This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2695450, IEEE Transactions on Signal Processing

Adaptive Low-rank Matrix Completion

Ruchi Tripathi, Boda Mohan, and Ketan Rajawat, *Member, IEEE*

***Abstract*—The low-rank matrix completion problem is fundamental to a number of tasks in data mining, machine learning, and signal processing. This paper considers the problem of adaptive matrix completion in time-varying scenarios. Given a sequence of incomplete and noise-corrupted matrices, the goal is to recover and track the underlying low rank matrices. Motivated from the classical least-mean square (LMS) algorithms for adaptive filtering, three LMS-like algorithms are proposed for estimating and tracking low-rank matrices. Performance of the proposed algorithms is provided in form of non-asymptotic bounds on the tracking mean-square error. Tracking performance of the algorithms is also studied via detailed simulations over real-world datasets.**

***Index Terms*—Least mean squares, low rank matrix completion, stochastic gradient descent**

I. Introduction

The least mean square (LMS) algorithm was first introduced for adaptive system identification by Widrow and Hoff [1]. Subsequently, the LMS and its variants have been applied to a number of archetypal signal processing tasks such as noise cancellation, equalization, parameter estimation, adaptive control, etc. Recent years have seen the development of the regularized LMS algorithm, where prior information is exploited to improve estimation and tracking accuracy [2]. Examples include zero-attracting LMS (ZA-LMS) [3] and its variants that utilize the projection [4], [5] or the shrinkage [6] operators. When the parameter of interest is sparse, the regularized LMS methods not only provide superior convergence rate and steady state error, but also incur low computational complexity [2].

Recovering low-rank matrices from missing and noisy observations is a fundamental task in collaborative filtering [7], compressed sensing [8], multi-class learning [9] [10], image processing [11], and dimensionality reduction [12]. Results for exact recovery of low-rank matrices using convex optimization algorithms were first provided in [13]–[15]. A large number of low-complexity matrix completion algorithms have been developed since then, and applied to various high dimensional problems [16]–[19]. This paper considers the problem of matrix completion in dynamic scenarios. Of particular interest is the so-called *adaptive* matrix completion problem, where the measurements are available in form of a sequence of noise-corrupted, time-varying, and incomplete but low-rank matrices. The goal is to output the corresponding sequence of complete matrices that best approximate the underlying matrices. The system model is motivated from various problems that arise in network monitoring, localization, and video denoising [20]. For instance, the matrix of pairwise latencies between nodes in a network is time-varying, approximately low-rank, and generally too large to measure [21], [22]. The full set of pairwise latencies must therefore be tracked from noisy measurements made on a random subset of node pairs [23].

This paper introduces three LMS-like algorithms for the adaptive matrix completion problem at hand. The proposed algorithms are based on the stochastic gradient descent interpretation of the classical LMS algorithm. In particular, the adaptive singular value thresholding (ASVT) algorithm is based on the online linearized Bregman iteration algorithm for sparse LMS [6]. The singular value regularized LMS (SVR-LMS) algorithm is based on the ZA-LMS algorithm for sparse system identification [2], [3]. Finally, the proximal LMS algorithm utilizes proximal forward-backward splitting algorithm that has also been used for sparse adaptive filtering [24]. The major contribution of the paper is the non-asymptotic tracking performance analysis of the three algorithms. The results developed here provide explicit characterization of the tracking performance in terms of the step-size, noise variance, and the time-variability of the underlying matrices. The low-complexity and the superior tracking performance of the proposed algorithms is showcased via detailed simulations over various synthetic and real-world datasets.

This paper is organized as follows. The subsequent subsection (Sec.I-A) briefly reviews some related literature. The system model considered here is detailed in Sec. II. The three proposed methods are introduced in Sec. III, along with the different non-asymptotic results. Detailed simulation results and comparisons with state-of-the-art techniques are provided in Sec. IV. Finally Sec. V concludes the paper.

A. Related Work

This section provides the literature review for the two main ingredients of the paper, namely, sparse LMS and matrix completion. A brief discussion regarding the tools utilized for analyzing the proposed algorithms and the applications considered is also provided.

The sparse adaptive system identification problem is motivated from the advances in compressive sensing, where the goal is to estimate a sparse vector parameter [25]. Initial sparse LMS algorithms included the ZA-LMS and the reweighted ZA-LMS algorithms, which utilize a regularizer derived from the ℓ_1-norm penalty [3], [26]. Variants utilizing regularizers derived from non-convex penalty functions were subsequently proposed in [27], [28], and a unified convergence analysis for such regularized LMS algorithms was presented in [2]. Along different lines, the proximal forward-backward splitting algorithm was utilized in [24] to develop a sparsity-aware adaptive filtering framework. Another sparse LMS algorithm

Manuscript submitted April 7, 2017. This work was supported by the Indo-French Centre for the Promotion of Advanced Research-CEFIPRA. The authors are with the Department of Electrical Engineering, IIT Kanpur, Kanpur (UP), India 208016 (email: {ruchi,ketan}@iitk.ac.in).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2695450, IEEE Transactions on Signal Processing

using the linearized Bregman iterations was introduced in [6]. Finally, projection-based sparse LMS algorithm utilizing a set-theoretic approach was introduced and analyzed in [29], [30].

Following the promising results on exact matrix completion in [15], a number of low-complexity algorithms have been proposed and analyzed. The singular value thresholding (SVT) algorithm was proposed and analyzed in [16], and serves as a starting point for the ASVT algorithm proposed here. Subsequent works introduced the fixed point continuation with approximate SVD (FPCA) algorithm [31], accelerated proximal gradient (APG) algorithm [32], singular value projection (SVP) algorithm [33], and so forth. Classical optimization techniques have also been applied to this end, and include algorithms utilizing the alternating directions method of multipliers [18], [34], stochastic gradient or subgradient descent [35], [36], and block coordinate descent [37]. Another class of algorithms exploit the geometry of the space of low-rank matrices [17], [38], and include algorithms such as OPTSPACE [39], GROUSE [40], and GRASTA [41]. Of these, GROUSE and GRASTA are posed within the subspace tracking framework, where the goal is to track a low-dimensional subspace from columns that arrive sequentially over time. Another related problem is that of online matrix completion, where the entries of a large matrix are revealed gradually over time [19]. Different from these approaches, the present model allows time-varying and noisy measurements that must be handled via stochastic algorithms.

The analysis of all three algorithms is carried out using tools similar to those used in analysis of the constant step-size stochastic gradient descent [42]. It is remarked that the convergence results for various sparse LMS algorithms do not apply to the present case [2], [3], [6]. Further, the analysis of the tracking performance of regularized LMS algorithms is largely missing in the literature. Tracking performance has however been well-studied for generic and deterministic convex optimization algorithms [43], [44].

The applicability of the proposed time-varying matrix completion algorithms is demonstrated via three well-known problems. The first problem considered is that of dynamic cooperative network localization using range measurements. The traditional static network localization problem is often solved via multidimensional scaling [45], semidefinite relaxation [46], alternating directions method of multipliers [47], or subspace-based methods [48]. Most of these techniques however requires a large number of iterations to converge per time instant and are therefore not practical for mobile networks. The problem of dynamic cooperative network localization with missing measurements has first considered in [49], which proposed a subspace tracking-based algorithm. The second problem considered here is that of Internet latency measurement and tracking [22], [50]–[53]. The problem is again challenging in a dynamic context, since the number of pairwise measurements required increases as the square of the number of nodes. To this end, existing approaches utilize dynamic network kriging [22] or compressive sensing-based methods [54] for imputing the missing measurements in moderate-sized networks. For very large-scale networks, the matrix completion approach to prediction of Internet delays was first proposed in [23]. The algorithms in the present paper extend the results in [23] to time-varying scenarios. The third problem is that of removing impulse or 'salt-and-pepper' noise from videos. The salt-and-pepper noise is common in digital videos, where certain pixels exhibit minimal or maximal values [55]. Appearance of such pixels occurs due to various effects such as failures in acquisition and communication errors. An offline algorithm for denoising via low-rank matrix completion was proposed in [56]. The algorithms in the present paper provide an online solution to the same problem.

The notation used in this paper is as follows. Bold upper (lower) case letters denote matrices (vectors). The (i,j)-th entry of a matrix $\mathbf{X}$ is denoted by $[\mathbf{X}]_{ij}$. For a vector $\mathbf{x}$, $\|\mathbf{x}\|$ denotes its ℓ_2 norm. For a matrix $\mathbf{X}$, $\|\mathbf{X}\|_F$ denotes its Frobenious norm, $\|\mathbf{X}\|_2$ denotes the ℓ_2 norm, and $\|\mathbf{X}\|_\star$ denotes it nuclear norm. Given two matrices $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^{m\times n}$, their inner product is given by $\langle \mathbf{X}, \mathbf{Y}\rangle := \sum_{i,j}[\mathbf{X}]_{ij}[\mathbf{Y}]_{ij}$. The all-zero matrix of size $M \times N$ is denoted by $\mathbf{0}_{M\times N}$, while the all-one matrix of the same size is denoted by $\mathbf{1}_{M\times N}$. The subscript is dropped whenever the size is clear from the context.

II. Problem Formulation and Background

The section begins with a brief review of the low rank matrix completion problem. Let $\mathbf{M} \in \mathbb{R}^{M\times N}$ denote the low rank matrix which is observed over a subset Ω of its entries. For convenience, the set Ω is also represented as a 0-1 matrix $\mathbf{J}$, so that $[\mathbf{J}]_{ij} = 1$ if $(i,j) \in \Omega$, and zero otherwise. The matrix $\mathbf{M}$ can be completed by solving the following rank minimization problem

$$\min_{\mathbf{X}} \ \text{rank}(\mathbf{X}) \tag{1a}$$

$$\text{s. t.} \quad [\mathbf{M}]_{ij} = [\mathbf{X}]_{ij} \qquad (i,j) \in \Omega. \tag{1b}$$

In general however, the rank function is non-convex, and (1) is NP-hard to solve [13]. A commonly employed heuristic replaces the rank function with its convex approximation, the nuclear norm. The resulting convex problem becomes [11],

$$\min_{\mathbf{X}} \|\mathbf{X}\|_\star \tag{2a}$$

$$\text{s. t.} \quad [\mathbf{M}]_{ij} = [\mathbf{X}]_{ij} \qquad (i,j) \in \Omega. \tag{2b}$$

It is well-known that (2) not only recovers the exact solution to (1) under certain conditions [11], but also performs empirically well in a large number of applications. Indeed, it is now possible to solve high-dimensional instances of (2) in an efficient manner. If the entries of $\mathbf{M}$ are also noisy, (2b) may be replaced with a constraint of the form $\|\mathbf{J} \odot (\mathbf{M} - \mathbf{X})\|_F^2 \leq \epsilon$, where $\odot$ denotes the Hadmard (entry wise) product. In this case, the parameter ϵ captures the mismatch between the entries of the observed and reconstructed matrices that one is willing to tolerate [7]. This paper considers the matrix completion problem in a dynamic context, where both $\mathbf{M}_t$ and $\mathbf{J}_t$ are time-varying. In theory, it is still possible to utilize the framework of (2) and obtain the low rank matrix $\mathbf{X}_t$ at each t. For many applications however, the evolution of $\mathbf{M}_t$ and $\mathbf{J}_t$ follows a particular model, which can be utilized to estimate $\mathbf{X}_t$, either with better accuracy or at a lower complexity.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2695450, IEEE Transactions on Signal Processing

Of particular interest is the noisy measurement model

$$[\mathbf{M}_t]_{ij} = [\mathbf{X}_t^\star]_{ij} + [\mathbf{E}_t]_{ij} \qquad (i,j) \in \Omega_t, \tag{3}$$

where the noise matrix $\mathbf{E}_t$ and the set of observed entries Ω_t are independent identically distributed (i.i.d.) random variables, and the underlying low-rank matrix $\mathbf{X}_t^\star$ evolves slowly but unpredictably over time. In particular, the subsequent analysis will be carried out by assuming that the change $\left\|\mathbf{X}_t^\star - \mathbf{X}_{t-1}^\star\right\|$ is bounded above by a small constant α.

III. Proposed Methods

This section provides three *adaptive* algorithms for the dynamic matrix completion problem following the model in (3). The proposed methods build upon the classical LMS algorithm, which is reviewed briefly in the next subsection.

A. The Least Mean Squares Algorithm

The classical LMS adaptive filter is a stochastic gradient descent (SGD) algorithm that allows sequential estimation of an unknown parameter $\mathbf{X}$ by minimizing a random loss function $\ell_t(\mathbf{X})$. The LMS updates take the form

$$\hat{\mathbf{X}}_{t+1} = \hat{\mathbf{X}}_t - \mu \nabla \ell_t(\hat{\mathbf{X}}_t) \tag{4}$$

where μ is the learning rate or step size, and the loss function is usually the mean-squared error (MSE); e.g., $\ell_t(\mathbf{X}) = \|\mathbf{J}_t \odot (\mathbf{M}_t - \mathbf{X})\|_F^2$. The term $\nabla \ell_t(\mathbf{X})$ may be viewed as an instantaneous approximation to the gradient of the average loss $\mathbb{E}\left[\ell_t(\mathbf{X})\right]$; see e.g. [57].

