Exact and Heuristic Methods for Simultaneous Sparse Coding

Alexandra Dache, Arnaud Vandaele, Nicolas Gillis
Université de Mons
Mons, Belgium
{firstname.lastname}@umons.ac.be

Nicolas Nadisic
Ghent University
Ghent, Belgium
nicolas.nadisic@ugent.be

***Abstract*—The simultaneous sparse coding (SSC) problem consists in approximating several data points as linear combinations of the same few basis elements selected within a given dictionary. It is key in many applications of machine learning and signal processing. Solving SSC up to global optimality has never been explored in the literature, to the best of our knowledge. In this paper, we propose a reformulation of SSC into a mixed-integer quadratic program (MIQP) solvable globally by generic solvers. We also consider a variant of SSC with nonnegativity constraints. We show experimentally that the global resolution can be applied in practice to medium-scale problems. For larger-scale problems, we propose a hybrid method that first uses a heuristic as a preprocessing to reduce the size of the original problem, and then solves the reduced problem exactly. Empirically, we show that our hybrid method outperforms existing heuristics on both synthetic data and in the unmixing of real-world hyperspectral images.**

***Index Terms*—Simultaneous sparse coding, multiple measurement vectors, mixed-integer reformulation, nonnegativity.**

I. Introduction and related work

Given a matrix $X \in \mathbb{R}^{m\times n}$, a dictionary $D \in \mathbb{R}^{m\times s}$, and a sparsity target $r \in \mathbb{N}$, the simultaneous sparse coding (SSC) problem consists in finding $H \in \mathbb{R}^{s\times n}$ with at most r non-zero rows such that $X \approx DH$. Formally, using the squared Frobenius norm as a data fidelity measure, SSC corresponds to the following optimization problem:

$$\min_H \|X - DH\|_F^2 \quad \text{s.t.} \quad \|H\|_{row-0} \leq r, \tag{1}$$

where the row-0 "norm" $\|H\|_{row-0} = |\{i|H(i,:) \neq 0\}|$ denotes the number of non-zero rows in H. In other words, we are looking for a matrix H such that all its columns are r-sparse (that is, they have at most r non-zero entries) and share the same support, that is, the set of the indices of non-zero entries. Note that (1) is equivalent to finding a subset J of columns of D such that $|J| \leq r$ and $X \approx D(:,J)\hat{H}$ for some matrix $\hat{H}$. SSC is also known by several other names, such as simultaneous sparse approximation (SCA) [1], multiple measurement vectors (MMV) [2], and joint sparse coding (JSC) [3].

This work was supported by the Fonds de la Recherche Scientifique - FNRS (F.R.S.-FNRS) and the Fonds Wetenschappelijk Onderzoek - Vlaanderen (FWO) under EOS Project no O005318F-RG47, by the Francqui foundation, and by the European Research Council (ERC consolidator grant no 101085607). NN is with Ghent University at the time of submission, but he contributed to this work while he was with the University of Mons.

SSC arises when the data points, the columns of X, can be expressed as different linear combinations of a few basis elements selected among an overcomplete dictionary, the columns of D. Applications in signal processing are very diverse and include compressed sensing, source separation, source localization; see for example [1] and the references therein. In this work, we focus on hyperspectral unmixing, see section IV-B for details.

The row-0 constraint makes SSC a combinatorial problem, with $\binom{s}{r}$ different possible supports, therefore it is hard to solve in large dimensions. For this reason, existing approaches for SSC are mostly heuristic algorithms, that are computationally fast but not guaranteed in general to recover an optimal solution. The most popular ones rely on greedy algorithms [4]–[6], or on convex relaxations [7], [8].

A popular variant of SSC considers a nonnegativity constraint on H. Formally, it reduces to the following problem, that we coin as NSSC:

$$\min_H \|X - DH\|_F^2 \quad \text{s.t.} \quad \begin{cases} \|H\|_{row-0} \leq r, \\ H \geq 0, \end{cases} \tag{2}$$

where $H \geq 0$ means H is entry-wise nonnegative. The nonnegativity constraint is natural in many applications involving physical values, so taking it explicitly into account generally improves the recovery and produces more interpretable solutions. Nonnegativity is also known to improve the regularity of ill-posed problems [9]. As opposed to SSC which has been studied extensively, only a few methods were proposed to tackle NSSC.

