UNITARY APPROXIMATE MESSAGE PASSING FOR MATRIX FACTORIZATION

Zhengdao Yuan‡, Qinghua Guo*, Yonina C. Eldar†, Yonghui Li†

‡ Open University of Henan, Zhengzhou 450002, China
* University of Wollongong, Wollongong, NSW 2522, Australia
† Weizmann Institute of Science, Rehovot 7610001, Israel
+ University of Sydney, Sydney, NSW 2006, Australia

ABSTRACT

We consider matrix factorization (MF) with certain constraints, which finds wide applications in various areas. Leveraging variational inference (VI) and unitary approximate message passing (UAMP), we develop a Bayesian approach to MF with an efficient message passing implementation, called UAMP-MF. With proper priors imposed on the factor matrices, UAMP-MF can be used to solve a range of problems formulated as MF, such as dictionary learning, compressive sensing with matrix uncertainty, robust principal component analysis, etc. Numerical examples are provided to show that UAMP-MF significantly outperforms state-of-the-art algorithms in terms of computational complexity, recovery accuracy and robustness.

Index Terms— Variational inference (VI), approximate message passing (AMP), matrix factorization (MF).

1. INTRODUCTION

We consider the problem of matrix factorization (MF) with the following model

\[ Y = HX + W, \]

where \( Y \in \mathbb{R}^{M \times L} \) denotes a known (observation) matrix, \( W \in \mathbb{R}^{M \times L} \) accounts for unknown perturbations or measurement errors, and \( H \in \mathbb{R}^{M \times N} \) and \( X \in \mathbb{R}^{N \times L} \) are two factor matrices to be obtained. Depending on concrete application scenarios, the MF problem in (1) often has specific requirements (constraints) on the matrices \( H \) and \( X \). For example, in dictionary learning (DL) [1], \( X \) is a sparse matrix and \( H \) is a dictionary matrix to be learned. In compressive sensing with matrix uncertainty (CSMU) [2], \( X \) is sparse and the sensing matrix \( H \) can be modeled as \( H = H' + H'' \), where the matrix \( H' \) is known, and \( H'' \) denotes an unknown perturbation matrix. The robust principal component analysis (RPCA) problem [3] can also be formulated as (1), where both \( H \) and \( X \) admit some structures [4].

A multitude of methods have been developed to solve various MF problems, such as the K-singular value decomposi-

tion (K-SVD) [5] and sparse modeling software (SPAMS) [6] for DL; inexact augmented Lagrange multiplier (IALM) [7], low-rank matrix fitting (LMaFit) [8] and Grassmannian robust adaptive subspace tracking (GRASTA) for RPCA [9]. Although these algorithms were specifically designed for particular problems, their performance or computational cost may still be a concern. Another line of the MF techniques is based on the approximate message passing (AMP) algorithm [10] and its variants, taking advantage of the low complexity of AMP. The bilinear generalized AMP (BiG-AMP) algorithm was proposed in [4] to deal with some MF problems, which, however, inherits the shortcoming of AMP and lacks robustness when applied to generic matrices \( H \) and \( X \). To achieve better robustness, a bilinear adaptive vector AMP (BA-AMP) algorithm was proposed in [11] and generalized in [12]. However, it treats \( H \) as a deterministic unknown matrix, and is incapable of handling constraints on \( H \). Moreover, there is still substantial room for improvement in accuracy, robustness and computational complexity.

In this work, leveraging variational inference (VI) [13] and unitary AMP (UAMP) [14], [15], we develop a more efficient Bayesian approach to MF. In choosing the variational distribution, instead of using the mean field approximation with full factorization, we only decouple \( H \) and \( X \) and treat them as two latent matrices to avoid performance loss, leading to the updates of two distributions on \( H \) and \( X \), which are difficult and expensive. By exploiting the structure of the variational messages on \( H \) and \( X \) and through a covariance matrix whitening process, we incorporate UAMP into VI to efficiently handle the distribution updates on \( H \) and \( X \), leading to a message passing algorithm called UAMP-MF. Enjoying the flexibility of a Bayesian approach and low complexity and robustness of UAMP, UAMP-MF can solve various MF problems in a unified way while with high efficiency. Due to space constraints, we focus on its applications to DL and RPCA in this paper. For more applications, refer to the extended version accessible at https://arxiv.org/abs/2208.00422.

