10-2012

Image reconstruction from compressive samples via a max-product EM algorithm

Zhao Song
Iowa State University

Aleksandar Dogandžić
Iowa State University, ald@iastate.edu

Image reconstruction from compressive samples via a max-product EM algorithm

Abstract
We propose a Bayesian expectation-maximization (EM) algorithm for reconstructing structured approximately sparse signals via belief propagation. The measurements follow an underdetermined linear model where the regression-coefficient vector is the sum of an unknown approximately sparse signal and a zero-mean white Gaussian noise with an unknown variance. The signal is composed of large- and small-magnitude components identified by binary state variables whose probabilistic dependence structure is described by a hidden Markov tree (HMT). Gaussian priors are assigned to the signal coefficients given their state variables and the Jeffreys' noninformative prior is assigned to the noise variance. Our signal reconstruction scheme is based on an EM iteration that aims at maximizing the posterior distribution of the signal and its state variables given the noise variance. We employ a max-product algorithm to implement the maximization (M) step of our EM iteration. The noise variance is a regularization parameter that controls signal sparsity. We select the noise variance so that the corresponding estimated signal and state variables (obtained upon convergence of the EM iteration) have the largest marginal posterior distribution. Our numerical examples show that the proposed algorithm achieves better reconstruction performance compared with the state-of-the-art methods

Keywords
Belief propagation, expectation maximization (EM) algorithm, hidden Markov tree (HMT), image reconstruction, max-product algorithm, structured sparsity, sparse signal reconstruction

Disciplines
Applied Statistics | Electrical and Computer Engineering | Signal Processing

Comments
This article is from *Proceedings of SPIE* 8499 (2012): 849908, doi:10.1117/12.930862. Posted with permission.

Image reconstruction from compressive samples via a max-product EM algorithm

Zhao Song and Aleksandar Dogandžić

ECpE Department, Iowa State University, 3119 Coover Hall, Ames, IA, 50011

ABSTRACT

We propose a Bayesian expectation-maximization (EM) algorithm for reconstructing structured approximately sparse signals via belief propagation. The measurements follow an underdetermined linear model where the regression-coefficient vector is the sum of an unknown approximately sparse signal and a zero-mean white Gaussian noise with an unknown variance. The signal is composed of large- and small-magnitude components identified by binary state variables whose probabilistic dependence structure is described by a hidden Markov tree (HMT). Gaussian priors are assigned to the signal coefficients given their state variables and the Jeffreys' noninformative prior is assigned to the noise variance. Our signal reconstruction scheme is based on an EM iteration that aims at maximizing the posterior distribution of the signal and its state variables given the noise variance. We employ a max-product algorithm to implement the maximization (M) step of our EM iteration. The noise variance is a regularization parameter that controls signal sparsity. We select the noise variance so that the corresponding estimated signal and state variables (obtained upon convergence of the EM iteration) have the largest marginal posterior distribution. Our numerical examples show that the proposed algorithm achieves better reconstruction performance compared with the state-of-the-art methods.

Keywords: Belief propagation, expectation maximization (EM) algorithm, hidden Markov tree (HMT), image reconstruction, max-product algorithm, structured sparsity, sparse signal reconstruction.

1. INTRODUCTION

The advent of compressive sampling (compressed sensing) in the past few years has sparked research activity in sparse signal reconstruction, whose main goal is to estimate the *sparsest* $p \times 1$ signal coefficient vector $\boldsymbol{s}$ from the $N \times 1$ measurement vector $\boldsymbol{y}$ satisfying the following underdetermined system of linear equations: $\boldsymbol{y} = H\,\boldsymbol{s}$, where H is an $N \times p$ *sensing matrix* and $N \leq p$.

A tree dependency structure is exhibited by the wavelet coefficients of many natural images.[1–5] A probabilistic Markov tree structure has been employed to model the statistical dependency between the state variables of wavelet coefficients.[3] An approximate *belief propagation* algorithm has been first applied to compressive sampling in the work by Baron *et al.*,[6] where it has been employed for Bayesian signal reconstruction for sparse Rademacher sensing matrices. Donoho *et al.*[7] simplified the sum-product algorithm by approximating messages with Gaussian distribution specified by two scalar parameters, leading to their *approximate message passing (AMP)* algorithm. Following the AMP framework, Schniter[8] proposed a *turbo-AMP* structured sparse signal recovery method based on loopy belief propagation and turbo equalization and applied it to reconstruct one-dimensional signals; Som and Schniter[5] apply the turbo-AMP approach to reconstruct compressible images. However, the above references do not employ the exact form of the messages and also have the following limitations: Baron *et al.*[6] rely on sparsity of the sensing matrix, the methods by Baron *et al.*[6] and Donoho *et al.*[7] apply to unstructured signals only, and the turbo-AMP approach[5,8] needs columns of the sensing matrix to be normalized, see [5, eq. (22)] and [8, Sec. IV.A].

In this paper, we combine the hierarchical measurement model by Figueiredo and Nowak[9] with a Markov tree prior on the binary state variables that identify the large- and small-magnitude signal coefficients and develop a Bayesian maximum *a posteriori* (MAP) expectation-maximization (EM) signal reconstruction scheme that aims at maximizing the posterior distribution of the signal and its state variables given the noise variance, where

E-mail: {zhaosong,ald}@iastate.edu.

Applications of Digital Image Processing XXXV, edited by Andrew G. Tescher, Proc. of SPIE
Vol. 8499, 849908 · © 2012 SPIE · CCC code: 0277-786/12/$18 · doi: 10.1117/12.930862

the maximization (M) step employs a max-product belief propagation algorithm. Unlike the previous work, we *do not* approximate the message form in our belief propagation scheme. Unlike the turbo-AMP scheme,[5,8] our reconstruction scheme *does not* require the columns of the sensing matrix to be normalized. Since there are no loops in the graphical model behind our M-step objective function, the M step of our EM algorithm is exact. We have proposed a similar EM algorithm for a random signal model with purely sparse deterministic signal component and a noninformative prior on this component.[10] The noise variance is a regularization parameter that controls signal sparsity and is selected so that the estimated signal and state variables have the largest marginal posterior distribution.

In Section 2, we introduce our measurement and prior models. Section 3 describes the proposed EM algorithm, where the M step implementation via the max-product algorithm is presented in Section 3.1. The selection of the regularization noise variance parameter is discussed in Section 4. Numerical simulations in Section 5 compare reconstruction performances of the proposed and existing methods.

We introduce the notation: I_n and $\mathbf{0}_{n\times 1}$ denote the identity matrix of size n and the $n \times 1$ vector of zeros, respectively; "^{T}" and $\|\cdot\|_p$ are the transpose and ℓ_p norm, respectively; $\mathcal{N}(\boldsymbol{x};\boldsymbol{\mu},\Sigma)$ denotes the probability distribution function (pdf) of a multivariate Gaussian random vector $\boldsymbol{x}$ with mean $\boldsymbol{\mu}$ and covariance matrix Σ; Inv-$\chi^2(\sigma^2;\nu,\sigma_0^2)$ denotes the pdf of a scaled inverse chi-square distribution with ν degrees of freedom and a scale parameter σ_0^2, see [11, p. 50 and App. A]; $|\mathcal{T}|$ is the cardinality of the set $\mathcal{T}$; $\upsilon(\cdot)$ is an invertible operator that transforms the matrix element indices into vector element indices*. Finally, ρ_H denotes the largest singular value of a matrix H, also known as the spectral norm of H, and "$\odot$" denotes the Hadamard (elementwise) product.

2. MEASUREMENT AND PRIOR MODELS

We model an $N \times 1$ real-valued measurement vector $\boldsymbol{y}$ using the standard additive Gaussian noise measurement model:[2,5]

$$p_{\boldsymbol{y}\,|\,\boldsymbol{\theta}}(\boldsymbol{y}\,|\,\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{y}\,;\, H\,\boldsymbol{s}, \sigma^2\, I_p) \tag{1}$$

where H is an $N \times p$ real-valued sensing matrix with $\mathrm{rank}(H) = N$ satisfying (without loss of generality)

$$\rho_H = 1 \tag{2}$$

$\boldsymbol{s} = [s_1, s_2, \ldots, s_p]^T$ is an unknown $p \times 1$ real-valued signal coefficient vector, and σ^2 is the unknown noise variance. The set of the unknown parameters is

$$\boldsymbol{\theta} = (\boldsymbol{s}, \sigma^2) \tag{3}$$

with parameter space

$$\Theta = \mathbb{R}^p \times [0, +\infty). \tag{4}$$

We adopt the Jeffreys' noninformative prior for the variance component σ^2:

$$p_{\sigma^2}(\sigma^2) \propto (\sigma^2)^{-1}. \tag{5}$$

Define the vector of binary state variables $\boldsymbol{q} = [q_1, q_2, \ldots, q_p]^T \in \{0,1\}^p$ that determine if the magnitudes of the signal components s_i, $i = 1, 2, \ldots, p$ are small ($q_i = 0$) or large ($q_i = 1$). Assume that s_i are conditionally independent given q_i and assign the following prior pdf to the signal coefficients:

$$p_{\boldsymbol{s}\,|\,\boldsymbol{q},\,\sigma^2}(\boldsymbol{s}\,|\,\boldsymbol{q},\,\sigma^2) = \prod_{i=1}^{p} [\mathcal{N}(s_i\,;\,0, \gamma^2\,\sigma^2)]^{q_i}\, [\mathcal{N}(s_i\,;\,0, \epsilon^2\,\sigma^2)]^{1-q_i} \tag{6a}$$

where γ^2 and ϵ^2 are known positive constants and, typically, $\gamma^2 \gg \epsilon^2$. Hence, the large- and small-magnitude signal coefficients s_i corresponding to $q_i = 1$ and $q_i = 0$ are modeled as zero-mean Gaussian random variables

