Factor Graphs with NUV Priors and Iteratively Reweighted Descent for Sparse Least Squares and More

Hans-Andrea Loeliger, Boxiao Ma, Hampus Malmberg, and Federico Wadehn
ETH Zurich, Dept. of Information Technology & Electrical Engineering

***Abstract*—Normal priors with unknown variance (NUV) are well known to include a large class of sparsity promoting priors and to blend well with Gaussian message passing. Essentially equivalently, sparsifying norms (including the L1 norm) as well as the Huber cost function from robust statistics have variational representations that lead to algorithms based on iteratively reweighted L2-regularization. In this paper, we rephrase these well-known facts in terms of factor graphs. In particular, we propose a smoothed-NUV representation of the Huber function and of a related nonconvex cost function, and we illustrate their use for sparse least-squares with outliers and in a natural (piecewise smooth) prior for imaging. We also point out pertinent iterative algorithms including variations of gradient descent and coordinate descent.**

I. Introduction

It seems obvious that sparsity inducing priors in otherwise linear Gaussian problems obliterate Gaussianity. Likewise (and essentially equivalently), least-squares problems with sparsity inducing regularization are no longer least squares problems. However, Gaussianity and least-squares are so attractive that workarounds have been sought and found. For example, approximate message passing (AMP) as in [1]–[3] works with temporary Gaussian approximations, and sparse solutions of least-squares problems can be computed by iteratively reweighted least-squares methods [4], [5].

Another workaround is offered by normal priors with unknown variance (NUV), the key idea of sparse Bayesian learning [6]–[9]. Specifically, consider a generic situation with $X = (X_1, \ldots, X_K)$ (taking values in $\mathbb{R}^K$), observations $Y = y = (y_1, \ldots, y_N) \in \mathbb{R}^N$, Gaussian likelihood function $p(y|x)$, and i.i.d. prior (possibly an improper prior)

$$p(x) = \prod_{k=1}^{K} p(x_k). \tag{1}$$

Assume now that $p(x_k)$ can be written as

$$p(x_k) = \sup_{s_k \geq 0} p(x_k|s_k)\rho(s_k) \tag{2}$$

with

$$p(x_k|s_k) = \frac{1}{\sqrt{2\pi}s_k} e^{-x_k^2/2s_k^2}, \tag{3}$$

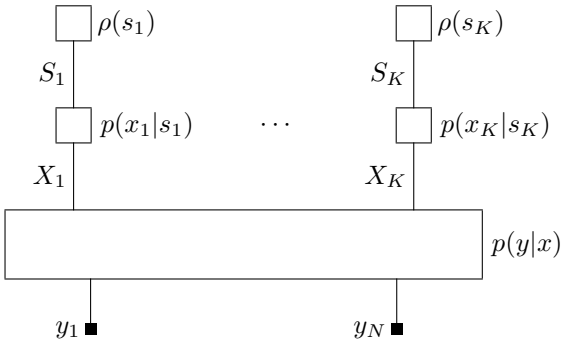

Fig. 1. Factor graph of the product in (5).

where ρ is a nonnegative function (not necessarily a probability density function). Then

$$p(y, x) = p(y|x) \prod_{k=1}^{K} \sup_{s_k \geq 0} p(x_k|s_k)\rho(s_k) \tag{4}$$

$$= \sup_{s_1, \ldots, s_K} p(y|x) \prod_{k=1}^{K} p(x_k|s_k)\rho(s_k), \tag{5}$$

as illustrated in Fig. 1.

Interestingly, (2) is not very restrictive at all: essentially all sparsity inducing priors (including, in particular, the Laplace distribution) can be represented in this way [7]. In fact, taking logarithms in (2) reveals it to be a thinly disguised variational representation as in [5], see also [10].

