

الگوریتم جدید و مقاوم AMP برای ماتریس‌های غیر iid و گوسی مبتنی بر تئوری بیز در نمونه‌برداری فشرده

فهیمة انصاری رام، مرتضی خادمی، عباس ابراهیمی مقدم و هادی صدوقی یزدی

یعنی به جای نمایش آن با یک بردار N تایی، می‌توان آن را با یک بردار M تایی ($M < N$) به نام بردار اندازه‌گیری^۳ $y \in \mathbb{R}^M$ نشان داد [۱] و [۲]. در اینجا y از رابطه زیر به دست می‌آید

$$y = \Phi s + \omega = \Phi \Psi x + \omega = Ax + \omega \quad (۱)$$

که ماتریس $\Phi_{M \times N}$ ماتریس اندازه‌گیری^۴ و ماتریس $A_{M \times N} = \Phi \Psi$ ماتریس نمونه‌برداری^۵ نامیده می‌شود و $\omega \in \mathbb{R}^M$ نویز مشاهده است. این نویز در اکثر مراجع سفید، گوسی و جمع‌شونده با میانگین صفر و واریانس σ_ω^2 فرض می‌شود. رابطه (۱) فرایند نمونه‌برداری یا کدگذاری توسط نمونه‌برداری فشرده را بیان می‌کند. در بخش کد بردار، هدف CS یافتن بردار x با معلوم بودن بردار اندازه‌گیری y و ماتریس نمونه‌برداری A می‌باشد. از آنجا که ماتریس A غیر مربعی و وارون‌ناپذیر است، بازیابی بردار x از y در حالت عادی دارای بی‌نهایت جواب است اما با در نظر گرفتن شرط تنک بودن بردار x ، تنها یک جواب از مجموعه جواب‌های به دست آمده، قابل قبول می‌باشد. بر این اساس از میان مجموعه جواب‌های به دست آمده جوابی صحیح است که تنک‌ترین باشد و نرم صفر بردار x ($\|x\|_0$) را حداقل کند [۱] و [۲]. پس داریم

$$\hat{x} = \arg \min_{x \in \mathbb{R}^N} \|x\|_0 \quad \text{subject to } \|y - Ax\|_2 \leq \xi \quad (۲)$$

به طوری که $\xi \geq 0$ می‌باشد. بنابراین با حل مسئله بهینه‌سازی فوق و تعیین \hat{x} می‌توان سیگنال بازسازی‌شده را توسط عبارت $\hat{s} = \Psi \hat{x}$ به دست آورد. متأسفانه (۲) یک مسئله با پیچیدگی غیر چندجمله‌ای^۶ است و در عمل نمی‌توان از این رویکرد استفاده کرد، لذا روش‌های مختلفی برای حل این مسئله با عملکردهای متفاوت و پیچیدگی‌های مختلف ارائه شده‌اند که چهار دسته آنها پرکاربردترند [۴]. اولین دسته الگوریتم‌های محدب یا حداقل‌سازی l_1 هستند که برای بازیابی، یک مسئله بهینه‌سازی محدب را حل می‌کنند. الگوریتم‌های BP^V [۵]، $BPDN^A$ [۶] و رگرسیون کمترین زاویه $LARS^9$ [۷]، مثال‌هایی از این دسته از الگوریتم‌ها می‌باشند اما این روش‌ها پیچیدگی محاسباتی خیلی بالایی دارند و برای مسایل با ابعاد بالا قابل استفاده نیستند. دسته دوم الگوریتم‌های حریصانه تکراری می‌باشند که نسبت به الگوریتم‌های محدب، هزینه محاسباتی کمتر و سرعت بالاتری دارند اما زمانی که سیگنال، خیلی تنک نباشد، هزینه محاسباتی خیلی افزایش می‌یابد. از جمله الگوریتم‌های حریصانه پرکاربرد،

چکیده: الگوریتم تقریب انتقال پیام (AMP) یک الگوریتم تکراری کم‌هزینه برای بازیابی سیگنال در نمونه‌برداری فشرده است. هنگامی که ماتریس نمونه‌برداری دارای مؤلفه‌هایی با توزیع گوسی مستقل و یکسان (iid) باشد، همگرایی AMP با تحلیل ریاضی اثبات می‌شود. اما برای سایر ماتریس‌های نمونه‌برداری به خصوص ماتریس‌های بدحالت، عملکرد این الگوریتم ضعیف شده و حتی ممکن است واگرا شود. این مشکل منجر به محدودیت استفاده از AMP در بعضی کاربردها از جمله تصویربرداری شده است. در این مقاله الگوریتمی جهت اصلاح AMP مبتنی بر تئوری بیز برای ماتریس‌های غیر iid ارائه شده است. نتایج شبیه‌سازی نشان می‌دهد که میزان مقاومت الگوریتم پیشنهادی برای ماتریس‌های غیر iid نسبت به روش‌های پیشین بیشتر می‌باشد. به عبارت دیگر این روش دارای دقت بیشتر در بازیابی است و با تکرار کمتری همگرا خواهد شد.

کلیدواژه: الگوریتم تقریب انتقال پیام (AMP)، ماتریس‌های بدحالت، ماتریس‌های سطر متعامد، ماتریس‌های گوسی iid، ماتریس‌های مرتبه پایین، نمونه‌برداری فشرده (CS).

۱- مقدمه

تئوری نمونه‌برداری نایکوئیست-شانون در پردازش سیگنال از اهمیت بسیاری برخوردار است. بر اساس این تئوری یک سیگنال با پهنای باند Ω باید با نرخ 2Ω نمونه در ثانیه (نرخ نایکوئیست) نمونه‌برداری شود تا بازیابی بدون خطا و قابل اطمینانی داشته باشد. در سال‌های اخیر تئوری دیگری به نام نمونه‌برداری فشرده^۱ (CS) ارائه شده که نشان می‌دهد می‌توان سیگنال‌ها را با تعداد داده کمتری نسبت به تعداد نمونه‌های لازم در نرخ نایکوئیست بازیابی کرد [۱] و [۲]. این تئوری در تصویربرداری، رادار، مبدل آنالوگ به دیجیتال و سیستم‌های مخابراتی کاربرد بسیار دارد [۳].

در نظریه نمونه‌برداری فشرده ثابت می‌شود که می‌توان سیگنال $s \in \mathbb{R}^N$ را که در حوزه توابع پایه و متعامد $\Psi \in \mathbb{R}^{N \times N}$ ، تنک^۲ است (یعنی $s = \Psi x$ به طوری که بردار $x \in \mathbb{R}^M$ تبدیل شده تنک^۳ s می‌باشد) از روی مجموعه‌ای از ترکیب خطی نمونه‌های s بازیابی کرد،

این مقاله در تاریخ ۱۶ اردیبهشت ماه ۱۳۹۸ دریافت و در تاریخ ۲۳ آذر ماه ۱۳۹۸ بازنگری شد.

فهیمة انصاری رام، گروه مخابرات، دانشکده فنی و مهندسی، دانشگاه فردوسی مشهد، مشهد، ایران، (email: fa.ansari@mail.um.ac.ir).

مرتضی خادمی (نویسنده مسئول)، گروه مخابرات، دانشکده فنی و مهندسی، دانشگاه فردوسی مشهد، مشهد، ایران، (email: khademi@um.ac.ir).

عباس ابراهیمی مقدم، گروه مخابرات، دانشکده فنی و مهندسی، دانشگاه فردوسی مشهد، مشهد، ایران، (email: a.ebrahimi@um.ac.ir).

هادی صدوقی یزدی، گروه کامپیوتر، دانشکده فنی و مهندسی، دانشگاه فردوسی مشهد، مشهد، ایران، (email: h-sadoghi@um.ac.ir).

1. Compressed Sensing

2. Sparse

3. Measurement Vector

4. Measurement Matrix

5. Sampling Matrix

6. NP-Hard

7. Basis Pursuit

8. Basis Pursuit Denoising

9. Least Angle Regression

ماتریس‌ها که ماتریس‌های تغییرناپذیر در چرخش به راست^{۱۸} هستند داشته‌اند. در این مقاله نیز الگوریتم جدیدی برای حل مسئله AMP برای خانواده بزرگ‌تری از ماتریس‌ها از دیدگاه تئوری بیز ارائه شده است. به عبارت دیگر در روش پیشنهادی برای الگوریتم رایج IST در هر تکرار، خطا فرض می‌شود و سپس به کمک معادله مشاهده (۱)، مبتنی بر تئوری بیز و تخمین MAP، این الگوریتم را بهبود می‌دهد. بدین وسیله الگوریتم پیشنهادی برای ماتریس‌های گوسی iid عملکردی بهتر از AMP و برای ماتریس‌های غیر iid (که AMP واگرا می‌شود) عملکردی بهتر از VAMP و OAMP خواهد داشت. این الگوریتم مانند VAMP و OAMP شامل دو بخش می‌باشد. بخش اول، آستانه‌گیری است که در آن از تابع آستانه‌گیر نرم و یا تقریبی از این تابع استفاده شده و تخمین پیشینی از بردار x به دست می‌آورد. بخش دوم (بخش تصحیح) مبتنی بر تئوری بیز بوده و تخمین پیشین را اصلاح می‌کند.

نتایج شبیه‌سازی نشان می‌دهد این الگوریتم در مقایسه با روش‌های پیشین عملکرد بهتری از نظر میانگین حداقل مربعات خطای نرمالیزه^{۱۹} (NMSE) داشته و در تکرار کمتری به جواب نهایی همگرا می‌شود. در ادامه، در بخش ۲ مروری بر الگوریتم‌های تکراری آستانه‌گیر پرکاربرد مانند انواع ساده IHT و IST، الگوریتم AMP و اهمیت آن و همچنین دو الگوریتم VAMP و OAMP که برای بهبود AMP عملکرد خوبی داشته‌اند ارائه می‌شود. بخش ۳ به ارائه الگوریتم پیشنهادی و مبانی ریاضی آن و همچنین نحوه به روز رسانی پارامترهای الگوریتم می‌پردازد. در بخش ۴، نتایج شبیه‌سازی و مقایسه الگوریتم پیشنهادی با سایر الگوریتم‌ها برای انواع مختلف ماتریس‌ها ارائه می‌گردد.

۲- الگوریتم‌های تکراری آستانه‌گیر برای بازیابی CS

در این بخش انواع روش‌های تکراری آستانه‌گیر (IHT، IST، AMP، VAMP و OAMP) توضیح داده می‌شود تا زمینه برای معرفی الگوریتم پیشنهادی فراهم گردد. هدف این روش‌ها تخمین سیگنال تنک با داشتن ماتریس A و بردار y می‌باشد.

۲-۱ الگوریتم تکراری آستانه‌گیر سخت (IHT)

یکی از ساده‌ترین الگوریتم‌های تکراری آستانه‌گیر، الگوریتم IHT^{۲۰} است که اولین بار در [۱۷] توسط رابطه بازگشتی زیر ارائه شده است

$$\hat{x}_{t+1} = \eta_H(\hat{x}_t + A^T(y - A\hat{x}_t); \lambda_t) \quad (۳)$$

که \hat{x}_t تخمین سیگنال تنک و λ_t مقدار آستانه در زمان t می‌باشد. تابع آستانه‌گیر سخت، $\eta_H(r; \lambda)$ ، به صورت مؤلفه به مؤلفه بر روی بردار r عمل می‌کند و به صورت زیر تعریف می‌شود

$$\eta_H(r; \lambda) = \begin{cases} r & |r| > \lambda \\ 0 & |r| \leq \lambda \end{cases}$$

الگوریتم IHT با مقدار اولیه تقریب $\hat{x} = 0$ شروع می‌شود. با گامی در مسیر نزول گرادیان مقدار تقریب اصلاح می‌گردد و مقدار اصلاح‌شده از طریق تابع آستانه‌گیری تنک می‌شود. این مراحل تا زمان رسیدن به معیار همگرایی تکرار می‌گردد. تکرارها در IHT ساده‌ترین روش در میان همه الگوریتم‌های ذکر شده است زیرا تنها عملیاتی که برای هر تکرار لازم دارد

MP^۱ و انواع آن از جمله OMP^۲ [۸] و [۹] و CoSaMP^۳ [۱۰] می‌باشند. می‌باشند. سومین دسته، روش‌های بیزی هستند که در آنها از توزیع پیشین سیگنال x ($p(x)$) و معادله مشاهده (۱) استفاده شده تا تخمین حداکثر توزیع پسین^۴ (MAP) یا حداقل میانگین مربعات خطا^۵ (MMSE) از سیگنال تنک x را به دست آورد [۱۱] تا [۱۵]. در این روش‌ها به کمک تابع توزیع احتمال پسین، تخمین آماری کاملی ارائه می‌شود اما این روش‌ها نیز برای مسایل با ابعاد زیاد به دلیل افزایش پیچیدگی محاسباتی، مناسب نمی‌باشند [۱۶]. دسته چهارم الگوریتم‌های تکراری آستانه‌گیر^۶ (IT) هستند که در بازیابی CS سریع‌تر از دسته اول می‌باشند. الگوریتم‌های IHT^۷ [۱۷] و IST^۸ [۱۸] از این دسته از الگوریتم‌ها هستند.

