Multi-Step Stochastic ADMM in High Dimensions: Applications to Sparse Optimization and Noisy Matrix Decomposition

Hanie Sedghi[*] Anima Anandkumar[†] Edmond Jonckheere [‡]

July 7, 2015

Abstract

We propose an efficient ADMM method with guarantees for high-dimensional problems. We provide explicit bounds for the sparse optimization problem and the noisy matrix decomposition problem. For sparse optimization, we establish that the modified ADMM method has an optimal convergence rate of $\mathcal{O}(s\log d/T)$, where s is the sparsity level, d is the data dimension and T is the number of steps. This matches with the minimax lower bounds for sparse estimation. For matrix decomposition into sparse and low rank components, we provide the first guarantees for any online method, and prove a convergence rate of $\tilde{\mathcal{O}}((s+r)\beta^2(p)/T)+\mathcal{O}(1/p)$ for a $p\times p$ matrix, where s is the sparsity level, r is the rank and $\Theta(\sqrt{p})\leq\beta(p)\leq\Theta(p)$. Our guarantees match the minimax lower bound with respect to s, r and T. In addition, we match the minimax lower bound with respect to the matrix dimension p, i.e. $\beta(p)=\Theta(\sqrt{p})$, for many important statistical models including the independent noise model, the linear Bayesian network and the latent Gaussian graphical model under some conditions. Our ADMM method is based on epoch-based annealing and consists of inexpensive steps which involve projections on to simple norm balls. Experiments show that for both sparse optimization and matrix decomposition problems, our algorithm outperforms the state-of-the-art methods. In particular, we reach higher accuracy with same time complexity.

Keywords: Stochastic ADMM, ℓ_1 regularization, multi block ADMM, sparse+low rank decomposition, convergence rate, high dimensional regime.

1 Introduction

Stochastic optimization techniques have been extensively employed for online machine learning on data which is uncertain, noisy or missing. Typically it involves performing a large number of inexpensive iterative updates, making it scalable for large-scale learning. In contrast, traditional batch-based techniques involve far more expensive operations for each update step. Stochastic optimization has been analyzed in a number of recent works, e.g., (Shalev-Shwartz, 2011; Boyd et al., 2011;

[*] University of Southern California, Email: hsedghi@usc.edu

[†] University of California, Irvine, Email: a.anandkumar@uci.edu

[‡] University of Southern California, Email: jonckhee@usc.edu

Agarwal et al., 2012b; Wang et al., 2013a; Johnson and Zhang, 2013; Shalev-Shwartz and Zhang, 2013).

The alternating direction method of multipliers (ADMM) is a popular method for online and distributed optimization on a large scale (Boyd et al., 2011), and is employed in many applications, e.g., (Wahlberg et al., 2012), (Esser et al., 2010), (Mota et al., 2012). It can be viewed as a decomposition procedure where solutions to sub-problems are found locally, and coordinated via constraints to find the global solution. Specifically, it is a form of augmented Lagrangian method which applies partial updates to the dual variables. ADMM is often applied to solve regularized problems, where the function optimization and regularization can be carried out locally, and then coordinated globally via constraints. Regularized optimization problems are especially relevant in the high dimensional regime since regularization is a natural mechanism to overcome ill-posedness and to encourage parsimony in the optimal solution, e.g., sparsity and low rank. Due to the efficiency of ADMM in solving regularized problems, we employ it in this paper.

In this paper, we design a modified version of the stochastic ADMM method for high-dimensional problems. We first analyze the simple setting, where the optimization problem consists of a loss function and a single regularizer, and then extend to the multi-block setting with multiple regularizers and multiple variables. For illustrative purposes, for the first setting, we consider the sparse optimization problem and for the second setting, the matrix decomposition problem respectively. Note that our results easily extend to other settings, e.g., those in Negahban et al. (2012).

We consider a simple modification to the (inexact) stochastic ADMM method (Ouyang et al., 2013) by incorporating multiple steps or epochs, which can be viewed as a form of annealing. We establish that this simple modification has huge implications in achieving tight convergence rates as the dimensions of the problem instances scale. In each iteration of the method, we employ projections on to certain norm balls of appropriate radii, and we decrease the radii in epochs over time. The idea of annealing was first introduced by Agarwal et al. (2012b) for dual averaging. Yet, that method cannot be extended for multivariable cases.

For instance, for the sparse optimization problem, we constrain the optimal solution at each step to be within an ℓ_1-norm ball of the initial estimate, obtained at the beginning of each epoch. At the end of the epoch, an average is computed and passed on to the next epoch as its initial estimate. Note that the ℓ_1 projection can be solved efficiently in linear time, and can also be parallelized easily (Duchi et al., 2008).

For matrix decomposition with a general loss function, the ADMM method requires multiple blocks for updating the low rank and sparse components. We apply the same principle and project the sparse and low rank estimates on to ℓ_1 and nuclear norm balls, and these projections can be computed efficiently.

Theoretical implications: The above simple modifications to ADMM have huge implications for high-dimensional problems. For sparse optimization, our convergence rate is $\mathcal{O}(\frac{s \log d}{T})$, for s-sparse problems in d dimensions in T steps. Our bound has the best of both worlds: efficient high-dimensional scaling (as $\log d$) and efficient convergence rate (as $\frac{1}{T}$). This also matches the minimax lower bound for the linear model and square loss function (Raskutti et al., 2011), which implies that our guarantee is unimprovable by any (batch or online) algorithm (up to constant factors). For matrix decomposition, our convergence rate is $\mathcal{O}((s+r)\beta^2(p)\log p/T)) + \mathcal{O}(\max\{s+r, p\}/p^2)$ for a $p \times p$ input matrix in T steps, where the sparse part has s non-zero entries and low rank part has rank r. For many natural noise models (e.g. independent noise, linear Bayesian networks),

$\beta^2(p) = p$, and the resulting convergence rate is minimax-optimal. Note that our bound is not only on the reconstruction error, but also on the error in recovering the sparse and low rank components. These are the first convergence guarantees for online matrix decomposition in high dimensions. Moreover, our convergence rate holds *with high probability* when noisy samples are input, in contrast to expected convergence rate, typically analyzed in literature. See Table 1, 2 for comparison of this work with related frameworks.

Practical implications: The proposed algorithms provide significantly faster convergence in high dimension and better robustness to noise. For sparse optimization, our method has significantly better accuracy compared to the stochastic ADMM method and better performance than RADAR, based on multi-step dual averaging (Agarwal et al., 2012b). For matrix decomposition, we compare our method with the state-of-art inexact ALM (Lin et al., 2010) method. While both methods have similar reconstruction performance, our method has significantly better accuracy in recovering the sparse and low rank components.

Related Work: ADMM: Existing online ADMM-based methods lack high-dimensional guarantees. They scale poorly with the data dimension (as $\mathcal{O}(d^2)$), and also have slow convergence for general problems (as $\mathcal{O}(\frac{1}{\sqrt{T}})$). Under strong convexity, the convergence rate can be improved to $\mathcal{O}(\frac{1}{T})$ but only in *expectation*: such analyses ignore the per sample error and consider only the expected convergence rate (see Table 1). In contrast, our bounds hold with high probability. Some stochastic ADMM methods, Goldstein et al. (2012), Deng (2012) and Luo (2012), provide faster rates for stochastic ADMM, than the rate noted in Table 1. However, they require strong conditions which are not satisfied for the optimization problems considered here, e.g., Goldstein et al. (2012) require both the loss function and the regularizer to be strongly convex.

It is also worth mentioning that our method provides error contraction, i.e., we can show error shrinkage after specific number of iterations whereas no other ADMM based method can guarantee this.

Related Work: Sparse Optimization: For the sparse optimization problem, ℓ_1 regularization is employed and the underlying true parameter is assumed to be sparse. This is a well-studied problem in a number of works (for details, refer to (Agarwal et al., 2012b)). Agarwal et al. (2012b) propose an efficient online method based on annealing dual averaging, which achieves the same optimal rates as the ones derived in this paper. The main difference is that our ADMM method is capable of solving the problem for multiple random variables and multiple conditions while their method cannot incorporate these extensions.

Related Work: Matrix Decomposition: To the best of our knowledge, online guarantees for high-dimensional matrix decomposition have not been provided before. Wang et al. (2013b) propose a multi-block ADMM method for the matrix decomposition problem but only provide convergence rate analysis in expectation and it has poor high dimensional scaling (as $\mathcal{O}(p^4)$ for a $p \times p$ matrix) without further modifications. Note that they only provide convergence rate on difference between loss function and optimal loss, whereas we provide the convergence rate on individual errors of the sparse and low rank components $\|\bar{S}(T) - S^*\|_{\mathbb{F}}^2$, $\|\bar{L}(T) - L^*\|_{\mathbb{F}}^2$. See Table 2 for comparison of guarantees for matrix decomposition problem.

Method	Assumptions	convergence
ST-ADMM (Ouyang et al., 2013)	L, convexity	$\mathcal{O}(d^2/\sqrt{T})$
ST-ADMM (Ouyang et al., 2013)	SC, E	$\mathcal{O}(d^2 \log T/T)$
BADMM (Wang and Banerjee, 2013)	convexity, E	$\mathcal{O}(d^2/\sqrt{T})$
RADAR (Agarwal et al., 2012b)	LSC, LL	$\mathcal{O}(s \log d/T)$
REASON 1 (this paper)	LSC, LL	$\mathcal{O}(s \log d/T)$
Minimax bound (Raskutti et al., 2011)	Eigenvalue conditions	$\mathcal{O}(s \log d/T)$

Table 1: *Comparison of online sparse optimization methods under s sparsity level for the optimal paramter, d dimensional space, and T number of iterations.*
SC = Strong Convexity, LSC = Local Strong Convexity, LL = Local Lipschitz, L = Lipschitz property, E = in Expectation
The last row provides minimax-optimal rate on error for any method. The results hold with high probability unless otherwise mentioned.

We compare our guarantees in the online setting with the batch guarantees of Agarwal et al. (2012a). Although other batch analyses exist for matrix decomposition, e.g., (Chandrasekaran et al., 2011; Candès et al., 2011; Hsu et al., 2011), they require stronger assumptions based on incoherence conditions for recovery, which we do not impose here. The batch analysis by Agarwal et al. (2012a) requires fairly mild condition such as "diffusivity" of the unknown low rank matrix. Moreover, the convergence rate for the batch setting by Agarwal et al. (2012a) achieves the minimax lower bound (under the independent noise model), and is thus, optimal, up to constant factors.

Note that when only the weak diffusivity condition is assumed, the matrix decomposition problem suffers from an approximation error, i.e. an error even in the noiseless setting. Both the minimax rate and the batch rates in (Agarwal et al., 2012a) have an approximation error. However, our approximation error is worse by a factor of p, although it is still decaying with respect to p.

Overview of Proof Techniques: Note that in the main text, we provide guarantees for fixed-epoch length. However, if we use variable-length epoch size we can get a $\log d$ improvement in the convergence rate. Our proof involves the following high-level steps to establish the convergence rate: (1) deriving convergence rate for the modified ADMM method (with variable-length epoch size) at the end of one epoch, where the ADMM estimate is compared with the batch estimate, (2) comparing the batch estimate with the true parameter, and then combining the two steps, and analyzing over multiple epochs to obtain the final bound. We can show that with the proposed parameter setting and varying epoch size, error can be halved by the end of each epoch. For the matrix decomposition problem, additional care is needed to ensure that the errors in estimating the sparse and low rank parts can be decoupled. This is especially non-trivial in our setting since we utilize multiple variables in different blocks which are updated in each iteration. Our careful analysis enables us to establish the first results for online matrix decomposition in the high-dimensional setting which match the batch guarantees for many interesting statistical models. (3) Next, we analyze how guarantees change for fixed epoch length. We prove that although the error halving stops after some iterations but the error does not increase noticeably to invalidate the analysis.

Method	Assumptions	Convergence rate
Multi-block-ADMM (Wang et al., 2013b)	L, SC, E	$\mathcal{O}(p^4/T)$
Batch method (Agarwal et al., 2012a)	LL, LSC, DF	$\mathcal{O}((s\log p + rp)/T) + \mathcal{O}(s/p^2)$
REASON 2 (this paper)	LSC, LL, DF	$\mathcal{O}((s+r)\beta^2(p)\log p/T)) + \mathcal{O}(\max\{s+r,p\}/p^2)$
Minimax bound (Agarwal et al., 2012a)	ℓ_2, IN, DF	$\mathcal{O}((s\log p + rp)/T) + \mathcal{O}(s/p^2)$

Table 2: *Comparison of optimization methods for sparse+low rank matrix decomposition for a $p \times p$ matrix under s sparsity level and r rank matrices and T is the number of samples.*
SC = Strong Convexity, LSC = Local Strong Convexity, LL = Local Lipschitz, L = Lipschitz for loss function, IN = Independent noise model, DF = diffuse low rank matrix under the optimal parameter. $\beta(p) = \Omega(\sqrt{p}), \mathcal{O}(p)$ and its value depends the model. The last row provides minimax-optimal rate on error for any method under the independent noise model. The results hold with high probability unless otherwise mentioned.
For Multi-block-ADMM (Wang et al., 2013b) the convergence rate is on the difference of loss function from optimal loss, for the rest of works in the table, the convergence rate is on $\|\bar{S}(T) - S^*\|_{\mathbb{F}}^2 + \|\bar{L}(T) - L^*\|_{\mathbb{F}}^2$.

1.1 Notation

In the sequel, we use lower case letter for vectors and upper case letter for matrices.$\|x\|_1$, $\|x\|_2$ refer to ℓ_1, ℓ_2 vector norms respectively. The term $\|X\|_*$ stands for nuclear norm of X. In addition, $\|X\|_2$, $\|X\|_{\mathbb{F}}$ denote spectral and Frobenius norms respectively. $|\!|\!|X|\!|\!|_\infty$ stands for induced infinity norm. We use vectorized ℓ_1, ℓ_∞ norm for matrices. i.e., $\|X\|_1 = \sum\limits_{i,j} |X_{ij}|$, $\|X\|_\infty = \max\limits_{i,j} |X_{ij}|$.

2 Problem Formulation

Consider the optimization problem

$$\theta^* \in \arg\min_{\theta \in \Omega} \mathbb{E}[f(\theta, x)], \tag{1}$$

where $x \in \mathbb{X}$ is a random variable and $f : \Omega \times \mathbb{X} \to \mathbb{R}$ is a given loss function. Since only samples are available, we employ the empirical estimate of $\widehat{f}(\theta) := 1/n \sum_{i\in[n]} f(\theta, x_i)$ in the optimization. For high-dimensional θ, we need to impose a regularization $\mathcal{R}(\cdot)$, and

$$\widehat{\theta} := \arg\min\{\widehat{f}(\theta) + \lambda_n \mathcal{R}(\theta)\}, \tag{2}$$

is the batch optimal solution.

For concreteness we focus on the sparse optimization and the matrix decomposition problem. It is straightforward to generalize our results to other settings, say (Negahban et al., 2012). For the first case, the optimum θ^* is a s-sparse solution, and the regularizer is the ℓ_1 norm, and we have

$$\widehat{\theta} = \arg\min\{\widehat{f}(\theta) + \lambda_n \|\theta\|_1\} \tag{3}$$

We also consider the matrix decomposition problem, where the underlying matrix $M^* = S^* + L^*$ is a combination of a sparse matrix S^* and a low rank matrix L^*. Here the unknown parameters are $[S^*; L^*]$, and the regularization $\mathcal{R}(\cdot)$ is a combination of the ℓ_1 norm, and the nuclear norm $\|\cdot\|_*$ on the sparse and low rank parts respectively. The corresponding batch estimate is given by

$$\begin{aligned} \widehat{M} &:= \arg\min\{f(M) + \lambda_n \|S\|_1 + \mu_n \|L\|_*\} \\ s.t. \quad M &= S + L, \quad \|L\|_\infty \leq \frac{\alpha}{p}. \end{aligned} \tag{4}$$

The $\|\cdot\|_\infty$ constraint on the low rank matrix will be discussed in detail later, and it is assumed that the true matrix L^* satisfies this condition.