Another interpretation follows from the *adaptive* convex optimization framework, where the updates are obtained from minimizing the empirical loss and a regularizer,

$$\hat{\mathbf{X}}_{t+1} = \arg\min_{\mathbf{X}} \mu \sum_{\tau=1}^{t} \ell_\tau(\mathbf{X}) + \frac{1}{2}\|\mathbf{X}\|_F^2 \tag{5a}$$

$$\approx \arg\min_{\mathbf{X}} \mu \sum_{\tau=1}^{t} \langle \nabla \ell_\tau(\mathbf{X}_\tau), \mathbf{X} - \mathbf{X}_\tau \rangle + \frac{1}{2}\|\mathbf{X}\|_F^2 \tag{5b}$$

where the first order approximation to the loss function is utilized in (5b). Differentiating the objective in (5b) and setting it equal to zero at $\hat{\mathbf{X}}_{t+1}$ yields

$$\hat{\mathbf{X}}_{t+1} = -\mu \sum_{\tau=1}^{t} \nabla \ell_\tau(\mathbf{X}_\tau) \tag{6}$$

Similarly, it holds that $\hat{\mathbf{X}}_t = -\mu \sum_{\tau=1}^{t-1} \nabla \ell_\tau(\mathbf{X}_\tau)$, which upon substituting into (6) yields the update in (4). It is remarked that higher order approximations can also be used in (5b), yielding higher order variants of LMS. The interpretations described here will subsequently be utilized in order to develop LMS variants that incorporate prior information regarding the unknown parameter.

Before detailing the different algorithms, some of the necessary assumptions are first stated.

A1. The measurements follow the model in (3) where the independent random variables $\mathbf{E}_t$ and $\mathbf{J}_t$ satisfy $\mathbb{E}\left[\mathbf{E}_t\right] = \mathbf{0}_{M\times N}$ and $\mathbb{E}\left[\mathbf{J}_t\right] = \rho \mathbf{1}_{M\times N}$ for all $t \geq 1$. Here, $0 < \rho < 1$ and $\mathbf{1}_{M\times N}$ is the all-one matrix of size $M \times N$.

A2. Both $\mathbf{J}_t$ and $\mathbf{E}_t$ are i.i.d. random variables, with $\sigma^2 := \mathbb{E}\left[\|\mathbf{J}_t \odot \mathbf{E}_t\|_F^2\right]$.

A3. The time-variations in $\mathbf{X}_t^\star$ are bounded as $\left\|\mathbf{X}_{t+1}^\star - \mathbf{X}_t^\star\right\| \leq \alpha$, where α is a constant.

It can be seen that the assumptions (**A1**)-(**A3**) are relatively mild, and hold for almost all scenarios of interest. The assumption in (**A3**) is particularly generic, and allows the small changes in $\mathbf{X}_t^\star$ to be arbitrary and even adversarial.

B. Adaptive *Singular Value Thresholding*

Prior parameter information can be incorporated within the *adaptive* convex optimization framework through appropriate modifications to the regularization term. In the present context, the following update rule

$$\hat{\mathbf{X}}_{t+1} = \arg\min_{\mathbf{X}} \mu \sum_{\tau=1}^{t} \ell_\tau(\mathbf{X}) + \lambda \|\mathbf{X}\|_\star + \frac{1}{2}\|\mathbf{X}\|_F^2 \tag{7a}$$

$$\approx \arg\min_{\mathbf{X}} \mu \sum_{\tau=1}^{t} \langle \nabla \ell_\tau(\mathbf{X}_\tau), \mathbf{X} - \mathbf{X}_\tau \rangle + \lambda \|\mathbf{X}\|_\star + \frac{1}{2}\|\mathbf{X}\|_F^2 \tag{7b}$$

encourages the estimates $\hat{\mathbf{X}}_{t+1}$ to have low rank, owing to the inclusion of the regularizer $\|\mathbf{X}\|_\star$. The tuning parameter allows one to trade off reconstruction error with the rank of the reconstructed matrix $\hat{\mathbf{X}}_{t+1}$. Since the nuclear norm is not differentiable, the optimality condition for (7b) is given by

$$\mu \sum_{\tau=1}^{t} \nabla \ell_\tau(\mathbf{X}_\tau) + \mathbf{Y}_{t+1} = 0 \tag{8}$$

$$\text{where } \mathbf{Y}_{t+1} \in \partial\left(\lambda \|\mathbf{X}\|_\star + \frac{1}{2}\|\mathbf{X}\|_F^2\right)\Big|_{\mathbf{X}=\hat{\mathbf{X}}_{t+1}}$$

and ∂ denotes the subgradient operator. Similarly for time t, it holds that

$$\mu \sum_{\tau=1}^{t-1} \nabla \ell_\tau(\mathbf{X}_\tau) + \mathbf{Y}_t = 0 \tag{9}$$

Combining (8) and (9), the update for $\mathbf{Y}_{t+1}$ becomes

$$\mathbf{Y}_{t+1} = \mathbf{Y}_t - \mu \nabla \ell_t(\mathbf{X}_t) \tag{10}$$

$$= \mathbf{Y}_t + \mu \mathbf{J}_t \odot \left(\mathbf{M}_t - \hat{\mathbf{X}}_t\right). \tag{11}$$

Observe next that (7b) can be re-written as follows

$$\begin{aligned}\hat{\mathbf{X}}_{t+1} &= \arg\min_{\mathbf{X}} \mu \left\langle \sum_{\tau=1}^{t} \nabla \ell_\tau(\mathbf{X}_\tau), \mathbf{X} \right\rangle + \lambda \|\mathbf{X}\|_\star + \frac{1}{2}\|\mathbf{X}\|_F^2 \\ &= \arg\min_{\mathbf{X}} -\langle \mathbf{Y}_{t+1}, \mathbf{X}\rangle + \lambda \|\mathbf{X}\|_\star + \frac{1}{2}\|\mathbf{X}\|_F^2 \\ &= \arg\min_{\mathbf{X}} \frac{1}{2}\|\mathbf{X} - \mathbf{Y}_{t+1}\|_F^2 + \lambda \|\mathbf{X}\|_\star \\ &=: \mathbf{D}_\lambda(\mathbf{Y}_{t+1})\end{aligned} \tag{12}$$

The SVT operator $\mathbf{D}_\lambda(\cdot)$ was first utilized in [16] for the SVT algorithm for completing low rank matrices. When applied to

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2695450, IEEE Transactions on Signal Processing

a matrix $\mathbf{A}$ with singular value decomposition $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$, the SVT operator yields

$$\mathbf{D}_\lambda(\mathbf{A}) = \mathbf{U}\mathbf{D}_\lambda(\mathbf{\Sigma})\mathbf{V}^T \tag{13}$$

$$= \mathbf{U}\text{Diag}\left([\sigma_1 - \lambda]_+ \ [\sigma_2 - \lambda]_+ \ \ldots\right)\mathbf{V}^T \tag{14}$$

where $\mathbf{\Sigma} = \text{Diag}\left(\sigma_1,\ \sigma_2,\ \ldots\right)$. As summarized in Algorithm 1, the *adaptive* SVT algorithm starts with an initial $\mathbf{Y}_1$, and iteratively updates $\hat{\mathbf{X}}_t$ and $\mathbf{Y}_t$ for all $t \geq 1$ via (11) and (12).

Algorithm 1: The ASVT Algorithm

Data: $\{\mathbf{M}_t\}_{t>1}$, $\{\mathbf{J}_t\}_{t\geq 1}$
Result: $\{\hat{\mathbf{X}}_t\}$
Initialization: $\mathbf{Y}_1, \lambda, \mu$;
for t = *1, 2, ...* **do**
  Update $\mathbf{Y}_{t+1} = \mathbf{Y}_t + \mu\mathbf{J}_t \odot \left(\mathbf{M}_t - \hat{\mathbf{X}}_t\right)$
  Calculate $\hat{\mathbf{X}}_{t+1} = \mathbf{D}_\lambda(\mathbf{Y}_{t+1})$
end

The following theorem characterizes the tracking error performance of the ASVT algorithm and is proved in Appendix B.

Theorem 1. *The tracking MSE of the ASVT algorithm satisfies the following properties for any $\eta > 0$ and $0 < \mu < 1$,*

$$\min_{1\leq t\leq T+1} \left\|\hat{\mathbf{X}}_t - \mathbf{X}_t^\star\right\|_F \leq \psi_{asvt} + \eta \quad a.\ s. \tag{15}$$

$$\mathbb{E}\left[\left\|\hat{\mathbf{X}}_t - \mathbf{X}_t^\star\right\|_F\right] \leq (1 - \rho\mu(2-\mu))^t \left\|\mathbf{Y}_1 - \mathbf{Y}_1^\star\right\|_F + \psi_{asvt} \tag{16}$$

where the random variable T is such that

$$\mathbb{E}\left[T\right] \leq \left\|\mathbf{Y}_1 - \mathbf{Y}_1^\star\right\|_F / \eta \tag{17}$$

and,

$$\psi_{asvt} := 2\rho\lambda + \frac{\mu\sigma + \alpha}{\rho\mu(2-\mu)}. \tag{18}$$

Theorem 1 specifies two different ways in which the tracking MSE evolves over time. Alternatively, the bounds in (15) and (16) also specify the rate at which the initial condition is forgotten. The result in (15) specifies the average duration that it takes for the minimum MSE to fall below a certain threshold. It also follows from (15) that the MSE crosses this threshold with probability one, that is, for almost every run of the ASVT algorithm. Further, since the starting point is arbitrary, the MSE crosses this threshold infinitely often. On the other hand, the bound in (16) explicates the exponential decay in the average MSE up to a tolerance ψ_{asvt}. The value ψ_{asvt} can also be interpreted as a bound on the steady-state MSE of the ASVT algorithm.

Different from the classical constant-step size stochastic gradient algorithms, Theorem 1 does not require μ to be small. In contrast, choosing an appropriate value of μ is critical to the tracking task as hand. Indeed, if μ is too small, the algorithm may not be able to track the changes in $\mathbf{X}_t^\star$. On the other hand, a larger μ allows tracking but makes the algorithm more sensitive to noise [57].

C. Singular Value Regularized LMS

The SVR-LMS algorithm utilizes a modified loss function that promotes the optimization variable $\mathbf{X}$ to have low rank. The proposed algorithm is inspired from the regularized LMS algorithm for tracking sparse vectors [36] [6] [35]. In the present case, the so called zero-attracting loss function is the instantaneous squared error with nuclear norm penalty

$$\ell_t(\mathbf{X}) = \frac{1}{2}\left\|\mathbf{J}_t \odot (\mathbf{M}_t - \mathbf{X})\right\|_F^2 + \lambda\left\|\mathbf{X}\right\|_\star$$

The non-differentiable regularizer $\|\mathbf{X}\|_\star$ encourages the minimizer of $\ell_t(\mathbf{X})$ to have low rank. The SVR-LMS updates take the following form:

$$\hat{\mathbf{X}}_{t+1} = \hat{\mathbf{X}}_t + \mu\mathbf{J}_t \odot \left(\mathbf{M}_t - \hat{\mathbf{X}}_t\right) - \mu\lambda\partial\left(\left\|\hat{\mathbf{X}}_t\right\|_\star\right) \tag{19}$$

where $\partial(\cdot)$ denotes the subgradient operator. For a rank-r matrix $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \in \mathbb{R}^{M\times N}$, let $\mathbf{U}_{1:M,1:r}$ denote the matrix formed by the first r columns of $\mathbf{U}$, and likewise for $\mathbf{V}$. Then it holds that [58]

$$\mathbf{U}_{1:M,1:r}\mathbf{V}_{1:N,1:r}^T \in \partial\left\|\mathbf{X}\right\|_\star. \tag{20}$$

Assuming that $\hat{\mathbf{X}}_t = \hat{\mathbf{U}}_t\hat{\mathbf{\Sigma}}_t\hat{\mathbf{V}}_t^T$ and denoting $\mathbf{U}_t := \hat{\mathbf{U}}_{t,1:M,1:r}$ and $\mathbf{V}_t := \hat{\mathbf{V}}_{t,1:N,1:r}$, the SVR-LMS updates become

$$\hat{\mathbf{X}}_{t+1} = \hat{\mathbf{X}}_t + \mu\mathbf{J}_t \odot (\mathbf{M}_t - \hat{\mathbf{X}}_t) - \mu\lambda\hat{\mathbf{U}}_t\hat{\mathbf{V}}_t^T \tag{21}$$

In the present context, the last term in (21) is zero-attracting, with its strength depending on the parameter λ.

The following two theorems, whose proofs are provided in Appendix C, establish the key properties of the SVR-LMS algorithm. For the purpose of these two theorems, let $R := \sup_t \text{rank}(\mathbf{X}_t)$, where we have that $R \leq \min(M, N)$.