2. DESIGN OF UAMP-MF

2.1. Probabilistic Formulation

With the Bayesian treatment of MF by transforming the constraints on matrices \( H \) and \( X \) to their priors \( p(H) \) and \( p(X) \)
properly, many MF problems can be handled in a unified way. In this work, we assume that the priors are separable, i.e., \( p(H) = \prod_{n} p(h_{m,n}) \) and \( p(X) = \prod_{t} p(x_{n,t}) \), which can be used for DL and RPCA, as shown in Section 3. With model (1), we have the following joint conditional distribution and its factorization

\[
p(X, H, \lambda | Y) \propto p(Y | X, H, \lambda) p(X) p(H)p(\lambda),
\]

where we assume that the entries of \( W \) are i.i.d. Gaussian with zero mean and precision \( \lambda \), and \( p(Y | X, H, \lambda) \) is a matrix Gaussian distribution, i.e.,

\[
p(Y | X, H, \lambda) = \mathcal{MN}(Y; HX, I_M, \lambda^{-1} I_L).
\]

In addition, we assume a Jefferys prior \( p(\lambda) \propto 1/\lambda \) [16] for the noise precision.

If the a posteriori distributions \( p(H|Y) \) and \( p(X|Y) \) can be found, then the estimates of \( H \) and \( X \) can be obtained, e.g., using the a posteriori means of \( H \) and \( X \) to serve as their estimates. However, it is intractable to find the exact a posteriori distributions in general, so we resort to VI to find their variational approximations. To this end, we define a variational distribution

\[
q(X, H, \lambda) = q(X)q(H)q(\lambda),
\]

and by minimizing the KL divergence between the true distribution and variational distribution, we expect that \( q(X) \approx p(X|Y) \), \( q(H) \approx p(H|Y) \) and \( q(\lambda) \approx p(\lambda|Y) \) However, finding the variational distributions is still challenging due to the high dimensions of \( H \) and \( X \), and the priors of \( H \) and \( X \) may also lead to intractable \( q(H) \) and \( q(X) \).

In this work, VIAMP is employed to solve these challenges with high efficiency. This also leads to Gaussian approximations to \( q(H) \) and \( q(X) \), so that the estimates of \( H \) and \( X \) (i.e., the a posteriori means of \( H \) and \( X \)) appear as the parameters of the distributions. Leveraging VIAMP, we carry out VI in a message passing manner, with the aid of a factor graph representation of the problem, depicted in Fig. 1. This leads to the message passing algorithm VIAMP-MF.

### 2.2. Algorithm Design

According to VI, \( q(X) \), \( q(H) \) and \( q(\lambda) \) are updated in an iterative manner. First, with \( q(H) \) and \( q(\lambda) \), we update \( q(X) \). As shown by the factor graph in Fig. 1, we need to compute the message \( m_{f_{X} \rightarrow X}(X) \) from the factor node \( f_{X} \) to the variable node \( X \) and then combine it with the prior \( p(X) \). Later we will see that \( q(H) \) is a matrix Gaussian distribution, i.e.,

\[
q(H) = \mathcal{MN}(H; \hat{H}, U_H, V_H)
\]

with a mean matrix \( \hat{H} \) (see Line 23 of Algorithm 1), a column covariance matrix \( V_H \) and row covariance matrix \( U_H = I_N \), and \( q(\lambda) \) is a Gamma distribution. It can be shown that the message from \( f_{X} \) to \( X \) can be expressed as a matrix Gaussian distribution (see the extended version for derivation), i.e.,

\[
m_{f_{X} \rightarrow X}(X) \propto \mathcal{MN}(X; \hat{X}, \hat{\lambda}^{-1} U_X, I_L),
\]

where \( \hat{X} = U_X \hat{H}^T Y \),

\[
U_X = (H^T H + \text{Tr}(U_H)V_H)^{-1},
\]

and \( \hat{\lambda} = \int \lambda q(\lambda) \). The computation of \( \hat{\lambda} \) is shown in (15).