الگوریتم‌های انتقال پیام^۹، اصلاح‌شده دسته سوم یعنی الگوریتم‌های تکراری هستند. این الگوریتم‌ها از مدل گراف‌ها^{۱۱} (فاکتور گراف) استفاده می‌کنند و در آنها متغیرهای اولیه (پیام‌ها) با شاخه‌های گراف مرتبط هستند. گراف مطرح در CS یک گراف دوطرفه با N گره در یک طرف (گره‌های متغیر) و M گره در طرف دیگر (گره‌های اندازه‌گیری) می‌باشد. این روش مزایای بسیاری از جمله پیچیدگی محاسباتی کم و اجرای آسان به صورت موازی یا توزیع‌شده دارد. در [۱۹] یک روش انتقال پیام برای بازیابی x با داشتن ماتریس A و بردار y به نام تقریب انتقال پیام^{۱۱} (AMP) ارائه شده که از نظر کارایی محاسباتی و تعداد اندازه‌گیری‌های مورد نیاز برای بازسازی دقیق x ، عملکرد خوبی دارد. زمانی که ماتریس تصادفی A شامل متغیرهای تصادفی با میانگین صفر، iid، گوسی یا شبه‌گوسی باشد رفتار دینامیکی AMP می‌تواند با یک رابطه بازگشتی ساده و اسکالر به نام تکامل حالت^{۱۲} (SE) تفسیر شود [۱۹] و [۲۰]. فرض iid بودن عناصر ماتریس A برای رابطه SE در AMP بحرانی است. اگر این ماتریس، iid نباشد (مخصوصاً زمانی که A بد حالت^{۱۳} باشد) دقت SE تضمین نمی‌شود و AMP ممکن است خیلی ضعیف عمل کند یا حتی واگرا شود [۲۱] تا [۲۳]. برای حل این مشکل روش‌هایی از جمله میرایی^{۱۴} [۲۱]، حذف میانگین [۲۳]، به روز رسانی به صورت متوالی^{۱۵} [۲۴] و حداقل‌سازی مستقیم با انرژی آزاد [۲۵] ارائه شده که همه اینها از واگرایی AMP جلوگیری می‌کنند ولی از سرعت همگرایی می‌کاهند. اخیراً دو الگوریتم دیگر^{۱۶} OAMP [۲۶] و VAMP^{۱۷} [۲۷] برای حل این مشکل معرفی شده‌اند که عملکرد نسبتاً خوبی برای خانواده بزرگ‌تری از

1. Matching Pursuit
2. Orthogonal Matching Pursuit
3. Compressive Sampling Matching Pursuit
4. Maximum A Posteriori
5. Minimum Mean Squared Error
6. Iterative Thresholding
7. Iterative Hard Thresholding
8. Iterative Soft Thresholding
9. Message Passing
10. Graphical Model
11. Approximate Message Passing
12. State Evolution
13. III-Conditioned
14. Damping
15. Sequential
16. Orthogonal Approximate Message Passing
17. Vector Approximate Message Passing

18. Right-Rotationally Invariant

19. Normalized Mean-Squared Error

20. Iterative Hard Thresholding

آستانه‌گیر پیوسته-لیپشیتزی^۴ می‌تواند باشد. اگر این تابع، تابع آستانه‌گیر نرم $\eta_S(\cdot; \cdot)$ باشد (رابطه (۴))، الگوریتم $AMP-l_1$ نامیده می‌شود. برای حالتی که مؤلفه‌های x دارای توزیع مستقل $(p(x) = \prod_{j=1}^N p_j(x_j))$ باشند و توزیع پیشین هر مؤلفه $p_j(x_j)$ معلوم باشد تابع آستانه‌گیر بهینه معمولاً به صورت فرم بسته به دست می‌آید. اما زمانی که توزیع پیشین $p_j(x_j)$ معلوم نباشد برای سیگنال‌های تنک x ، تابع آستانه‌گیر نرم، یک انتخاب حداقل‌ساز-حداکثری (minimax) بهینه^۵ می‌باشد [۳۱]. به این معنی که تابع آستانه‌گیر نرم، حداکثر خطا را برای خانواده سیگنال‌های تنک با توزیع پیشین $p_j(x_j) = (1-\rho)\delta(x_j) + \rho\tilde{p}_j(x_j)$ ، $\rho \in (0, 1)$ و هر توزیع مجهول و دلخواه $\tilde{p}_j(x_j)$ حداقل می‌کند.

همان طور که مشاهده می‌شود، $AMP-l_1$ خیلی شبیه به الگوریتم تکراری آستانه‌گیر نرم است و تفاوت اصلی آن با IST در اضافه‌بودن عبارت $b_i v_{i-1}$ برای رابطه v_i (۶-الف) می‌باشد که به این عبارت اصطلاحاً تصحیح آنراگر^۶ می‌گویند. این عبارت از تئوری انتشار باور (BP)^۷ در مدل گراف‌ها به دست آمده که نقش مهمی در عملکرد الگوریتم AMP دارد و نشان داده می‌شود که وجود این عبارت مصالحه بین تنکی و نرخ نمونه‌برداری را به صورت قابل توجهی بهبود می‌دهد [۴]. به طور خلاصه می‌توان گفت الگوریتم AMP یک الگوریتم تکراری است که نسبت به سایر روش‌های پیشین، سریع‌تر می‌باشد و به خوبی روش‌های بهینه‌ساز محذب، عمل می‌کند. این نتایج هم از نظر تجربی و هم به صورت تئوری اثبات شده است [۱۹].

زمانی که A یک ماتریس تصادفی شامل مؤلفه‌های iid گوسی یا شبه‌گوسی با میانگین صفر و با واریانس M^{-1} بوده، نویز مشاهده ω (۱) گوسی و تابع $\eta(\cdot; \cdot)$ لیپشیتز^۸ و جداپذیر^۹ باشد، عبارت تصحیح آنراگر، تکرارهای AMP را از هم جدا می‌کند و AMP رفتار جالبی از خود نشان می‌دهد [۱۹]. به این شکل که ورودی تابع $\eta(\cdot; \cdot)$ که برابر است با

$$r_i = \hat{x}_i + A^T v_i \quad (۸)$$

می‌تواند دقیقاً به صورت $r_i = x_i + \mathcal{N}(0, \sigma_i^2 I)$ با $\sigma_i^2 = (1/M) \|v_i\|_r^2$ مدل شود. به بیانی دیگر عبارت تصحیح آنراگر تضمین می‌کند که ورودی تابع $\eta(\cdot; \cdot)$ یک نسخه نویزی شده با نویز AWGN از سیگنال x با واریانس معلوم σ_i^2 باشد. این ویژگی تحلیل دقیق الگوریتم AMP [۱۹] و [۳۲] و تنظیم پارامترهای بهینه تابع آستانه‌گیر (λ_i) را ممکن می‌سازد [۳۳] و نیز همگرایی خطی \hat{x}_i به جواب نهایی را تضمین می‌کند [۴]. یکی از مهم‌ترین محدودیت‌های الگوریتم AMP این است که همگرایی آن تنها برای ماتریس‌های بزرگ با عناصر iid و شبه‌گوسی A تضمین می‌شود و به شدت به ساختار A وابسته است. برای مثال AMP حتی برای ماتریسی که کمی بدحالت و یا دارای میانگین غیر صفر باشد، واگرا خواهد شد [۲۰] تا [۲۲].

برای حل این مشکل روش‌هایی از جمله روش میرایی [۲۱]، حذف میانگین [۲۳]، به روز رسانی متوالی^{۱۰} [۲۴] و حداقل‌سازی مستقیم با انرژی آزاد^{۱۱} [۲۵] ارائه شده تا احتمال واگرایی AMP را کاهش دهند.

ضرب یک بردار در ماتریس A یا A^T است. ثابت می‌شود که این الگوریتم به سمت جواب درست تحت شرایط خاص همگرا می‌شود [۴]. البته نتایج حاصل از این الگوریتم در عمل ضعیف‌تر از نتایج روش‌های محذب می‌باشد. به عبارت دیگر این الگوریتم در مصالحه بین تنکی و نرخ نمونه‌برداری^۱ خیلی بدتر از حداقل‌سازی l_1 عمل می‌کند [۴].

۲-۲ الگوریتم تکراری آستانه‌گیر نرم (IST)

یک الگوریتم IT دیگر که نقش مهمی در بحث‌های ما دارد الگوریتم تکراری آستانه‌گیر نرم^۲ (IST) [۱۸] می‌باشد. فرض کنید که تابع $\eta_S(\cdot; \lambda): \mathbb{R}^N \rightarrow \mathbb{R}^N$ تابع آستانه‌گیر نرم است که مؤلفه‌ای عمل می‌کند و به صورت زیر تعریف می‌شود

$$[\eta_S(r; \lambda)]_j = \text{sgn}(r_j) \max\{|r_j| - \lambda, 0\} \quad (۴)$$

که r_j مؤلفه j ام بردار r می‌باشد. آن گاه هر تکرار الگوریتم IST به روش زیر صورت می‌گیرد

$$v_i = y - A\hat{x}_i \quad (۵-الف)$$

$$\hat{x}_{i+1} = \eta_S(\hat{x}_i + \beta A^T v_i; \lambda) \quad (۵-ب)$$

که β طول گام، v_i باقیمانده خطای مشاهدات در تکرار t ام و λ مقدار آستانه می‌باشند. یکی از چالش‌های مهم در الگوریتم IST و IHT که بر عملکرد آنها مؤثر است تنظیم پارامتر آستانه است و در [۲۸] دو روش برای انتخاب این پارامتر ارائه شده است. تنها تفاوت الگوریتم IST با IHT در استفاده از تابع آستانه‌گیر نرم است. هرچند همگرایی الگوریتم IST تحت شرط $\beta \in (0, 1/\|A\|_r)$ اثبات می‌شود [۲۹] اما گاهی اوقات این الگوریتم با سرعت پایینی همگرا می‌شود و در نتیجه روش‌هایی برای اصلاح و افزایش سرعت آن ارائه شده است. مشهورترین این روش‌ها، الگوریتم تکراری آستانه‌گیر سریع و نرم^۳ (FIST) [۳۰] که سریع‌تر از IST همگرا می‌شود.

۳-۲ الگوریتم AMP

الگوریتم AMP که اولین بار در [۱۹] برای بازیابی CS ارائه شد به صورت زیر بیان می‌شود

$$v_i = y - A\hat{x}_i + b_i v_{i-1} \quad (۶-الف)$$

$$r_i = \hat{x}_i + A^T v_i \quad (۶-ب)$$

$$\hat{x}_{i+1} = \eta(r_i; \lambda_i) \quad (۶-ج)$$

که در این رابطه

$$\hat{x}_i = 0 \quad (۷-الف)$$

$$v_{i-1} = 0$$

$$b_{i+1} \leq \eta'(r_i, \lambda_i) \geq \frac{1}{M} \sum_{j=1}^N \left. \frac{\partial \eta(r; \lambda_j)}{\partial r_j} \right|_{r=\hat{x}_i + A^T v_i} \quad (۷-ب)$$

و $\hat{x}_i \in \mathbb{R}^N$ تخمین کنونی از مقدار واقعی سیگنال، x ، $v_i \in \mathbb{R}^M$ باقیمانده کنونی و r_i ورودی تابع آستانه‌گیر است و همچنین تابع آستانه‌گیر $\eta(\cdot; \cdot)$ (که به صورت عنصر به عنصر عمل می‌کند) هر تابع

4. Lipschitz-Continuous
5. Minimax Optimal
6. Onsager Correction
7. Belief Propagation
8. Lipschitz
9. Separable
10. Sequential Updating
11. Direct Free-Energy Minimization

1. Sparsity-Measurement Tradeoff
2. Iterative Soft Thresholding
3. Fast Iterative Soft Thresholding

دیورژانس^۲ صدق کند و یا به عبارتی تابع $\eta_t(\cdot; \cdot)$ به گونه‌ای انتخاب شود شود که $E\{\eta_t(\cdot; \cdot)\} = 0$ باشد. در نتیجه عبارت آنراگر در باقیمانده حذف می‌شود. نشان داده شده که استفاده از این دو شرط سبب می‌شود که خطا به صورت استاتیکی متعادل شود و به همین دلیل آن را AMP متعادل یا به اختصار OAMP می‌نامند [۲۵]. در [۲۵] ماتریس W_t را به سه صورت در نظر گرفته که بهترین نتایج مربوط به ماتریس W_t^{LMMSE} بوده است که به صورت زیر تعریف می‌شود

$$W_t^{LMMSE} = \gamma_t^* A^T (\gamma_t^* A A^T + \sigma_w^2 I)^{-1} \quad (۱۰-الف)$$

$$\gamma_t^* = \frac{\|y - A\hat{x}_t\|^2 - M\sigma_w^2}{tr(A^T A)} \quad (۱۰-ب)$$

همچنین نتایج شبیه‌سازی نشان می‌دهد که این الگوریتم برای ماتریس نمونه‌بردار بخشی متعادل و یا بدحالت که در آن AMP خوب عمل نمی‌کند، عملکرد خوبی دارد [۲۶].

۲-۵ الگوریتم VAMP

الگوریتم دیگری که اخیراً برای رفع محدودیت AMP در برابر گوسی بودن ماتریس A معرفی شده است، الگوریتم VAMP است [۲۷]. این الگوریتم دارای تمام خواص مطلوب AMP می‌باشد یعنی پیچیدگی کم در هر تکرار، نیاز به تکرار خیلی کم برای همگرایی و تبعیت ورودی تابع آستانه‌گیر از مدل AWGN $(r_t = x_t + \mathcal{N}(0, \sigma_t^2 I))$. همچنین این الگوریتم برای خانواده بزرگ‌تری از ماتریس‌ها که ماتریس‌های تغییرناپذیر در چرخش به راست هستند به خوبی عمل می‌کند. جزئیات الگوریتم VAMP در شکل ۱ آمده است. این الگوریتم شامل دو بخش رفع نویز و تخمینگر LMMSE می‌باشد که هر بخش شامل چهار گام مشابه است: (۱) تخمین (خطوط ۴ و ۱۱)، (۲) محاسبه دیورژانس (خطوط ۵ و ۱۲)، (۳) تصحیح آنراگر (خطوط ۷ و ۱۴) و (۴) محاسبه واریانس (خطوط ۶ و ۱۳). تنها تفاوت بین دو بخش انتخاب تخمینگر در گام اول هر بخش است که بخش اول شامل یک تابع غیر خطی عنصر به عنصر (خط ۴) جهت رفع نویز و بخش دوم شامل یک تخمینگر LMMSE (خط ۱۱) می‌باشد. این الگوریتم خیلی شبیه به الگوریتم OAMP با W_t^{LMMSE} است و در هر دو از تخمینگر LMMSE استفاده شده است.