*This operator is based on the MATLAB wavelet decomposition function `wavedec2` with Haar wavelet and has also been used by He and Carin[4] and Som and Schniter.[5]

with variances $\gamma^2\,\sigma^2$ and $\epsilon^2\,\sigma^2$, respectively. Consequently, γ^2 and ϵ^2 are relative variances (to the noise variance σ^2) of the large- and small-magnitude signal coefficients. Equivalently,

$$p_{\boldsymbol{s}\,|\,\boldsymbol{q},\,\sigma^2}(\boldsymbol{s}\,|\,\boldsymbol{q},\,\sigma^2) = \mathcal{N}(\boldsymbol{s}\,;\,\mathbf{0}_{p\times 1}, \sigma^2\,D(\boldsymbol{q})) \tag{6b}$$

where

$$D(\boldsymbol{q}) = \mathrm{diag}\{(\gamma^2)^{q_1}\,(\epsilon^2)^{1-q_1}, (\gamma^2)^{q_2}\,(\epsilon^2)^{1-q_2}, \ldots, (\gamma^2)^{q_p}\,(\epsilon^2)^{1-q_p}\}. \tag{6c}$$

We now introduce the Markov tree prior probability mass function (pmf) on the state variables q_i.[3,5] To make this probability model easier to understand, we introduce two-dimensional signal element indices (i_1, i_2). Recall that the conversion operator $\upsilon(\cdot)$ is invertible; hence, there is a one-to-one correspondence between the corresponding one- and two-dimensional signal element indices. A parent wavelet coefficient with a two-dimensional position index (i_1, i_2) has four children in the finer wavelet decomposition level with two-dimensional indices $(2\,i_1 - 1, 2\,i_2 - 1)$, $(2\,i_1 - 1, 2\,i_2)$, $(2\,i_1, 2\,i_2 - 1)$ and $(2\,i_1, 2\,i_2)$, see Fig. 1(a). The parent-child dependency assumption implies that, if a parent coefficient in a certain wavelet decomposition level has small (large) magnitude, then its children coefficients in the next finer wavelet decomposition level tend to have small (large) magnitude as well. Denote by ρ and c the numbers of rows and columns of the image, and by L the number of wavelet decomposition levels (tree depth).

We set the prior pmf $p_{\boldsymbol{q}}(\boldsymbol{q})$ as follows. In the first wavelet decomposition level ($l = 1$), assign

$$p_{q_i}(1) = \Pr\{q_i = 1\} = \begin{cases} 1, & i \in \mathcal{A} \\ P_{\mathrm{root}}, & i \in \mathcal{T}_{\mathrm{root}} \end{cases} \tag{7a}$$

where

$$\mathcal{A} = \upsilon\big(\{1, 2, \ldots, \rho/2^L\} \times \{1, 2, \ldots, c/2^L\}\big) \tag{7b}$$

$$\mathcal{T}_{\mathrm{root}} = \upsilon\big([\{1, 2, \ldots, \rho/2^{L-1}\} \times \{1, 2, \ldots, c/2^{L-1}\}]\big) \backslash \mathcal{A} \tag{7c}$$

are the sets of indices of the approximation and root node coefficients and $P_{\mathrm{root}} \in (0, 1)$ is a known constant denoting the prior probability that a root node signal coefficient has large magnitude, see Fig. 1(b). In the levels $l = 2, 3, \ldots, L$, assign

$$p_{q_i\,|\,q_{\pi(i)}}(1\,|\,q_{\pi(i)}) = \begin{cases} P_{\mathrm{H}}, & q_{\pi(i)} = 1 \\ P_{\mathrm{L}}, & q_{\pi(i)} = 0 \end{cases} \tag{7d}$$

where $\pi(i)$ denotes the index of the parent of node i. Here, $P_{\mathrm{H}} \in (0, 1)$ and $P_{\mathrm{L}} \in (0, 1)$ are known constants denoting the probabilities that the signal coefficient s_i is large if the corresponding parent signal coefficient is large or small, respectively.

Our wavelet tree structure consists of $|\mathcal{T}_{\mathrm{root}}|$ trees and spans all signal wavelet coefficients except the approximation coefficients; hence, the set of indices of the wavelet coefficients within the trees is

$$\mathcal{T} = \upsilon\big([\{1, 2, \ldots, \rho\} \times \{1, 2, \ldots, c\}]\big) \setminus \mathcal{A} \tag{7e}$$

Define also the set of leaf variable node indices within the tree structure as

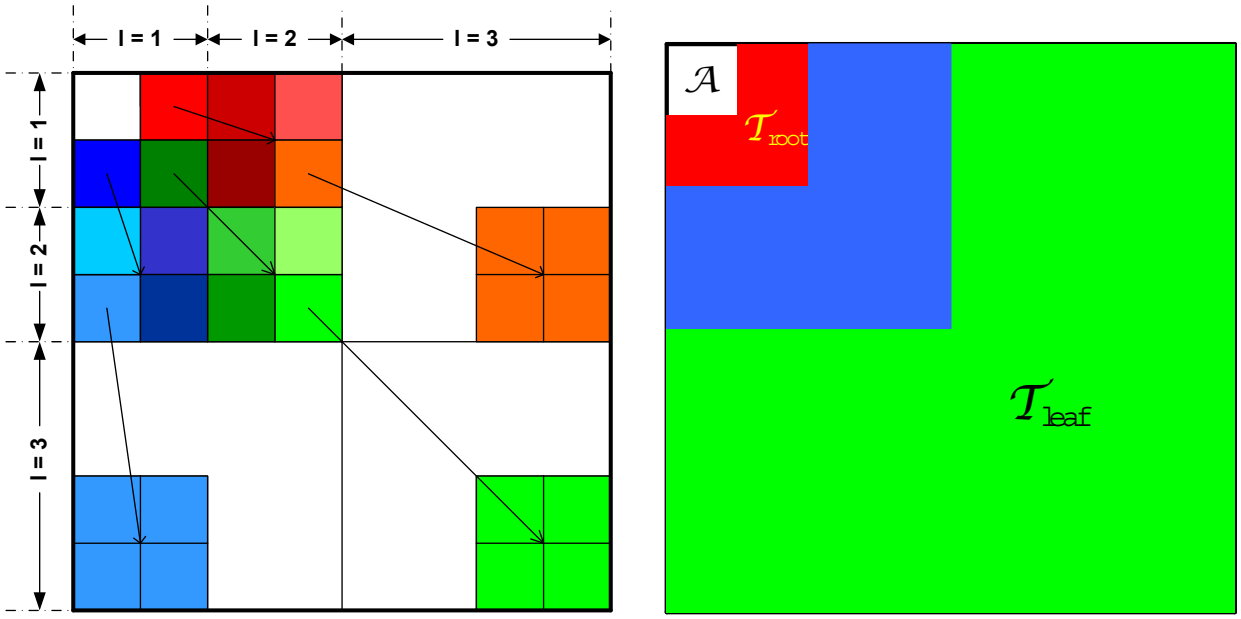
$$\mathcal{T}_{\mathrm{leaf}} = \upsilon\big([\{1, 2, \ldots, \rho\} \times \{1, 2, \ldots, c\}] \setminus [\{1, 2, \ldots, \rho/2\} \times \{1, 2, \ldots, c/2\}]\big) \tag{7f}$$

see Fig. 1(b). More complex models are possible; see e.g., He and Carin[4] and Som and Schniter,[5] which, however, need at least 10 hyperparameters to specify the prior for the same wavelet tree and did not report large-scale examples. Here, we only need five tuning parameters $P_{\mathrm{root}}, P_{\mathrm{H}}, P_{\mathrm{L}}, \gamma^2$, and ϵ^2, each with a clear meaning. A fairly crude choice of these parameters is sufficient for achieving good reconstruction performance, see Section 5.