Then the message \( m_{f_{Y} \rightarrow X}(X) \) needs to be combined with the prior \( p(X) \) to obtain \( q(X) \). Note that \( X = [x_1, \ldots, x_t] \) and \( \hat{X} = [\hat{x}_1, \ldots, \hat{x}_t] \). The result in (5) indicates that \( x_i \sim N(\hat{x}_i; \lambda, \lambda^{-1} U_X) \), and all vectors in \( X \) have a common covariance matrix, which will greatly simplify the computations later. With this result, for each \( x_i \), we have the following pseudo observation model

\[
x_i = x_i + e_i,
\]

where \( e_i \sim N(0, \lambda^{-1} U_X) \), i.e., the model noise is not white. We can whiten the noise by left-multiplying both sides of (8) by \( U_X^{-1} \), leading to \( U_X^{-1} x_i = U_X^{-1} x_i + w_i \), where \( w_i = U_X^{-1} e_i \) is white and Gaussian with covariance \( \lambda^{-1} I \). Considering all the vectors in \( X \), which share the same whitening matrix, we have

\[
U_X^{-1} X = U_X^{-1} X + \Omega_X,
\]

where \( \Omega_X \) is white and Gaussian.

With model (9) and the prior \( p(X) \), we use VIAMP to update \( q(X) \). Following VIAMP, a unitary transformation needs to be performed with the unitary matrix \( C_X^T \) obtained from the SVD of \( U_X^{-1} \) (or eigenvalue decomposition (EVD) as the matrix is definite and symmetric), i.e., \( U_X^{-1} = C_X \Lambda C_X^T \) with \( \Lambda \) being a diagonal matrix. After performing the unitary transformation, we have

\[
R_X = \Phi_X X + \Omega_X^{'},
\]

where \( R_X = C_X^T U_X^{-1} X \), \( \Phi_X = C_X^T U_X^{-1} = \Lambda C_X^{-1} \) and \( \Omega_X^{'}, \Lambda X, \Omega X \), which is still Gaussian and white. From the above, the direct way to obtain \( R_X \) and \( \Phi_X \) in model (10) are costly. Instead of computing \( U_X^{-1} \) and \( U_X^{-1} \) followed by SVD of \( U_X^{-1} \), we perform EVD to \( U_X^{-1} = W_X = H^T H + \text{Tr}(U_H)V_H \), i.e., \( [C_X, D_X] = \text{eig}(W_X) \), where the diagonal matrix \( D_X = \Lambda^{-1/2} \). Hence \( \Phi_X = \Lambda C_X^T = D_X^{-1/2} C_X^T \),
where the computation of $D_X^{-1/2}$ is trivial as $D_X$ is a diagonal matrix. Meanwhile, the computation of the pseudo observation matrix $R_X$ can also be simplified:

$$R_X = C_X^T U_X^{-1/2} X = D_X^{-1/2} C_X^T \hat{H}^T Y.$$  \hspace{1cm} (11)

For convenience, we rewrite the unitary transformed pseudo observation model as

$$\frac{D_X^{-1/2} C_X^T \hat{H}^T Y}{\Phi_X} = \frac{D_X^{-1/2} C_X^T X + \Omega_X'.}{\Phi_X}.$$  \hspace{1cm} (12)

The above leads to Lines 1-3 of the UAMP-MF algorithm. Due to the prior $p(X)$, the use of exact $q(X)$ often makes the message update intractable. Following (U)AMP, we perform the minimum mean squared error (MMSE) estimation based on the pseudo observation model (12) with prior $p(X)$, i.e., project it to be Gaussian. These correspond to Lines 4-11 of UAMP-MF. Noting that the prior $p(X)$ is separable, i.e., $p(X) = \prod_{n,l} p(x_{nl})$, the operations in Lines 10 and 11 are element-wise, i.e., the function $G_X(Q_X, V_{Q_X})$ is an element-wise function similar to the one in the AMP algorithm. $(G'_X(Q_X, V_{Q_X})$ denotes its derivative). This is explained as follows. Due to the decoupling of (U)AMP, we assume the following scalar pseudo models