All existing methods for (N)SSC rely on heuristics, but having guarantees on the quality of the solution is important in some applications. Therefore global optimization may be desirable even if it requires more computing time. Also, by solving globally the optimization problem, the analyst knows that the possible error comes from either model misfit or acquisition noise, and not from the solver.

Contribution and outline of the paper. Our first contribution, in section II, is a reformulation of SSC into a mixed-integer quadratic program (MIQP), including an optional nonnegativity constraint. We show that medium-sized problems can be solved globally using a generic solver. In section III, we introduce a hybrid method that first uses a heuristic as a preprocess to reduce the size of the original problem, and

then solves exactly the reduced problem. We show empirically in section IV that our strategy outperforms state-of-the-art heuristics while being computationally efficient, with experiments on both synthetic data and the unmixing of real-world hyperspectral images.

II. MIQP Reformulation

The optimization of problems including both continuous and integer variables is known as mixed-integer programming (MIP). The generic solving of MIP is an active area of research [10], and existing solvers can tackle efficiently problems with linear or quadratic objectives, under linear or quadratic constraints. Therefore, in this section we reformulate the SSC problem (1) as a MIP with a quadratic objective (MIQP) so it can be solved with generic solvers.

In SSC (1), the row-0 constraint is combinatorial but it can be rewritten using auxiliary binary variables, as done for example in [11] for the ℓ_0 "norm" of a vector. Let the vector $y \in \{0,1\}^s$ represent the sparsity of the rows of H, that is,

$$y_i = 0 \Leftrightarrow H(i,:) = 0 \text{ for all } i. \tag{3}$$

The row-0 "norm" can therefore be written as a simple sum, $\|H\|_{row-0} = \sum_i y_i$, and the constraint in (1) can be rewritten using the following linear constraint

$$\sum_i y_i \leq r.$$

Using a constant traditionally called "big M" to bound the optimized variables, the constraint (3) can be rewritten as

$$-My_i \leq H(i,j) \leq My_i \text{ for all } i,j.$$

Choosing an appropriate value for M is nontrivial and usually depends on application knowledge. The smaller the M, the more restricted the feasible range of the entries of H, and the faster the solving. However, the value of M must be larger than the largest entry of the optimal (unknown) H so that the solution is in the admissible domain.

The SSC problem (1) can therefore be reformulated, with a quadratic objective and linear constraints, as the following MIQP:

$$\begin{aligned} \min_{H,y} & \sum_j H(:,j)^T D^T DH(:,j) - 2X(:,j)^T DH(:,j) \\ \text{s.t.} & \begin{cases} -My_i \leq H(i,j) \leq My_i \text{ for all } i,j, \\ \sum_i y_i \leq r, \end{cases} \end{aligned} \tag{4}$$

where y is the binary auxiliary variable defined in (3). For the objective, we simply used the fact that

$$\|X - DH\|_F^2 = \|X\|_F^2 - 2\,\mathrm{tr}(X^\top DH) + \mathrm{tr}(H^\top D^\top DH),$$

where $\mathrm{tr}(.)$ is the trace of a matrix (sum of its diagonal elements). The non-negative variant (2) is easily obtained by replacing the left-hand side of the constraint on the $H(i,j)$'s with 0. The parameter r is set by the user and its corresponds to the number of columns of the dictionary one wants to select.

Thanks to this formulation (4), the problem can be solved exactly with generic solvers. We have created a Matlab script that builds this model and calls the generic solver Gurobi[1] to obtain a globally optimal solution. This exact method is quite expensive computationally, but it can be used to tackle moderate-size problems. An experiment on synthetic data shows the evolution of the resolution time as a function of the dimensions s and n with $r = 4$. On Figure 1, we display the resolution time of the model for synthetic data with $n = 2s$. This exact method can tackle in under 15 minutes a SSC problem with $s = 60$, $n = 120$, $r = 4$, and a relatively well-conditioned matrix D (see section IV for details on the experimental setup).