هر چند دو الگوریتم VAMP و OAMP از دو دیدگاه مختلف به حل مسئله پرداخته‌اند اما نتیجه هر دو مشابه هم شده است. تنها تفاوت این دو الگوریتم در تقریب عبارت واریانس است که می‌توان این اختلاف را در (۳۰) و (۳۱) در [۲۶] و خطوط ۷ و ۱۴ از الگوریتم ۲ در [۲۷] با هم مقایسه کرد.

شکل ۲ که بر اساس شبیه‌سازی الگوریتم‌های پنج‌گانه ذکر شده به دست آمده است، میانگین حداقل مربعات خطای نرمالیزه (یا به اختصار NMSE) بر حسب تکرار، برای الگوریتم‌های مختلف در $\text{SNR} = 40 \text{ dB}$ را نشان می‌دهد به طوری که $\text{NMSE} = \frac{\|\hat{x}_t - x^*\|_2}{\|x^*\|_2}$ و از ۱۰۰۰ تحقق مستقل تصادفی (ω, A, x^*) میانگین‌گیری شده است. در شکل ۲-الف نتایج الگوریتم‌های IST، FIST، AMP، VAMP برای ماتریس نمونه‌بردار گوسی و iid با هم مقایسه شده‌اند.

همان طور که در این شکل مشاهده می‌شود، الگوریتم‌های IST، AMP، FIST، VAMP و OAMP به ترتیب پس از ۴۰، ۲۵۰، ۵۰۰۰ و ۲۰ تکرار به جواب نهایی مشابه همگرا می‌شوند و به عبارتی

این روش‌ها در میزان مؤثر بودن محدودند به این معنی که در این

```

1: Select initial  $r_{10}$  and  $\gamma_{10} \geq 0$ .
2: for  $t = 0, 1, \dots, T_{\text{max}}$  do
3:   //Denosing
4:    $\hat{x}_t = \eta(r_t; 1/\gamma_t)$ 
5:    $\alpha_t = \langle \eta'(r_t; 1/\gamma_t) \rangle$ 
6:    $\eta_t = \gamma_t / \alpha_t$ 
7:    $\gamma_{2t} = \eta_t - \gamma_t$ 
8:    $r_{2t} = (\eta_{2t} \hat{x}_{2t} - \gamma_{2t} r_t) / \gamma_{2t}$ 
9:
10:  //LMMSE estimation
11:   $\hat{x}_{2t} = g_2(r_{2t}, \gamma_{2t})$ 
12:   $= (\gamma_{2t} A^T A + \gamma_{2t} I)^{-1} (\gamma_{2t} A^T y + \gamma_{2t} r_{2t})$ 
13:   $\alpha_{2t} = \langle g'(r_{2t}; \gamma_{2t}) \rangle$ 
14:   $\eta_{2t} = \gamma_{2t} / \alpha_{2t}$ 
15:   $\gamma_{1t+1} = \eta_{2t} - \gamma_{2t}$ 
16:   $r_{1t+1} = (\eta_{1t} \hat{x}_{1t} - \gamma_{1t} r_{2t}) / \gamma_{1t+1}$ 
17: End for
18: Return  $\hat{x}_{2T}$ 

```

شکل ۱: الگوریتم VAMP [۲۷].

الگوریتم‌ها سرعت همگرایی کاهش می‌یابد. همچنین این روش‌ها برای ماتریس‌هایی که میزان انحرافشان از ماتریس گوسی و iid زیاد باشد واگرا می‌شوند. اخیراً دو الگوریتم دیگر OAMP [۲۶] و VAMP [۲۷] برای حل این مشکل معرفی شده‌اند که عملکرد نسبتاً خوبی برای خانواده بزرگ‌تری از ماتریس‌ها که ماتریس‌های تغییرناپذیر در چرخش به راست (RRI) هستند، داشته‌اند. یک ماتریس RRI یک ماتریس تصادفی است که توزیع آن بعد از ضرب از راست با هر ماتریس متعادل ثابت، تغییر نمی‌کند. در ادامه توضیح مختصری برای این دو الگوریتم ارائه می‌شود.

۲-۴ الگوریتم OAMP

الگوریتم OAMP برای حل محدودیت AMP در برابر iid و گوسی بودن ماتریس A ارائه شده و نشان می‌دهد که این الگوریتم برای خانواده وسیع‌تری از ماتریس‌ها همگرا می‌شود [۲۶]. روابط به روز رسانی در این الگوریتم به صورت زیر می‌باشد

$$v_t = y - A\hat{x}_t \quad (۹-الف)$$

$$r_t = \hat{x}_t + W_t(v_t - A\hat{x}_t) \quad (۹-ب)$$

$$\hat{x}_{t+1} = \eta_t(r_t; \lambda_t) \quad (۹-ج)$$

همان طور که مشاهده می‌شود این الگوریتم خیلی شبیه به الگوریتم AMP است، فقط در (۹-الف) به جای A^T از ماتریس W_t استفاده شده است که این ماتریس باید دارای شرط ناهمبستگی^۱، یعنی $tr(I - W_t A) = 0$ باشد. همچنین تابع $\eta_t(\cdot; \cdot)$ باید در شرط بدون

$$e_t^- = x_t - \hat{x}_t^- \quad (۱۱)$$

```

1: Select initial  $\hat{x}_0, P_0, Q_0$ 
2: for  $t=0, 1, \dots, T$  do
3:   // Thresholding
4:    $v_t = y - A\hat{x}_{t-1}$ 
5:    $\hat{x}_t^- = \eta(\hat{x}_{t-1} + A^T v_t; \lambda_{2t})$ 
6:    $J_\eta = \text{diag}(\frac{\partial \eta(\cdot)}{\partial r})(I - A^T A)$ 
7:    $P_t^- = J_\eta(\hat{x}_{t-1})P_{t-1}J_\eta^T(\hat{x}_{t-1}) + Q_{t-1}$ 
8:
9:   //Correction
10:   $R = \sigma_w^2 I$ 
11:   $W_t^- = P_t^- A^T (A P_t^- A^T + R)^{-1}$ 
12:   $v_t^- = (v_t - A\hat{x}_t^-)$ 
13:   $\hat{x}_t = \hat{x}_t^- + W_t^- v_t^-$ 
14:   $P_t = (I - W_t^- A) P_t^-$ 
15:   $Q_t = \alpha Q_{t-1} + (1-\alpha)(W_t^- v_t^-)(W_t^- v_t^-)^T$ 
16: End for
17: Return  $\hat{x}_T$ 

```

شکل ۳: الگوریتم پیشنهادی KAMP.

کوواریانس خطای مورد انتظار (P_t^-) که کوواریانس خطای تخمین پیشین نامیده می‌شود، به عنوان امید ریاضی حاصل ضرب خطای تخمین حالت در خودش به صورت زیر بیان می‌شود

$$P_t^- = E[e_t^- e_t^{-T}] \quad (۱۲)$$

در مرحله بعد که مرحله تصحیح یا به روز رسانی است، تخمین پیشین به کمک مشاهدات، ترکیب شده و یک تخمین اصلاح‌شده ارائه می‌شود (خطوط ۱۱ تا ۱۳ از الگوریتم KAMP). سپس کوواریانس خطای پسین (P_t) به صورت زیر تعریف می‌شود (خط ۱۴ از الگوریتم KAMP (شکل ۳))

$$P_t = E\{(x_t - \hat{x}_t)(x_t - \hat{x}_t)^T\} \quad (۱۳)$$

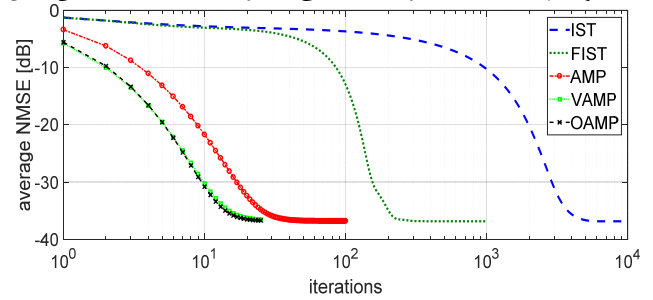
خطوط ۱۰ و ۱۵ از الگوریتم KAMP به ترتیب به روز رسانی ماتریس کوواریانس نوین مشاهده (R) و حالت (Q) هستند.

۳-۱ مبانی ریاضی KAMP

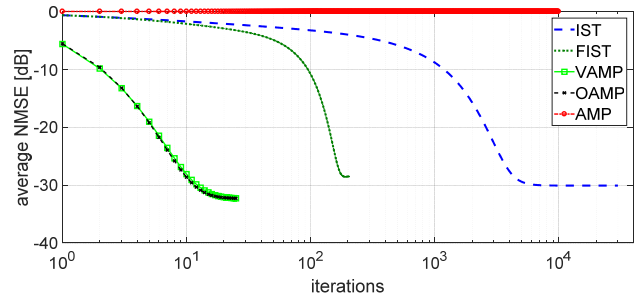
باقیمانده الگوریتم‌های VAMP و OAMP در بخش آستانه‌گیر، خیلی شبیه به الگوریتم IST است و بر خلاف الگوریتم AMP شامل عبارت آنزاگر نیست. در این دو الگوریتم جایگزینی عبارت آنزاگر با بخش LMMSE باعث ارتقای عملکرد در مقایسه با الگوریتم AMP شده است. در این تحقیق، فرض شده الگوریتم رایج IST در هر تکرار دارای خطا است. روش پیشنهادی ارائه‌شده مبتنی بر تئوری بیز، این الگوریتم را بهبود می‌دهد تا برای ماتریس‌های گوسی iid با تکرار کمتر عملکردی بهتر از AMP و برای ماتریس‌های غیر iid، (که AMP واگرا می‌شود) عملکردی بهتر از VAMP و OAMP داشته باشد.

در این تحقیق، ابتدا تابع $g(\cdot; \cdot)$ به صورت زیر تعریف می‌شود

الگوریتم FIST نسبت به IST با تکرار خیلی کمتری به جواب نهایی می‌رسد. در حالی که الگوریتم AMP برای رسیدن به جواب نهایی نسبت به الگوریتم FIST به تکرارهای خیلی کمتری نیاز دارد. در عین حال



(الف)



(ب)

شکل ۲: میانگین NMSE بر حسب تکرار الگوریتم‌های AMP، VAMP، FIST، IST برای SNR = ۴۰ dB (الف) ماتریس A گوسی و iid و (ب) ماتریس A بدحالت با عدد حالت $k(A) = ۱۵$.

مشاهده می‌شود که الگوریتم VAMP تقریباً با نیمی از تکرارهای AMP به جواب نهایی همگرا می‌شود. در شکل ۲-ب که در آن ماتریس A کمی بدحالت با عدد حالت $k(A) = ۱۵$ است، الگوریتم AMP واگرا می‌شود (برای وضوح بهتر، NMSE بیشتر از صفر برش خورده است). اما همان طور که دیده می‌شود، الگوریتم VAMP نیاز به تکرار خیلی کمتری نسبت به FIST برای رسیدن به جواب نهایی دارد و الگوریتم FIST نیز نسبت به IST با تکرار خیلی کمتری همگرا می‌شود (نحوه تولید ماتریس بدحالت با عدد حالت مشخص در بخش ۴-۲ توضیح داده شده است).

۳- الگوریتم پیشنهادی

در این بخش الگوریتم جدیدی برای بهبود AMP با ماتریس‌های غیر iid ارائه شده است. این الگوریتم مانند VAMP و OAMP شامل دو بخش می‌باشد. بخش اول، آستانه‌گیری است که در آن از تابع آستانه‌گیر نرم و یا تقریبی از این تابع استفاده شده و بخش دوم یا بخش تصحیح که مبتنی بر تئوری بیز می‌باشد. شکل ۳، الگوریتم پیشنهادی می‌باشد که KAMP نام‌گذاری شده است.

بر طبق این الگوریتم، ابتدا حالات اولیه سیستم و کوواریانس خطای تخمین، مقادری اولیه می‌شوند و سپس الگوریتم وارد یک حلقه تکرار می‌گردد. در هر تکرار (مثلاً تکرار t) ابتدا با استفاده از تابع آستانه‌گیر نرم و یا تقریبی از آن، بر اساس تخمین پسین تکرار قبل (\hat{x}_{t-1})، تخمین پیشینی از x_t زده می‌شود که با \hat{x}_t^- نشان داده می‌شود (خط ۴ و ۵). سپس ماتریس ژاکوبین تابع آستانه‌گیر $\eta(\cdot; \cdot)$ تشکیل می‌شود (خط ۶). خطای تخمین \hat{x}_t^- که به عنوان اختلاف حالت واقعی و تخمین پیشین بیان می‌شود به صورت زیر تعریف می‌گردد

که J_g ژاکوبین تابع $g(\cdot; \cdot)$ است. اثبات این قضیه در پیوست الف آمده است. این قضیه خطوط ۵ و ۸ الگوریتم پیشنهادی را نتیجه می‌دهد. **قضیه دوم:** با فرض این که توزیع پسین در زمان t نیز گوسی است، میانگین و کوواریانس آن به صورت زیر به دست می‌آید

$$P(x_t | y_{:t}) = \mathcal{N}(x_t; \hat{x}_t, P_t) \quad (25)$$

$$\hat{x}_t = \hat{x}_{t-1} + G_t(y_t - A\hat{x}_{t-1}) \quad (26)$$

$$P_t = (I - G_t A) P_{t-1} \quad (27)$$

به طوری که

$$G_t = P_t^- A^T (A P_t^- A^T + R)^{-1} \quad (28)$$

اثبات روابط بالا در پیوست ب آورده شده و این قضیه هم خطوط ۹ تا ۱۲ الگوریتم KAMP را منجر می‌شود.