The logaritm of the prior pmf $p_{\boldsymbol{q}}(\boldsymbol{q})$ is

$$\begin{aligned} \ln p_{\boldsymbol{q}}(\boldsymbol{q}) = \mathrm{const} + \Big[\sum_{i\in\mathcal{A}} \ln \mathbb{1}(q_i = 1)\Big] + \Big[\sum_{i\in\mathcal{T}_{\mathrm{root}}} q_i\,\ln P_{\mathrm{root}} + (1 - q_i)\,\ln(1 - P_{\mathrm{root}})\Big] \\ + \Big[\sum_{i\in\mathcal{T}\setminus\mathcal{T}_{\mathrm{root}}} q_i\,q_{\pi(i)}\,\ln P_{\mathrm{H}} + (1 - q_i)\,q_{\pi(i)}\,\ln(1 - P_{\mathrm{H}}) + q_i\,(1 - q_{\pi(i)})\,\ln P_{\mathrm{L}} + (1 - q_i)\,(1 - q_{\pi(i)})\,\ln(1 - P_{\mathrm{L}})\Big] \end{aligned} \tag{7g}$$

where const denotes the terms that are not functions of $\boldsymbol{q}$.



(a) (b)

Figure 1. (a) Wavelet quadtree structure with $L = 3$ levels and (b) types of wavelet decomposition coefficients.

2.1 Bayesian Inference

Define the vectors of state variables and signal coefficients

$$\boldsymbol{\xi} = \begin{bmatrix} \boldsymbol{\xi}_1^T & \boldsymbol{\xi}_2^T & \cdots & \boldsymbol{\xi}_p^T \end{bmatrix}^T, \quad \boldsymbol{\xi}_i = \begin{bmatrix} q_i, & s_i \end{bmatrix}^T. \tag{8}$$

The joint posterior distribution of $\boldsymbol{\xi}$ and σ^2 is

$$\begin{aligned} p_{\boldsymbol{\xi},\,\sigma^2\,|\,\boldsymbol{y}}(\boldsymbol{\xi},\,\sigma^2\,|\,\boldsymbol{y}) &\propto p_{\boldsymbol{y}\,|\,\boldsymbol{\theta}}(\boldsymbol{y}\,|\,\boldsymbol{\theta})\, p_{\boldsymbol{s}\,|\,\boldsymbol{q},\,\sigma^2}(\boldsymbol{s}\,|\,\boldsymbol{q},\,\sigma^2)\, p_{\boldsymbol{q}}(\boldsymbol{q})\, p_{\sigma^2}(\sigma^2) \\ &\propto (\sigma^2)^{-(p+N+2)/2}\, \exp[-0.5\,\|\boldsymbol{y} - H\,\boldsymbol{s}\|_2^2/\sigma^2 - 0.5\,\boldsymbol{s}^T\,D^{-1}(\boldsymbol{q})\,\boldsymbol{s}/\sigma^2]\,(\epsilon^2/\gamma^2)^{0.5\,\sum_{i=1}^p q_i}\, p_{\boldsymbol{q}}(\boldsymbol{q}) \end{aligned} \tag{9}$$

which implies

$$p_{\sigma^2\,|\,\boldsymbol{\xi},\,\boldsymbol{y}}(\sigma^2\,|\,\boldsymbol{\xi},\,\boldsymbol{y}) = \text{Inv-}\chi^2\left(\sigma^2\,\middle|\,p+N,\,\frac{\|\boldsymbol{y} - H\,\boldsymbol{s}\|_2^2 + \boldsymbol{s}^T\,D^{-1}(\boldsymbol{q})\,\boldsymbol{s}}{p+N}\right) \tag{10a}$$

$$p_{\boldsymbol{\xi}\,|\,\boldsymbol{y}}(\boldsymbol{\xi}\,|\,\boldsymbol{y}) = \frac{p_{\boldsymbol{\xi},\,\sigma^2\,|\,\boldsymbol{y}}(\boldsymbol{\xi},\,\sigma^2\,|\,\boldsymbol{y})}{p_{\sigma^2\,|\,\boldsymbol{\xi},\,\boldsymbol{y}}(\sigma^2\,|\,\boldsymbol{\xi},\,\boldsymbol{y})} \propto p_{\boldsymbol{q}}(\boldsymbol{q})\,(\epsilon^2/\gamma^2)^{0.5\,\sum_{i=1}^p q_i} \Big/ \left[\frac{\|\boldsymbol{y} - H\,\boldsymbol{s}\|_2^2 + \boldsymbol{s}^T\,D^{-1}(\boldsymbol{q})\,\boldsymbol{s}}{p+N}\right]^{(p+N)/2} \tag{10b}$$

and

$$p_{\boldsymbol{\xi}\,|\,\sigma^2,\boldsymbol{y}}(\boldsymbol{\xi}\,|\,\sigma^2,\boldsymbol{y}) \propto \exp[-0.5\,\|\boldsymbol{y} - H\,\boldsymbol{s}\|_2^2/\sigma^2 - 0.5\,\boldsymbol{s}^T\,D^{-1}(\boldsymbol{q})\,\boldsymbol{s}/\sigma^2]\,(\epsilon^2/\gamma^2)^{0.5\,\sum_{i=1}^p q_i}\, p_{\boldsymbol{q}}(\boldsymbol{q}). \tag{10c}$$

We wish to maximize (10b) with respect to $\boldsymbol{\xi}$, but cannot perform this task directly. Consequently, we adopt the following indirect approach: We first develop an EM algorithm for maximizing $p_{\boldsymbol{\xi}\,|\,\sigma^2,\boldsymbol{y}}(\boldsymbol{\xi}\,|\,\sigma^2,\boldsymbol{y})$ in (10c) for a given σ^2 (Section 3) and then propose a grid search scheme for selecting the best regularization parameter σ^2 so that the estimated signal and state variables have the largest marginal posterior distribution (10b) (Section 4).

3. AN EM ALGORITHM FOR MAXIMIZING $P_{\boldsymbol{\xi}\,|\,\sigma^2,\boldsymbol{y}}(\boldsymbol{\xi}\,|\,\sigma^2,\boldsymbol{y})$

Motivated by [9, Sec. V.A], we introduce the following hierarchical two-stage model:

$$p_{\boldsymbol{y}\,|\,\boldsymbol{z},\sigma^2}(\boldsymbol{y}\,|\,\boldsymbol{z},\sigma^2) = \mathcal{N}\big(\boldsymbol{y}\,;\,H\,\boldsymbol{z},\sigma^2\,(I_N - H\,H^T)\big) \tag{11a}$$

$$p_{\boldsymbol{z}\,|\,\boldsymbol{s}}(\boldsymbol{z}\,|\,\boldsymbol{s}) = \mathcal{N}(\boldsymbol{z}\,;\,\boldsymbol{s},\sigma^2\,I_p) \tag{11b}$$

where $\boldsymbol{z}$ is an $p \times 1$ vector of *missing data*. Observe that the assumption (2) guarantees that the covariance matrix $\sigma^2 (I_N - H H^T)$ in (11a) is positive semidefinite.

Our EM algorithm for maximizing $p_{\boldsymbol{\xi} \,|\, \sigma^2, \boldsymbol{y}}(\boldsymbol{\xi} \,|\, \sigma^2, \boldsymbol{y})$ in (10c) consists of iterating between the following expectation (E) and maximization (M) steps:

$$\text{E step:} \qquad \boldsymbol{z}^{(j)} = [z_1^{(j)}, z_2^{(j)}, \ldots, z_p^{(j)}]^T = \boldsymbol{s}^{(j)} + H^T (\boldsymbol{y} - H\, \boldsymbol{s}^{(j)}) \tag{12a}$$

and

$$\text{M step:} \quad \boldsymbol{\xi}^{(j+1)} = \arg\max_{\boldsymbol{\xi}} \Big\{ -0.5\, [\|\boldsymbol{z}^{(j)} - \boldsymbol{s}\|_2^2 + \boldsymbol{s}^T\, D^{-1}(\boldsymbol{q})\, \boldsymbol{s}]/\sigma^2 + \ln[p_{\boldsymbol{q}}(\boldsymbol{q})] + 0.5\, \ln(\epsilon^2/\gamma^2) \sum_{i=1}^{p} q_i \Big\} \tag{12b}$$

where j denotes the iteration index. To simplify the notation, we omit the dependence of the iterates $\boldsymbol{\xi}^{(j)}$ on σ^2 in this section. Denote by $\boldsymbol{\xi}^{(+\infty)}$ and $\boldsymbol{s}^{(+\infty)}$ the estimates of $\boldsymbol{\xi}$ and $\boldsymbol{s}$ obtained upon convergence of the above EM iteration.

