information exists that links two or more model parameters together. As an example, think about sensors grouped by geographical position or genes that activate in clusters. In other words, the unknown sparse representation of the considered model is characterized by a specific structure that can be exploited to improve algorithmic performance in terms of accuracy or observation size (see, e.g., [6, 7]). This is also the objective of this paper, which considers a sparse estimation problem whose unknown parameters obey a distinctive sparsity pattern.

This work has been supported by the Spanish Government under grants TEC2014-59255-C3-1 and TEC2015-69868-C2-2-R (ADVENTURE), and by the Catalan Government under grant 2017-SGR-01479.

2. PROBLEM FORMULATION

We consider a classic multi-measurement vector (MMV) regression problem where the measurement matrix, $\mathbf{Y} \in \mathbb{R}^{N\times L}$, is modeled as the product of the sampling matrix, $\mathbf{S} \in \mathbb{R}^{N\times M}$, and the true parameter matrix, $\mathbf{X}^* \in \mathbb{R}^{M\times L}$, plus additive white Gaussian noise, $\mathbf{W} \in \mathbb{R}^{N\times L}$:

$$\mathbf{Y} = \mathbf{S}\mathbf{X}^* + \mathbf{W}.$$

Specifically, we are interested in the underdetermined problem where $N < M$. Even if such problem is ill-conditioned, compressed sensing (CS) results (see, e.g., [8]) show that $\mathbf{X}^*$ can be recovered from $\mathbf{Y}$ with high accuracy under some mild assumptions on $\mathbf{S}$ as long as $\mathbf{X}^*$ is sparse. One possible solution is to approximate $\mathbf{X}^*$ by

$$\hat{\mathbf{X}} = \arg\min_{\mathbf{X}} \frac{1}{2}\|\mathbf{Y} - \mathbf{S}\mathbf{X}\|_{\mathrm{F}}^2 + \lambda\|\mathbf{X}\|_1 \tag{1}$$

where $\|\cdot\|_{\mathrm{F}}$ denotes the Frobenius matrix norm, $\|\mathbf{X}\|_1 = \sum_{i,j}|x_{i,j}|$ and where λ is a real positive constant tuned to achieve the desired sparsity order.

However, our model is characterized by a distinguishing sparsity structure (see also Fig. 1), which we would like to exploit to improve the estimate precision. Specifically, we consider the case where the nonzero elements of $\mathbf{X}^*$ are concentrated in few rows. Moreover, each of these active rows has a fixed number of nonzero elements, namely r. In other words, for each row $\mathbf{r}_m \in \mathbb{R}^L$ of $\mathbf{X}$, $m = 1, 2, \ldots, M$, problem (1) presents the additional constraint that either

$$\|\mathbf{r}_m\|_0 = 0 \qquad \text{or} \qquad \|\mathbf{r}_m\|_0 = r \tag{2}$$

with $\|\cdot\|_0$ the so-called "0-norm," that is the number of nonzero elements of a vector.[1] One readily sees that this constraint is nonconvex and should be handled with care.

We refer to [9] as an example of a practical application of this model. There, the authors consider a random multiple access channel where active users are allowed to transmit in only r randomly chosen slots within each L-slot-long frame. In that case, the columns of $\mathbf{S}$ would represent the pilot signatures used to identify the devices, while matrix $\mathbf{X}^*$ would contain the channel coefficients that must be estimated at the receiver side for coherent detection of the information message.

2.1. Literature Overview

Even though there exists a notable number of works dealing with structured sparsity, none seems to capture the specificities of the model presented here. Consequently, only suboptimal solutions can

[1] By extension, when applied to a scalar, $\|x\|_0 = 1$ if $x \neq 0$ and zero otherwise.