طبق (۲۸)، الگوریتم پیشنهادی همانند الگوریتم OAMP در هر تکرار نیاز به محاسبه معکوس ماتریس دارد. بنابراین پیچیدگی‌اش به مانند OAMP، در هر تکرار نسبت به AMP افزایش می‌یابد. البته روش‌هایی تکراری مانند روش CG^۱ [۳۴] برای محاسبه معکوس ماتریس و کاهش پیچیدگی آن وجود دارد که در تحقیقاتی مانند [۲۵] از آن استفاده شده و می‌توان در الگوریتم پیشنهادی نیز از آن استفاده کرد؛ اما این کار، هدف این مقاله نبوده و در تحقیقات بعدی مد نظر خواهد بود.

۳-۲ نحوه به روز رسانی ماتریس Q و R

در روش ارائه‌شده نویز سیستم و نویز مشاهده، با توزیع نرمال و میانگین صفر فرض می‌شود. برای کارکرد مناسب این الگوریتم باید از ماتریس کوواریانس نویز سیستم Q و ماتریس کوواریانس نویز مشاهده R اطلاع داشت. ماتریس R عکس میزان دقت اندازه‌گیری را نشان می‌دهد و افزایش این ماتریس به این معنی است که اعتماد به داده‌های اندازه‌گیری کم است و تأثیر داده‌های اندازه‌گیری بر به روز کردن حالت‌ها کمتر می‌شود. ماتریس Q عکس میزان دقت مدل سیستم را نشان می‌دهد و افزایش این ماتریس به این معنی است که اعتماد به مدل سیستم کم است و تأثیر داده‌های اندازه‌گیری بر به روز کردن حالت‌ها بیشتر می‌شود. به علت ماهیت نویز در حالت کلی روشی برای تعیین کوواریانس نویزها وجود ندارد. در صورتی که کوواریانس‌های R و Q صحیح باشد، در الگوریتم ارائه‌شده، ماتریس G_t طبق (۲۸) به صورت بهینه محاسبه می‌شود. هرچه R و Q از مقدار حقیقی خودش فاصله داشته باشد، ماتریس G_t نامناسب‌تر است که این مسئله فرایند تخمین حالت در (۲۶) را با مشکل مواجه می‌کند. برای مقابله با این مشکل، هم‌زمان با تخمین حالت، کوواریانس‌های R و Q به صورت روابط بازگشتی با منطق خاصی اصلاح می‌شوند تا خطای تخمین حالت کاهش یابد. در صورتی که واریانس نویز (σ_{ω}^2) معلوم باشد ماتریس کوواریانس نویز مشاهده $R = \sigma_{\omega}^2 I$ در نظر گرفته می‌شود و در غیر این صورت واریانس نویز به کمک روش ارائه‌شده در [۳۵] تخمین زده می‌شود. برای تخمین ماتریس Q نیز از روش ارائه‌شده در [۳۶] استفاده می‌گردد.

۴- نتایج شبیه‌سازی

الگوریتم پیشنهادی (KAMP) برای کاهش محدودیت AMP در برابر ماتریس‌های غیر iid پیشنهاد شده است. برای ارزیابی عددی، این

$$g(x; \lambda) \triangleq \eta(x + A^T(y - Ax); \lambda) \quad (14)$$

تابع $g(\cdot; \cdot)$ علاوه بر دو پارامتر x و λ به دو پارامتر y و A نیز وابسته است اما چون این دو پارامتر از معلومات مسئله هستند لذا برای سادگی به جای نوشتن $g(x, A, y; \lambda)$ ، دو پارامتر y و A حذف و $g(x; \lambda)$ نوشته می‌شود. الگوریتم KAMP بر این فرض استوار است که \hat{x}_t به دست آمده در الگوریتم IST (ب-۵) در هر تکرار شامل خطای ε_t می‌باشد و این رابطه به صورت (۱۵-الف) بازنویسی می‌شود. از طرف دیگر بردار مشاهده در هر تکرار به صورت (۱۵-ب) تعریف می‌شود

$$x_t = g(x_{t-1}; \lambda) + \varepsilon_t \quad (15-الف)$$

$$y_t = Ax_t + \omega_t \quad (15-ب)$$

که در آن $x_t \in \mathbb{R}^N$ ، $\omega_t \in \mathbb{R}^M$ ، $y_t \in \mathbb{R}^M$ ، $\varepsilon_t \in \mathbb{R}^N$ به ترتیب بردارهای حالت یا سیستم، مشاهده یا اندازه، نویز مشاهده و نویز سیستم در زمان t هستند. رابطه (۱۵-الف) معادله حالت سیستم و یا مدل سیستم و (۱۵-ب) معادله مشاهده نامیده می‌شود. هدف، یافتن جواب بهینه مبتنی بر رویکرد بیز به کمک دو رابطه (۱۵) می‌باشد. ابتدا تخمین پیشین، \hat{x}_t^- و پسین، \hat{x}_t ، به صورت زیر تعریف می‌شود

$$\hat{x}_t^- \triangleq E[x_t | y_{:t-1}] \quad (16)$$

$$\hat{x}_t \triangleq E[x_t | y_{:t}] \quad (17)$$

بر اساس تئوری بیز داریم

$$P(x_t | y_{:t}) = \frac{P(y_t | x_t) P(x_t | y_{:t-1})}{P(y_t | y_{:t-1})} \quad (18)$$

$$\hat{x}_t = E[x_t | y_{:t}] = \int x_t P(x_t | y_{:t}) dx_t \quad (19)$$

برای حل رابطه بالا چند فرض اولیه در نظر گرفته می‌شود:

۱) توزیع پیشین $P(x_t | y_{:t-1})$ و توزیع پسین $P(x_t | y_{:t})$ با تقریب گوسی در نظر گرفته شده باشند (در الگوریتم VAMP نیز برای پیام‌ها که همان توزیع پسین و پیشین می‌باشد این فرض در نظر گرفته شده است) و میانگین و کوواریانس آنها برابر است با

$$\hat{x}_t = E[x_t | y_{:t}] \quad (20)$$

$$P_t \triangleq \text{cov}[x_t | y_{:t}]$$

$$\hat{x}_t^- = E[x_t | y_{:t-1}] \quad (21)$$

$$P_t^- \triangleq \text{cov}[x_t | y_{:t-1}]$$

۲) نویز ω_t و ε_t نویزهای سفید، گوسی، عمود بر هم و مستقل از x_t و به ترتیب دارای ماتریس‌های کوواریانس $R_t \triangleq E[\omega_t \omega_t^T] > 0$ و $Q_t \triangleq E[\varepsilon_t \varepsilon_t^T] > 0$ می‌باشند.

برای اثبات فیلتر کالمن از دیدگاه تئوری بیز نیز دو فرض بالا در نظر گرفته می‌شود و به همین دلیل الگوریتم پیشنهادی، KAMP نام‌گذاری شده است.

قضیه اول: طبق فرض بالا، توزیع پیشین گوسی است و میانگین و واریانس آن به صورت زیر به دست می‌آید

$$P(x_t | y_{:t-1}) = \mathcal{N}(x_t; \hat{x}_t^-, P_t^-) \quad (22)$$

$$\hat{x}_t^- \approx g(\hat{x}_{t-1}) \quad (23)$$

$$P_t^- = J_g(\hat{x}_{t-1}) P_{t-1} J_g^T(\hat{x}_{t-1}) + Q_{t-1} \quad (24)$$

الگوریتم به کمک نرم‌افزار MATLAB شبیه‌سازی شده است. در تمام شبیه‌سازی‌ها، سیگنال تنک x به صورت تصادفی با توزیع برنولی-جدول ۱: پارامترهای شبیه‌سازی.

پارامترهایی که در هر تکرار به روز می‌شود	پارامترهای معلوم مسئله
تخمین نهایی در تکرار t ام \hat{x}_t	طول بردار مشاهده y
تخمین پیشین \hat{x}_{t-1}	طول سیگنال x و s
کواریانس خطای پیشین P_{t-1}	احتمال وجود مؤلفه‌های غیر صفر در سیگنال x
کواریانس خطای پسین P_t	نسبت سیگنال به نویز
ماتریس کواریانس نویز مشاهده R_t	تعداد تکرار هر الگوریتم
ماتریس کواریانس نویز حالت Q_t	مقدار اولیه عناصر قطر اصلی ماتریس Q
	عدد حالت ماتریس A
	میانگین ماتریس گوسی و iid ماتریس A
	رتبه ماتریس A
	بردار مشاهده y
	ماتریس نمونه‌برداری A

جدول ۲: متوسط NMSE (بر حسب دسی‌بل) در SNRهای مختلف برای ماتریس گوسی و iid. نتایج برای الگوریتم‌های OAMP، VAMP و KAMP بعد از ۱۰۰ تکرار و برای AMP بعد از ۱۰۰۰ تکرار به دست آمده است.

SNR (dB)	AMP	VAMP	OAMP	KAMP
۵	-۴٫۱۰	-۴٫۱۴	-۴٫۴۱۸	-۵٫۰۸۸
۱۰	-۸٫۴۰۵	-۸٫۴۹۵	-۸٫۶۴	-۹٫۳۶
۲۰	-۱۷٫۰۲	-۱۷٫۱۵	-۱۷٫۲۹	-۱۹٫۶۹
۴۰	-۳۶٫۷۵	-۳۶٫۵۸	-۳۶٫۶۹	-۴۰٫۹

صفر، ماتریس سطری متعامد و همچنین ماتریس با مرتبه پایین^۱ که در [۲۱] و [۲۷] این ماتریس‌ها بررسی شده است، در نظر گرفته می‌شود.^۲ در جدول ۱ فهرستی از تمام پارامترهایی که قبل از اجرای الگوریتم باید معلوم باشند و همچنین پارامترهایی که در هر تکرار الگوریتم به روز می‌شوند ارائه شده است.

۴-۱ نتایج برای ماتریس گوسی و iid

شکل ۴ الگوریتم پیشنهادی را با سایر الگوریتم‌ها زمانی که ماتریس A ، گوسی iid با مؤلفه‌های $A_{i,j} \sim \mathcal{N}(0, 1/M)$ و مستقل از هم باشند با هم مقایسه می‌کند. این نتایج حاصل از میانگین‌گیری NMSE برای ۱۰۰۰ تحقق مستقل $\{x, A, \omega\}$ که تصادفی تولید شده‌اند، می‌باشد. همان طور که در شکل ۴ مشاهده می‌شود، سه الگوریتم AMP، VAMP و OAMP به NMSE نسبتاً یکسانی همگرا می‌شوند اما الگوریتم KAMP با NMSE کمتری (حدوداً ۴ dB کمتر) نسبت به سه الگوریتم دیگر به جواب نهایی همگرا می‌شود. از طرفی شیب کاهش NMSE در KAMP بیش از سایر الگوریتم‌هاست و به عبارتی دیگر با تعداد تکرار کمتری به جواب نهایی همگرا می‌شود. به عنوان مثال الگوریتم‌های AMP، VAMP و OAMP به ترتیب بعد از ۲۲، ۲۲ و ۳۸ تکرار به NMSE کمتر از -۳۶٫۵ dB دست می‌یابند در حالی که الگوریتم KAMP تنها با ۷ تکرار (یعنی کمتر از یک‌پنجم تکرارهای AMP و یک‌سوم تکرارهای VAMP و OAMP) به NMSE کمتر از -۳۷ dB می‌رسد. بنابراین برای ماتریس‌های گوسی و iid الگوریتم KAMP در هر تکرار نسبت به سه الگوریتم دیگر دارای عملکرد بهتری است. از طرف دیگر، همان طور که قبلاً گفته شد عملکرد VAMP با OAMP خیلی شبیه است.

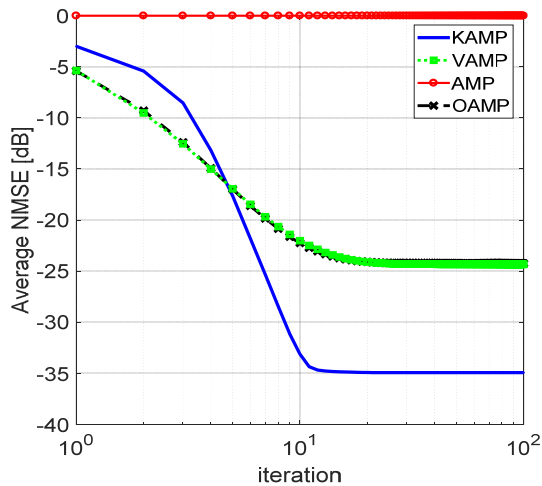
جهت مقایسه الگوریتم‌ها در SNRهای مختلف، آزمایش‌های مشابهی برای SNRهای متفاوت انجام شده و نتایج NMSE در هر SNR برای الگوریتم‌های مختلف در جدول ۲ ارائه شده است. همان طور که مشاهده می‌شود، الگوریتم پیشنهادی در همه SNRها دارای NMSE کمتری نسبت به سایر الگوریتم‌هاست.

