Bayesian anti-sparse coding

Clment Elvira, Pierre Chainais and Nicolas Dobigeon

E-mail: {Clement.Elvira, Pierre.Chainais}@ec-lille.fr, Nicolas.Dobigeon@enseeiht.fr

TECHNICAL REPORT – 2015, December

Univ. Lille, CNRS, Centrale Lille, UMR 9189 - CRIStAL - Centre de Recherche en Informatique Signal et Automatique de Lille, F-59000 Lille, France

University of Toulouse, IRIT/INP-ENSEEIHT

2 rue Camichel, BP 7122, 31071 Toulouse cedex 7, France

Abstract

Sparse representations have proven their efficiency in solving a wide class of inverse problems encountered in signal and image processing. Conversely, enforcing the information to be spread uniformly over representation coefficients exhibits relevant properties in various applications such as digital communications. Anti-sparse regularization can be naturally expressed through an ℓ_∞-norm penalty. This paper derives a probabilistic formulation of such representations. A new probability distribution, referred to as the democratic prior, is first introduced. Its main properties as well as three random variate generators for this distribution are derived. Then this probability distribution is used as a prior to promote anti-sparsity in a Gaussian linear inverse problem, yielding a fully Bayesian formulation of anti-sparse coding. Two Markov chain Monte Carlo (MCMC) algorithms are proposed to generate samples according to the posterior distribution. The first one is a standard Gibbs sampler. The second one uses Metropolis-Hastings moves that exploit the proximity mapping of the log-posterior distribution. These samples are used to approximate maximum a posteriori and minimum mean square error estimators of both parameters and hyperparameters. Simulations on synthetic data illustrate the performances of the two proposed samplers, for both complete and over-complete dictionaries. All results are compared to the recent deterministic variational FITRA algorithm.

This reports provides complementary results to the paper [1].

Part of this work has been funded thanks to the BNPSI ANR project no. ANR-13-BS-03-0006-01.

Index Terms

democratic distribution, anti-sparse representation, proximal operator, inverse problem.

I. Introduction

Sparse representations have been widely advocated for as an efficient tool to address various problems encountered in signal and image processing. As an archetypal example, they were the core concept underlying most of the lossy data compression schemes, exploiting compressibility properties of natural signals and images over appropriate bases. Sparse approximations, generally resulting from a *transform coding* process, lead for instance to the famous image, audio and video compression standards JPEG, MP3 and MPEG [2], [3]. More recently and partly motivated by the advent of both the compressive sensing and dictionary learning paradigms, sparsity has been intensively exploited to regularize (e.g., linear) ill-posed inverse problems. The ℓ_0-norm and the ℓ_1-norm as its convex relaxation are among the most popular sparsity promoting penalties. Following the ambivalent interpretation of penalized regression optimization [4], Bayesian inference naturally offers an alternative and flexible framework to derive estimators associated with sparse coding problems. For instance, it is well known that a straightforward Bayesian counterpart of the LASSO shrinkage operator [5] can be obtained by adopting a Laplace prior [6]. Designing other sparsity inducing priors has motivated numerous research works. They generally rely on hierarchical mixture models [7]–[10], heavy tail distributions [11]–[13] or Bernoulli-compound processes [14]–[16].

In contrast, the use of the ℓ_∞-norm within an objective criterion has remained somehow confidential in the signal processing literature. One may cite the minimax or Chebyshev approximation principle, whose practical implementation has been made possible thanks to the Remez exchange algorithm [17] and leads to a popular design method of finite impulse response digital filters [18], [19]. Besides, when combined with a set of linear equality constraints, minimizing a ℓ_∞-norm is referred to as the minimum-effort control problem in the optimal control framework [20], [21]. Much more recently, a similar problem has been addressed by Lyubarskii *et al.* in [22] where the *Kashin's representation* of a given vector over a tight frame is introduced as the expansion coefficients with the smallest possible dynamic range. Spreading the information over representation coefficients in the most uniform way is a desirable feature in various applicative

contexts, e.g., to design robust analog-to-digital conversion schemes [23], [24] or to reduce the peak-to-average power ratio (PAPR) in multi-carrier transmissions [25], [26]. Resorting to an uncertainty principle (UP), Lyubarskii *et al.* have also introduced several examples of frames yielding computable Kashin's representations, such as random orthogonal matrices, random subsampled discrete Fourier transform (DFT) matrices, and random sub-Gaussian matrices [22]. The properties of the alternate optimization problem, which consists of minimizing the maximum magnitude of the representation coefficients for an upper-bounded ℓ_2-reconstruction error, have been deeply investigated in [27], [28]. In these latest contributions, the optimal expansion is called the *democratic representation* and some bounds associated with archetypal matrices ensuring the UP are derived. In [29], the constrained signal representation problems considered in [22] and [28] is converted into their penalized counterpart. More precisely, the so-called *spread* or *anti-sparse representations* result from a variational optimization problem where the admissible range of the coefficients has been penalized through a ℓ_∞-norm

$$\min_{\mathbf{x}\in\mathbb{R}^N} \frac{1}{2\sigma^2} \|\mathbf{y}-\mathbf{H}\mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_\infty . \tag{1}$$

In (1), $\mathbf{H}$ defines the $M \times N$ representation matrix and σ^2 stands for the variance of the residual resulting from the approximation. Again, the anti-sparse property brought by the ℓ_∞-norm penalization enforces the information brought by the measurement vector $\mathbf{y}$ to be evenly spread over the representation coefficients in $\mathbf{x}$ with respect to the dictionary $\mathbf{H}$. It is worth noting that recent applications have capitalized on these latest theoretical and algorithmic advances, including approximate nearest neighbor search [30] and PAPR reduction [31].

Surprisingly, up to our knowledge, no probabilistic formulation of these democratic representations has been proposed in the literature. The present paper precisely attempts to fill this gap by deriving a Bayesian formulation of the anti-sparse coding problem (1) considered in [29]. Note that this objective differs from the contribution in [32] where a Bayesian estimator associated with an ℓ_∞-norm loss function has been introduced. Instead, we merely introduce a Bayesian counterpart of the variational problem (1). The main motivations for deriving the proposed Bayesian strategy for anti-sparse coding are threefold. Firstly, Bayesian inference is a flexible methodology that may allow other parameters and hyperparameters (e.g., residual variance σ^2, regularization parameters λ) to be jointly estimated with the parameter of interest $\mathbf{x}$. Secondly, through the choice of the considered Bayes risk, it permits to define a wide class

of estimators, beyond the traditional penalized maximum likelihood estimator resulting from the solution of (1). Finally, within this framework, Markov chain Monte Carlo algorithms can be conveniently designed to generate samples according to the posterior distribution. Contrary to deterministic optimization algorithms which provide only one point estimate, these samples can be subsequently used to build a comprehensive statistical description of the solution.

To this purpose, a new probability distribution as well as its main properties are introduced in Section II. In particular, we show that $p(\mathbf{x}) \propto \exp(-\lambda \|\mathbf{x}\|_\infty)$ properly defines a probability density function (pdf). In Section III, this so-called *democratic distribution* is used as a prior distribution in a linear Gaussian inverse problem, which provides a straightforward equivalent of the problem (1) under the maximum *a posteriori* paradigm. Moreover, exploiting relevant properties of the democratic distribution, this section describes two Markov chain Monte Carlo (MCMC) algorithms as alternatives to the deterministic solvers proposed in [28], [29]. The first one is a standard Gibbs sampler which sequentially generates samples according to the conditional distributions associated with the joint posterior distribution. The second MCMC algorithm relies on a proximal Monte Carlo step recently introduced in [33]. This step exploits the proximal operator associated to the logarithm of the target distribution to sample random vectors asymptotically distributed according to this non-smooth density. Section IV illustrates the performances of the proposed algorithms on numerical experiments. Concluding remarks are reported in Section V.

II. Democratic distribution

This section introduces the democratic distribution and the main properties related to its marginal and conditional distributions. Finally, two random variate generators are proposed.

A. Probability density function

Lemma 1. *Let $\mathbf{x} \in \mathbb{R}^N$ and $\lambda \in \mathbb{R}_+$. The integral of the function $\exp(-\lambda \|\mathbf{x}\|_\infty)$ over $\mathbb{R}^N$ is properly defined and the following equality holds*

$$\int_{\mathbb{R}^N} \exp(-\lambda \|\mathbf{x}\|_\infty)\, d\mathbf{x} = N! \left(\frac{2}{\lambda}\right)^N.$$

Proof. See Appendix A. □

TABLE I

LIST OF SYMBOLS.

Symbol	Description
N, n	Dimension, index of representation vector
M, m	Dimension, index of observed vector
$\mathbf{x}$, x_n	Representation vector, its n^{th} component
$\mathbf{y}$, y_m	Observation vector, its m^{th} component
$\mathbf{H}$	Coding matrix
$\mathbf{e}$	Additive noise vector
λ	Parameter of the democratic distribution
μ	Re-parametrization of λ such that $\lambda = N\mu$
$\mathcal{D}_N(\lambda)$	Democratic distribution of parameter λ over $\mathbb{R}^N$
$C_N(\lambda)$	Normalizing constant of the distribution $\mathcal{D}_N(\lambda)$
$\mathcal{K}_J$	A J-element subset $\{i_1 \dots i_J\}$ of $\{1, \dots, N\}$
$\mathcal{U}$, $\mathcal{G}$, $\mathcal{IG}$	Uniform, gamma and inverse gamma distributions
$d\mathcal{G}$	Double-sided gamma distribution
$\mathcal{N}_{\mathcal{I}}$	Truncated Gaussian distribution over $\mathcal{I}$
$\mathcal{C}_n$	Double convex cones partitioning $\mathbb{R}^N$
c_n, $\mathcal{I}_n$	Weights and intervals defining the conditional distribution $p\left(x_n \vert \mathbf{x}_{\setminus n}\right)$
g, g_1, g_2	Negative log-distribution ($g = g_1 + g_2$)
δ	Parameter of the proximity operator
ε_j, d_j, $\phi_\delta(\mathbf{x})$	Family of distinct values of $\vert\mathbf{x}\vert$, their respective multiplicity and family of local maxima of $\text{prox}^{\delta}_{\lambda\|\cdot\|_\infty}$
$q(\cdot\vert\cdot)$	Proposal distribution
$\omega_{in}, \mu_{in}, s_n^2, \mathcal{I}_{in}$	Weights, parameters and intervals defining the conditional distribution $p\left(x_n \vert \mathbf{x}_{\setminus n}, \mu, \sigma^2, \mathbf{y}\right)$

As a corollary of Lemma 1, the democratic distribution can be defined as follows.

Definition 1. *A N-real-valued random vector $\mathbf{x} \in \mathbb{R}^N$ is said to be distributed according to the democratic distribution $\mathcal{D}_N(\lambda)$, namely $\mathbf{x} \sim \mathcal{D}_N(\lambda)$, when the corresponding pdf is*

$$p\left(\mathbf{x}\right) = \frac{1}{C_N(\lambda)} \exp\left(-\lambda \left\|\mathbf{x}\right\|_\infty\right) \tag{2}$$

with $C_N(\lambda) \triangleq N! \left(\frac{2}{\lambda}\right)^N$.

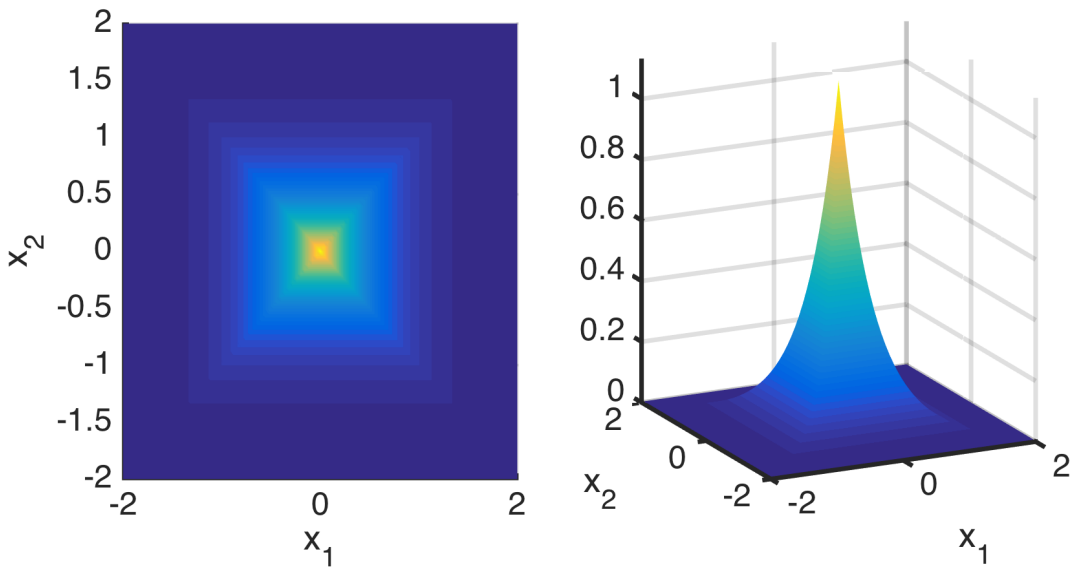

Fig. 1. The democratic pdf $\mathcal{D}_N(\lambda)$ for $N = 2$ and $\lambda = 3$.

As an illustration, the pdf of the bidimensional democratic pdf for $\lambda = 3$ is depicted in Fig. 1.

Remark 1. *It is interesting to note that the democratic distribution belongs to the exponential family. Indeed, its pdf can be factorized as*

$$p(\mathbf{x}) = a(\mathbf{x})\ b(\lambda)\ \exp\left(\eta(\lambda) T(\mathbf{x})\right) \tag{3}$$

where $a(\mathbf{x}) = 1$, $b(\lambda) = 1/C_N(\lambda)$, $\eta(\lambda) = -\lambda$ *and* $T(\mathbf{x}) = \|\mathbf{x}\|_\infty$ *defines sufficient statistics.*

B. Moments

The two first moments of the democratic distribution are available through the following property.

Property 1. *Let* $\mathbf{x} = [x_1, \ldots, x_N]^T$ *be a random vector obeying the democratic distribution*

$\mathcal{D}_N(\lambda)$. *The mean and the covariance matrix are given by:*

$$\mathrm{E}\left[x_n\right] = 0 \qquad \forall n \in \{1, \ldots, N\} \tag{4}$$

$$\mathrm{var}\left[x_n\right] = \frac{(N+1)(N+2)}{3\lambda^2} \qquad \forall n \in \{1, \ldots, N\} \tag{5}$$

$$\mathrm{cov}\left[x_i, x_j\right] = 0 \qquad \forall i \neq j. \tag{6}$$

Proof. See Appendix B. □

C. Marginal distributions

The marginal distributions of any democratically distributed vector $\mathbf{x}$ are given by the following Lemma.

Lemma 2. *Let* $\mathbf{x} = [x_1, \ldots, x_N]^T$ *be a random vector obeying the democratic distribution* $\mathcal{D}_N(\lambda)$. *For any positive integer* $J < N$, *let* $\mathcal{K}_J$ *denote a* J*-element subset of* $\{1, \ldots, N\}$ *and* $\mathbf{x}_{\backslash \mathcal{K}_J}$ *the sub-vector of* $\mathbf{x}$ *whose* J *elements indexed by* $\mathcal{K}_J$ *have been removed. Then the marginal pdf of the sub-vector* $\mathbf{x}_{\backslash \mathcal{K}_J} \in \mathbb{R}^{N-J}$ *is given by*

$$p\left(\mathbf{x}_{\backslash \mathcal{K}_J}\right) = \frac{2^J}{C_N(\lambda)} \sum_{j=0}^{J} \binom{J}{j} \frac{(J-j)!}{\lambda^{J-j}} \left\|\mathbf{x}_{\backslash \mathcal{K}_J}\right\|_\infty^j \times \exp\left(-\lambda \left\|\mathbf{x}_{\backslash \mathcal{K}_J}\right\|_\infty\right). \tag{7}$$

Proof. See Appendix C. □

In particular, as a straightforward corollary of this lemma, two specific marginal distributions of $\mathcal{D}_N(\lambda)$ are given by the following property.

Property 2. *Let* $\mathbf{x} = [x_1, \ldots, x_N]^T$ *be a random vector obeying the democratic distribution* $\mathcal{D}_N(\lambda)$. *The components* x_n *(*$n = 1, \ldots, N$*) of* $\mathbf{x}$ *are identically and marginally distributed according to the following* N*-component mixture of double-sided Gamma distributions (see Appendix D)*

$$x_n \sim \frac{1}{N} \sum_{j=1}^{N} d\mathcal{G}\left(j, \lambda\right). \tag{8}$$

Moreover, the pdf of the sub-vector $\mathbf{x}_{\backslash n}$ *of* $\mathbf{x}$ *whose* n*th element has been removed is :*

$$p\left(\mathbf{x}_{\backslash n}\right) = \frac{1 + \lambda \left\|\mathbf{x}_{\backslash n}\right\|_\infty}{N\ C_{N-1}(\lambda)} \exp\left(-\lambda \left\|\mathbf{x}_{\backslash n}\right\|_\infty\right). \tag{9}$$

Proof. See Appendix C. □

These two specific marginal distributions $p\left(\mathbf{x}_{\setminus n}\right)$ and $p\left(x_n\right)$ are depicted in Fig. 2 (top and bottom, right).