One way to use a NUV representation as in (2) is to estimate $s = (s_1, \ldots, s_K)$ in (5) by expectation maximization (EM) [7], [9]. In every iteration of the EM algorithm, the temporary estimate of s is plugged into (5), turning $X_1, \ldots, X_K$ into Gaussian random variables. This works very well. The main limitation is that the EM algorithm requires the posterior variances of $X_1, \ldots, X_K$ in each iteration, which may be infeasible for large problems. However, crude approximations of these variances may do, cf. [11].

In this paper, we focus on another class of algorithms that do not require the posterior variances of $X_1, \ldots, X_K$. In essence, (4) and (5) suggest algorithms to compute the MAP estimate

$$\operatorname*{argmax}_x p(y, x) = \operatorname*{argmax}_x \sup_s p(y|x) \prod_{k=1}^{K} p(x_k|s_k)\rho(s_k) \tag{6}$$

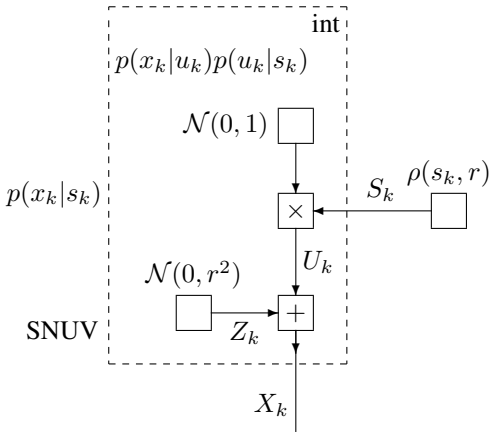

Fig. 2. Factor graph of $p(x_k|s_k)\rho(s_k,r)$ with $p(x_k|s_k)$ as in (8) and (9). The dashed box will be referred to as "smoothed NUV" (SNUV).

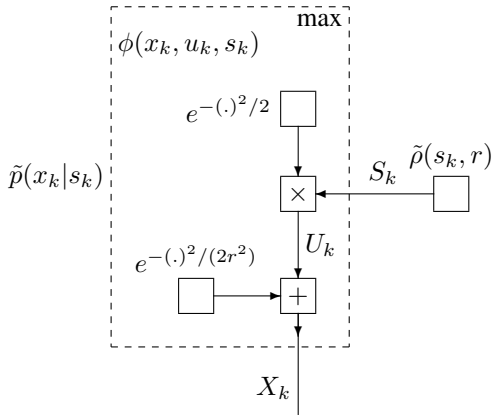

Fig. 3. Factor graph of variational representation (with U_k to be eliminated by maximization) of $\tilde{p}(x_k|s_k)\tilde{\rho}(s_k,r)$ with $\tilde{p}(x_k|s_k)$ as in (13) and (14).

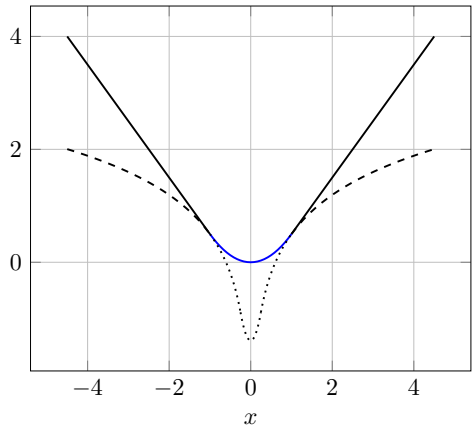

Fig. 4. Solid (black and blue): Huber function (24) with $\beta = r = 1$. Dashed and solid blue: cost function (29) with $r = 1$. Dashed and dotted: (29) with $r = 1/4$.

that iterate between an ascent step over x with fixed s and maximization over s with fixed x. The first step is entirely Gaussian while the second step decouples into easy (usually closed-form) scalar optimizations. Note that any such algorithm is guaranteed to converge to a local maximum (or a saddle point) of $p(y,x)$. In the first step, if we choose to maximize over x (rather than just ascending), we effectively obtain a reweighted-$L2$ algorithm. However, a single step of gradient ascent over x, or a round of coordinate ascent over all components of x, may be more attractive (cf. Section IV).