Theorem 2. *The asymptotic tracking bias in iterates produced by the SVR-LMS algorithm is bounded as follows:*

$$\limsup_{t\to\infty}\left\|\mathbb{E}\left[\hat{\mathbf{X}}_t - \mathbf{X}_t^\star\right]\right\| \leq \frac{\lambda\sqrt{R}}{\rho} + \frac{\alpha}{\mu\rho} \tag{22}$$

Theorem 3. *The tracking MSE of the SVR-LMS algorithm satisfies the following properties for any $\eta > 0$:*

$$\min_{1\leq t\leq T+1} \left\|\hat{\mathbf{X}}_t - \mathbf{X}_t^\star\right\|_F \leq \psi_{svrlms} + \eta \quad a.\ s. \tag{23}$$

$$\mathbb{E}\left[\left\|\hat{\mathbf{X}}_t - \mathbf{X}_t^\star\right\|_F\right] \leq (1 - \rho\mu(2-\mu))^t \left\|\hat{\mathbf{X}}_1 - \mathbf{X}_1^\star\right\|_F + \psi_{svrlms} \tag{24}$$

where the random variable T is such that

$$\mathbb{E}\left[T\right] \leq \left\|\hat{\mathbf{X}}_1 - \mathbf{X}_1^\star\right\|_F / \eta \tag{25}$$

and,

$$\psi_{svrlms} := \frac{\sqrt{\sigma^2 + \lambda^2 R}}{\rho(2-\mu)} + \frac{\alpha}{\rho\mu(2-\mu)}. \tag{26}$$

Theorem 2 provides an upper bound on the maximum steady-state bias achieved by the SVR-LMS algorithm. Interestingly, the bound in (22) does not depend on the standard deviation of the measurement noise σ. The interpretation for the results in Theorem 3 is similar to that in Theorem 1. Specifically, the bounds in (15) and (24) provide the rate at which the initial condition is forgotten, and the resulting steady-state bound on the tracking MSE.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2695450, IEEE Transactions on Signal Processing

Algorithm 2: The SVR-LMS Algorithm

Data: $\{\mathbf{M}_t\}_{t>1}$, $\{\mathbf{J}_t\}_{t\geq 1}$
Result: $\{\hat{\mathbf{X}}_t\}$
Initialization: λ, μ;
for $t = 1, 2, \ldots$ **do**
  Evaluate the singular value decomposition $\hat{\mathbf{X}}_t = \hat{\mathbf{U}}_t \hat{\mathbf{\Sigma}}_t \hat{\mathbf{V}}_t^T$
  Update $\hat{\mathbf{X}}_{t+1} = \hat{\mathbf{X}}_t + \mu \mathbf{J}_t \odot (\mathbf{M}_t - \hat{\mathbf{X}}_t) - \mu\lambda \hat{\mathbf{U}}_t \hat{\mathbf{V}}_t^T$
end

Algorithm 3: The Proximal LMS Algorithm

Data: $\{\mathbf{M}_t\}_{t>1}$, $\{\mathbf{J}_t\}_{t\geq 1}$
Result: $\{\hat{\mathbf{X}}_t\}$
Initialization: λ, μ;
for $t = 1, 2, \ldots$ **do**
  Update $\hat{\mathbf{X}}_{t+1} = \mathbf{D}_{\lambda\mu}\left(\hat{\mathbf{X}}_t + \mu \mathbf{J}_t \odot (\mathbf{M}_t - \hat{\mathbf{X}}_t)\right)$
end

D. Proximal LMS

The proximal LMS (PLMS) algorithm makes use of the proximal forward-backward splitting (PFBS) algorithm, first introduced for solving generic signal recovery problems [59]. Given convex functions $f_1 : \mathbb{R}^n \to \mathbb{R}$ and $f_2 : \mathbb{R}^n \to \mathbb{R}$, consider the general optimization problem

$$\min_{\mathbf{x}\in\mathcal{X}} f_1(\mathbf{x}) + f_2(\mathbf{x}) \tag{27}$$

where it holds that $f_2(\cdot)$ is differentiable over the convex set $\mathcal{X}$. Starting at an arbitrary $\hat{\mathbf{x}}_1 \in \mathcal{X}$, the PFBS algorithm entails the following update at iteration $t \geq 1$,

$$\hat{\mathbf{x}}_{t+1} = \text{prox}_{\mu f_1}\left(\hat{\mathbf{x}}_t - \mu \nabla f_2(\hat{\mathbf{x}}_t)\right) \tag{28}$$

where the proximal point operator is defined as

$$\text{prox}_{\mu f_1}(\mathbf{x}) = \arg\min_{\mathbf{y}\in\mathcal{X}} \mu f_1(\mathbf{x}) + \frac{1}{2}\|\mathbf{x} - \mathbf{y}\|^2. \tag{29}$$

The PFBS algorithm has been applied to adaptive filtering problems in [24], [57]. Within the present context, the convex loss function at time t is given by

$$\ell_t(\mathbf{X}) = \frac{1}{2}\|\mathbf{J}_t \odot (\mathbf{M}_t - \mathbf{X})\|_F^2 + \lambda \|\mathbf{X}\|_\star. \tag{30}$$

Observe here that the first summand in (30) is differentiable, resulting in the following proximal LMS updates

$$\hat{\mathbf{X}}_{t+1} = \mathbf{D}_{\lambda\mu}\left(\hat{\mathbf{X}}_t + \mu \mathbf{J}_t \odot (\mathbf{M}_t - \hat{\mathbf{X}}_t)\right). \tag{31}$$

The proximity operator for the present case turns out to be the singular value thresholding operator defined in (14). The following theorem, whose proof is provided in Appendix D, summarizes the main result of this subsection.

Theorem 4. *The tracking MSE of the proximal-LMS algorithm satisfies the following properties for any $t_0 \geq 1$ and given $\eta > 0$:*

$$\min_{1\leq t\leq T+1} \left\|\hat{\mathbf{X}}_t - \mathbf{X}_t^\star\right\|_F \leq \psi_{plms} + \eta \quad a.\ s. \tag{32}$$

$$\mathbb{E}\left[\left\|\hat{\mathbf{X}}_t - \mathbf{X}_t^\star\right\|_F\right] \leq (1 - \rho\mu(2-\mu))^t \left\|\hat{\mathbf{X}}_1 - \mathbf{X}_1^\star\right\| + \psi_{plms} \tag{33}$$

where the random variable T is such that

$$\mathbb{E}[T] \leq \left\|\hat{\mathbf{X}}_{t_0} - \mathbf{X}_{t_0}^\star\right\|_F / \eta \tag{34}$$

and,

$$\psi_{plms} := \frac{\lambda\mu R + \mu\sigma + \alpha}{\rho\mu(2-\mu)}. \tag{35}$$

It is evident that the overall form and interpretation of the results in Theorem 4 is the same as those in Theorems 1 and 3. The key difference between the three algorithms is among their worst case tracking MSEs. It is remarked that since all the results are in form of upper bounds on the MSE they cannot be compared directly.

IV. Experimental Results

This section provides the performance analysis for the three LMS variants proposed here. Experiments are carried out over a number of synthetic and real-world datasets, and performance is compared against other state-of-the-art matrix completion algorithms, namely, SVT [16], OPTSPACE [39], FPCA [31], and GROUSE [40]. It is remarked that since the existing algorithms are not *adaptive*, they must be run for several iterations per time slot. The singular value thresholding (SVT) algorithm [16] is similar to the ASVT algorithm proposed here, and produces the sequence of iterates $\{\mathbf{X}_k, \mathbf{Y}_k\}$ at each time slot. For comparison, SVT is run for 100 iterations per time slot, with threshold $\tau = 250$ and step size $\delta = 1$. The fixed point continuation algorithm (FPCA) [31] also utilizes the shrinkage operator, and is run until the stopping criteria is met. The default value of the parameters is used, so that tolerance (xtol) is 10^{-6}, $\mu = 10^{-8}$, $\tau = 1$, and maximum number of iterations is 500. OPTSPACE [39] exploits the geometry of the low-rank matrix space, and is essentially the steepest gradient descent algorithm on the Grassmanian manifold of r-dimensional subspaces, where r is the estimated rank of the given incomplete matrix. For the present case, OPTSPACE is run for 100 iterations per time slot, with tolerance 10^{-8}. Finally, GROUSE [40] is an incremental gradient descent algorithm that also utilizes the geometry of low-rank matrices. For the present case, GROUSE is run with a maximum of 500 iterations and a step size of 0.1. The proposed algorithms are all run with single update per time slot for all the datasets.

A. Mobile network localization: Synthetic dataset

Low rank matrices are generated from pairwise distances of $N = 50$ mobile nodes, simulated within a 1×1 unit^{2}area. The nodes lie on a $d = 3$ dimensional space, and the coordinates of the i-th node evolve according to the autoregressive model $\mathbf{x}_t^{(i)} = \nu \mathbf{x}(i)_{t-1} + \sqrt{1-\nu^2} n_t^{(i)}$, where $\nu = 0.95$ and $n_t^{(i)} \sim \mathcal{N}(0, \mathbf{I})$. The matrix of interest is the squared euclidean distance matrix generated as $[\mathbf{X}_t^\star]_{ij} = \left\|\mathbf{x}_t^{(i)} - \mathbf{x}_t^{(j)}\right\|_2^2$ and has rank $2d$. The noisy distance measurements are generated

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2695450, IEEE Transactions on Signal Processing

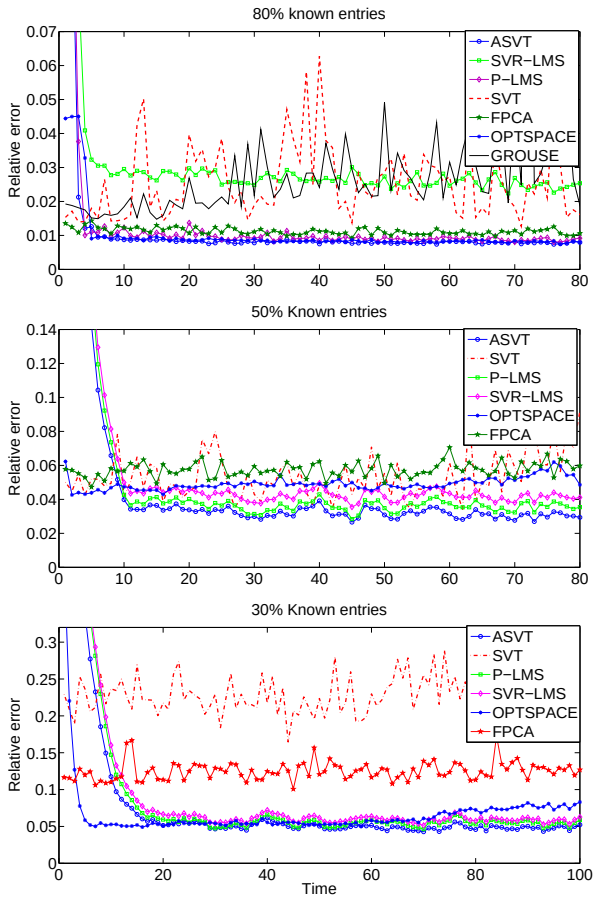

Fig. 1. Performance comparison of ASVT, PLMS and SVR-LMS with existing algorithms (SVT [16], FPCA [31], OPTSPACE [39], GROUSE [40]) at various percentages of known entries

according to the model in (3). The relative error at time t is given by

$$\text{Relative error}(t) = \frac{\|\hat{\mathbf{X}}_t - \mathbf{X}_t\|_F}{\|\mathbf{X}_t\|_F} \tag{36}$$

where $\hat{\mathbf{X}}_t$ is the recovered matrix at time t. Fig. 1 shows the performance of the ASVT, SVR-LMS, and PLMS algorithms for various loss probabilities on matrices of dimension $M = N = 50$ and rank $r = 5$. The parameters λ and μ for the three algorithms are determined by rough tuning, and turn out to be $\mu = 1$, $\lambda_{ASVT} = 120$, $\lambda_{SVR-LMS} = 6$, and $\lambda_{PLMS} = 6$. For comparison, Fig. 1 also depicts the performance of the four offline algorithms, namely, SVT, GROUSE, OPTSPACE, and FPCA, run with default parameter settings detailed earlier. From Fig. 1, it can be observed that the ASVT algorithm eventually outperforms all other algorithms for all loss probabilities, closely followed by the PLMS algorithm. The performance of the SVR-LMS algorithm is relatively poor when only a few entries are missing, but improves as the loss probability goes up. As expected, most offline algorithms perform poorly since they start afresh at each time slot, and thus do not exploit the matrix factorization carried out in the previous time slots. Further, the *adaptive* algorithms start with poor initial performance but improve and reach near-optimal performance within at most 30 time slots.

Algorithm	$M = N = 50$	$M = N = 100$	$M = N = 1000$
SVR-LMS	0.183	0.636	44.041
PLMS	0.186	0.64	45.521
ASVT	0.755	2.697	58.758
FPCA$^{(500)}$	19.614	29.859	2820.943
SVT$^{(100)}$	26.425	60.146	260.108
OPTSPACE$^{(100)}$	44.035	183.599	2979.663
GROUSE$^{(500)}$	273.781	853.741	$>$ 3600

TABLE I
RUN TIME IN SECONDS, AVERAGED OVER 100 RANDOM MATRICES.

An important aspect of the proposed *adaptive* algorithms is their low computational complexity, as compared to the offline ones. For a rough comparison, MATLAB codes for all algorithms are utilized to compare their run times, as shown in Table 1. Run times are reported for experiments carried out on an Intel core i7-4770 computer running at 3.40 GHz with 16GB DDR3 3.40 GHz RAM. A 64-bit version of MATLAB 14 was used and execution times were measured using the default system clock. The performance is compared for increasingly larger matrices with 80% known entries, and is averaged over 100 random matrices. Numbers in brackets denote the default maximum number of iterations, although the offline algorithms may stop upon convergence. From Table 1, it is clear that the proposed *adaptive* algorithms are the faster by at least two orders of magnitude. This confirms the utility of the proposed algorithms within the adaptive filtering framework considered here.

Next, the three algorithms are compared with regards to their tracking performance when the data exhibits sudden changes. In the context of mobile network localization, such a situation may arise due to an abrupt full communications failure or a 'reset.' Since the nodes continue to move while the communication system is offline, the algorithm sees the network change its configuration abruptly. In such scenarios, it is necessary that the nodes be re-localized as quickly as possible. With 50% known entries, Fig. 2 shows the effect of changing the node coordinates abruptly at $t = 100$. It can be observed that all the three algorithms recover within about ten time slots. The SVR-LMS algorithm is however slightly worse than the ASVT and the PLMS algorithms, in terms of both, asymptotic performance as well as recovery from errors.