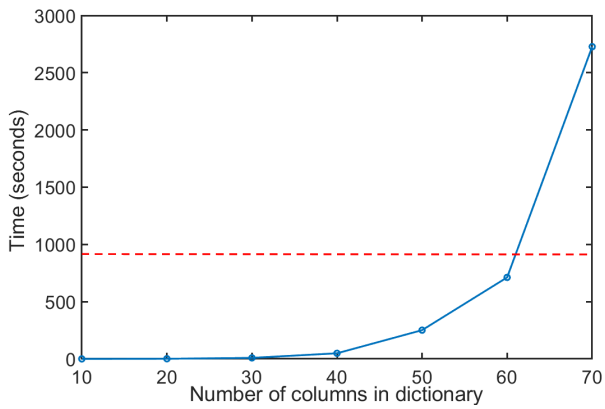

Fig. 1. Evolution of time as a function of s, the number of columns in the dictionary (we fixed $r = 4$ and the number of columns of X is $n = 2s$). The red dotted line corresponds to a computing time of 15 minutes.

Unfortunately, because of this exponentially increasing computing time, this exact approach is not suitable for larger-scale problems. For this purpose, we introduce a hybrid method in the next section, based on both heuristics and exact solving.

III. Hybrid method

The computational complexity of solving SSC exactly via MIQP makes it impractical for large real-world problems, which is why most existing works rely on approximate but fast heuristics. By definition, heuristics are not guaranteed in general to find the exact solution of a given SSC problem, that is, to identify the right support J of nonzero rows of H. However, they may identify correctly some rows belonging to the support. In particular, when solving a SSC problem with a given sparsity target r, running a heuristic with a larger sparsity target, $r' > r$, may well identify a set of rows that contains, among others, the rows belonging to the correct support. Therefore we introduce a two-step hybrid method: First, it computes a support of cardinality r' using a fast heuristic. This reduces the original problem to a subproblem restricted to this support, that is, s is reduced to r' in (1). Second, it computes a support of cardinality r by solving exactly this smaller subproblem using MIQP. This method is described in Algorithm 1.

[1]Although it is a closed-source commercial software, see https://www.gurobi.com/, free use is possible for academic users.

Algorithm 1: Hybrid method for SSC.

Input: Input matrix $X \in \mathbb{R}^{m \times n}$, dictionary $D \in \mathbb{R}^{m \times s}$, factorization rank $r \in \mathbb{N}$, intermediate rank $r' \in \mathbb{N}$ with $r' > r$, a heuristic algorithm for SSC.

Output: Matrix $H^* \in \mathbb{R}^{s \times n}$ such that $X \approx DH^*$ and $\|H^*\|_{row-0} \leq r$

1 $H' \leftarrow$ heuristic(D,X,r')

2 $J' \leftarrow \{i | H'(i,:) \neq 0\}$

3 $H^* \leftarrow \text{argmin}_{H, \|H\|_{row-0} \leq r} \|X - D(:,J')H\|_F^2$

On line 1, we solve approximately the original SSC problem using a given heuristic, but with a sparsity target $r' > r$. Note that any existing heuristic for SSC can be used at this step. The result is an intermediary matrix H' with a row-sparsity of r' and a row-support J'. This step can be seen as a preprocessing of the dictionary, to reduce it to r' columns. On line 3, we build a smaller subproblem by considering only the columns of D indexed by J', and we solve this subproblem exactly using the MIQP reformulation introduced in section II.

The choice of the parameters is quite intuitive. The parameter r corresponds to the number of basis elements or features one wants to extract from the dictionary, and it is often given naturally by the application at hand. For example, in hyperspectral unmixing, r is the number of materials we want to identify in an image; see section IV-B for details. On the other hand, r' should be as large as possible to increase the chances of extracting the best columns of the dictionary during the preprocessing step, and as small as needed to ensure a reasonable computing time.

In terms of complexity, the MIQP problem on line 3 of Algorithm 1 has $\binom{r'}{r}$ possible solutions, which is significantly smaller than the $\binom{s}{r}$ feasible solutions of the original problem (1) when $s \gg r'$. Note that each feasible solution requires solving a convex quadratic optimization problem in rn variables (corresponding to the r selected non-zero rows of H) which can be done, for example, in $\mathcal{O}(mnr)$ operations with a first-order method. In practice, the generic solvers use pruning techniques such as branch-and-bound and generally evaluate only a fraction of the feasible solutions. Therefore the computational cost is generally far from this worst case.