In the optimization literature, the advantages of such algorithms are well known [5]. In this paper, we rephrase this approach in terms of factor graphs [12], [13], with a focus on a generalization of (2) to smoothed-NUV representations that appears to be new.

The paper is structured as follows. The smoothed-NUV (SNUV) representation of some cost functions and priors will be described in Section II. Two exemplary applications will be given in Section III. Algorithms are addressed in Section IV. Section V concludes the paper.

The following notation will be used. $\mathcal{N}(\mu,\sigma^2)$ denotes a normal distribution with mean μ and variance σ^2. The k-th column of a matrix A will be denoted $A_{\cdot,k}$. A diagonal matrix with diagonal elements $\alpha_1,\ldots,\alpha_K$ will be denoted $\mathrm{diag}(\alpha_1,\ldots,\alpha_K)$. All logarithms are natural logarithms. For the factor graph notation we refer to [12], [13].

II. Smoothed-NUV Representations of Some Cost Functions

In this paper, we focus on scalar functions as in (2), but with more general factors $p(x_k|s_k)$ than (3). Specifically, we consider functions with factor graph representations as in Figures 2 and 3.

We begin with Fig. 2, which expresses X_k as the sum of two independent zero-mean normal random variables Z_k and U_k with fixed variance r^2 and unknown variance S_k^2, respectively, and with a (possibly improper) prior $\rho(s_k,r)$ on S_k. The inside of the dashed box in Fig. 2 represents the function

$$p(x_k|u_k)p(u_k|s_k) = \frac{1}{\sqrt{2\pi}r}e^{-(x_k-u_k)^2/(2r^2)}\frac{1}{\sqrt{2\pi}s_k}e^{-u_k^2/(2s_k^2)}, \tag{7}$$

and the exterior function of the dashed box is

$$p(x_k|s_k) = \int_{-\infty}^{\infty} p(x_k|u_k)p(u_k|s_k)\,du_k \tag{8}$$

$$= \frac{1}{\sqrt{2\pi(r^2+s_k^2)}}e^{-x_k^2/\left(2(r^2+s_k^2)\right)}. \tag{9}$$

We will see below that $\rho(s_k,r)$ can be chosen such that

$$-\log p(x_k) = -\sup_{s_k\geq 0}\log\left(p(x_k|s_k)\rho(s_k,r)\right) \tag{10}$$

is any of the functions plotted in Fig. 4. In the special case $r=0$, we recover the NUV representation (2). In the special case $\rho(s_k,r) = \delta(s_k)$ (the Dirac delta), $p(x_k)$ is Gaussian with variance r^2.

A. Probabilistic Representation vs. Variational Representation

Fig. 2 and Fig. 3 are similar, but not identical. The former embodies a probabilistic view: for any fixed s_k, $p(x_k|s_k)$ is a (properly normalized) Gaussian density and U_k is eliminated by marginalization. The latter embodies a variational view, where U_k is eliminated by maximization. Nonetheless, Figures 2 and 3 can represent the same set of functions $p(x_k)$, cf. (15) and (16) below.

Specifically, the inside of the dashed box in Fig. 3 represents the function

$$\phi(x_k, u_k, s_k) \triangleq e^{-(x_k-u_k)^2/(2r^2)} e^{-u_k^2/(2s_k^2)}. \tag{11}$$

