Low-Latency Reweighted Belief Propagation Decoding for LDPC Codes

Jingjing Liu and Rodrigo C. de Lamare

***Abstract*—In this paper we propose a novel message passing algorithm which exploits the existence of short cycles to obtain performance gains by reweighting the factor graph. The proposed decoding algorithm is called variable factor appearance probability belief propagation (VFAP-BP) algorithm and is suitable for wireless communications applications with low-latency and short blocks. Simulation results show that the VFAP-BP algorithm outperforms the standard BP algorithm, and requires a significantly smaller number of iterations when decoding either general or commercial LDPC codes.**

***Index Terms*—Belief propagation, reweighting the factor graph, message passing, low-latency, commercial LDPC codes.**

I. Introduction

LOW-DENSITY parity-check (LDPC) codes are recognized as a class of linear block codes which can achieve near-Shannon capacity with linear-time encoding and parallelizable decoding algorithms. LDPC codes were first invented by Gallager in his doctoral dissertation [1] and rediscovered by MacKay *et al.* in the 1990s [2]. The advantages of LDPC codes arise from the sparse (low-density) parity-check matrices which can be uniquely depicted by graphical representations, referred to as Tanner graphs [3]. Equipped with efficient decoders, LDPC codes have found applications in a number of communication standards, such as DVB-S2 and Wi-Fi 802.11.

The belief propagation (BP) algorithm, sometimes also called sum-product algorithm (SPA), is a powerful algorithm to approximately solve inference problems in statistical physics, computer vision, distributed hypothesis testing, cooperative localization, and error control coding. This message passing algorithm computes accurate marginal distributions of variables corresponding to each node of a graphical model, and is exceptionally useful when optimal inference decoding is computationally prohibitive due to the large size of a graph. Additionally, the BP algorithm is capable of producing the exact inference solutions if the graphical model is acyclic (i.e., a tree), while it does not guarantee to converge if the graph possesses short cycles which significantly deteriorate the overall performance [4]. Since the BP algorithm started to be applied as a decoding algorithm for turbo and LDPC codes, various versions of BP graph-based decoding algorithms have been reported in the area. However, the lack of a convergence guarantee and the high-latency due to many decoding iterations are still open issues for researchers when it comes to effectively decoding LDPC codes in wireless communications applications, where a large amount of data transmission and data storage are required. Recently, Wymeersch *et al.* [5], [6] introduced the uniformly reweighted BP (URW-BP) algorithm which exploits BP's distributed nature and reduces the factor appearance probability (FAP) in [4] to a constant value. In [6], the URW-BP has been shown to outperform the standard BP in terms of LDPC decoding among other applications.

In this paper, we investigate the idea of reweighting a suitable part of the factorized graph while also statistically taking the effect of short cycles into account. By combining the reweighting strategy with the knowledge of the short cycles obtained by the cycle counting algorithms [8], [9], we present the variable FAP BP (VFAP-BP) algorithm. The VFAP-BP algorithm assigns distinct FAP values to each parity-check node on the basis of the structure of short cycles rather than a complex global graphical optimization. We also extend the application of reweighted message passing decoding algorithms from symmetric graphs to asymmetric graphs. Simulation results show that the proposed VFAP-BP algorithm consistently outperforms URW-BP for irregular LDPC codes, and offers a better bit-error rate (BER) performance than the standard BP for both regular and irregular codes when using a small number of iterations. As a result, VFAP-BP considerably improves the convergence behavior of the BP decoder, which allows a lower decoding latency.

The organization of this paper is as follows: Section II introduces the background in terms of understanding standard BP rules and URW-BP algorithms. In Section III, the proposed VFAP-BP algorithm is presented in detail. Section IV shows the simulation results along with discussions. Finally, Section V concludes the paper.