Although this hybrid method is not guaranteed in general to identify the right support, it is quite efficient in practice, as we show in section IV.

IV. Experiments

Several experiments are now carried out to illustrate the capacity of our hybrid method to produce more accurate solutions. We considered two greedy heuristic methods: S-SP and S-OMP from [5]. We compare heuristic methods with their hybrid versions, that is, using a heuristic combined with exact solving as in Algorithm 1 on both synthetic and real-world data.

All experiments were performed on a personal computer with an i7 processor with a clock frequency of 2.80GHz. To make our experiments easily reproducible, we provide the code and test scripts in an online repository[2].

A. Synthetic data sets

In this section, we study the performance of our hybrid method on synthetic data, when noise varies.

We generate synthetic data sets as follows. The dictionary D is constructed by selecting randomly s columns of the USGS hyperspectral library[3]. The set J of r columns of the dictionary D is then chosen at random. The entries in the rows of H indexed by J are drawn randomly according to a normal distribution, and the other rows are set to zero. The matrix X is then constructed as $X = DH$. A Gaussian noise matrix N is added to X, we define the noise level of noisy $X_n = X + N$ as $\frac{\|D(:,J)H(J,:)\|_F}{\|N\|_F}$. Then, we solve the resulting SSC problem with noisy data, with the original heuristics S-SP and S-OMP and with our hybrid method (Algorithm 1) using both heuristics in step 1.

The parameters are set to $s = 100$, $n = 100$, $r = 3$, and for the hybrid methods $r' = 30$. The experiment is repeated 10 times for each of the 10 noise levels between 10^{-3} and $10^{-0.5}$, so each point of the following plots is an average over 10 random data sets. We plot for each method and as a function of the noise level:

- the computing time on fig. 2;
- the column recovery rate on fig. 3, that is, the percentage of columns of the original support in the generated data that are correctly recovered in the computed solution;
- the relative reconstruction error on fig. 4, that is $\frac{\|X - DH^*\|_F}{\|X\|_F}$ with H^* the computed row-sparse solution.

The results on fig. 3 show that, even with very little noise, the heuristic methods fail to identify the correct columns. They have a recovery rate between 40 and 60%. On the other hand, both variants of our hybrid method have a recovery rate above 80%. It diminishes for a noise higher than 5% but in all cases they outperform the corresponding heuristics. On fig. 4, we see that for small noise levels the error of the solution computed by our hybrid method is close to 0, meaning the reconstruction is almost perfect. Then, the error grows slower than linear with the noise level. On the contrary, the solutions given by the heuristics are always above 40% of error, which represent a mostly failed reconstruction. The trade-off for the better performance of our hybrid method is a relatively slow speed. We see on fig. 2 that although the computing time of the hybrid method increases very little with the noise level, it is always significantly higher than the heuristics.

B. Real-world hyperspectral unmixing

A well-known application of nonnegative SSC is hyperspectral unmixing (HU), that we describe briefly here; see for example [12] for more details. A hyperspectral image is an

[2] https://gitlab.com/Alexia1305/SSC

[3] https://www.usgs.gov

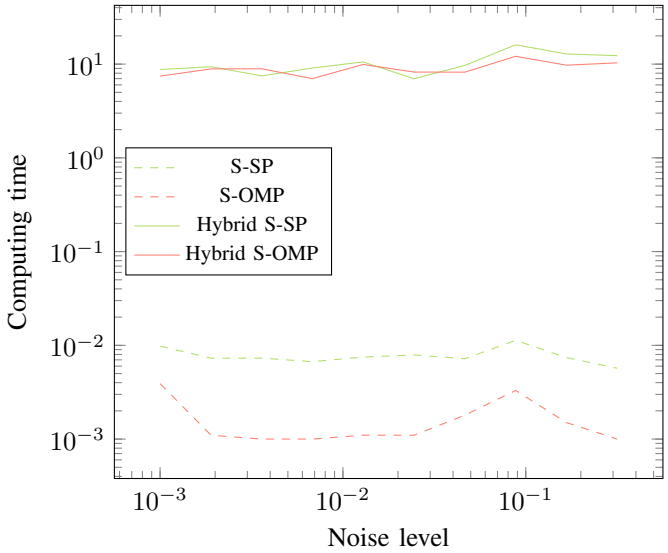

Fig. 2. Evolution of the computing time in seconds for different levels of noise. Plotted values are the average over 10 random data sets. Axis are in log scale.