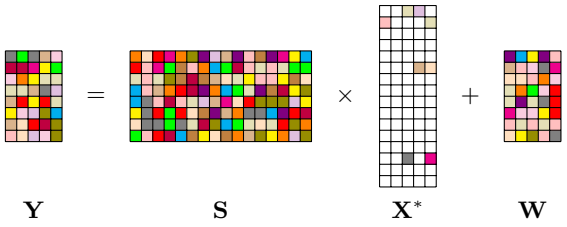

Fig. 1. Graphical representation of the considered sparse signal model: White blocks stand for zero matrix entries.

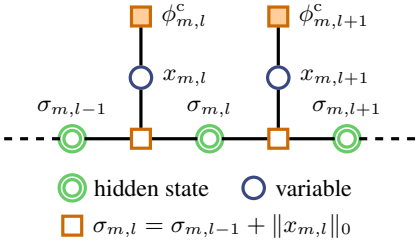

Fig. 2. Factor-graph representation of problem (5).

be found that may introduce penalizing approximations. For instance, we could decide to ignore constraint (2) and simply look for a row-sparse solution with one of the many algorithms that have been devised for this purpose (see, e.g., [10] and references therein). Note that, typically, algorithms promoting row sparsity result in a uniform energy distribution among the entries of each active row and, generally, $\|\mathbf{r}_m\|_0 = L$ for all active m. A possible fix consists in an *hard-decision* step where we force to zero all entries other than the r ones with the highest magnitude.

A slightly more sophisticated approach consists in modifying constraint (2) to require that matrix $\mathbf{X}$ is a row-sparse matrix with sparse rows. Such a structure can be induced by a double regularizer, as explained in [3]. Namely, problem (1) subject to (2) is relaxed to

$$\hat{\mathbf{X}} = \arg\min_{\mathbf{X}} \frac{1}{2}\|\mathbf{Y} - \mathbf{S}\mathbf{X}\|_{\mathrm{F}}^2 + \mu\|\mathbf{X}\|_{2,1} + \lambda\|\mathbf{X}\|_1 \qquad (3)$$

where $\mu\|\mathbf{X}\|_{2,1} = \mu\sum_{m=1}^{M}\sqrt{\sum_{l=1}^{L} x_{m,l}^2}$, $\mu > 0$, is the regularizer forcing row sparsity, while $\lambda\|\mathbf{X}\|_1$, $\lambda > 0$ is the regularizer forcing general sparsity and, in turn, sparsity within active rows. The resulting problem is convex and can be solved very efficiently. However, as before, there is no control on the number of nonzero elements per active row and one should rely again on the hard-decision step to obtain exactly r nonzero entries per active row. Note that this step can be carried out only once after solving (3) or, if applicable, after every iteration of the solving algorithm (e.g., of the proximal gradient method).

Other works in the literature allow for a more accurate characterization of the sparsity structure. For example, one may modify (3) and replace the $\ell_{2,1}/\ell_1$ regularizer by a new ad-hoc one [11]. Another approach would be to extend classical greedy algorithms like the Orthogonal Matching Pursuit (OMP) algorithm following the ideas in, e.g., [12, 6]. Both cases, however, require an exhaustive search over the *atoms* of the sparsity model: This can be a severe limitation for the problem at hand where each row shows $\binom{L}{r}$ different activation patterns.

3. PROPOSED APPROACH

In this section we propose a novel approach for the estimation of $\mathbf{X}^*$ subject to (2) that draws inspiration from the Generalized Distributive Law (GDL) and all the related message-passing algorithms [13, 14]. The resulting method successively improves on the approximation

$$\hat{\mathbf{X}} = \arg\min_{\mathbf{X}} \frac{1}{2}\|\mathbf{Y} - \mathbf{S}\mathbf{X}\|_{\mathrm{F}}^2 + \lambda\|\mathbf{X}\|_0 \qquad \text{s.t.} \qquad (2) \qquad (4)$$

of the true $\mathbf{X}^*$ [note the difference in the regularizer with respect to (1)] by carrying out simple computations on the columns and rows of $\mathbf{X}$ in an iterative fashion.