Maximizing over u_k yields

$$\underset{u_k}{\operatorname{argmax}}\, \phi(x_k, u_k, s_k) = \frac{x_k s_k^2}{r^2 + s_k^2} \tag{12}$$

and further (after some calculations)

$$\tilde{p}(x_k|s_k) \triangleq \max_{u_k} \phi(x_k, u_k, s_k) \tag{13}$$

$$= e^{-x_k^2/\left(2(r^2+s_k^2)\right)}. \tag{14}$$

It follows that Fig. 2 (with integration over u_k) and Fig. 3 (with maximization over u_k) represent the same function

$$p(x_k|s_k)\rho(s_k, r) = \tilde{p}(x_k|s_k)\tilde{\rho}(s_k, r) \tag{15}$$

if we choose

$$\rho(s_k, r) = \tilde{\rho}(s_k, r)\sqrt{2\pi(r^2 + s_k^2)}. \tag{16}$$

In particular, with (16), we have

$$p(x_k) = \max_{s_k \geq 0} p(x_k|s_k)\rho(s_k, r) \tag{17}$$

$$= \max_{s_k \geq 0} \tilde{p}(x_k|s_k)\tilde{\rho}(s_k, r). \tag{18}$$

B. Huber Cost Function and L1 Norm

In order to achieve (24) below, we choose

$$\tilde{\rho}(s_k, r) = e^{-\beta^2 s_k^2/2} \tag{19}$$

or, equivalently,

$$\rho(s_k, r) = \sqrt{2\pi(r^2 + s_k^2)}\, e^{-\beta^2 s_k^2/2} \tag{20}$$

with $\beta > 0$ for $s_k \geq 0$ and $\tilde{\rho}(s_k, r) = \rho(s_k, r) = 0$ for $s_k < 0$. In this case,

$$\underset{s_k \geq 0}{\operatorname{argmax}}\, \tilde{p}(x_k|s_k)\tilde{\rho}(s_k, r)$$

$$= \underset{s_k \geq 0}{\operatorname{argmin}} \left(\frac{x_k^2}{2(r^2 + s_k^2)} + \frac{\beta^2 s_k^2}{2} \right) \tag{21}$$

$$= \begin{cases} 0, & |x_k| < \beta r^2 \\ \sqrt{|x_k|/\beta - r^2}, & |x_k| \geq \beta r^2, \end{cases} \tag{22}$$

resulting in

$$-\log p(x_k) = \kappa(x_k) \tag{23}$$

$$\triangleq \begin{cases} x_k^2/(2r^2), & |x_k| < \beta r^2 \\ \beta|x_k| - \beta^2 r^2/2, & |x_k| \geq \beta r^2. \end{cases} \tag{24}$$

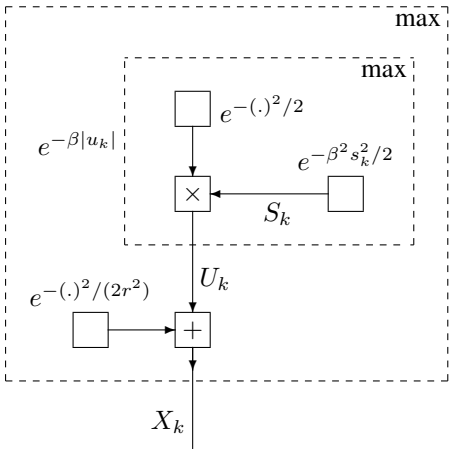

Fig. 5. Regrouping Fig. 3 corresponds to writing the Huber function as the Moreau–Yosida regularization of the absolute-value function.

This convex function is known as the Huber cost function [15], cf. Fig. 4. For $r > 0$, (24) is strictly convex and everywhere continuously differentiable.

For $r = 0$, (24) reduces to the absolute-value function. In this case, (21)–(24) amount to the standard variational representation of the absolute-value function [5]. This applies, in particular, to U_k in Fig. 3. Regrouping Fig. 3 as in Fig. 5 amounts to writing the Huber function as the Moreau–Yosida regularization (or Moreau envelope) of the absolute-value function [14, Chapt. 3].

In the iteratively reweighted algorithms of Section IV, the functions (19), (20), and (24) are not used for the actual computations, which use only (22). The same comment applies to the analogous expressions (25), (29), and (27) below.