گوسی ساخته می‌شود به طوری که احتمال وجود مؤلفه‌های غیر صفر (ρ) باشد. این مؤلفه‌ها به صورت iid با توزیع $\mathcal{N}(0, 1)$ تولید می‌شوند و سایر مؤلفه‌ها صفر می‌باشند. همچنین فرض می‌شود که توزیع x مجهول است و بنابراین در همه الگوریتم‌ها از تابع آستانه‌گیر نرم (ST) که minimax بهینه است استفاده می‌شود. از طرف دیگر نویز مشاهده به صورت تصادفی و گوسی با مؤلفه‌های $\mathcal{N}(0, \sigma_\omega^2)$ ایجاد شده و σ_ω^2 به گونه‌ای انتخاب می‌شود که نسبت سیگنال به نویز (σ_x^2) واریانس نویز مشاهده (σ_ω^2) معلوم فرض شده است. از طرفی نتایج آزمایش نشان می‌دهد که الگوریتم KAMP بهترین عملکرد را برای $R = \sigma_\omega^2 I$ نداشته و به صورت تجربی دریافتیم که برای $R = \sigma_\omega^2 I$ (به طوری که $\hat{\sigma}_\omega^2 = 3\sigma_\omega^2$) نتایج بهتری خواهد داشت. طبق خط ۱ از الگوریتم KAMP (شکل ۳)، متغیرهای Q ، P و \hat{x} باید مقداردهی اولیه شوند. در الگوریتم پیشنهادی برای \hat{x} مانند سایر الگوریتم‌ها بردار صفر در نظر گرفته می‌شود. همچنین نتایج تمام آزمایش‌ها نشان داد که انتخاب مقدار اولیه P تأثیری در عملکرد KAMP ندارد، بنابراین در همه شبیه‌سازی‌ها برای P یک ماتریس قطری به صورت $P = 0.1 \times I_{N \times N}$ در نظر گرفته می‌شود اما انتخاب مقدار Q تأثیر زیادی بر عملکرد الگوریتم پیشنهادی دارد. به صورت تجربی دریافتیم که هرچه میزان انحراف ماتریس نمونه‌برداری از ماتریس گوسی و iid بیشتر باشد، میزان اعتماد به مدل سیستم کمتر می‌شود و طبق بخش ۳-۲، مقدار Q باید زیاد انتخاب شود و بالعکس.

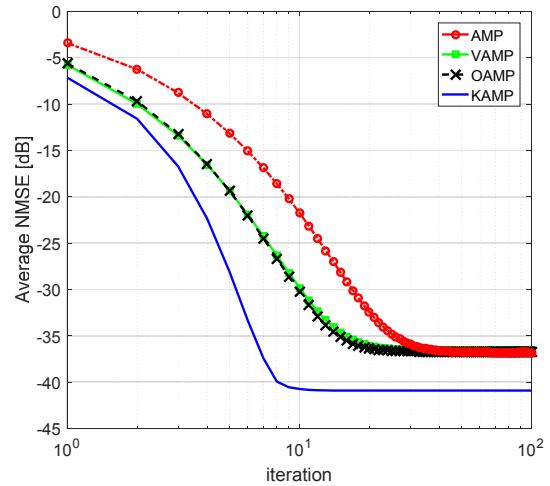
جهت بررسی میزان مقاومت الگوریتم پیشنهادی در برابر ماتریس‌های غیر گوسی و iid و مقایسه این الگوریتم با الگوریتم‌های VAMP، AMP و OAMP، ماتریس نمونه‌برداری $A_{M \times N}$ ($M = 250$ و $N = 500$) در پنج حالت ماتریس گوسی و iid، ماتریس بدحالت، ماتریس با میانگین غیر

1. Low-Rank Product Matrices

۲. کد الگوریتم VAMP در [۳۷] موجود است و از نویسندگان [۲۶] به خاطر این که کد الگوریتم OAMP را در اختیار ما گذاشتند، قدردانی می‌شود.



شکل ۵: نمودار میانگین NMSE در هر تکرار برای الگوریتم‌های مختلف با ماتریس نمونه‌بردار بدحالت $k(A) = 100$ ، A و $\text{SNR} = 40 \text{ dB}$.



شکل ۴: میانگین NMSE در هر تکرار برای الگوریتم‌های مختلف با ماتریس نمونه‌بردار گوسی و iid و $\text{SNR} = 40 \text{ dB}$.

الگوریتم‌های مختلف با عدد حالت $k(A) = 100$ بررسی می‌کند. همان طور که در شکل ۵ مشاهده می‌گردد در این حالت الگوریتم AMP واگرا می‌شود. الگوریتم VAMP و OAMP همگرا می‌شوند اما مقدار NMSE آنها در جواب نهایی نسبتاً زیاد می‌باشد و بنابراین دارای خطای بازبایی زیادی هستند. در حالی که الگوریتم KAMP نسبت به دو الگوریتم دیگر دارای جواب نهایی خیلی بهتری می‌باشد و متوسط NMSE آن حدوداً ۱۱ dB کمتر از VAMP و OAMP است. البته در تکرارهای اولیه مقدار متوسط NMSE در الگوریتم KAMP بیشتر از دو الگوریتم دیگر است اما بعد از چند تکرار اولیه، NMSE حاصل از KAMP نسبت به دو الگوریتم دیگر به شدت افت پیدا می‌کند که دلیل آن این است که در تکرارهای اولیه چون مقدار اولیه Q زیاد انتخاب شده است، واریانس نویز حالت در (۱۵-الف) زیاد می‌باشد و در نتیجه اعتماد به مدل سیستم کم است. بنابراین خطای تخمین زیاد شده است اما بعد از چند تکرار اولیه، عناصر قطر اصلی Q به صورت خودکار طبق خط ۱۵ از الگوریتم KAMP (شکل ۳)، کاهش می‌یابند و کم‌کم اعتماد به مدل سیستم، افزایش پیدا می‌کند.

جدول ۳ متوسط NMSE را در SNRهای مختلف برای ماتریس نمونه‌بردار خیلی بدحالت $k(A) = 10000$ با نشان می‌دهد. همان طور که مشاهده می‌شود، الگوریتم پیشنهادی در همه SNRها دارای دقت بهتری نسبت به سایر الگوریتم‌هاست. این جدول نشان می‌دهد که با کاهش SNR، خطای بازبایی در همه الگوریتم‌ها افزایش می‌یابد به طوری که برای SNRهای خیلی کم (۵ دسی‌بل) با ماتریس خیلی بدحالت ($k(A) = 10000$)، هیچ کدام از الگوریتم‌ها، تخمین خوبی ندارند اما برای SNRهای بالاتر، برخلاف الگوریتم AMP که همچنان واگرا می‌شود، الگوریتم‌های VAMP، OAMP و KAMP عملکردشان بهتر می‌شود.

در شکل ۶ میانگین NMSE حاصل از هر الگوریتم بر اساس عدد حالت ماتریس در بازه $k(A) = 1$ (یعنی سطر-متعامد^۴) تا $k(A) = 1 \times 10^6$ (یعنی خیلی بدحالت^۵) و تحت ۱۰۰ تحقق مستقل $\{A, x, w\}$ با $\text{SNR} = 60 \text{ dB}$ رسم شده است. برای وضوح بهتر، NMSE تا ماکسیمم مقدار صفر، برش خورده است. همان طور که در این شکل مشاهده

جدول ۳: متوسط NMSE (بر حسب دسی‌بل) در SNRهای مختلف برای ماتریس نمونه‌بردار بدحالت A با $k(A) = 10000$. نتایج برای الگوریتم‌های VAMP، OAMP و KAMP بعد از ۱۰۰ تکرار و برای AMP بعد از ۱۰۰۰ تکرار به دست آمده است.

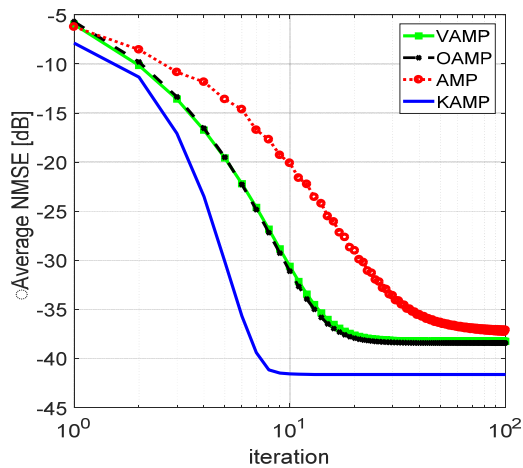
SNR (dB)	AMP	VAMP	OAMP	KAMP
۵	۰	۰	۰	۰
۱۰	۰	-۰٫۱۳۹	۰	-۰٫۹۳۵
۲۰	۰	-۲٫۶۱۲	-۰٫۲۳	-۲٫۸۰۷
۴۰	۰	-۹٫۱۰۲	-۸٫۲	-۱۰٫۸۸
۶۰	۰	-۲۰٫۴۳	-۲۰٫۹	-۴۲٫۸۱

۴-۲ نتایج برای ماتریس بدحالت

نتایج شکل ۴ نشان می‌دهد که همه الگوریتم‌ها برای ماتریس گوسی و iid همگرا خواهند شد، اما همان طور که پیش از این ذکر شد، رفتار این الگوریتم‌ها برای سایر ماتریس‌ها از جمله ماتریس‌های بدحالت متفاوت است. در این بخش میزان مقاومت هر الگوریتم برای ماتریس‌های بدحالت A با هم مقایسه می‌شود. برای این منظور هر تحقق A به شکل تجزیه مقادیر منفرد^۱ (SVD) آن $A = U \times \text{Diag}(s) \times V \in \mathbb{R}^{M \times N}$ تشکیل می‌شود که ماتریس‌های U و V مستقل و دارای توزیع هار^۲ (ماتریس‌های تصادفی مستقل و متعامد) و مقادیر ویژه بردار $s \in \mathbb{R}^M$ دارای تصاعد هندسی می‌باشند. به این معنی که $s_i/s_{i-1} = k^{(M/i)}$ به ازای $i = 1, \dots, M-1$ و $\sum_{i=1}^M s_i = N$ که در اینجا $k \geq 1$ عدد حالت^۳ نامیده می‌شود. عدد حالت معیاری برای میزان انحراف ماتریس بدحالت از ماتریس گوسی و iid با میانگین صفر می‌باشد. بنابراین در الگوریتم KAMP برای ماتریس‌های بدحالت، مقدار اولیه Q (Q) به عدد حالت ماتریس A وابسته است و اگر این ماتریس به صورت قطری و $Q = q \times I_{N \times N}$ تعریف شود، رابطه‌ای مستقیم بین مقدار q و عدد حالت ماتریس A ($k(A)$) در نظر گرفته می‌شود. البته باید q کران بالا و پایین نیز داشته باشد (که در اینجا $0.1 < q < 100$) در نظر گرفته شده است) لذا q به صورت $q = \max(\min(0.1 k(A), 100), 0.1)$ مقداردهی می‌شود. شکل ۵ میانگین NMSE را در هر تکرار برای

1. Singular Value Decomposition
2. Haar-Distributed
3. Condition Number

4. Row Orthogonal
5. Highly Ill-Conditioned



شکل ۷: میانگین NMSE در هر تکرار برای الگوریتم‌های مختلف با ماتریس سطری متعامد در $\text{SNR} = 40 \text{ dB}$.

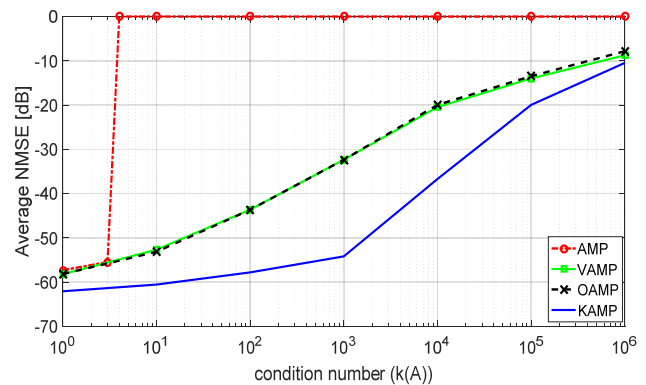
الگوریتم‌ها دارای NMSE کمتری می‌باشد و هرچه SNR بیشتر باشد این اختلاف بیشتر خواهد بود. برای مثال مقدار متوسط NMSE الگوریتم KAMP به ترتیب 0.16 dB ، 0.63 dB ، 2.3 dB ، 3.2 dB و 3.9 dB کمتر از الگوریتم OAMP در SNRهای 5 dB ، 10 dB ، 30 dB و 40 dB است.

در شکل ۷ متوسط NMSE بر حسب تکرار برای ماتریس‌های سطری متعامد حاصل از الگوریتم‌های مختلف در $\text{SNR} = 40 \text{ dB}$ رسم شده است. همان طور که مشاهده می‌شود، در این حالت نیز الگوریتم KAMP دارای جواب نهایی بهتر نسبت به سایر الگوریتم‌ها می‌باشد و همچنین با تعداد تکرار کمتری به جواب نهایی همگرا می‌شود. به عبارت دیگر شیب کاهش NMSE در الگوریتم KAMP از تکرار ۲ تا ۱۰ خیلی زیاد است و به طور کلی نسبت به سایر الگوریتم‌ها در هر تکرار عملکرد بهتری داشته است.