3.1. A Simplified Case

For the sake of clarity, let us suppose for now that constraint (2) only holds for a single row, say $m = 1$ without loss of generality, while all other rows are free to move in the solution space. Then, problem (4) can be rewritten as

$$\hat{\mathbf{X}} = \arg\min_{\{x_{1,l}\}_{l=1}^{L}} \sum_{l=1}^{L} \phi_{1,l}^{\mathrm{c}}(x_{1,l}) \qquad \text{s.to} \qquad \|\mathbf{r}_1\|_0 \in \{0, r\} \quad (5)$$

where

$$\phi_{1,l}^{\mathrm{c}}(x_{1,l}) = \min_{\{x_{m,l}\}_{m=2}^{M}} \frac{1}{2}\|\mathbf{y}_l - \mathbf{S}\mathbf{x}_l\|_2^2 + \lambda\|\mathbf{x}_l\|_0 \qquad (6)$$

and where $\mathbf{x}_l$ and $\mathbf{y}_l$ are the lth columns of $\mathbf{X}$ and $\mathbf{Y}$, respectively. We will refer to $\phi_{1,l}^{\mathrm{c}}(x_{1,l})$ as the *column-wise marginal* (of the objective function) with respect to entry $x_{1,l}$.

Now, let us introduce a set of hidden state variables $\sigma_{1,l} \in \{0, 1, \ldots, L\}$, with $l = 0, 1, \ldots, L$, and define the state transition according to

$$\sigma_{1,0} = 0$$
$$\sigma_{1,l} = \sigma_{1,l-1} + \|x_{1,l}\|_0.$$

Also, we associate a cost to the transition $\sigma_{1,l-1} \to \sigma_{1,l}$ given by $\phi_{1,l}^{\mathrm{c}}(0)$ if $\|x_{1,l}\|_0 = 0$ and by $\phi_{1,l}^{\mathrm{c}}(*) = \min_{x_{1,l}\neq 0} \phi_{1,l}^{\mathrm{c}}(x_{1,l})$ when $\|x_{1,l}\|_0 = 1$. Then, solving problem (5) is equivalent to finding the most likely (minimum cost) sequence of states with either $\sigma_{1,L} = 0$ or $\sigma_{1,L} = r$ and can be achieved by a simple message-passing algorithm on a factor graph similar to the one depicted in Fig. 2 [13, 14]. More specifically, we end up with a Viterbi-like algorithm on a state trellis like the one of Fig. 3.

Remark: Note that, for the Viterbi algorithm to work, there should generally be a noticeable difference between the values $\phi_{1,l}^{\mathrm{c}}(0)$ and $\phi_{1,l}^{\mathrm{c}}(*)$. Indeed, if this is not the case, all sequences take the same cost value and the minimization is meaningless. This is the reason why we introduced the "0-norm" in (4): Due to the discontinuous nature of $\|\cdot\|_0$, the event $\phi_{m,l}^{\mathrm{c}}(0) = \phi_{m,l}^{\mathrm{c}}(*)$ is much less likely to happen as compared to other regularizers based on proper continuous norms like, e.g., $\|\cdot\|_1$.

Wrapping up, if constraint (2) referred to a single row, problem (4) would admit an "exact" solution that could be computed in a single iteration by solving, first, column problems (6) and, second, row problem (5) as explained above.[2] Next, we analyze why the approach is no longer "exact" when constraint (2) holds for all rows of $\mathbf{X}$.

[2]We write "exact" between quotation marks since some approximations are introduced when computing the column-wise marginals, as explained in Section 3.3.

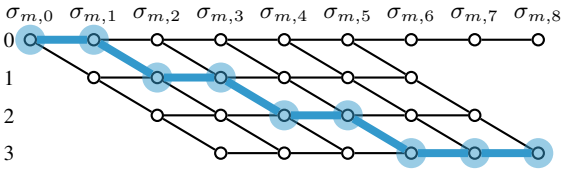

Fig. 3. Trellis representation of row problem (5) for $L = 8$ and $r = 3$.