II. Standard BP Algorithm and its Variations

A. Standard BP Algorithm for Decoding LDPC Codes

Suppose we have K information bits being transmitted and a set of codewords $\boldsymbol{x}$ with block length N is formed by an LDPC encoder, such that the code rate R is K/N and the corresponding parity-check is an $M \times N (M = N - K)$ sparse matrix $\boldsymbol{H}$ containing at least 99% of 0 entries. After the transmission, the objective of the decoder is to find an $1 \times N$ estimated codeword $\hat{\boldsymbol{x}}$ which satisfies the parity-check condition $\boldsymbol{H}\hat{\boldsymbol{x}}^{\boldsymbol{T}} = \boldsymbol{0}$. Thus, we can interpret the decoding process as finding $\hat{\boldsymbol{x}} = \arg\max p(\boldsymbol{x}|\boldsymbol{y})$. Using Bayes' rule the a posteriori distribution becomes

$$p(\boldsymbol{x}|\boldsymbol{y}) = \frac{p(\boldsymbol{y}|\boldsymbol{x})p(\boldsymbol{x})}{p(\boldsymbol{y})}, \tag{1}$$

Manuscript received June 13, 2012. The associate editor coordinating the review of this letter and approving it for publication was L. Dolecek.

The authors are with Department of Electronics, The University of York, Heslington, York, YO10 5DD, UK (e-mail: {jl622; rcdl500}@ohm.york.ac.uk).

The authors are grateful to Henk Wymeersch who provided MATLAB codes of URW-BP algorithm for comparison purpose.

Digital Object Identifier 10.1109/LCOMM.2012.12.121307

where the likelihood ratios $p(\boldsymbol{y}|\boldsymbol{x})$ can be obtained from the channel and $p(\boldsymbol{x})$ is the prior information. Nevertheless, directly calculating $p(\boldsymbol{x}|\boldsymbol{y})$ or $p(\boldsymbol{y})$ is computationally prohibitive because of the size of $\boldsymbol{x}$ [4]. For this reason, we resort to BP as a near-optimal message passing algorithm which can approximate either $p(\boldsymbol{x}|\boldsymbol{y})$ or $p(\boldsymbol{y})$.

In the application of decoding, the BP algorithm performs distributed local computations so as to approximate a maximum likelihood solution of $p(x_j|\boldsymbol{y})$ for $(j = 0, 1, \ldots, N-1)$. The LDPC codes can be represented by a factor graph where the M square nodes stand for M parity-check equations and the N circle nodes relate to N encoded binary bits. There is an edge connecting the check node C_i and the variable node V_j in the factor graph if the entry h_{ij} of the parity-check matrix $\boldsymbol{H}$ equals 1. All the check nodes and the variable nodes work cooperatively and iteratively so as to estimate $p(x_j|\boldsymbol{y})$ for $(j = 0, 1, \ldots, N-1)$ [3]. Following a set of message passing rules, the variable nodes (check nodes) process the incoming message and send the extrinsic information to their neighboring check nodes (variable nodes) back and forth in an iterative fashion, until all M parity check conditions are met ($\boldsymbol{H}\hat{\boldsymbol{x}}^{\boldsymbol{T}} = \boldsymbol{0}$) or the decoder reaches the maximum number of iterations.

B. URW-BP Algorithm for High-Order Interactions

When the factor graph is a tree with no cycles, the standard BP algorithm is able to perform accurate approximation in a few iterations. In the presence of cycles, it normally requires a larger number of iterations and may fail to converge [2]. In [4], the authors developed a novel TRW-BP algorithm which improves the convergence of BP by reweighting certain portions of the factorized graphical representation. However, the TRW-BP algorithm only considers a factorized graph with pairwise interactions, and is not suitable for distributed inference problems since it optimizes the reweighting parameters over spanning trees. These issues have been addressed by the URW-BP algorithm reported in [6], which extends the pairwise factorizations of TRW-BP to hypergraphs, and replaces a series of globally optimized parameters with a simple constant. With a small number of decoding iterations, the URW-BP algorithm has been verified to outperform the standard BP algorithm for regular LDPC codes that possess a roughly uniform structure [5]. However, how to choose the optimal ρ is still an open issue.