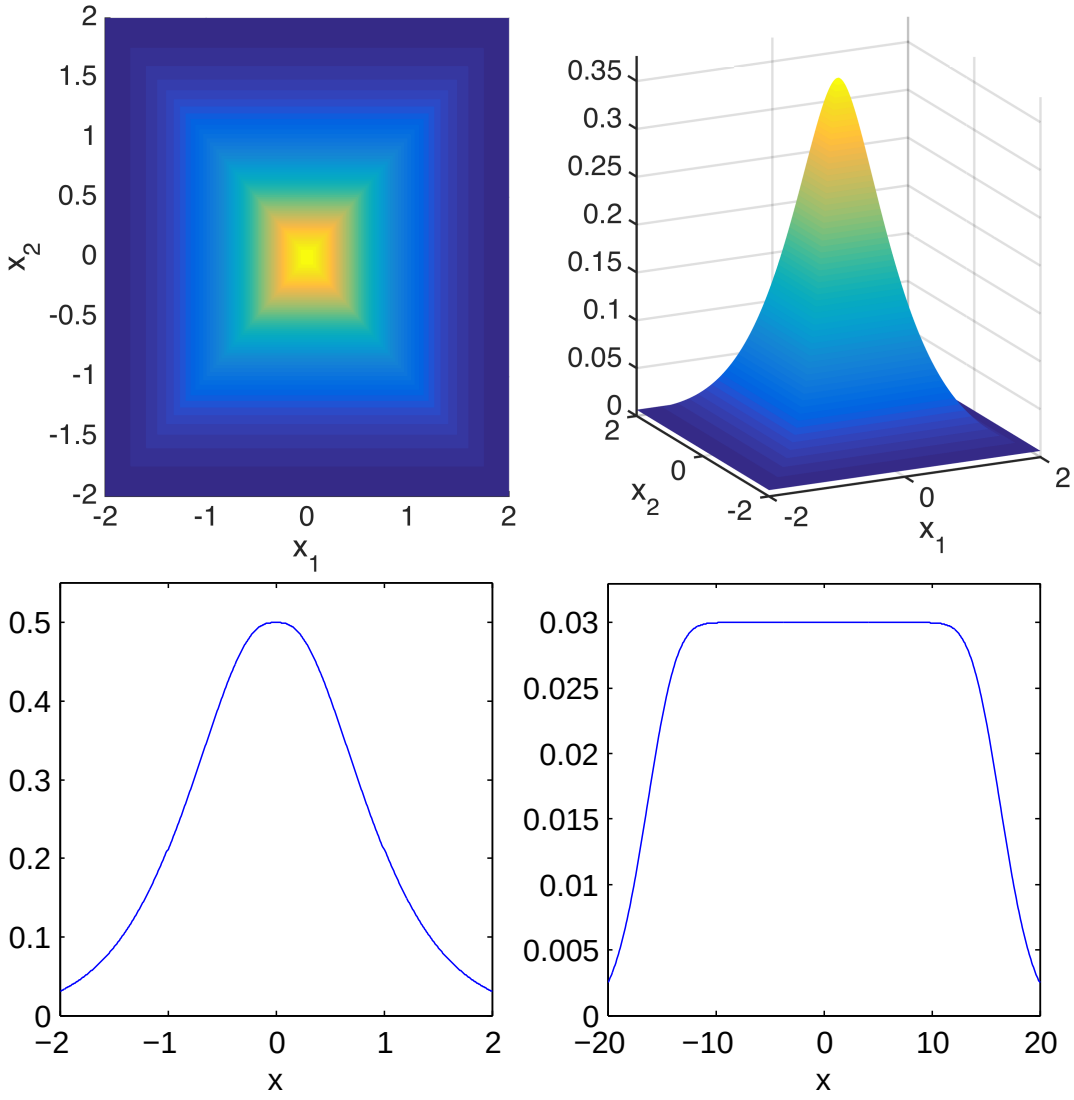

Fig. 2. Top: marginal distribution of $\mathbf{x}_{\setminus n}$ when $\mathbf{x} \sim \mathcal{D}_N\left(\lambda\right)$ for $N = 3$ and $\lambda = 3$. Bottom: marginal distribution of x_n when $\mathbf{x} \sim \mathcal{D}_N\left(\lambda\right)$ for $N = 3$ (left) or $N = 50$ (right) and $\lambda = 3$.

Remark 2. *It is worth noting that the distribution in* (8) *can be rewritten as*

$$p\left(x_n\right) = \frac{\lambda}{2N}\left(\sum_{j=0}^{N-1} \frac{\lambda^j}{j!}\left|x_n\right|^j\right) \exp\left(-\lambda\left|x_n\right|\right)$$

which behaves as $p(x_n) \approx \frac{\lambda}{2N}$ *when* $N \to +\infty$*. This means that the components of* $\mathbf{x}$ *tend to be marginally distributed according to uniform distributions over* $\mathbb{R}$ *in high dimension. This behavior is depicted in Fig. 2 bottom-right.*

D. Conditional distributions

Before introducing conditional distributions associated with any democratically distributed random vector, let partition $\mathbb{R}^N$ into a set of N non-overlapping double-convex cones $\mathcal{C}_n \subset \mathbb{R}^N$ $(n = 1, \ldots, N)$ defined by

$$\mathcal{C}_n \triangleq \left\{ \mathbf{x} = [x_1, \ldots, x_N]^T \in \mathbb{R}^N \ : \ \forall j \neq n, \ |x_n| > |x_j| \right\}. \tag{10}$$

These sets are directly related to the index of the so-called *dominant component* of a given democratically distributed vector $\mathbf{x}$. More precisely, if $\|\mathbf{x}\|_\infty = |x_n|$, then $\mathbf{x} \in \mathcal{C}_n$ and the n^{th} component x_n of $\mathbf{x}$ is said to be the dominant component.

An example is given in Fig. 3 where $\mathcal{C}_1 \subset \mathbb{R}^2$ is depicted. These double-cones partition $\mathbb{R}^N$ into N equiprobable sets with respect to (w.r.t.) the democratic distribution, as stated in the following property.

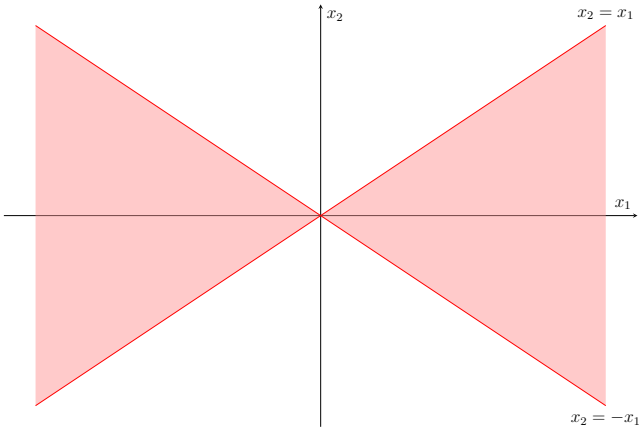

Fig. 3. The double-cone $\mathcal{C}_1$ of $\mathbb{R}^2$ appears as the light red area while the complementary double-cone $\mathcal{C}_2$ is the uncolored area.

Property 3. *Let* $\mathbf{x} = [x_1, \ldots, x_N]^T$ *be a random vector obeying the democratic distribution* $\mathcal{D}_N(\lambda)$*. Then the probability that this vector belongs to a given double-cone is*

$$\mathrm{P}\left[\mathbf{x} \in \mathcal{C}_n\right] = \frac{1}{N}. \tag{11}$$

Proof. See Appendix E, paragraph E-A. □

Remark 3. *This property simply exhibits intuitive intrinsic symmetries of the democratic distribution: the dominant component of a democratically distributed vector is located with equal probabilities in any of the cones $\mathcal{C}_n$.*

Moreover, the following lemma yields some results on conditional distributions related to these sets.

Lemma 3. *Let $\mathbf{x} = [x_1, \ldots, x_N]^T$ be a random vector obeying the democratic distribution $\mathcal{D}_N(\lambda)$. Then the following results hold*

$$x_n|\mathbf{x} \in \mathcal{C}_n \sim d\mathcal{G}(N, \lambda) \tag{12}$$

$$\mathbf{x}_{\backslash n}|\mathbf{x} \in \mathcal{C}_n \sim \mathcal{D}_{N-1}(\lambda) \tag{13}$$

$$x_j|x_n, \mathbf{x} \in \mathcal{C}_n \sim \mathcal{U}(-|x_n|, |x_n|) \quad (j \neq n) \tag{14}$$

and

$$\mathrm{P}\left[\mathbf{x} \in \mathcal{C}_n|\mathbf{x}_{\backslash n}\right] = \frac{1}{1 + \lambda \left\|\mathbf{x}_{\backslash n}\right\|_\infty} \tag{15}$$

$$p\left(\mathbf{x}_{\backslash n}|\mathbf{x} \notin \mathcal{C}_n\right) = \frac{\lambda}{N-1} \frac{\left\|\mathbf{x}_{\backslash n}\right\|_\infty}{C_{N-1}(\lambda)} e^{-\lambda \left\|\mathbf{x}_{\backslash n}\right\|_\infty}. \tag{16}$$

Proof. See Appendix E, paragraph E-B. □

Remark 4. *According to* (12)*, the marginal distribution of the dominant component is a double-sided Gamma distribution. Conversely, according to* (13)*, the vector of the non-dominant components is marginally distributed according to a democratic distribution. Conditionally upon the dominant component, these non-dominant components are independently and uniformly distributed on the admissible set, as shown in* (14)*. The probability in* (15) *shows that the probability that the nth component dominates increases when the other components are of low amplitude.*

Finally, based on Lemma 3, the following property related to the conditional distributions of $\mathcal{D}_N(\lambda)$ can be stated.

Property 4. *Let* $\mathbf{x} = [x_1, \ldots, x_N]^T$ *be a random vector obeying the democratic distribution* $\mathcal{D}_N(\lambda)$*. The pdf of the conditional distribution of a given component* x_n *given* $\mathbf{x}_{\backslash n}$ *is*

$$p\left(x_n|\mathbf{x}_{\backslash n}\right) = (1-c_n)\,\frac{1}{2\left\|\mathbf{x}_{\backslash n}\right\|_\infty}\mathbf{1}_{\mathcal{I}_n}(x_n) + c_n\frac{\lambda}{2}e^{-\lambda\left(|x_n|+\left\|\mathbf{x}_{\backslash n}\right\|_\infty\right)}\mathbf{1}_{\mathbb{R}\backslash\mathcal{I}_n}(x_n) \tag{17}$$

with $\mathcal{I}_n \triangleq \left(-\left\|\mathbf{x}_{\backslash n}\right\|_\infty, \left\|\mathbf{x}_{\backslash n}\right\|_\infty\right)$ *and where* $c_n = \mathrm{P}\left[\mathbf{x} \in \mathcal{C}_n|\mathbf{x}_{\backslash n}\right]$ *is defined by* (15)*.*

Proof. See Appendix E, paragraph E-C. □

Remark 5. *The pdf in* (17) *defines a mixture of one uniform distribution and two shifted exponential distributions with probabilities* $1-c_n$ *and* $c_n/2$*, respectively. An example of this pdf is depicted in Fig 4. This property opens the door to a natural random variate generator according to the democratic distribution through the use of a standard Gibbs sampler. This random generation strategy is detailed in paragraph II-F2.*

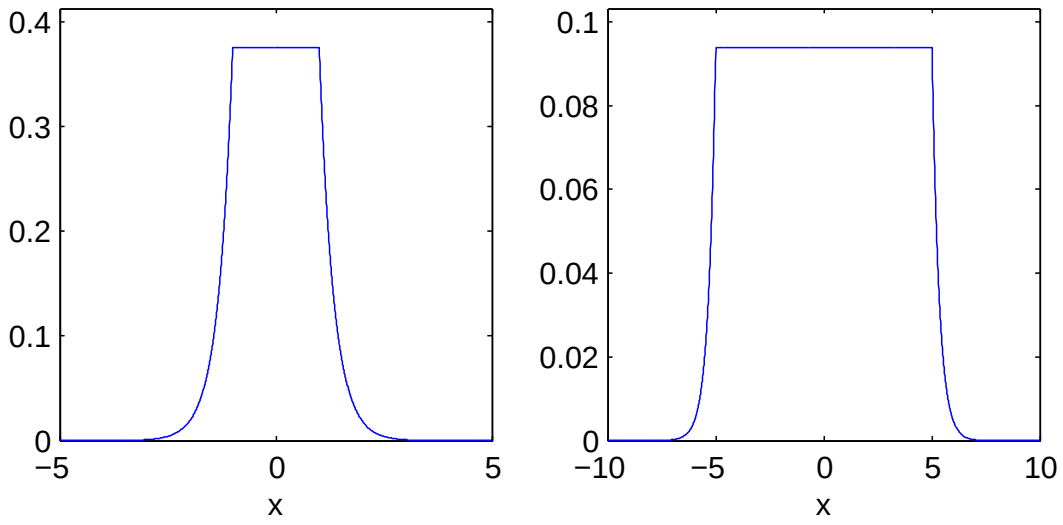

Fig. 4. Conditional distribution of $x_n|\mathbf{x}_{\backslash n}$ when $\mathbf{x} \sim \mathcal{D}_N(\lambda)$ for $N = 2$, $\lambda = 3$ and $\left\|\mathbf{x}_{\backslash n}\right\|_\infty = 1$ (left) or $\left\|\mathbf{x}_{\backslash n}\right\|_\infty = 10$ (right).

E. Proximity operator of the negative log-pdf

The pdf of the democratic distribution $\mathcal{D}_N(\lambda)$ can be written as $p(\mathbf{x}) \propto \exp\left(-g_1\left(\mathbf{x}\right)\right)$ with

$$g_1\left(\mathbf{x}\right) = \lambda\left\|\mathbf{x}\right\|_\infty. \tag{18}$$

This paragraph introduces the proximity mapping operator associated with the negative log-distribution $g_1(\mathbf{x})$ (defined up to a multiplicative constant). This proximal operator will be subsequently resorted to implement Monte Carlo algorithms to draw samples from the democratic distribution $\mathcal{D}_N(\lambda)$ (see paragraph II-F) as well as posterior distributions derived from a democratic prior (see paragraph III-B3). In this context, it is convenient to define the proximity operator of $g_1(\cdot)$ as [34]

$$\text{prox}_{g_1}^{\delta}(\mathbf{x}) = \underset{\mathbf{u}\in\mathbb{R}^N}{\text{argmin}}\ \lambda \|\mathbf{u}\|_\infty + \frac{1}{2\delta}\|\mathbf{x}-\mathbf{u}\|_2^2 \tag{19}$$

Up to the authors' knowledge, no closed-form expression is available for (19). However, this operator can be explicitly computed following the algorithmic scheme detailed in Algo. 1, based on the following property.

Property 5. *Let $\mathbf{x} \in \mathbb{R}^N$ and $\delta \in \mathbb{R}_+$. We denote $\varepsilon_1, \ldots, \varepsilon_J$ the J distinct values among $\{|x_n|\}_{n=1}^N$ and $d_1, \ldots, d_J$ their respective multiplicity orders with $\sum_{j=1}^J d_j = N$. Then the n-th component of $\boldsymbol{\rho} \triangleq \text{prox}_{g_1}^{\delta}(\mathbf{x})$, denoted as ρ_n, is given by*

$$\rho_n = \begin{cases} \text{sign}(x_n)\phi_\delta(\mathbf{x}) & \text{if } |x_n| \geq \phi_\delta(\mathbf{x}) \\ x_n & \text{otherwise} \end{cases} \tag{20}$$

where

$$\phi_\delta(\mathbf{x}) = \max\left(0, \phi_\delta^1(\mathbf{x}), \ldots, \phi_\delta^J(\mathbf{x})\right) \tag{21}$$

and, for $j = 1, \ldots, J$,

$$\phi_\delta^j(\mathbf{x}) = \frac{1}{\sum_{k=1}^j d_k}\left(\sum_{k=1}^j d_k\varepsilon_k - \lambda\delta\right). \tag{22}$$

Proof. See Appendix F. □

From Property 5, under this proximity mapping, all components greater than a threshold are reduced to a common value, while all the others remain unchanged.

Remark 6. *As stated earlier, this proximity operator will be resorted while designing Monte Carlo algorithms able to generate random samples according to distributions derived from $g_1(\cdot)$. In this specific context, almost surely, the multiplicity orders d_j ($j = 1, \ldots, J$) are all equal to one, i.e., $J = N$ and $\varepsilon_n = |x_n|$ ($n = 1, \ldots, N$).*

Algorithm 1: Algorithm to compute $\text{prox}_{g_1}^{\delta}(\mathbf{x})$

Input: $\mathbf{x}$, δ

Identify $\varepsilon_1 \dots \varepsilon_J$ as the J different values of $|x_n|$'s, and $d_1 \dots d_J$ their respective multiplicity order ;

for $j \leftarrow 1$ **to** J **do**

  Compute $\phi_\delta^j(\mathbf{x})$ following (22) ;

end

Compute $\phi_\delta(\mathbf{x}) = \max\left(0, \phi_\delta^1(\mathbf{x}), \dots, \phi_\delta^J(\mathbf{x})\right)$;

for $n \leftarrow 1$ **to** N **do**

  if $|x_n| \geq \phi_\delta(\mathbf{x})$ **then**

    $\rho_n = \text{sign}(x_n)\phi_\delta(\mathbf{x})$;

  else

    $\rho_n = x_n$;

  end

end

Set $\boldsymbol{\rho} = [\rho_1, \dots, \rho_N]^T$;

Output: $\boldsymbol{\rho} = \text{prox}_{g_1}^{\delta}(\mathbf{x})$

F. Random variate generation

This paragraph introduces three distinct random variate generators that allow to draw samples according to the democratic distribution.

1) Exact random variate generator: Property 3 combined with Lemma 3 permits to rewrite the joint distribution of a democratically distributed vector according to the following chain rule

$$\begin{aligned} p(\mathbf{x}) &= \sum_{n=1}^{N} p\left(\mathbf{x}_{\setminus n} | x_n, \mathbf{x} \in \mathcal{C}_n\right) p\left(x_n | \mathbf{x} \in \mathcal{C}_n\right) \mathrm{P}\left[\mathbf{x} \in \mathcal{C}_n\right] \\ &= \sum_{n=1}^{N} \left[\prod_{j \neq n} p\left(x_j | x_n, \mathbf{x} \in \mathcal{C}_n\right)\right] p\left(x_n | \mathbf{x} \in \mathcal{C}_n\right) \mathrm{P}\left[\mathbf{x} \in \mathcal{C}_n\right] \end{aligned}$$

where $\mathrm{P}\left[\mathbf{x} \in \mathcal{C}_n\right]$, $p\left(x_n | \mathbf{x} \in \mathcal{C}_n\right)$ and $p\left(x_j | x_n, \mathbf{x} \in \mathcal{C}_n\right)$ are given in (11), (12) and (14), respectively. This finding can be fully exploited to design an efficient and exact random variate generator

for the democratic distribution, see Algo. 2.

Algorithm 2: Democratic random variate generator using an exact sampling scheme.

Input: Parameter λ, dimension N

% Drawing the cone of the dominant component

Sample n_{dom} uniformly on the set $\{1 \dots N\}$;

% Drawing the dominant component

Sample $x_{n_{\mathrm{dom}}}$ according to (12);

% Drawing the non-dominant components

for $j \leftarrow 1$ **to** N *(*$j \neq n_{\mathrm{dom}}$*)* **do**

  Sample x_j according to (14);

end

Output: $\mathbf{x} = [x_1, \dots, x_N]^T \sim \mathcal{D}_N(\lambda)$

2) Gibbs sampler-based random generator: Property 4 can be exploited to design a democratic random variate generator through the use of a Gibbs sampling scheme. It consists of successively drawing the components x_n according to the conditional distributions (17), defined as the mixtures of uniform and truncated Laplacian distributions. After a given number T_{bi} of burn-in iterations, this generator, described in Algo. 3, provides samples asymptotically distributed according to the democratic distribution $\mathcal{D}_N(\lambda)$.