Note that the M step in (12b) is equivalent to finding the mode of the following distribution:

$$p_{\boldsymbol{\xi} \,|\, \sigma^2, \boldsymbol{z}}(\boldsymbol{\xi} \,|\, \sigma^2, \boldsymbol{z}^{(j)}) \propto p_{\boldsymbol{\xi}_{\mathcal{A}} \,|\, \sigma^2, \boldsymbol{z}}(\boldsymbol{\xi}_{\mathcal{A}} \,|\, \sigma^2, \boldsymbol{z}^{(j)})\, p_{\boldsymbol{\xi}_{\mathcal{T}} \,|\, \sigma^2, \boldsymbol{z}}(\boldsymbol{\xi}_{\mathcal{T}} \,|\, \sigma^2, \boldsymbol{z}^{(j)}) \tag{13}$$

where $\boldsymbol{\xi}_{\mathcal{A}}$ and $\boldsymbol{\xi}_{\mathcal{T}}$ consist of $\boldsymbol{\xi}_i, i \in \mathcal{A}$ and $\boldsymbol{\xi}_i, i \in \mathcal{T}$, respectively, and

$$p_{\boldsymbol{\xi}_{\mathcal{A}} \,|\, \sigma^2, \boldsymbol{z}}(\boldsymbol{\xi}_{\mathcal{A}} \,|\, \sigma^2, \boldsymbol{z}) \propto \Big\{ \prod_{i \in \mathcal{A}} \mathcal{N}(z_i\,;\, s_i, \sigma^2)\, \mathcal{N}(s_i\,;\, 0, \gamma^2\, \sigma^2)\, \mathbb{1}(q_i = 1) \Big\} \tag{14a}$$

$$p_{\boldsymbol{\xi}_{\mathcal{T}} \,|\, \sigma^2, \boldsymbol{z}}(\boldsymbol{\xi}_{\mathcal{T}} \,|\, \sigma^2, \boldsymbol{z}) \propto \Big\{ \prod_{i \in \mathcal{T}} \mathcal{N}(z_i\,;\, s_i, \sigma^2)\, [\mathcal{N}(s_i\,;\, 0, \gamma^2\, \sigma^2)]^{q_i}\, [\mathcal{N}(s_i\,;\, 0, \epsilon^2\, \sigma^2)]^{1-q_i} \Big\}\, p_{\boldsymbol{q}_{\mathcal{T}}}(\boldsymbol{q}_{\mathcal{T}}). \tag{14b}$$

Here, (14a) follows from (7a) and (14b) corresponds to the hidden Markov tree (HMT) probabilistic model that contains no loops. Fig. 2 depicts an HMT that is a part of the probabilistic model (14b). Maximizing $p_{\boldsymbol{\xi}_{\mathcal{A}} \,|\, \sigma^2, \boldsymbol{z}}(\boldsymbol{\xi}_{\mathcal{A}} \,|\, \sigma^2, \boldsymbol{z}^{(j)})$ in (14a) with respect to $\boldsymbol{\xi}_i$, $i \in \mathcal{A}$ yields the M step for the approximation signal coefficients:

$$\boldsymbol{\xi}_i^{(j+1)} = \Big[1, \quad \gamma^2\, z_i^{(j)}/(1+\gamma^2)\Big]^T, \quad i \in \mathcal{A} \tag{15}$$

where we have used the fact that

$$\arg\max_{s_i} \mathcal{N}(z_i\,;\, s_i, \sigma^2)\, \mathcal{N}(s_i\,;\, 0, \tau^2) = \tau^2\, z_i/(\sigma^2 + \tau^2). \tag{16}$$

In the following section, we employ the max-product belief propagation algorithm[12–14] to each tree in our wavelet tree structure, with the goal to find the mode of $p_{\boldsymbol{\xi}_{\mathcal{T}} \,|\, \sigma^2, \boldsymbol{z}}(\boldsymbol{\xi}_{\mathcal{T}} \,|\, \sigma^2, \boldsymbol{z})$; then, our M step in (12b) for the nodes $i \in \mathcal{T}$ reduces to applying this algorithm to find the mode of $p_{\boldsymbol{\xi}_{\mathcal{T}} \,|\, \sigma^2, \boldsymbol{z}}(\boldsymbol{\xi}_{\mathcal{T}} \,|\, \sigma^2, \boldsymbol{z}^{(j)})$.

3.1 Maximizing $p_{\boldsymbol{\xi}_{\mathcal{T}} \,|\, \sigma^2, \boldsymbol{z}}(\boldsymbol{\xi}_{\mathcal{T}} \,|\, \sigma^2, \boldsymbol{z})$

We represent the HMT probabilistic model for $p_{\boldsymbol{\xi}_{\mathcal{T}} \,|\, \sigma^2, \boldsymbol{z}}(\boldsymbol{\xi}_{\mathcal{T}} \,|\, \sigma^2, \boldsymbol{z})$ via *potential functions* as [see (14b)]

$$p_{\boldsymbol{\xi}_{\mathcal{T}} \,|\, \sigma^2, \boldsymbol{z}}(\boldsymbol{\xi}_{\mathcal{T}} \,|\, \sigma^2, \boldsymbol{z}) \propto \Big[\prod_{i \in \mathcal{T} \backslash \mathcal{T}_{\text{root}}} \psi_i(\boldsymbol{\xi}_i)\, \psi_{i,\pi(i)}(q_i, q_{\pi(i)}) \Big] \Big[\prod_{i \in \mathcal{T}_{\text{root}}} \psi_i(\boldsymbol{\xi}_i) \Big] \tag{17}$$

where

$$\psi_i(\boldsymbol{\xi}_i) = \begin{cases} \mathcal{N}(z_i\,;\, s_i, \sigma^2)\, [\mathcal{N}(s_i\,;\, 0, \gamma^2\, \sigma^2)]^{q_i}\, [\mathcal{N}(s_i\,;\, 0, \epsilon^2\, \sigma^2)]^{1-q_i}, & i \in \mathcal{T} \backslash \mathcal{T}_{\text{root}} \\ \mathcal{N}(z_i\,;\, s_i, \sigma^2)\, [P_{\text{root}}\, \mathcal{N}(s_i\,;\, 0, \gamma^2\, \sigma^2)]^{q_i}\, [(1 - P_{\text{root}})\, \mathcal{N}(s_i\,;\, 0, \epsilon^2\, \sigma^2)]^{1-q_i}, & i \in \mathcal{T}_{\text{root}} \end{cases} \tag{18a}$$

and, for $i \in \mathcal{T} \backslash \mathcal{T}_{\text{root}}$,

$$\psi_{i,\pi(i)}(q_i, q_{\pi(i)}) = [{P_{\text{H}}}^{q_i}\, (1 - P_{\text{H}})^{1-q_i}]^{q_{\pi(i)}}\, [{P_{\text{L}}}^{q_i}\, (1 - P_{\text{L}})^{1-q_i}]^{1-q_{\pi(i)}}. \tag{18b}$$

Our algorithm for maximizing (17) consists of computing and passing upward and downward messages and calculating and maximizing beliefs.

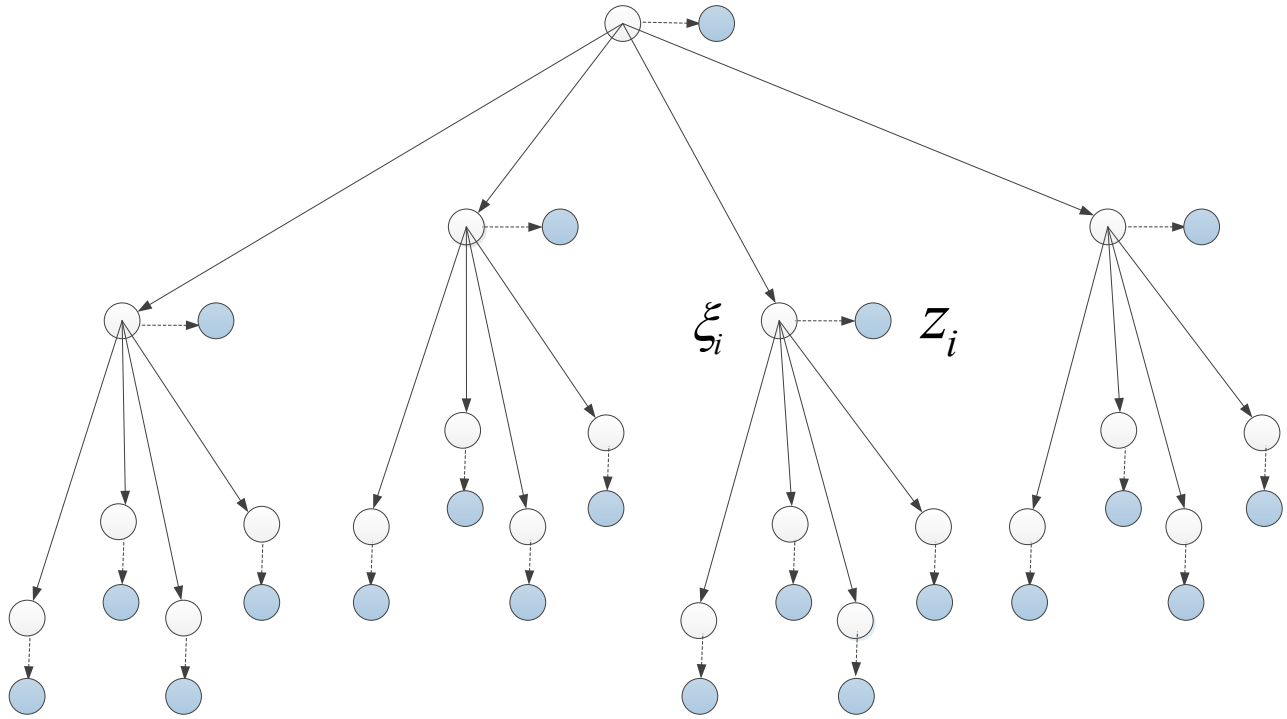


Figure 2. A hidden Markov tree, part of the probabilistic model (14b).