III. Proposed VFAP-BP Decoding Algorithm

This section presents the proposed VFAP-BP algorithm, in which we devise a simple criterion for determining the reweighting parameters so as to improve the decoding performance with respect to both regular and irregular LDPC codes. The idea behind the proposed algorithm is inspired by the fact that the existence of short cycles creates the statistical dependency among the incoming messages being exchanged by nodes, such that the outgoing messages inaccurately have a high reliability or equivalently a low quality. This problem is often referred as "overconfident" or "overestimation" [4], [6]. As shown in Fig. 1, the URW-BP tackles the "overconfidence" by delegating a uniform reweighting parameter ρ_u to each parity-check node, resulting in less concentrated and more robust beliefs [6]. On the other hand, it is well-known that not all short cycles are equally detrimental with respect to decoding performance. Specifically, check nodes having a large number of short cycles are more likely to form clusters of small cycles, which significantly obstruct the convergence of BP algorithm within limited iterations [7]. By assigning various FAP values $\rho_i(i = 0, 1, \ldots, M-1)$ in Fig. 1, the proposed VFAP-BP algorithm takes advantage of the reweighting strategy as well as the knowledge of the structure of short cycles. According to [6], URW-BP and the optimized TRW-BP are equivalent as the factor graph G has a symmetric factorization that refers to codes with regular design. Otherwise, a uniform choice of ρ does not guarantee to improve the convergence of the BP algorithm. On the other hand, a symmetric factor graph is not required for the proposed VFAP-BP algorithm since it adjusts the reweighting parameter based on the knowledge of short cycles, rather on the factorization of the graph. For this reason, it is also suitable for LDPC codes with irregular designs. In the following, we briefly explain the algorithm that we employ to count short cycles in the factor graph, then introduce the message passing rules and the VFAP-BP decoding algorithm flow.

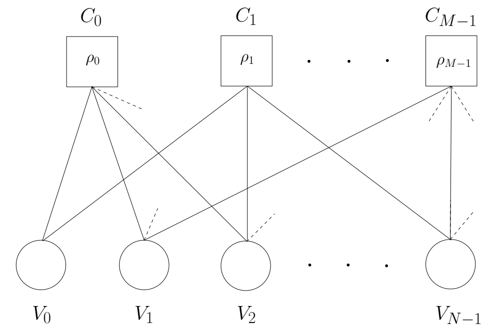

Fig. 1. The graphical model depicts BP decoding algorithms for LDPC codes, where $\rho_i(i = 0, 1, \ldots, M-1) = 1$ corresponds to the standard BP, $\rho_i(i = 0, 1, \ldots, M-1) = \rho_u$ corresponds to the URW-BP, and $\rho_i(i = 0, 1, \ldots, M-1) = \rho_v$ or 1 depending on a variable condition corresponds to the proposed VFAP-BP.

Counting short cycles exactly in an arbitrary graph seems computationally impossible. However, the cycle counting algorithm [8] transforms the problem of counting cycles into that of counting the so-called lollipop walks through matrix multiplications. Note that the counting cycle algorithm in [9] can also be applied and works more efficiently when the sparse graph becomes larger. As a consequence, resorting to either algorithm provides the knowledge of the girth g in the factor graph and the number of length-g cycles with respect to every check node $C_i(i = 0, 1, \ldots, M-1)$. In this work, we focus on the value of g, $s_i(i = 0, 1, \ldots, M-1$ the number of length-g cycles passing a check node C_i, and μ_g the average number of length-g cycles passing a check node. In a similar way to [4] and [6], the reweighting vector $\rho_i = [\rho_0, \rho_1, \ldots, \rho_{M-1}]$ consists of variable factor appearance probabilities (FAP), which originally describe the probabilities of any check node appearing in a potential spanning tree. As shown in Fig. 1, every check node C_i is assigned to a FAP value such that the outgoing messages from a check node are either unchanged or partially reweighted. This depends on whether the outgoing massages from a check node contribute to the extrinsic message passing or not. A check node obstructs convergence or lead to low-quality beliefs due to creating dependency within the cluster of short cycles. As a result, two cases can be distinguished by a simple criterion: if $s_i < \mu_g$ the check node C_i is regarded as constructive then $\rho_i = 1$; otherwise this check node is determined as a destructive node and we have $\rho_i = \rho_v$, where $\rho_v = 2/\bar{n_D}$, and $\bar{n_D}$ is the the average connectivity for N variable nodes which is computed as

$$\bar{n_D} = \frac{1}{\int_0^1 \upsilon(x)dx} = \frac{M}{N\int_0^1 \nu(x)dx}, \qquad (2)$$

where $\upsilon(x)$ and $\nu(x)$ are the distributions of variable nodes and check nodes.