3) P-MALA-based random generator: An alternative to draw samples according to the democratic distribution is the proximal Metropolis-adjusted Langevin algorithm (P-MALA) introduced in [33]. P-MALA builds a Markov chain $\left\{\mathbf{x}^{(t)}\right\}_{t=1}^{T_{\mathrm{MC}}}$ whose stationary distribution is of the form

$$p(\mathbf{x}) \propto \exp(-g(\mathbf{x}))$$

where g is a positive convex function with $\lim_{\|\mathbf{x}\| \to \infty} g(\mathbf{x}) = +\infty$. It relies on successive Metropolis Hastings moves with Gaussian proposal distributions whose mean has been chosen as the proximal operator of $g(\cdot)$ evaluated at the current state of the chain. In the particular case of the democratic distribution, P-MALA can be implemented by exploiting the derivations in paragraph II-E, where $g(\cdot) = g_1(\cdot)$ has been defined in (18). More precisely, at iteration t of

Algorithm 3: Democratic random variate generator using a Gibbs sampling scheme.

Input: Parameter λ, dimension N, number of burn-in iterations T_{bi}, total number of iterations T_{MC}, initialization $\mathbf{x}^{(1,0)} = \left[x_1^{(1,0)}, \ldots, x_N^{(1,0)}\right]_{\mathrm{MC}}^T$

for $t \leftarrow 1$ **to** T_{MC} **do**
  for $n \leftarrow 1$ **to** N **do**
    Set $\mathbf{x}_{\setminus n}^{(t,n-1)} = \left[x_1^{(t,n)}, \ldots, x_{n-1}^{(t,n)}, x_{n+1}^{(t,n-1)}, \ldots, x_N^{(t,n-1)}\right]^T$;
    Draw $x_n^{(t,n)}$ according to (17);
    Set $\mathbf{x}^{(t,n)} = \left[x_1^{(t,n)}, \ldots, x_{n-1}^{(t,n)}, x_n^{(t,n)}, x_{n+1}^{(t,n-1)}, \ldots, x_N^{(t-1,n-1)}\right]^T$;
  end
  Set $\mathbf{x}^{(t+1,0)} = \mathbf{x}^{(t,N)}$;
end

Output: $\mathbf{x}^{(t,0)} \sim \mathcal{D}_N(\lambda)$ (for $t > T_{\mathrm{bi}}$)

the sampler, a candidate $\mathbf{x}^*$ is proposed as

$$\mathbf{x}^* | \mathbf{x}^{(t-1)} \sim \mathcal{N}\left(\mathrm{prox}_{g_1}^{\delta/2}\left(\mathbf{x}^{(t-1)}\right), \delta \mathbf{I}_N\right). \tag{23}$$

Then this candidate is accepted as the new state $\mathbf{x}^{(t)}$ with probability

$$\alpha = \min\left(1, \frac{p\left(\mathbf{x}^* | \lambda\right)}{p\left(\mathbf{x}^{(t-1)} | \lambda\right)} \frac{q\left(\mathbf{x}^{(t-1)} | \mathbf{x}^*\right)}{q\left(\mathbf{x}^* | \mathbf{x}^{(t-1)}\right)}\right) \tag{24}$$

where $q\left(\mathbf{x}^* | \mathbf{x}^{(t-1)}\right)$ is the pdf of the Gaussian distribution given by (23). The algorithmic parameter δ is empirically chosen such that the acceptance rate of the sampler lies between 0.4 and 0.6. The full algorithmic scheme is available in Algo. 4.

4) Random generator performance comparison: Figure 5 compares the first 15 lags of the empirical autocorrelation function (ACF), computed with 500 samples drawn from the democratic distribution $\mathcal{D}_N(3)$ for $N = 2$ (left) and $N = 50$ (right) using the exact (top), Gibbs sampler-based (middle) and P-MALA (bottom) variate generators. In lower dimensional cases, the chain generated with exact sampling has remarkably lower autocorrelation.

Computational times required to generate 1000 samples from the democratic distribution for various dimensions are reported in Table II. These results show that the Gibbs sampler-based method has a significantly higher cost when compared with the exact random generator.

Algorithm 4: Democratic random variate generator using P-MALA.

Input: Parameter λ, dimension N, number of burn-in iterations T_{bi}, total number of iterations T_{MC}, algorithmic parameter δ, initialization $\mathbf{x}^{(0)}$

for $t \leftarrow 1$ **to** T_{MC} **do**
  Draw $\mathbf{x}^*|\mathbf{x}^{(t-1)} \sim \mathcal{N}\left(\mathrm{prox}_{g_1}^{\delta/2}\left(\mathbf{x}^{(t-1)}\right), \delta\mathbf{I}_N\right)$;
  Compute α following (24) ;
  Draw $w \sim \mathcal{U}(0,1)$;
  if $w < \alpha$ **then**
    Set $\mathbf{x}^{(t)} = \mathbf{x}^*$;
  else
    Set $\mathbf{x}^{(t)} = \mathbf{x}^{(t-1)}$;
  end
end

Output: $\mathbf{x}^{(t)} \sim \mathcal{D}_N(\lambda)$ (for $t > T_{\mathrm{bi}}$)

TABLE II

CUMULATIVE TIMES TO DRAW 500 SAMPLES ACCORDING THE DEMOCRATIC DISTRIBUTION $\mathcal{D}_N(\lambda)$ WITH $\lambda = 3$ FOR VARIOUS N.

N	Exact (ms)	Gibbs (ms)	P-MALA (ms)
2	2.06×10^{-1}	2.05×10^{3}	1.11×10^{2}
5	3.30×10^{-1}	2.66×10^{3}	1.12×10^{2}
10	3.46×10^{-1}	3.60×10^{3}	1.14×10^{2}
50	9.24×10^{-1}	1.01×10^{4}	1.27×10^{2}
100	1.71×10^{0}	1.85×10^{4}	1.46×10^{2}

Whereas the exact sampler easily scales in higher dimension, the Gibbs based sampler needs approximately 10^4 more time. This is explained by its intrinsic algorithmic structure: the Gibbs sampler-based method requires to draw a multinomial variable for each component, followed by either exponential or uniform distributed variables. Conversely, exact random generator only needs to generate one gamma distributed variable and $(N-1)$ uniform samples.

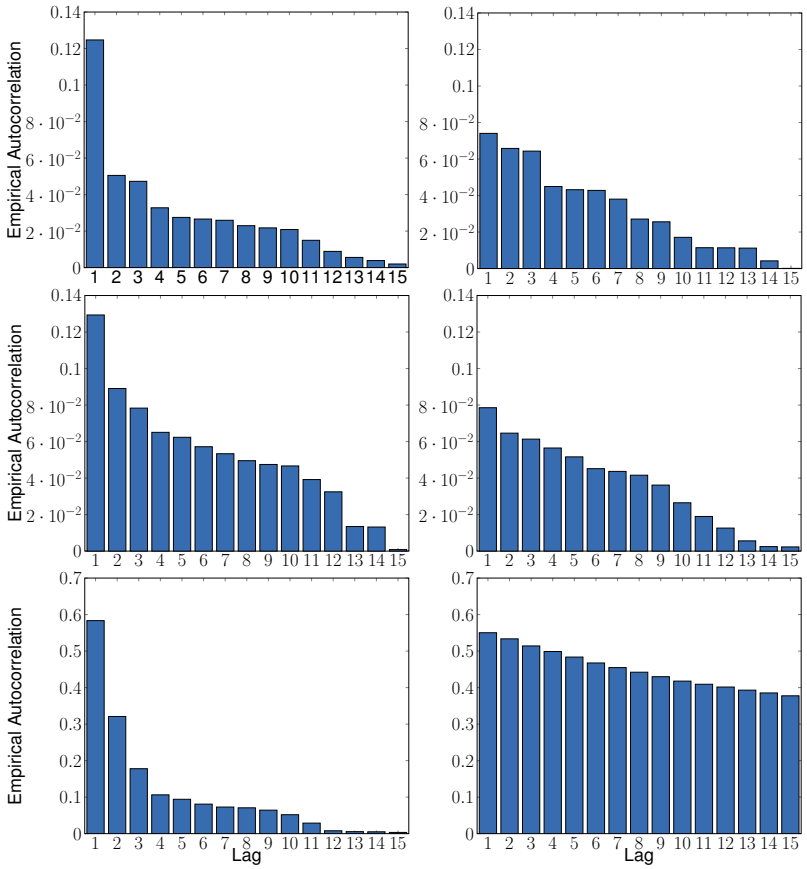

Fig. 5. First 15 lags of the empirical autocorrelation function when generating 500 samples drawn from $\mathcal{D}_N(\lambda)$ using the exact (top), Gibbs sampler-based (middle) and P-MALA (bottom) variate generators with $\lambda = 3$ for $N = 2$ (left) and $N = 50$ (right).

From these findings, the Gibbs sampler-based and P-MALA strategies might seem out of interest since significantly outperformed by the exact random generator in terms of time computation and mixing performance. However, both exhibit interesting properties that can be exploited in a more general scheme. First, the Gibbs sampler-based generator shows that each component of a democratically distributed vector can be easily generated conditionally on the others. Then, P-MALA exploits the algorithmic derivation of the proximity operator associated with $g_1(\cdot)$ to draw vectors asymptotically distributed according to the democratic distribution. This opens the door to extended schemes for sampling according to a posterior distribution resulting from a democratic prior when possibly no exact sampler is available. This will be discussed in Section III.

III. Democratic prior in a linear inverse problem

This section aims to provide a Bayesian formulation of the model underlying the problem described by (1). From a Bayesian perspective, the solution of (1) can be straightforwardly interpreted as the MAP estimator associated with a linear observation model characterized by an additive Gaussian residual and complemented by a democratic prior assumption. Assuming a Gaussian residual results in a quadratic discrepancy measure and, as a consequence, in a quadratic data fidelity term as in (1). Setting the anti-sparse coding problem into a fully Bayesian framework paves the way to a comprehensive statistical description of the solution. Moreover, it permits to implement other algorithmic strategies beyond MAP estimation, namely Markov chain Monte Carlo techniques.

A. Hierarchical Bayesian model

Let $\mathbf{y} = [y_1 \dots y_M]^T$ denote an observed measurement vector. These observations are assumed to be related to an unknown description vector $\mathbf{x} = [x_1 \dots x_N]^T$ through a known coding matrix $\mathbf{H}$ according to the linear model

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{e}. \tag{25}$$

The residual vector $\mathbf{e} = [e_1 \dots e_N]^T$ is assumed to be distributed according to a centered multivariate Gaussian distribution $\mathcal{N}(\mathbf{0}_M, \sigma^2 \boldsymbol{I}_M)$. The Bayesian model is introduced in what follows. It relies on the definition of the likelihood function associated with the observation vector $\mathbf{y}$ and on the choice of prior distributions for the unknown parameters, i.e., the representation vector $\mathbf{x}$ and the residual variance σ^2, assumed to be a priori independent.

1) Likelihood function: The Gaussian property of the additive residual term yields the following likelihood function

$$f(\mathbf{y}|\mathbf{x}, \sigma^2) = \left(\frac{1}{2\pi\sigma^2}\right)^{\frac{M}{2}} \exp\left[-\frac{1}{2\sigma^2}\left\|\mathbf{y} - \mathbf{H}\mathbf{x}\right\|_2^2\right]. \tag{26}$$

2) Residual variance prior: A noninformative Jeffreys prior distribution is chosen for the residual variance σ^2

$$f\left(\sigma^2\right) \propto \frac{1}{\sigma^2}. \tag{27}$$

3) Description vector prior: As motivated earlier, the democratic distribution introduced in Section II is assigned as the prior distribution of the N-dimensional unknown vector $\mathbf{x}$

$$\mathbf{x} \mid \lambda \sim \mathcal{D}_N(\lambda). \tag{28}$$

In the following, the hyperparameter λ is set as $\lambda = N\mu$, where μ is assumed to be unknown. Enforcing the parameter of the democratic distribution to depend on the problem dimension allows the prior to be scaled with this dimension. Indeed, as stated in (12), the absolute value of the dominant component is distributed according to the Gamma distribution $\mathcal{G}(N, \lambda)$, whose mean and variance are N/λ and N/λ^2, respectively. With the proposed scalability, the prior mean is constant with the dimension

$$\mathrm{E}\left[|x_n| \mid \mathbf{x} \in \mathcal{C}_n, \mu\right] = 1/\mu \tag{29}$$

and the variance tends to zero

$$\mathrm{var}\left[|x_n| \mid \mathbf{x} \in \mathcal{C}_n, \mu\right] = 1/(N\mu^2). \tag{30}$$

4) Hyperparameter prior: The prior modeling introduced in the previous paragraph is complemented by assigning prior distribution to the unknown hyperparameter μ, introducing a second level in the Bayesian hierarchy. More precisely, a conjugate Gamma distribution is chosen as a prior for μ

$$\mu \sim \mathcal{G}(a, b). \tag{31}$$

since the conjugacy property allows the posterior distribution to be easily derived. The values of a and b will be chosen to obtain a flat prior.

5) Posterior distribution: The posterior distribution of the unknown parameter vector $\boldsymbol{\theta} = \{\mathbf{x}, \sigma^2, \mu\}$ can be computed from the following hierarchical structure:

$$f(\boldsymbol{\theta}|\mathbf{y}) \propto f(\mathbf{y}|\mathbf{x}, \sigma^2) f(\mathbf{x}, \sigma^2|\mu) f(\mu) \tag{32}$$

where

$$f(\mathbf{x}, \sigma^2|\mu) = f(\sigma^2) f(\mathbf{x}|\mu) \tag{33}$$

and $f(\mathbf{y}|\mathbf{x},\sigma^2)$, $f(\sigma^2)$, $f(\mathbf{x}|\mu)$ and $f(\mu)$ have been defined in Eq. (26) to (31), respectively. Thus, this posterior distribution can be written as

$$\begin{aligned} f(\mathbf{x},\sigma^2,\mu|\mathbf{y}) &\propto \exp\left(-\frac{1}{2\sigma^2}\left\|\mathbf{y}-\mathbf{H}\mathbf{x}\right\|_2^2\right) \\ &\times \frac{1}{C_N(\mu N)}\exp\left(-\mu N\left\|\mathbf{x}\right\|_\infty\right) \\ &\times \left(\frac{1}{\sigma^2}\right)^{\frac{M}{2}+1}\mathbf{1}_{\mathbb{R}_+}(\sigma^2) \\ &\times \frac{b^a}{\Gamma(b)}\mu^{a-1}\exp\left(-b\mu\right). \end{aligned} \tag{34}$$

As expected, for given values of the residual variance σ^2 and the democratic parameter $\lambda=\mu N$, maximizing the posterior (34) can be formulated as the optimization problem in (1), for which some algorithmic strategies have been for instance introduced in [28] and [29]. In this paper, a different route has been taken by deriving inference schemes relying on MCMC algorithms. This choice permits to include the nuisance parameters σ^2 and μ into the model and to estimate them jointly with the representation vector $\mathbf{x}$. Moreover, since the proposed MCMC algorithms generate a collection $\left\{\left(\mathbf{x}^{(t)},\mu^{(t)},\sigma^{2(t)}\right)\right\}_{t=1}^{N_{\mathrm{MC}}}$ asymptotically distributed according to the posterior of interest (32), they provide a good knowledge of the statistical distribution of the solutions.

B. MCMC algorithm

This section introduces two MCMC algorithms to generate samples according to the posterior (34). They are two specific instances of Gibbs samplers which generate samples according to the conditional distributions associated with the posterior (34), following Algo. 5. As shown below, the steps for sampling according to the conditional distributions of the residual variance $f(\sigma^2|\mathbf{y},\mathbf{x})$ and the democratic parameter $f\left(\mu|\mathbf{x}\right)$ are straightforward. In addition, generating samples for the representation vector $f(\mathbf{x}|\mu,\mathbf{y})$ can be achieved component-by-component using N Gibbs moves, following the strategy in paragraph II-F2. However, for high dimensional problems, such a crude strategy may suffer from poor mixing properties, leading to slow convergence of the algorithm. To alleviate this issue, an alternative approach consists of sampling the full vector $\mathbf{x}|\mu,\mathbf{y}$ using a P-MALA step [33], similar to the one proposed in paragraph II-F3. These two strategies are detailed in the following paragraphs.

Algorithm 5: Gibbs sampler

Input: Observation vector $\mathbf{y}$, coding matrix $\mathbf{H}$, hyperparameters a and b, number of burn-in iterations T_{bi}, total number of iterations T_{MC}, algorithmic parameter δ, initialization $\mathbf{x}^{(0)}$

for $t \leftarrow 1$ **to** T_{MC} **do**

% Drawing the residual variance

Sample $\sigma^{2(t)}$ according to (35). ;

% Drawing the democratic parameter

Sample $\mu^{(t)}$ according to (37). ;

% Drawing the representation vector

Sample $\mathbf{x}^{(t)}$ using, either (see paragraph III-B3)

- Gibbs steps, i.e., following (38) ;
- P-MALA step, i.e., following (41) and (42);

end

Output: A collection of samples $\left\{\mu^{(t)}, \sigma^{2(t)}, \mathbf{x}^{(t)}\right\}_{t=T_{\mathrm{bi}}+1}^{T_{\mathrm{MC}}}$ asymptotically distributed according to (34).

1) Sampling the residual variance: Sampling according to the conditional distribution of the residual variance can be conducted according to the following inverse-gamma distribution

$$\sigma^2|\mathbf{y}, \mathbf{x} \sim \mathcal{IG}\left(\frac{M}{2}, \frac{1}{2}\left\|\mathbf{y}-\mathbf{H}\mathbf{x}\right\|_2^2\right) \tag{35}$$

2) Sampling the democratic hyperparameter: Looking carefully at (34), the conditional posterior distribution of the democratic parameter μ is

$$f(\mu|\mathbf{x}) \propto \mu^N \exp\left(-\mu\left\|\mathbf{x}\right\|_\infty\right) \mu^{a-1} \exp\left(-b\mu\right). \tag{36}$$

Therefore, sampling according to $f(\mu|\mathbf{x})$ is achieved as follows

$$\mu|\mathbf{x} \sim \mathcal{G}(a+N, b+N\left\|\mathbf{x}\right\|_\infty) \tag{37}$$

3) Sampling the description vector: Following the technical developments of paragraph II-F, two strategies can be considered to generate samples according to the conditional posterior distribution of the representation vector $f(\mathbf{x}|\mu, \sigma^2, \mathbf{y})$. They are detailed below.