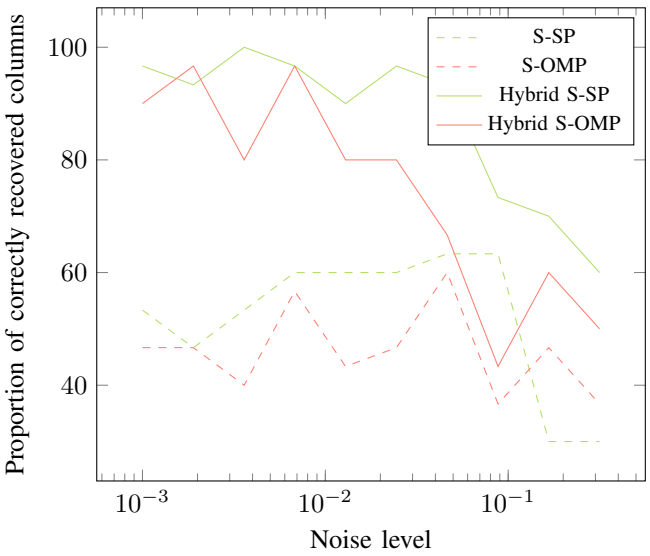

Fig. 3. Evolution of the proportion of correctly recovered columns in percents for different levels of noise. Plotted values are the average over 10 random data sets. The x-axis is in log scale.

image of scene acquired within many narrow spectral bands, usually a few hundreds. Therefore for each pixel we have a precise electromagnetic spectrum, that gives information about the materials present in the pixels. Following the linear mixing model, the spectrum of a given pixel is the additive linear combination of the material it contains. The goal of hyperspectral unmixing is to decompose the hyperspectral image into a collection of constituent spectral signatures (called endmembers) and into a set of corresponding abundances.

If X is the data matrix where each column corresponds to the spectrum of a pixel of the hyperspectral image, and D a dictionary whose columns correspond to the spectra of

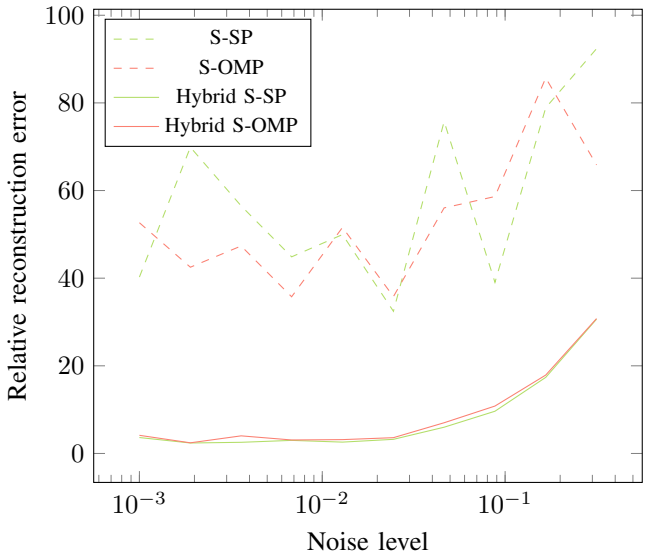

Fig. 4. Evolution of the relative reconstruction error in percents for different levels of noise. Plotted values are the average over 10 random data sets. The x-axis is in log scale.

some known materials, then the entries of H represent the abundance of each material in each pixel. The nonnegativity constraint is natural as abundances are physical nonnegative quantities, the NSSC model (2) is therefore equivalent to solving hyperspectral unmixing.

A common assumption in hyperspectral unmixing is the so-called *pure-pixel assumption*, stating that for each material present in the image, there is at least one pixel containing only this material. It is usually verified in real-world hyperspectral images when the spatial resolution is good enough. Using this assumption, we can use the input data X itself as a self-dictionary, that is, use the dictionary $D = X$. The problem is then equivalent to nonnegative matrix factorization under the separability assumption [13].