The message passing rules of the proposed VFAP-BP algorithm are similar to those derived in [6] for URW-BP algorithm. We denote the beliefs by log-likelihood ratios (LLRs), and are initialized by $L(x_j) = \log\frac{p(y_j|x_j=1)}{p(y_j|x_j=0)} = 2\frac{y_j}{\sigma^2}$ for an additive white Gaussian noise (AWGN) channel, where σ^2 is the noise variance. The message sent from V_j to C_i is given by

$$\Psi_{ji} = L(x_j) + \sum_{i'\in\mathcal{N}(j)\backslash i} \rho_{i'}\Lambda_{i'j} - (1-\rho_i)\Lambda_{ij}. \qquad (3)$$

where $i' \in \mathcal{N}(j)\backslash i$ is the neighboring set of check nodes of V_j except C_i. The quantity Λ_{ij} denotes messages sent from C_i to V_j in the previous iteration, then for all check nodes C_i for $(i = 0, 1, \ldots, M-1)$ we update Λ_{ij} as:

$$\Lambda_{ij} = 2\tanh^{-1}\Big(\prod_{j'\in\mathcal{N}(i)\backslash j} \tanh\frac{\Psi_{j'i}}{2}\Big), \qquad (4)$$

where ‘$\tanh(\cdot)$‘ denotes the hyperbolic tangent function. Finally, we have the belief $b(x_j)$ with respect to x_j described by

$$b(x_j) = L(x_j) + \sum_{i\in\mathcal{N}(j)} \rho_i\Lambda_{ij}. \qquad (5)$$

Using the above message passing rules, the proposed VFAP-BP decoding algorithm is depicted in Table I. Note that $\rho_v = 2/\bar{n_D}$ at the initialization is an approximation of the optimized FAP value according to [4]. As an improvement to the URW-BP, the proposed VFAP-BP requires an additional complexity of $\mathcal{O}(gN)$ due to the cycle counting algorithm. Nevertheless, the extra complexity is very small when compared to a global optimization with complexity of $\mathcal{O}(M^2N)$. Notice that the computation of counting cycles can be further simplified if the algorithm in [9] is applied for larger sparse graphs. More importantly, the proposed algorithm is capable of improving the performance of BP to decode LDPC codes with uniform structures (regular codes) and with non-uniform structures (irregular codes).

IV. Simulation Results

In this section, we compare the proposed VFAP-BP with the standard BP and URW-BP using simulations. To illustrate the potential application of the proposed algorithm, we have tested a wide range of LDPC codes with different design methods, two of which are MacKay's regular codes [2] and irregular Quasi-cyclic (QC)-LDPC codes selected as standard codes for WiMax 802.16e [10]. The regular code $(3, 6)$ has a block length $N = 1000$ and rate $R = 0.5$, while the irregular code has a block length $N = 576$, rate $R = 0.5$, and degree distributions $\upsilon(x) = 0.21 \times x^5 + 0.33 \times x^2 + 0.46 \times x$ and $\nu(x) = 0.33 \times x^6 + 0.67 \times x^5$. Notice that for the purpose of a fair comparison the optimized ρ_u of URW-BP is acquired from the numerical method, similarly to [5], [6], which is normally larger than $2/\bar{n_D}$ ($\rho_u \approx 0.92$ for the regular code while $\rho_u \approx 0.85$ for the irregular code).

In Fig. 2, at different signal-to-noise ratios (SNRs) the convergence behaviors of the proposed VFAP-BP, the URW-BP, and the standard BP algorithms are compared for decoding the regular LDPC code within a small number of iterations. The VFAP-BP converges faster than the other algorithms at SNR of 2 dB but its advantage diminishes at higher SNR values, resulting from the fact that for the standard BP or

TABLE I
THE ALGORITHM FLOW OF THE VFAP-BP ALGORITHM

Initialization:

1: Find the girth g and s_i the number of length-g cycles passing the check node C_i ;
2: Determine variable FAPs for each check node: if $s_i < \mu_g$ $\rho_i = 1$, otherwise $\rho_i = \rho_v$ where $\rho_v = 2/\bar{n_D}$;

VFAP-BP decoding:

Step 1: Set I_{max} the maximum number of iterations and initialize $L(\boldsymbol{x}) = 2\frac{\boldsymbol{y}}{\sigma^2}$;
Step 2: Update the message passed from variable node V_j to check node C_i using (3), where $\Lambda_{i'j}$ and Λ_{ij} are $0s$ at the first iteration;
Step 3: Update the message passed from variable node C_i to check node V_j;
Step 4: Update the belief $b(x_j)$ using (5) and decide $\hat{\boldsymbol{x}}$;
Step 5: Decoding stops if $\boldsymbol{H}\hat{\boldsymbol{x}}^{\boldsymbol{T}} = \boldsymbol{0}$ or I_{max} is reached, otherwise go back to Step 2.