Component-wise Gibbs sampling: A first possibility to draw a vector $\mathbf{x}$ according to $f(\mathbf{x}|\mu,\sigma^2,\mathbf{y})$ is to successively sample according to the conditional distribution of each component given the others, namely, $f(x_n|\mathbf{x}_{\backslash n},\mu,\sigma^2,\mathbf{y})$, as in algorithm of paragraph II-F2. More precisely, straightforward computations yield the following 3-mixture of truncated Gaussian distributions for this conditional

$$x_n|\mathbf{x}_{\backslash n},\mu,\sigma^2,\mathbf{y} \sim \sum_{i=1}^{3} \omega_{in}\mathcal{N}_{\mathcal{I}_{in}}\left(\mu_{in}, s_n^2\right) \tag{38}$$

where $\mathcal{N}_{\mathcal{I}}(\cdot,\cdot)$ denotes the Gaussian distribution truncated on the $\mathcal{I}$ and the truncation sets are defined as

$$\begin{aligned}
\mathcal{I}_{1n} &= \left(-\infty, -\left\|\mathbf{x}_{\backslash n}\right\|_\infty\right) \\
\mathcal{I}_{2n} &= \left(-\left\|\mathbf{x}_{\backslash n}\right\|_\infty, \left\|\mathbf{x}_{\backslash n}\right\|_\infty\right) \\
\mathcal{I}_{3n} &= \left(\left\|\mathbf{x}_{\backslash n}\right\|_\infty, +\infty\right).
\end{aligned}$$

The probabilities ω_{in} $(i=1,2,3)$ as well as the (hidden) means μ_{in} $(i=1,2,3)$ and variance s_n^2 of these truncated Gaussian distributions are given in Appendix G. This specific nature of the conditional distribution is intrinsically related to the nature of the conditional prior distribution stated in Property 4, which has already exhibited a 3-component mixture: one uniform distribution and two (shifted) exponential distributions defined over $\mathcal{I}_{2n}$, $\mathcal{I}_{1n}$ and $\mathcal{I}_{3n}$, respectively (see Remark 5). Note that sampling according to truncated distributions can be achieved using the strategy proposed in [35].

P-MALA: Similarly to the strategy developed in paragraph II-F3 to sample according to the prior distribution, sampling according to the conditional distribution $f(\mathbf{x}|\mu,\sigma^2,\mathbf{y})$ can be achieved using a P-MALA step [33]. In this case, the distribution of interest can be written as

$$f(\mathbf{x}|\mu,\sigma^2,\mathbf{y}) \propto \exp\left(g\left(\mathbf{x}\right)\right)$$

where $g\left(\mathbf{x}\right)$ derives from the Gaussian (negative log-) likelihood function and the (negative log-) distribution of the democratic prior so that

$$g(\mathbf{x}) = \frac{1}{2\sigma^2}\left\|\mathbf{y}-\mathbf{H}\mathbf{x}\right\|_2^2 + \lambda\left\|\mathbf{x}\right\|_\infty \tag{39}$$

with $\lambda = \mu N$. However, up to the authors' knowledge, the proximal operator associated with $g(\cdot)$ in (39) has no close form solution. To alleviate this problem, a first order approximation is

considered[1], as recommended in [33]

$$\text{prox}_g^{\delta/2}(\mathbf{x}) \approx \text{prox}_{g_1}^{\delta/2}\left(\mathbf{x} + \delta\, \nabla\left[\frac{1}{2\sigma^2}\left\|\mathbf{y} - \mathbf{H}\mathbf{x}\right\|_2^2\right]\right) \tag{40}$$

where $g_1(\cdot) = \lambda \left\|\cdot\right\|_\infty$ has been defined in paragraph II-F3 and the corresponding proximity mapping is available through Algo. 1. Finally, at iteration t of the main algorithm, sampling according to the conditional distribution $f(\mathbf{x}|\mu, \sigma^2, \mathbf{y})$ consists of drawing a candidate

$$\mathbf{x}^*|\mathbf{x}^{(t-1)} \sim \mathcal{N}\left(\text{prox}_g^{\delta/2}\left(\mathbf{x}^{(t-1)}\right), \delta\mathbf{I}_N\right) \tag{41}$$

and to accept this candidate as the new state $\mathbf{x}^{(t)}$ with probability

$$\alpha = \min\left(1, \frac{f\left(\mathbf{x}^*|\mu, \sigma^2, \mathbf{y}\right)}{f\left(\mathbf{x}^{(t-1)}|\mu, \sigma^2, \mathbf{y}\right)} \frac{q\left(\mathbf{x}^{(t-1)}|\mathbf{x}^*\right)}{q\left(\mathbf{x}^*|\mathbf{x}^{(t-1)}\right)}\right). \tag{42}$$

C. Inference

The sequences $\left\{\mathbf{x}^{(t)}, \sigma^{2(t)}, \mu^{(t)}\right\}_{t=1}^{T_{\text{MC}}}$ generated by the MCMC algorithms proposed in paragraph III-B are used to approximate Bayesian estimators. After a burn-in period of N_{bi} iterations, the set of generated samples $\mathcal{X} = \left\{\mathbf{x}^{(t)}\right\}_{t=T_{\text{bi}}+1}^{T_{\text{MC}}}$ is asymptotically distributed according to the marginal posterior distribution $f\left(\mathbf{x}|\mathbf{y}\right)$, resulting from the marginalization of the joint posterior distribution $f\left(\mathbf{x}, \sigma^2, \mu|\mathbf{y}\right)$ in (34) over the nuisance parameters σ^2 and μ

$$\begin{aligned} f\left(\mathbf{x}|\mathbf{y}\right) &= \int f\left(\mathbf{x}, \sigma^2, \mu|\mathbf{y}\right) d\sigma^2 d\mu & (43)\\ &\propto \left\|\mathbf{y} - \mathbf{H}\mathbf{x}\right\|_2^{-\frac{M}{2}} \left(b + N\left\|\mathbf{x}\right\|_\infty\right)^{-(a+N)}. & (44)\end{aligned}$$

As a consequence, while the minimum mean square error (MMSE) estimator of the representation vector $\mathbf{x}$ can be approximated as an empirical average over the set $\mathcal{X}$

$$\begin{aligned} \hat{\mathbf{x}}_{\text{MMSE}} &= \text{E}\left[\mathbf{x}|\mathbf{y}\right] & (45)\\ &\simeq \frac{1}{T_r}\sum_{t=1}^{T_{\text{MC}}} \mathbf{x}^{(t)}. & (46)\end{aligned}$$

[1]Note that a similar step is involved in the fast iterative truncation algorithm (FITRA) [31], a deterministic counterpart of the proposed algorithm and considered in the next section for comparison.

with $T_r = T_{\mathrm{MC}} - T_{\mathrm{bi}}$, the marginal maximum a posteriori (mMAP) estimator can be approximated as

$$\hat{\mathbf{x}}_{\mathrm{mMAP}} = \underset{\mathbf{x}\in\mathbb{R}^N}{\operatorname{argmax}}\, f\left(\mathbf{x}|\mathbf{y}\right) \tag{47}$$

$$\simeq \underset{\mathbf{x}^{(t)}\in\mathcal{X}}{\operatorname{argmax}}\, f\left(\mathbf{x}^{(t)}|\mathbf{y}\right). \tag{48}$$

IV. Experiments

This section reports several simulation results to illustrate the performance of the Bayesian anti-sparse coding algorithms introduced in Section III. In paragraph IV-A, the validity of the samplers derived in paragraph III-B has been assessed following the experimental scheme in [36], exploiting some properties of the democratic distributions enounced in Section II. Paragraph IV-B evaluates the performances of the two versions of the samplers (i.e., using Gibbs or P-MALA steps) on a toy example, by considering measurements resulting from a representation vector whose coefficients are all equal up to a sign. Finally, paragraph IV-C compares the performances of the proposed algorithm and its deterministic counterpart introduced in [31]. For all experiments, the coding matrices $\mathbf{H}$ have been chosen as randomly subsampled columnwise DCT matrices since they have shown to yield democratic representations with small ℓ_∞-norm and good democracy bounds [28]. However, note that a deep investigation of these bounds is out of the scope of the present paper.

A. Assessment of the Bayesian anti-sparse coding algorithms

We first consider a first experiment to assess the validity of the Monte Carlo algorithms introduced in paragraph III-B and, in particular, the two methods proposed to sample according to the conditional distribution of the representation vector $f\left(\mathbf{x}|\mathbf{y},\sigma^2,\mu\right)$, see step 7 in Algo. 5. To this aim, the so-called *successive conditional sampling* strategy proposed by Geweke in [36] is followed for fixed nuisance parameters σ^2 and μ. This procedure does not fully assert the correctness of the sampler but it may help to detect errors, e.g., as in [37]. More precisely, it consists of drawing a sequence $\left\{\mathbf{x}^{(t)},\mathbf{y}^{(t)}\right\}_{t=1}^{T_{\mathrm{MC}}}$ asymptotically distributed according to the joint distribution $f\left(\mathbf{x},\mathbf{y}|\sigma^2,\mu\right)$ using the Gibbs sampler described in Algo. 6.

One can notice that this algorithm boils down to successively sample according to the Gaussian likelihood distribution $f\left(\mathbf{y}|\mathbf{x},\sigma^2,\mu\right)$ and the conditional posterior of interest $f\left(\mathbf{x}|\mathbf{y},\sigma^2,\mu\right)$. This

Algorithm 6: Successive conditional sampling

Input: Residual variance σ^2, democratic parameter μ, coding matrix $\mathbf{H}$.

Sample $\mathbf{x}^{(0)}$ according to $\mathcal{D}_N(\mu N)$;

for $t \leftarrow 1$ **to** T_{MC} **do**

  Sample $\mathbf{y}^{(t)}|\mathbf{x}^{(t-1)}, \sigma^2 \sim \mathcal{N}(\mathbf{H}\,\mathbf{x}^{(t)}, \sigma^2)$;

  Sample $\mathbf{x}^{(t)}|\mathbf{y}^{(t)}, \mu, \sigma^2$ using, either

  - Gibbs steps, i.e., following (38) ;
  - P-MALA step, i.e., following (41) and (42);

end

Output: A collection of samples $\left\{\mathbf{y}^{(t)}, \mathbf{x}^{(t)}\right\}_{t=T_{\mathrm{bi}}+1}^{T_{\mathrm{MC}}}$ asymptotically distributed according to $f\left(\mathbf{y}, \mathbf{x}|\sigma^2, \mu\right)$

later step is achieved using either the component-wise Gibbs sampler or P-MALA technique described in paragraph III-B3. Within this framework, the generated samples $\mathbf{x}^{(t)}$ should be asymptotically distributed according to the prior democratic distribution $f\left(\mathbf{x}|\mu\right)$. Thus they can be exploited to specifically assess the validity of this step, by resorting to the properties of this distribution, see Section II. In this experiment, we propose to focus on one of these properties: the absolute value of the dominant component $f\left(|x_n|\big|\mu, x_n \in \mathcal{C}_n\right)$ follows the gamma distribution $\mathcal{G}(N, \lambda)$ with $\lambda = N\mu$, see (12). Figure 6 (left) compares the theoretical pdf of this gamma distribution with the empirical pdfs computed from $T_{\mathrm{MC}} = 2 \times 10^4$ samples generated by the Geweke's scheme with the Gibbs (top) and P-MALA (bottom) samplers, where $M = N = 3$, $\mu = 2$ and $\sigma^2 = 0.25$. Figure 6 (right) shows the corresponding quantile-quantile (Q-Q) plots which tends to ascertain the validity of the two versions of the MCMC algorithm.

B. Performance analysis on a toy example

This paragraph focuses on a toy example to illustrate the convergence of the two versions of the proposed algorithm, i.e., based on Gibbs or P-MALA steps detailed in paragraphs III-B3 and III-B3, respectively. To this end, a simple experimental set-up with $M = N = 16$ has been considered where anti-sparse vectors $\mathbf{x}$ have been generated with coefficients randomly chosen among $\{-N^{-1}, +N^{-1}\}$. Observation vectors are then obtained following the free-noise forward

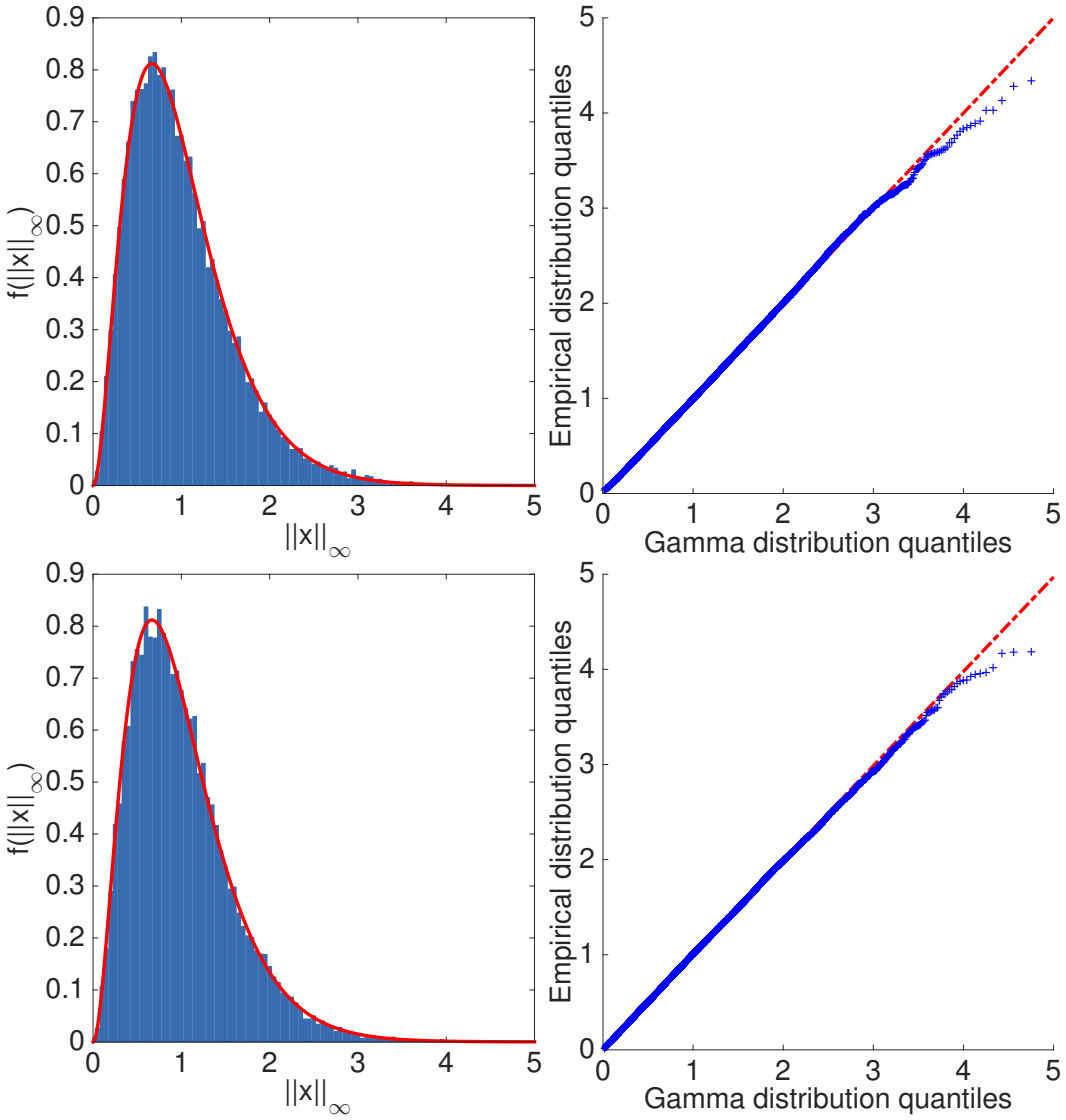

Fig. 6. Left: theoretical Gamma pdf of the dominant component absolute value (red) and empirical pdf (blue) resulting from 2×10^4 samples generated by successive conditional sampling (Algo. 6) with the Gibbs sampler (top) and P-MALA (bottom). Right: corresponding Q-Q plots.

model $\mathbf{y} = \mathbf{H}\mathbf{x}$.

The proposed MCMC algorithms are used to generate samples $(\mathbf{x}^{(t)}, \sigma^{(t)}, \mu^{(t)})_{t=T_{\mathrm{bi}}}^{T_{\mathrm{MC}}}$ asymptotically distributed according to the joint posterior distribution (34) with $T_{\mathrm{MC}} = 3000$ iterations including $T_{\mathrm{bi}} = 2000$ burn-in iterations. The MMSE and mMAP estimators of representation vector have been approximated from these samples following the strategy described in paragraph

III-C. Two criteria have been used to evaluate the performance of these estimators

$$\text{SNR}_{\mathbf{x}} = 10\log_{10}\frac{\|\mathbf{x}\|_2^2}{\|\mathbf{x}-\hat{\mathbf{x}}\|_2^2} \tag{49}$$

$$\text{PAPR} = \frac{N\,\|\hat{\mathbf{x}}\|_\infty^2}{\|\hat{\mathbf{x}}\|_2^2} \tag{50}$$

where $\hat{\mathbf{x}}$ refers to the MMSE or mMAP estimator of $\mathbf{x}$. The signal-to-noise ratio $\text{SNR}_{\mathbf{x}}$ measures the quality of the estimation with respect to the unknown (democratic) signal $\mathbf{x}$. Conversely, the peak-to-average power ratio PAPR quantifies anti-sparsity by measuring the ratio between the crest of the estimated signal and its average value. Note that the proposed algorithms do not aim at directly minimizing the PAPR: the use of a democratic distribution prior should promote anti-sparsity and therefore estimates with low PAPR.