We consider an online version of the optimization problem where we optimize the program in (2) under each data sample instead of using the empirical estimate of f for an entire batch. We consider an inexact version of the online ADMM method, where we compute the gradient $\hat{g}_i \in \nabla f(\theta, x_i)$ at each step and employ it for optimization. In addition, we consider an epoch based setting, where we constrain the optimal solution to be close to the initial estimate at the beginning of the epoch. This can be viewed as a form of regularization and we constrain more (i.e. constrain the solution to be closer) as time goes by, since we expect to have a sharper estimate of the optimal solution. This limits the search space for the optimal solution and allows us to provide tight guarantees in the high-dimensional regime.

We first consider the simple case of sparse setting in (3), where the ADMM has double blocks,and then extend it to the sparse+low rank setting of (4), which involves multi-block ADMM.

3 ℓ_1 Regularized Stochastic Optimization

We consider the optimization problem $\theta^* \in \arg\min \mathbb{E}[f(\theta, x)]$, $\theta \in \Omega$ where θ^* is a sparse vector. The loss function $f(\theta, x_k)$ is a function of a parameter $\theta \in \mathbb{R}^d$ and samples x_i. In stochastic setting, we do not have access to $\mathbb{E}[f(\theta, x)]$ nor to its subgradients. In each iteration we have access to one noisy sample. In order to impose sparsity we use regularization. Thus we solve a sequence

$$\theta_k \in \underset{\theta \in \Omega'}{\arg\min}\; f(\theta, x_k) + \lambda \|\theta\|_1, \quad \Omega' \subset \Omega, \tag{5}$$

where the regularization parameter $\lambda > 0$ and the constraint sets Ω' change from epoch to epoch.

3.1 Epoch-based Online ADMM Algorithm

We now describe the modified inexact ADMM algorithm for the sparse optimization problem in (5), and refer to it as REASON 1, see Algorithm 1. We consider epochs of length T_0, and in each epoch i, we constrain the optimal solution to be within an ℓ_1 ball with radius R_i centered around $\tilde{\theta}_i$, which is the initial estimate of θ^* at the start of the epoch. The θ-update is given by

$$\theta_{k+1} = \underset{\|\theta - \tilde{\theta}_i\|_1^2 \leq R_i^2}{\arg\min} \{\langle \nabla f(\theta_k), \theta - \theta_k \rangle - \langle z_k, \theta - y_k \rangle + \frac{\rho}{2}\|\theta - y_k\|_2^2 + \frac{\rho_x}{2}\|\theta - \theta_k\|_2^2\}$$

Note that this is an inexact update since we employ the gradient $\nabla f(\cdot)$ rather than optimize directly on the loss function $f(\cdot)$ which is expensive. The above program can be solved efficiently since it is a projection on to the ℓ_1 ball, whose complexity is linear in the sparsity level of the gradient,

Algorithm 1 Regularized Epoch-based Admm for Stochastic Optimization in high-dimensioN 1 (REASON 1)

```
Input $\rho, \rho_x > 0$, epoch length $T_0$ , initial prox center $\tilde{\theta}_1$, initial radius $R_1$, regularization parameter $\{\lambda_i\}_{i=1}^{k_T}$.
Define $Shrink_\kappa(\cdot)$ shrinkage operator in (7)
for Each epoch $i = 1, 2, ..., k_T$ do
    Initialize $\theta_0 = y_0 = \tilde{\theta}_i$
    for Each iteration $k = 0, 1, ..., T_0 - 1$ do
        $\theta_{k+1} = \underset{\|\theta - \tilde{\theta}_i\|_1 \leq R_i}{\arg\min} \{\langle \nabla f(\theta_k), \theta - \theta_k \rangle - \langle z_k, \theta - y_k \rangle + \frac{\rho}{2}\|\theta - y_k\|_2^2 + \frac{\rho_x}{2}\|\theta - \theta_k\|_2^2\}$    (6)
        $y_{k+1} = \text{Shrink}_{\lambda_i/\rho}(\theta_{k+1} - \frac{z_k}{\rho})$
        $z_{k+1} = z_k - \tau(\theta_{k+1} - y_{k+1})$
    end for
    Return : $\overline{\theta}(T_i) := \frac{1}{T}\sum_{k=0}^{T_0-1} \theta_k$ for epoch $i$ and $\tilde{\theta}_{i+1} = \overline{\theta}(T_i)$.
    Update : $R_{i+1}^2 = R_i^2/2$.
end for
```

when performed serially, and $\mathcal{O}(\log d)$ when performed in parallel using d processors (Duchi et al., 2008). For details of θ-update implementation see Appendix E.1.

For the regularizer, we introduce the variable y, and the y-update is

$$y_{k+1} = \arg\min\{\lambda_i\|y_k\|_1 - \langle z_k, \theta_{k+1} - y \rangle + \frac{\rho}{2}\|\theta_{k+1} - y\|_2^2\}$$

This update can be simplified to the form given in REASON 1, where $\text{Shrink}_\kappa(\cdot)$ is the soft-thresholding or shrinkage function (Boyd et al., 2011).

$$\text{Shrink}_\kappa(a) = (a - \kappa)_+ - (-a - \kappa)_+ \tag{7}$$

Thus, each step in the update is extremely simple to implement. When an epoch is complete, we carry over the average $\overline{\theta}(T_i)$ as the next epoch center and reset the other variables.

3.2 High-dimensional Guarantees

We now provide convergence guarantees for the proposed method under the following assumptions.

Assumption A1: Local strong convexity (LSC) : The function $f : S \to \mathbb{R}$ satisfies an R-local form of strong convexity (LSC) if there is a non-negative constant $\gamma = \gamma(R)$ such that

$$f(\theta_1) \geq f(\theta_2) + \langle \nabla f(\theta_2), \theta_1 - \theta_2 \rangle + \frac{\gamma}{2}\|\theta_2 - \theta_1\|_2^2.$$

for any $\theta_1, \theta_2 \in S$ with $\|\theta_1\|_1 \leq R$ and $\|\theta_2\|_1 \leq R$.

Note that the notion of strong convexity leads to faster convergence rates in general. Intuitively, strong convexity is a measure of curvature of the loss function, which relates the reduction in the loss function to closeness in the variable domain. Assuming that the function f is twice continuously differentiable, it is strongly convex, if and only if its Hessian is positive semi-definite, for all feasible θ. However, in the high-dimensional regime, where there are fewer samples than data dimension, the Hessian matrix is often singular and we do not have global strong convexity. A solution is to impose local strong convexity which allows us to provide guarantees for high dimensional problems. The notion of local strong convexity has been exploited before in a number of works on high dimensional analysis, e.g., (Negahban et al., 2012; Agarwal et al., 2012a,b).

Assumption A2: Sub-Gaussian stochastic gradients: Let $e_k(\theta) := \nabla f(\theta, x_k) - \mathbb{E}[\nabla f(\theta, x_k)]$. For all θ such that $\|\theta - \theta^*\|_1 \leq R$, there is a constant $\sigma = \sigma(R)$ such that for all $k > 0$,

$$\mathbb{E}[\exp(\|e_k(\theta)\|_\infty^2)/\sigma^2] \leq \exp(1)$$

Remark: The bound holds with $\sigma = \mathcal{O}(\sqrt{\log d})$ whenever each component of the error vector has sub-Gaussian tails (Agarwal et al., 2012b).

Assumption A3: Local Lipschitz condition: For each $R > 0$, there is a constant $G = G(R)$ such that

$$|f(\theta_1) - f(\theta_2)| \leq G\|\theta_1 - \theta_2\|_1 \tag{8}$$

for all $\theta_1, \theta_2 \in S$ such that $\|\theta - \theta^*\|_1 \leq R$ and $\|\theta_1 - \theta^*\|_1 \leq R$.

We choose the algorithm parameters as below where λ_i is the regularization for ℓ_1 term, ρ and ρ_x are penalties in θ-update as in (6) and τ is the step size for the dual update.

$$\lambda_i^2 = \frac{\gamma}{s\sqrt{T_0}}\sqrt{R_i^2 \log d + \frac{G^2 R_i^2}{T_0} + \sigma_i^2 R_i^2 w_i^2} \tag{9}$$

$$\rho \propto \frac{\sqrt{T_0 \log d}}{R_i}, \quad \rho_x > 0, \quad \tau = \rho.$$

Theorem 1. *Under Assumptions A1 − A3, λ_i as in (9) , we use fixed epoch length $T_0 = T\log d/k_T$ where T is the total number of iterations. Assuming this setting ensures $T_0 = \mathcal{O}(\log d)$, for any θ^* with sparsity s, we have*

$$\|\bar{\theta}_T - \theta^*\|_2^2 = \mathcal{O}\left(s \;\frac{\log d + (w^2 + \log(k_T/\log d))\sigma^2}{T}\;\frac{\log d}{k_T}\right),$$

with probability at least $1 - 3\exp(w^2/12)$, where $\bar{\theta}_T$ is the average for the last epoch for a total of T iterations and

$$k_T = \log_2 \frac{\gamma^2 R_1^2 T}{s^2(\log d + 12\sigma^2 w^2)}.$$

For proof, see Appendix B.6.

Improvement of $\log d$ factor : The above theorem covers the practical case where the epoch length T_0 is fixed. We can improve the above results using varying epoch lengths (which depend on the problem parameters) such that $\|\bar{\theta}_T - \theta^*\|_2^2 = \mathcal{O}(s\log d/T)$. See Theorem 3 in Appendix A.

Optimal Guarantees: The above results indicate a convergence rate of $\mathcal{O}(s \log d/T)$ which matches the minimax lower bounds for sparse estimation (Raskutti et al., 2011). This implies that our guarantees are *unimprovable* up to constant factors.

Comparison with Agarwal et al. (2012b): The RADAR algorithm proposed by Agarwal et al. (2012b) also achieves a rate of $\mathcal{O}(s \log d/T)$ which matches with ours. The difference is our method is capable of solving problems with multiple variables and constraints, as discussed in the next section, while RADAR cannot be generalized to do so.

Remark on Lipschitz property: In fact, our method requires a weaker condition than local Lipschitz property. We only require the following bounds on the dual variable: $\|z_{k+1} - z_k\|_1$ and $\|z_k\|_\infty$. Both these are upper bounded by $G + 2(\rho_x + \rho)R_i$. In addition the ℓ_1 constraint does not influence the bound on the dual variable. For details see Section B.1.

Remark on need for ℓ_1 constraint: We use ℓ_1 constraint in the θ-update step, while the usual ADMM method does not have such a constraint. The ℓ_1 constraint allows us to provide efficient high dimensional scaling (as $\mathcal{O}(\log d)$). Specifically, this is because one of the terms in our convergence rate consists of $\langle e_k, \theta_k - \hat{\theta}_i \rangle$, where e_k is the error in the gradient (see Appendix B.2). We can use the inequality

$$\langle e_k, \theta_k - \hat{\theta}_i \rangle \leq \|e_k\|_\infty \|\theta_k - \hat{\theta}_i\|_1.$$

From Assumption A2, we have a bound on $\|e_k\|_\infty = \mathcal{O}(\log d)$, and by imposing the ℓ_1 constraint, we also have a bound on the second term, and thus, we have an efficient convergence rate. If instead ℓ_p penalty is imposed for some p, the error scales as $\|e(\theta)\|_q^2$, where ℓ_q is the dual norm of ℓ_p. For instance, if $p = 2$, we have $q = 2$, and the error can be as high as $\mathcal{O}(d/T)$ since $\|e(\theta)\|_2^2 \leq d\sigma$. Note that for the ℓ_1 norm, we have ℓ_∞ as the dual norm, and $\|e(\theta)\|_\infty \leq \sigma = \mathcal{O}(\sqrt{\log d})$ which leads to optimal convergence rate in the above theorem. Moreover, this ℓ_1 constraint can be efficiently implemented, as discussed in Section 3.1.

4 Extension to Doubly Regularized Stochastic Optimization

We now consider the problem of matrix decomposition into a sparse matrix $S \in \mathbb{R}^{p \times p}$ and a low rank matrix $L \in \mathbb{R}^{p \times p}$ based on the loss function f on $M = S + L$. The batch program is given in Equation (4) and we now design an online program based on multi-block ADMM algorithm, where the updates for M, S, L are carried out independently.

In the stochastic setting, we consider the optimization problem $M^* \in \arg\min \mathbb{E}[f(M, X)]$, where we want to decompose M into a sparse matrix $S \in \mathbb{R}^{p \times p}$ and a low rank matrix $L \in \mathbb{R}^{p \times p}$. $f(M, X_k)$ is a function of parameter M and samples X_k. X_k can be a matrix (e.g. independent noise model) or a vector (e.g. Gaussian graphical model). In stochastic setting, we do not have access to $\mathbb{E}[f(M, X)]$ nor to its subgradients. In each iteration we have access to one noisy sample and update our estimate based on that. We impose the desired properties with regularization. Thus, we solve a sequence

$$M_k := \arg\min\{\widehat{f}(M, X_k) + \lambda\|S\|_1 + \mu\|L\|_*\} \qquad s.t. \quad M = S + L, \quad \|L\|_\infty \leq \frac{\alpha}{p}. \tag{10}$$

4.1 Epoch-based Multi-Block ADMM Algorithm

We now extend the ADMM method proposed in REASON 1 to multi-block ADMM. The details are in Algorithm 2, and we refer to it as REASON 2. Recall that the matrix decomposition setting assumes that the true matrix $M^* = S^* + L^*$ is a combination of a sparse matrix S^* and a low rank matrix L^*. In REASON 2, the updates for matrices M, S, L are done independently at each step.

For the M-update, the same linearization approach as in REASON 1 is used

$$M_{k+1} = \arg\min\{\mathrm{Tr}(\nabla f(M_k), M - M_k) - \mathrm{Tr}(Z_k, M - S_k - L_k) + \frac{\rho}{2}\|M - S_k - L_k\|_{\mathbb{F}}^2 + \frac{\rho_x}{2}\|M - M_k\|_{\mathbb{F}}^2\}.$$

This is an unconstrained quadratic optimization with closed-form updates, as shown in REASON 2. The update rules for S, L are result of doing an inexact proximal update by considering them as a single block, which can then be decoupled as follows. For details, see Section 5.2.

$$\arg\min_{\|S-\tilde{S}_i\|_1^2 \le R_i^2} \quad \lambda_i\|S\|_1 + \frac{\rho}{2\tau_k}\|S - (S_k + \tau_k G_{M_k})\|_{\mathbb{F}}^2, \tag{11}$$

$$\arg\min_{\substack{\|L-\tilde{L}_i\|_*^2 \le \tilde{R}_i^2 \\ \|L\|_\infty \le \alpha/p}} \quad \lambda_i\|L\|_* + \frac{\rho}{2\tau_k}\|L - (L_k + \tau_k G_{M_k})\|_{\mathbb{F}}^2, \tag{12}$$

where $G_{M_k} = M_{k+1} - S_k - L_k - \frac{1}{\rho}Z_k$.

As before, we consider epochs of length T_0 and project the estimates S and L around the epoch initializations $\tilde{S}_i$ and $\tilde{L}_i$. We do not need to constrain the update of matrix M. We impose an ℓ_1-norm project for the sparse estimate S. For the low rank estimate L, we impose a nuclear norm projection around the epoch initialization $\tilde{L}_i$. Intuitively, the nuclear norm projection , which is an ℓ_1 projection on the singular values, encourages sparsity in the spectral domain leading to low rank estimates. In addition, we impose an ℓ_∞ constraint of α/p on each entry of L, which is different from the update of S. Note that the ℓ_∞ constraint is also imposed for the batch version of the problem (4) in (Agarwal et al., 2012a), and we assume that the true matrix L^* satisfies this constraint. For more discussions, see Section 4.2.

Note that each step of the method is easily implementable. The M-update is in closed form. The S-update involves optimization with projection on to the given ℓ_1 ball which can be performed efficiently (Duchi et al., 2008), as discussed in Section 3.1. For implementation details see Appendix E.2.