B. Internet latency matrices

End-to-end latency is an important quality-of-service parameter that is regularly monitored by all Internet service providers. However, for a network with N nodes, measuring the full $N \times N$ latency matrix is resource intensive and impractical. Signal processing tools have been proposed to impute these missing measurements; see e.g. [22], [60]. The internet distance prediction framework performs imputation by exploiting the low-rank property of the latency matrices [21]. The latency matrices are also time-varying, due to slow changes in routes, intermittent node failures, and BGP policy changes. This motivates the application of the proposed algorithms for tracking and imputation. We consider latency matrices available from AT&T [61] ($M = N = 24$) & Keynote [62] ($M = N = 9$). Different from the localization

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2695450, IEEE Transactions on Signal Processing

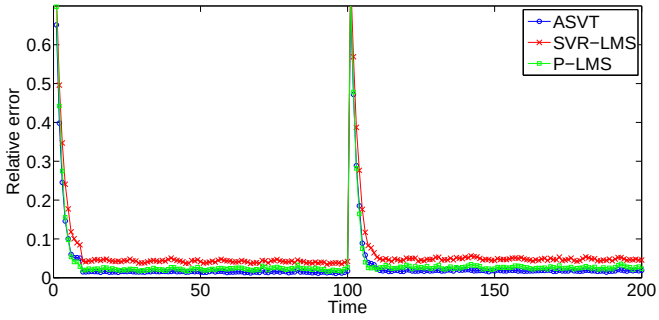

Fig. 2. Tracking performance of ASVT, PLMS, and SVR-LMS algorithms for 50% known entries. The underlying matrix changes abruptly at $t = 100$.

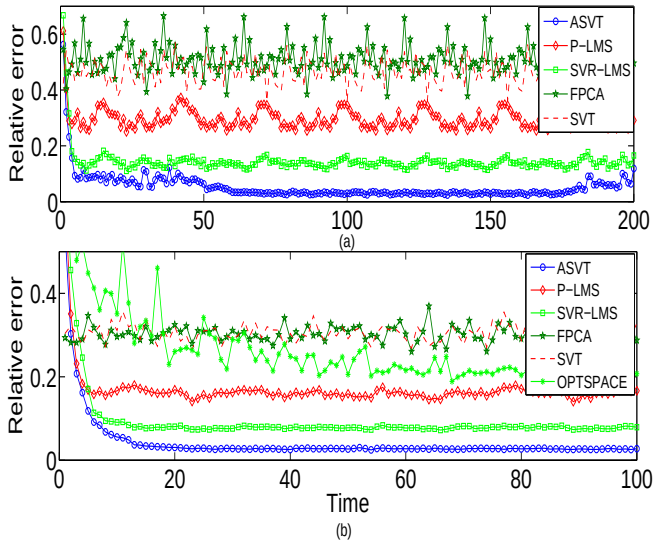

Fig. 3. Performance of various algorithms over (a) Keynote (b) AT&T data.

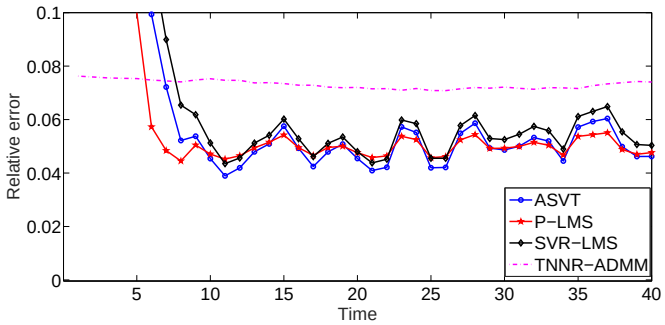

Fig. 4. Performance comparison of ASVT, PLMS, and SVR-LMS algorithms against the TNNR-ADMM [18].

problem considered earlier, the latency matrices are only approximately low rank, and highly time-varying. Within the Keynote dataset in particular, more than half the entries of the matrix change at every time slot. Fig. 3(a) & 3(b) show the performance of all six algorithms on both the datasets for 50% missing entries. Note that OPTSPACE performed poorly for the Keynote dataset, and is not shown in Fig. 3(b). It is evident from the plots that the proposed algorithms are capable of tracking highly time-varying low-rank matrices. As in Sec. IV-A, the proposed algorithms even outperform the offline algorithms that require many iterations per time slot.

C. Denoising Streaming Video

Matrix completion and rank minimization algorithms have been used for a large number image processing algorithms. For instance, it is possible to remove mixed and impulsive noise from images by modeling them as low rank matrices, and applying matrix completion algorithms [18]. Similar ideas also apply to video, which is simply a sequence of slowly changing images [20]. Indeed, the proposed *adaptive* matrix completion algorithms can be used for denoising streaming video, where frames must be processed in real-time. We consider a ten-second long, 720×1280 pixel video corrupted with impulse noise at random but known locations, which constitute 50% of the pixels. Each frame of the video is used to construct the $\mathbf{M}_t$ and $\mathbf{J}_t$ matrices, and the three channels (R, G, and B) are tracked separately. The proposed algorithms are again run with single update per time slot. For comparison, we consider the TNNR-ADMM [18] image denoising algorithm and apply it to each frame of the video with tolerance $\epsilon = 10^{-4}$, rank $r = 5$, and 50 iterations. Fig. 4 shows the relative error performance for the different algorithms. As expected the proposed algorithms exploit the temporal correlation between frames, and therefore outperform the TNNR-ADMM algorithm. The performance difference is also visible within the reconstructed video frames shown in Fig. 5. Compared to the original frame (Fig. 5(a)), the frames reconstructed using the proposed *adaptive* algorithms (Fig. 5 (c), (d), (e)), are all better than those obtained from TNNR-ADMM (Fig. 5(f)). [1]

V. Conclusion

This paper introduced algorithms for adaptive matrix completion, motivated from the classical least-mean square algorithms. Three LMS-like algorithms were proposed for sequential recovery and tracking of time-varying incomplete matrices. The three algorithms, namely, adaptive singular value thresholding algorithm (ASVT), singular value regularized LMS (SVR-LMS), and the proximal LMS (PLMS) are inspired from various adaptive algorithms for sparse system identification. The tracking performance of the three algorithms is carried out in detail, and the non-asymptotic results are provided for the same. Simulations over synthetic and real-world datasets demonstrate the superior tracking performance of the proposed algorithms compared to other state-of-the-art matrix completion algorithms.

Appendix A
Preliminaries

This appendix lists some of the preliminary results that will be repeatedly used in the proofs. As specified earlier, the noisy measurement model in (3) is adopted. The random variables $\mathbf{J}_t$ and $\mathbf{E}_t$ are assumed to be independent, with $\mathbb{E}[\mathbf{J}_t] = \rho \mathbf{1}_{M\times N}$, where $\mathbf{1}_{M\times N}$ is the $M \times N$ all-one matrix. Observe that $\mathbf{X}_t^\star$

[1] For full videos, the readers are referred to https://www.youtube.com/playlist?list=PL4ZN5_o1XEcZXqouF__rI6Ms7x7PSwRwS

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2695450, IEEE Transactions on Signal Processing

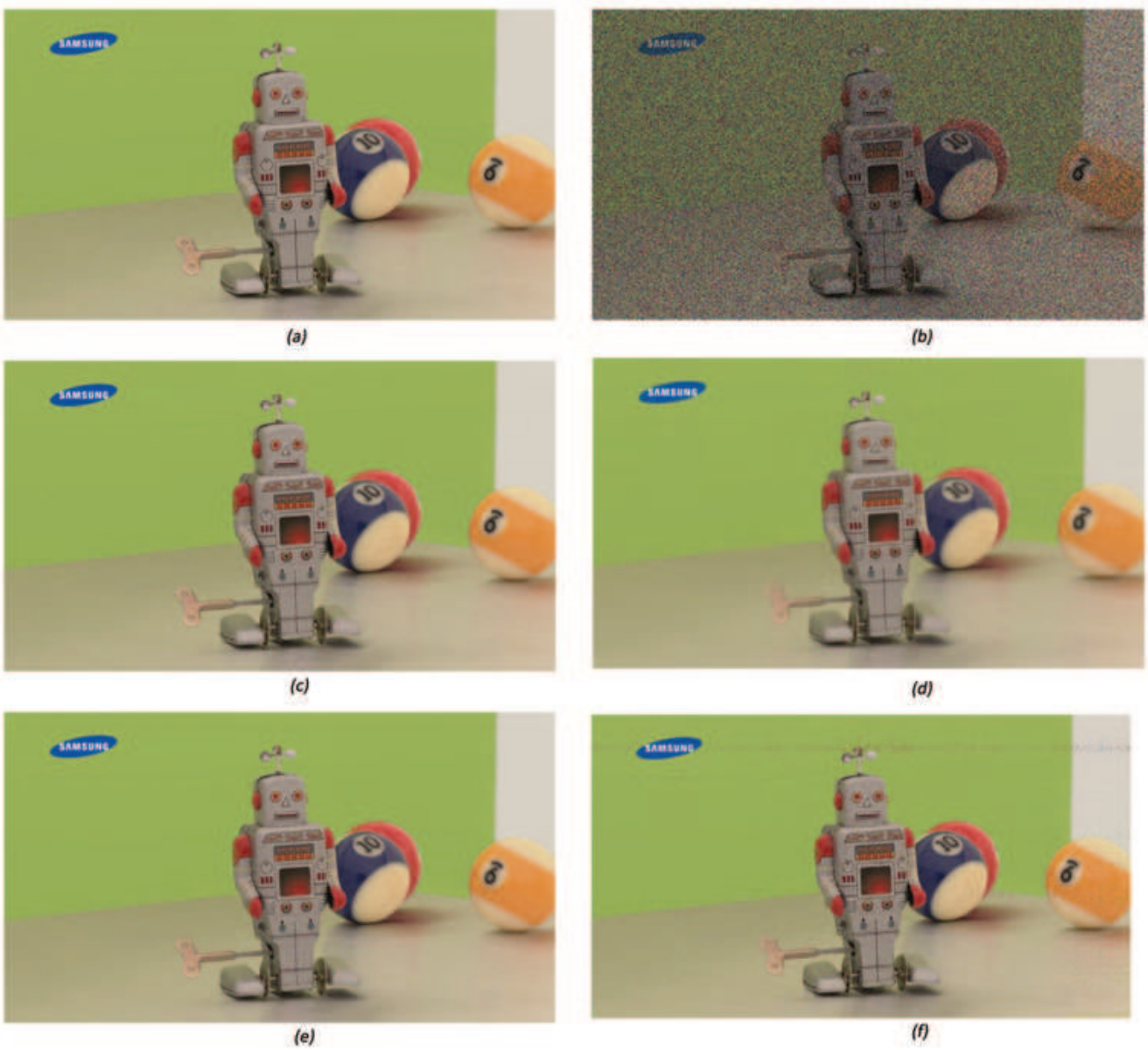

Fig. 5. (a) Original and (b) observed frames. Reconstructed frame using (c) ASVT (d) PLMS (e) SVR-LMS (f) TNNR-ADMM.

is a solution to the following stochastic convex optimization problem:

$$\mathbf{X}_t^\star = \arg\min_{\mathbf{X}} \lambda' \|\mathbf{X}\|_\star + \frac{1}{2}\|\mathbf{X}\|_F^2 \tag{37a}$$

$$\text{s.t.} \qquad \mathbb{E}[\mathbf{J}_t \odot (\mathbf{M}_t - \mathbf{X})] = 0 \tag{37b}$$

for any $\lambda' \geq 0$. Here, the expectation is with respect to the random variables $\mathbf{J}_t$ and $\mathbf{E}_t$.

Given $\mathbf{X}$, let $\mathbf{Z}(\mathbf{X})$ and $\mathbf{Z}_0(\mathbf{X})$ be the subgradients such that

$$\mathbf{Z}(\mathbf{X}) \in \partial(\lambda' \|\mathbf{X}\|_\star + \frac{1}{2}\|\mathbf{X}\|_F^2) \tag{38}$$

$$\mathbf{Z}_0(\mathbf{X}) \in \partial(\|\mathbf{X}\|_\star) \tag{39}$$

and observe that $\mathbf{Z}(\mathbf{X}) = \lambda'\mathbf{Z}_0(\mathbf{X}) + \mathbf{X}$. Further, the following properties hold from [16, Sec. 4.1] and [59],

$$\langle \mathbf{Z}(\mathbf{X}) - \mathbf{Z}(\mathbf{X}_t^\star), \mathbf{X} - \mathbf{X}_t^\star \rangle \geq \|\mathbf{X} - \mathbf{X}_t^\star\|_F^2 \tag{40}$$

$$\langle \mathbf{Z}_0(\mathbf{X}), \mathbf{X}' \rangle \leq \|\mathbf{Z}_0(\mathbf{X})\|_2 \|\mathbf{X}'\|_F \leq \|\mathbf{X}'\|_F \tag{41}$$

$$\|\mathbf{D}_\lambda(\mathbf{X}) - \mathbf{D}_\lambda(\mathbf{X}')\| \leq \|\mathbf{X} - \mathbf{X}'\| \tag{42}$$

for any $\mathbf{X}, \mathbf{X}' \in \mathbb{R}^{M\times N}$. Next, we will use the fact that for any real numbers a, b, and c, we have from Cauchy-Schwarz and triangle inequalities that

$$\sqrt{a^2 + b + c^2} \leq a + \frac{b}{2a} + c \tag{43}$$

as long as $a^2 + b \geq 0$.

Finally, the subsequent sections also utilize the following lemma, that follows from the Supermartingale convergence theorem, and generalizes the results in [42, Theorem 3.1] and [63, Prop. 5, Prop. 8]. For $t \geq 1$, we define $\mathcal{F}_t$ to be the σ-algebra generated by the random variables $\{\mathbf{E}_\tau, \Omega_\tau\}_{\tau=1}^t$. By extension, also let $\mathcal{F}_0$ to be the trivial sigma algebra.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2695450, IEEE Transactions on Signal Processing

Lemma 1. *Consider nonnegative stochastic processes $U(\mathbb{N})$and $V(\mathbb{N})$ with realizations $u(\mathbb{N})$ and $v(\mathbb{N})$ and a sequence of nested σ-algebras $\{\mathcal{F}_t\}_{t\geq 1}$ measuring $u(1:t)$ and $v(1:t)$. If $\mathbb{E}\left[u(t+1)\mid\mathcal{F}_t\right]\leq u(t)-v(t)+\varepsilon$ for all $t\geq 1$ and constant $\varepsilon\geq 0$, then the following bound holds for arbitrary $t_0\geq 1$:*

$$\min_{t_0\leq\tau\leq T+t_0} v(\tau)\leq\varepsilon+\eta \tag{44}$$

with probability one, where $\eta>0$, and the random variable T satisfies $\mathbb{E}\left[T\mid\mathcal{F}_{t_0-1}\right]\leq\frac{\mathbb{E}\left[u(t_0)|\mathcal{F}_{t_0-1}\right]}{\eta}$.