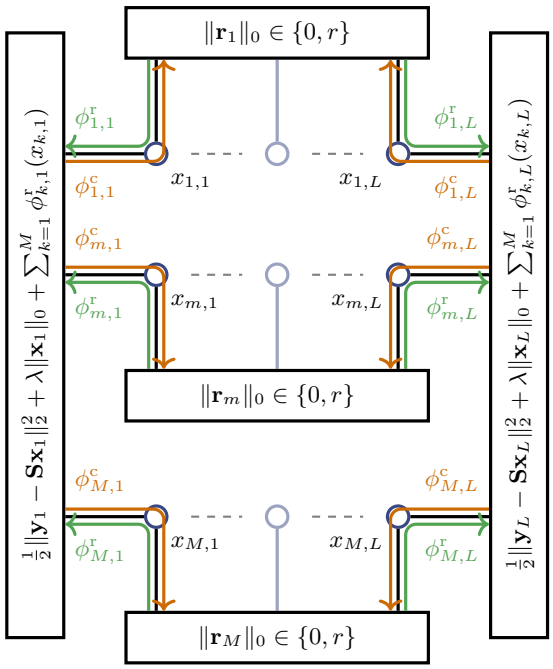

Fig. 4. Factor graph of the main problem and associated messages. Each row function node can be further decomposed as in Fig. 2.

3.2. The Exhaustive Case

When constraint (2) holds for all rows of $\mathbf{X}$, the above approach is not rigorous anymore. Indeed, the presented solution relies on the fact that, except for row 1, the objective function of (4) is separable in the columns of $\mathbf{X}$. This fact allows transforming (4) into (5) by means of the marginalization operation in (6). Conversely, when all constraints are in place, all variables are coupled to one another and the decomposition is not possible anymore.

As explained in [13, 14], this difference is also evident from the cycles of the factor-graph representation of the problem, depicted in Fig. 4. Note that, for the simplified case, all row function nodes disappear except for the first one, removing all cycles. In that case, GDL-based algorithms are proven to converge in a single iteration and give the exact solution. On the other hand, when cycles do exist, GDL-algorithms do not return the exact solution in a single iteration. Nevertheless, they typically give a good approximation after a reasonable number of iterations (belief-propagation algorithms employed for, e.g., decoding LDPCs are classic examples of this behavior).

For the iterative, fully-constrained version of our algorithm, expression (6) for the column-wise marginals must be replaced by

$$\phi^{\mathrm{c}}_{m,l}(x_{m,l}) = \min_{\{x_{k,l}\}_{k\neq m}} \frac{1}{2}\|\mathbf{y}_l - \mathbf{S}\mathbf{x}_l\|_2^2 + \lambda\|\mathbf{x}_l\|_0 + \sum_{k\neq m} \phi^{\mathrm{r}}_{k,l}(x_{k,l}) \tag{7}$$

and should be computed for all entries $x_{m,l}$, as opposed to the first row only as in the simplified case (see also Section 3.3 below). In this new expression we have introduced the *row-wise marginals* of the objective function, namely

$$\phi^{\mathrm{r}}_{m,l}(x_{m,l}) = \min_{\{x_{m,j}\}_{j\neq l}} \sum_{j\neq l} \phi^{\mathrm{c}}_{m,j}(x_{m,j}) \tag{8}$$

subject to $x_{m,1}, x_{m,2}, \ldots, x_{m,L}$ corresponding to a feasible sequence of hidden states. It is not difficult to prove that these functions can be obtained as a side result of the Viterbi algorithm presented in the previous section. Also, they only take two values: $\phi^{\mathrm{r}}_{m,l}(0)$ when $x_{m,l} = 0$ and $\phi^{\mathrm{r}}_{m,l}(*)$ when $x_{m,l} \neq 0$.

The resulting algorithm, which iterates between (7) and (8) until an exiting condition is met, is reported in Algorithm 1.