3.1.1 Computing and Passing Upward Messages

We propagate the upward messages from the lowest decomposition level (i.e., the leaves) towards the root of the tree. Fig. 3(a) depicts the computation of the upward message from variable node $\boldsymbol{\xi}_i$ to its parent node $\boldsymbol{\xi}_{\pi(i)}$ wherein we also define a *child* of $\boldsymbol{\xi}_i$ as a variable node $\boldsymbol{\xi}_k$ with index $k \in \mathrm{ch}(i)$, where $\mathrm{ch}(i)$ is the index set of the children of i: for $i = \upsilon(i_1, i_2)$, $\mathrm{ch}(i) = \{\upsilon\big((2\,i_1 - 1, 2\,i_2 - 1), (2\,i_1 - 1, 2\,i_2), (2\,i_1, 2\,i_2 - 1), (2\,i_1, 2\,i_2)\big)\}$. Here, we use a circle and an edge with an arrow to denote a variable node and a message, respectively. The upward messages have the following general form:[13]

$$m_{i\to\pi(i)}(q_{\pi(i)}) = \alpha \max_{\boldsymbol{\xi}_i} \Big\{ \psi_i(\boldsymbol{\xi}_i)\, \psi_{i,\pi(i)}(q_i, q_{\pi(i)}) \prod_{k\in\mathrm{ch}(i)} m_{k\to i}(q_i) \Big\} \tag{19}$$

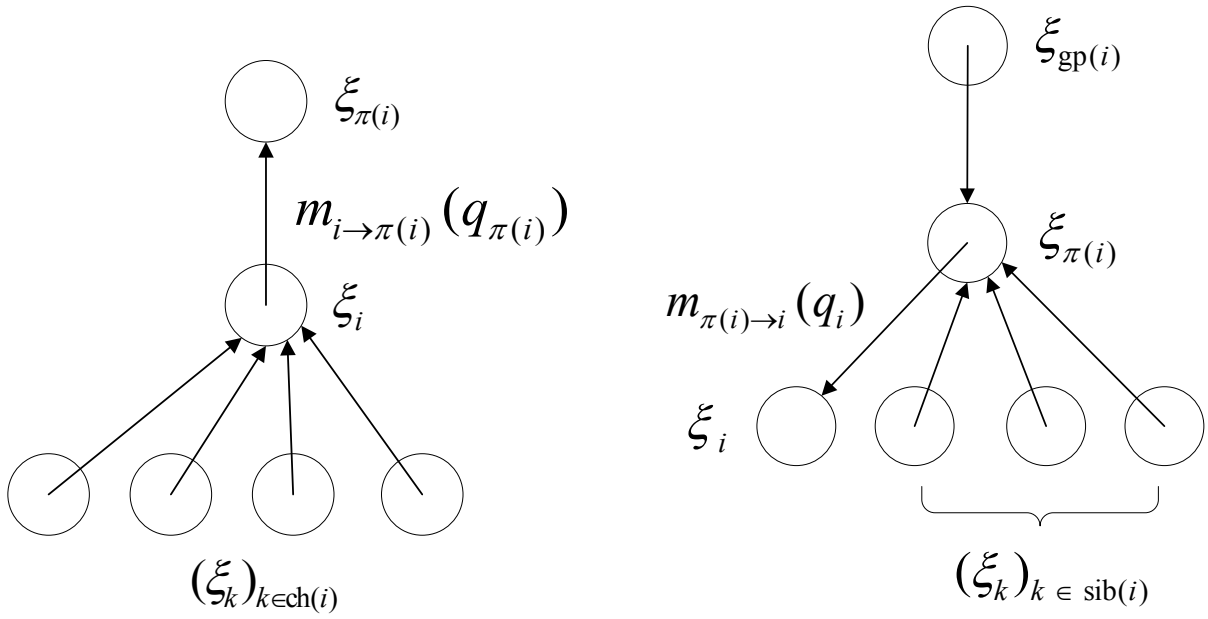
where $\alpha > 0$ denotes a normalizing constant used for computational stability.[13] For nodes that have no children (corresponding to the level L, i.e., $i \in \mathcal{T}_{\mathrm{leaf}}$), we set the multiplicative term $\prod_{k\in\mathrm{ch}(i)} m_{k\to i}(\boldsymbol{\xi}_i)$ in (19) to one.

The only two candidates for $\boldsymbol{\xi}_i$ in the maximization of (19) are $[0, \widehat{s}_i(0)]^T$ and $[1, \widehat{s}_i(1)]^T$, see (16) and (18), where

$$\widehat{s}_i(0) = \frac{\epsilon^2}{1+\epsilon^2}\, z_i, \quad \widehat{s}_i(1) = \frac{\gamma^2}{1+\gamma^2}\, z_i. \tag{20}$$

Substituting these candidates into (19) and normalizing the messages yields

$$m_{i\to\pi(i)}(q_{\pi(i)}) = [\mu_i^{\mathrm{u}}(0)]^{1-q_{\pi(i)}}\, [\mu_i^{\mathrm{u}}(1)]^{q_{\pi(i)}} \tag{21a}$$



(a) (b)

Figure 3. Computing and passing (a) upward and (b) downward messages.

where $[\mu_i^{\mathrm{u}}(0), \mu_i^{\mathrm{u}}(1)]^T = \boldsymbol{\mu}_i^{\mathrm{u}}$,

$$\boldsymbol{\mu}_i^{\mathrm{u}} = \frac{[\max\{\boldsymbol{\nu}_{0,i}^{\mathrm{u}} \odot \boldsymbol{\eta}_i^{\mathrm{u}}\},\, \max\{\boldsymbol{\nu}_{1,i}^{\mathrm{u}} \odot \boldsymbol{\eta}_i^{\mathrm{u}}\}]^T}{\max\{\boldsymbol{\nu}_{0,i}^{\mathrm{u}} \odot \boldsymbol{\eta}_i^{\mathrm{u}}\} + \max\{\boldsymbol{\nu}_{1,i}^{\mathrm{u}} \odot \boldsymbol{\eta}_i^{\mathrm{u}}\}} = \frac{\big[\exp(\ln\max\{\boldsymbol{\nu}_{0,i}^{\mathrm{u}} \odot \boldsymbol{\eta}_i^{\mathrm{u}}\} - \ln\max\{\boldsymbol{\nu}_{1,i}^{\mathrm{u}} \odot \boldsymbol{\eta}_i^{\mathrm{u}}\}),\quad 1\big]^T}{1 + \exp(\ln\max\{\boldsymbol{\nu}_{0,i}^{\mathrm{u}} \odot \boldsymbol{\eta}_i^{\mathrm{u}}\} - \ln\max\{\boldsymbol{\nu}_{1,i}^{\mathrm{u}} \odot \boldsymbol{\eta}_i^{\mathrm{u}}\})} \tag{21b}$$

$$\boldsymbol{\nu}_{0,i}^{\mathrm{u}} = \big[1 - P_{\mathrm{L}},\quad P_{\mathrm{L}}\big]^T \odot \boldsymbol{\phi}(z_i) \tag{21c}$$

$$\boldsymbol{\nu}_{1,i}^{\mathrm{u}} = \big[1 - P_{\mathrm{H}},\quad P_{\mathrm{H}}\big]^T \odot \boldsymbol{\phi}(z_i) \tag{21d}$$

$$\boldsymbol{\eta}_i^{\mathrm{u}} = \begin{cases} \bigodot_{k\in\mathrm{ch}(i)} \boldsymbol{\mu}_k^{\mathrm{u}}, & i \in \mathcal{T}\backslash\mathcal{T}_{\mathrm{leaf}} \\ [1,\quad 1]^T, & i \in \mathcal{T}_{\mathrm{leaf}} \end{cases} \tag{21e}$$

$$\boldsymbol{\phi}(z) = \big[\exp\{-0.5\, z^2/(\sigma^2 + \sigma^2\,\epsilon^2)\}/\epsilon,\quad \exp\{-0.5\, z^2/(\sigma^2 + \sigma^2\,\gamma^2)\}/\gamma\big]^T \tag{21f}$$

and $\epsilon = \sqrt{\epsilon^2} > 0$ and $\gamma = \sqrt{\gamma^2} > 0$. To derive (21a), we have used the following fact [see (16)]:

$$\max_{\boldsymbol{s}_i} \mathcal{N}(z_i\,;\, s_i, \sigma^2)\,\mathcal{N}(s_i\,;\, 0, \tau^2) = \frac{1}{\sqrt{2\,\pi\,\sigma^2}\,\sqrt{2\,\pi\,\tau^2}}\,\exp[-0.5\, z_i^2/(\sigma^2 + \tau^2)]. \tag{22}$$

A numerically stable implementation of (21b) that we employ is illustrated in the second expression in (21b). Similarly, the elementwise products in (21c)–(21e) are implemented as exponentiated sums of logarithms of the product terms.