Proof: Without loss of generality, let $v(t_0)>\varepsilon+\eta$ or else, the statement in Lemma 1 clearly holds. Let $T\geq 1$ be such that $v(t_0+T)\leq\varepsilon+\eta$ and $v(t)>\varepsilon+\eta$ for all $t<T+t_0$. Define

$$\hat{v}(t):=\begin{cases}v(t)-\varepsilon & t<T+t_0\\ 0 & t\geq T+t_0\end{cases} \tag{45}$$

$$\hat{u}(t):=\begin{cases}u(t) & t<T+t_0\\ u(t_0+T) & t\geq T+t_0\end{cases} \tag{46}$$

and observe that for $t<t_0+T$, it holds that

$$\begin{aligned}\mathbb{E}\left[\hat{u}(t+1)\mid\mathcal{F}_t\right]&=\mathbb{E}\left[u(t+1)\mid\mathcal{F}_t\right] &(47)\\ &\leq u(t)-(v(t)-\varepsilon) &(48)\\ &\leq\hat{u}(t)-\hat{v}(t), &(49)\end{aligned}$$

while for $t\geq t_0+T$, we have that $\mathbb{E}\left[\hat{u}(t+1)\mid\mathcal{F}_t\right]=\mathbb{E}\left[u(t_0+T)\mid\mathcal{F}_t\right]=u(t_0+T)=\hat{u}(t)-\hat{v}(t)$. From the supermartingale convergence theorem, we have that $\sum_{t=t_0}^{\infty}\hat{v}(t)<\infty$ so that $T<\infty$ with probability one. Furthermore, taking expectation conditioned on $\mathcal{F}_{t_0-1}$ in (49), we have for all $t\geq t_0\geq 1$,

$$\mathbb{E}\left[\hat{u}(t+1)\mid\mathcal{F}_{t_0-1}\right]\leq\mathbb{E}\left[\hat{u}(t)\mid\mathcal{F}_{t_0-1}\right]-\mathbb{E}\left[\hat{v}(t)\mid\mathcal{F}_{t_0-1}\right] \tag{50}$$

$$\leq\mathbb{E}\left[u(t_0)\mid\mathcal{F}_{t_0-1}\right]-\sum_{\tau=t_0}^{t}\mathbb{E}\left[\hat{v}(t)\mid\mathcal{F}_{t_0-1}\right] \tag{51}$$

Since $\hat{u}(\mathbb{N})$ is a non-negative sequence, we have from (45) that

$$\begin{aligned}\mathbb{E}\left[u(t_0)\mid\mathcal{F}_{t_0-1}\right]&\geq\lim_{t\to\infty}\sum_{\tau=t_0}^{t}\mathbb{E}\left[\hat{v}(\tau)\mid\mathcal{F}_{t_0-1}\right] &(52)\\ &=\mathbb{E}\left[\sum_{\tau=t_0}^{t_0+T-1}v(t)-\varepsilon\mid\mathcal{F}_{t_0-1}\right] &(53)\\ &>\mathbb{E}\left[\sum_{\tau=t_0}^{t_0+T-1}\eta\mid\mathcal{F}_{t_0-1}\right]=\mathbb{E}\left[T\mid\mathcal{F}_{t_0-1}\right]\eta &(54)\end{aligned}$$

which yields the desired result. The special case of this bound for $t_0=1$ takes the form $\mathbb{E}\left[T\right]\leq u(t_0)/\eta$, and is used in the statements of the theorems. ■

APPENDIX B
CONVERGENCE OF ASVT

Before proceeding with the proof, we remark that the ASVT algorithm can be viewed as a stochastic version of the dual subgradient algorithm for solving (37a). Associating the dual variable $\mathbf{Y}$ with the constraint in (37b), the Lagrangian can be written as

$$L(\mathbf{X},\mathbf{Y})=\lambda'\left\|\mathbf{X}\right\|_{\star}+\frac{1}{2}\left\|\mathbf{X}\right\|_F^2+\mathbb{E}\left[\langle\mathbf{Y},\mathbf{J}_t\odot(\mathbf{M}_t-\mathbf{X})\rangle\right]$$

The classical dual subgradient descent algorithm for solving (37a) starts with an initial $\mathbf{Y}_0$ and utilizes the following iterations for all $k\geq 0$ [64],

$$\mathbf{Y}_{k+1}=\mathbf{Y}_k+\mu\mathbb{E}\left[\mathbf{J}_t\odot(\mathbf{M}_t-\mathbf{X}_k)\right] \tag{55a}$$

$$\begin{aligned}\mathbf{X}_k&=\arg\min_{\mathbf{X}}\lambda'\left\|\mathbf{X}\right\|_{\star}+\frac{1}{2}\left\|\mathbf{X}\right\|_F^2+\mathbb{E}\left[\langle\mathbf{Y}_k,\mathbf{J}_t\odot(\mathbf{M}_t-\mathbf{X})\rangle\right]\\ &=\arg\min_{\mathbf{X}}\lambda'\left\|\mathbf{X}\right\|_{\star}+\frac{1}{2}\left\|\mathbf{X}\right\|_F^2-\mathbb{E}\left[\langle\mathbf{J}_t\odot\mathbf{Y}_k,\mathbf{X}\rangle\right]\\ &=\arg\min_{\mathbf{X}}\lambda'\left\|\mathbf{X}\right\|_{\star}+\frac{1}{2}\left\|\mathbf{X}\right\|_F^2-\langle\rho\mathbf{1}_{M\times N}\odot\mathbf{Y}_k,\mathbf{X}\rangle\\ &=\arg\min_{\mathbf{X}}\lambda'\left\|\mathbf{X}\right\|_{\star}+\left\|\mathbf{X}-\rho\mathbf{Y}_k\right\|_F^2\\ &=\mathbf{D}_{\lambda}(\mathbf{Y}_k) &(55b)\end{aligned}$$

where $\lambda=\lambda'/\rho$. Further, let the primal-dual optimal pair be $(\mathbf{X}_t^{\star},\mathbf{Y}_t^{\star})$. It can be seen that Algorithm 1 is the corresponding stochastic dual subgradient algorithm, with only one update per time-instant t. Specifically, starting at an arbitrary $\mathbf{Y}_0$, the updates at time $t\geq 0$ take the form:

$$\mathbf{Y}_{t+1}=\mathbf{Y}_t+\mu\mathbf{J}_t\odot(\mathbf{M}_t-\hat{\mathbf{X}}_t) \tag{56}$$

$$\begin{aligned}\hat{\mathbf{X}}_t&=\arg\min_{\mathbf{X}}\lambda'\left\|\mathbf{X}\right\|_{\star}+\frac{1}{2}\left\|\mathbf{X}\right\|_F^2+\langle\mathbf{Y}_t,\mathbb{E}\left[\mathbf{J}_t\right]\odot(\mathbf{M}_t-\mathbf{X})\rangle\\ &=\mathbf{D}_{\lambda}(\mathbf{Y}_t) &(57)\end{aligned}$$

Proof of Theorem 1: We begin with the following observation:

$$\begin{aligned}&\left\|\mathbf{Y}_{t+1}-\mathbf{Y}_t^{\star}\right\|_F^2\\ &=\left\|\mathbf{Y}_t-\mathbf{Y}_t^{\star}+\mu\mathbf{J}_t\odot(\mathbf{X}_t^{\star}+\mathbf{E}_t-\hat{\mathbf{X}}_t)\right\|_F^2 &(58)\\ &=\left\|\mathbf{Y}_t-\mathbf{Y}_t^{\star}\right\|_F^2+\mu^2\left\|\mathbf{J}_t\odot(\hat{\mathbf{X}}_t-\mathbf{X}_t^{\star})\right\|_F^2+\mu^2\left\|\mathbf{J}_t\odot\mathbf{E}_t\right\|_F^2\\ &\quad-2\mu\langle\mathbf{Y}_t-\mathbf{Y}_t^{\star},\mathbf{J}_t\odot(\hat{\mathbf{X}}_t-\mathbf{X}_t^{\star})\rangle\\ &\quad+2\mu\langle\mathbf{Y}_t-\mathbf{Y}_t^{\star}-\mu\mathbf{J}_t\odot(\hat{\mathbf{X}}_t-\mathbf{X}_t^{\star}),\mathbf{J}_t\odot\mathbf{E}_t\rangle &(59)\end{aligned}$$

Observing that $\mathbf{Y}_t$ is $\mathcal{F}_t$-measurable, and taking conditional expectations given $\mathcal{F}_t$ in (59),

$$\begin{aligned}&\mathbb{E}[\left\|\mathbf{Y}_{t+1}-\mathbf{Y}_t^{\star}\right\|_F^2\mid\mathcal{F}_t]\\ &\leq\left\|\mathbf{Y}_t-\mathbf{Y}_t^{\star}\right\|_F^2+\mu^2\left\|\hat{\mathbf{X}}_t-\mathbf{X}_t^{\star}\right\|_F^2+\mu^2\sigma^2\\ &\quad-2\mu\langle\mathbf{Y}_t-\mathbf{Y}_t^{\star},\mathbb{E}\left[\mathbf{J}_t\right]\odot(\hat{\mathbf{X}}_t-\mathbf{X}_t^{\star})\rangle &(60)\\ &=\left\|\mathbf{Y}_t-\mathbf{Y}_t^{\star}\right\|_F^2+\mu^2\left\|\hat{\mathbf{X}}_t-\mathbf{X}_t^{\star}\right\|_F^2+\mu^2\sigma^2\\ &\quad-2\mu\rho\langle\mathbf{Y}_t-\mathbf{Y}_t^{\star},\hat{\mathbf{X}}_t-\mathbf{X}_t^{\star}\rangle &(61)\end{aligned}$$

where we have used the fact that $\left\|\mathbf{J}_t\odot(\hat{\mathbf{X}}_t-\mathbf{X}_t^{\star})\right\|_F^2\leq\left\|\hat{\mathbf{X}}_t-\mathbf{X}_t^{\star}\right\|_F^2$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2695450, IEEE Transactions on Signal Processing

The first order optimality condition for (57) implies that there exist subgradients

$$\mathbf{Z}(\hat{\mathbf{X}}_t) = \lambda' \mathbf{Z}_0(\hat{\mathbf{X}}_t) + \hat{\mathbf{X}}_t = \mathbb{E}\left[\mathbf{J}_t\right] \odot \mathbf{Y}_t = \rho \mathbf{Y}_t \tag{62}$$

$$\mathbf{Z}(\mathbf{X}_t^\star) = \lambda' \mathbf{Z}_0(\mathbf{X}_t^\star) + \mathbf{X}^\star = \rho \mathbf{Y}_t^\star. \tag{63}$$

Consequently, from (40), we have that

$$\rho \langle \mathbf{Y}_t - \mathbf{Y}_t^\star, \hat{\mathbf{X}}_t - \mathbf{X}_t^\star \rangle \geq \left\| \hat{\mathbf{X}}_t - \mathbf{X}_t^\star \right\|_F^2, \tag{64}$$

so that (61) can be written as

$$\begin{aligned} &\mathbb{E}[\|\mathbf{Y}_{t+1} - \mathbf{Y}_t^\star\|_F^2 \,|\mathcal{F}_t] \\ &\leq \|\mathbf{Y}_t - \mathbf{Y}_t^\star\|_F^2 - \beta\rho \langle \mathbf{Y}_t - \mathbf{Y}_t^\star, \hat{\mathbf{X}}_t - \mathbf{X}_t^\star \rangle + \mu^2\sigma^2 \end{aligned} \tag{65}$$

where $\beta = \mu(2-\mu)$. Observe that since $\rho \in (0,1)$ and $\mu \in (0,1)$, it also holds that $\beta\rho \in (0,1)$. Next, from the Cauchy-Schwartz inequality and (42), we have that

$$\begin{aligned} &\beta\rho \langle \mathbf{Y}_t - \mathbf{Y}_t^\star, \hat{\mathbf{X}}_t - \hat{\mathbf{X}}_t^\star \rangle \\ &\quad \leq \beta\rho \left\| \mathbf{Y}_t - \mathbf{Y}_t^\star \right\|_F \left\| \hat{\mathbf{X}}_t - \mathbf{X}_t^\star \right\|_F && (66) \\ &\quad \leq \beta\rho \left\| \mathbf{Y}_t - \mathbf{Y}_t^\star \right\|_F \left\| \mathbf{D}_\lambda(\mathbf{Y}_t) - \mathbf{D}_\lambda(\mathbf{Y}_t^\star) \right\|_F && (67) \\ &\quad \leq \beta\rho \left\| \mathbf{Y}_t - \mathbf{Y}_t^\star \right\|_F^2 && (68) \\ &\quad \leq \left\| \mathbf{Y}_t - \mathbf{Y}_t^\star \right\|_F^2 && (69) \end{aligned}$$