$$q_{nl} = x_{nl} + w_{nl}, n = 1, \ldots, N, l = 1, \ldots, L,$$  \hspace{1cm} (13)

where $q_{nl}$ is the $(n, l)$th element of $Q_X$ in Line 9 of the UAMP-MF algorithm, $w_{nl}$ represents a Gaussian noise with mean zero and variance $v_{nl}$, and $v_{nl}$ is the $(n, l)$th element of $V_{Q_X}$ in Line 8 of the UAMP-MF algorithm. This is significant as the complex estimation is reduced to much simpler MMSE estimation based on a number of scalar models (13) with prior $p(x_{nl})$. With the notations in Lines 10 and 11, for each entry $x_{nl}$ in $X$, the MMSE estimation leads to a Gaussian distribution $\hat{q}(x_{nl}) = N(x_{nl}; \hat{x}_{nl}, v_{nl})$ where $\hat{x}_{nl}$ and $v_{nl}$ is the $(n, l)$th element of $X$ in Line 11 and the $(n, l)$th element of $\Xi_X$ in Line 10, respectively. We can see that each element $x_{nl}$ has its own variance. To facilitate subsequent processing, we make an approximation by performing an average operation to each row of $\Xi_X$, i.e., replacing the entries of each row of $\Xi_X$ by their average. Then $\{\hat{q}(x_{nl})\}$ are collectively characterized by a matrix normal distribution, i.e., $q(X) = \mathcal{MN}(X; \hat{X}, U_X, V_X)$ with $U_X = \text{diag}(\text{mean}(\Xi_X, 2))$ and $V_X = I_L$, where $\text{mean}(\Xi_X, 2)$ represents the average operation on the rows of $\Xi_X$. This leads to Line 12 of UAMP-MF. Due to the symmetry between matrix $H$ and matrix $X$ in the model, the derivations of the message computations for $q(H)$ are similar, which is omitted here due to limited space. See Lines 13-24 of the algorithm.

With $q(H)$ and $q(X)$, we can compute the message from $f_Y$ to $\lambda$, $m_Y \rightarrow \lambda(\lambda) \propto \lambda^{ML} \exp \left( - \lambda C \right)$, where

$$C = \| Y - \hat{H} \hat{X} \|^2 + M \text{Tr}(\hat{X}^T V_H)$$
$$+ L \text{Tr}(U_X H^T H) + M \text{Tr}(U_X V_H).$$  \hspace{1cm} (14)

Then the mean of $\lambda$ is obtained as

$$\hat{\lambda} = \int_\lambda \lambda q(\lambda) = ML/C,$$  \hspace{1cm} (15)

which is Line 25 of the UAMP-MF Algorithm.

**Algorithm 1 UAMP-MF**

**Initialization:** $U_H = I_M, V_H = I_N, H = I_{MN}, V_X = I_L, \Xi_X = 1_{NL}, S_X = 0_{NL}, \Xi_H = 1_{MN}, S_H = 0_{MN}$.