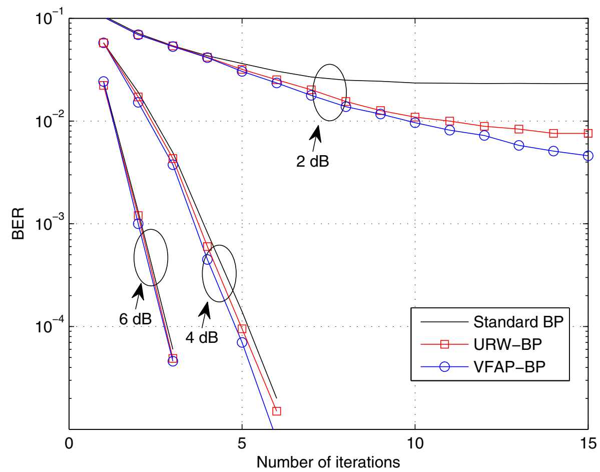

Fig. 2. Comparison of the convergent behaviors of the URW-BP, VFAP-BP and standard BP algorithms for decoding regular LDPC codes, where SNR equals to 2 dB, 4 dB and 6 dB.

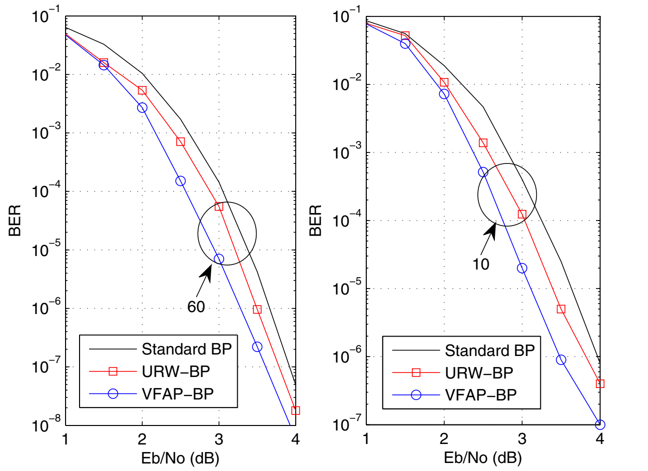

Fig. 3. Comparison of the BER performances of the VFAP-BP, URW-BP and standard BP algorithms while decoding regular LDPC codes with a maximum of 10 and 60 decoding iterations.

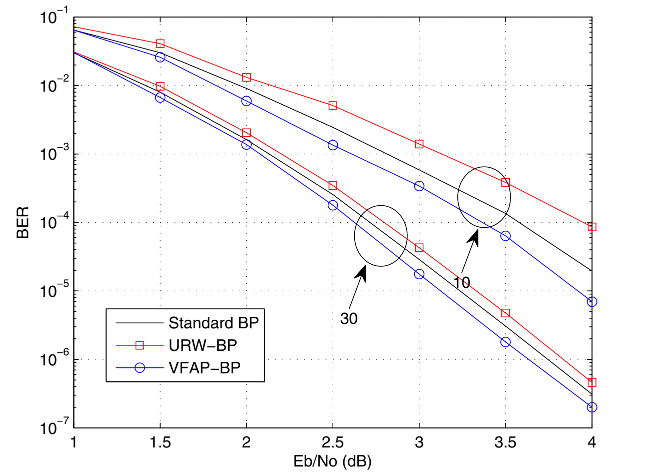

Fig. 5. Comparison of the BER performances of the VFAP-BP, URW-BP and standard BP algorithms while decoding irregular QC-LDPC codes with a maximum of 10 and 30 decoding iterations.