which allows us to use (43), and obtain

$$\begin{aligned} &\mathbb{E}[\left\| \mathbf{Y}_{t+1} - \mathbf{Y}_t^\star \right\|_F |\mathcal{F}_t] \leq \sqrt{\mathbb{E}[\|\mathbf{Y}_{t+1} - \mathbf{Y}_t^\star\|_F^2 \,|\mathcal{F}_t]} \\ &\leq \left\| \mathbf{Y}_t - \mathbf{Y}_t^\star \right\|_F - \beta\rho \frac{\langle \mathbf{Y}_t - \mathbf{Y}_t^\star, \hat{\mathbf{X}}_t - \mathbf{X}_t^\star \rangle}{\left\| \mathbf{Y}_{t+1} - \mathbf{Y}_t^\star \right\|_F} + \mu\sigma && (70) \\ &= \left\| \mathbf{Y}_t - \mathbf{Y}_t^\star \right\|_F - \beta\rho \left\| \mathbf{Y}_t - \mathbf{Y}_t^\star \right\|_F \\ &\qquad + \beta\rho\lambda' \frac{\langle \mathbf{Y}_t - \mathbf{Y}_t^\star, \mathbf{Z}_0(\hat{\mathbf{X}}_t) - \mathbf{Z}_0(\mathbf{X}_t^\star) \rangle}{\left\| \mathbf{Y}_t - \mathbf{Y}_t^\star \right\|_F} + \mu\sigma && (71) \\ &\leq (1-\beta\rho) \left\| \mathbf{Y}_t - \mathbf{Y}_t^\star \right\|_F + \beta\rho\lambda' \left\| \mathbf{Z}_0(\hat{\mathbf{X}}_t) - \mathbf{Z}_0(\mathbf{X}_t^\star) \right\|_2 + \mu\sigma && (72) \\ &\leq (1-\beta\rho) \left\| \mathbf{Y}_t - \mathbf{Y}_t^\star \right\|_F + 2\beta\rho^2\lambda + \mu\sigma && (73) \end{aligned}$$

where we have substituted $\mathbf{X} = \mathbf{Z}(\mathbf{X}) - \lambda' \mathbf{Z}_0(\mathbf{X})$ to obtain (71), applied Holder's inequality to obtain (72), and used triangle inequality in (73). The key inequality required for this proof is obtained through the use of triangle inequality in (73) as follows:

$$\begin{aligned} &\mathbb{E}[\left\| \mathbf{Y}_{t+1} - \mathbf{Y}_{t+1}^\star \right\|_F |\mathcal{F}_t] \\ &\leq \mathbb{E}[\left\| \mathbf{Y}_{t+1} - \mathbf{Y}_t^\star \right\|_F |\mathcal{F}_t] + \left\| \mathbf{Y}_{t+1}^\star - \mathbf{Y}_t^\star \right\| && (74) \\ &\leq (1-\beta\rho) \left\| \mathbf{Y}_t - \mathbf{Y}_t^\star \right\|_F + 2\beta\rho^2\lambda + \mu\sigma + \alpha && (75) \end{aligned}$$

Application of Lemma 1 and the use of the non-expansive property of the singular value thresholding operator (cf. (42)) yields

$$\min_{t_0 \leq t \leq t_0+T} \left\| \hat{\mathbf{X}}_t - \mathbf{X}_t^\star \right\| \leq \psi_{\text{asvt}} + \eta \tag{76}$$

with probability one and $\mathbb{E}\left[T | \mathcal{F}_{t_0}\right] \leq \frac{1}{\eta} \left\| \mathbf{Y}_{t_0} - \mathbf{Y}_{t_0}^\star \right\|_F$ for a given $\eta > 0$. Likewise, taking expectations on both sides of (75), we have that

$$\mathbb{E}\left[\left\| \mathbf{X}_{t+1} - \mathbf{X}_{t+1}^\star \right\|_F\right] \leq \mathbb{E}\left[\left\| \mathbf{Y}_{t+1} - \mathbf{Y}_{t+1}^\star \right\|_F\right]$$

$$\begin{aligned} &\leq (1-\beta\rho)\mathbb{E}\left[\left\| \mathbf{Y}_t - \mathbf{Y}_t^\star \right\|_F\right] + 2\beta\rho^2\lambda + \mu\sigma + \alpha && (77) \\ &\leq (1-\beta\rho)^{t-t_0}\mathbb{E}\left[\left\| \mathbf{Y}_{t_0} - \mathbf{Y}_{t_0}^\star \right\|_F\right] + \psi_{\text{asvt}} && (78) \end{aligned}$$

The required bounds in Theorem 1 follow upon substituting $t_0 = 1$ in (76) and (78), and replacing $t-1$ with t in (78). ■

Appendix C
Convergence Analysis of SVR-LMS Algorithm

Proof of Theorem 2: The SVR-LMS update in (21) can be written as

$$\begin{aligned} &\hat{\mathbf{X}}_{t+1} = \hat{\mathbf{X}}_t + \mu\mathbf{J}_t \odot (\mathbf{M}_t - \hat{\mathbf{X}}_t) - \mu\lambda\hat{\mathbf{U}}_t\hat{\mathbf{V}}_t^T && (79) \\ &\Rightarrow \hat{\mathbf{X}}_{t+1} - \mathbf{X}_{t+1}^\star \\ &= \hat{\mathbf{X}}_t - \mathbf{X}_t^\star + \mu\mathbf{J}_t \odot (\mathbf{X}_t^\star + \mathbf{E}_t - \hat{\mathbf{X}}_t) - \mu\lambda\hat{\mathbf{U}}_t\hat{\mathbf{V}}_t^T \\ &\qquad + (\mathbf{X}_t^\star - \mathbf{X}_{t+1}^\star) \\ &= \hat{\mathbf{X}}_t - \mathbf{X}_t^\star - \mu\mathbf{J}_t \odot (\hat{\mathbf{X}}_t - \mathbf{X}_t^\star) - \mu\mathbf{J}_t \odot \mathbf{E}_t - \mu\lambda\hat{\mathbf{U}}_t\hat{\mathbf{V}}_t^T \\ &\qquad + (\mathbf{X}_t^\star - \mathbf{X}_{t+1}^\star) && (80) \end{aligned}$$

Taking conditional expectation with respect to $\mathcal{F}_t$ yields,

$$\begin{aligned} &\mathbb{E}\left[\hat{\mathbf{X}}_{t+1} - \mathbf{X}_{t+1}^\star \mid \mathcal{F}_t\right] = (\hat{\mathbf{X}}_t - \mathbf{X}_t^\star) - \mu\mathbb{E}\left[\mathbf{J}_t\right] \odot (\hat{\mathbf{X}}_t - \mathbf{X}_t^\star) \\ &\qquad - \mu\lambda\hat{\mathbf{U}}_t\hat{\mathbf{V}}_t^T + \mathbf{X}_t^\star - \mathbf{X}_{t+1}^\star \\ &= (1-\mu\rho)(\hat{\mathbf{X}}_t - \mathbf{X}_t^\star) - \mu\lambda\hat{\mathbf{U}}_t\hat{\mathbf{V}}_t^T + (\mathbf{X}_t^\star - \mathbf{X}_{t+1}^\star) \end{aligned} \tag{81}$$

Next, taking total expectation on both sides, we obtain

$$\begin{aligned} \boldsymbol{\zeta}_{t+1} &:= \mathbb{E}\left[\hat{\mathbf{X}}_{t+1} - \mathbf{X}^\star\right] \\ &= (1-\mu\rho)\boldsymbol{\zeta}_t - \mu\lambda\mathbb{E}\left[\hat{\mathbf{U}}_t\hat{\mathbf{V}}_t^T\right] + (\mathbf{X}_t^\star - \mathbf{X}_{t+1}^\star) \end{aligned}$$

Noting that $\left\| \hat{\mathbf{U}}_t\hat{\mathbf{V}}_t^T \right\|_F^2 \leq R$ and $\left\| \mathbf{X}_t^\star - \mathbf{X}_{t+1}^\star \right\|_F \leq \alpha$, we have that

$$\begin{aligned} \left\| \boldsymbol{\zeta}_{t+1} \right\| &\leq (1-\mu\rho)^{t-t_0} \left\| \boldsymbol{\zeta}_{t_0} \right\| + (\mu\lambda\sqrt{R} + \alpha) \\ &\qquad \sum_{s=t_0+1}^{t} (1-\mu\rho)^{t-s} \left\| \vartheta_s \right\| \\ &\leq (1-\mu\rho)^t B_0 + \frac{\sqrt{R}\lambda}{\rho} + \frac{\alpha}{\mu\rho} \\ \Rightarrow \limsup_{t\to\infty} \|\boldsymbol{\zeta}_t\| &\leq \frac{\sqrt{R}\lambda}{\rho} + \frac{\alpha}{\mu\rho} \end{aligned} \tag{82}$$

which is the desired result. ■

Proof of Theorem 3: Using the relationship in (80), it holds that

$$\begin{aligned} &\left\| \hat{\mathbf{X}}_{t+1} - \mathbf{X}_t^\star \right\|_F^2 \\ &= \left\| \hat{\mathbf{X}}_t - \mathbf{X}_t^\star \right\|_F^2 + \mu^2 \left\| \mathbf{J}_t \odot (\hat{\mathbf{X}}_t - \mathbf{X}_t^\star) \right\|_F^2 + \mu^2 \left\| \mathbf{J}_t \odot \mathbf{E}_t \right\|_F^2 \\ &\quad + \mu^2\lambda^2 \left\| \hat{\mathbf{U}}_t\hat{\mathbf{V}}_t^T \right\|_F^2 - 2\mu\langle \mathbf{J}_t \odot (\hat{\mathbf{X}}_t - \mathbf{X}_t^\star), \hat{\mathbf{X}}_t - \mathbf{X}_t^\star \rangle \\ &\quad - 2\mu\langle \mathbf{J}_t \odot \mathbf{E}_t, (\mathbf{I} - \mu\mathbf{J}_t) \odot (\hat{\mathbf{X}}_t - \mathbf{X}_t^\star) - \mu\lambda\hat{\mathbf{U}}_t\hat{\mathbf{V}}_t^T \rangle \\ &\quad - \mu\lambda\langle (\mathbf{I} - \mu\mathbf{J}_t) \odot (\hat{\mathbf{X}}_t - \mathbf{X}_t^\star), \hat{\mathbf{U}}_t\hat{\mathbf{V}}_t^T \rangle \end{aligned} \tag{83}$$

As in the proof of Theorem 2, the conditional expectation of the right-hand side of (83), given $\mathcal{F}_t$ must be evaluated. To this end, note that since $\mathbf{J}_t \odot \mathbf{J}_t = \mathbf{J}_t$, we have that

$$\mathbb{E}\left[\left\| \mathbf{J}_t \odot (\hat{\mathbf{X}}_t - \mathbf{X}_t^\star) \right\|_F^2 \mid \mathcal{F}_t \right]$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2695450, IEEE Transactions on Signal Processing

$$= \mathbb{E}\left[\langle \mathbf{J}_t \odot (\hat{\mathbf{X}}_t - \mathbf{X}_t^\star), \mathbf{J}_t \odot (\hat{\mathbf{X}}_t - \mathbf{X}_t^\star)\rangle \mid \mathcal{F}_t\right]$$
$$= \langle \mathbb{E}\left[\mathbf{J}_t\right] \odot (\hat{\mathbf{X}}_t - \mathbf{X}_t^\star), \hat{\mathbf{X}}_t - \mathbf{X}_t^\star\rangle$$
$$= \rho \left\|\hat{\mathbf{X}}_t - \mathbf{X}_t^\star\right\|_F^2 \tag{84}$$

Using the fact that $\mathbf{E}_t$ is zero mean, the required conditional expectation becomes

$$\mathbb{E}\left[\left\|\hat{\mathbf{X}}_{t+1} - \mathbf{X}_t^\star\right\|_F^2 \mid \mathcal{F}_t\right]$$
$$\leq (1 + \mu^2\rho - 2\mu\rho)\left\|\hat{\mathbf{X}}_t - \mathbf{X}_t^\star\right\|_F^2 + \mu^2 \left\|\mathbf{J}_t \odot \mathbf{E}_t\right\|_F^2$$
$$+ \mu^2\lambda^2 \left\|\hat{\mathbf{U}}_t\hat{\mathbf{V}}_t^T\right\|_F^2 - (1-\rho)\mu\lambda\langle \hat{\mathbf{X}}_t - \mathbf{X}_t^\star, \hat{\mathbf{U}}_t\hat{\mathbf{V}}_t^T\rangle. \tag{85}$$

Next, since $\hat{\mathbf{U}}_t\hat{\mathbf{V}}_t^T \in \partial(\|\mathbf{X}\|_\star) \mid_{\mathbf{X}=\hat{\mathbf{X}}_t}$ and $0 \in \partial(\|\mathbf{X}\|_\star) \mid_{\mathbf{X}=\mathbf{X}^\star}$, it follows from (40) that $\langle \hat{\mathbf{U}}_t\hat{\mathbf{V}}_t^T, \hat{\mathbf{X}}_t - \mathbf{X}_t^\star\rangle \geq 0$. Further, using the bound $\left\|\hat{\mathbf{U}}_t\hat{\mathbf{V}}_t^T\right\|_F^2 \leq R$ for all t and (**A2**), it is possible to write (85) as

$$\mathbb{E}\left[\left\|\hat{\mathbf{X}}_{t+1} - \mathbf{X}_t^\star\right\|_F^2 \mid \mathcal{F}_t\right]$$
$$\leq (1-\beta\rho)\left\|\hat{\mathbf{X}}_t - \mathbf{X}_t^\star\right\|_F^2 + \mu^2\sigma^2 + \mu^2\lambda^2 R. \tag{86}$$

Application of (43) yields

$$\mathbb{E}\left[\left\|\hat{\mathbf{X}}_{t+1} - \mathbf{X}_t^\star\right\|_F \mid \mathcal{F}_t\right] \leq \sqrt{\mathbb{E}\left[\left\|\hat{\mathbf{X}}_{t+1} - \mathbf{X}_t^\star\right\|_F^2 \mid \mathcal{F}_t\right]}$$
$$\leq \left\|\hat{\mathbf{X}}_t - \mathbf{X}_t^\star\right\|_F - \beta\rho\left\|\hat{\mathbf{X}}_t - \mathbf{X}_t^\star\right\|_F + \mu\sqrt{\sigma^2 + \lambda^2 R} \tag{87}$$