C. Plain SNUV

By "plain SNUV", we mean that $\rho(s_k, r)$ is constant. For (29) below to look nice, we choose

$$\rho(s_k, r) = \sqrt{2\pi}. \tag{25}$$

In this case,

$$\underset{s_k \geq 0}{\operatorname{argmax}}\, p(x_k|s_k)\rho(s_k, r)$$

$$= \underset{s_k \geq 0}{\operatorname{argmin}} \left(\frac{x_k^2}{2(r^2 + s_k^2)} + \log\sqrt{r^2 + s_k^2} \right) \tag{26}$$

$$\triangleq \begin{cases} 0, & x_k^2 < r^2 \\ \sqrt{x_k^2 - r^2}, & x_k^2 \geq r^2, \end{cases} \tag{27}$$

resulting in

$$-\log p(x_k) = \kappa(x_k) \tag{28}$$

$$\triangleq \begin{cases} x_k^2/(2r^2) + \log r, & x_k^2 < r^2 \\ \log|x_k| + 1/2, & x_k^2 \geq r^2. \end{cases} \tag{29}$$

This function is not convex, cf. Fig. 4. For $r > 0$, (29) is everywhere continuously differentiable.

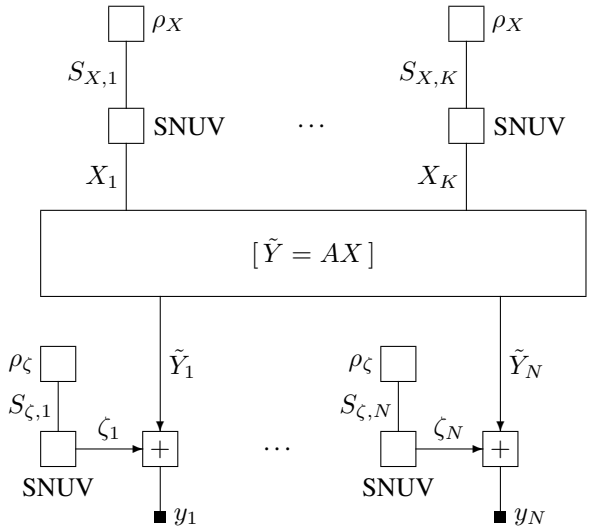

Fig. 6. Factor graph of least-squares problem as in Section III-A with smoothed-NUV factors (SNUV) as in Fig. 2. The large box represents the constraint $\tilde{Y} = AX$.

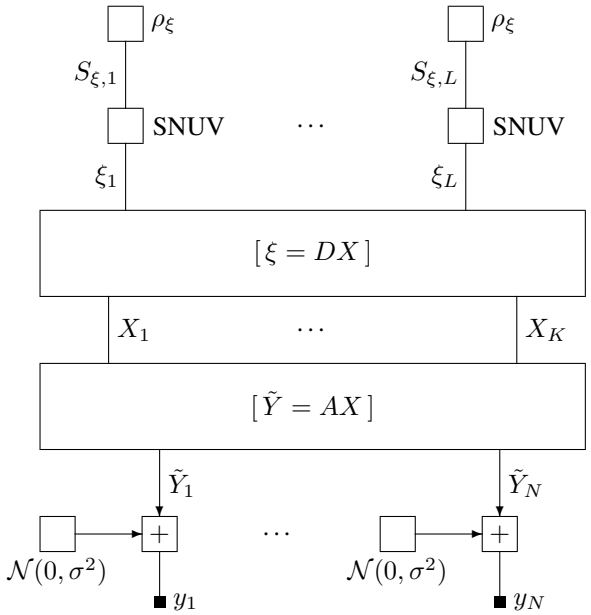

Fig. 7. Factor graph of imaging problems as in Section III-B with smoothed-NUV factors (SNUV) as in Fig. 2. The large boxes represent linear constraints.