Algorithm 1 GDL-based algorithm

1: $\phi^{\mathrm{r}}_{m,l}(x_{m,l}) \leftarrow 0$ for all m, l
2: **repeat**
3: compute $\phi^{\mathrm{c}}_{m,l}(x_{m,l})$ according to (7) for all m, l
4: compute $\phi^{\mathrm{r}}_{m,l}(x_{m,l})$ according to (8) for all m, l, with $(x_{m,1}, x_{m,2}, \ldots, x_{m,L})$ feasible sequences
5: **until** an exit condition is met.

3.3. Column-wise Minimization

The algorithm presented in the previous section requires the computation of the column-wise marginals $\phi^{\mathrm{c}}_{m,l}(x_{m,l})$ according to (7). Nevertheless, this is a combinatorial problem and its solution is not trivial. A possible approach is outlined below.

To start with, recall that the row marginals, $\phi^{\mathrm{r}}_{m,l}(\cdot)$, are on–off functions, meaning that they can take only two values: $\phi^{\mathrm{r}}_{m,l}(0)$ when $x_{m,l} = 0$ and $\phi^{\mathrm{r}}_{m,l}(*)$ otherwise. Then, (7) can be approximated by the modified OMP algorithm in Algorithm 2. In the algorithm, $\mathbf{s}_j$ is the jth (normalized) column of $\mathbf{S}$ and $\mathbf{S}_{\Omega_i}$ denotes the submatrix of $\mathbf{S}$ with columns indexed by Ω_i, the parameter support at step i. Moreover, $\mathbf{S}^{\dagger}_{\Omega_i}$ stands for its Moore–Penrose pseudoinverse.

Algorithm 2 Modified OMP

1: $\boldsymbol{\rho}_0 \leftarrow \mathbf{y}_l$, $i \leftarrow 0$, $\Omega_0 \leftarrow \emptyset$
2: **repeat**
3: $i \leftarrow i + 1$
4: **for all** $j \notin \Omega_i$ **do**
5: $\gamma_j \leftarrow \frac{1}{2}(\mathbf{s}_j^{\mathrm{T}}\boldsymbol{\rho}_{i-1})^2 + \phi^{\mathrm{r}}_{j,l}(0) - \phi^{\mathrm{r}}_{j,l}(*) - \lambda$
6: **end for**
7: $k_i \leftarrow \arg\max_j \gamma_j$
8: $\Omega_i \leftarrow \Omega_{i-1} \cup \{k_i\}$
9: $\boldsymbol{\rho}_i \leftarrow (\mathbf{I}_N - \mathbf{S}_{\Omega_i}\mathbf{S}^{\dagger}_{\Omega_i})\mathbf{y}_l$
10: **until** $\gamma_{k_i} < 0$

Recall that we are not interested in the full characterization of $\phi^{\mathrm{c}}_{m,l}(x_{m,l})$ for all $x_{m,l} \in \mathbb{R}$, but only in $\phi^{\mathrm{c}}_{m,l}(0)$ and $\phi^{\mathrm{c}}_{m,l}(*) = \min_{x_{m,l}\neq 0} \phi^{\mathrm{c}}_{m,l}(x_{m,l})$. The first value can be computed with Algorithm 2 by forcing $\phi^{\mathrm{r}}_{m,l}(*) = +\infty$, which prevents the algorithm from selecting index m in step 7. Similarly, by setting $\phi^{\mathrm{r}}_{m,l}(0) =$