**Repeat**

1: \hspace{0.1cm} $W_X = \hat{H}^T H + M V_H$
2: \hspace{0.1cm} $[C_X, D_X] = \text{eig}(W_X)$
3: \hspace{0.1cm} $R_X = D_X^{-1/2} C_X^T \hat{H}^T Y$, $\Phi_X = D_X^{-1/2} C_X^T$
4: \hspace{0.1cm} $V_{P_X} = |\Phi_X|^2 \Xi_X$
5: \hspace{0.1cm} $P_X = \Phi_X \hat{X} - V_{P_X} \cdot S_X$
6: \hspace{0.1cm} $V_{S_X} = 1 / (V_{P_X} + \lambda^{-1} I)$
7: \hspace{0.1cm} $S_X = V_{S_X} \cdot (R_X - P_X)$
8: \hspace{0.1cm} $V_{Q_X} = 1 / (|\Phi_X|^2 V_{S_X})$
9: \hspace{0.1cm} $Q_X = \hat{X} + V_{Q_X} \cdot (\Phi_H^T S_X)$
10: \hspace{0.1cm} $\Xi_X = V_{Q_X} \cdot G_X(Q_X, V_{Q_X})$
11: \hspace{0.1cm} $\hat{X} = G_X(Q_X, V_{Q_X})$
12: \hspace{0.1cm} $U_X = \text{diag}(\text{mean}(\Xi_X, 2))$
13: \hspace{0.1cm} $W_H = \hat{X}^T X^T + I U_X$
14: \hspace{0.1cm} $[C_H, D_H] = \text{eig}(W_H)$
15: \hspace{0.1cm} $R_H = D_H^{-1/2} C_H^T \hat{H} X^T$, $\Phi_H = D_H^{-1/2} C_H^T$
16: \hspace{0.1cm} $V_{P_H} = |\Phi_H|^2 \Xi_H$
17: \hspace{0.1cm} $P_H = \Phi_H \hat{H} - V_{P_H} \cdot S_H$
18: \hspace{0.1cm} $V_{S_H} = 1 / (V_{P_H} + \lambda^{-1} I)$
19: \hspace{0.1cm} $S_H = V_{S_H} \cdot (R_H - P_H)$
20: \hspace{0.1cm} $V_{Q_H} = 1 / (|\Phi_H|^2 V_{S_H})$
21: \hspace{0.1cm} $Q_H = \hat{H} + V_{Q_H} \cdot (\Phi_H^T S_H)$
22: \hspace{0.1cm} $\Xi_H = V_{Q_H} \cdot G_H(Q_H, V_{Q_H})$
23: \hspace{0.1cm} $\hat{H} = G_H(Q_H, V_{Q_H})$
24: \hspace{0.1cm} $U_H = \text{diag}(\text{mean}(\Xi_H, 1))$
25: \hspace{0.1cm} $\hat{\lambda} = ML/C$ with $C$ given in (14)

Until terminated

**3. APPLICATIONS AND NUMERICAL RESULTS**

**3.1. UAMP-MF for RPCA**

In RPCA [3], we aim to estimate a low-rank matrix observed in the presence of noise and outliers, with the model

$$Y = AB + E + W,$$  \hspace{1cm} (16)

where the product of a tall matrix $A \in \mathbb{R}^{M \times N}$ and wide matrix $B \in \mathbb{R}^{N \times L}$ is a low-rank matrix, and $E$ is a sparse outlier matrix. To enable the use of UAMP-MF, the model...
UAMP-MF can be much faster than BiG-AMP while delivering better performance. IALM, LMaFit and VSBL can be faster than UAMP-MF, but their performance is not comparable to UAMP-MF.

3.2. UAMP-MF for Dictionary Learning

In DL, we aim to find a dictionary $H \in \mathcal{R}^{M \times N}$ that allows the training samples $Y \in \mathcal{R}^{M \times L}$ to be coded as

$$Y = HX + W$$

(20)

for sparse matrix $X$ and some perturbation $W$. For the entries of $H$, we use the Gaussian prior

$$p(H) = \prod_{mn} p(h_{mn}) = \prod_{mn} N(h_{mn}; 0, 1),$$

(21)

and for the entries of $X$, we can again use the sparsity inducing hierarchical Gaussian-Gamma as (19).

We compare UAMP-MF with K-SVD [5], SPAMS [6] and Bad-VAMP [11]. It is noted that, in the case of DL, compared to BiG-AMP, Bad-VAMP is significantly enhanced in terms of robustness and performance [11]. To test the robustness of the algorithms, we generate correlated matrix $H$ using the same way as [19] with a correlation parameter $\rho$. The performance of DL is evaluated using the relative NMSE [11], defined as $\text{NMSE}(H) = \min_{J} \frac{||HJ - H||^2}{||H||^2}$, where $J$ is a matrix accounting for the permutation and scale ambiguities. We show the performance of the algorithms versus $N$ in Fig. 3 (a) and (b), where $M = N = 100$, $L = 600$ and $\text{SNR}=50dB$, and $\rho = 0.1$ in (a) and $\rho = 0.3$ in (b). We can see that K-SVD and SPAMS do not work well. Compared to Bad-VAMP, UAMP-MF performs significantly better, especially when $\rho = 0.3$. The runtime comparison in Fig. 3 (c) shows that UAMP-MF is much faster than other algorithms.