For the L-update, we introduce an additional auxiliary variable Y and we have

$$\begin{aligned}
L_{k+1} &= \min_{\|L-\tilde{L}_i\|_*^2 \le \tilde{R}_i^2} \quad \lambda_i\|L\|_* - \mathrm{Tr}(U_k, L - Y_k) + \frac{\rho}{2}\|L - Y_k\|_{\mathbb{F}}^2, \\
Y_{k+1} &= \min_{\|Y\|_\infty \le \alpha/p} \quad \frac{\rho}{2\tau_k}\|L - (L_k + \tau_k G_{M_k})\|_{\mathbb{F}}^2 + \frac{\rho}{2}\|L_{k+1} - Y\|_{\mathbb{F}}^2 - \mathrm{Tr}(U_k, L_{k+1} - Y), \\
U_{k+1} &= U_k - \tau(L_{k+1} - Y_{k+1}).
\end{aligned}$$

The L-update can now be performed efficiently by computing a SVD, and then running the projection step (Duchi et al., 2008). Note that approximate SVD computation techniques can be employed for efficiency here, e.g., (Lerman et al., 2012). The Y-update is projection on to the infinity norm ball which can be found easily. Let $Y_{(j)}$ stand for j-th entry of vector(Y). The for

Algorithm 2 Regularized Epoch-based Admm for Stochastic Optimization in high-dimensioN 2 (REASON 2)

```
Input $\rho, \rho_x > 0$, epoch length $T_0$ , regularizers $\{\lambda_i, \mu_i\}_{i=1}^{k_T}$, initial prox center $\tilde{S}_1, \tilde{L}_1$, initial radii $R_1, \tilde{R}_1$.
Define $Shrink_\kappa(a)$ shrinkage operator in (7), $G_{M_k} = M_{k+1} - S_k - L_k - \frac{1}{\rho} Z_k$.
for Each epoch $i = 1, 2, ..., k_T$ do
    Initialize $S_0 = \tilde{S}_i, L_0 = \tilde{L}_i, M_0 = S_0 + L_0$
    for Each iteration $k = 0, 1, ..., T_0 - 1$ do
        $M_{k+1} = \frac{-\nabla f(M_k) + Z_k + \rho(S_k + L_k) + \rho_x M_k}{\rho + \rho_x}$
        $S_{k+1} = \min_{\|S - \tilde{S}_i\|_1 \leq R_i} \lambda_i \|S\|_1 + \frac{\rho}{2\tau_k} \|S - (S_k + \tau_k G_{M_k})\|_{\mathbb{F}}^2$
        $L_{k+1} = \min_{\|L - \tilde{L}_i\|_* \leq \tilde{R}_i} \mu_i \|L\|_* + \frac{\rho}{2} \|L - Y_k - U_k/\rho\|_{\mathbb{F}}^2$
        $Y_{k+1} = \min_{\|Y\|_\infty \leq \alpha/p} \frac{\rho}{2\tau_k} \|Y - (L_k + \tau_k G_{M_k})\|_{\mathbb{F}}^2 + \frac{\rho}{2} \|L_{k+1} - Y - U_k/\rho\|_{\mathbb{F}}^2$
        $Z_{k+1} = Z_k - \tau(M_{k+1} - (S_{k+1} + L_{k+1}))$
        $U_{k+1} = U_k - \tau(L_{k+1} - Y_{k+1}).$
    end for
    Set: $\tilde{S}_{i+1} = \frac{1}{T_0} \sum_{k=0}^{T_0 - 1} S_k$ and $\tilde{L}_{i+1} := \frac{1}{T_0} \sum_{k=0}^{T_0 - 1} L_k$
    if $R_i^2 > 2(s + r + \frac{(s+r)^2}{p\gamma^2}) \frac{\alpha^2}{p}$ then
        Update $R_{i+1}^2 = R_i^2/2, \tilde{R}_{i+1}^2 = \tilde{R}_i^{\,2}/2$
    else
        STOP
    end if
end for
```

any j-th entry of vector(Y), solution will be as follows

$$\begin{aligned} &\text{If} \quad |(L_{k+1} + \frac{\tau_k}{\tau_k + 1}(G_{M_k} - U_k/\rho))_{(j)}| \leq \frac{\alpha}{p}, \\ &\quad \text{then} \quad Y_{(j)} = (L_{k+1} + \frac{\tau_k}{\tau_k + 1}(G_{M_k} - U_k/\rho))_{(j)}. \\ &\text{Else} \quad Y_{(j)} = \text{sign}\left((L_{k+1} + \frac{\tau_k}{\tau_k + 1}(G_{M_k} - U_k/\rho))_{(j)} - \frac{\alpha}{p}\right) \frac{\alpha}{p}. \end{aligned}$$

As before, the epoch averages are computed and used as initializations for the next epoch.

4.2 High-dimensional Guarantees

We now provide guarantees that REASON 2 efficiently recovers both the sparse and the low rank estimates in high dimensions efficiently. We need the following assumptions, in addition to Assumptions A1 and A2 from the previous section.

Assumption A4: Spectral Bound on the Gradient Error Let $E_k(M, X_k) := \nabla f(M, X_k) - \mathbb{E}[\nabla f(M, X_k)]$, $\|E_k\|_2 \leq \beta(p)\sigma$, where $\sigma := \|E_k\|_\infty$.

Recall from Assumption A2 that $\sigma = \mathcal{O}(\log p)$, under sub-Gaussianity. Here, we require spectral bounds in addition to $\|\cdot\|_\infty$ bound in A2.

Assumption A5: Bound on spikiness of low-rank matrix $\|L^*\|_\infty \leq \frac{\alpha}{p}$.

Intuitively, the ℓ_∞ constraint controls the "spikiness" of L^*. If $\alpha \approx 1$, then the entries of L are $\mathcal{O}(1/p)$, i.e. they are "diffuse" or "non-spiky", and no entry is too large. When the low rank matrix L^* has diffuse entries, it cannot be a sparse matrix, and thus, can be separated from the sparse S^* efficiently. In fact, the ℓ_∞ constraint is a weaker form of the *incoherence*-type assumptions needed to guarantee identifiability (Chandrasekaran et al., 2011) for sparse+low rank decomposition.

Assumption A6: Local strong convexity (LSC) The function $f : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^{n_1 \times n_2}$ satisfies an R-local form of strong convexity (LSC) if there is a non-negative constant $\gamma = \gamma(R)$ such that $f(B_1) \geq f(B_2) + \mathrm{Tr}(\nabla f(B_2)(B_1 - B_2)) + \frac{\gamma}{2}\|B_2 - B_1\|_{\mathbb{F}}$, for any $\|B_1\| \leq R$ and $\|B_2\| \leq R$, which is essentially the matrix version of Assumption A1. Note that we only require LSC condition on $S + L$ and not jointly on S and L.

We choose algorithm parameters as below where λ_i, μ_i are the regularization for ℓ_1 and nuclear norm respectively, ρ, ρ_x correspond to penalty terms in M-update and τ is dual update step size.

$$\lambda_i^2 = \frac{\gamma\sqrt{R_i^2 + \tilde{R}_i^2}}{(s+r)\sqrt{T_0}} \sqrt{\log p + \frac{G^2}{T_0} + \beta^2(p)\sigma_i^2 w_i^2 +} \frac{\rho_x^2(R_i^2 + \tilde{R}_i^2)}{T_0} + \frac{\alpha^2}{p^2} + \frac{\beta^2(p)\sigma^2}{T_0}\left(\log p + w_i^2\right), \quad (13)$$

$$\mu_i^2 = c_\mu \lambda_i^2, \quad \rho \propto \sqrt{\frac{T_0 \log p}{R_i^2 + \tilde{R}_i^2}}, \quad \rho_x > 0, \quad \tau = \rho.$$

Theorem 2. *Under assumptions A2 − A6, parameter settings* (13) *, let* T *denote total number of iterations and* $T_0 = T\log p / k_T$. *Assuming that above setting guarantees* $T_0 = \mathcal{O}(\log p)$,

$$\|\bar{S}(T) - S^*\|_{\mathbb{F}}^2 + \|\bar{L}(T) - L^*\|_{\mathbb{F}}^2 = \quad (14)$$
$$\mathcal{O}\left((s+r)\frac{\log p + \beta^2(p)\sigma^2\left(w^2 + \log(k_T/\log p)\right)}{T}\frac{\log p}{k_T}\right) + \left(1 + \frac{s+r}{\gamma^2 p}\right)\frac{\alpha^2}{p},$$

with probability at least $1 - 6\exp(-w^2/12)$,

$$k_T \simeq -\log\left(\frac{(s+r)^2}{\gamma^2 R_1^2 T}\left[\log p + \beta^2(p)\sigma^2 w^2\right]\right).$$

For proof, see Appendix D.6

Improvement of $\log p$ factor : The above result can be improved by a $\log p$ factor by considering varying epoch lengths (which depend on the problem parameters). The resulting convergence rate is $\mathcal{O}((s+r)p\log p/T + \alpha^2/p)$. See Theorem 4 in Appendix C.

Scaling of $\beta(p)$: We have the following bounds $\Theta(\sqrt{p}) \leq \beta(p)\Theta(p)$. This implies that the convergence rate is $\mathcal{O}((s+r)p\log p/T + \alpha^2/p)$, when $\beta(p) = \Theta(\sqrt{p})$ and when $\beta(p) = \Theta(p)$, it is $\mathcal{O}((s+r)p^2\log p/T + \alpha^2/p)$. The upper bound on $\beta(p)$ arises trivially by converting the max-norm $\|E_k\|_\infty \leq \sigma$ to the bound on the spectral norm $\|E_k\|_2$. In many interesting scenarios, the lower bound on $\beta(p)$ is achieved, as outlined in Section 4.2.1.

Comparison with the batch result: Agarwal et al. (2012a) consider the batch version of the same problem (4), and provide a convergence rate of $\mathcal{O}(s\log p + rp)/T + s\alpha^2/p^2)$. This is also the minimax lower bound under the independent noise model. With respect to the convergence rate, we match their results with respect to the scaling of s and r, and also obtain a $1/T$ rate. We match the scaling with respect to p (up to a log factor), when $\beta(p) = \Theta(\sqrt{p})$ attains the lower bound, and we discuss a few such instances below. Otherwise, we are worse by a factor of p compared to the batch version. Intuitively, this is because we require different bounds on error terms E_k in the online and the batch settings. For online analysis, we need to bound $\sum_{k=1}^{T_i} \|E_k\|_2/T_i$ over each epoch, while for the batch analysis, we need to bound $\|\sum_{k=1}^{T_i} E_k\|_2/T_i$, which is smaller. Intuitively, the difference for the two settings can be explained as follows: for the batch setting, since we consider an empirical estimate, we operate on the averaged error, while we are manipulating each sample in the online setting and suffer from the error due to that sample. We can employ efficient concentration bounds for the batch case (Tropp, 2012), while for the online case, no such bounds exist in general. From these observations, we conjecture that our bounds in Theorem 4 are *unimproveable* in the online setting.

Approximation Error: Note that the optimal decomposition $M^* = S^* + L^*$ is not identifiable in general without the incoherence-style conditions (Chandrasekaran et al., 2011; Hsu et al., 2011). In this paper, we provide efficient guarantees without assuming such strong incoherence constraints. This implies that there is an *approximation error* which is incurred even in the noiseless setting due to model non-identifiability. Agarwal et al. (2012a) achieve an approximation error of $s\alpha^2/p^2$ for their batch algorithm. Our online algorithm has an approximation error of $\max\{s + r, p\}\alpha^2/p^2$, which is worse, but is still decaying with p. It is not clear if this bound can be improved by any other online algorithm.

4.2.1 Optimal Guarantees for Various Statistical Models

We now list some statistical models under which we achieve the batch-optimal rate for sparse+low rank decomposition.

1) Independent Noise Model: Assume we sample i.i.d. matrices $X_k = S^* + L^* + N_k$, where the noise N_k has independent bounded sub-Gaussian entries with $\max_{i,j} \mathrm{Var}(N_k(i,j)) = \sigma^2$. We consider the square loss function, i.e. $\|X_k - S - L\|_{\mathbb{F}}^2$. In this case, $E_k = X_k - S^* - L^* = N_k$. From [Thm. 1.1](Vu, 2005), we have w.h.p that $\|N_k\| = \mathcal{O}(\sigma\sqrt{p})$. We match the batch bound of (Agarwal et al., 2012a) in this setting. Moreover, Agarwal et al. (2012a) provide a minimax lower bound for this model, and we match it as well. Thus, we achieve the optimal convergence rate for online matrix decomposition under the independent noise model.

2) Linear Bayesian Network: Consider a p-dimensional vector $y = Ah + n$, where $h \in \mathbb{R}^r$ with $r \leq p$, and $n \in \mathbb{R}^p$. The variable h is hidden, and y is the observed variable. We assume that the vectors h and n are each zero-mean sub-Gaussian vectors with i.i.d entries, and are independent of one another. Let σ_h^2 and σ_n^2 be the variances for the entries of h and n respectively. Without loss of generality, we assume that the columns of A are normalized, as we can always rescale A and σ_h appropriately to obtain the same model. Let $\Sigma_{y,y}^*$ be the true covariance matrix of y. From the independence assumptions, we have $\Sigma^*_{y,y} = S^* + L^*$, where $S^* = \sigma_n^2 I$ is a diagonal matrix and $L^* = \sigma_h^2 AA^\top$ has rank at most r.

In each step k, we obtain a sample y_k from the Bayesian network. For the square loss function f, we have the error $E_k = y_k y_k^\top - \Sigma^*_{y,y}$. Applying [Cor. 5.50](Vershynin, 2010), we have, with w.h.p.

$$\|n_k n_k^\top - \sigma_n^2 I\|_2 = \mathcal{O}(\sqrt{p}\sigma_n^2), \quad \|h_k h_k^\top - \sigma_h^2 I\|_2 = \mathcal{O}(\sqrt{p}\sigma_h^2). \tag{15}$$

We thus have with probability $1 - Te^{-cp}$, $\|E_k\|_2 \leq \mathcal{O}\left(\sqrt{p}(\|A\|^2\sigma_h^2 + \sigma_n^2)\right), \quad \forall\, k \leq T$. When $\|A\|_2$ is bounded, we obtain the optimal bound in Theorem 4, which matches the batch bound. If the entries of A are *generically* drawn (e.g., from a Gaussian distribution), we have $\|A\|_2 = \mathcal{O}(1+\sqrt{r/p})$. Moreover, such generic matrices A are also "diffuse", and thus, the low rank matrix L^* satisfies Assumption A5, with $\alpha \sim \text{polylog}(p)$. Intuitively, when A is generically drawn, there are diffuse connections from hidden to observed variables, and we have efficient guarantees under this setting.

Thus, our online method matches the batch guarantees for linear Bayesian networks when the entries of the observed vector y are conditionally independent given the latent variable h. When this assumption is violated, the above framework is no longer applicable since the true covariance matrix $\Sigma^*_{y,y}$ is *not* composed of a sparse matrix. To handle such models, we consider matrix decomposition of the inverse covariance or the precision matrix $M^* := \Sigma^{*-1}_{y,y}$, which can be expressed as a combination of sparse and low rank matrices, for the class of latent Gaussian graphical models, described in Section 5.3. Note that the result cannot be applied directly in this case as loss function is not locally Lipschitz. Nevertheless, in Section 5.3 we show that we can take care of this problem.

5 Proof Ideas and Discussion

5.1 Proof Ideas for REASON 1

1. In general, it is not possible to establish error contraction for stochastic ADMM at the end of each step. We establish error contracting at the end of certain time epochs, and we impose different levels of regularizations over different epochs. We perform an induction on the error, i.e. if the error at the end of k^{th} epoch is $\|\bar{\theta}(T_i) - \theta^*\|_2^2 \leq cR_i^2$, we show that in the subsequent epoch, it contracts as $\|\bar{\theta}(T_{i+1}) - \theta^*\|_2^2 \leq cR_i^2/2$ under appropriate choice of T_i, R_i and other design parameters. This is possible when we establish feasibility of the optimal solution θ^* in each epoch. Once this is established, it is straightforward to obtain the result in Theorem 3.

2. To show error contraction, we break down the error $\|\bar{\theta}(T_i) - \theta^*\|_2$ into two parts, viz., $\|\bar{\theta}(T_i) - \hat{\theta}(T_i)\|_2$ and $\|\hat{\theta}(T_i) - \theta^*\|_2$, where $\hat{\theta}(T_i)$ is the optimal batch estimate over the i-th epoch. The first term $\|\bar{\theta}(T_i) - \hat{\theta}(T_i)\|_2$ is obtained on the lines of analysis of stochastic ADMM, e.g., (Wang and Banerjee, 2013). Nevertheless, our analysis differs from that of (Wang and Banerjee, 2013), as theirs is not a stochastic method. i.e., the sampling error is not considered. Moreover, we show that the parameter ρ_x can be chosen as a constant while the earlier work (Wang and Banerjee, 2013) requires a stronger constraint $\rho_x = \sqrt{T_i}$. For details, see Appendix B.1. In addition, the ℓ_1 constraint that we impose enables us to provide tight bounds for the high dimensional regime. The second term $\|\hat{\theta}(T_i) - \theta^*\|_2$ is obtained by exploiting the local strong convexity properties of the loss function, on lines of (Agarwal et al.,

2012b). There are additional complications in our setting, since we have an auxiliary variable y for update of the regularization term. We relate the two variables through the dual variable, and use the fact that the dual variable is bounded. Note that this is a direct result from local Lipschitz property and it is proved in Lemma 5 in Appendix B.1. In fact, in order to prove the guarantees, we need bounded duality which is a weaker assumption than local Lipschitz property. We discuss this in Section 5.3.