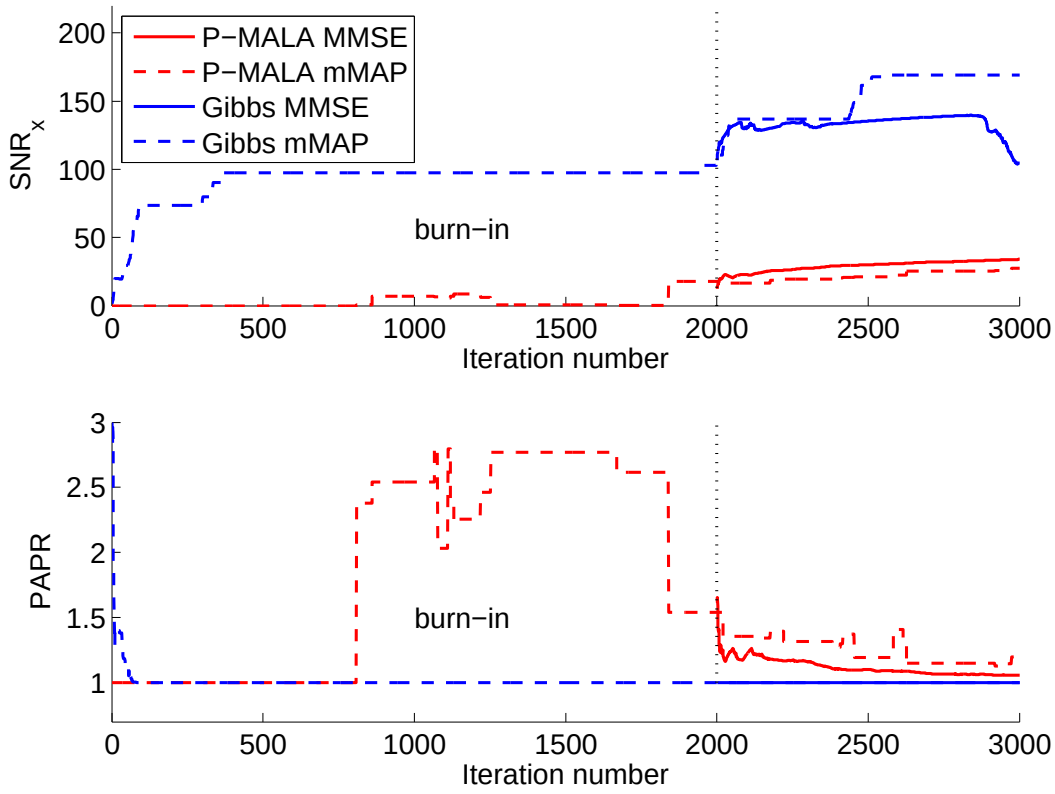

Fig. 7. As functions of the iteration number, $\text{SNR}_{\mathbf{x}}$ (top) and PAPR (bottom) associated with mMAP (dashed lines) and MMSE (continuous lines) estimates computed using the proposed algorithm based on Gibbs steps (blue) and P-MALA steps (red). The end of the burn-in period is localized with a vertical black dotted line.

Figure 7 shows the evolution of both criteria through iterations for both versions of the

algorithm. Note that, according to (46) and (48), the mMAP estimators are approximated from all the generated samples while the MMSE estimates are only computed from samples generated after the burn-in period, located with a vertical line. The plots associated with the mMAP estimates show that, after less than 200 iterations, the Gibbs sampler generates vectors with PAPR lower that 1.05 and $\text{SNR}_{\mathbf{x}}$ higher than 75dB. The MMSE estimator computed from these samples quickly converges to similar results (after the burn-in period). Conversely, the estimators approximated from the samples generated using the P-MALA steps need 10 times more iterations to converge towards solutions with PAPR around 1.2 and $\text{SNR}_{\mathbf{x}}$ close to 40dB. However, considering the computational time with a personal computer equipped with a 2.8Ghz Intel i5 processor, the simulation of 3000 samples requires 20 seconds using Gibbs sampling and only 2 seconds using P-MALA steps. These observations highlight the fact that the algorithm based on P-MALA steps is much faster even though it needs more samples than the full Gibbs sampler to build robust estimators. To alleviate this limitation, the strategy adopted in the next experiments performs 20 Metropolis-Hastings moves (42) within a single iteration of the MCMC algorithm (as recommended in [33]).

C. Performance comparison

1) Experimental set-up: In this experiment, the observation vector $\mathbf{y}$ is composed of coefficients independently and identically distributed according to a Gaussian distribution, as in [28]. The proposed MCMC algorithm is applied to infer the anti-sparse representation $\mathbf{x}$ of this measurement vector $\mathbf{y}$ with respect to the $M \times N$ coding matrix $\mathbf{H}$ for two distinct scenarios. Scenario 1 considers a small dimension problem with $M = 50$ and $N = 70$. In Scenario 2, a higher dimension problem has been addressed, i.e., with $M = 128$ and N ranging from 128 to 256, which permits to evaluate the performance of the algorithm as a function of the ratio N/M. In Scenario 1 (resp., Scenario 2), the proposed mMAP and MMSE estimators are computed from a total of $T_{\text{MC}} = 12 \times 10^3$ (resp., $T_{\text{MC}} = 55 \times 10^3$) iterations, including $T_{\text{bi}} = 10 \times 10^3$ (resp., $T_{\text{bi}} = 50 \times 10^3$) burn-in iterations. For this latest scenario, the algorithm based on Gibbs steps, see paragraph III-B3, has not been considered because of its computational burden, which experimentally justifies the interest of the proximal MCMC-based approach for large scale problems.

Algorithm performances have been evaluated over 20 Monte Carlo simulations thanks to two

figures-of-merit. The anti-sparse level of the recovered representation vector $\hat{\mathbf{x}}$ has been measured using the PAPR defined in (50). Since the actual representation vector $\mathbf{x}$ is not available anymore (as in the previous experiment), the reconstruction error denoted $\text{SNR}_{\mathbf{y}}$ and defined by

$$\text{SNR}_{\mathbf{y}} = 10 \log_{10} \frac{\|\mathbf{y}\|_2^2}{\|\mathbf{y} - \mathbf{H}\hat{\mathbf{x}}\|_2^2}. \tag{51}$$

is the second figure-of-merit.

The proposed algorithm is compared with a recent PAPR reduction technique, detailed in [31]. This fast iterative truncation algorithm (FITRA) is a deterministic counterpart of the proposed MCMC algorithm and solves the ℓ_∞-penalized least-squares problem (1). Similarly to various variational techniques, FITRA needs the prior knowledge of the hyperparameters λ (anti-sparsity level) and σ^2 (residual variance) or, equivalently, of the regularization parameter β defined (up to a constant) as the product of the two hyperparameters, i.e., $\beta \triangleq 2\lambda\sigma^2$. As a consequence, in the following experiments, this parameter β has been chosen according to 3 distinct rules. The first one, denoted FITRA-mmse, consists of applying FITRA with $\beta = 2\hat{\lambda}_{\text{MMSE}}\hat{\sigma}^2_{\text{MMSE}}$, where $\hat{\lambda}_{\text{MMSE}}$ and $\hat{\sigma}^2_{\text{MMSE}}$ are the MMSE estimates obtained with the proposed P-MALA based algorithm. In the second and third configurations, the regularization parameter β has been tuned to reach two solutions corresponding to either a target reconstruction error $\text{SNR}_{\mathbf{y}} = 20$dB (and free PAPR) or a target anti-sparsity level $\text{PAPR} = 1.5$ (and free $\text{SNR}_{\mathbf{y}}$), denoted FITRA-snr and FITRA-papr, respectively. For all these configurations, FITRA has been run with a maximum of 500 iterations. Moreover, to illustrate the regularizing effect of the democratic prior (or, similarly, the ℓ_∞-penalization), the proposed algorithm and the 3 configurations of FITRA have been finally compared with the least-squares (LS) solution as well as the MMSE and mMAP estimates resulting from a Bayesian model based on a Gaussian prior (or, similarly, an ℓ_2-penalization).

2) Results: Table III shows the results in Scenario 1 ($M = 50$ and $N = 70$) for all considered algorithms in terms of $\text{SNR}_{\mathbf{y}}$ and PAPR. For this scenario, the full Gibbs method needs approximately 6 minutes while P-MALA needs 8 seconds only. The mMAP and the MMSE estimates provided by P-MALA reach reconstruction errors of $\text{SNR}_{\mathbf{y}} = 23.7$dB and $\text{SNR}_{\mathbf{y}} = 22.4$dB, respectively. The mMAP estimate obtained using the full Gibbs sampler performs quite similarly while numerous estimates from the Gibbs MMSE have converged to solutions that do not ensure correct reconstruction, which explains worse $\text{SNR}_{\mathbf{y}}$ results. This is the signature of an unstable behavior of the Gibbs MMSE estimate. When using FITRA-mmse,

solutions with similar $\text{SNR}_{\mathbf{y}}$ but lower PAPR are recovered. Both MCMC and FITRA algorithms have provided anti-sparse representations with lower PAPR than LS or ℓ_2-penalized solutions, which confirms the interest of the democratic prior or, equivalently, the ℓ_∞-penalization.

TABLE III

SCENARIO 1: RESULTS IN TERMS OF $\text{SNR}_{\mathbf{y}}$ AND PAPR FOR VARIOUS ALGORITHMS.

	$\text{SNR}_{\mathbf{y}}$	PAPR
P-MALA MMSE	22.4	3.69
P-MALA mMAP	23.7	2.82
Gibbs MMSE	9.6	3.69
Gibbs mMAP	12.7	2.82
FITRA-mmse	21.8	1.53
FITRA-snr	19.9	1.86
FITRA-papr	9.3	1.5
LS	306.4	7.27
Gaussian prior MMSE	81.4	7.04
Gaussian prior mMAP	154.6	6.92

Fig. 8 displays the results for a given realization of the measurement vector $\mathbf{y}$ where the $\text{SNR}_{\mathbf{y}}$ is plotted as a function of PAPR. To provide a whole characterization of FITRA and illustrate the trade-off between the expected reconstruction error and anti-sparsity level, the solutions provided by FITRA corresponding to a wide range of regularization parameter β are shown. The mMAP and MMSE solutions recovered by the two versions of the proposed algorithm are also reported in this $\text{SNR}_{\mathbf{y}}$ vs. PAPR plot. They are located close to the critical region between solutions with low PAPR and $\text{SNR}_{\mathbf{y}}$ and solutions with high PAPR and $\text{SNR}_{\mathbf{y}}$: the proposed method recovers relevant solutions in an unsupervised way. While Gibbs estimates seem to reach a better compromise than P-MALA ones, we emphasize that the Gibbs sampler is in fact relatively unstable and therefore less robust. Moreover, the Gibbs sampler does not scale to high dimensions due to its prohibitive computational cost.

Scenario 2 permits to evaluate the performances of the algorithm as a function of the ratio N/M. For measurement vectors of fixed dimension $M = 128$, the anti-sparse coding algorithms aim at recovering representation vectors of increasing dimensions $N = 128, \ldots, 256$. As a

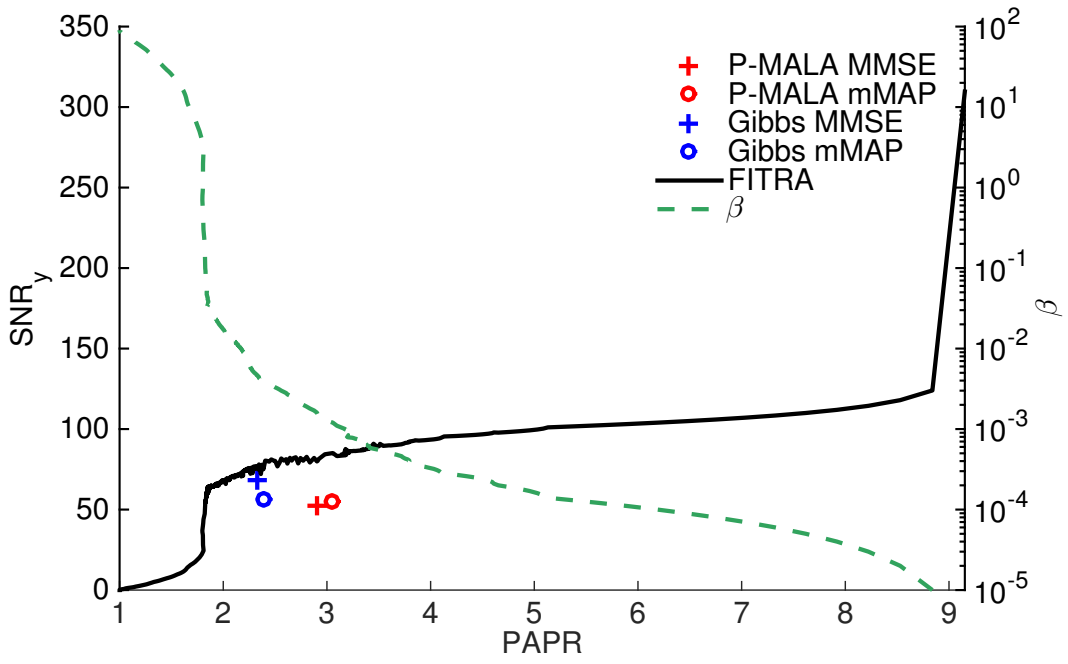

Fig. 8. Scenario 1: $\mathrm{SNR}_{\mathbf{y}}$ as a function of PAPR. The path for the FITRA regularization parameter β is depicted as green dashed line with the scale in the left y-axis.

consequence, for a given PAPR level of anti-sparsity, the $\mathrm{SNR}_{\mathbf{y}}$ is expected to be an increasing function of N/M. Fig. 9 confirms this intuition since the performance of FITRA-papr in terms of $\mathrm{SNR}_{\mathbf{y}}$ (slightly) increases when N/M increases. Conversely, for a given level $\mathrm{SNR}_{\mathbf{y}}$ of reconstruction error, the PAPR is expected to be a decreasing function of this ratio. Fig. 10 shows that the PAPR of FITRA-snrdecreases when N/M increases. Moreover, Fig. 9 shows that, in term of $\mathrm{SNR}_{\mathbf{y}}$, the behavior of the P-MALA mMAP and MMSE estimates is similar to FITRA-mmse's. However, they are able to achieve lower PAPR once the ratio N/M is greater than 1.3 and 1.4, respectively, as illustrated in Fig. 10.

V. Conclusion

This paper introduced a fully Bayesian framework for anti-sparse coding of a given measurement vector on a known and potentially over-complete dictionary. To derive a Bayesian formulation of the problem, a new probability distribution was introduced. Various properties of this so-called *democratic distribution* were exhibited, which permitted to design an exact random variate generator as well as two MCMC-based methods. This distribution was used as a

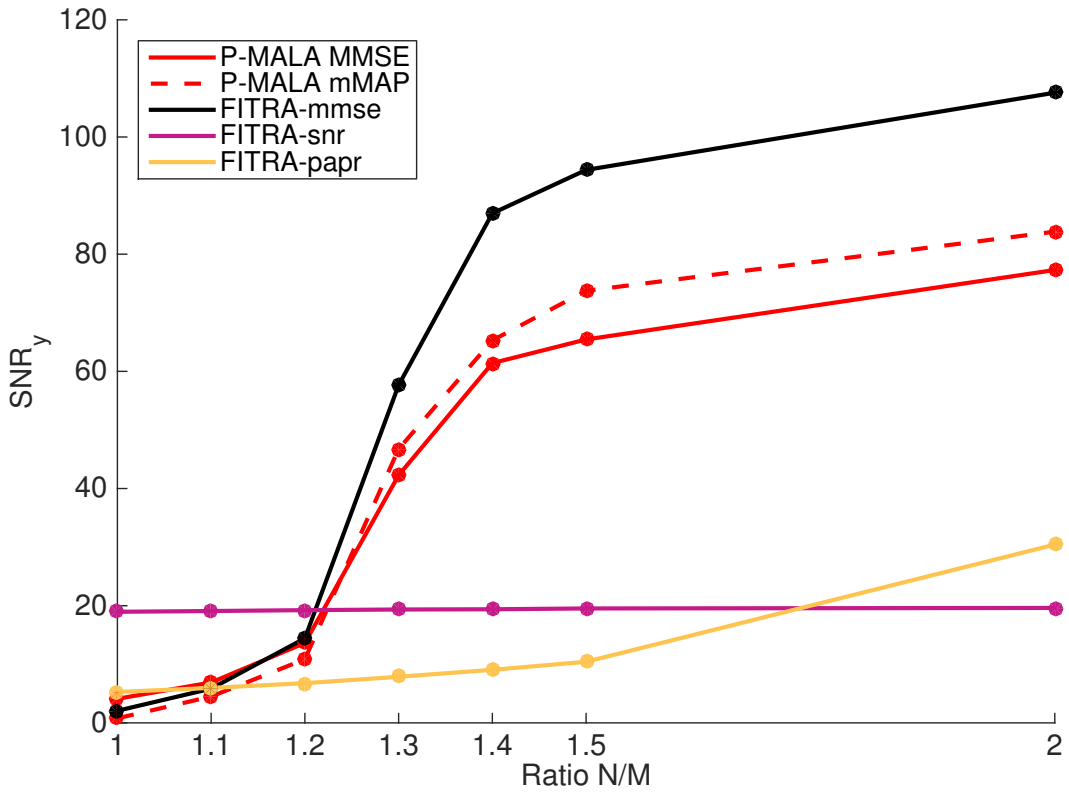

Fig. 9. Scenario 2: $\mathrm{SNR}_{\mathbf{y}}$ as a function of the ratio N/M.

prior for the representation vector in a linear Gaussian inverse problem, a probabilistic version of the anti-sparse coding problem. The residual variance as well as the anti-sparsity level were included in a fully Bayesian model, to be estimated jointly with the anti-sparse code. A Gibbs sampler was derived to generate samples distributed according to the joint posterior distribution of the coefficients of representation, the residual variance and the anti-sparse level. A second sampler was also proposed to scale to higher dimensions. To this purpose, the proximity mapping of the ℓ_∞-norm was considered to design a P-MALA within Gibbs algorithm. The generated samples were used to approximate two Bayesian estimators of the representation vector, namely the MMSE and mMAP estimators.

The validity of the proposed algorithms was first assessed following the successive conditional sampling scheme proposed in [36]. Then, they were evaluated through various experiments, and compared with FITRA a variational counterpart of the proposed MCMC algorithms. While fully unsupervised, they produced solutions comparable to FITRA in terms of reconstruction error and

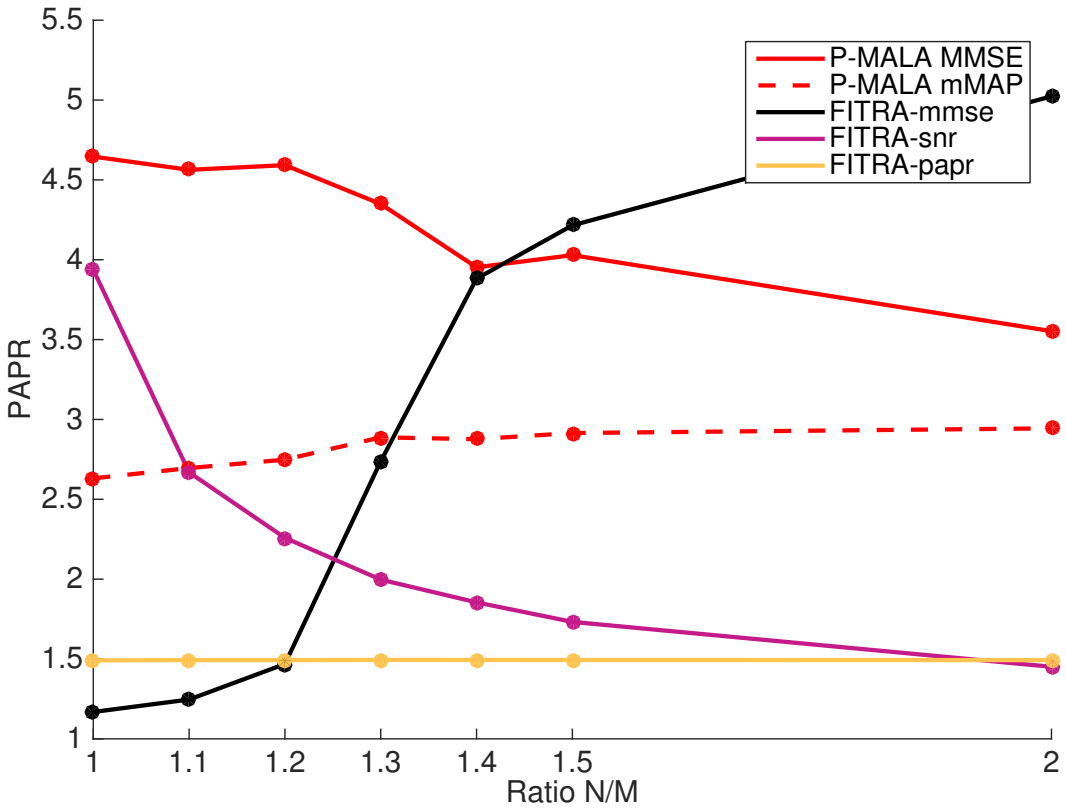

Fig. 10. Scenario 2: PAPR as a function of the ratio N/M.