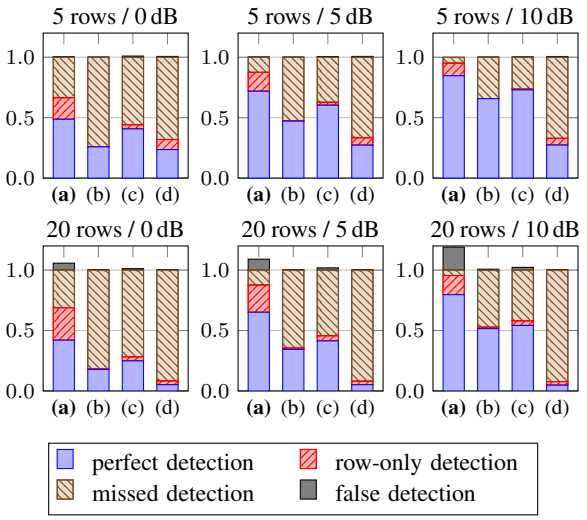

Fig. 5. Performance comparison for different detection algorithms: **(a)** the GDL-based algorithm presented in this paper, (b) algorithm based on (1), (c) algorithm based on (3), (d) RA-ORMP MMV algorithm [10]. For the last three cases, the algorithm is followed by an hard-decision step that forces to zero all row elements other than the r ones with the highest magnitude. The number of active rows is fixed to either 5 (top) or 20 (bottom) and the SNR takes the values (from left to right) 0 dB, 5 dB and 10 dB.

$+\infty$, we ensure that index m is chosen in step 7 at the very first iteration.

4. NUMERICAL RESULTS

We give here an assessment of the above results by numerical simulations. The entries of the sampling matrix $\mathbf{S}$, with dimensions $N = 40$ and $M = 200$, are generated as independent and identically distributed Gaussian random variables with zero mean and unitary variance. The number of observations is $L = 10$. The parameter matrix $\mathbf{X}^* \in \mathbb{R}^{M\times L}$ has a fixed number of active rows (either 5 or 20) and, for each one of them, only $r = 2$ randomly selected entries are generated as, again, zero mean unitary variance random variables. All other elements of $\mathbf{X}^*$ are set to zero. The additive white Gaussian noise has variance $1/\text{SNR}$, with $\text{SNR} \in \{0, 5, 10\}$ dB being the signal-to-noise ratio.

In Fig. 5, the proposed algorithm [labeled **(a)**] is compared to three of the methods mentioned in Section 2.1, namely:

(b) a classic CS column-by-column solution of (1);

(c) the solution to the $\ell_{2,1}/\ell_1$-regularized minimization problem (3);

(d) the RA-ORMP algorithm for row-sparse MMV problems proposed in [10].

For all these three options, row constraints (2) are enforced only after the algorithm has returned an unconstrained solution. Specifically, the r entries with the highest magnitude in each row are kept active while all the others are set to zero. Note that, for all the reported cases (that is, for all SNR values and number of active rows) and for algorithms (b)–(d), the regularizer parameters λ and μ have been tuned by extensive simulations to minimize the number of errors in the recovery of the support of $\mathbf{X}^*$. On the other hand, for the proposed algorithm **(a)**, we used the same parameter λ as for the single-column algorithm (b) since, from the discussion in the previous section, **(a)** can be considered an improvement on (b) and column optimizations are indeed carried out in the first step of Algorithm 1.

Performances are measured in terms of perfect row detection (the algorithm correctly selects an active row as well as its r active elements), row-only detection (the algorithm returns a wrong r-tuple of active elements in an actually active row), missed detection and false detection of active rows. More precisely, the histograms in Fig. 5 show the normalized count of the four events out of one thousand runs. As one can readily see, the proposed algorithm outperforms (by up to 25%) all other options in terms of perfect detection, since it is the only one that inherently exploits the specified row structure. The missed detection probability is also significantly reduced. The only drawback is an increase in the false detection probability when the number of active rows of $\mathbf{X}^*$ is large (equal to 20 in our examples), but this can be probably corrected by specifically tuning the regularizer parameter λ (as opposed to using the one from (b), as explained above).

As for the benchmark algorithms, (b) and (c) show similar performances, with the latter being slightly better. This was expected since model (3) is somehow more representative of the parameter structure than model (1). Finally, algorithm (d) is the one with the poorest results: Since the number of active entries per active row is low (2 out of 10), the energy of active and inactive rows is probably too similar for algorithms like the RA-ORMP to work properly.