3. For fixed epoch length, the error shrinkage stops after some epochs but the error does not increase significantly afterwards. Following lines of (Agarwal et al., 2012b), we prove that for this case the convergence rate is worse by a factor of $\log d$.

5.2 Proof Ideas for REASON 2

We now provide a short overview of proof techniques for establishing the guarantees in Theorem 2. It builds on the proof techniques used for proving Theorem 1, but is significantly more involved since we now need to decouple the errors for sparse and low rank matrix estimation, and our ADMM method consists of multiple blocks. The main steps are as follows

1. It is convenient to define $W = [S; L]$ to merge the variables L and S into a single variable W, as in (Ma et al., 2012). Let $\phi(W) = \|S\|_1 + \frac{\mu_i}{\lambda_i}\|L\|_*$, and $A = [I, I]$. The ADMM update for S and L in REASON 2, can now be rewritten as a single update for variable W. Consider the update

   $$W_{k+1} = \arg\min_W \{\lambda_i \phi(W) + \frac{\rho}{2}\|M_{k+1} - AW - \frac{1}{\rho}Z_k\|_{\mathbb{F}}^2\}.$$

   The above problem is not easy to solve as the S and L parts are coupled together. Instead, we solve it inexactly through one step of a proximal gradient method as in (Ma et al., 2012) as

   $$\arg\min_W \{\lambda_i \phi(W) + \frac{\rho}{2\tau_k}\|W - [W_k + \tau_k A^\top(M_{k+1} - AW_k - \frac{1}{\rho}Z_k)]\|_{\mathbb{F}}^2\}. \tag{16}$$

   Since the two parts of $W = [S; L]$ are separable in the quadratic part now, Equation (16) reduces to two decoupled updates on S and L as given by (11) and (12).

2. It is convenient to analyze the W update in Equation (16) to derive convergence rates for the online update in one time epoch. Once this is obtained, we also need error bounds for the batch procedure, and we employ the guarantees from Agarwal et al. (2012a). As in the previous setting of sparse optimization, we combine the two results to obtain an error bound for the online updates by considering multiple time epochs.

   It should be noted that we only require LSC condition on $S + L$ and not jointly on S and L. This results in an additional higher order term when analyzing the epoch error and therefore does not play a role in the final convergence bound. The LSC bound provides us with sum of sparse and low rank errors for each epoch. i.e., $\|\hat{S}_i - \bar{S}(T_i) + \hat{L}_i - \bar{L}(T_i)\|_{\mathbb{F}}^2$. Next we need to decouple these errors.

3. An added difficulty in the matrix decomposition problem is decoupling the errors for the sparse and low rank estimates. To this end, we impose norm constraints on the estimates of

S and L, and carry them over from epoch to epoch. On the other hand, at the end of each epoch M is reset. These norm constraints allows us to control the error. Special care needs to be taken in many steps of the proof to carefully transform the various norm bounds, where a naive analysis would lead to worse scaling in the dimensionality p. We instead carefully project the error matrices on to on and off support of S^* for the ℓ_1 norm term, and similarly onto the range and its complement of L^* for the nuclear norm term. This allows us to have a convergence rate with a $s + r$ term, instead of p.

4. For fixed epoch length, the error shrinkage stops after some epochs but the error does not increase significantly afterwards. Following lines of (Agarwal et al., 2012b), we prove that for this case the convergence rate is worse by a factor of $\log p$.

Thus, our careful analysis leads to tight guarantees for online matrix decomposition. For Proof outline and detailed proof of Theorem 2 see Appendix C.1 and D respectively.

5.3 Graphical Model Selection

Our framework cannot directly handle the case where loss function is the log likelihood objective. This is because for log likelihood function Lipschitz constant can be large and this leads to loose bounds on error. Yet, as we discuss shortly, our analysis needs conditions weaker than Local Lipschitz property. We consider both settings, i.e., fully observed graphical models and latent Gaussian graphical models. We apply sparse optimization to the former and tackle the latter with sparse + low rank decomposition.

5.3.1 Sparse optimization for learning Gaussian graphical models

Consider a p-dimensional Gaussian random vector $[x_1, ..., x_p]^\top$ with a sparse inverse covariance or precision matrix Θ^*. Consider the ℓ_1-regularized maximum likelihood estimator (batch estimate),

$$\widehat{\Theta} := \underset{\Theta \succ 0}{\arg\min} \{\mathrm{Tr}(\widehat{\Sigma}\Theta) - \log\det\{\Theta\} + \lambda_n \|\Theta\|_1\}, \tag{17}$$

where $\widehat{\Sigma}$ is the empirical covariance matrix for the batch. This is a well-studied method for recovering the edge structure in a Gaussian graphical model, i.e. the sparsity pattern of Θ^* (Ravikumar et al., 2011). We have that the loss function is strongly convex for all Θ within a ball[1].

However, the above loss function is not (locally) Lipschitz in general, since the gradient[2] $\nabla f(x, \Theta) = xx^\top - \Theta^{-1}$ is not bounded in general. Thus, the bounds derived in Theorem 1 do not directly apply here. However, our conditions for recovery are somewhat weaker than local Lipschitz property, and we provide guarantees for this setting under some additional constraints.

Let $\Gamma^* = \Theta^{*-1} \otimes \Theta^{*-1}$ denote the Hessian of log-determinant barrier at true information matrix. Let $Y_{(j,k)} := X_j X_k - \mathbb{E}[X - jX_k]$ and note that $\Gamma^*_{(j,k),(l,m)} = \mathbb{E}[Y_{(j,k) Y_{(l,m)}}]$ (Ravikumar et al., 2011). A bound on $|||\Gamma^*|||_\infty$ limits the influence of the edges on each other, and we need this bound for guaranteed convergence. Yet, this bound contributes to a higher order term and does not show up in the convergence rate.

[1] Let $Q = \{\theta \in \mathbb{R}^n : \alpha I_n \preceq \Theta\beta I_n\}$ then $-\log\det\Theta$ is strongly convex on Q with $\gamma = \frac{1}{\beta^2}$ (d'Aspremont et al., 2008).

[2] The gradient computation can be expensive since it involves computing the matrix inverse. However, efficient techniques for computing an approximate inverse can be employed, on lines of (Hsieh et al., 2011).

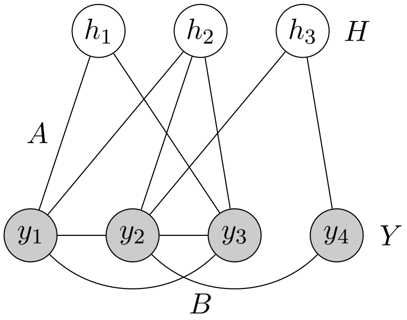

Figure 1: Graphical representation of a latent variable model.

Corollary 1. *Under Assumptions A1, A2 when the initialization radius R_1 satisfies $R_1 \leq \frac{0.25}{\|\Sigma^*\|_{\mathbb{F}}}$, under the negative log-likelihood loss function, REASON 1 has the following bound (for dual update step size $\tau = \sqrt{T_0}$)*

$$\|\bar{\theta}_T - \theta^*\|_2^2 \leq c_0 \frac{s}{\gamma^2 T} \cdot \frac{\log d}{k_T} \left[\log d + \sigma^2 \left(w^2 + 24 \log(k_T / \log d)\right)\right]$$

The proof does not follow directly from Theorem 1, since it does not utilize Lipschitz property. However, the conditions for Theorem 1 to hold are weaker than (local) Lipschitz property and we utilize it to provide the above result. For proof, see Appendix B.7. Note that in case epoch length is not fixed and depends on the problem parameters, the bound can be improved by a $\log d$ factor.

Comparing to Theorem 1, the local Lipschitz constant G^4 is replaced by $\sigma^2 \|\!|\Gamma^*|\!\|^2$. We have $G = \mathcal{O}(d)$, and thus we can obtain better bounds in the above result, when $\|\!|\Gamma^*|\!\|$ is small and the initialization radius R_1 satisfies the above condition. Intuitively, the initialization condition (constraint on R_1) is dependent on the strength of the correlations. For the weak-correlation case, we can initialize with large error compared to the strongly correlated setting.

5.3.2 Sparse + low rank decomposition for learning latent Gaussian graphical models

Consider the Bayesian network on p-dimensional observed variables as

$$y = A\,h + B\,y + n, \quad y, n \in \mathbb{R}^p,\ h \in \mathbb{R}^r, \tag{18}$$

as in Figure 1 where h, y and n are drawn from a zero-mean multivariate Gaussian distribution. The vectors h and n are independent of one another, and $n \sim \mathcal{N}(0, \sigma_n^2 I)$. Assume that A has full column rank. Without loss of generality, we assume that A has normalized columns, and that h has independent entries (Pitman and Ross, 2012). For simplicity, let $h \sim \mathcal{N}(0, \sigma_h^2 I)$ (more generally, its covariance is a diagonal matrix). Note that the matrix $B = 0$ in the previous setting (the previous setting allows for more general sub-Gaussian distributions, and here, we limit ourselves to the Gaussian distribution). For the model in (18), the precision matrix M^* with respect to the marginal distribution on the observed vector y is given by

$$M^* := {\Sigma^*}_{y,y}^{-1} = \widetilde{M}_{y,y}^* - \widetilde{M}_{y,h}^* (\widetilde{M}_{h,h}^*)^{-1} \widetilde{M}_{h,y}^*, \tag{19}$$

where $\widetilde{M}^* = \Sigma^{*-1}$, and Σ^* is the joint-covariance matrix of vectors y and h. It is easy to see that the second term in (19) has rank at most r. The first term in (19) is sparse under some natural constraints, viz., when the matrix B is sparse, and there are a small number of *colliders* among the observed variables y. A triplet of variables consisting of two *parents* and their *child* in a Bayesian network is termed as a collider. The presence of colliders results in additional edges when the Bayesian network on y and h is converted to an undirected graphical model, whose edges are given by the sparsity pattern $\widetilde{M^*_{y,y}}$, the first term in (19). Such a process is known as *moralization* (Lauritzen, 1996), and it involves introducing new edges between the parents in the directed graph (the graph of the Bayesian networks), and removing the directions to obtain an undirected model. Therefore, when the matrix B is sparse, and there are a small number of colliders among the observed variables y, the resulting sub-matrix $\widetilde{M^*_{y,y}}$ is also sparse.

We thus have the precision matrix M^* in (19) as $M^* = S^* + L^*$, where S^* and L^* are sparse and low rank components. We can find this decomposition via regularized maximum likelihood. The batch estimate is given by Chandrasekaran et al. (2012)

$$\{\hat{S}, \hat{L}\} := \arg\min\{\mathrm{Tr}(\widehat{\Sigma}_n M) - \log\det M + \lambda_n \|S\|_1 + \mu_n \|L\|_*\}, \tag{20}$$

$$s.t. \quad M = S + L. \tag{21}$$

This is a special case of (4) with the loss function $f(M) = \mathrm{Tr}(\widehat{\Sigma}_n\, M) - \log\det M$. In this case, we have the error $E_k = y_k y_k^\top - M^{*-1}$. Since $y = (I - B)^{-1}(Ah + n)$, we have the following bound w.h.p.

$$\|E_k\|_2 \leq \mathcal{O}\left(\frac{\sqrt{p}\cdot(\|A\|_2^2\sigma_h^2 + \sigma_n^2)\log(pT)}{\sigma_{\min}(I-B)^2}\right), \quad \forall\, k \leq T,$$

where $\sigma_{\min}(\cdot)$ denotes the minimum singular value. The above result is obtained by alluding to (15).

When $\|A\|_2$ and $\sigma_{\min}(I-B)$ are bounded, we thus achieve optimal scaling for our proposed online method. As discussed for the previous case, when A is generically drawn, $\|A\|_2$ is bounded. To bound $\sigma_{\min}(I-B)$, a sufficient condition is *walk-summability* on the sub-graph among the observed variables y. The class of walk-summable models is efficient for inference (Malioutov et al., 2006) and structure learning (Anandkumar et al., 2012), and they contain the class of attractive models. Thus, it is perhaps not surprising that we obtain efficient guarantees for such models for our online algorithm.

We need to slightly change the algorithm REASON 2 for this scenario as follows: for the M-update in REASON 2, we add a ℓ_1 norm constraint on M as $\|M_k - \tilde{S}_i - \tilde{L}_i\|_1^2 \leq \breve{R}^2$, and this can still be computed efficiently, since it involves projection on to the ℓ_1 norm ball, see Appendix E.1. We assume a good initialization M which satisfies $\|M - M^*\|_1^2 \leq \breve{R}^2$.

This ensures that M_k in subsequent steps is non-singular, and that the gradient of the loss function f in (20), which involves M_k^{-1}, can be computed. As observed in section 5.3.1 on sparse graphical model selection, the method can be made more efficient by computing approximate matrix inverses (Hsieh et al., 2013). As observed before, the loss function f satisfies the local strong convexity property, and the guarantees in Theorem 2 are applicable.

There is another reason for using the ℓ_1 bound. Note that the loss function is not generally Lipschitz in this case. However, our conditions for recovery are somewhat weaker than local Lipschitz property, and we provide guarantees for this setting under some additional constraints. Let $\Gamma^* = M^* \otimes M^*$. As explained in Section 5.3.1, a bound on $|\!|\!|\Gamma^*|\!|\!|_\infty$ limits the influence on the edges on each other, and we need this bound for guaranteed convergence. Yet, this bound contributes to a higher order term and does not show up in the convergence rate.

Dimension	Run Time (s)	Method	error at 0.02T	error at 0.2T	error at T
d=20000	T=50	ST-ADMM	1.022	1.002	0.996
		RADAR	0.116	2.10e-03	6.26e-05
		REASON 1	1.5e-03	2.20e-04	1.07e-08
d=2000	T=5	ST-ADMM	0.794	0.380	0.348
		RADAR	0.103	4.80e-03	1.53e-04
		REASON 1	0.001	2.26e-04	1.58e-08
d=20	T=0.2	ST-ADMM	0.212	0.092	0.033
		RADAR	0.531	4.70e-03	4.91e-04
		REASON 1	0.100	2.02e-04	1.09e-08

Table 3: *Least square regression problem, epoch size* $T_i = 2000$, *Error*$= \frac{\|\theta-\theta^*\|_2}{\|\theta^*\|_2}$.

Corollary 2. *Under Assumptions A1, A2, A4, A5, when the radius* $\breve{R}$ *satisfies* $\breve{R} \leq \frac{0.25}{\|\Sigma^*\|_{\mathbb{F}}}$, *under the negative log-likelihood loss function, REASON 2 has the following bound (for dual update step size* $\tau = \sqrt{T_0}$*)*

$$\|\bar{S}(T) - S^*\|_{\mathbb{F}}^2 + \|\bar{L}(T) - L^*\|_{\mathbb{F}}^2 \leq$$
$$\frac{c_0(s+r)}{T} \cdot \frac{\log p}{k_T}\left[\log p + \beta^2(p)\sigma^2\left(w^2 + \log(k_T/\log p)\right)\right] + \max\{s+r, p\}\frac{\alpha^2}{p}.$$

The proof does not follow directly from Theorem 2, since it does not utilize Lipschitz property. However, the conditions for Theorem 2 to hold are weaker than (local) Lipschitz property and we utilize it to provide the above result. For proof, see Appendix D.7. Note that in case epoch length is not fixed and depends on the problem parameters, the bound can be improved by a $\log p$ factor.