PAPR, with the noticeable advantage to be fully unsupervised. In all experiments, as expected, the democratic prior distribution, was able to promote anti-sparse solutions of the coding problem. For that specific task, the mMAP estimator generally provided more relevant solutions than the MMSE estimator. Moreover, the P-MALA-based algorithm seemed to be more robust than the full Gibbs sampler and had the ability to scale to high dimension problems, both in term of computational times and performances.

Future works include the unsupervised estimation of the coding matrix jointly with the sparse code. This would open the door to the design of encoding matrices that would ensure equal spreading of the information over their atoms. Furthermore, since the P-MALA based sampler showed promising results, it would be relevant to investigate the geometric ergodicity of the sampler. Unlike most of the illustrative examples considered in [33], this property can not be easily stated for the democratic distribution since it is not $\mathcal{C}^1$ but only $\mathcal{C}^0$.

ACKNOWLEDGMENTS

The authors would like to thank Dr. Vincent Mazet, Université de Strasbourg, France, for providing the code to draw according to truncated Gaussian distributions following [35].

APPENDIX A
PROPERNESS OF THE DEMOCRATIC DISTRIBUTION

In this appendix, we provide a proof of Lemma 1. Let f be the function defined by $f : \mathbf{x} \to \exp\left(-\lambda \left\|\mathbf{x}\right\|_\infty\right)$. First, one can remark that

$$\forall \mathbf{x} \in \mathbb{R}^N,\ 0 \leq f(\mathbf{x}) \leq \exp\left(-\lambda \frac{1}{N} \sum_n^N |x_n|\right).$$

Since the sum of the upper bound function is finite, the integral of f is properly defined.

The following details the evaluation of the integral. To this aim, we will take advantage of several properness of the function of interest. First f is even, so domain can be reduced to $\mathbb{R}_+^N$. Then, by consider the set of cones introduced by (10), let's denote for each n $\mathcal{C}_n^+$ the trucation of $\mathcal{C}_n$ to $\mathbb{R}_+^N$. By means of the symetries, the integral of f on each half positive cone is equal. It follows

$$\begin{aligned}
C_N(\lambda) &= 2^N \int_{\cup_{n=1}^N \mathcal{C}_n^+} \exp\left(-\lambda \left\|\mathbf{x}\right\|_\infty\right) d\mathbf{x} \\
&= 2^N N \int_{\mathcal{C}_1^+} \exp\left(-\lambda x_1\right) d\mathbf{x} \\
&= 2^N N \int_{x_1=0}^{+\infty} \int_{x_2 \ldots x_N=0}^{x_1} \exp\left(-\lambda x_1\right) d\mathbf{x} \\
&= 2^N N \int_{x_1=0}^{+\infty} x_1^{N-1} \exp\left(-\lambda x_1\right) dx_1 \\
&= \frac{2^N N!}{\lambda^{N-1}} \int_{x_1=0}^{+\infty} \exp\left(-\lambda x_1\right) dx_1 \\
&= \frac{2^N N!}{\lambda^N}
\end{aligned}$$

APPENDIX B
TWO FIRST MODES OF THE DEMOCRATIC DISTRIBUTION

Let $\mathbf{x} = [x_1, \ldots, x_N]^T$ obeying the democratic distribution $\mathcal{D}_N(\lambda)$.

Different approaches are available to compute the mean. The fastest one consists in noticing that the function of interest, $\mathbf{x} \to x\ p\left(x\right)$ is odd and the integration is symmetric with respect to 0.

To compute the variance, a possible solution consists in using the marginal distribution (8), which is a sum of N double-sided Gamma distributions whose first parameter goes from 1 to N. Hence, using (8) and for $n \in [1N]$

$$\begin{aligned}
\mathrm{Var}(X_n) &= \int_{\mathbb{R}} x_n^2 p\left(x_n\right) \mathrm{d}\, x_n \\
&= \frac{1}{N} \sum_{i=1}^{N} \int_{\mathbb{R}} x_n^2\ p\left(Y_i = x_n | Y_i \sim d\mathcal{G}\left(i, \lambda\right)\right) \mathrm{d}\, x_n \\
&= \frac{1}{N} \sum_{i=1}^{N} \int_{\mathbb{R}} x_n^2\ p\left(Y_i = x_n | Y_i \sim d\mathcal{G}\left(i, \lambda\right)\right) \mathrm{d}\, x_n \\
&= \frac{1}{N} \sum_{i=1}^{N} \mathrm{Var}(Y_i | Y_i \sim d\mathcal{G}\left(i, \lambda\right)) \\
&= \frac{1}{N} \sum_{i=1}^{N} \frac{i(i+1)}{\lambda^2} \\
&= \frac{1}{N\lambda^2} \frac{N(N+1)}{2} + \frac{N(N+1)(2N+1)}{6} \\
&= \frac{(N+1)(N+2)}{3\lambda^2}
\end{aligned}$$

For more details about the variance of the double-sided Gamma distribution, see **??**

For n_1 and n_2 such that $n_1 \neq n_2$, the covariance can be computing following the same scheme, using (8) proposed in Lemma 2

$$\mathrm{Cov}\left[X_{n_1}, X_{n_2}\right] = \int_{\mathbb{R}^2} x_{n_1} x_{n_2}\, \mathrm{d}\, x_{n1}\, \mathrm{d}\, x_{n2}.$$

Since the marginal distribution $p\left(X_{n_1}, X_{n_2}\right)$ is expressed in terms of $\| \ \|_\infty$, the integration

domain will again be split into cones, leading to

$$\begin{aligned}
&\int_{\mathcal{C}_1^+} x_{n_1} x_{n_2} p\left(x_{n_1} x_{n_2}\right) \mathrm{d}\, x_{n1}\, \mathrm{d}\, x_{n2} \\
&= \int_{x_{n_1}=0}^{+\infty} \int_{x_{n_2}=-\left|x_{n_1}\right|}^{\left|x_{n_1}\right|} x_{n_1} x_{n_2} p\left(x_{n_1} x_{n_2}\right) \mathrm{d}\, x_{n_1}\, \mathrm{d}\, x_{n_2} \\
&= \frac{C_{N-2}(\lambda)}{C_N(\lambda)} \sum_{j=0}^{N-2} \frac{\lambda^j}{j!} \int_{x_{n_1}=0}^{+\infty} \int_{x_{n_2}=-\left|x_{n_1}\right|}^{\left|x_{n_1}\right|} x_{n_1} x_{n_2} \left|x_{n_1}\right|^j \\
&\qquad \times \exp\left(-\lambda \left|x_{n_1}\right|\right) \mathrm{d}\, x_{n_1}\, \mathrm{d}\, x_{n_2} \\
&= \frac{C_{N-2}(\lambda)}{C_N(\lambda)} \sum_{j=0}^{N-2} \frac{\lambda^j}{j!} \int_{x_{n_1}=0}^{+\infty} x_{n_1} \left|x_{n_1}\right|^j \times \exp\left(-\lambda \left|x_{n_1}\right|\right) \\
&\qquad \times \underbrace{\left(\int_{-|x_1|}^{+|x_1|} \frac{1}{2} x_2^2 \, \mathrm{d}\, x_{n_2} \right)}_{0} \mathrm{d}\, x_{n_1} \\
&= 0
\end{aligned}$$

Similarly, the same result is obtained when integrating over $\mathcal{C}_1^-$, $\mathcal{C}_2^+$ and $\mathcal{C}_2^-$.

APPENDIX C
MARGINAL DISTRIBUTIONS ASSOCIATED WITH $\mathcal{D}_N(\lambda)$

This appendix derives the marginal distribution of any subset, exhibited by Lemma 2. Let $\mathbf{x} = [x_1, \ldots, x_N]^T$ be a random vector drawn from the democratic distribution $\mathcal{D}_N(\lambda)$, $J < N$ be a positive integer, $\mathcal{K}_J$ a J-element subset of $\{1, \ldots, N\}$ and $\mathbf{x}_{\setminus \mathcal{K}_J}$ the sub-vector of $\mathbf{x}$ whose J elements indexed by $\mathcal{K}_J$ have been removed.

First

$$\begin{aligned}
p\left(\mathbf{x}_{\setminus \mathcal{K}_J}\right) &= \int_{\mathbb{R}^J} \frac{1}{C_N(\lambda)} p\left(\mathbf{x}\right) \mathrm{d}\, x_j^{\otimes j \in \mathcal{K}_J} \\
&= \frac{2^J}{C_N(\lambda)} \int_{\mathbb{R}_+^J} \exp\left(-\lambda \left\|\mathbf{x}\right\|_\infty\right) \mathrm{d}\, x_j^{\otimes j \in \mathcal{K}_J}
\end{aligned} \tag{52}$$

As in Appendix A, dividing $\mathbb{R}_+^J$ will ease computation but here, in some subsets of $\mathbb{R}_+^J$, $\left\|\mathbf{x}\right\|_\infty$ will be driven by $\left\|\mathbf{x}_{\setminus \mathcal{K}_J}\right\|_\infty$. The proposed strategy consists in separating the hyper-square $\left[0, \left\|\mathbf{x}_{\setminus \mathcal{K}_J}\right\|_\infty\right]^J$, where all marginalized variables are dominated by $\left\|\mathbf{x}_{\setminus \mathcal{K}_J}\right\|_\infty$, and the

complementary where computation can be done as in Appendix A. The trick consists in properly tessellating $\mathbb{R}_+^J$.

Let consider, for all positive integer j lower than J and any real positive a, the set of measurable rectangles $\mathcal{R}_j^J(a)$, where j are bounded by 0 and a, and all other lie between a and $+\infty$. For instance, an element of $\mathcal{R}_2^3(a)$ is $[0,a]\times[a,+\infty]\times[0,a]$. One can show that

$$\mathbb{R}_+^J = \bigcup_{j=0}^{J} \left(\bigcup_{R\in\mathcal{R}_j^J\left(\left\|\mathbf{x}_{\setminus\mathcal{K}_J}\right\|_\infty\right)} R \right) \tag{53}$$

This tessellation is not regular, but it has been built such that all measurable rectangles are disjoint. Furthermore, the symmetries of the probability distribution makes, for a given j, the sum over any element of $\mathcal{R}_j^J\left(\left\|\mathbf{x}_{\setminus\mathcal{K}_J}\right\|_\infty\right)$ equal, which means that there is $J+1$ different values combined with the cardinal of each set. For $j=J$, the measurable rectangle is a square, and the sum is directly the integration of a constant function over a cuboid

$$\begin{aligned}\int_{\left[0,\left\|\mathbf{x}_{\setminus\mathcal{K}_J}\right\|_\infty\right]^J} &\exp\left(-\lambda\left\|\mathbf{x}\right\|_\infty\right)\mathrm{d}\,x_j^{\otimes j\in\mathcal{K}_J} \\ &= \left\|\mathbf{x}_{\setminus\mathcal{K}_J}\right\|_\infty^J \exp\left(-\lambda\left\|\mathbf{x}_{\setminus\mathcal{K}_J}\right\|_\infty\right)\end{aligned} \tag{54}$$

Now, for any integer j lower than J, integration is done as in Appendix A: intervals that go to infinity are separated in truncated cones, where a known dimension is dominant. Then $J-j-1$ integrations by parts help going back to an easier form. For a given j, let R denote an element of $\mathcal{R}_j^J\left(\left\|\mathbf{x}_{\setminus\mathcal{K}_J}\right\|_\infty\right)$. To make things more readable, dimensions $1\dots J$ are the ones which are marginalized, the $J-j$ first ones are summed up to infinity and the desired dominant dimension

is the first one

$$\begin{aligned}
&\int_R \exp\left(-\lambda \left\|\mathbf{x}\right\|_\infty\right) \mathrm{d}\, x_l^{\otimes l\in\mathcal{K}_J} \\
&= \quad (J-j) \int_{\substack{x_1 = \left\|\mathbf{x}_{\setminus\mathcal{K}_J}\right\|_\infty \\ x_2 \dots x_{J-j} \in [0,x_1]^{j-1}}}^{+\infty} \quad \int_{x_{J-j+1}\dots x_J=0}^{\left\|\mathbf{x}_{\setminus\mathcal{K}_J}\right\|_\infty} \exp\left(-\lambda x_1\right) \mathrm{d}\, x_l^{\otimes l\in\mathcal{K}_J} \\
&= (J-j) \left\|\mathbf{x}_{\setminus\mathcal{K}_J}\right\|_\infty^j \int_{x_1=\left\|\mathbf{x}_{\setminus\mathcal{K}_J}\right\|_\infty}^{+\infty} x_1^{J-j-1} \exp\left(-\lambda x_1\right) \mathrm{d}\, x_1 \\
&= \frac{(J-j)!}{\lambda^{J-j-1}} \left\|\mathbf{x}_{\setminus\mathcal{K}_J}\right\|_\infty^j \int_{x_1=\left\|\mathbf{x}_{\setminus\mathcal{K}_J}\right\|_\infty}^{+\infty} \exp\left(-\lambda x_1\right) \mathrm{d}\, x_1 \\
&= \frac{(J-j)!}{\lambda^{J-j}} \left\|\mathbf{x}_{\setminus\mathcal{K}_J}\right\|_\infty^j \exp\left(-\lambda \left\|\mathbf{x}_{\setminus\mathcal{K}_J}\right\|_\infty\right)
\end{aligned} \tag{55}$$

Next, the cardinality of each set is intimately linked to the number of permutations, which leads to

$$\#\mathcal{R}_j^J\left(\left\|\mathbf{x}_{\setminus\mathcal{K}_J}\right\|_\infty\right) = \binom{J}{j} \tag{56}$$

Finally, separating the integral as suggested in (53) and using (54), (55), (56) in (52) directly leads to the expected result.

APPENDIX D
THE DOUBLE-SIDED GAMMA DISTRIBUTION

The double-sided Gamma distribution $d\mathcal{G}\left(a,b\right)$ is defined as a generalization over $\mathbb{R}$ of the standard Gamma distribution $\mathcal{G}\left(a,b\right)$ with the following pdf :

$$\forall x \in \mathbb{R}, \quad f_{d\mathcal{G}}(x) = \frac{b^a}{2\Gamma(a)} \left|x\right|^{a-1} \exp\left(-b\left|x\right|\right) \tag{57}$$

Although the probability distributions are equal up to a multiplicative constant, the two distributions have different moments. Steps of computation are similar, but it is worthy to precise their values in the next property.

Property 6. *Let X be distributed according to the double-sided Gamma distribution $d\mathcal{G}\left(a,b\right)$. Then the following properties hold :*

$$\mathbf{E}\left[X\right] = 0 \tag{58}$$

$$\mathrm{Var}\left[X\right] = \frac{a(a+1)}{b^2} \tag{59}$$

Proof.

$$\mathbf{E}\left[X\right] = \int_{\mathbb{R}} x \frac{b^a}{2\Gamma(a)} \left|x\right|^{a-1} \exp\left(-b\left|x\right|\right) \mathrm{d}\, x$$

Since the function to be integrated is odd and the domain symmetric around 0, the mean is null.

Using the substitution $y = bx$ leads to

$$\begin{aligned}
\mathrm{Var}\left[X\right] &= \frac{b^a}{\Gamma(a)} \int_{\mathbb{R}_+} x^{a+1} \exp\left(-bx\right) \mathrm{d}\, x \\
&= \frac{1}{b^2\Gamma(a)} \int_{\mathbb{R}_+} y^{(a+2)-1} \exp\left(-y\right) \mathrm{d}\, y \\
&= \frac{1}{b^2\Gamma(a)} \Gamma(a+2) \\
&= \frac{a(a+1)}{b^2}
\end{aligned}$$

□

APPENDIX E
CONDITIONAL DISTRIBUTIONS ASSOCIATED WITH $\mathcal{D}_N\left(\lambda\right)$

A. Probability of x_n being a dominant component

The probability $\mathrm{P}\left[\mathbf{x} \in \mathcal{C}_n\right]$ can be obtained by computing the volume of the cone $\mathcal{C}_n$. The evaluation is done as followed, once again by exploiting intrinsic symmetries

$$\begin{aligned}
\mathrm{P}\left[\mathbf{x} \in \mathcal{C}_n\right] &= 2^N C_N(\lambda) \int_{x_n=0}^{+\infty} \int_{x_{\setminus n}=0}^{x_n} \exp\left(-\lambda x_n\right) \mathrm{d}x_1 \ldots \mathrm{d}\, x_N \\
&= 2^N C_N(\lambda) \int_{x_n=0}^{+\infty} x_1^{N-1} \exp\left(-\lambda x_n\right) \mathrm{d}\, x_1 \\
&= \int_{x_n=0}^{+\infty} \frac{\lambda}{N} \exp\left(-\lambda x_n\right) \mathrm{d}\, x_1 \\
&= \frac{1}{N}
\end{aligned}$$

B. Conditional distributions of the dominant or non-dominant components

This appendix is dedicated to the proof of the results (12), (13) and (14) exhibited in Lemma 3. Additionally, we will obtain (11) and (16).