۴-۴ نتایج برای ماتریس با میانگین غیر صفر

الگوریتم AMP ممکن است برای حالتی که ماتریس A گوسی و iid با میانگین غیر صفر باشد واگرا شود. در این قسمت، میزان مقاومت الگوریتم پیشنهادی نسبت به میانگین غیر صفر ماتریس نمونه‌بردار بررسی می‌شود. برای این منظور عناصر این ماتریس به صورت مستقل با توزیع $A_{i,j} \sim \mathcal{N}(\mu, 1/M)$ ایجاد می‌شوند که μ معیاری برای میزان انحراف ماتریس A از ماتریس نمونه‌بردار گوسی و iid با میانگین صفر است. در این حالت برای الگوریتم KAMP، مقدار اولیه Q ($Q = q \times I_{N \times N}$) به میانگین ماتریس A (μ) وابسته است و q به صورت $q = \max(\min(10\mu, 100), 0.1)$ مقداردهی می‌شود. مانند قبل واریانس نویز باید طوری تنظیم شود که SNR مقدار مشخص شده گردد.

شکل ۸ متوسط NMSE حاصل از هر الگوریتم را در هر تکرار برای $\mu = 0.1$ و $\mu = 1$ و 100 تحقق مستقل $\{A, x, \omega\}$ با $\text{SNR} = 40 \text{ dB}$ نشان می‌دهد. در هر دو حالت الگوریتم AMP واگرا شده است. همان طور که در شکل ۸-الف دیده می‌شود، برای ماتریس نمونه‌بردار با میانگین کم ($\mu = 0.1$) رفتار دو الگوریتم OAMP و VAMP تقریباً مشابه است و به مقدار یکسانی از متوسط NMSE همگرا می‌شوند اما الگوریتم KAMP در همه تکرارها بهتر از دو الگوریتم دیگر عمل می‌کند و متوسط NMSE آن در جواب نهایی، حدوداً 4.5 dB کمتر از VAMP و OAMP می‌باشد. شکل ۸-ب که ماتریس نمونه‌بردار دارای میانگین $\mu = 1$ می‌باشد، نشان می‌دهد که دو الگوریتم VAMP



شکل ۸: میانگین NMSE حاصل از هر الگوریتم بر اساس عدد حالت ماتریس با $\text{SNR} = 60 \text{ dB}$. نتایج برای الگوریتم‌های OAMP، VAMP و KAMP بعد از ۱۰۰ تکرار و برای AMP بعد از ۱۰۰۰ تکرار به دست آمده است.

جدول ۴: متوسط NMSE بر حسب دسی‌بل در SNRهای مختلف برای ماتریس سطری متعامد. نتایج برای الگوریتم‌های OAMP، VAMP و KAMP بعد از ۱۰۰ تکرار و برای AMP بعد از ۱۰۰۰ تکرار به دست آمده است.

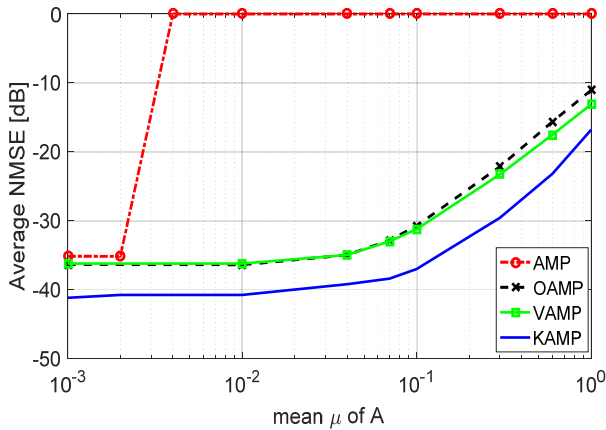
SNR (dB)	AMP	VAMP	OAMP	KAMP
5	-4	-51.058	-5.53	-5.688
10	-8.467	-9.316	-9.661	-10.29
30	-27.74	-28.53	-28.72	-31.01
40	-37.54	-38.24	-38.41	-41.62
60	-57.31	-58.19	-58.19	-62.07

می‌شود، با افزایش عدد حالت ماتریس نمونه‌بردار و یا به عبارتی با افزایش میزان انحراف ماتریس نمونه‌بردار از ماتریس گوسی و iid، NMSE همه الگوریتم‌ها افزایش می‌یابد اما شیب افزایش NMSE برای الگوریتم‌های مختلف متفاوت است. مثلاً الگوریتم AMP حتی برای ماتریس A که بد حالت خفیف ($k(A) > 3$) باشد واگرا می‌شود. عملکرد OAMP و VAMP هم مانند نتایج قبلی خیلی شبیه به هم می‌باشند و نسبت به AMP خیلی بهبود یافته‌اند اما الگوریتم KAMP برای تمام بازه $10^6 > k(A) > 1$ دارای مقدار متوسط NMSE کمتر از دو الگوریتم دیگر می‌باشد. البته این اختلاف در بازه $10^5 < k(A) < 10^6$ خیلی بیشتر است و به عنوان مثال برای $10^4 < k(A) < 10^5$ ، NMSE آن بیش از 22 dB کمتر از دو الگوریتم دیگر است. در حالی که برای $k(A) = 1$ و $k(A) = 10^6$ به ترتیب حدوداً 4 dB و 2.5 dB بهتر از VAMP و OAMP عمل می‌کند.

۴-۳ نتایج برای ماتریس سطری متعامد

در این بخش مقدار متوسط NMSE بر اساس SNR برای ماتریس سطری متعامد A بررسی می‌شود. ماتریس سطری متعامد A ، مانند بخش ۴-۲ و با عدد حالت $k(A) = 1$ تولید می‌شود که در این حالت سطرهای ماتریس A بر هم متعامد می‌شوند.

جدول ۴ مقدار متوسط NMSE حاصل از الگوریتم AMP، VAMP، OAMP و KAMP را برای پنج سطح SNR نشان می‌دهد. این مقدار از متوسط‌گیری ۱۰۰ تحقق مستقل $\{A, x, \omega\}$ به دست آمده است. همان طور که مشاهده می‌شود، در این حالت الگوریتم AMP نیز همگرا می‌شود اما مقدار NMSE آن نسبت به سایر الگوریتم‌ها بیشتر می‌باشد. از طرف دیگر در همه SNRها، الگوریتم KAMP از سایر



شکل ۹: متوسط NMSE بر حسب پارامتر میانگین (μ) ماتریس A با $\text{SNR} = 40 \text{ dB}$. نتایج برای الگوریتم‌های OAMP، VAMP و KAMP بعد از ۱۰۰ تکرار و برای AMP بعد از ۱۰۰۰ تکرار به دست آمده است.

جدول ۵: متوسط NMSE (بر حسب دسی‌بل) در SNRهای مختلف برای ماتریس گوسی IID با میانگین $\mu = 0.1$. نتایج برای الگوریتم‌های OAMP، VAMP و KAMP بعد از ۱۰۰ تکرار و برای AMP بعد از ۱۰۰۰ تکرار به دست آمده است.

SNR (dB)	AMP	VAMP	OAMP	KAMP
۵	۰	-۴,۰۱۷	-۴,۱۶۷	-۴,۵۳۵
۱۰	۰	-۸,۱۱۷	-۸,۲۵۸	-۹,۲۰۲
۲۰	۰	-۱۶,۹۸	-۱۷,۱۲	-۱۹,۵۴
۴۰	۰	-۳۶,۳۹	-۳۶,۲۲	-۴۰,۷۶

می‌شود. الگوریتم KAMP برای تمام مقادیر μ از دو الگوریتم دیگر عملکرد بهتری داشته و دارای NMSE کمتری می‌باشد.

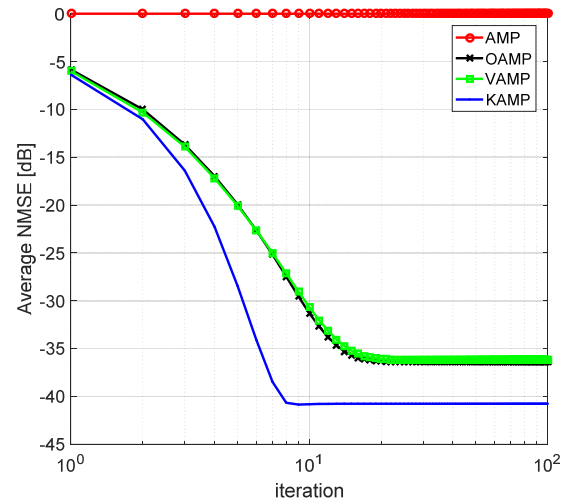
۴-۵ نتایج برای ماتریس‌های مرتبه پایین

یکی دیگر از ماتریس‌هایی که بررسی شده است ماتریس‌های مرتبه پایین می‌باشند که به صورت $A = (1/\sqrt{MR})HG$ تولید می‌شوند به طوری که $R < M$ ، $G \in \mathbb{R}^{R \times N}$ ، $H \in \mathbb{R}^{M \times R}$ و عناصر G و H گوسی و iid با میانگین صفر و واریانس یک می‌باشند. در این حالت از نسبت R/N به عنوان معیاری برای میزان انحراف ماتریس A از ماتریس گوسی و iid با میانگین صفر استفاده می‌شود [۲۱]. به طوری که هرچه مقدار R/N کمتر باشد، میزان انحراف ماتریس نمونه‌بردار A از ماتریس گوسی و iid با میانگین صفر، بیشتر می‌شود. برای این حالت در الگوریتم KAMP، مقدار q مقدار $(Q = q \times I_{N \times N})$ با نسبت R/N رابطه عکس دارد و q به صورت $q = \max(\min(2N/R, 100), 0.1)$ مقداردهی می‌شود.

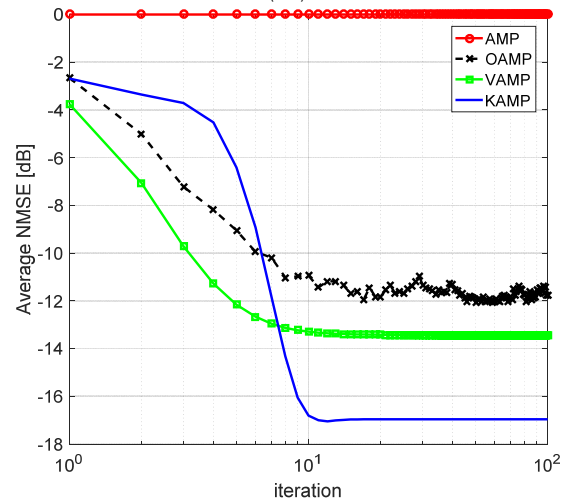
شکل ۱۰ متوسط NMSE الگوریتم‌های مختلف را در هر تکرار برای $R/N = 0.4$ و $\text{SNR} = 40 \text{ dB}$ نشان می‌دهد. در این حالت نیز الگوریتم AMP واگرا می‌شود و جواب نهایی در الگوریتم KAMP نسبت به دو الگوریتم VAMP و OAMP دارای NMSE کمتری است.

جدول ۶ متوسط NMSE را برای SNRهای مختلف با ماتریس نمونه‌بردار مرتبه پایین $R/N = 0.4$ ، تحت ۱۰۰ تحقق مستقل $\{A, x, \omega\}$ نشان می‌دهد. همان طور که مشاهده می‌شود الگوریتم AMP برای همه SNRها واگرا می‌شود و الگوریتم KAMP در همه SNRها دارای NMSE کمتری نسبت به سایر الگوریتم‌هاست.

در شکل ۱۱ متوسط NMSE حاصل از هر الگوریتم نسبت به ضریب R/N در بازه $R/N = 1$ تا $R/N = 0.4$ تحت ۱۰۰ تحقق مستقل $\{A, x, \omega\}$ با $\text{SNR} = 40 \text{ dB}$ رسم شده است. این شکل میزان مقاومت



(الف)



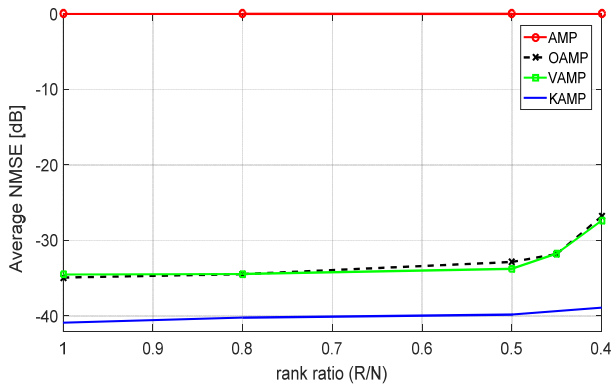
(ب)

شکل ۸: متوسط NMSE در هر تکرار با $\text{SNR} = 40 \text{ dB}$ برای ماتریس گوسی iid با میانگین (الف) $\mu = 0.1$ و (ب) $\mu = 1$.

OAMP برخلاف قبل دارای عملکرد متفاوتی هستند. در این حالت الگوریتم OAMP عملکرد مناسبی ندارد و در جواب نهایی دچار نوسان می‌شود. از طرفی دیگر الگوریتم KAMP در تکرارهای اولیه دارای متوسط NMSE بیشتر از دو الگوریتم دیگر می‌باشد و همان طور که قبلاً گفته شد، دلیل این امر اینست که مقدار Q در این حالت زیاد انتخاب شده است، اما بعد از ۷ تکرار متوسط NMSE آن کمتر از OAMP و VAMP می‌شود و در جواب نهایی به ترتیب ۳/۵ dB و ۵ dB بهتر از VAMP و OAMP عمل می‌کند.

جدول ۵ الگوریتم‌های مختلف را از نظر خطا (متوسط NMSE) در SNRهای مختلف با ماتریس نمونه‌بردار گوسی iid با میانگین $\mu = 0.1$ مقایسه می‌کند. همان طور که مشاهده می‌شود، الگوریتم پیشنهادی در همه SNRها دارای NMSE کمتری نسبت به سایر الگوریتم‌هاست.