6 Experiments

6.1 REASON 1

For sparse optimization problem we compare REASON 1 with RADAR and ST-ADMM under the least-squares regression setting. Samples (x_t, y_t) are generated such that $x_t \in \text{Unif}[-B, B]$ and $y_t = \langle \theta^*, x \rangle + n_t$. θ^* is s-sparse with $s = \lceil \log d \rceil$. $n_t \sim \mathcal{N}(0, \eta^2)$. With $\eta^2 = 0.5$ in all cases. We consider $d = 20, 2000, 20000$ and $s = 1, 3, 5$ respectively. The experiments are performed on a 2.5 GHz Intel Core i5 laptop with 8 GB RAM. See Table 3 for experiment results. It should be noted that RADAR is provided with information of θ^* for epoch design and recentering. In addition, both RADAR and REASON 1 have the same initial radius. Nevertheless, REASON 1 reaches better accuracy within the same run time even for small time frames. In addition, we compare relative error $\|\theta - \theta^*\|_2/\|\theta^*\|_2$ in REASON 1 and ST-ADMM in the first epoch. We observe that in higher dimension error fluctuations for ADMM increases noticeably (see Figure 2). Therefore, projections of REASON 1 play an important role in denoising and obtaining good accuracy.

Epoch Size For fixed- epoch size, if epoch size is designed such that the relative error defined above has shrunk to a stable value, then we move to the next epoch and the algorithm works as

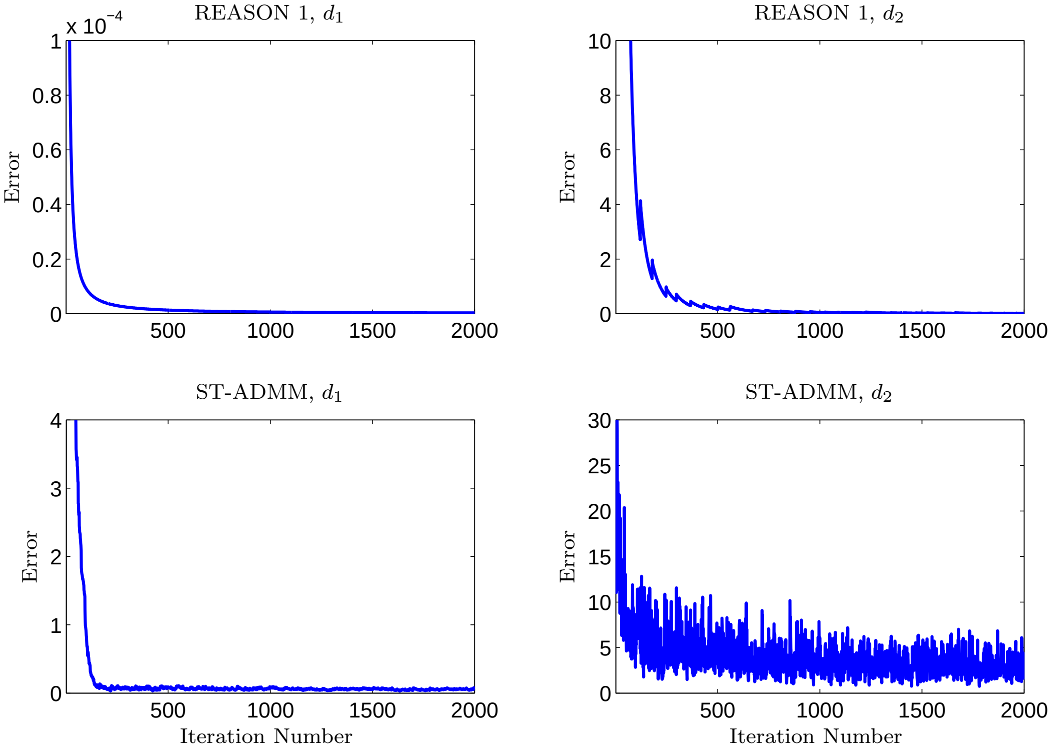

Figure 2: Least square regression, Error$= \frac{\|\theta-\theta^*\|_2}{\|\theta^*\|_2}$ vs. iteration number, $d_1 = 20$ and $d_2 = 20000$.

expected. If we choose a larger epoch than this value we do not gain much in terms of accuracy at a specific iteration. On the other hand if we use a small epoch size such that the relative error is still noticeable, this delays the error reduction and causes some local irregularities.

6.2 REASON 2

We compare REASON 2 with state-of-the-art inexact ALM method for matrix decomposition problem[1] In this problem M is the noisy sample the algorithm receives. Since we have direct access to M, the M-update is eliminated.

Table 4 shows that with equal time, inexact ALM reaches smaller $\frac{\|M^*-S-L\|_\mathbb{F}}{\|M^*\|_\mathbb{F}}$ error while in fact this does not provide a good decomposition. On the other hand, REASON 2 reaches useful individual errors in the same time frame. Experiments with $\eta^2 \in [0.01, 1]$ reveal similar results. This emphasizes the importance of projections in REASON 2. Further investigation on REASON 2 shows that performing one of the projections (either ℓ_1 or nuclear norm) suffices to reach this performance. The same precision can be reached using only one of the projections. Addition of the second projection improves the performance marginally. Performing nuclear norm projections are much more expensive since they require SVD. Therefore, it is more efficient to perform the ℓ_1 projection. Similar experiments on exact ALM shows worse performance than inexact ALM and

[1] ALM codes are downloaded from `http://perception.csl.illinois.edu/matrix-rank/home.html` and REASON 2 code is available at `https://github.com/haniesedghi/REASON2`.

Run Time	$T = 50$ sec			$T = 150$ sec		
Error	$\frac{\|M^*-S-L\|_{\mathbb{F}}}{\|M^*\|_{\mathbb{F}}}$	$\frac{\|S-S^*\|_{\mathbb{F}}}{\|S^*\|_{\mathbb{F}}}$	$\frac{\|L^*-L\|_{\mathbb{F}}}{\|L^*\|_{\mathbb{F}}}$	$\frac{\|M^*-S-L\|_{\mathbb{F}}}{\|M^*\|_{\mathbb{F}}}$	$\frac{\|S-S^*\|_{\mathbb{F}}}{\|S^*\|_{\mathbb{F}}}$	$\frac{\|L^*-L\|_{\mathbb{F}}}{\|L^*\|_{\mathbb{F}}}$
REASON 2	2.20e-03	0.004	0.01	5.55e-05	1.50e-04	3.25e-04
inexact ALM	5.11e-05	0.12	0.27	8.76e-09	0.12	0.27

Table 4: *REASON 2 and inexact ALM, matrix decomposition problem.* $p = 2000$, $\eta^2 = 0.01$

are thus omitted.

7 Conclusion

In this paper, we consider a modified version of the stochastic ADMM method for high-dimensional problems. We first analyze the simple setting, where the optimization problem consists of a loss function and a single regularizer, and then extend to the multi-block setting with multiple regularizers and multiple variables. For the sparse optimization problem, we showed that we reach the minimax-optimal rate in this case, which implies that our guarantee is unimproveable by any (batch or online) algorithm (up to constant factors). We then consider the matrix decomposition problem into sparse and low rank components, and propose a modified version of the multi-block ADMM algorithm. Experiments show that for both sparse optimization and matrix decomposition problems, our algorithm outperforms the state-of-the-art methods. In particular, we reach higher accuracy with same time complexity. There are various future problems to consider. One is to provide lower bounds on error for matrix decomposition problem in case of strongly convex loss if possible. Agarwal et al. (2012a) do not provide bounds for strongly convex functions. Another approach can be to extend our method to address nonconvex programs. Loh and Wainwright (2013) and Wang et al. (2013c) show that if the problem is nonconvex but has additional properties, it can be solved by methods similar to convex loss programs. In addition, we can extend our method to coordinate descent methods such as (Roux et al., 2012).

Acknowledgment

We acknowledge detailed discussions with Majid Janzamin and thank him for valuable comments on sparse and sparse + low rank recovery. The authors thank Alekh Agarwal for detailed discussions about his work and the minimax bounds. We thank him for pointing out that the condition regarding bounded dual variables is related to local Lipschitz condition.

A. Anandkumar is supported in part by Microsoft Faculty Fellowship, NSF Career award CCF-1254106, NSF Award CCF-1219234, and ARO YIP Award W911NF-13-1-0084.

References

A. Agarwal, S. Negahban, and M. Wainwright. Noisy matrix decomposition via convex relaxation: Optimal rates in high dimensions. *The Annals of Statistics*, 40(2):1171–1197, 2012a.

A. Agarwal, S. Negahban, and M. J. Wainwright. Stochastic optimization and sparse statistical recovery: Optimal algorithms for high dimensions. In *NIPS*, pages 1547–1555, 2012b.

A. Anandkumar, V. Tan, F. Huang, and A.S. Willsky. High-dimensional gaussian graphical model selection:walk summability and local separation criterion. *Journal of Machine Learning*, 13: 22932337, August 2012.

S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

E. J. Candès, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *Journal of the ACM (JACM)*, 58(3):11, 2011.

V. Chandrasekaran, S. Sanghavi, Pablo A Parrilo, and A. S Willsky. Rank-sparsity incoherence for matrix decomposition. *SIAM Journal on Optimization*, 21(2):572–596, 2011.

V. Chandrasekaran, P. A Parrilo, and Alan S Willsky. Latent variable graphical model selection via convex optimization. *The Annals of Statistics*, 40(4):1935–1967, 2012.

Alexandre d'Aspremont, Onureena Banerjee, and Laurent El Ghaoui. First-order methods for sparse covariance selection. *SIAM Journal on Matrix Analysis and Applications*, 30(1):56–66, 2008.

W. Deng, W.and Yin. On the global and linear convergence of the generalized alternating direction method of multipliers. Technical report, DTIC Document, 2012.

J. Duchi, S. Shalev-Shwartz, Y. Singer, and T. Chandra. Efficient projections onto the ℓ_1-ball for learning in high dimensions. In *Proceedings of the 25th international conference on Machine learning*, pages 272–279. ACM, 2008.

E. Esser, X. Zhang, and T. Chan. A general framework for a class of first order primal-dual algorithms for convex optimization in imaging science. *SIAM Journal on Imaging Sciences*, 3 (4):1015–1046, 2010.

T Goldstein, B. ODonoghue, and S. Setzer. Fast alternating direction optimization methods. *CAM report*, pages 12–35, 2012.

C. Hsieh, M. A Sustik, I. Dhillon, P. Ravikumar, and R. Poldrack. Big & quic: Sparse inverse covariance estimation for a million variables. In *Advances in Neural Information Processing Systems*, pages 3165–3173, 2013.

Cho-Jui Hsieh, Matyas A Sustik, Inderjit S Dhillon, and Pradeep D Ravikumar. Sparse inverse covariance matrix estimation using quadratic approximation. In *NIPS*, pages 2330–2338, 2011.

Daniel Hsu, Sham M Kakade, and Tong Zhang. Robust matrix decomposition with sparse corruptions. *Information Theory, IEEE Transactions on*, 57(11):7221–7234, 2011.

R. Johnson and T. Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 315–323. 2013.

S.L. Lauritzen. *Graphical models*. Clarendon Press, 1996.

Gilad Lerman, Michael McCoy, Joel A Tropp, and Teng Zhang. Robust computation of linear models, or how to find a needle in a haystack. *arXiv preprint arXiv:1202.4044*, 2012.

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.

Po-Ling Loh and Martin J Wainwright. Regularized m-estimators with nonconvexity: Statistical and algorithmic theory for local optima. In *Advances in Neural Information Processing Systems*, pages 476–484, 2013.

Zhi-Quan Luo. On the linear convergence of the alternating direction method of multipliers. *arXiv preprint arXiv:1208.3922*, 2012.

S. Ma, L. Xue, and H. Zou. Alternating direction methods for latent variable gaussian graphical model selection. *arXiv preprint arXiv:1206.1275v2*, 2012.

Dmitry M Malioutov, Jason K Johnson, and Alan S Willsky. Walk-sums and belief propagation in gaussian graphical models. *The Journal of Machine Learning Research*, 7:2031–2064, 2006.

J. FC Mota, J. MF Xavier, P. MQ Aguiar, and M. Puschel. Distributed admm for model predictive control and congestion control. In *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, pages 5110–5115. IEEE, 2012.

S. Negahban, P. Ravikumar, M. Wainwright, and B. Yu. A unified framework for high-dimensional analysis of M-estimators with decomposable regularizers. *Statistical Science*, 27(4):538–557, 2012.

H. Ouyang, N. He, L. Tran, and A. G Gray. Stochastic alternating direction method of multipliers. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pages 80–88, 2013.

Jim Pitman and Nathan Ross. Archimedes, gauss, and stein. *Notices AMS*, 59:1416–1421, 2012.

G. Raskutti, M. J. Wainwright, and B. Yu. Minimax rates of estimation for high-dimensional linear regression over ℓ_q-balls. *IEEE Trans. Information Theory*, 57(10):6976—6994, October 2011.

P. Ravikumar, M.J. Wainwright, G. Raskutti, and B. Yu. High-dimensional covariance estimation by minimizing ℓ_1-penalized log-determinant divergence. *Electronic Journal of Statistics*, (4): 935–980, 2011.

Nicolas Le Roux, Mark Schmidt, and Francis Bach. A stochastic gradient method with an exponential convergence rate for strongly-convex optimization with finite training sets. Technical report, 2012.

S. Shalev-Shwartz. Online learning and online convex optimization. *Foundations and Trends in Machine Learning*, 4(2):107–194, 2011.

S. Shalev-Shwartz and T. Zhang. Accelerated mini-batch stochastic dual coordinate ascent. In C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 378–385. 2013.

J. Tropp. User-friendly tail bounds for sums of random matrices. *Foundations of Computational Mathematics*, 12(4):389–434, 2012.

Roman Vershynin. Introduction to the non-asymptotic analysis of random matrices. *arXiv preprint arXiv:1011.3027*, 2010.

Van H Vu. Spectral norm of random matrices. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pages 423–430. ACM, 2005.

B. Wahlberg, S. Boyd, M. Annergren, and Y. Wang. An admm algorithm for a class of total variation regularized estimation problems. *arXiv preprint arXiv:1203.1828*, 2012.

C. Wang, X. Chen, A. Smola, and E. Xing. Variance reduction for stochastic gradient optimization. In C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 181–189. 2013a.

H. Wang and A. Banerjee. Bregman alternating direction method of multipliers. *arXiv preprint arXiv:1306.3203*, 2013.

X. Wang, M. Hong, S. Ma, and Z. Luo. Solving multiple-block separable convex minimization problems using two-block alternating direction method of multipliers. *arXiv preprint arXiv:1308.5294*, 2013b.

Zhaoran Wang, Han Liu, and Tong Zhang. Optimal computational and statistical rates of convergence for sparse nonconvex learning problems. *arXiv preprint arXiv:1306.4960*, 2013c.

G. Watson. Characterization of the subdifferential of some matrix norms. *Linear Algebra and its Applications*, 170(0):33–45, 1992.

A Guarantees for REASON 1

First, we provide guarantees for the theoretical case such that epoch length depends on epoch radius. This provides intuition on how the algorithm is designed. The fixed-epoch algorithm is a special case of this general framework. We first state and prove guarantees for general framework. Next, we leverage these results to prove Theorem 1.

Let the design parameters be set as

$$\begin{aligned}
T_i &= C\frac{s^2}{\gamma^2}\left[\frac{\log d + 12\sigma_i^2 \log(3/\delta_i)}{R_i^2}\right], \\
\lambda_i^2 &= \frac{\gamma}{s\sqrt{T_i}}\sqrt{R_i^2 \log d + \frac{G^2R_i^2 + \rho_x^2 R_i^4}{T_i} + \sigma_i^2 R_i^2 \log(3/\delta_i)}, \\
\rho &\propto \frac{\sqrt{\log d}}{R_i\sqrt{T_i}}, \quad \rho_x > 0, \quad \tau = \rho.
\end{aligned} \tag{22}$$

Theorem 3. *Under assumptions $A1 - A3$ and parameter settings (22), there exists a constant $c_0 > 0$ such that REASON 1 satisfies for all $T > k_T$,*

$$\|\bar{\theta}_T - \theta^*\|_2^2 \leq c_0 \frac{s}{\gamma^2 T}\left[e \log d + \sigma^2 w^2 + \log k_T\right], \tag{23}$$

with probability at least $1 - 6\exp(-w^2/12)$, where $k_T = \log_2 \frac{\gamma^2 R_1^2 T}{s^2(\log d + 12\sigma^2 \log(\frac{6}{\delta}))}$, and c_0 is a universal constant.

For Proof outline and detailed proof of Theorem 3 see Appendix A.1 and B respectively.

A.1 Proof outline for Theorem 3

The foundation block for this proof is Proposition 1.