More details and examples such as non-negative MF, CSMU and sparse MF can be found in the extended version of this paper.
4. REFERENCES

[1] R. Rubinstein, A. M. Bruckstein, and M. Elad, “Dictionaries for sparse representation modeling,” Proceedings of the IEEE, vol. 98, no. 6, pp. 1045–1057, June 2010.

[2] H. Zhu, G. Leus, and G. B. Giannakis, “Sparsity-cognizant total least-squares for perturbed compressive sampling,” IEEE Transactions on Signal Processing, vol. 59, no. 5, pp. 2002–2016, May 2011.

[3] Emmanuel J. Candes, Xiaodong Li, Yi Ma, and John Wright, “Robust principal component analysis?,” J. ACM, vol. 58, no. 3, June 2011.

[4] Jason T. Parker, Philip Schniter, and Volkan Cevher, “Bilinear generalized approximate message passing, part i: Derivation,” IEEE Transactions on Signal Processing, vol. 62, no. 22, pp. 5839–5853, 2014.

[5] M. Aharon, M. Elad, and A. Bruckstein, “K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation,” IEEE Transactions on Signal Processing, vol. 54, no. 11, pp. 4311–4322, 2006.

[6] Julien Mairal, Francis Bach, Jean Ponce, and Guillermo Sapiro, “Online learning for matrix factorization and sparse coding,” Journal of Machine Learning Research, vol. 11, pp. 19–60, 2010.

[7] Z. Lin, M. Chen, and Y. Ma, “The augmented Lagrange multiplier method for exact recovery of corrupted low-rank matrices,” http://export.arxiv.org/abs/1009.5055, 2010.

[8] Zaiwen Wen, Wotao Yin, and Yin Zhang, “Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm,” Mathematical Programming Computation, vol. 4, no. 4, pp. 333–361, 2012.

[9] Jun He, Laura Balzano, and John C.S. Lui, “Online robust subspace tracking from partial information,” ArXiv, vol. abs/1109.3827, 2011.

[10] D. L. Donoho, A. Maleki, and A. Montanari, “Message passing algorithms for compressed sensing: I. motivation and construction,” in 2010 IEEE Information Theory Workshop on Information Theory (ITW 2010, Cairo), Jan 2010, pp. 1–5.

[11] S. Sarkar, A. K. Fletcher, S. Rangan, and P. Schniter, “Bilinear recovery using adaptive vector-amp,” IEEE Transactions on Signal Processing, vol. 67, no. 13, pp. 3383–3396, July 2019.

[12] Xiangming Meng and Jiang Zhu, “Bilinear adaptive generalized vector approximate message passing,” IEEE Access, vol. 7, pp. 4807–4815, 2019.

[13] John Winn and Christopher M Bishop, “Variational message passing,” J. Mach. Learn. Res., vol. 6, pp. 661–694, Apr. 2005.

[14] Qinghua Guo and Jiatao Xi, “Approximate message passing with unitary transformation,” arXiv preprint arXiv:1504.04799, Apr. 2015.

[15] Man Luo, Qinghua Guo, Ming Jin, Yonina C. Eldar, Defeng Huang, and Xiangming Meng, “Unitary approximate message passing for sparse bayesian learning,” IEEE Transactions on Signal Processing, vol. 69, pp. 6023–6039, 2021.

[16] Michael E. Tipping, “Sparse bayesian learning and the relevance vector machine,” J. Mach. Learn. Res., vol. 1, pp. 211–244, sep 2001.

[17] Jason T. Parker, Philip Schniter, and Volkan Cevher, “Bilinear generalized approximate message passing Part ii: Applications,” IEEE Transactions on Signal Processing, vol. 62, no. 22, pp. 5854–5867, 2014.

[18] S. D. Babacan, M. Luessi, R. Molina, and A. K. Katsaggelos, “Sparse bayesian methods for low-rank matrix estimation,” IEEE Transactions on Signal Processing, vol. 60, no. 8, 2011.

[19] Zhengdao Yuan, Qinghua Guo, and Man Luo, “Approximate message passing with unitary transformation for robust bilinear recovery,” IEEE Trans. Signal Process., vol. 69, pp. 617–630, Dec. 2020.