First, the Bayes rule gives $p\left(x_n|\mathbf{x}\in\mathcal{C}_n\right)=\frac{p(x_n,\,\mathbf{x}\in\mathcal{C}_n)}{\mathrm{P}[\mathbf{x}\in\mathcal{C}_n]}$, where $p\left(\mathbf{x}\in\mathcal{C}_n\right)$ has been given by Property 3. Then, we marginalize over all other variables, which makes appear polygonal terms in x_n

$$\begin{aligned}p\left(x_n|\mathbf{x}\in\mathcal{C}_n\right)&=\frac{N2^{N-1}}{C_N(\lambda)}\int_{x_i=0,i\neq N}^{|x_n|}\exp\left(-\lambda\left|x_n\right|\right)\mathrm{d}\,x_{i\neq n}\\&=\frac{\lambda^N}{2(N-1)!}\left|x_n\right|^{N-1}\,\exp\left(-\lambda\left|x_n\right|\right)\end{aligned}$$

Which is exactly the double-sided Gamma distribution claimed in (12).

The two next results, equation (13) and (14), are obtained following the same scheme: the Bayes rule produces a truncated distribution, where some components are marginalized

$$\begin{aligned}p\left(\mathbf{x}_{\setminus n}|\mathbf{x}\in\mathcal{C}_n\right)&=\frac{N}{C_N(\lambda)}\int_{x_n=\left\|x_{\setminus n}\right\|_\infty}^{+\infty}\exp\left(-\left|x_n\right|\right)\mathrm{d}\,x_n\\&=\frac{1}{C_{N-1}(\lambda)}\exp\left(-\lambda\left\|x_{\setminus n}\right\|_\infty\right)\end{aligned}$$

which is a democratic distribution as claimed in (13).

For $x_j|x_n,\mathbf{x}\in\mathcal{C}_n$ we have

$$\begin{aligned}p\left(x_j|x_n,\mathbf{x}\in\mathcal{C}_n\right)&=\frac{p\left(x_j,x_n,\mathbf{x}\in\mathcal{C}_n\right)}{\mathrm{P}\left[\mathbf{x}\in\mathcal{C}_n\right]\;p\left(x_n|\mathbf{x}\in\mathcal{C}_n\right)}\\&=\frac{N\;2\;\Gamma(N)}{\lambda^N\left|x_n\right|^{N-1}\exp\left(-\lambda\left|x_n\right|\right)}\;\frac{1}{C_N(\lambda)}\\&\quad\times\int^{|x_n|}\exp\left(-\lambda\left|x_n\right|\right)\mathrm{d}\,x_{i,i\neq j,n}^{\otimes N-2}\\&\quad\begin{cases}x_i=-\left|x_n\right|\\\quad i\neq j,n\end{cases}\\&=\frac{1}{2\left|x_n\right|}\mathbf{1}_{|x_j|\leq|x_n|}(x_j),\end{aligned}$$

which leads to (14).

To prove (11), it is necessary to start from the definition of the conditional distribution

$$\begin{aligned} p\left(\mathbf{x} \in \mathcal{C}_n | \mathbf{x}_{\backslash n}\right) &= \frac{p\left(\mathbf{x} \in \mathcal{C}_n,\ \mathbf{x}_{\backslash n}\right)}{p\left(\mathbf{x}_{\backslash n}\right)} \\ &= \frac{1}{p\left(\mathbf{x}_{\backslash n}\right)} \times \int_{\mathcal{C}_n} p\left(\mathbf{x} | \mathbf{x} \sim \mathcal{D}_N(\lambda)\right) \mathrm{d}x_n \\ &= \frac{1}{p\left(\mathbf{x}_{\backslash n}\right)} \times \int_{\mathcal{C}_n} \frac{\lambda^N}{2^N N!} \exp\left(-\lambda \left\|\mathbf{x}\right\|_\infty\right) \mathrm{d}x_n \\ &= \frac{1}{p\left(\mathbf{x}_{\backslash n}\right)} \times 2 \int_{\left\|x_{\backslash n}\right\|_\infty}^{+\infty} \frac{\lambda^N}{2^N N!} \exp\left(-\lambda x_n\right) \mathrm{d}x_n \\ &= \frac{1}{p\left(\mathbf{x}_{\backslash n}\right)} \times \frac{\lambda^{N-1}}{2^{N-1} N!} \exp\left(-\lambda \left\|x_{\backslash n}\right\|_\infty\right) \end{aligned}$$

where the marginal distribution $p\left(\mathbf{x}_{\backslash n}\right)$ has been derived in (9). Once plugged, the computation directly leads to (11).

The last result to prove is $p\left(\mathbf{x}_{\backslash n} | \mathbf{x} \notin \mathcal{C}_n\right)$. The scheme is similar to the previous proof: the n-th component is marginalized after using the definition of the conditional distribution

$$\begin{aligned} p\left(\mathbf{x}_{\backslash n} | \mathbf{x} \notin \mathcal{C}_n\right) &= \frac{p\left(\mathbf{x}_{\backslash n},\ \mathbf{x} \notin \mathcal{C}_n\right)}{\mathrm{P}\left[\mathbf{x} \notin \mathcal{C}_n\right]} \\ &= \frac{1}{\mathrm{P}\left[\mathbf{x} \notin \mathcal{C}_n\right]} \times \int_{\mathcal{C}_n^c} p\left(\mathbf{x} | \mathbf{x} \sim \mathcal{D}_N(\lambda)\right) \mathrm{d}\, x_n \\ &= \frac{1}{\mathrm{P}\left[\mathbf{x} \notin \mathcal{C}_n\right]} \times 2 \int_0^{\left\|x_{\backslash n}\right\|_\infty} \frac{\lambda^N}{2^N N!} \exp\left(-\lambda \left\|x\right\|_\infty\right) \mathrm{d}\, x_n \\ &= \frac{1}{\mathrm{P}\left[\mathbf{x} \notin \mathcal{C}_n\right]} \times \frac{\lambda^{N-1}}{2^{N-1} N!} \left\|x_{\backslash n}\right\|_\infty \exp\left(-\lambda \left\|x_{\backslash n}\right\|_\infty\right) \end{aligned}$$

Then, $\mathrm{P}\left[\mathbf{x} \notin \mathcal{C}_n\right] = 1 - \mathrm{P}\left[\mathbf{x} \in \mathcal{C}_n\right] = \frac{N-1}{N}$ using Property 3. This directly leads to (16).

C. Full conditional distributions

This appendix describes how to compute the conditional distribution $p\left(x_n | \mathbf{x}_{\backslash n}\right)$. The proposed strategy consists in conditioning the probability by the event $x_n \in \mathcal{C}_n$. Hence

$$\begin{aligned} p\left(x_n | \mathbf{x}_{\backslash n}\right) = p\left(x_n | x_{\backslash n}, \mathbf{x} \in \mathcal{C}_n\right)\ \mathrm{P}\left[\mathbf{x} \in \mathcal{C}_n | x_{\backslash n}\right] \\ +\ p\left(x_n | x_{\backslash n}, \mathbf{x} \notin \mathcal{C}_n\right)\ \mathrm{P}\left[\mathbf{x} \notin \mathcal{C}_n | x_{\backslash n}\right] \end{aligned} \tag{60}$$

It remains to determine each of the four probabilities.

The use of two nested Bayes rules leads to

$$p\left(x_n|x_{\backslash n}, \mathbf{x} \in \mathcal{C}_n\right) = \frac{1}{p\left(x_{\backslash n}|\mathbf{x} \in \mathcal{C}_n\right) \mathrm{P}\left[\mathbf{x} \in \mathcal{C}_n\right]} p\left(x_{\backslash n}, x_n\right) \mathbf{1}_{\mathcal{C}_n}(\mathbf{x}) \tag{61}$$

where $p\left(x_{\backslash n}, x_n\right)$ is nothing else but the pdf of the democratic distribution. Because it is known here that $\mathbf{x}$ belongs to the cone $\mathcal{C}_n$, $\|\mathbf{x}\|_\infty$ can be replaced by its value $|x_n|$. Moreover, $\mathrm{P}\left[\mathbf{x} \in \mathcal{C}_n\right]$ has been given in (3). Then $p\left(x_{\backslash n}|\mathbf{x} \in \mathcal{C}_n\right)$ is computed as followed

$$\begin{aligned} p\left(x_{\backslash n}|\mathbf{x} \in \mathcal{C}_n\right) &= \int_{\mathbb{R}} \frac{1}{\mathrm{P}\left[\mathbf{x} \in \mathcal{C}_n\right]} p\left(\mathbf{x}, \mathbf{x} \in \mathcal{C}_n\right) \mathrm{d}\, x_n \\ &= N \int_{\mathcal{C}_n} p\left(\mathbf{x}|\mathbf{x} \sim \mathcal{D}_N\left(\lambda\right)\right) \mathrm{d}\, x_n \\ &= N \int_{\left\|x_{\backslash n}\right\|_\infty}^{+\infty} \frac{\lambda^N}{2^N N!} \exp\left(-\lambda\left|x_n\right|\right) \mathrm{d}\, x_n \\ &= \frac{\lambda^{(N-1)}}{2^{(N-1)}(N-1)!} \exp\left(-\lambda\left|x_n\right|\right) \end{aligned} \tag{62}$$

Straightforward computations when combining (2), (3) and (62) in (61) lead to

$$p\left(x_n|x_{\backslash n}, \mathbf{x} \in \mathcal{C}_n\right) = \frac{\lambda}{2} \exp\left(-\lambda(|x_n| - \left\|x_{\backslash n}\right\|_\infty)\right) \ \mathbf{1}_{|x_n| \geq \left\|x_{\backslash n}\right\|_\infty}(x_n). \tag{63}$$

According to equation (11)

$$\mathrm{P}\left[\mathbf{x} \in \mathcal{C}_n|x_{\backslash n}\right] = \frac{1}{1 + \lambda\left\|\mathbf{x}_{\backslash n}\right\|_\infty}.$$

The computation of $p\left(x_n|x_{\backslash n}, \mathbf{x} \notin \mathcal{C}_n\right)$ is done as in (61). Nested Bayes rules provide

$$p\left(x_n|x_{\backslash n}, \mathbf{x} \notin \mathcal{C}_n\right) = \frac{1}{p\left(x_{\backslash n}|\mathbf{x} \notin \mathcal{C}_n\right) \mathrm{P}\left[\mathbf{x} \notin \mathcal{C}_n\right]} p\left(x_{\backslash n}, x_n\right) \mathbf{1}_{\mathcal{C}_n^c}(\mathbf{x}). \tag{64}$$

The computation of $p\left(x_{\backslash n}|\mathbf{x} \notin \mathcal{C}_n\right)$, which is similar to (62), gives

$$\begin{aligned} &p\left(x_{\backslash n}|\mathbf{x} \notin \mathcal{C}_n\right) \\ &= \frac{\lambda\left\|x_{\backslash n}\right\|_\infty}{N-1} \frac{\lambda^{(N-1)}}{2^{(N-1)}(N-1)!} \exp\left(-\lambda\left|x_n\right|\right). \end{aligned} \tag{65}$$

As done for (63), inserting (65) in (64) leads to

$$p\left(x_n|x_{\backslash n}, \mathbf{x} \notin \mathcal{C}_n\right) = \frac{1}{2\left\|x_{\backslash n}\right\|_\infty} \ \mathbf{1}_{|x_n| \leq \left\|x_{\backslash n}\right\|_\infty}(x_n) \tag{66}$$

$p\left(\mathbf{x} \notin \mathcal{C}_n | x_{\backslash n}\right)$ can be computed directly or by using the Bayes rule associated with (9), (16) and (3). Nevertheless, one can notice

$$\begin{aligned} \mathrm{P}\left[\mathbf{x} \notin \mathcal{C}_n | x_{\backslash n}\right] &= 1 - \mathrm{P}\left[\mathbf{x} \in \mathcal{C}_n | x_{\backslash n}\right] \\ &= \frac{\lambda \left\|\mathbf{x}_{\backslash n}\right\|_\infty}{1 + \lambda \left\|\mathbf{x}_{\backslash n}\right\|_\infty} \end{aligned} \tag{67}$$

using (11).

Finally the conditional (17) is obtained combining respectively (63), (11), (66) and (67) as proposed in (60).

APPENDIX F
PROXIMITY OPERATOR OF $\mathbf{x} \mapsto \|\mathbf{x}\|_\infty$

To prove the algorithmic solution of the proximity operator of g_1 exhibited by Property 5, we start by finding an equivalent problem, using

$$\boldsymbol{\rho} = \mathrm{prox}_{g_1}^{\delta}(\mathbf{x}) \Leftrightarrow \begin{cases} \tilde{\boldsymbol{\rho}} = \mathrm{prox}_{g_1}^{\delta}(|\mathbf{x}|) \\ \forall n,\ \rho_n = \mathrm{sign}(x_n)\tilde{\rho_n} \end{cases}. \tag{68}$$

Thus, the problem is equivalent to evaluate the proximity operator in $|\mathbf{x}|$ while keeping in mind the sign of each x_n. The following lemma will help to simplify once more the problem

Lemma 4. *Let $\tilde{\boldsymbol{\rho}}$ be the proximity operator of g_1, evaluated in $|\mathbf{x}|$ with parameter δ. Then, the two following results hold*

$$\forall n, \qquad 0 \leq \tilde{\rho}_n \leq \|\mathbf{x}\|_\infty \tag{69}$$

$$\forall n, \qquad \tilde{\rho}_n \neq \|\tilde{\boldsymbol{\rho}}\|_\infty \ \Rightarrow\ \tilde{\rho}_n = |x_n| \tag{70}$$

Proof. Both results are straightforward when using two vectors: one that fits exactly the statement and the other equal to the first except one coordinate. Then this coordinate does not respect the statement. Evaluating the function to be minimized in both points shows that those which do not fit the statement can neither be a minimum, nor the proximity operator. □

Using (68), $\mathbf{x}$ can be reduced to a vector $\mathbf{x}$ where all coordinates are positive. We also consider now that the coefficients x_n are ordered, i.e., $x_1 > x_2 > \ldots x_N \geq 0$. The case when two elements

are equals, which is almost surely impossible from a stochastic point-of-view, can be treated by incorporating the number of time a value appears in $\mathbf{x}$. Those orders of multiplicity are denoted by $d_1 \dots d_J$ in Property 5. For sake of simplicity, we will consider that all the x_n are different here. The function to be minimized is also noted $h_{\mathbf{x}}$

$$\forall \boldsymbol{u} \in \mathbb{R}^N, \quad h_{\mathbf{x}}(\boldsymbol{u}) \;=\; g_1(\boldsymbol{u}) + \frac{1}{2\delta} \left\| \boldsymbol{u} - \mathbf{x} \right\|_2^2$$

Because of the $\|\cdot\|_\infty$ term, variations of $h_{\mathbf{x}}$ are not easy to derive. However, the minima of $h_{\mathbf{x}}$ are located in a more interesting subset. Let $S_n(\mathbf{x})$ denote the subset of $\mathbb{R}^N$ where all coordinates lower or equal to x_n are identical, while the others are equal to $x_{n+1}, \dots, x_N$

$$S_n(\mathbf{x}) = \left\{ \mathbf{u} \in \mathbb{R}_+^N, u_1 \dots u_n = t,\ t \in [|x_{n+1}|, |x_n|[, u_{n+1} = |x_{n+1}| \dots u_N = |x_N| \right\}. \tag{71}$$

Equations (69) and (70) from Lemma 4 show that the proximity operator of interest belongs to the disjoint union of the $S_n(\mathbf{x})$, noted $S(\mathbf{x}) : S(\mathbf{x}) = \bigcup_{n=1}^N S_n(\mathbf{x})$.

Since all $S_n(\mathbf{x})$ are disjoint, one can study $h_{\mathbf{x}}$ independently on each subset. Interestingly, each $S_n(\mathbf{x})$ can be parametrized by one element, noted t in (71).

The point is to use the mentioned re-parametrization, to reformulate the minimization of $h_{\mathbf{x}}$ into N minimizations of one-dimensional functions. Let $\tilde{h}_{\mathbf{x}}$ be the one-dimensional function defined on $S_{\mathbf{x}}$, and t the unique real number such that $t = \|\boldsymbol{u}\|_\infty$ where $\boldsymbol{u}$ is an element of $S_n(\mathbf{x})$

$$\forall n, \forall \boldsymbol{u} \in S_n(\mathbf{x}),\ \tilde{h}_{\mathbf{x}}(t) = h_{\mathbf{x}}(\boldsymbol{u}) = t + \frac{1}{2\delta} \sum_{l=1}^{n} (|x_l| - t)^2. \tag{72}$$

Thus $\tilde{h}_{\mathbf{x}}$ is polynomial on each $S_n(\mathbf{x})$ as well as continuous and differentiable. A short study of those polynomial functions leads to N potential minima, one per subset $S_n(\mathbf{x})$, denoted

$$\phi_\delta^n(\mathbf{x}) = \frac{1}{n} \left(\sum_{l=1}^{n} |x_l| - \delta \right). \tag{73}$$

Furthermore, $\tilde{h}_{\mathbf{x}}$ is decreasing on the left of each $\phi_\delta^n(\mathbf{x})$ and increasing on the right. Note that all $\phi_\delta^n(\mathbf{x})$ might not be defined, because out of $S_n(\mathbf{x})$.

In fact, only one $\phi_\delta^n(\mathbf{x})$ exists. The plan of the proof is as follows: starting with the first existing $\phi_\delta^n(\mathbf{x})$, we will show that all the other ones are not defined. Let n_0 be the first existing

$\phi_\delta^n(\mathbf{x})$. Then $\phi_\delta^{n+1}(\mathbf{x}) \dots \phi_\delta^N(\mathbf{x})$ are $N - n_0 - 1$ other potential candidates. The case where no one is defined will be treated at the end. By definition of n_0, $\phi_\delta^{n_0}(\mathbf{x}) \in [x_{n_0+1} x_{n_0}[$, so $\phi_\delta^{n_0}(\mathbf{x}) \geq x_{n_0+1}$. In that case

$$\begin{aligned} \phi_\delta^{n_0+1} &= \frac{1}{n_0+1}\left(n_0 \phi_\delta^{n_0} + x_{n_0+1}\right) \\ &\geq x_{n_0+1}. \end{aligned} \tag{74}$$

This results means $\phi_\delta^{n_0+1}$ does not belong to $S_{n_0+1}(\mathbf{x})$, so is not defined. By induction, it is possible to show that for all integer n greater than n_0, $\phi_\delta^n \geq x_n$, thus no one is defined. Since no one exists before n_0, only one $\phi_\delta^n(\mathbf{x})$ is defined and $\tilde{h}_\mathbf{x}$ has just one minimum. This leads to the proximal given in (20). In the case where n_0 does not exist, $\tilde{h}_\mathbf{x}$ is monotonically increasing, so the minimum is in 0, and $\text{prox}_{g_1}^\delta(\mathbf{x}) = \mathbf{0}$.