Proposition 1. *Suppose f satisfies Assumptions $A1, A2$ with parameters γ and σ_i respectively and assume that $\|\theta^* - \tilde{\theta}_i\|_1^2 \leq R_i^2$. We apply the updates in REASON 1 with parameters as in (22). Then, there exists a universal constant c such that for any radius R_i*

$$\begin{aligned}
&f(\bar{\theta}(T_i)) - f(\hat{\theta}_i) + \lambda_i\|\bar{y}(T_i)\|_1 - \lambda_i\|\hat{\theta}_i\|_1 \leq \frac{R_i\sqrt{\log d}}{\sqrt{T_i}} + \frac{GR_i}{T_i} + \frac{\rho_x R_i^2}{T_i} \qquad (24a)\\
&+ \frac{R_i\sigma_i}{\sqrt{T_i}}\sqrt{12\log(3/\delta_i)}, \\
&\|\bar{\theta}(T_i) - \theta^*\|_1^2 \leq \frac{c'}{\sqrt{C}}R_i^2. \qquad (24b)
\end{aligned}$$

where $\rho_0 = \rho_x + \rho$ and both bounds are valid with probability at least $1 - \delta_i$.

Note that our proof for epoch optimum improves proof of (Wang and Banerjee, 2013) with respect to ρ_x. For details, see Section B.1.

In order to prove Proposition 1, we need to prove some more lemmas.

To move forward from here please note the following notations: $\Delta_i = \hat{\theta}_i - \theta^*$ and $\hat{\Delta}(T_i) = \bar{\theta}_i - \hat{\theta}_i$.

Lemma 1. *At epoch i assume that $\|\theta^* - \tilde{\theta}_i\|_1 \le R_i$. Then the error Δ_i satisfies the bounds*

$$\|\hat{\theta}_i - \theta^*\|_2 \le \frac{4}{\gamma}\sqrt{s}\lambda_i, \tag{25a}$$

$$\|\hat{\theta}_i - \theta^*\|_1 \le \frac{8}{\gamma}s\lambda_i. \tag{25b}$$

Lemma 2. *Under the conditions of Proposition 1 and with parameter settings* (22) *, we have*

$$\|\hat{\Delta}(T_i)\|_2^2 \le \frac{c'}{\sqrt{C}}\frac{1}{s}R_i^2,$$

with probability at least $1 - \delta_i$*.*

B Proof of Theorem 3

The first step is to ensure that $\|\theta^* - \tilde{\theta}_i\| \le R_i$ holds at each epoch so that Proposition 1 can be applied in a recursive manner. We prove this by induction on the epoch index. By construction, this bound holds at the first epoch. Assume that it holds for epoch i. Recall that T_i is defined by (22) where $C \ge 1$ is a constant we can choose. By substituting this T_i in inequality (24b), the simplified bound (24b) further yields

$$\|\bar{\theta}(T_i) - \theta^*\|_1^2 \le \frac{c'}{\sqrt{C}}R_i^2.$$

Thus, by choosing C sufficiently large, we can ensure that $\|\bar{\theta}(T_i) - \theta^*\|_1^2 \le R_i^2/2 := R_{i+1}^2$. Consequently, if θ^* is feasible at epoch i, it stays feasible at epoch $i+1$. Hence, by induction we are guaranteed the feasibility of θ^* throughout the run of algorithm.

As a result, Lemma 2 applies and we find that

$$\|\hat{\Delta}(T_i)\|_2^2 \le \frac{c}{s}R_i^2. \tag{26}$$

We have now bounded $\hat{\Delta}(T_i) = \bar{\theta}(T_i) - \hat{\theta}_i$ and Lemma 1 provides a bound on $\Delta_i = \hat{\theta}_i - \theta^*$, such that the error $\Delta^*(T_i) = \bar{\theta}(T_i) - \theta^*$ can be controlled by triangle inequality. In particular, by combining (25a) with (26), we get

$$\|\Delta^*(T_i)\|_2^2 \le c\{\frac{1}{s}R_i^2 + \frac{16}{s}R_i^2\},$$

i.e.

$$\|\Delta^*(T_i)\|_2^2 \le c\frac{R_1^2 2^{-(i-1)}}{s}. \tag{27}$$

The bound holds with probability at least $1 - 3\exp(-w_i^2/12)$. Recall that $R_i^2 = R_1^2 2^{-(i-1)}$. Since $w_i^2 = w^2 + 24\log i$, we can apply union bound to simplify the error probability as $1 - 6\exp(-w^2/12)$. Throughout this report we use $\delta_i = 3\exp(-w_i^2/12)$ and $\delta = 6\exp(-w^2/12)$ to simplify the equations.

To complete the proof we need to convert the error bound (27) from its dependence on the number of epochs k_T to the number of iterations needed to complete k_T epochs, i.e. $T(K) = \sum_{i=1}^{k} T_i$. Note that here we use T_i from (33), to show that when considering the dominant terms, the definition in (22) suffices. Here you can see how negligible terms are ignored.

$$\begin{aligned} T(k) &= \sum_{i=1}^{k} C\left[\frac{s^2}{\gamma^2}\left[\frac{\log d + 12\sigma_i^2\log(3/\delta_i)}{R_i^2}\right] + \frac{s}{\gamma}\frac{G}{R_i} + \frac{s}{\gamma}\rho_x\right] \\ &= C\sum_{i=1}^{k}\left[\frac{s^2\{\log d + \gamma/sG + \sigma^2(w^2+24\log k)\}2^{i-1}}{\gamma^2 R_1^2} + \frac{sG}{\gamma R_1}\sqrt{2}^{i-1} + \frac{s}{\gamma}\rho_x\right]. \end{aligned}$$

Hence,

$$T(k) \le C\left[\frac{s^2}{\gamma^2 R_1^2}\{\log d + \sigma^2(w^2+24\log k)\}2^k + \frac{s}{\gamma R_1}G\sqrt{2}^k + \frac{s}{\gamma}\rho_x\right].$$

$T(k) \le S(k)$, therefore $k_T \ge S^{-1}(T)$.

$$S(k) = C\left[\frac{s^2}{\gamma^2 R_1^2}\{\log d + \sigma^2(w^2+24\log k)\}2^k + \frac{s}{\gamma R_1}G\sqrt{2}^k + \frac{s}{\gamma}\rho_x\right].$$

Ignoring the dominated terms and using a first order approximation for $\log(a+b)$,

$$\begin{aligned} \log(T) &\simeq \log C + k_T + \log\left[\frac{s^2}{\gamma^2 R_1^2}\{\log d + \sigma^2(w^2+24\log k)\}\right], \\ k_T &\simeq \log T - \log C - \log\left[\frac{s^2}{\gamma^2 R_1^2}\{\log d + \sigma^2(w^2+24\log k)\}\right]. \end{aligned}$$

Therefore,

$$2^{-k_T} = \frac{Cs^2}{\gamma^2 T R_1^2}\{\log d + \sigma^2(w^2+24\log k)\}.$$

Putting this back into (27), we get that

$$\begin{aligned} \|\Delta^*(T_i)\|_2^2 &\le c\frac{R_1^2}{s}\frac{Cs^2}{\gamma^2 T R_1^2}\{\log d + \sigma^2(w^2+24\log k)\} \\ &\le c\frac{s}{\gamma^2 T}\{\log d + \sigma^2(w^2+24\log k)\}. \end{aligned}$$

Using the definition $\delta = 6\exp(-w^2/12)$, above bound holds with probability $1-\delta$. Simplifying the error in terms of δ by replacing w^2 with $12\log(6/\delta)$, gives us (23).

B.1 Proofs for Convergence within a Single Epoch for Algorithm 1

Lemma 3. *For $\bar{\theta}(T_i)$ defined in Algorithm 1 and $\hat{\theta}_i$ the optimal value for epoch i, let $\rho = c_1\sqrt{T_i}$, ρ_x some positive constant, $\rho_0 = \rho + \rho_x$ and $\tau = \rho$ where $c_1 = \frac{\sqrt{\log d}}{R_i}$. We have that*

$$\begin{aligned} & f(\bar{\theta}(T_i)) - f(\hat{\theta}_i) + \lambda_i\|\bar{y}(T_i)\|_1 - \lambda_i\|\hat{\theta}_i\|_1 \le \\ & \frac{R_i\sqrt{\log d}}{\sqrt{T_i}} + \frac{GR_i}{T_i} + \frac{\rho_x R_i^2}{T_i} + \frac{\sum_{k=1}^{T_i}\langle e_k, \hat{\theta}_i - \theta_k\rangle}{T_i}. \end{aligned} \tag{28}$$

Remark : Please note that as opposed to (Wang and Banerjee, 2013) we do not require $\rho_x \propto \sqrt{T_i}$. We show that our parameter setting also works.

Proof. First we show that our update rule for θ is equivalent to not linearizing f and using another Bregman divergence. This helps us in finding a better upper bound on error that does not require bounding the subgradient. Note that linearization does not change the nature of analysis. The reason is that we can define $B_f(\theta, \theta_k) = f(\theta) - f(\theta_k) + \langle \nabla f(\theta_k), \theta - \theta_k \rangle$, which means $f(\theta) - B_f(\theta, \theta_k) = f(\theta_k) + \langle \nabla f(\theta_k), \theta - \theta_k \rangle$.

Therefore,

$$\underset{\|\theta-\tilde{\theta}_i\|_1^2 \le R_i^2}{\arg\min} \; \{\langle \nabla f(\theta_k), \theta - \theta_k \rangle\} = \underset{\|\theta-\tilde{\theta}_i\|_1^2 \le R_i^2}{\arg\min} \; \{f(\theta) - B_f(\theta, \theta_k)\}.$$

As a result, we can write down the update rule of θ in REASON 1 as

$$\begin{aligned}\theta_{k+1} = \underset{\|\theta-\tilde{\theta}_i\|_1^2 \le R_i^2}{\arg\min} \; & \{f(\theta) - B_f(\theta, \theta_k) + z_k^T(\theta - y_k) + \rho B_\phi(\theta, y_k) \\ & + \rho_x B_{\phi_x'}(\theta, \theta_k)\}.\end{aligned}$$

We also have that $B_{\phi_x}(\theta, \theta_k) = B_{\phi_x'}(\theta, \theta_k) - \frac{1}{\rho_x} B_f(\theta, \theta_k)$, which simplifies the update rule to

$$\theta_{k+1} = \underset{\|\theta-\tilde{\theta}_i\|_1^2 \le R_i^2}{\arg\min} \; \{f(\theta) + \langle z_k, \theta - y_k \rangle + \rho B_\phi(\theta, y_k) + \rho_x B_{\phi_x}(\theta, \theta_k)\}. \tag{29}$$

We notice that equation (29) is equivalent to Equation (7) (Wang and Banerjee, 2013). Note that as opposed to (Wang and Banerjee, 2013), in our setting ρ_x can be set as a constant. Therefore, for completeness we provide proof of convergence and the convergence rate for our setting.

Lemma 4. *Convergence of REASON 1: The optimization problem defined in REASON 1 converges.*

Proof. On lines of (Wang and Banerjee, 2013), let $\mathbf{R}(k+1)$ stand for residuals of optimality condition. For convergence we need to show that $\lim_{k\to\infty} \mathbf{R}(k+1) = 0$. Let $w_k = (\theta_k, y_k, z_k)$. Define

$$D(w^*, w_k) = \frac{1}{\tau\rho}\|z^* - z_k\|_2^2 + B_\phi(y^*, y_k) + \frac{\rho_x}{\rho} B_\phi(\theta^*, \theta_k).$$

By Lemma 2 Wang and Banerjee (2013)

$$\mathbf{R}(t+1) \le D(w^*, w_k) - D(w^*, w_{k+1}).$$

Therefore,

$$\begin{aligned}\sum_{k=1}^{\infty} \mathbf{R}(t+1) &\le D(w^*, w_0) \\ &= \frac{1}{\tau\rho}\|z^*\|_2^2 + B_\phi(y^*, y_0) + \frac{\rho_x}{\rho} B_\phi(\theta^*, \theta_0) \\ &\le \lim_{T\to\infty} \frac{R_i^2}{\log d\; T}\|\nabla f(\theta^*)\|_2^2 + 2R_i^2 + \frac{\rho_x}{\sqrt{T}\log d} R_i^3.\end{aligned}$$

Therefore, $\lim_{k\to\infty} \mathbf{R}(k+1) = 0$ and the algorithm converges. □

If in addition we incorporate sampling error, then Lemma 1 (Wang and Banerjee, 2013) changes to

$$\begin{aligned}
&f(\theta_{k+1}) - f(\hat{\theta}_i) + \lambda_i \|y_{k+1}\|_1 - \lambda_i \|\hat{\theta}_i\|_1 \leq \\
&- \langle z_k, \theta_{k+1} - y_{k+1} \rangle - \frac{\rho}{2}\{\|\theta_{k+1} - y_k\|_2^2 + \|\theta_{k+1} - y_{k+1}\|_2^2\} + \langle e_k, \hat{\theta}_i - \theta_k \rangle \\
&+ \frac{\rho}{2}\{\|\hat{\theta}_i - y_k\|_2^2 - \|\hat{\theta}_i - y_{k+1}\|_2^2\} + \rho_x\{B_{\phi_x}(\hat{\theta}_i, \theta_k) - B_{\phi_x}(\hat{\theta}_i, \theta_{k+1}) \\
&- B_{\phi_x}(\theta_{k+1}, \theta_k)\}.
\end{aligned}$$

The above result follows from convexity of f, the update rule for θ (Equation (29)) and the three point property of Bregman divergence.

Next, we show the bound on the dual variable.

Lemma 5. *The dual variable in REASON 1 is bounded. i.e.,*

$$\|z_k\|_1 \leq G + 2\rho_0 R_i, \quad \textit{where} \quad \rho_0 := \rho_x + \rho.$$

Proof. Considering the update rule for θ, we have the Lagrangian

$$\mathcal{L} = f(\theta) + \langle z_k, \theta - y_k \rangle + \rho B_\phi(\theta, y_k) + \rho_x B_{\phi_x}(\theta, \theta_k) + \zeta \left(\|\theta_{k+1} - \tilde{\theta}_i\|_1 - R_i \right),$$

where ζ is the Lagrange multiplier corresponding to the ℓ_1 bound. We hereby emphasize that ζ does not play a role in size of the dual variable. i.e., considering the ℓ_1 constraint, three cases are possible:

1. $\|\theta_{k+1} - \tilde{\theta}_i\|_1 > R_i$. By complementary slackness, $\zeta = 0$.
2. $\|\theta_{k+1} - \tilde{\theta}_i\|_1 < R_i$. By complementary slackness, $\zeta = 0$.
3. $\|\theta_{k+1} - \tilde{\theta}_i\|_1 = R_i$. This case is equivalent to the non-constrained update and no projection will take place. Therefore, z will be the same as in the non-constrained update.