در شکل ۹ متوسط NMSE بر حسب پارامتر میانگین (μ) در بازه ۰.۰۱ تا ۱ رسم شده است. همان طور که مشاهده می‌شود، الگوریتم AMP برای ماتریس A با کمی انحراف از ماتریس گوسی iid با میانگین صفر یعنی $\mu \geq 0.04$ ، واگرا می‌شود. سه الگوریتم دیگر هر چند با افزایش میانگین مقدار متوسط NMSE آنها افزایش می‌یابد اما عملکرد خیلی بهتری نسبت به AMP دارند. دو الگوریتم OAMP و VAMP در بازه $0.1 < \mu < 0.01$ دارای عملکرد مشابهی هستند اما در بازه ۰.۱ تا ۱، شیب افزایش متوسط NMSE برای OAMP نسبت به VAMP بیشتر



شکل ۱۱: متوسط NMSE حاصل از هر الگوریتم نسبت به ضریب R/N با $\text{SNR} = 40 \text{ dB}$. نتایج برای الگوریتم‌های OAMP، VAMP و KAMP بعد از ۱۰۰ تکرار و برای AMP بعد از ۱۰۰۰ تکرار به دست آمده است.

جدول ۶: متوسط NMSE (بر حسب دسی‌بل) در SNRهای مختلف برای ماتریس مرتبه پایین با $R/N = 0.4$. نتایج برای الگوریتم‌های OAMP، VAMP و KAMP بعد از ۱۰۰ تکرار و برای AMP بعد از ۱۰۰۰ تکرار به دست آمده است.

SNR (dB)	AMP	VAMP	OAMP	KAMP
۵	۰	-۲۶۶	-۱۸	-۲۹۶
۱۰	۰	-۵۶۷۶	-۵۰	-۶۱۹۵
۲۰	۰	-۱۲,۲۷	-۱۱,۸۹	-۱۴,۹۱
۴۰	۰	-۲۷,۳۶	-۲۶,۸۲	-۳۸,۳۸

الگوریتم با الگوریتم‌های پیشنهادی دیگر مشاهده می‌شود که نسبت به آن روش‌ها دارای عملکرد بهتری از لحاظ دقت می‌باشد و در تکرار کمتری به جواب نهایی همگرا می‌شود.

پیوست

پیوست الف: اثبات قضیه اول

فرض می‌کنیم که یک تخمین بهینه در تکرار $t-1$ به صورت $\hat{x}_{t-1} = E[x_{t-1} | y_{1:t-1}]$ با کوواریانس P_{t-1} داشته باشیم و \hat{x}_t را که تخمین پیشین x در تکرار t ام می‌نامیم به این صورت به دست بیاوریم

$$\hat{x}_t = E[x_t | y_{1:t}] = E[g(x_{t-1}) + \omega_{t-1} | y_{1:t}] = E[g(x_{t-1}) | y_{1:t-1}] \quad (۱-پ)$$

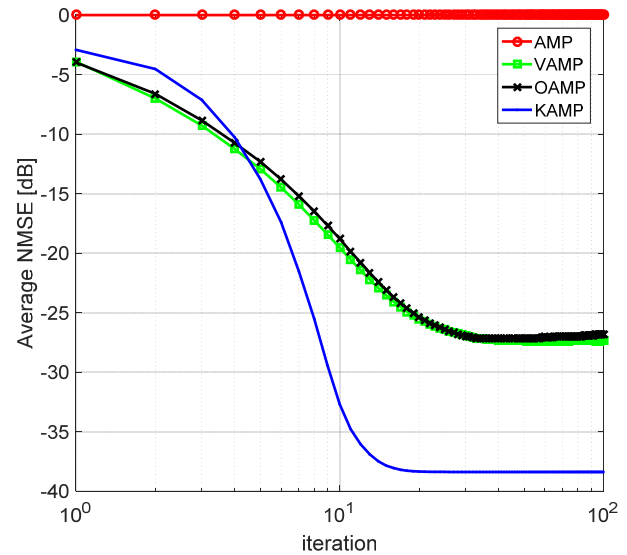
بسط مرتبه اول $g(\cdot)$ در سری تیلور حول \hat{x}_{t-1} به صورت زیر خواهد بود

$$g(x_{t-1}) = g(\hat{x}_{t-1}) + J_g(\hat{x}_{t-1})(x_{t-1} - \hat{x}_{t-1}) = g(\hat{x}_{t-1}) + J_g(\hat{x}_{t-1})e_{t-1} \quad (۲-پ)$$

که $e_{t-1} = x_{t-1} - \hat{x}_{t-1}$ و J_g ژاکوبین تابع g است و به این صورت تعریف می‌شود

$$J_g = \begin{bmatrix} \frac{\partial g_1}{\partial x_1} & \dots & \frac{\partial g_1}{\partial x_N} \\ \vdots & \ddots & \vdots \\ \frac{\partial g_N}{\partial x_1} & \dots & \frac{\partial g_N}{\partial x_N} \end{bmatrix} \quad (۳-پ)$$

و با توجه به تعریف تابع $g(\cdot)$ در (۱۴) و $r = x + A^T(y - Ax)$ می‌توان نوشت



شکل ۱۰: متوسط NMSE الگوریتم‌های مختلف در هر تکرار برای ماتریس مرتبه پایین با $R/N = 0.4$ و $\text{SNR} = 40 \text{ dB}$.

الگوریتم‌های مختلف را در برابر ماتریس مرتبه پایین، مقایسه می‌کند. همان طور که مشاهده می‌شود، الگوریتم KAMP نسبت به دو الگوریتم دیگر برای تمام مقادیر R/N دارای NMSE کمتری می‌باشد و برای $R/N < 0.5$ شیب افزایش مقدار متوسط NMSE دو الگوریتم VAMP و OAMP افزایش می‌یابد در حالی که برای الگوریتم KAMP این شیب تقریباً در تمام بازه‌ها یکسان و ملایم است. در نتیجه برای $R/N = 0.4$ که ماتریس A از ماتریس گوسی و iid بیشترین میزان انحراف را دارد اختلاف NMSE حاصل از الگوریتم پیشنهادی از دو الگوریتم دیگر حداکثر می‌باشد. به عبارت دیگر برای $R/N = 1$ الگوریتم KAMP حدوداً ۶ dB بهتر از الگوریتم OAMP و VAMP است در حالی که برای $R/N = 0.4$ ، این اختلاف به بیش از ۱۱ dB می‌رسد. لازم به ذکر است که الگوریتم AMP از همان ابتدا یعنی برای ماتریس A با $R/N = 1$ واگرا می‌شود.

۵- نتیجه‌گیری

با وجود نتایج امیدوارکننده الگوریتم بازیابی AMP در CS، به علت مشکل همگرایی این الگوریتم، در بسیاری از کاربردها نمی‌توان از آن استفاده کرد. اگرچه تکنیک‌های AMP همگرایی را برای ماتریس‌های iid و شبه‌گوسی اثبات کرده‌اند اما برای بسیاری از ماتریس‌ها که در مسایل کاربردی استفاده می‌شوند این الگوریتم ممکن است واگرا شود. به عبارت دیگر انحراف کوچکی از مدل iid و شبه‌گوسی ماتریس A می‌تواند موجب واگرایی AMP شود. در این پژوهش، الگوریتمی جدید برای بازیابی سیگنال در نمونه‌برداری فشرده با نام KAMP به منظور اصلاح الگوریتم AMP برای ماتریس‌های نمونه‌بردار غیر iid پیشنهاد شد. این الگوریتم شامل دو بخش بود. در بخش اول که مشابه الگوریتم IST است و شامل تابع آستانه‌گیر می‌باشد و در بخش دوم جواب حاصل از بخش اول مبتنی بر تئوری بیز و تخمین MAP اصلاح می‌شود. الگوریتم KAMP جهت ارزیابی، برای ماتریس‌های گوسی و iid، ماتریس‌های بدحالت، ماتریس‌های سطری متعامد، ماتریس‌های گوسی iid با میانگین غیر صفر و ماتریس‌های مرتبه پایین شبیه‌سازی شد. نتایج شبیه‌سازی نشان داد که این الگوریتم برخلاف AMP برای طیف وسیع‌تری از ماتریس‌ها همگرا می‌شود و از طرف دیگر با مقایسه این

اثبات: میانگین و کوواریانس به صورت زیر است

$$m_{y_t|x_t} = E(y_t|x_t) = E(Ax_t + \omega_t|x_t) = Ax_t \quad (۱۴-پ)$$

$$\text{cov}(y_t|x_t) = E[(y_t - m_{y_t|x_t})(y_t - m_{y_t|x_t})^T|x_t] = E[\omega_t\omega_t^T|x_t] = R \quad (۱۵-پ)$$

با توجه به دو لم بالا و قضیه اول (پ-۹) را می توان به صورت زیر نوشت

$$P(x_t|y_{:t}) = \frac{\mathcal{N}(y_t; Ax_t, R)\mathcal{N}(x_t; \hat{x}_t, P_t)}{\mathcal{N}(y_t; g(\hat{x}_t), AP_t^T A^T + R)} \quad (۱۶-پ)$$

که مخرج کسر مستقل از x_t است. در نتیجه می توان آن را یک مقدار ثابت در نظر گرفت. بنابراین داریم

$$P(x_t|y_{:t}) = A \exp[-\frac{1}{\gamma}(y_t - Ax_t)^T R(y_t - Ax_t) - \frac{1}{\gamma}(x_t - \hat{x}_t)^T P_t^-(x_t - \hat{x}_t)] \quad (۱۷-پ)$$

چون توزیع گوسی است، بنابراین $E(x_t|y_{:t}) = \arg \max[P(x_t|y_{:t})]$ (یعنی $x_{MAP} = x_{MMSE}$) پس کافی است برای یافتن ماکسیمم مقدار $P(x_t|y_{:t})$ از عبارت (پ-۱۷) نسبت به x_t مشتق گرفته و برابر با صفر قرار داده تا مقدار بهینه x_t پیدا شود.

چون تابع اکیداً صعودی است پس از لگاریتم تابع بالا مشتق می گیریم

$$\frac{\partial P(x_t|y_{:t})}{\partial x_t} \Big|_{x_t=x_{MAP}} = 0 \Rightarrow \quad (۱۸-پ)$$

$$\hat{x}_{MAP} = (A^T R^{-1} A + (P_t^-)^{-1})^{-1} [(P_t^-)^{-1} \hat{x}_t^- + A^T R^{-1} y]$$

که اگر از لم معکوس ماتریس استفاده کنیم خواهیم داشت

$$\hat{x}_{MAP} = [P_t^- - P_t^- A^T (R + AP_t^- A^T) AP_t^-]^{-1} \times [(P_t^-)^{-1} \hat{x}_t^- + A^T R^{-1} y] \quad (۱۹-پ)$$

$$\hat{x}_t^- - G_t A \hat{x}_t^- + P_t^- A^T R^{-1} y_t - G_t AP_t^- A^T R^{-1} y_t = \hat{x}_t^- + G_t (y_t - A \hat{x}_t^-) \quad (۲۰-پ)$$

در رابطه بالا تساوی a با تعریف کردن $G_t \triangleq P_t^- A^T (AP_t^- A^T + R)^{-1}$ و تساوی b با فاکتورگیری و با توجه به این که $(I - G_t A) P_t^- A^T R^{-1} = G_t$ ، به دست آمده و بنابراین (پ-۲۰) ثابت می شود. برای اثبات قسمت دوم قضیه ۲ (پ-۲۱) یعنی $P_t = (I - G_t A) P_t^-$ (رابطه (پ-۲۱)) به صورت زیر عمل می کنیم

$$(x_t - \hat{x}_t)^a = x_t - \hat{x}_t^- - G_t (y_t - A \hat{x}_t^-)^b = (I - G_t A)(x_t - \hat{x}_t^-) + G_t \omega_t \quad (۲۰-پ)$$

به طوری که تساوی a با جایگذاری (پ-۱۹) به جای \hat{x}_t و تساوی b با جایگذاری (پ-۱۵) به جای y_t به دست آمده است. در نتیجه برابر است با

$$P_t = \text{cov}[x_t|y_{:t}] = E[(x_t - \hat{x}_t)(x_t - \hat{x}_t)^T|y_{:t}]^a = (I - G_t A) P_t^- (I - G_t A)^T + G_t R_t G_t^T = (I - G_t A) P_t^- - (I - G_t A) P_t^- A^T G_t^T + G_t R_t G_t^T \quad (۲۱-پ)$$

$$(I - G_t A) P_t^-$$

که تساوی a با جایگذاری (پ-۲۰) و تساوی c با توجه به این که $(I - G_t A) P_t^- A^T G_t^T = G_t R_t G_t^T$ به دست آمده است.

$$\frac{\partial g(x; \lambda)}{\partial x} = \frac{\partial g(x; \lambda)}{\partial r} \times \frac{\partial r}{\partial x} = \frac{\partial \eta(r; \lambda)}{\partial r} \times (I - A^T A) = \begin{bmatrix} [\frac{\partial \eta(r; \lambda)}{\partial r}]_1 & & & \\ \frac{\partial r}{\partial x_1} & & & \\ & \ddots & & \\ & & \frac{[\frac{\partial \eta(r; \lambda)}{\partial r}]_N}{\partial r_N} & \end{bmatrix} \times (I - A^T A) \quad (۲-پ)$$

که چون تابع $\eta(r; \lambda)$ نسبت به r یک تابع جداپذیر است، بنابراین ماتریس ژاکوبین $\eta(r; \lambda)$ نسبت به r یک ماتریس قطری است. در نتیجه امید $g(x_{t-1})$ به شرط $y_{:t-1}$ به صورت زیر به دست می آید

$$E[g(x_{t-1})|y_{:t-1}] \approx g(\hat{x}_{t-1}) + J_g(\hat{x}_{t-1}) E[e_{t-1}|y_{:t-1}] \quad (۵-پ)$$