3.1.2 Computing and Passing Downward Messages

Upon obtaining all the upward messages, we now compute the downward messages and propagate them from the root towards the lowest level (i.e., the leaves). Fig. 3(b) depicts the computation of the downward message from the parent $\boldsymbol{\xi}_{\pi(i)}$ to the variable node $\boldsymbol{\xi}_i$, which involves upward messages to $\boldsymbol{\xi}_{\pi(i)}$ from its other children, i.e. the *siblings* of $\boldsymbol{\xi}_i$, marked as $\boldsymbol{\xi}_k$, $k \in \mathrm{sib}(i)$. This downward message also requires the message sent to $\boldsymbol{\xi}_{\pi(i)}$ from its parent node, which is the *grandparent* of $\boldsymbol{\xi}_i$, denoted by $\boldsymbol{\xi}_{\mathrm{gp}(i)}$. The downward messages have the following general form:[13]

$$m_{\pi(i)\to i}(q_i) = \alpha \max_{\boldsymbol{\xi}_{\pi(i)}} \Big\{ \psi_{\pi(i)}(\boldsymbol{\xi}_{\pi(i)})\, \psi_{i,\pi(i)}(q_i, q_{\pi(i)})\, m_{\mathrm{gp}(i)\to\pi(i)}(\pi(i)) \prod_{k\in\mathrm{sib}(i)} m_{k\to\pi(i)}(q_{\pi(i)}) \Big\} \tag{23}$$

where $\alpha > 0$ denotes a normalizing constant used for computational stability. For the variable nodes i in the second decomposition level that have no grandparents (i.e., $\pi(i) \in \mathcal{T}_{\rm root}$), we set the multiplicative term $m_{{\rm gp}(i)\to\pi(i)}(\pi(i))$ in (23) to one.

The only two candidates for $\boldsymbol{\xi}_{\pi(i)}$ in the maximization of (23) are $[0, \widehat{s}_{\pi(i)}(0)]^T$ and $[1, \widehat{s}_{\pi(i)}(1)]^T$, see (16), (18), and (20). Consequently,

$$m_{\pi(i)\to i}(q_i) = [\mu_i^{\rm d}(0)]^{1-q_i}\,[\mu_i^{\rm d}(1)]^{q_i} \tag{24a}$$

for $\pi(i) \in \mathcal{T}\backslash\mathcal{T}_{\rm leaf}$, where $[\mu_i^{\rm d}(0), \mu_i^{\rm d}(1)]^T = \boldsymbol{\mu}_i^{\rm d}$ and

$$\boldsymbol{\mu}_i^{\rm d} = \frac{[\max\{\boldsymbol{\nu}_{0,i}^{\rm d} \odot \boldsymbol{\eta}_i^{\rm d}\},\, \max\{\boldsymbol{\nu}_{1,i}^{\rm d} \odot \boldsymbol{\eta}_i^{\rm d}\}]^T}{\max\{\boldsymbol{\nu}_{0,i}^{\rm d} \odot \boldsymbol{\eta}_i^{\rm d}\} + \max\{\boldsymbol{\nu}_{1,i}^{\rm d} \odot \boldsymbol{\eta}_i^{\rm d}\}} = \frac{\big[\exp(\ln\max\{\boldsymbol{\nu}_{0,i}^{\rm d} \odot \boldsymbol{\eta}_i^{\rm d}\} - \ln\max\{\boldsymbol{\nu}_{1,i}^{\rm d} \odot \boldsymbol{\eta}_i^{\rm d}\}),\quad 1\big]^T}{1 + \exp(\ln\max\{\boldsymbol{\nu}_{0,i}^{\rm d} \odot \boldsymbol{\eta}_i^{\rm d}\} - \ln\max\{\boldsymbol{\nu}_{1,i}^{\rm d} \odot \boldsymbol{\eta}_i^{\rm d}\})} \tag{24b}$$

$$\boldsymbol{\nu}_{0,i}^{\rm d} = \begin{bmatrix}1-P_{\rm L}, & 1-P_{\rm H}\end{bmatrix}^T \odot \boldsymbol{\phi}(z_{\pi(i)}) \odot \Big[\bigodot_{k\in{\rm sib}(i)} \boldsymbol{\mu}_k^{\rm u}\Big] \tag{24c}$$

$$\boldsymbol{\nu}_{1,i}^{\rm d} = \begin{bmatrix}P_{\rm L}, & P_{\rm H}\end{bmatrix}^T \odot \boldsymbol{\phi}(z_{\pi(i)}) \odot \Big[\bigodot_{k\in{\rm sib}(i)} \boldsymbol{\mu}_k^{\rm u}\Big] \tag{24d}$$

$$\boldsymbol{\eta}_i^{\rm d} = \begin{cases} \begin{bmatrix}1-P_{\rm root}, & P_{\rm root}\end{bmatrix}^T, & \pi(i) \in \mathcal{T}_{\rm root} \\ \boldsymbol{\mu}_{\pi(i)}^{\rm d}, & \pi(i) \in (\mathcal{T}\backslash\mathcal{T}_{\rm root})\backslash\mathcal{T}_{\rm leaf} \end{cases}. \tag{24e}$$

We have used (22) to derive (24a).

The above upward and downward messages have discrete representations, which is practically important and is a consequence of the fact that we use a Gaussian prior on the signal coefficients, see (6). Indeed, in contrast with the existing message passing algorithms for compressive sampling,[5–8] our max-product scheme employs *exact* messages.

3.1.3 Maximizing Beliefs

Upon computing and passing all the upward and downward messages, we maximize the beliefs, which have the following general form:[13†]

$$b(\boldsymbol{\xi}_i) = \alpha\,\psi_i(\boldsymbol{\xi}_i)\,m_{\pi(i)\to i}(q_i) \prod_{k\in{\rm ch}(i)} m_{k\to i}(q_i) \tag{25}$$

for each $i \in \mathcal{T}$, where $\alpha > 0$ is a normalizing constant. We then use these beliefs to obtain the mode

$$\widehat{\boldsymbol{\xi}}_{\mathcal{T}} = \arg\max_{\boldsymbol{\xi}_{\mathcal{T}}} p_{\boldsymbol{\xi}_{\mathcal{T}}\,|\,\boldsymbol{z}}(\boldsymbol{\xi}_{\mathcal{T}}\,|\,\boldsymbol{z}) \tag{26}$$

where the elements of $\widehat{\boldsymbol{\xi}}_{\mathcal{T}}$ are [see (20)]

$$\widehat{\boldsymbol{\xi}}_i = \begin{bmatrix}\widehat{q}_i, & \widehat{s}_i(\widehat{q}_i)\end{bmatrix}^T = \arg\max_{\boldsymbol{\xi}_i} b(\boldsymbol{\xi}_i) = \begin{cases} \begin{bmatrix}1, & \widehat{s}_i(1)\end{bmatrix}^T, & \beta_i(1) \geq \beta_i(0) \\ \begin{bmatrix}0, & \widehat{s}_i(0)\end{bmatrix}^T, & \text{otherwise} \end{cases}, \quad i \in \mathcal{T} \tag{27a}$$

and

$$\boldsymbol{\beta}_i = \begin{bmatrix}\beta_i(0), & \beta_i(1)\end{bmatrix}^T = \begin{cases} \begin{bmatrix}1-P_{\rm root}, & P_{\rm root}\end{bmatrix}^T \odot \boldsymbol{\phi}(z_i) \odot \boldsymbol{\eta}_i^{\rm u}, & i \in \mathcal{T}_{\rm root} \\ \boldsymbol{\phi}(z_i) \odot \boldsymbol{\mu}_i^{\rm d} \odot \boldsymbol{\eta}_i^{\rm u}, & i \in \mathcal{T}\setminus\mathcal{T}_{\rm root} \end{cases}. \tag{27b}$$

We have used (16) and (22) to solve the maximization in (27a) and derive (27a)–(27b).

†In (25), we set $m_{\pi(i)\to i}(q_i) = 1$ if $i \in \mathcal{T}_{\rm root}$ and $\prod_{k\in{\rm ch}(i)} m_{k\to i}(q_i) = 1$ if $i \in \mathcal{T}_{\rm leaf}$.