III. Examples and Applications

A. Approximately Sparse Least Squares with Outliers

For a given matrix $A \in \mathbb{R}^{N\times K}$ and $y \in \mathbb{R}^N$, we wish to determine $x \in \mathbb{R}^K$ and $\zeta \triangleq Ax - y$ such that

$$\sum_{k=1}^{K} \kappa_X(x_k) + \sum_{n=1}^{N} \kappa_\zeta(\zeta_n) \tag{30}$$

is as small as possible, where κ_X and κ_ζ are cost functions as in (24) or as in (29). Thus $p(x_k) \triangleq e^{-\kappa_X(x_k)}$ and $p(\zeta_n) \triangleq e^{-\kappa_\zeta(\zeta_n)}$ are priors as in Section II, with parameters r_X and $s_X = (s_{X,1}, \ldots, s_{X,K})$, and r_ζ and $s_\zeta = (s_{\zeta,1}, \ldots, s_{\zeta,N})$, respectively. The corresponding factor graph is shown in Fig. 6.

Clearly, for fixed $S_X = s_X$ and fixed $S_\zeta = s_\zeta$, the factor graph reduces to a linear Gaussian factor graph.

Let $(\hat{x}, \hat{s}_X, \hat{\zeta}, \hat{s}_\zeta)$ be a maximizer of the total model $p(y, x, s_X, \zeta, s_\zeta)$, i.e., $(\hat{x}, \hat{\zeta})$ is a minimizer of (30). (In the nonconvex case, take any maximizer (or minimizer, respectively), e.g., as found by some specific algorithm.) For $r_X > 0$, $\hat{x}$ is not generally sparse. However, $\hat{s}_X$ is generally sparse even for $r_X > 0$. In typical applications, $\hat{s}_{X,k} \neq 0$ indicates a significant component of $\hat{x}$. Likewise, $\hat{s}_\zeta$ is generally sparse and $\hat{s}_{\zeta,n} \neq 0$ indicates an outlier in y.

B. Priors for Imaging

Let $X = (X_1, \ldots, X_K)$ be grayscale pixel or voxel values. (The generalization to color images is straightforward, cf. [16].) Many imaging problems (denoising, deblurring, tomographic reconstruction, ...) can be formulated as follows: based on observations $y \in \mathbb{R}^N$, we estimate X by

$$\hat{x} = \operatorname*{argmin}_x \frac{1}{\sigma^2}\|Ax - y\| + \sum_{\ell=1}^{L} \kappa(\xi_\ell) \tag{31}$$

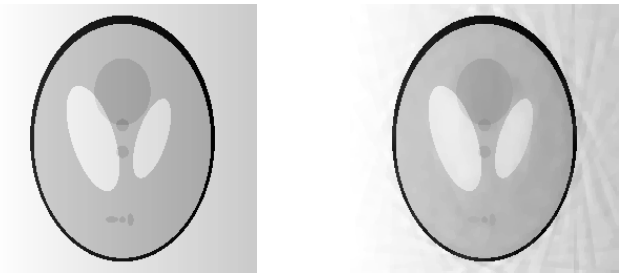

Fig. 8. Tomographic reconstruction of a standard test object with smoothly changing background from very few simulated projections: regularization with plain SNUV (left) and with standard TV (right).

with known observation matrix $A \in \mathbb{R}^{N\times K}$ and where $\xi = Dx \in \mathbb{R}^L$ (with a suitable matrix D) is the vector of all differences between neighboring pixels. If κ is chosen to be any of the cost functions in Section II, (31) can be represented by the factor graph in Fig. 7. (Similar priors have been proposed, e.g., in [17], [18].)

In our numerical experiments (focussing on tomography), the best results are achieved with plain SNUV as in (29) (apparently first proposed in [11]), which beats the Huber cost function (24), which in turn beats the standard total-variation (TV) regularization [19], cf. Fig. 8. Detailed results will be reported elsewhere.