5. FINAL REMARKS

In this paper we have presented a GDL-inspired algorithm that tackles MMV regression problems where the unknown parameter matrix presents few active rows. More importantly, zero and nonzero entries of each active row follow characteristic patterns. Indeed, even if we deal only with the simple case where the number of active elements per active row is fixed and equal to r, it is easy to see that the same approach can be extended to more complex structures, as long as they can be expressed by a succession of hidden states (see Section 3).

Our simulation results show a considerable performance gain over other solutions that rely on a relaxation of the row pattern constraint. This is especially important in applications that require the estimated parameter matrix to be compliant with the sparsity structure, regardless of the extra complexity introduced by an iterative, message-passing algorithm like the one presented here.

6. REFERENCES

[1] Trevor Hastie, Robert Tibshirani, and Martin Wainwright, *Statistical Learning with Sparsity: The Lasso and Generalizations*, Monographs on Statistics and Applied Probability. CRC Press, 2015.

[2] Francis Bach, Rodolphe Jenatton, Julien Mairal, and Guillaume Obozinski, "Optimization with sparsity-inducing penalties," *Found. Trends Mach. Learn.*, vol. 4, no. 1, pp. 1–106, Aug. 2012.

[3] Alexandre Gramfort, Daniel Strohmeier, Jens Haueisen, Matti S. Hämäläinen, and Matthieu Kowalski, "Time-frequency mixed-norm estimates: Sparse M/EEG imaging with non-stationary source activations," *Neuroimage*, vol. 70, pp. 410–422, Apr. 2013.

[4] Michael Lustig, David L. Donoho, Juan M. Santos, and John M. Pauly, "Compressed sensing MRI," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 72–82, Mar. 2008.

[5] Lukas Meier, Sara van der Geer, and Peter Bühlmann, "The group lasso for logistic regression," *J. R. Statist. Soc. B*, vol. 70, no. 1, pp. 53–71, Jan. 2008.

[6] Junzhou Huang, Tong Zhang, and Dimitris Metaxas, "Learning with structured sparsity," *J. Mach. Learn. Res.*, vol. 12, pp. 3371–3412, Nov. 2011.

[7] Rodolphe Jenatton, Jean-Yves Audibert, and Francis Bach, "Structured variable selection with sparsity-inducing norms," *J. Mach. Learn. Res.*, vol. 12, pp. 2777–2824, July 2011.

[8] Emmanuel J. Candès and Michael B. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, Mar. 2008.

[9] Mònica Navarro, Evangelos Kosmatos, David Gregoratti, Adriano Pastore, Stephan Pfletschinger, and Panagiotis Demestichas, "PLNC decoding: Enabler for massive MTC in 5G networks," in *Proc. ISWCS*, Aug. 2018.

[10] Mike E. Davies and Yonina C. Eldar, "Rank awarness in joint sparse recovery," *IEEE Trans. Inf. Theory*, vol. 58, no. 2, pp. 1135–1146, Feb. 2012.

[11] Rodolphe Jenatton, Julien Mairal, Guillaume Obozinski, and Francis Bach, "Proximal methods for hierarchical sparse coding," *J. Mach. Learn. Res.*, vol. 12, pp. 2297–2334, July 2011.

[12] Richard G. Baraniuk, Volkan Cevher, Marco F. Duarte, and Chinmay Hegde, "Model-based compressive sensing," *IEEE Trans. Inf. Theory*, vol. 56, no. 4, pp. 1982–2001, Apr. 2010.

[13] Srinivas M. Aji and Robert J. McEliece, "The generalized distributive law," *IEEE Trans. Inf. Theory*, vol. 46, no. 2, pp. 325–343, Mar. 2000.

[14] Frank R. Kschischang, Brendan J. Frey, and Hans-Andrea Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.