که $E[e_{t-1}|y_{:t-1}] = 0$ و بنابراین مقدار تخمین پیشین x_t برابر است با

$$\hat{x}_t^- \approx g(\hat{x}_{t-1}) \quad (۶-پ)$$

و خطای تخمین پیشین e_t^- که به عنوان اختلاف حالت واقعی و تخمین پیشین بیان می شود به صورت زیر به دست می آید

$$e_t^- = x_t - \hat{x}_t^- = g(x_{t-1}) + \varepsilon_{t-1} - g(\hat{x}_{t-1}) \approx J_g(\hat{x}_{t-1}) e_{t-1} + \varepsilon_{t-1} \quad (۷-پ)$$

کوواریانس خطای مورد انتظار را کوواریانس خطای تخمین (پیشین) می نامیم و با نوشتار P_t^- نمایش می دهیم

$$P_t^- = E[e_t^- e_t^{-T}] = J_g(\hat{x}_{t-1}) E[e_{t-1} e_{t-1}^T] J_g^T(\hat{x}_{t-1}) + E[\varepsilon_{t-1} \varepsilon_{t-1}^T] = J_g(\hat{x}_{t-1}) P_{t-1}^- J_g^T(\hat{x}_{t-1}) + Q_{t-1} \quad (۸-پ)$$

پیوست ب: اثبات قضیه دوم

اثبات این قضیه از طریق قانون بیز به صورت زیر است

$$P(x_t|y_{:t}) = \frac{P(y_t|x_t)P(x_t|y_{:t-1})}{P(y_t|y_{:t-1})} \quad (۹-پ)$$

برای اثبات قضیه بالا نیاز به اثبات دو لم زیر است:
لم ۱: توزیع احتمال y_t به شرط $y_{:t-1}$ یک گوسی به صورت زیر است

$$P(y_t|y_{:t-1}) = \mathcal{N}(y_t; g(\hat{x}_t), AP_t^- A^T + R) \quad (۱۰-پ)$$

اثبات: میانگین و کوواریانس به صورت زیر است

$$E(y_t|y_{:t-1}) = E(Ax_t + \omega_t|y_{:t-1}) = E[A(g(x_{t-1}) + \varepsilon_t) + \omega_t|y_{:t-1}] = AE[(g(x_{t-1})|y_{:t-1}) + AE[\varepsilon_t|y_{:t-1}] + E[\omega_t|y_{:t-1}]] = Ag(\hat{x}_{t-1}) = A\hat{x}_t^- \quad (۱۱-پ)$$

$$\text{cov}(y_t|y_{:t-1}) = \text{cov}(Ax_t + \omega_t|y_{:t-1}) = A \text{cov}(x_t|y_{:t-1}) A^T + \text{cov}(\omega_t|y_{:t-1}) = AP_t^- A^T + R_t \quad (۱۲-پ)$$

لم ۲: درست نمایی مشاهده y_t به شرط x_t ، یک گوسی به صورت زیر است

$$P(y_t|x_t) = \mathcal{N}(y_t; Ax_t, R) \quad (۱۳-پ)$$

Information Theory, ISIT'14, pp. 1812-1816, Honolulu, HI, USA, 29 Jun.-4 Jul. 2014.

- [24] A. Manoel, F. Krzakala, E. W. Tramel, and L. Zdeborov'a, "Swept approximate message passing for sparse estimation," in *Proc 32nd Int. Conf. on Machine Learning, ICML'15*, vol. 37, pp. 1123-1132, Jul. 2015.
- [25] S. Rangan, A. K. Fletcher, P. Schniter, and U. S. Kamilov, "Inference for generalized linear models via alternating directions and Bethe free energy minimization," in *Proc. IEEE Int. Symp. on Information Theory, ISIT'15*, pp. 1640-1644, Hong Kong, China, 14-19 Jun. 2015.
- [26] J. Ma and L. Ping, "Orthogonal AMP," *IEEE Access*, vol. 5, pp. 2020-2033, Jan. 2017.
- [27] S. Rangan, P. Schniter, and A. K. Fletcher, "Vector approximate message passing," *IEEE Trans. on Information Theory*, vol. 65, no. 10, pp. 6664-6684, May 2019.
- [28] A. Maleki and D. L. Donoho, "Optimally tuned iterative thresholding algorithm for compressed sensing," *IEEE J. Select. Top. Signal Processing*, vol. 4, no. 2, pp. 330-341, Apr. 2010.
- [29] I. Daubechies, M. Defrise, and C. D. Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Commun. Pure Appl. Math.*, vol. 57, no. 11, pp. 1413-1457, Nov. 2004.
- [30] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183-202, Mar. 2009.
- [31] A. Montanari, *Graphical Models Concepts in Compressed Sensing*, in *Compressed Sensing: Theory and Applications*, (Y. C. Eldar and G. Kutyniok, Eds.), Cambridge Univ. Press, 2012.
- [32] D. L. Donoho, A. Maleki, and A. Montanari, "Noise sensitivity phase transition," *IEEE Trans. Inform. Theory*, vol. 57, no. 10, pp. 6920-6941, Oct. 2011.
- [33] A. Mousavi, A. Maleki, and R. G. Baraniuk, "Parameterless optimal approximate message passing," arXiv preprint, 2013.
- [34] H. A. van der Vorst, *Iterative Krylov Methods for Large Linear Systems*, vol. 13, Cambridge, U.K.: Cambridge Univ. Press, 2003.
- [35] Alyson K. Fletcher and Philip Schniter, "Learning and Free Energies for Vector Approximate Message Passing," arXiv preprint 2016.
- [36] S. Akhlaghi, N. Zhou, and Z. Huang, "Adaptive Adjustment of Noise Covariance in Kalman Filter for Dynamic State Estimation," ArXiv preprints, Feb. 2017.
- [37] "Generalized approximate message passing," SourceForge.net project GAMPmatlab, Available on-line at <http://gampmatlab.sourceforge.net/>

فهمیه انصاری رام تحصیلات خود را در مقاطع کارشناسی و کارشناسی ارشد مهندسی برق به ترتیب در سال‌های ۱۳۸۸ و ۱۳۹۱ از دانشگاه فردوسی مشهد به پایان رسانده است و هم‌اکنون دانشجوی دکتری مهندسی برق مخابرات در دانشکده مهندسی دانشگاه فردوسی مشهد می‌باشد. زمینه‌های تحقیقاتی مورد علاقه ایشان عبارتند از: پردازش سیگنال و تصویر، فشرده‌سازی، نمونه‌برداری فشرده CS.

مرتضی خادمی تحصیلات خود را در مقاطع کارشناسی و کارشناسی ارشد مهندسی برق به ترتیب در سال‌های ۱۳۶۶ و ۱۳۶۶ در دانشگاه صنعتی اصفهان به پایان رسانده است. ایشان از سال ۱۳۶۶ الی ۱۳۷۰ به عنوان عضو هیأت علمی (مربی) در دانشگاه فردوسی مشهد به کار مشغول بود. پس از آن به دوره دکتری مهندسی برق در دانشگاه ولونگونگ (استرالیا) وارد گردیده و در سال ۱۳۷۴ موفق به اخذ درجه دکترا در مهندسی برق از دانشگاه مذکور گردید. دکتر خادمی از سال ۱۳۷۴ مجدداً در دانشکده مهندسی دانشگاه فردوسی مشهد مشغول به فعالیت گردید و اینک نیز استاد این دانشکده است. زمینه‌های علمی مورد علاقه نام‌برده شامل موضوعاتی مانند مخابرات ویدئویی، فشرده‌سازی ویدئو، پردازش تصویر، پردازش سیگنال‌های پزشکی و پنهان‌سازی اطلاعات در ویدئو می‌باشد.

عباس ابراهیمی مقدم مدرک کارشناسی و کارشناسی ارشد برق گرایش مخابرات خود را به ترتیب از دانشگاه‌های صنعتی شریف و صنعتی خواجه نصیر اخذ کرده است. ایشان مدرک دکتری خود را از دانشگاه مک‌مستر کانادا دریافت کرده و از سال ۱۳۹۰ به عنوان استادیار در دانشگاه فردوسی مشهد فعالیت علمی می‌نمایند. زمینه‌های تحقیقاتی مورد علاقه ایشان پردازش گفتار، پردازش تصویر و ویدئو، بینایی ماشین و پردازش سیگنال‌های حیاتی می‌باشد.

هادی صدوقی یزدی هم‌اکنون استاد گروه مهندسی کامپیوتر در دانشگاه فردوسی مشهد می‌باشد. ایشان در سال ۱۳۷۳ مدرک کارشناسی مهندسی برق خود را از دانشگاه

مراجع

- [1] E. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. on Inform. Theory*, vol. 52, no. 2, pp. 489-509, Feb. 2006.
- [2] D. L. Donoho, "Compressed sensing," *IEEE Trans. on Inform. Theory*, vol. 52, no. 4, pp. 1289-1306, Apr. 2006.
- [3] S. Qaisar, R. M. Bilal, W. Iqbal, M. Naureen, and S. Lee, "Compressive sensing: from theory to applications, a survey," *J. of Communications and Networks*, vol. 15, no. 5, pp. 443-456, Oct. 2013.
- [4] A. Maleki, *Approximate Message Passing Algorithms for Compressed Sensing*, PhD Thesis, Stanford University, 2011.
- [5] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," *SIAM Rev.*, vol. 43, no. 1, pp. 129-159, Mar. 2001.
- [6] W. Lu and N. Vaswani, "Modified basis pursuit denoising (modified BPDN) for noisy compressive sensing with partially known support," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing, ICASSP'10*, pp. 3926-3929, Dallas, TX, USA, 14-19 Mar. 2010.
- [7] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *Annals of Statistics*, vol. 32, no. 2, pp. 407-499, Apr. 2004.
- [8] S. Chen, S. A. Billing, and W. Lue, "Orthogonal least squares methods and their application to nonlinear system identification," *International J. of Control*, vol. 50, no. 5, pp. 1873-1896, 1989.
- [9] Y. C. Pati, R. Rezaifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition," in *Proc. 27th Asilomar Conf. on Signals, Systems and Computers*, vol. 1, pp. 40-44, Pacific Grove, CA, USA, 1-3 Nov. 1993.
- [10] D. Needell and J. A. Tropp, "CoSaMP: iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301-321, May 2008.
- [11] S. Ji, Y. Xue, and L. Carin, "Bayesian compressive sensing," *IEEE Trans. on Signal Processing*, vol. 56, no. 6, pp. 2346-2356, Jun. 2008.
- [12] H. Zayyani, M. Babaie-Zadeh, and C. Jutten, "An iterative Bayesian algorithm for sparse component analysis in presence of noise," *IEEE Trans. on Signal Processing*, vol. 57, no. 11, pp. 4378-4390, Nov. 2009.
- [13] S. Derin Babacan, R. Molina, and A. K. Katsaggelos, "Bayesian compressive sensing using Laplace priors," *IEEE Trans. on Image Processing*, vol. 19, no. 1, pp. 53-63, Jan. 2010.
- [14] D. Baron, S. Sarvotham, and R. G. Baraniuk, "Bayesian compressive sensing via belief propagation," *IEEE Trans. on Signal Processing*, vol. 58, no. 1, pp. 269-280, Jan. 2010.
- [15] R. Giri and B. D. Rao, Type I and Type II Bayesian Methods for Sparse Signal Recovery Using Scale Mixtures, arXiv preprint arXiv:1507.05087, 2015.
- [16] E. C. Marques, N. Maciel, L. A. B. Naviner, and H. Caie, "A review of sparse recovery algorithms," *IEEE Access*, vol. 7, pp. 1300-1322, Dec. 2018.
- [17] T. Blumensath and M. E. Davies, "Iterative hard thresholding for compressed sensing," *Applied and Computational Harmonic Analysis*, vol. 27, no. 3, pp. 265-274, Nov. 2009.
- [18] A. Chambolle, R. A. DeVore, N. Lee, and B. J. Lucier, "Nonlinear wavelet image processing: variational problems, compression, and noise removal through wavelet shrinkage," *IEEE Trans. Image Process.*, vol. 7, no. 3, pp. 319-335, Mar. 1998.
- [19] D. L. Donoho, A. Maleki, and A. Montanari, "Message passing algorithms for compressed sensing," in *Proc. Nat. Acad. Sci.*, vol. 106, pp. 18914-18919, Nov. 2009.
- [20] M. Bayati and A. Montanari, "The dynamics of message passing on dense graphs, with applications to compressed sensing," *IEEE Trans. Inf. Theory*, vol. 57, no. 2, pp. 764-785, Feb. 2011.
- [21] J. Vila, P. Schniter, S. Rangan, F. Krzakala, and L. Zdeborov'a, "Adaptive damping and mean removal for the generalized approximate message passing algorithm," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process., ICASSP'15*, pp. 2021-2025, Brisbane, QLD, Australia, 19-24 Apr. 2015.
- [22] S. Rangan, P. Schniter, and A. Fletcher, "On the convergence of approximate message passing with arbitrary matrices," in *Proc. IEEE Int. Symp. on Information Theory, ISIT'14*, pp. 236-240, Honolulu, HI, USA, 29 Jun.-4 Jul. 2014.
- [23] F. Caltagirone, L. Zdeborov'a, and F. Krzakala, "On convergence of approximate message passing," in *Proc. IEEE Int. Symp. on*

فردوسی مشهد و کارشناسی ارشد و دکتری مهندسی برق را به ترتیب در سال‌های ۱۳۷۵ و ۱۳۸۴ از دانشگاه تربیت مدرس دریافت نمود. زمینه‌های تحقیقاتی مورد علاقه ایشان عبارتند از: شناسایی آماری الگو، یادگیری ماشین، بینایی ماشین، پردازش سیگنال.