IV. On Algorithms

We now briefly address algorithms. For the sake of concreteness, we focus on the example of Section III-A.

A. Iteratively Reweighted Coordinate Descent

An easy and safe algorithm for the minimization of (30) consists of alternating between

1) minimizing (30) with fixed s_X and s_ζ, first over x_1 (with fixed $x_2, \ldots, x_K$), then over x_2 (with fixed $x_1, x_3, \ldots, x_K$), etc., and
2) parallel updates of s_X (with fixed x) and of s_ζ (with fixed $\zeta = y - Ax$) according to (22) or (27).

In Step 1, $x_1, \ldots, x_K$ are updated according to

$$x_k^{\text{new}} = g_k^{\mathsf{T}}(y - \breve{y}_k) \tag{32}$$

with

$$\breve{y}_k \triangleq A(x_1^{\text{new}}, \ldots, x_{k-1}^{\text{new}}, 0, x_{k+1}^{\text{old}}, \ldots, x_K^{\text{old}})^{\mathsf{T}}, \tag{33}$$

$$g_k \triangleq \frac{W(s_\zeta)A_{\cdot,k}}{(r_X^2 + s_{X,k}^2)^{-1} + (A_{\cdot,k})^{\mathsf{T}}W(s_\zeta)A_{\cdot,k}} \tag{34}$$

and

$$W(s_\zeta) \triangleq \operatorname{diag}\Big((r_\zeta^2 + s_{\zeta,1}^2)^{-1}, \ldots, (r_\zeta^2 + s_{\zeta,N}^2)^{-1}\Big). \tag{35}$$

Note that (33) can be updated (rather than recomputed) for each k.

This algorithm has no parameters and is guaranteed to converge to a local minimum (or a saddle point) of (30). For convex cost functions such as (24), the algorithm is guaranteed to converge to the global minimum. Range constraints on X_k are easily accomodated.

The computational cost per execution of Step 1 is roughly the same as the cost of computing $A^{\mathsf{T}}A$, i.e., the cost of computing the gradient (with fixed variances). This algorithm is often quite efficient, especially if A is sparse.

B. On Other Algorithms

Replacing Step 1 of the above algorithm by minimization over the whole vector x yields a reweighted-$L2$ algorithm (cf. [4], [5]), which normally requires fewer iterations than the algorithm above. However, for large sparse matrices, the overall complexity is higher.

Steepest descent can be applied directly to the cost function (30). Alternatively, Step 1 of the Algorithm in Section IV-A can be replaced by a single steepest-descent step over x (with fixed variances). This latter version makes step size control easier, especially for nonconvex cost functions such as (29) with small r.

AMP as in [1], [2] works well for certain large random matrices A; otherwise, it often fails to converge. Vector AMP as in [3] requires the singular-value decomposition of A, which may be infeasible for large matrices.

A completely different approach (from sparse Bayesian learning [6]–[9]) is to first estimate the variances by expectation maximization, and then to estimate the Gaussian vector X, cf. the pertinent remarks in Section I.

V. Conclusion

Variational representations of sparsifying cost functions are naturally expressed in factor graphs. This applies, in particular, to sparse least squares problems, which are naturally represented by linear Gaussian factor graphs with NUV (normal with unknown variance) factors. Variations include approximate sparsity, outliers, and nonconvex cost functions. The underlying math is basically well known, but the specific smoothed-NUV representations of Section II appear to be new. We also point out a nonconvex prior for imaging which improves upon the state of the art.

Pertinent natural algorithms iterate between a Gaussian ascent (or least-squares descent) step and closed-form scalar maximizations over the unknown variances. An obvious version of the former is coordinate ascent (or descent), which has no parameters and is guaranteed to converge.

References

[1] D. L. Donoho, A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sensing", *Proc. National Academy of Sciences,* vol. 106, no. 45, pp. 18914–18919, 2009.