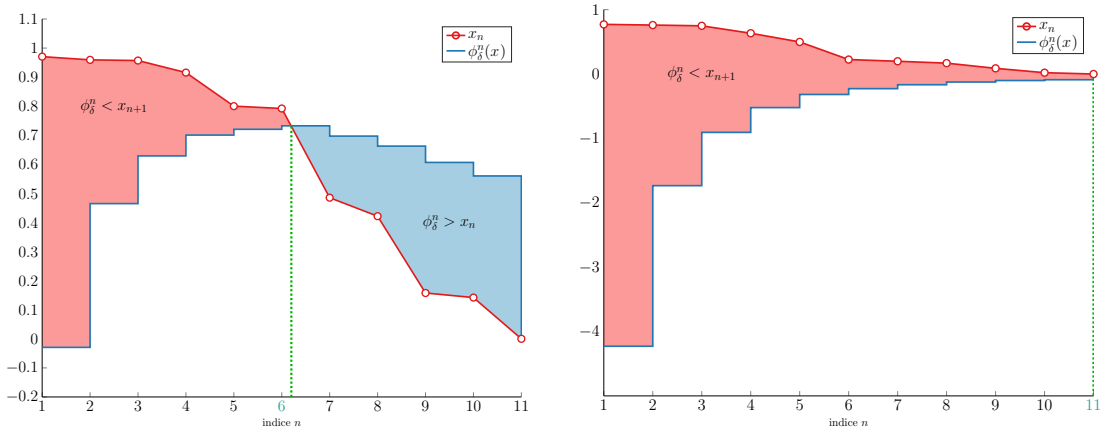

Fig. 11. Illustration of the bahaviour of the sequence $\phi_\delta^n(\mathbf{x})$ for two randomly choosen vectors $\mathbf{x}$ where components have been sorted such that $x_1 > x_2 \dots$. Left : The sequence of $\phi_\delta^n(\mathbf{x})$ for $\delta = 1$, depicted in blue line, is first increasing until a first $\phi_\delta^n(\mathbf{x})$ is defined since $\phi_\delta^n(\mathbf{x}) \notin [x_{n+1}, x_n[$. The sequence is then decreasing, and no more $\phi_\delta^n(\mathbf{x})$ is defined, for the same reason. Thus, the corresponding proximal operator value is ϕ_δ^6. Right : The same sequence, for $\lambda = 5$ never cross the sequence of x_n and is always below zero. Thus, the corresponding proximal operator is the null vector.

It is possible to obtain an algorithmic solution to find n_0, as claimed in (21). It comes from the monotonicity of the sequence $\phi_\delta^n(\mathbf{x})$. In fact, this sequence is first increasing until n_0 and then decreasing, as illustrated in Fig. 11. Indeed, by first rewriting

$$\phi_\delta^{n_0-1}(\mathbf{x}) = \frac{1}{n_0-1}\left(n_0 \phi_\delta^{n_0}(\mathbf{x}) - x_{n_0}\right).$$

Since $\phi_\delta^{n_0}(\mathbf{x}) \in [x_{n_0+1}, x_{n_0}[$, $-x_{n_0} < -\phi_\delta^{n_0}(\mathbf{x})$ then $\phi_\delta^{n_0-1}(\mathbf{x}) < \phi_\delta^{n_0}(\mathbf{x})$. Using this scheme, it is possible to show by induction, starting from n_0 to 1, the following property

$$P^1(n) \ : \ \phi_\delta^{n-1}(\mathbf{x}) \leq \phi_\delta^n(\mathbf{x}) \text{ and } \phi_\delta^{n-1}(\mathbf{x}) < x_{n-1}. \tag{75}$$

For $n \geq n_0$, the sequence decreases. This can be shown using $x_{n_0-1} \leq \phi_\delta^{n_0}$ and the following move as initialization

$$\phi_\delta^{n_0+1} = \frac{1}{n_0+1}\left(n_0\phi_\delta^{n_0} + x_{n_0+1}\right) \leq \phi_\delta^{n_0}.$$

Using and following the same scheme as in initialization, we can prove, for $n \geq n_0$

$$P^2(n) \ : \ \phi_\delta^{n+1}(\mathbf{x}) \leq \phi_\delta^n(\mathbf{x}) \text{ and } \phi_\delta^n(\mathbf{x}) \geq x_{n+1}. \tag{76}$$

The sequence of $(\phi_\delta^n(\mathbf{x}))$ is increasing until n_0, and decreasing after.

Hence, one solution to find $\phi_\delta(\mathbf{x}) = \phi_\delta^{n_0}(\mathbf{x})$ consists in computing all the $\phi_\delta^n(\mathbf{x})$, including 0, and taking the greatest, as stated in (21).

APPENDIX G
POSTERIOR DISTRIBUTION OF THE REPRESENTATION COEFFICIENTS

The (hidden) mean and variances of the truncated Gaussian distributions involved in the mixture distribution (38) are given by

$$\begin{aligned}
\mu_{1n} &= \frac{1}{\|\mathbf{h}_n\|^2}\left(\mathbf{h}_n^T\mathbf{e}_n + \sigma^2\lambda\right) \\
\mu_{2n} &= \frac{1}{\|\mathbf{h}_n\|^2}\left(\mathbf{h}_n^T\mathbf{e}_n\right) \\
\mu_{3n} &= \frac{1}{\|\mathbf{h}_n\|^2}\left(\mathbf{h}_n^T\mathbf{e}_n - \sigma^2\lambda\right)
\end{aligned}$$

and

$$s_n^2 = \frac{\sigma^2}{\|\mathbf{h}_n\|_2^2}$$

where $\mathbf{h}_i$ denotes the ith column of $\mathbf{H}$ and $\mathbf{e}_n = \mathbf{y} - \sum_{i\neq n} x_i\mathbf{h}_i$. Moreover, the weights associated with each mixture component are

$$\omega_{in} = \frac{u_{in}}{\sum_{j=1}^3 u_{jn}} \tag{77}$$

with

$$
\begin{aligned}
u_{1n} &= \exp\left(\frac{\mu_{1n}^2}{2s_n^2} + \lambda \left\|\mathbf{x}_{\backslash n}\right\|_\infty\right) \phi_{\mu_{1n},s_n^2}\left(-\left\|\mathbf{x}_{\backslash n}\right\|_\infty\right) \\
u_{2n} &= \exp\left(\frac{\mu_{2n}^2}{2s_n^2}\right) \\
&\quad \times \left[\phi_{\mu_{2n},s_n^2}\left(\left\|\mathbf{x}_{\backslash n}\right\|_\infty\right) - \phi_{\mu_{2n},s_n^2}\left(-\left\|\mathbf{x}_{\backslash n}\right\|_\infty\right)\right] \\
u_{3n} &= \exp\left(\frac{\mu_{3n}^2}{2s_n^2} + \lambda \left\|\mathbf{x}_{\backslash n}\right\|_\infty\right) \\
&\quad \times \left(1 - \phi_{\mu_{3n},s_n^2}\left(\left\|\mathbf{x}_{\backslash n}\right\|_\infty\right)\right)
\end{aligned}
$$

where $\phi_{\mu,s^2}(\cdot)$ is the cumulated distribution function of the normal distribution $\mathcal{N}(\mu, s^2)$.

Proof. Throughout the study, conditioning the probabilities to the membership to a cone has often eased computation. Property 4 has already made use of it to find the pdf of the conditional distribution of a given component x_n given $\mathbf{x}_{\backslash n}$. Hence, by invoking equation (17) from property 4

$$
\begin{aligned}
p\left(x_n|\mathbf{y}, \sigma^2, \mathbf{x}_{\backslash n}\right) &= p\left(\mathbf{y}|\mathbf{x}, \sigma^2\right) p\left(x_n|\mathbf{x}_{\backslash n}\right) \\
&= p\left(\mathbf{y}|\mathbf{x}, \sigma^2\right) \frac{\lambda \left\|\mathbf{x}_{\backslash n}\right\|_\infty}{1 + \lambda \left\|\mathbf{x}_{\backslash n}\right\|_\infty} \frac{1}{2 \left\|\mathbf{x}_{\backslash n}\right\|_\infty} \mathbf{1}_{\mathcal{I}_n}(x_n) \\
&\quad + p\left(\mathbf{y}|\mathbf{x}, \sigma^2\right) \frac{1}{1 + \lambda \left\|\mathbf{x}_{\backslash n}\right\|_\infty} \frac{\lambda}{2} e^{-\lambda\left(|x_n| + \left\|\mathbf{x}_{\backslash n}\right\|_\infty\right)} \mathbf{1}_{\mathbb{R}\backslash\mathcal{I}_n}(x_n).
\end{aligned} \tag{78}
$$

Where $p\left(\mathbf{y}|\mathbf{x}, \sigma^2\right)$ is the likelihood function and is given by (26). By the the presence of indicator functions, the posterior appears to be a mixture model. It is then mandatory to obtain the constant terms of the posterior to compute the coefficients of the mixture. By denoting $(\boldsymbol{h}_n)$ the set of N column of $\mathbf{H}$, one can classically split the exponential term in the likelihood function into two parts

$$
\begin{aligned}
\|\mathbf{y} - \mathbf{H}\mathbf{x}\|_2^2 = \left\|\mathbf{y} - x_n\boldsymbol{h}_n - (\sum_{i\neq n} x_i\boldsymbol{h}_i)\right\|_2^2 &= x_n^2 \|\boldsymbol{h}_n\|_2^2 - 2x_n\boldsymbol{h}_n^T(\mathbf{y} - \sum_{i\neq n} x_i\boldsymbol{h}_i) \\
&\quad + \left[\mathbf{y}^T\mathbf{y} - 2\mathbf{y}^T(\sum_{i\neq n} x_i\boldsymbol{h}_i) + (\sum_{i\neq n} x_i\boldsymbol{h}_i)^T(\sum_{i\neq n} x_i\boldsymbol{h}_i)\right].
\end{aligned} \tag{79}
$$

By remarking the terms under brackets does not depend on x_n, they will vanish in the Common constant. Then, let Φ be the constant term common to the two parts of the posterior

$$
\Phi = \left(\frac{1}{2\pi\sigma^2}\right)^{M/2} \exp\left[\frac{1}{2\sigma^2}\left(\mathbf{y}^T\mathbf{y} - 2\mathbf{y}^T(\sum_{i\neq n} x_i\boldsymbol{h}_i) + (\sum_{i\neq n} x_i\boldsymbol{h}_i)^T(\sum_{i\neq n} x_i\boldsymbol{h}_i)\right)\right]. \tag{80}
$$

By splitting the set $\mathbb{R}\backslash\mathcal{I}_n$ into $\mathbb{R}_-\backslash\mathcal{I}_n \bigcup \mathbb{R}_+\backslash\mathcal{I}_n$, the posterior takes the following form

$$p\left(x_n|\mathbf{y},\sigma^2,\mathbf{x}_{\backslash n}\right) = f_1(x_n)\mathbf{1}_{\mathbb{R}_-\backslash\mathcal{I}_n}(x_n) + f_2(x_n)\mathbf{1}_{\mathcal{I}_n}(x_n) + f_3(x_n)\mathbf{1}_{\mathbb{R}_+\backslash\mathcal{I}_n}(x_n).$$

Factorizing the previous expression directly leads to the mixture of truncated normal distributions.

□

REFERENCES

[1] C. Elvira, P. Chainais, and N. Dobigeon, "Bayesian anti-sparse coding," 2015, submitted.

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

[3] M. A. Davenport, M. F. Duarte, Y. C. Eldar, and G. Kutyniok, "Introduction to compressed sensing," in *Compressed Sensing: Theory and Applications*, Y. C. Eldar and G. Kutyniok, Eds. Cambridge, UK: Cambridge University Press, 2012, ch. 1, pp. 1–64.

[4] R. Gribonval, "Should penalized least squares regression be interpreted as maximum a posteriori estimation?" *IEEE Trans. Signal Process.*, vol. 59, no. 5, pp. 2405–2410, May 2011.

[5] R. Tibshirani, "Regression shrinkage and selection via the LASSO," *J. Roy. Stat. Soc. Ser. B*, vol. 58, no. 1, pp. 267–288, 1996.

[6] S. D. Babacan, R. Molina, and A. K. Katsaggelos, "Bayesian compressive sensing using Laplace priors," *IEEE Trans. Image Process.*, vol. 19, no. 1, pp. 53–63, Jan. 2010.

[7] F. Caron and A. Doucet, "Sparse Bayesian nonparametric regression," in *Proc. Int. Conf. Machine Learning (ICML)*, Helsinki, Finland, July 2008.

[8] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," *J. Mach. Learning Research*, vol. 1, pp. 211–244, 2001.

[9] A. Lee, F. Caron, A. Doucet, and C. Holmes, "A hierarchical Bayesian framework for constructing sparsity-inducing priors," *arXiv.org*, Sept. 2010.

[10] K. E. Themelis, A. A. Rontogiannis, and K. D. Koutroumbas, "A novel hierarchical Bayesian approach for sparse semisupervised hyperspectral unmixing," *IEEE Trans. Signal Process.*, vol. 60, no. 2, pp. 585–599, Feb. 2012.

[11] S. Ji, Y. Xue, and L. Carin, "Bayesian compressive sensing," *IEEE Trans. Signal Process.*, vol. 56, no. 6, pp. 2346–2356, June 2008.

[12] D. Tzikas, A. Likas, and N. Galatsanos, "Variational Bayesian sparse kernel-based blind image deconvolution with Student's-t priors," *IEEE Trans. Image Process.*, vol. 18, no. 4, pp. 753–764, April 2009.

[13] C. Févotte and S. J. Godsill, "A Bayesian approach for blind separation of sparse sources," *IEEE Trans. Audio, Speech, Language Process.*, vol. 14, no. 6, pp. 2174–2188, Nov. 2006.

[14] N. Dobigeon, A. O. Hero, and J.-Y. Tourneret, "Hierarchical Bayesian sparse image reconstruction with application to MRFM," *IEEE Trans. Image Process.*, vol. 18, no. 9, pp. 2059–2070, Sept. 2009.

[15] C. Soussen, J. Idier, D. Brie, and J. Duan, "From Bernoulli-Gaussian deconvolution to sparse signal restoration," *IEEE Trans. Signal Process.*, vol. 29, no. 10, pp. 4572–4584, Oct. 2011.

[16] L. Chaari, H. Batatia, N. Dobigeon, and J.-Y. Tourneret, "A hierarchical sparsity-smoothness Bayesian model for $\ell_0-\ell_1-\ell_2$ regularization," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing (ICASSP)*, Florence, Italy, May 2014, pp. 1901–1905.

[17] E. Remes, "Sur une propriété extrémale des polynômes de Tchebychef," *Communications de l'Institut des Sciences Mathématiques et Mécaniques de l'Université de Kharkoff et de la Société Mathématique de Kharkoff*, vol. 13, no. 1, pp. 93–95, 1936.

[18] T. W. Parks and J. H. McClellan, "Chebyshev approximation for nonrecursive digital filters with linear phase," *IEEE Trans. Circ. Theory*, vol. 19, no. 2, pp. 189–194, March 1972.

[19] J. H. McClellan and T. W. Parks, "A personal history of the Parks-McClellan algorithm," *IEEE Signal Process. Mag.*, vol. 22, no. 2, pp. 82–86, March 2005.

[20] L. W. Neustadt, "Minimum effort control systems," *J. SIAM Control*, vol. 1, no. 1, pp. 16–31, 1962.

[21] J. A. Cadzow, "Algorithm for the minimum-effort problem," *IEEE Trans. Autom. Contr.*, vol. 16, no. 1, pp. 60–63, 1971.

[22] Y. Lyubarskii and R. Vershynin, "Uncertainty principles and vector quantization," *IEEE Trans. Inf. Theory*, vol. 56, no. 7, pp. 3491–3501, July 2010.

[23] Z. Cvetković, "Resilience properties of redundant expansions under additive noise and quantization," *IEEE Trans. Inf. Theory*, vol. 29, no. 3, pp. 644–656, March 2003.

[24] A. R. Calderbank and I. Daubechies, "The pros and cons of democracy," *IEEE Trans. Inf. Theory*, vol. 48, no. 6, pp. 1721–1725, June 2002.

[25] B. Farrell and P. Jung, "A Kashin approach to the capacity of the discrete amplitude constrained Gaussian channel," in *Proc. Int. Conf. Sampling Theory and Applications (SAMPTA)*, Marseille, France, May 2009.

[26] J. Ilic and T. Strohmer, "PAPR reduction in OFDM using Kashin's representation," in *Proc. IEEE Int. Workshop Signal Process. Adv. Wireless Comm.*, Perugia, Italy, 2009, pp. 444–448.

[27] C. Studer, Y. Wotao, and R. G. Baraniuk, "Signal representations with minimum ℓ_∞-norm," in *Proc. Ann. Allerton Conf. Comm. Control Comput. (Allerton)*, 2012, pp. 1270–1277.

[28] C. Studer, T. Goldstein, W. Yin, and R. G. Baraniuk, "Democratic representations," *IEEE Trans. Inf. Theory*, submitted. [Online]. Available: http://arxiv.org/abs/1401.3420/

[29] J.-J. Fuchs, "Spread representations," in *Proc. IEEE Asilomar Conf. Signals, Systems, Computers*, 2011.

[30] H. Jegou, T. Furon, and J.-J. Fuchs, "Anti-sparse coding for approximate nearest neighbor search," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing (ICASSP)*, 2012, pp. 2029–2032.

[31] C. Studer and E. G. Larsson, "PAR-aware large-scale multi-user MIMO-OFDM downlink," *IEEE J. Sel. Areas Comm.*, vol. 31, no. 2, pp. 303–313, Feb. 2013.

[32] J. Tan, D. Baron, and L. Dai, "Wiener filters in Gaussian mixture signal estimation with ℓ_∞-norm error," *IEEE Trans. Inf. Theory*, vol. 60, no. 10, pp. 6626–6635, Oct. 2014.

[33] M. Pereyra, "Proximal Markov chain Monte Carlo algorithms," *Stat. Comput.*, May 2015.

[34] J.-J. Moreau, "Fonctions convexes duales et points proximaux dans un espace hilbertien," *CR Acad. Sci. Paris Sér. A Math*, vol. 255, pp. 2897–2899, 1962.

[35] N. Chopin, "Fast simulation of truncated gaussian distributions," vol. 21, pp. 275–288, 2011.

[36] J. Geweke, "Getting it right: Joint distribution tests of posterior simulators," *J. Amer. Stat. Assoc.*, vol. 99, pp. 799–804, Sept. 2004.

[37] D. Knowles and Z. Ghahramani, “Nonparametric bayesian sparse factor models with application to gene expression modeling,” *Ann. Appl. Stat.*, vol. 5, no. 2B, pp. 1534–1552, 06 2011.