Subsequently, using the triangle inequality, we have the key relationship required for this proof

$$\mathbb{E}\left[\left\|\hat{\mathbf{X}}_{t+1} - \mathbf{X}_{t+1}^\star\right\|_F \mid \mathcal{F}_t\right]$$
$$\leq (1-\beta\rho)\left\|\hat{\mathbf{X}}_t - \mathbf{X}_t^\star\right\|_F + \alpha + \mu\sqrt{\sigma^2 + \lambda^2 R} \tag{88}$$

Consequently, it follows from Lemma 1 that

$$\min_{t_0 \leq t \leq t_0+T} \left\|\hat{\mathbf{X}}_t - \mathbf{X}_t^\star\right\| \leq \psi_{\text{svrlms}} + \eta \tag{89}$$

with probability one and $\mathbb{E}\left[T|\mathcal{F}_{t_0}\right] \leq \frac{1}{\eta}\left\|\hat{\mathbf{X}}_{t_0} - \mathbf{X}_{t_0}^\star\right\|_F$ for a given $\eta > 0$, as required. Likewise, taking expectations on both sides of (88), we have that

$$\mathbb{E}\left[\left\|\hat{\mathbf{X}}_{t+1} - \mathbf{X}_{t+1}^\star\right\|_F\right]$$
$$\leq (1-\beta\rho)\mathbb{E}\left[\left\|\hat{\mathbf{X}}_t - \mathbf{X}_t^\star\right\|_F\right] + \mu\sqrt{\sigma^2 + \lambda^2 R} + \alpha \tag{90}$$
$$\leq (1-\beta\rho)^{t-t_0}\mathbb{E}\left[\left\|\hat{\mathbf{X}}_{t_0} - \mathbf{X}_{t_0}^\star\right\|_F\right] + \psi_{\text{svrlms}}. \tag{91}$$

As earlier, the required results in Theorem 3 follow by setting $t_0 = 1$. ∎

Appendix D
Convergence Analysis of PLMS algorithm

The proof of the proximal LMS utilizes the non-expansiveness of the proximity operator [59] . The overall structure of the proof is similar to that in [65], modified for the case of constant $\mu > 0$.

Proof of Theorem 4: Using the fact that the proximity operator is non-expansive, we have that

$$\left\|\hat{\mathbf{X}}_{t+1} - \mathbf{D}_{\lambda\mu}(\mathbf{X}_t^\star)\right\|_F^2$$
$$= \left\|\mathbf{D}_{\lambda\mu}(\hat{\mathbf{X}}_t + \mu\mathbf{J}_t \odot (\mathbf{M}_t - \hat{\mathbf{X}}_t)) - \mathbf{D}_{\lambda\mu}(\mathbf{X}_t^\star)\right\|_F^2$$
$$\leq \left\|\hat{\mathbf{X}}_t - \mathbf{X}_t^\star + \mu\mathbf{J}_t \odot (\mathbf{M}_t - \hat{\mathbf{X}}_t)\right\|_F^2$$
$$= \left\|\hat{\mathbf{X}}_t - \mathbf{X}_t^\star\right\|_F^2 + \mu^2\left\|\mathbf{J}_t \odot (\mathbf{M}_t - \hat{\mathbf{X}}_t)\right\|_F^2$$
$$- 2\mu\langle \hat{\mathbf{X}}_t - \mathbf{X}_t^\star, \mathbf{J}_t \odot (\hat{\mathbf{X}}_t - \mathbf{X}_t^\star)\rangle$$
$$+ 2\mu\langle \hat{\mathbf{X}}_t - \mathbf{X}_t^\star, \mathbf{J}_t \odot \mathbf{E}_t\rangle. \tag{92}$$

Recalling that $\mathcal{F}_t$ is the σ-algebra generated by the random variables $\{\mathbf{E}_\tau, \Omega_\tau\}_{\tau=1}^{t-1}$, and taking conditional expectation given $\mathcal{F}_t$ yields

$$\mathbb{E}\left[\left\|\hat{\mathbf{X}}_{t+1} - \mathbf{D}_{\lambda\mu}(\mathbf{X}_t^\star)\right\|_F^2 \mid \mathcal{F}_t\right]$$
$$\leq \left\|\hat{\mathbf{X}}_t - \mathbf{X}_t^\star\right\|_F^2$$
$$+ \mu^2\mathbb{E}\left[\left\|\mathbf{J}_t \odot (\mathbf{M}_t - \hat{\mathbf{X}}_t)\right\|_F^2 \mid \mathcal{F}_t\right]$$
$$- 2\mu\langle \hat{\mathbf{X}}_t - \mathbf{X}_t^\star, \mathbb{E}\left[\mathbf{J}_t\right] \odot (\hat{\mathbf{X}}_t - \mathbf{X}_t^\star)\rangle$$
$$+ 2\mu\langle \hat{\mathbf{X}}_t - \mathbf{X}_t^\star, \mathbb{E}\left[\mathbf{J}_t \odot \mathbf{E}_t\right]\rangle. \tag{93}$$

Recall that since $\mathbf{J}_t \odot \mathbf{M}_t = \mathbf{J}_t \odot \mathbf{X}_t^\star + \mathbf{J}_t \odot \mathbf{E}_t$, it holds that

$$\mathbb{E}\left[\left\|\mathbf{J}_t \odot (\mathbf{M}_t - \hat{\mathbf{X}}_t)\right\|_F^2 \mid \mathcal{F}_t\right]$$
$$= \mathbb{E}\left[\left\|\mathbf{J}_t \odot \mathbf{E}_t\right\|_F^2\right] + \mathbb{E}\left[\left\|\mathbf{J}_t \odot (\hat{\mathbf{X}}_t - \mathbf{X}_t^\star)\right\|_F^2 \mid \mathcal{F}_t\right]$$
$$- 2\langle \mathbb{E}\left[\mathbf{J}_t \odot \mathbf{E}_t\right], \hat{\mathbf{X}}_t - \mathbf{X}_t^\star \mid \mathcal{F}_t\rangle$$
$$= \sigma^2 + \rho\left\|\hat{\mathbf{X}}_t - \mathbf{X}_t^\star\right\|_F^2. \tag{94}$$

Therefore, (93) can be written as

$$\mathbb{E}\left[\left\|\hat{\mathbf{X}}_{t+1} - \mathbf{D}_{\lambda\mu}(\mathbf{X}_t^\star)\right\|_F^2 \mid \mathcal{F}_t\right]$$
$$\leq \left\|\hat{\mathbf{X}}_t - \mathbf{X}_t^\star\right\|_F^2 + \mu^2\rho\left\|\mathbf{X}_t^\star - \hat{\mathbf{X}}_t\right\|_F^2$$
$$- 2\mu\rho\left\|\mathbf{X}_t^\star - \hat{\mathbf{X}}_t\right\| + \mu^2\sigma^2$$
$$\leq (1-\beta\rho)\left\|\hat{\mathbf{X}}_t - \mathbf{X}_t^\star\right\|_F^2 + \mu^2\sigma^2. \tag{95}$$

Therefore, application of (43) yields

$$\mathbb{E}\left[\left\|\hat{\mathbf{X}}_{t+1} - \mathbf{D}_{\lambda\mu}(\mathbf{X}_t^\star)\right\|_F \mid \mathcal{F}_t\right]$$
$$\leq \sqrt{\mathbb{E}\left[\left\|\hat{\mathbf{X}}_{t+1} - \mathbf{X}_t^\star\right\|_F^2 \mid \mathcal{F}_t\right]}$$
$$\leq \left\|\hat{\mathbf{X}}_t - \mathbf{X}_t^\star\right\|_F - \beta\rho\left\|\hat{\mathbf{X}}_t - \mathbf{X}_t^\star\right\|_F + \mu\sigma \tag{96}$$

Further, from [59, eq. (2.12)], we have that $\|\mathbf{D}_{\lambda\mu}(\mathbf{X}_t^\star) - \mathbf{X}_t^\star\| \leq \lambda\mu R$. As in proofs of Theorems 1 and 3, the use of triangle inequality yields the key relationship required for this proof

$$\mathbb{E}\left[\left\|\hat{\mathbf{X}}_{t+1} - \mathbf{X}_{t+1}^\star\right\|_F \mid \mathcal{F}_t\right]$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2695450, IEEE Transactions on Signal Processing

$$\leq (1-\beta\rho)\left\|\hat{\mathbf{X}}_t - \mathbf{X}_t^\star\right\|_F + \alpha + \lambda\mu R + \mu\sigma \qquad (97)$$

Consequently, it follows from Lemma 1 that

$$\min_{t_0\leq t\leq t_0+T}\left\|\hat{\mathbf{X}}_t - \mathbf{X}_t^\star\right\| \leq \psi_{\text{plms}} + \eta \qquad (98)$$

with probability one and $\mathbb{E}\left[T|\mathcal{F}_{t_0}\right] \leq \frac{1}{\eta}\left\|\hat{\mathbf{X}}_{t_0} - \mathbf{X}_{t_0}^\star\right\|_F$ for a given $\eta > 0$. Likewise, taking expectations on both sides of (97), we have that

$$\begin{aligned}
&\mathbb{E}\left[\left\|\hat{\mathbf{X}}_{t+1} - \mathbf{X}_{t+1}^\star\right\|_F\right] \\
&\leq (1-\beta\rho)\mathbb{E}\left[\left\|\hat{\mathbf{X}}_t - \mathbf{X}_t^\star\right\|_F\right] + \mu\sigma + \alpha \qquad (99)\\
&\leq (1-\beta\rho)^{t-t_0}\mathbb{E}\left[\left\|\hat{\mathbf{X}}_{t_0} - \mathbf{X}_{t_0}^\star\right\|_F\right] + \psi_{\text{plms}}. \qquad (100)
\end{aligned}$$

The final bounds in Theorem 4 follow by substituting $t_0 = 1$. ■

REFERENCES

[1] B. Widrow and S. Stearns, *Adaptive Signal Processing*. NJ: Prentice Hall, 1985.

[2] Y. Chen, Y. Gu, and A. O. Hero, "Regularized least-mean-square algorithms," *arXiv preprint*, 2010. [Online]. Available: http://arxiv.org/abs/1012.5066

[3] Y. Chen, Y. Gu, and A. O. Hero III, "Sparse LMS for system identification," in *Proc. of the IEEE ICASSP*, Taipei, Taiwan, 2009, pp. 3125–28.

[4] S. Theodoridis, Y. Kopsinis, K. Slavakis, and S. Chouvardas, "Sparsity-aware adaptive learning: A set theoretic estimation approach," in *Proc. of the ALCOSP*, Caen, France, 2013, pp. 748–756.

[5] K. Slavakis and I. Yamada, "The adaptive projected subgradient method constrained by families of quasi-nonexpansive mappings and its application to online learning," *SIAM Journal on Optimization*, vol. 23, pp. 126–152, 2013.

[6] T. Hu and D. B. Chklovskii, "Sparse lms via online linearized bregman iteration," in *Proc. of IEEE ICASSP*, Florence, Italy, 2014.

[7] E. J. Candes and Y. Plan, "Matrix completion with noise," *Proc. IEEE*, vol. 98, pp. 925–936, 2010.

[8] K. Lee and Y. Bresler, "AD-MiRA: Atomic decomposition for minimum rank approximation," *IEEE Trans. Inf. Theory*, vol. 56, no. 9, pp. 4402–16, 2010.

[9] Y. Amit, M. Fink, N. Srebro, and S. Ullman, "Uncovering shared structures in multiclass classification," in *Proc. of IEEE MEMS*, Nagoya, Japan, 1997, pp. 290–294.

[10] A. Argyriou, T. Evgeniou, and M. Pontil, "Convex multi-task feature learning," *Machine Learning*, vol. 73, pp. 243–272, 2008.

[11] B. Recht, M. Fazel, and P. A. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," *SIAM review*, vol. 52, pp. 471–501, 2010.

[12] J. Wright, A. Ganesh, S. Rao, Y. Peng, and Y. Ma, "Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization," in *Advances in Neural Information Processing Systems*, Vancouver, 2009, pp. 2080–2088.

[13] M. Fazel, H. Hindi, and S. P. Boyd, "A rank minimization heuristic with application to minimum order system approximation," in *Proc. of IEEE ACC*, vol. 6, Arlington, VA, USA, 2001, pp. 4734–4739.

[14] E. J. Candès and T. Tao, "The power of convex relaxation: Near-optimal matrix completion," *IEEE Trans. Inf. Theory*, vol. 56, no. 5, pp. 2053–80, 2010.

[15] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Foundations of Computational mathematics*, vol. 9, pp. 717–772, 2009.

[16] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on Optimization*, vol. 20, pp. 1956–1982, 2010.

[17] N. Boumal and P.-A. Absil, "RTRMC: A riemannian trust-region method for low-rank matrix completion," in *NIPS*, Granada, Spain, 2011, pp. 406–414.

[18] Y. Hu, D. Zhang, J. Ye, X. Li, and X. He, "Fast and accurate matrix completion via truncated nuclear norm regularization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, pp. 2117–2130, 2013.

[19] C. Dhanjal, R. Gaudel, and S. Clemencon, "Online matrix completion through nuclear norm regularisation," in *Proc. of the Intl. Conf. on Data Mining*, Philadelphia, PA, USA, April 2014, pp. 623–631.

[20] H. Ji, C. Liu, Z. Shen, and Y. Xu, "Robust video denoising using low rank matrix completion," in *Proc. of IEEE CVPR*, San Fransisco, 2010, pp. 1791–1798.