[2] M. Borgerding, P. Schniter, J. Vila, and S. Rangan, "Generalized approximate message passing for cosparse analysis compressive sensing," *40th IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP),* Brisbane, Australia, April 19–24, 2015, pp. 3756–3760.

[3] S. Rangan, Ph. Schniter, and A. K. Fletcher, "Vector approximate message passing," *2017 IEEE Int. Symp. on Information Theory,* Aachen, Germany, June 2017.

[4] I. Daubechies, R. DeVore, M. Fornasier, and C. S. Güntürk, "Iteratively reweighted least squares minimization for sparse recovery," *Comm. Pure & Appl. Math.,* Vol. 63, No. 1, pp. 1–38, 2010.

[5] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski "Optimization with sparsity-inducing penalties," *Foundations and Trends in Machine Learning:* Vol. 4, No. 1, pp. 1–106, 2012.

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

[7] J. Palmer, D. P. Wipf, K. Kreutz-Delgado, and B. D. Rao, "Variational EM algorithms for non-Gaussian latent variable models," *Adv. Neural Inf. Proc. Systems (NIPS),* 2006.

[8] D. Wipf and S. Nagarajan, "A new view of automatic relevance determination," *Adv. Neural Inf. Proc. Systems (NIPS),* pp. 1625–1632, 2008.

[9] H.-A. Loeliger, L. Bruderer, H. Malmberg, F. Wadehn, and N. Zalmai "On sparsity by NUV-EM, Gaussian message passing, and Kalman smoothing," *2016 Information Theory & Applications Workshop (ITA),* San Diego, CA, Feb. 2016.

[10] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul, "An Introduction of Variational Methods for Graphical Models," *Machine Learning,* vol. 37, pp. 183–233, 1999.

[11] N. Zalmai, C. Luneau, C. Stritt, and H.-A. Loeliger, "Tomographic reconstruction using a new voxel-domain prior and Gaussian message passing," *2016 Europ. Signal Proc. Conf. (EUSIPCO),* Budapest, Hungary, Aug. 29 – Sept. 2, 2016.

[12] H.-A. Loeliger, "An introduction to factor graphs," *IEEE Signal Proc. Mag.,* Jan. 2004, pp. 28–41.

[13] H.-A. Loeliger, J. Dauwels, Junli Hu, S. Korl, Li Ping, and F. R. Kschischang, "The factor graph approach to model-based signal processing," *Proceedings of the IEEE,* vol. 95, no. 6, pp. 1295–1322, June 2007.

[14] N. Parikh and S. Boyd, *Proximal Algorithms.* now publishers, Foundations and Trends in Optimization, vol. 1, 2014.

[15] P. J. Huber and E. M. Roncetti, *Robust Statistics,* 2nd ed. John Wiley & Sons, 2009.

[16] Boxiao Ma, N. Zalmai, R. Torfason, C. Stritt, and H.-A. Loeliger, "Color image segmentation using iterative edge cutting, NUV-EM, and Gaussian message passing," *5th IEEE Global Conf. on Signal and Information Processing (GlobalSIP 2017)*, Montreal, Canada, Nov. 14–16, 2017.

[17] P. Charbonnier, L. Blanc-Féraud, G. Aubert, and M. Barlaud, "Deterministic edge-preserving regularization in computed imaging," *IEEE Trans. Image Proc.,* vol. 6, pp. 298–311, Feb. 1997.

[18] M. Foare, N. Pustelnik, and L. Condat, "A New Proximal Method for Joint Image Restoration and Edge Detection with the Mumford-Shah Model," *2018 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP),* Calgary, Canada, April 15–20, 2018.

[19] E. Y. Sidky and X. Pan, "Image reconstruction in circular cone-beam computed tomography by constrained, total-variation minimization," *Physics in Medicine and Biology,* vol. 53, no. 17, pp. 4777–4807, 2008.