Having above analysis in mind, the upper bound on the dual variable can be found as follows By optimality condition on θ_{k+1}, we have

$$-z_k = \nabla f(\theta_{k+1}) + \rho_x(\theta_{k+1} - \theta_k) + \rho(\theta_{k+1} - y_k). \tag{30}$$

By definition of the dual variable and the fact that $\tau = \rho$, we have that

$$z_k = z_{k-1} - \rho(\theta_k - y_k)$$

Hence, we have that $-z_{k-1} = \nabla f(\theta_{k+1}) + (\rho_x + \rho)(\theta_{k+1} - \theta_k)$. Therefore,

$$\|z_{k-1}\|_1 \leq G + 2\rho_0 R_i, \quad \text{where} \quad \rho_0 := \rho_x + \rho.$$

It is easy to see that this is true for all z_k at each epoch. □

Consequently,

$$\begin{aligned}\frac{-1}{\tau}\langle z_k, z_k - z_{k+1}\rangle &= \frac{1}{\tau}\langle 0 - z_k, z_k - z_{k+1}\rangle \\ &= \frac{1}{2\tau}\left(\|z_{k+1}\|^2 - \|z_k\|^2 - \|z_{k+1} - z_k\|^2\right).\end{aligned}$$

Ignoring the negative term in the upper bound and noting $z_0 = 0$, we get

$$\begin{aligned}\frac{1}{T_i}\sum_{k=1}^{T_i} -\langle z_k, \theta_{k+1} - y_{k+1}\rangle \le \frac{1}{2\tau T_i}\|z_{T_i}\|^2 &\le \frac{1}{2\tau T_i}(G + 2\rho_0 R_i)^2 \\ &\simeq \frac{R_i\sqrt{\log d}}{\sqrt{T_i}} + \frac{GR_i}{T_i}.\end{aligned}$$

Note that since we consider the dominating terms in the final bound, terms with higher powers of T_i can be ignored throughout the proof. Next, following the same approach as in Theorem 4 (Wang and Banerjee, 2013) and considering the sampling error, we get,

$$\begin{aligned}&f(\bar{\theta}(T_i)) - f(\hat{\theta}_i) + \lambda_i\|\bar{y}(T_i)\|_1 - \lambda_i\|\hat{\theta}_i\|_1 \\ &\le \frac{R_i\sqrt{\log d}}{\sqrt{T_i}} + \frac{GR_i}{T_i} + \frac{c_1}{\sqrt{T_i}}\|\hat{\theta}_i - y_0\|_2^2 + \frac{\rho_x}{T_i}B_{\phi_x}(\hat{\theta}_i, \theta_0) + \frac{1}{T_i}\sum_{k=1}^{T_i}\langle e_k, \hat{\theta}_i - \theta_k\rangle.\end{aligned}$$

We have $\theta_0 = y_0 = \tilde{\theta}_i$ and $z_0 = 0$. Moreover, $B_{\phi_x}(\theta, \theta_k) = B_{\phi'_x}(\theta, \theta_k) - \frac{1}{\rho_x}B_f(\theta, \theta_k)$. Therefore,

$$\begin{aligned}&f(\bar{\theta}(T_i)) - f(\hat{\theta}_i) + \lambda_i\|\bar{y}(T_i)\|_1 - \lambda_i\|\hat{\theta}_i\|_1 \\ &\le \frac{R_i\sqrt{\log d}}{\sqrt{T_i}} + \frac{GR_i}{T_i} + \frac{c_1}{\sqrt{T_i}}\|\hat{\theta}_i - \tilde{\theta}_i\|_2^2 + \frac{\rho_x}{T_i}\{B_{\phi'_x}(\hat{\theta}_i, \tilde{\theta}_i) - B_f(\hat{\theta}_i, \tilde{\theta}_i)\} + \sum_{k=1}^{T_i}\langle e_k, \hat{\theta}_i - \theta_k\rangle \\ &\le \frac{R_i\sqrt{\log d}}{\sqrt{T_i}} + \frac{GR_i}{T_i} + \frac{\sqrt{\log d}}{R_i\sqrt{T_i}}\|\hat{\theta}_i - \tilde{\theta}_i\|_2^2 + \frac{\rho_x}{T_i}B_{\phi'_x}(\hat{\theta}_i, \tilde{\theta}_i) + \sum_{k=1}^{T_i}\langle e_k, \hat{\theta}_i - \theta_k\rangle.\end{aligned}$$

We note that $\rho_x B_{\phi'_x}(\hat{\theta}_i, \tilde{\theta}_i) = \frac{\rho_x}{2}\|\hat{\theta}_i - \tilde{\theta}_i\|_2^2$.
Considering the ℓ_2 terms, remember that for any vector x, if $s > r > 0$ then $\|x\|_s \le \|x\|_r$. Therefore,

$$\frac{\sqrt{\log d}}{R_i}\|\hat{\theta}_i - \tilde{\theta}_i\|_2^2 \le \frac{\sqrt{\log d}}{R_i}\|\hat{\theta}_i - \tilde{\theta}_i\|_1^2 \le \frac{\sqrt{\log d}}{R_i}R_i^2 = R_i\sqrt{\log d}.$$

□

B.2 Proof of Proposition 1: Inequality (24a)

Note the shorthand $e_k = \hat{g}_k - \nabla f(\theta_k)$, where $\hat{g}_k$ stands for empirically calculated subgradient of $f(\theta_k)$.

From Lemma 3, we have that

$$\begin{aligned}&f(\bar{\theta}(T_i)) - f(\hat{\theta}_i) + \lambda_i\|\bar{y}(T_i)\|_1 - \lambda_i\|\hat{\theta}_i\|_1 \\ &\le \frac{R_i\sqrt{\log d}}{\sqrt{T_i}} + \frac{GR_i}{T_i} + \frac{\rho_x R_i^2}{T_i} + \frac{\sum_{k=1}^{T_i}\langle e_k, \hat{\theta}_i - \theta_k\rangle}{T_i}.\end{aligned}$$

Using Lemma 7 from (Agarwal et al., 2012b), we have that

$$
\begin{aligned}
& f(\bar{\theta}(T_i)) - f(\hat{\theta}_i) + \lambda_i \|\bar{y}(T_i)\|_1 - \lambda_i \|\hat{\theta}_i\|_1 \\
& \leq \frac{R_i\sqrt{\log d}}{\sqrt{T_i}} + \frac{GR_i}{T_i} + \frac{\rho_x R_i^2}{T_i} + \frac{R_i \sigma_i w_i}{\sqrt{T_i}} \\
& = \frac{R_i\sqrt{\log d}}{\sqrt{T_i}} + \frac{GR_i}{T_i} + \frac{\rho_x R_i^2}{T_i} + \frac{R_i \sigma_i}{\sqrt{T_i}} \sqrt{12 \log(3/\delta_i)}.
\end{aligned}
$$

with probability at least $1 - \delta_i$. In the last equality we use $\delta_i = 3\exp(-w_i^2/12)$.

B.3 Proof of Lemma 1

Proof follows the same approach as Lemma 1 (Agarwal et al., 2012b). Note that since we assume exact sparsity the term $\|\theta^*_{S^c}\|_1$ is zero for our case and is thus eliminated. Needless to say, it is an straightforward generalization to consider approximate sparsity from this point.

B.4 Proof of Lemma 2

Using LSC assumption and the fact that $\hat{\theta}_i$ minimizes $f(\cdot) + \|\cdot\|_1$, we have that

$$
\begin{aligned}
\frac{\gamma}{2}\|\hat{\Delta}(T_i)\|_2^2 & \leq f(\bar{\theta}(T_i)) - f(\hat{\theta}(T_i)) + \lambda_i(\|\bar{y}(T_i)\|_1 - \|\hat{\theta}_i\|_1) \\
& \leq \frac{R_i\sqrt{\log d}}{\sqrt{T_i}} + \frac{GR_i}{T_i} + \frac{\rho_x R_i^2}{T_i} + \frac{R_i \sigma_i}{\sqrt{T_i}} \sqrt{12 \log \frac{3}{\delta_i}},
\end{aligned}
$$

with probability at least $1 - \delta_i$.

B.5 Proof of Proposition 1: Inequality $(24b)$

Throughout the proof, let $\Delta^*(T_i) = \bar{\theta}_i - \theta^*$ and $\hat{\Delta}(T_i) = \bar{\theta}_i - \hat{\theta}_i$, we have that $\Delta^*(T_i) - \hat{\Delta}(T_i) = \hat{\theta}_i - \theta^*$. Now we want to convert the error bound in $(24a)$ from function values into ℓ_1 and ℓ_2-norm bounds by exploiting the sparsity of θ^*. Since the error bound in $(24a)$ holds for the minimizer $\hat{\theta}_i$, it also holds for any other feasible vector. In particluar, applying it to θ^* leads to,

$$
\begin{aligned}
& f(\bar{\theta}(T_i)) - f(\theta^*) + \lambda_i \|\bar{y}(T_i)\|_1 - \lambda_i \|\theta^*\|_1 \\
& \leq \frac{R_i\sqrt{\log d}}{\sqrt{T_i}} + \frac{GR_i}{T_i} + \frac{\rho_x R_i^2}{T_i} + \frac{R_i \sigma_i}{\sqrt{T_i}} \sqrt{12 \log \frac{3}{\delta_i}},
\end{aligned}
$$

with probability at least $1 - \delta_i$.

For the next step, we find a lower bound on the left hand side of this inequality.

$$
\begin{aligned}
& f(\bar{\theta}(T_i)) - f(\theta^*) + \lambda_i \|\bar{y}(T_i)\|_1 - \lambda_i \|\theta^*\|_1 \geq \\
& f(\theta^*) - f(\theta^*) + \lambda_i \|\bar{y}(T_i)\|_1 - \lambda_i \|\theta^*\|_1 = \\
& \lambda_i \|\bar{y}(T_i)\|_1 - \lambda_i \|\theta^*\|_1,
\end{aligned}
$$

where the first inequality results from the fact that θ^* optimizes $f(\theta)$. Thus,

$$
\|\bar{y}(T_i)\|_1 \leq \|\theta^*\|_1 + \frac{R_i\sqrt{\log d}}{\lambda_i\sqrt{T_i}} + \frac{GR_i}{\lambda_i T_i} + \frac{\rho_x R_i^2}{\lambda_i T_i} + \frac{R_i \sigma_i}{\lambda_i\sqrt{T_i}} \sqrt{12 \log \frac{3}{\delta_i}}.
$$

Now we need a bound on $\|\bar{\theta}(T_i) - \bar{y}(T_i)\|_1$, we have

$$\begin{aligned}
\|\bar{\theta}(T_i) - \bar{y}(T_i)\|_1 &= \|\frac{1}{T_i}\sum_{k=0}^{T_i-1}(\theta_k - y_k)\|_1 \\
&= \|\frac{1}{\tau T_i}\sum_{k=0}^{T_i-1}(z_{k+1} - z_k)\|_1 \\
&= \frac{1}{\tau T_i}\|z_{T_i}\|_1 \\
&\le \frac{G + 2\rho_0 R_i}{T_i \tau} = \frac{GR_i}{T_i\sqrt{T_i}\sqrt{\log d}} + \frac{R_i}{T_i}.
\end{aligned}$$

By triangle inequality

$$\|\bar{\theta}(T_i)\|_1 - \|\bar{y}(T_i)\|_1 \le \|\bar{\theta}(T_i) - \bar{y}(T_i)\|_1,$$

Hence, after ignoring the dominated terms,

$$\|\bar{\theta}(T_i)\|_1 \le \|\theta^*\|_1 + \frac{R_i\sqrt{\log d}}{\lambda_i\sqrt{T_i}} + \frac{GR_i}{\lambda_i T_i} + \frac{\rho_x R_i^2}{\lambda_i T_i} + \frac{R_i\sigma_i}{\lambda_i\sqrt{T_i}}\sqrt{12\log(3/\delta_i)} + \frac{R_i}{T_i}.$$

By Lemma 6 in Agarwal et al. (2012b),

$$\|\Delta^*(T_i)_{S^c}\|_1 \le \|\Delta^*(T_i)_S\|_1 + \frac{R_i\sqrt{\log d}}{\lambda_i\sqrt{T_i}} + \frac{GR_i}{\lambda_i T_i} + \frac{\rho_x R_i^2}{\lambda_i T_i} + \frac{R_i\sigma_i}{\lambda_i\sqrt{T_i}}\sqrt{12\log(3/\delta_i)} + \frac{R_i}{T_i}.$$

with probability at least $1 - 3\exp(-w_i^2/12)$.

We have $\Delta^*(T_i) - \hat{\Delta}(T_i) = \hat{\theta}_i - \theta^*$. Therefore,

$$\begin{aligned}
&\|\hat{\theta}_i - \theta^*\|_1 = \\
&\|\Delta^*_S(T_i) - \hat{\Delta}_S(T_i)\|_1 + \|\Delta^*_{S^c}(T_i) - \hat{\Delta}_{S^c}(T_i)\|_1 \ge \\
&\{\|\Delta^*_S(T_i)\|_1 - \|\hat{\Delta}_S(T_i)\|_1\} - \{\|\Delta^*_{S^c}(T_i)\|_1 - \|\hat{\Delta}_{S^c}(T_i)\|_1\}.
\end{aligned}$$

Consequently,

$$\|\hat{\Delta}_{S^c}(T_i)\|_1 - \|\hat{\Delta}_S(T_i)\|_1 \le \|\Delta^*_{S^c}(T_i)\|_1 - \|\Delta^*_S(T_i)\|_1 + \|\hat{\theta}_i - \theta^*\|_1.$$

Using Equation ($25b$), we get

$$\|\hat{\Delta}_{S^c}(T_i)\|_1 \le \|\hat{\Delta}_S(T_i)\|_1 + \frac{8s\lambda_i}{\gamma} + \frac{R_i\sqrt{\log d}}{\lambda_i\sqrt{T_i}} + \frac{GR_i}{\lambda_i T_i} + \frac{\rho_x R_i^2}{\lambda_i T_i} + \frac{R_i\sigma_i}{\lambda_i\sqrt{T_i}}\sqrt{12\log(3/\delta_i)} + \frac{R_i}{T_i}.$$

Hence, further use of the inequality $\|\hat{\Delta}_S(T_i)\|_1 \le \sqrt{s}\|\hat{\Delta}(T_i)\|_2$ allows us to conclude that there exists a universal constant c such that

$$\|\hat{\Delta}(T_i)\|_1^2 \le 4s\|\hat{\Delta}(T_i)\|_2^2 + c\left[\frac{s^2\lambda_i^2}{\gamma^2} + \frac{R_i^2\log d}{\lambda_i^2 T_i} + \frac{G^2R_i^2}{\lambda_i^2T_i^2} + \frac{\rho_x^2R_i^4}{\lambda_i^2T_i^2} + \frac{12R_i^2\sigma_i^2\log(\frac{3}{\delta_i})}{T_i\lambda_i^2} + \frac{R_i^2}{T_i^2}\right], \tag{31}$$

with probability at least $1 - \delta_i$.

Optimizing the above bound with choice of λ_i gives us (22). From here on all equations hold with probability at least $1 - \delta_i$, we have

$$\begin{aligned}\|\hat{\Delta}(T_i)\|_1^2 \leq \frac{8s}{\gamma}&\Big[f(\bar{\theta}(T_i)) - f(\hat{\theta}(T_i)) + \lambda_i(\|\bar{Y}(T_i)\|_1 - \|\hat{\theta}_i\|_1)\Big]\\ &+\frac{2cs}{\gamma\sqrt{T_i}}\left[R_i\sqrt{\log d} + \frac{GR_i}{\sqrt{T_i}} + \frac{\rho_x R_i^2}{\sqrt{T_i}} + R_i\sigma_i\sqrt{12\log(\frac{3}{\delta_i})}\right] + \frac{R_i^2}{T_i^2}.\end{aligned}$$

Thus, for some other c, we have that

$$\|\hat{\Delta}(T_i)\|_1^2 \leq c\frac{s}{\gamma}\left[\frac{R_i\sqrt{\log d}}{\sqrt{T_i}} + \frac{GR_i}{T_i} + \frac{\rho_x R_i^2}{T_i} + \frac{R_i\sigma_i}{\sqrt{T_i}}\sqrt{12\log(\frac{3}{\delta_i})}\right] + \frac{R_i^2}{T_i^2}. \tag{32}$$

Combining the above inequality with error bound $(25b)$ for $\hat{\theta}_i$ and using triangle inequality leads to

$$\begin{aligned}\|\Delta^*(T_i)\|_1^2 &\leq 2\|\hat{\Delta}(T_i)\|_1^2 + 2\|\theta^* - \hat{\theta}_i\|_1^2\\ &\leq 2\|\hat{\Delta}(T_i)\|_1^2 + \frac{64}{\gamma^2}s^2\lambda_i^2\\ &\leq c'\frac{s}{\gamma}\left[\frac{R_i\sqrt{\log d}}{\sqrt{T_i}} + \frac{GR_i}{T_i} + \frac{\rho_x R_i^2}{T_i} + \frac{R_i\sigma_i}{\sqrt{T_i}}\sqrt{12\log\frac{3}{\delta_i}}\right] + \frac{R_i^2}{T_i^2}.\end{aligned}$$

Finally, in order to use $\bar{\theta}(T_i)$ as the next prox center $\tilde{\theta}_{i+1}$, we would also like to control the error $\|\bar{\theta}(T_i) - \hat{\theta}_{i+1}\|_1^2$. Since $\lambda_{i+1} \leq \lambda_i$ by assumption, we obtain the same form of error bound as in (32). We want to run the epoch till all these error terms drop to $R_{i+1}^2 := R_i^2/2$. Therefore, we set the epoch length T_i to ensure that. All above conditions are met if we choose the epoch length

$$T_i = C\left[\frac{s^2}{\gamma^2}\left[\frac{\log d + 12\sigma_i^2\log(3/\delta_i)}{R_i^2}\right] + \frac{sG}{\gamma R_i} + \frac{s}{\gamma}\rho_x\right], \tag{33}$$

for a suitably large universal constant C. Note that since we consider the dominating terms in the final bound, the last two terms can be ignored. By design of T_i, we have that

$$\|\Delta^*(T_i)\|_1^2 \leq \frac{c'}{\sqrt{C}}R_i^2,$$

which completes this proof.