In this experiment, we perform NSSC with self-dictionary using our hybrid method (Algorithm 1) on real-world hyperspectral images. This is equivalent to selecting a subset $X(:, J')$ of r' columns of the input data with a heuristic, and then performing exact NSSC to extract a set of r columns from this subset of pixels $X(:, J')$. We use four real-world hyperspectral images [14], see Table I for details. As a

TABLE I
SUMMARY OF THE HYPERSPECTRAL IMAGES STUDIED IN THIS WORK.

Data set	m	n	r	r'
San Diego	158	$400 \times 400 = 160000$	8	80
Urban	162	$307 \times 307 = 94249$	6	60
Terrain	166	$500 \times 307 = 153500$	5	50
Samson	156	$95 \times 95 = 9025$	3	30

pre-processing heuristic, we use here the clustering-based algorithm H2NMF [15]. Note that the size of the problems considered make impossible to solve them directly with an MIQP solver. We compare our results with two standard

methods:

- FGNSR [16], an algorithm based on convex relaxation which deals specifically with self-dictionary NSSC,
- NMFdico [6], [17], a greedy algorithm for NSSC.

The pre-processing with H2NMF is also used for FGNSR and NMFdico. After selecting the subset $X(:, J)$ of r columns of X, the computation of matrix $\hat{H}$ is a standard nonnegative least squares (NNLS) problem and we solve it using a block coordinate descent scheme [18], as in [16]. To evaluate the computed solutions without knowing the ground-truth solution, we measure the relative reconstruction error $\frac{\|X - X(:,J)\hat{H}\|_F}{\|X\|_F}$ with X the original input matrix, and J the index set of the r selected columns. The parameter r corresponds to the number of material we expect to find in the image, and we choose it as in the literature, see for example [14]. We set the parameter $r' = 10r$.

TABLE II
RESULTS OF THE UNMIXING OF REAL-WORLD HYPERSPECTRAL IMAGES. TIME IS IS SECONDS, ERROR IS THE RELATIVE RECONSTRUCTION ERROR IN PERCENTS. BOLD NUMBERS CORRESPOND TO OUR HYBRID METHOD (ALGORITHM 1) USING H2NMF AS A PRE-PROCESSING HEURISTIC.

	Data	San Diego	Urban	Terrain	Samson
	r	8	6	5	3
	r'	80	60	50	30
Time	FGNRS	0.04	0.03	0.03	0.01
	NMFdico	0.01	0.01	0.02	0.01
	Ours (Alg. 1)	**83.6**	**7.76**	**1.21**	**0.64**
Error	FGNRS	9.21	6.03	3.73	3.48
	NMFdico	9.05	6.03	3.52	3.2
	Ours (Alg. 1)	**8.35**	**4.27**	**3.32**	**3.06**

Table II shows the results of this experiment. In terms of relative reconstruction error, our hybrid method outperforms competing algorithms for all data sets. The difference is especially important for larger data sets with larger r, in San Diego and Urban. The trade-off is a computing time that is significantly larger and seems to grow exponentially with r. However, our method still carries out the unmixing of large real-world images in a few minutes or seconds using a personal computer. The acquisition of hyperspectral images is a complex and time-consuming task, and hyperspectral unmixing is usually a one-time operation, so our method is useful when the user can spend a little more time to obtain a better quality unmixing.

To further accelerate the method, we could use a time limit for the MIQP solver, that is, stop the solver after a fixed amount of time and return the best solution found so far. Indeed, in many practical cases the solver finds the optimal solution quite fast, and then spends most of the computing time to guarantee the optimality; see [19] for a numerical example. With early stopping, we lose the optimality guarantee but we can have a good solution in only a fraction of the time.

V. CONCLUSION

In this paper, we studied the simultaneous sparse coding problem and its nonnegative variant. We introduced an approach to solve it up to global optimality using a MIQP reformulation, and showed that it can effectively handle medium-sized problems. To be able to tackle larger-scale problems, we introduced a hybrid method, that first uses a heuristic to pre-select columns of the dictionary to form a smaller intermediary problem and then solves this smaller problem globally with the MIQP reformulation. We showed on both synthetic data and real-world hyperspectral unmixing tasks that our hybrid method outperforms existing algorithms on large-scale problems, at the cost of a higher computing time.