4. SELECTING σ^2

We apply our EM algorithm in Section 3 using a range of values of the regularization parameter σ^2. We traverse the grid of K values of σ^2 sequentially and use the signal estimate from the previous grid point to initialize the signal estimation at the current grid point; in particular, we move from a larger σ^2 (say $\sigma^2_{\rm old}$) to the next smaller $\sigma^2_{\rm new}(< \sigma^2_{\rm old})$ and use $\boldsymbol{s}^{(+\infty)}(\sigma^2_{\rm old})$ (obtained upon convergence of the EM iteration in Section 3 for $\sigma^2 = \sigma^2_{\rm old}$) to initialize the EM iteration at $\sigma^2_{\rm new}$. The largest σ^2 and the initial signal estimate at this grid point are selected as

$$\sigma^2_{\rm MAX} = \|\boldsymbol{y}\|_2^2/(p+N+2), \quad \boldsymbol{s}^{(0)}(\sigma^2_{\rm MAX}) = \boldsymbol{0}_{p\times 1} \tag{28a}$$

and the consecutive grid points $\sigma^2_{\rm new}$ and $\sigma^2_{\rm old}$ satisfy

$$\sigma^2_{\rm new} = \sigma^2_{\rm old}/d \tag{28b}$$

where $d > 1$ is a selected constant.

Finally, we select the σ^2 from the above grid of candidates that yields the largest marginal posterior distribution (10b). Denote this σ^2 by $\sigma^2_\star$; then, we select the final estimates of $\boldsymbol{\xi}$ and $\boldsymbol{s}$ as $\boldsymbol{\xi}^{(+\infty)}(\sigma^2_\star)$ and $\boldsymbol{s}^{(+\infty)}(\sigma^2_\star)$, respectively.

5. NUMERICAL EXAMPLES

We compare the reconstruction performances of the following methods:

- our proposed *max-product EM* algorithm in Section 3 with the variance parameter σ^2 selected as described in Section 4 (labeled MP-EM) using $K = 12$ grid points with $d = 2$ and the tuning parameters chosen as

$$\gamma^2 = 1000, \quad \epsilon^2 = 0.1, \quad P_{\rm root} = P_{\rm H} = 0.2, \quad P_{\rm L} = 10^{-5}; \tag{29}$$

- the turbo-AMP approach[5] with a MATLAB implementation at `http://www.ece.osu.edu/~schniter/turboAMPimaging` and the tuning parameters chosen as the default values in this implementation;

- the fixed-point continuation active set algorithm[15] (labeled $\rm FPC_{AS}$) that aims at minimizing the Lagrangian cost function

$$0.5\,\|\boldsymbol{y} - H\,\boldsymbol{s}\|_2^2 + \tau\,\|\boldsymbol{s}\|_1 \tag{30a}$$

  with the regularization parameter τ computed as

$$\tau = 10^a\,\|H^T\,\boldsymbol{y}\|_\infty \tag{30b}$$

  where a is a tuning parameter chosen to achieve good reconstruction performance;

- the Barzilai-Borwein version of the gradient-projection for sparse reconstruction method with debiasing in [16, Sec. III.B] (labeled GPSR) with the convergence threshold `tolP` $= 10^{-5}$ and tuning parameter a in (30b) chosen to achieve good reconstruction performance;

- the double overrelaxation (DORE) thresholding method in [17, Sec. III] initialized by the zero sparse signal estimate:

$$\boldsymbol{s}^{(0)} = \boldsymbol{0}_{p\times 1}; \tag{31}$$

- the normalized iterative hard thresholding (NIHT) scheme[18] initialized by the zero $\boldsymbol{s}^{(0)}$ in (31);

- the model-based iterative hard thresholding (MB-IHT) algorithmn[1] using a greedy tree approximation,[19] initialized by the zero $\boldsymbol{s}^{(0)}$ in (31).

For the MP-EM, DORE, NIHT, and MB-IHT iterations, we use the following convergence criterion:

$$\|\boldsymbol{s}^{(p+1)} - \boldsymbol{s}^{(p)}\|_2^2 \,/\, p < 10^{-14}. \tag{32}$$

For GPSR and FPC_{AS}, we tuned the regularization parameter τ manually by varying a with the set $\{-1,\ -2,\ -3,\ -4,\ -5,\ -6,\ -7,\ -8,\ -9\}$: the best reconstruction performances are achieved for $a = -3$.

The sensing matrix H has the following structure:

$$H = \frac{1}{\rho_\Phi}\,\Phi\,\Psi \tag{33}$$

where Φ is the $N \times p$ sampling matrix and Ψ is the $p \times p$ orthogonal inverse Haar wavelet transform matrix (satisfying $\Psi\,\Psi^T = I_p$) with

$$L = 4 \tag{34}$$

wavelet decomposition levels. Note that H in (33) satisfies (2). In our examples presented here, the sampling matrices Φ are random Gaussian (see Section 5.1) or structurally random[20] (see Section 5.2).

5.1 Image Reconstruction Using Gaussian I.I.D. Sensing Matrices

We reconstruct the 128×128 'Cameraman' image from compressive samples generated using random sampling matrices Φ with independent, identically distributed (i.i.d.) standard normal elements. Our performance metric is the normalized mean square error (MSE) of a signal coefficient vector estimate $\widetilde{\boldsymbol{s}}$:

$$\text{MSE}\{\widetilde{\boldsymbol{s}}\} = \frac{\text{E}_{\,\Phi}[\|\widetilde{\boldsymbol{s}} - \boldsymbol{s}\|_2^2]}{\|\boldsymbol{s}\|_2^2} \tag{35}$$

computed using 10 Monte Carlo trials.

We set the sparsity level r for NIHT and DORE as $2000\,N/p$ and $2500\,N/p$ for MB-IHT, tuned for good MSE performance.

Recall that the turbo-AMP approach needs columns of the sensing matrix to be normalized, see [5, eq. (22)]. When applying the turbo-AMP method, we scale the sensing matrix as follows:

$$H_{\text{scale}} = (1/\sqrt{N})\,\Phi\,\Psi \tag{36}$$

so that it has approximately normalized columns; this approximation becomes more accurate as we increase N. With measurements $\boldsymbol{y}$ and scaled sensing matrix H_{scale}, turbo-AMP returns the scaled signal estimate $\boldsymbol{s}_{\text{scale}}$, and we compute the final turbo-AMP signal estimate as $(\rho_\Phi/\sqrt{N})\,\boldsymbol{s}_{\text{scale}}$, whose performance is evaluated using (35).

Fig. 4 shows the MSE performances of different algorithms as functions of the normalized number of measurements (subsampling factor) N/p. MP-EM achieves the best MSE when $N/p \leq 0.35$. The MSEs of GPSR and FPC_{AS} are close to each other and smaller than those of DORE, NIHT, and MB-IHT for all N/p and the MSE of MP-EM is 1.8 to 2.5 times smaller than that of GPSR and FPC_{AS}, see Fig. 4.

The relatively poor performance of MB-IHT is likely due to the inflexibility of the greedy tree approximation and *deterministic tree structure* that it employs[‡]; a relatively poor performance of MB-CoSAMP (which employs the same deterministic tree structure) has also been reported in [5, Sec. IV.B].

For $N/p \leq 0.35$, turbo-AMP performs similarly to DORE, NIHT, and MB-IHT, but it outperforms all other methods for $N/p > 0.35$. Such a good performance of turbo-AMP at large N is likely facilitated by the fact that the norms of the columns of (36) become closer to unity as N grows.

[‡]This deterministic tree structure model requires that, for binary states equal to one (identifying large signal coefficients), the binary states of all their ancestors must be one as well.

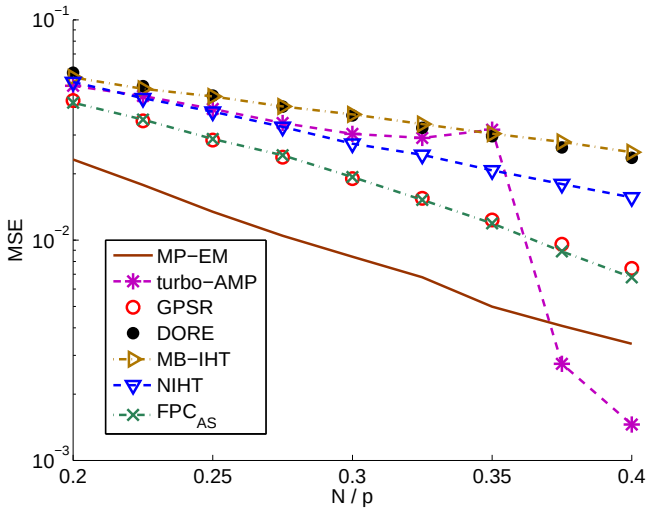


Figure 4. MSEs as functions of the subsampling factor N/p.

5.2 Image Reconstruction Using a Structurally Random Sampling Matrix

We now reconstruct the standard 256×256 'Lena' and 'Cameraman' images from structurally random compressive samples, which implies that the sensing matrix H has orthonormal rows (i.e., $HH^T = I_N$)[20] and, consequently, $\rho_\Phi = \rho_H = 1$. Our performance metric is the peak signal-to-noise ratio (PSNR) of an estimated signal $\widetilde{\boldsymbol{s}}$:

$$\text{PSNR (dB)} = 10 \log_{10} \left\{ \frac{[(\Psi \boldsymbol{s})_{\text{MAX}} - (\Psi \boldsymbol{s})_{\text{MIN}}]^2}{\|\widetilde{\boldsymbol{s}} - \boldsymbol{s}\|_2^2 / p} \right\}. \tag{37}$$

We set the sparsity level r for NIHT and DORE as $12500\, N/p$ and $15000\, N/p$ for MB-IHT, tuned for good PSNR performance.