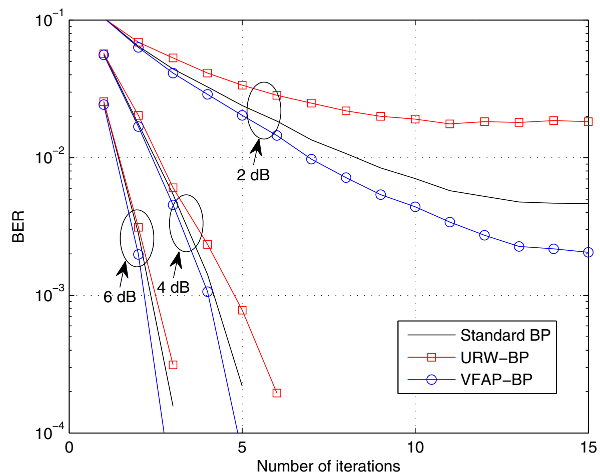

Fig. 4. Comparison of the convergent behaviors of the URW-BP, VFAP-BP and standard BP algorithms for decoding irregular QC-LDPC codes, where SNR equals to 2 dB, 4 dB and 6 dB.

URW-BP with a uniform FAP the convergence guarantees are strengthened when the noise variance is reduced [11]. Fig. 3 reveals the decoding performances of three algorithms where the VFAP-BP outperforms others whereas the performance gain decreases as more iterations are performed. In the case of irregular codes, the proposed VFAP-BP algorithm still works better than the standard BP while the asymmetric factorization of the irregular graph deteriorates the performance of URW-BP, as shown in Fig. 4. Moreover, Fig. 5 demonstrates that the VFAP-BP outperforms the standard BP up to 0.5 dB with a maximum of 10 iterations, even though the performance gap narrows when the number of iterations increases. Consequently, for both regular and irregular codes the proposed VFAP-BP is able to provide a better decoding performance than URW-BP and the standard BP with a limited number of iterations.

V. Conclusion

In this paper, we have devised a message passing decoding algorithm that employs the reweighting approach and exploits the knowledge of the graph structure with short cycles. The proposed VFAP-BP algorithm has shown a good convergence behavior when compared to the standard BP and the URW-BP algorithms within a limited number of decoding iterations, which is desirable in wireless communication systems with low delay or low latency requirements. Unlike URW-BP, VFAP-BP can also improve the decoding performance over the standard BP when decoding irregular LDPC codes, since it does not require a symmetric factor graph.

References

[1] R. G. Gallager, “Low-density parity check codes,” *IRE Trans. Inf. Theory.*, vol. 39, no. 1, pp. 37–45, Jan. 1962.

[2] D. J. C. Mackay and R. M. Neal, “Near Shannon limit performance of low density parity check codes,” *Electron. Lett.*, vol. 33, no. 6, pp. 457–458, Mar. 1997.

[3] R. M. Tanner, “A recursive approach to low complexity codes,” *IEEE Trans. Inf. Theory*, vol. 27, no. 9, pp. 533–547, Sep. 1981.

[4] M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky, “A new class of upper bounds on the log partition function,” *IEEE Trans. Inf. Theory*, vol. 51, no. 7, pp. 2313–2335, July 2005.

[5] H. Wymeersch, F. Penna, and V. Savic, “Uniformly reweighted belief propagation: a factor graph approach,” in *Proc. 2011 IEEE International Symposium on Information Theory*, pp. 2000–2004.

[6] H. Wymeersch, F. Penna, and V. Savic, “Uniformly reweighted belief propagation for estimation and detection in wireless networks,” *IEEE Trans. Wireless Commun.*, vol. 11. no. 4, pp. 1587–1595, Apr. 2012.

[7] T. Tian, C. R. Jones, J. D. Villasenor, and R. D. Wesel, “Selective avoidance of cycles in irregular LDPC code construction,” *IEEE Trans. Commun.*, vol. 52, no. 8, pp. 1242–1247, Aug. 2004.

[8] T. R. Halford and K. M. Chugg, “An algorithm for counting short cycles in bipartite graph,” *IEEE Trans. Inf. Theory*, vol. 52, no. 1, pp. 287–292, Jan. 2006.

[9] M. K. Dehkordi and A. H. Banihashemi, “A message-passing algorithm for counting short cycles in a graph,” *2010 IEEE Information Theory Workshop*.

[10] IEEE 802.16e, Air interface for fixed and mobile broadband wireless access systems, IEEE P802.16e/D12 Draft, Oct. 2005.

[11] T. G. Roosta, M. J. Wainwright, and S. S. Sastry, “Convergence analysis of reweighted sum-product algorithms,” *IEEE Trans. Signal Process.*, vol. 56, no. 9, pp. 4293–4305, Sep. 2008.