REFERENCES

[1] A. Rakotomamonjy, "Surveying and comparing simultaneous sparse approximation (or group-lasso) algorithms," *Signal processing*, vol. 91, no. 7, pp. 1505–1526, 2011.

[2] S. F. Cotter, B. D. Rao, K. Engan, and K. Kreutz-Delgado, "Sparse solutions to linear inverse problems with multiple measurement vectors," *IEEE Transactions on Signal Processing*, vol. 53, no. 7, pp. 2477–2488, 2005.

[3] X. Zheng, X. Sun, K. Fu, and H. Wang, "Automatic annotation of satellite images via multifeature joint sparse coding with spatial relation constraint," *IEEE Geoscience and Remote Sensing Letters*, vol. 10, no. 4, pp. 652–656, 2012.

[4] J. A. Tropp, A. C. Gilbert, and M. J. Strauss, "Algorithms for simultaneous sparse approximation. part i: Greedy pursuit," *Signal processing*, vol. 86, no. 3, pp. 572–588, 2006.

[5] D. Kim and J. P. Haldar, "Greedy algorithms for nonnegativity-constrained simultaneous sparse recovery," *Signal processing*, vol. 125, pp. 274–289, 2016.

[6] J. E. Cohen and N. Gillis, "A new approach to dictionary-based nonnegative matrix factorization," in *EUSIPCO*, 2017, pp. 493–497.

[7] J. A. Tropp, "Algorithms for simultaneous sparse approximation. part ii: Convex relaxation," *Signal Processing*, vol. 86, no. 3, pp. 589–602, 2006.

[8] L. Belmerhnia, E.-H. Djermoune, C. Carteret, and D. Brie, "Simultaneous variable selection for the classification of near infrared spectra," *Chemometr. Intell. Lab. Syst.*, vol. 211, pp. 104268, 2021.

[9] A. M. Bruckstein, M. Elad, and M. Zibulevsky, "On the uniqueness of nonnegative sparse solutions to underdetermined systems of equations," *IEEE Transactions on Information Theory*, vol. 54, no. 11, pp. 4813–4820, 2008.

[10] R. E. Bixby, "A brief history of linear and mixed-integer programming computation," *Documenta Mathematica*, pp. 107–121, 2012.

[11] S. Bourguignon, J. Ninin, H. Carfantan, and M. Mongeau, "Exact sparse approximation problems via mixed-integer programming: Formulations and computational performance," *IEEE Transactions on Signal Processing*, vol. 64, no. 6, pp. 1405–1419, 2015.

[12] J. M. Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, Q. Du, P. Gader, and J. Chanussot, "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 5, no. 2, pp. 354–379, 2012.

[13] S. Arora, R. Ge, R. Kannan, and A. Moitra, "Computing a nonnegative matrix factorization–provably," in *ACM Symposium on Theory of Computing*, 2012, pp. 145–162.

[14] F. Zhu, "Hyperspectral unmixing: Ground truth labeling, datasets, benchmark performances and survey," *preprint arXiv:1708.05125*, 2017.

[15] N. Gillis, D. Kuang, and H. Park, "Hierarchical clustering of hyperspectral images using rank-two nonnegative matrix factorization," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 4, pp. 2066–2078, 2015.

[16] N. Gillis and R. Luce, "A fast gradient method for nonnegative sparse regression with self-dictionary," *IEEE Transactions on Image Processing*, vol. 27, no. 1, pp. 24–37, 2018.

[17] J. E. Cohen and N. Gillis, "Dictionary-based tensor canonical polyadic decomposition," *IEEE Transactions on Signal Processing*, vol. 66, no. 7, pp. 1876–1889, 2017.

[18] N. Gillis and F. Glineur, "Accelerated multiplicative updates and hierarchical als algorithms for nonnegative matrix factorization," *Neural computation*, vol. 24, no. 4, pp. 1085–1105, 2012.

[19] M. Abdolali and N. Gillis, "Simplex-structured matrix factorization: Sparsity-based identifiability and provably correct algorithms," *SIAM J. Math. Data Sci.*, vol. 3, no. 2, pp. 593–623, 2021.