When applying the turbo-AMP method, we scale the sensing matrix as follows:

$$H_{\text{scale}} = (\sqrt{p/N})\, \Phi\, \Psi \tag{38}$$

With measurements $\boldsymbol{y}$ and scaled sensing matrix H_{scale}, turbo-AMP returns the scaled signal estimate $\boldsymbol{s}_{\text{scale}}$, and we compute the final turbo-AMP signal estimate as $(\sqrt{p/N})\, \boldsymbol{s}_{\text{scale}}$, whose performance is evaluated using (37). Our empirical experience shows that the sensing matrix scaling in (38) improves the reconstruction performance of the turbo-AMP algorithm in this example.

Fig. 5(a) shows the PSNRs achieved by various methods when reconstructing the 256×256 'Lena' image. For $N/p < 0.4$, the proposed MP-EM method outperforms all other methods, where the performance improvement compared with the closest competitor varies between 1.3 dB and 2 dB. For $N/p = 0.4$, turbo-AMP outperforms all other methods.

Fig. 5(b) shows the PSNRs achieved by various methods when reconstructing the 256×256 'Cameraman' image. For $N/p < 0.4$, the proposed MP-EM method outperforms all other methods by at least 2.2 dB. For $N/p = 0.4$, turbo-AMP outperforms all other methods; however, it performs quite poorly for $N/p < 0.35$.

Fig. 6 shows the reconstructed 256×256 'Cameraman' images by different methods for $N/p = 0.375$. The MP-EM algorithm achieves better reconstructed image quality compared with the other methods.

6. CONCLUDING REMARKS

We presented a Bayesian EM algorithm for image reconstruction from compressive samples using a Markov tree prior for the image wavelet coefficients. We employed the max-product belief propagation algorithm to implement the M step of the proposed EM iteration. Compared with the existing message passing algorithms

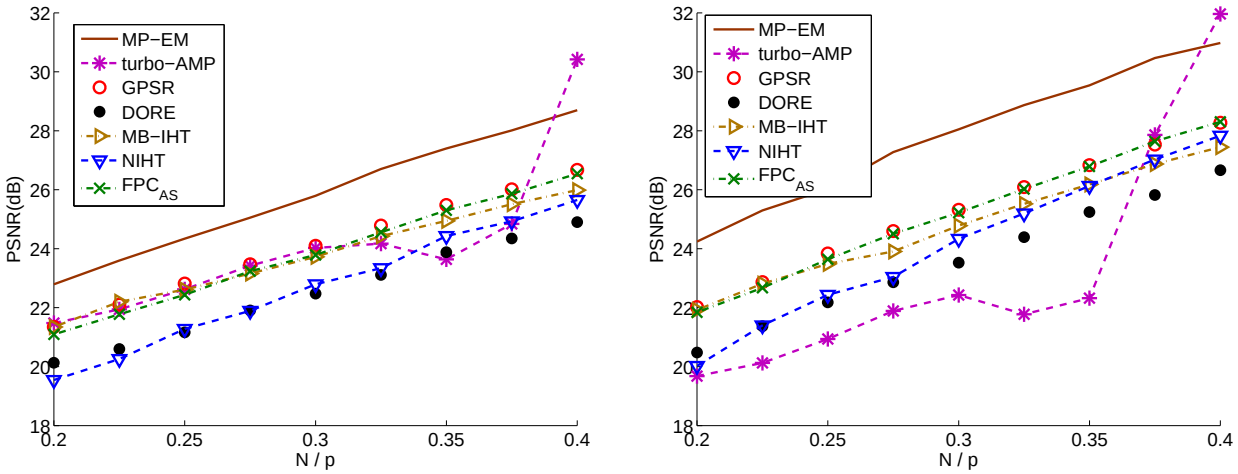


Figure 5. PSNRs as functions of the subsampling factor N/p for the 256×256 (a) 'Lena' and (b) 'Cameraman' images.

in the compressive sampling area, our method does not approximate the message form. The simulation results show that our algorithm often outperforms existing algorithms for standard test images and sampling operators.

The performance of our MP-EM scheme can be improved by employing multiple initializations. In particular, it is of interest to examine initialization of each grid point by the turbo-AMP estimate (in addition to the initialization in Section 4), which would likely improve the performance of MP-EM at large N/p. Our future work will also include the convergence analysis of the MP-EM algorithm and incorporating other measurement models.

REFERENCES

[1] Baraniuk, R., Cevher, V., Duarte, M. F., and Hegde, C., "Model-based compressive sensing," *IEEE Trans. Inf. Theory* **56**, 1982–2001 (Apr. 2010).

[2] Cevher, V., Indyk, P., Carin, L., and Baraniuk, R., "Sparse signal recovery and acquisition with graphical models," *IEEE Signal Process. Mag.* **27**, 92–103 (Nov. 2010).

[3] Crouse, M. S., Nowak, R. D., and Baraniuk, R. G., "Wavelet-based statistical signal processing using hidden Markov models," *IEEE Trans. Signal Process.* **46**, 886–902 (Apr. 1998).

[4] He, L. and Carin, L., "Exploiting structure in wavelet-based Bayesian compressive sensing," *IEEE Trans. Signal Process.* **57**, 3488–3497 (Sept. 2009).

[5] Som, S. and Schniter, P., "Compressive imaging using approximate message passing and a Markov-tree prior," *IEEE Trans. Signal Process.* **60**(99), 3439–3448 (2012).

[6] Baron, D., Sarvotham, S., and Baraniuk, R. G., "Bayesian compressive sensing via belief propagation," *IEEE Trans. Signal Process.* **58**, 269–280 (Jan. 2010).

[7] Donoho, D., Maleki, A., and Montanari, A., "Message-passing algorithms for compressed sensing," *Proc. Nat. Acad. Sci* **106**(45), 18914–18919 (2009).

[8] Schniter, P., "Turbo reconstruction of structured sparse signals," in [*Proc. Conf. Inform. Sci. Syst.*], 1–6 (2010).

[9] Figueiredo, M. A. T. and Nowak, R. D., "An EM algorithm for wavelet-based image restoration," *IEEE Trans. Image Process.* **12**, 906–916 (2003).

[10] Song, Z. and Dogandžić, A., "A Bayesian max-product EM algorithm for reconstructing structured sparse signals," in [*Proc. Conf. Inform. Sci. Syst.*], (2012).

[11] Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B., [*Bayesian Data Analysis*], Chapman & Hall, second ed. (2004).

[12] Koller, D. and Friedman, N., [*Probabilistic Graphical Models*], MIT Press, Cambridge, MA (2009).

(a) True Image

(b) MP-EM (PSNR = 30.47 dB)

(c) turbo-AMP (PSNR = 27.86 dB)

(d) MB-IHT (PSNR = 26.87 dB)

(e) GPSR (PSNR = 27.53 dB)

(f) FPC_{AS} (PSNR = 27.64 dB)

(g) NIHT (PSNR = 27.03 dB)

(h) DORE (PSNR = 25.82 dB)

Figure 6. The 'Cameraman' image reconstructed by various methods for $N/p = 0.375$.

[13] Weiss, Y. and Freeman, W., "On the optimality of solutions of the max-product belief-propagation algorithm in arbitrary graphs," *IEEE Trans. Inf. Theory* **47**(2), 736–744 (2001).

[14] Pearl, J., [*Probabilistic Reasoning in Intelligent systems: Networks of Plausible Inference*], Morgan Kaufmann, San Mateo, CA (1988).

[15] Wen, Z., Yin, W., Goldfarb, D., and Zhang, Y., "A fast algorithm for sparse reconstruction based on shrinkage, subspace optimization, and continuation," *SIAM J. Sci. Comput.* **32**(4), 1832–1857 (2010).

[16] Figueiredo, M., Nowak, R., and Wright, S., "Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems," *IEEE J. Sel. Topics Signal Process.* **1**(4), 586–597 (2007).

[17] Qiu, K. and Dogandžić, A., "Sparse signal reconstruction via ECME hard thresholding," *IEEE Trans. Signal Process.* **60** (2012).

[18] Blumensath, T. and Davies, M. E., "Normalized iterative hard thresholding: Guaranteed stability and performance," *IEEE J. Selected Topics Signal Processing* **4**(2), 298–309 (2010).

[19] Baraniuk, R. G., "Optimal tree approximation with wavelets," in [*Proc. SPIE Wavelet Applicat. Signal Image Processing VII*], 196–207 (1999).

[20] Do, T., Gan, L., Nguyen, N., and Tran, T., "Fast and efficient compressive sensing using structurally random matrices," *IEEE Trans. Signal Process.* **60**(1), 139–154 (2012).