[21] W. Cong, Z. Feng-li, and Y. Xiao-xiang, "ADeMaC: An adaptive decentralized network latency matrix completion algorithm," in *Proc. of IEEE ICCP*, Cambridge, USA, 2013, pp. 374–377.

[22] K. Rajawat, E. Dall'Anese, and G. Giannakis, "Dynamic network delay cartography," *IEEE Trans. Inf. Theory*, vol. 60, no. 5, pp. 2910–20, 2014.

[23] Y. Liao, W. Du, P. Geurts, and G. Leduc, "DMFSGD: A decentralized matrix factorization algorithm for network distance prediction," *IEEE/ACM Trans. Netw.*, vol. 21, pp. 1511–1524, 2013.

[24] I. Yamada, S. Gandy, and M. Yamagishi, "Sparsity-aware adaptive filtering based on a douglas-rachford splitting," in *EUSIPCO-2011*, Barcelona,Spain, 2011, pp. 1929–1933.

[25] R. G. Baraniuk, "Compressive sensing," *IEEE Signal Process. Lett.*, vol. 24, pp. 118–124, 2007.

[26] O. Taheri and S. A. Vorobyov, "Reweighted ℓ_1-norm penalized LMS for sparse channel estimation and its analysis," *Signal Processing*, vol. 104, pp. 70–79, 2014.

[27] M. Ghazal, A. Amer, and A. Ghrayeb, "Norm constraint LMS algorithm for sparse system identification," *IEEE Signal Process. Lett.*, vol. 16, pp. 774–777, 2009.

[28] G. Gui and F. Adachi, "Improved least mean square algorithm with application to adaptive sparse channel estimation," *EURASIP*, vol. 20, pp. 1–18, Nov. 2013.

[29] S. Chouvardas, K. Slavakis, S. Theodoridis, and I. Yamada, "Stochastic analysis of hyperslab-based adaptive projected subgradient method under bounded noise," *IEEE Signal Process. Lett.*, vol. 20, pp. 729–732, 2013.

[30] K. Slavakis, Y. Kopsinis, and S. Theodoridis, "Adaptive algorithm for sparse system identification using projections onto weighted ℓ_1 balls," in *Proc. of IEEE ICASSP*, Dalas,Texas,USA, 2010, pp. 3742–3745.

[31] S. Ma, D. Goldfarb, and L. Chen, "Fixed point and bregman iterative methods for matrix rank minimization," *Mathematical Programming*, vol. 128, pp. 321–353, 2011.

[32] K.-C. Toh and S. Yun, "An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems," *Pacific Journal of Optimization*, vol. 6, pp. 615–640, 2010.

[33] P. Jain, R. Meka, and I. S. Dhillon, "Guaranteed rank minimization via singular value projection," in *Advances in Neural Information Processing Systems*, Vancouver, 2010, pp. 937–945.

[34] Y. Xu, W. Yin, Z. Wen, and Y. Zhang, "An alternating direction algorithm for matrix completion with nonnegative factors," *Frontiers of Mathematics in China*, vol. 7, no. 2, pp. 365–384, 2012.

[35] B. Recht and C. Ré, "Parallel stochastic gradient algorithms for large-scale matrix completion," *Mathematical Programming Computation*, vol. 5, pp. 201–226, 2013.

[36] H. Avron, S. Kale, V. Sindhwani, and S. P. Kasiviswanathan, "Efficient and practical stochastic subgradient descent for nuclear norm regularization," in *Proc. of ICML*, 2012, pp. 1231–1238.

[37] Y. Xu and W. Yin, "A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion," *SIAM Journal on imaging sciences*, vol. 6, pp. 1758–1789, 2013.

[38] W. Dai, E. Kerman, and O. Milenkovic, "A geometric approach to low-rank matrix completion," *IEEE Trans. Inf. Theory*, vol. 58, no. 1, pp. 237–47, 2012.

[39] R. H. Keshavan, A. Montanari, and S. Oh, "Matrix completion from a few entries," *IEEE Trans. Inf. Theory*, vol. 56, no. 6, pp. 2980–98, 2010.

[40] L. Balzano, R. Nowak, and B. Recht, "Online identification and tracking of subspaces from highly incomplete information," in *Proc. of IEEE Allerton*, Monticello, USA, 2010, pp. 704–711.

[41] J. He, L. Balzano, and A. Szlam, "Incremental gradient on the grassmannian for online foreground and background separation in subsampled video," in *Proc. of IEEE CVPR*, 2012, pp. 1568–1575.

[42] A. Nedic and D. P. Bertsekas, "Incremental subgradient methods for nondifferentiable optimization," *SIAM Journal on Optimization*, vol. 12, no. 1, pp. 109–138, 2001.

[43] A. Simonetto and G. Leus, "On non-differentiable time-varying optimization," in *Proc. of IEEE CAMSAP*, 2015, pp. 505–508.

[44] A. Koppel, A. Simonetto, A. Mokhtari, G. Leus, and A. Ribeiro, "Target tracking with dynamic convex optimization," in *Proc. of IEEE GlobalSIP*, 2015, pp. 1210–1214.

[45] J. A. Costa, N. Patwari, and A. O. Hero III, "Distributed weighted-multidimensional scaling for node localization in sensor networks," *ACM Transactions on Sensor Networks*, vol. 2, no. 1, pp. 39–64, 2006.

[46] A. Simonetto and G. Leus, "Distributed maximum likelihood sensor network localization," *IEEE Trans. on Signal Processing*, vol. 62, no. 6, pp. 1424–1437, 2014.

[47] S. Kumar, R. Jain, and K. Rajawat, "Asynchronous optimization over heterogeneous networks via consensus admm," *IEEE Trans. on Signal and Inf. Proc. over Networks*, 2016 (to be published).

[48] S. Xi, M. D. Zoltowski, and L. Dong, "Iterative mmse cooperative localization with incomplete pair-wise range measurements," in *Proc. of SPIE*, vol. 7706, 2010, pp. 0F1–0F8.

[49] H. Jamali-Rad and G. Leus, "Dynamic multidimensional scaling for low-complexity mobile network tracking," *IEEE Trans. on Signal Processing*, vol. 60, no. 8, pp. 4485–4491, 2012.

[50] K. P. Gummadi, S. Saroiu, and S. D. Gribble, "King: Estimating latency between arbitrary internet end hosts," in *Proc. of the ACM SIGCOMM Workshop on Internet Measurement*. ACM, 2002, pp. 5–18.

[51] P. Francis, S. Jamin, C. Jin, Y. Jin, D. Raz, Y. Shavitt, and L. Zhang, "Idmaps: A global internet host distance estimation service," *IEEE/ACM Trans. On Networking*, vol. 9, no. 5, pp. 525–540, 2001.

[52] F. Dabek, R. Cox, F. Kaashoek, and R. Morris, "Vivaldi: A decentralized network coordinate system," in *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 4. ACM, 2004, pp. 15–26.

[53] G. Mateos and K. Rajawat, "Dynamic network cartography: Advances in network health monitoring," *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 129–143, 2013.

[54] M. Coates, Y. Pointurier, and M. Rabbat, "Compressed network monitoring," in *IEEE Workshop on Statistical Signal Processing*, 2007, pp. 418–422.

[55] I. Djurović, "Bm3d filter in salt-and-pepper noise removal," *EURASIP Journal on Image and Video Processing*, vol. 2016, no. 1, pp. 1–11, 2016.

[56] Y. Zhang, Y. Liu, X. Li, and C. Zhang, "Salt and pepper noise removal in surveillance video based on low-rank matrix recovery," *Computational Visual Media*, vol. 1, no. 1, pp. 59–68, 2015.

[57] A. H. Sayed, *Fundamentals of adaptive filtering*. John Wiley & Sons, 2003.

[58] G. A. Watson, "Characterization of the subdifferential of some matrix norms," *Linear Algebra and its Applications*, vol. 170, pp. 33–45, 1992.

[59] P. L. Combettes and V. R. Wajs, "Signal recovery by proximal forward-backward splitting," *Multiscale Modeling & Simulation*, vol. 4, pp. 1168–1200, 2005.

[60] Y. Vardi, "Network tomography: Estimating source-destination traffic intensities from link data," *Journal of the American Statistical Association*, vol. 91, pp. 365–377, 1996.

[61] https://ipnetwork.bgtmo.ip.att.net/pws/network_delay.html.

[62] http://www.keynote.com/resources/internet-health-report.

[63] D. P. Bertsekas, "Incremental proximal methods for large scale convex optimization," *Mathematical programming*, vol. 129, no. 2, pp. 163–195, 2011.

[64] ——, *Convex optimization theory*. Belmont, MA: Athena Scientific, 2009.

[65] L. Rosasco, S. Villa, and B. C. Vũ, "Convergence of stochastic proximal gradient algorithm," *arXiv preprint*, 2014. [Online]. Available: http://arxiv.org/abs/1403.5074

[66] T. Chen and H. R. Wu, "Adaptive impulse detection using center-weighted median filters," *IEEE Signal Processing Letters*, vol. 8, no. 1, pp. 1–3, 2001.

[67] H. Hwang and R. A. Haddad, "Adaptive median filters: new algorithms and results," *IEEE Transactions on image processing*, vol. 4, no. 4, pp. 499–502, 1995.

[68] V. Solo and X. Kong, *Adaptive signal processing algorithms: stability and performance*. Prentice-Hall, Inc., 1994.

[69] F. H. Clarke, *Optimization and nonsmooth analysis*. Siam, 1990.

[70] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[71] L. Bottou, "Online learning and stochastic approximations," *On-line learning in neural networks*, vol. 17, p. 25, 1998.

[72] Y. Wang, J. Yang, W. Yin, and Y. Zhang, "A new alternating minimization algorithm for total variation image reconstruction," *SIAM Journal on Imaging Sciences*, vol. 1, pp. 248–272, 2008.

[73] J. Yang and Y. Zhang, "Alternating direction algorithms for ℓ_1-problems in compressive sensing," *SIAM journal on scientific computing*, vol. 33, pp. 250–278, 2011.

[74] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis," *Journal of the ACM (JACM)*, vol. 58, p. 11, 2011.

[75] M. Ghazal, A. Amer, and A. Ghrayeb, "A real-time technique for spatio–temporal video noise estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, pp. 1690–1699, 2007.

[76] D. Slock, "On the convergence behavior of the LMS and the normalized lms algorithms," *IEEE Signal Process. Lett.*, vol. 41, pp. 2811–2825, 1993.

[77] M. Fazel, "Matrix rank minimization with applications," Ph.D. dissertation, Stanford University, 2002.

[78] H. Ghasemi, M. Malek-Mohammadi, M. Babaie-Zadeh, and C. Jutten, "SRF: Matrix completion based on smoothed rank function," *IEEE Audio, Speech, Language Process.*, vol. 20, pp. 3672–3675, 2011.

[79] H. Avron, S. Kale, S. Kasiviswanathan, and V. Sindhwani, "Efficient and practical stochastic subgradient descent for nuclear norm regularization(full version)," *arXiv preprint arXiv:1206.6384*, 2012.

[80] Z. Lin, M. Chen, and Y. Ma, "The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices," *arXiv preprint arXiv:1009.5055*, 2010.

[81] J. Cai and S. Osher, "Fast singular value thresholding without singular value decomposition," *UCLA CAM Report*, vol. 5, 2010.

[82] R. Mazumder, T. Hastie, and R. Tibshirani, "Spectral regularization algorithms for learning large incomplete matrices," *The Journal of Machine Learning Research*, vol. 11, pp. 2287–2322, 2010.

[83] W. G.A., "Characterization of the subdifferential of some matrix norms," *The Journal of Machine Learning Research*, vol. 11, pp. 33 – 45, 1992.

[84] J. He, L. Balzano, and J. Lui, "Online robust subspace tracking from partial information," *arXiv preprint arXiv:1109.3827*, 2011.

[85] A. Ribeiro, "Ergodic stochastic optimization algorithms for wireless communication and networking," *IEEE Signal Process. Lett.*, vol. 58, pp. 6369–6386, 2010.

[86] Z.-F. Jin, Z. Wan, X. Zhao, and Y. Xiao, "A penalty decomposition method for rank minimization problem with affine constraints," *Applied Mathematical Modelling*, vol. 39, no. 16, pp. 4859 – 4870, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0307904X15002486

[87] D. B. Chua, E. D. Kolaczyk, and M. Crovella, "Network kriging," *IEEE J. Sel. Areas Commun.*, vol. 24, pp. 2263–2272, 2006.

[88] H. Xu, C. Caramanis, and S. Sanghavi, "Robust pca via outlier pursuit," in *Advances in Neural Information Processing Systems*, Vancouver, 2010, pp. 2496–2504.

[89] G. Liu, Z. Lin, and Y. Yu, "Robust subspace segmentation by low-rank representation," in *Proc. of ICML*, Haifa, Israel, 2010, pp. 663–670.

[90] S. S. Ram, A. Nedic, and V. V. Veeravalli, "Incremental stochastic subgradient algorithms for convex optimization," *SIAM J. on Optimization*, vol. 20, no. 2, pp. 691–717, 2009.

[91] F. K. W. Chan, H.-C. So, and W.-K. Ma, "A novel subspace approach for cooperative localization in wireless sensor networks using range measurements," *IEEE Trans. on Signal Processing*, vol. 57, no. 1, pp. 260–269, 2009.

[92] Y. Liao, "Learning to predict end-to-end network performance," Ph.D. dissertation, University of Liege, Liege, Belgium, 2013.

[93] S. Wu, Y. Chen, X. Fu, and J. Li, "Ncshield: securing decentralized, matrix factorization-based network coordinate systems," in *IEEE Intl. Workshop on Quality of Service*. IEEE, 2012, pp. 1–9.