B.6 Proof of Guarantees with Fixed Epoch Length, Sparse Case

This is a special case of Theorem 3 (Appendix). The key difference between this case and optimal epoch length setting of Theorem 3 is that in the latter we guaranteed error halving by the end of each epoch whereas with fixed epoch length that statement may not be possible after the number of epochs becomes large enough. Therefore, we need to show that in such case the error does not increase much to invalidate our analysis. Let k^* be the epoch number such that error halving holds true until then. Next we demonstrate that error does not increase much for $k > k^*$.

Given a fixed epoch length $T_0 = \mathcal{O}(\log d)$, we define

$$k^* := \sup\left\{ i : 2^{j/2+1} \leq \frac{cR_1\gamma}{s}\sqrt{\frac{T_0}{\log d + \sigma_i^2 w^2}} \text{ for all epochs } j \leq i \right\}, \tag{34}$$

where $w = \log(6/\delta)$.

First we show that if we run REASON 1 with fixed epoch length T_0 it has error halving behavior for the first k^* epochs.

Lemma 6. *For $T_0 = \mathcal{O}(\log d)$ and k^* as in (34), we have*

$$\|\tilde{\theta}_k - \theta^*\|_1 \leq R_k \quad \textit{and} \quad \|\tilde{\theta}_k - \bar{\theta}_k\|_1 \leq R_k \quad \textit{for all} \quad 1 \leq k \leq k^* + 1.$$

with probability at least $1 - 3k\exp(-w^2/12)$. Under the same conditions, there exists a universal constant c such that

$$\|\tilde{\theta}_k - \theta^*\|_2 \leq c\frac{R_k}{\sqrt{s}} \quad \textit{and} \quad \|\tilde{\theta}_k - \bar{\theta}_k\|_2 \leq c\frac{R_k}{\sqrt{s}} \quad \textit{for all} \quad 2 \leq k \leq k^* + 1.$$

Next, we analyze the behavior of REASON 1 after the first k^* epochs. Since we cannot guarantee error halving, we can also not guarantee that θ^* remains feasible at later epochs. We use Lemma 7 to control the error after the first k^* epochs.

Lemma 7. *Suppose that Assumptions $A1 - A3$ in the main text are satisfied at epochs $i = 1, 2, \ldots$. Assume that at some epoch k, the epoch center $\tilde{\theta}_k$ satisfies the bound $\|\tilde{\theta}_k - \theta^*\|_2 \leq c_1 R_k/\sqrt{s}$ and that for all epochs $j \geq k$, the epoch lengths satisfy the bounds*

$$\frac{s}{\gamma}\sqrt{\frac{\log d + \sigma_i^2 w_i^2}{T_j}} \leq \frac{R_k}{2} \quad \textit{and} \quad \frac{\log d}{T_i} \leq c_2.$$

Then for all epochs $j \geq k$, we have the error bound $\|q_j - \theta^*\|_2^2 \leq c_2 \frac{R_k^2}{s}$ with probability at least $1 - 3\sum_{i=k+1}^{j} \exp(-w_i^2/12)$.

In order to check the condition on epoch length in Lemma 7, we notice that with k^* as in (34), we have

$$c\frac{s}{\gamma}\sqrt{\frac{\log d + \sigma_i^2 w^2}{T_0}} \leq R_1 2^{-k^*/2-1} = \frac{R_{k^*+1}}{2}.$$

Since we assume that constants σ_k are decreasing in k, the inequality also holds for $k \geq k^* + 1$, therefore Lemma 7 applies in this setting.

The setting of epoch length in Theorem 1 ensures that the total number of epochs we perform is

$$k_0 = \log\left(\frac{R_1\gamma}{s}\sqrt{\frac{T}{\log d + \sigma^2 w^2}}\right).$$

Now we have two possibilities. Either $k_0 \leq k^*$ or $k_0 \geq k^*$. In the former, Lemma 6 ensures that the error bound $\|\tilde{\theta}_{k_0} - \theta^*\|_2^2 \leq cR_{k_0}^2/s$. In the latter case, we use Lemma 7 and get the error bound $cR_{k^*}^2/s$. Substituting values of k_0, k^* in these bounds completes the proof.

Proof of Lemma 6 and Lemma 7 follows directly from that of Lemma 5 and Lemma 3 in (Agarwal et al., 2012b).

B.7 Proof of Guarantees for Sparse Graphical Model selection Problem

Here we prove Corollary 1. According to C.1, in order to prove guarantees, we first need to bound $\|z_{k+1}-z_k\|_1$ and $\|z_k\|_\infty$. According to Equation (30) and considering the imposed ℓ_1 bound, this is equivalent to bound $\|g_{k+1}-g_k\|_1$ and $\|g_k\|_\infty$. The rest of the proof follows on lines of Theorem 1 proof. On the other hand, Lipschitz property requires a bound on $\|g_k\|_1$, which is much more stringent.

Assuming we are in a close proximity of Θ^*, we can use Taylor approximation to locally approximate Θ^{-1} by Θ^{*-1} as in (Ravikumar et al., 2011)

$$\Theta^{-1} = \Theta^{*-1} - \Theta^{*-1}\Delta\Theta^{*-1} + \mathcal{R}(\Delta),$$

where $\Delta = \Theta - \Theta^*$ and $\mathcal{R}(\Delta)$ is the remainder term. We have

$$\|g_{k+1}-g_k\|_1 \le |\!|\!|\Gamma^*|\!|\!|_\infty \|\Theta_{k+1}-\Theta_k\|_1,$$

and

$$\begin{aligned}\|g_k\|_\infty &\le \|g_k - \mathbb{E}(g_k)\|_\infty + \|\mathbb{E}(g_k)\|_\infty \\ &\le \|e_k\|_\infty + \|\Sigma^* - \Theta_k^{-1}\|_\infty \le \sigma + \|\Gamma^*\|_\infty \|\Theta_{k+1}-\Theta_k\|_1.\end{aligned}$$

The term $\|\Theta_{k+1}-\Theta_k\|_1$ is bounded by $2R_i$ by construction. We assume $|\!|\!|\Gamma^*|\!|\!|_\infty$ and $\|\Gamma^*\|_\infty$ are bounded.

The error Δ needs to be "small enough" for the $\mathcal{R}(\Delta)$ to be negligible, and we now provide the conditions for this. By definition, $\mathcal{R}(\Delta) = \sum_{k=2}^{\infty}(-1)^k(\Theta^{*-1}\Delta)^k\Theta^{*-1}$. Using triangle inequality and sub-multiplicative property for Frobenious norm,

$$\|\mathcal{R}(\Delta)\|_{\mathbb{F}} \le \frac{\|\Theta^{*-1}\|_{\mathbb{F}}\|\Delta\Theta^{*-1}\|_{\mathbb{F}}^2}{1-\|\Delta\Theta^{*-1}\|_{\mathbb{F}}}.$$

For $\|\Delta\|_{\mathbb{F}} \le 2R_i \le \frac{0.5}{\|\Theta^{*-1}\|_{\mathbb{F}}}$, we get

$$\|\mathcal{R}(\Delta)\|_{\mathbb{F}} \le \|\Theta^{*-1}\|_{\mathbb{F}}.$$

We assume $\|\Sigma^*\|_{\mathbb{F}}$ is bounded.

Note that $\{R_i\}_{i=1}^{k_T}$ is a decreasing sequence and we only need to bound R_1. Therefore, if the variables are closely-related we need to start with a small R_1. For weaker correlations, we can start in a bigger ball. The rest of the proof follows the lines of proof for Theorem 3, by replacing G^2 by $|\!|\!|\Gamma^*|\!|\!|_\infty R_i(\sigma + \|\Gamma^*\|_\infty R_i)$. Ignoring the higher order terms gives us Corollary 1.

C Guarantees for REASON 2

First, we provide guarantees for the theoretical case such that epoch length depends on epoch radius. This provides intuition on how the algorithm is designed. The fixed-epoch algorithm is a special case of this general framework. We first state and prove guarantees for general framework.

Next, we leverage these results to prove Theorem 1. Let the design parameters be set as

$$
\begin{aligned}
T_i &\simeq C\left[(s+r+\frac{s+r}{\gamma})^2\left(\frac{\log p+\beta^2(p)\sigma_i^2\log(6/\delta_i)+}{R_i^2}\right)+(s+r+\frac{s+r}{\gamma})\left(\frac{G}{R_i}+\rho_x\right)\right], \qquad (35)\\
\lambda_i^2 &= \frac{\gamma}{(s+r)\sqrt{T_i}}\sqrt{(R_i^2+\tilde{R}_i^2)\log p+\frac{G^2(R_i^2+\tilde{R}_i^2)}{T_i}+\beta^2(p)(R_i^2+\tilde{R}_i^2)\sigma_i^2\log\frac{3}{\delta_i}}\\
&\quad+\frac{\rho_x(R_i^2+\tilde{R}_i^2)}{T_i}+\frac{\alpha^2}{p^2}+\frac{\beta^2(p)\sigma^2}{T_i}\left(\log p+\log\frac{1}{\delta_i}\right),
\end{aligned}
$$

$$
\mu_i^2=c_\mu\lambda_i^2,\quad \rho\propto\sqrt{\frac{T_i\log p}{R_i^2+\tilde{R}_i^2}},\quad \rho_x>0,\quad \tau=\rho.
$$

Theorem 4. *Under assumptions* $A2-A6$ *and parameter settings as in* (35)*, there exists a constant* $c_0>0$ *such that REASON 2 satisfies the following for all* $T>k_T$,

$$
\begin{aligned}
&\|\bar{S}(T)-S^*\|_{\mathbb{F}}^2+\|\bar{L}(T)-L^*\|_{\mathbb{F}}^2\leq\\
&\frac{c_0(s+r)}{T}\left[\log p+\beta^2(p)\sigma^2\left(w^2+\log k_T\right)\right]+\left(1+\frac{s+r}{\gamma^2 p}\right)\frac{\alpha^2}{p}.
\end{aligned}
$$

with probability at least $1-6\exp(-w^2/12)$ *and*

$$
k_T\simeq-\log\left(\frac{(s+r)^2}{\gamma^2R_1^2T}\left[\log p+\beta^2(p)\sigma^2w^2\right]\right).
$$

For Proof outline and detailed proof of Theorem 4 see Appendix C.1 and D respectively.

C.1 Proof outline for Theorem 4

The foundation block for this proof is Proposition 2.

Proposition 2. *Suppose* f *satisfies Assumptions* $A1-A6$ *with parameters* γ *and* σ_i *respectively and assume that* $\|S^*-\tilde{S}_i\|_1^2\leq R_i^2$, $\|L^*-\tilde{L}_i\|_1^2\leq\tilde{R}_i^2$. *We apply the updates in REASON 2 with parameters as in* (35)*. Then, there exists a universal constant* c *such that for any radius* $R_i,\tilde{R}_i$, $\tilde{R}_i=c_rR_i, 0\leq c_r\leq 1$,

$$
\begin{aligned}
&f(\bar{M}(T_i))+\lambda_i\phi(\bar{W}(T_i))-f(\hat{M}_i)-\lambda_i\phi(\hat{W}(T_i)) \qquad (36a)\\
&\leq\sqrt{\frac{R_i^2+\tilde{R}_i^2}{T_i}}\sqrt{\log p}+\frac{R_i^2+\tilde{R}_i^2}{T_i}\rho_x+\frac{G\sqrt{R_i^2+\tilde{R}_i^2}}{T_i}\\
&\quad+\frac{\beta(p)(R_i+\tilde{R}_i)\sigma_i\sqrt{12\log\frac{3}{\delta_i}}}{\sqrt{T_i}}+\frac{\beta(p)G(R_i+\tilde{R}_i)\sigma_i\sqrt{12\log\frac{3}{\delta_i}}}{T_i\sqrt{\log p}},\\
&\|\bar{S}(T_i)-S^*\|_1^2\leq\frac{c'}{\sqrt{C}}R_i^2+c(s+r+\frac{(s+r)^2}{p\gamma^2})\frac{\alpha^2}{p}, \qquad (36b)\\
&\|\bar{L}(T_i)-L^*\|_*^2\leq\frac{c'}{\sqrt{C}}\frac{1}{1+\gamma}R_i^2+c\frac{(s+r)^2}{p\gamma^2}\frac{\alpha^2}{p}.
\end{aligned}
$$

where both bounds are valid with probability at least $1-\delta_i$.

In order to prove Proposition 2, we need two more lemmas.

To move forward, we use the following notations: $\Delta(T_i) = \hat{S}_i - S^* + \hat{L}_i - L^*$, $\Delta^*(T_i) = \bar{S}(T_i) - S^* + \bar{L}(T_i) - L^*$ and $\hat{\Delta}(T_i) = \bar{S}_i - \hat{S}_i + \bar{L}_i - \hat{L}_i$. In addition $\Delta_S(T_i) = \hat{S}_i - S^*$, with alike notations for $\Delta_L(T_i)$. For on and off support part of $\Delta(T_i)$, we use $(\Delta(T_i))_{supp}$ and $(\Delta(T_i))_{supp^c}$.

Lemma 8. *At epoch i assume that $\|S^* - \tilde{S}\|_1^2 \le R_i^2$, $\|L^* - \tilde{L}\|_1^2 \le \tilde{R}_i^2$. Then the errors $\Delta_S(T_i), \Delta_L(T_i)$ satisfy the bound*

$$\|\hat{S}_i - S^*\|_{\mathbb{F}}^2 + \|\hat{L}_i - L^*\|_{\mathbb{F}}^2 \le c\{s\frac{\lambda_i^2}{\gamma^2} + r\frac{\mu_i^2}{\gamma^2}\}.$$

Lemma 9. *Under the conditions of Proposition 2 and with parameter settings* (35), (35), *we have*

$$\begin{aligned}
&\|\hat{S}_i - \bar{S}(T_i)\|_{\mathbb{F}}^2 + \|\hat{L}_i - \bar{L}(T_i)\|_{\mathbb{F}}^2 \\
&\le \frac{2}{\gamma}\left(\sqrt{\frac{R_i^2 + \tilde{R}_i^2}{T_i}}\sqrt{\log p} + \frac{R_i^2 + \tilde{R}_i^2}{T_i}\rho_x + \frac{G\sqrt{R_i^2 + \tilde{R}_i^2}}{T_i}\right. \\
&\quad \left. + \frac{\beta(p)(R_i + \tilde{R}_i)\sigma_i\sqrt{12\log(3/\delta_i)}}{\sqrt{T_i}} + \frac{\beta(p)G(R_i + \tilde{R}_i)\sigma_i\sqrt{12\log(3/\delta_i)}}{T - i\sqrt{\log p}}\right) + (\frac{2\alpha}{\sqrt{p}} + \frac{p}{\tau T_i})^2,
\end{aligned}$$

with probability at least $1 - \delta_i$.

D Proof of Theorem 4

The first step is to ensure that $\|S^* - \tilde{S}_i\|_1^2 \le R_i^2$, $\|L^* - \tilde{L}_i\|_1^2 \le \tilde{R}_i^2$ holds at each epoch so that Proposition 2 can be applied in a recursive manner. We prove this in the same manner we proved Theorem 1, by induction on the epoch index. By construction, this bound holds at the first epoch. Assume that it holds for epoch i. Recall that T_i is defined by (35) where $C \ge 1$ is a constant we can choose. By substituting this T_i in inequality (36b), the simplified bound (36b) further yields

$$\|\Delta_S^*(T_i)\|_1^2 \le \frac{c'}{\sqrt{C}}R_i^2 + c(s + r + \frac{(s+r)^2}{p\gamma^2})\frac{\alpha^2}{p},$$

Thus, by choosing C sufficiently large, we can ensure that $\|\bar{S}(T_i) - S^*\|_1^2 \le R_i^2/2 := R_{i+1}^2$. Consequently, if S^* is feasible at epoch i, it stays feasible at epoch $i+1$. Hence, we guaranteed the feasibility of S^* throughout the run of algorithm by induction. As a result, Lemma 8 and 9 apply and for $\tilde{R}_i = c_r R_i$, we find that

$$\|\Delta_S^*(T_i)\|_{\mathbb{F}}^2 \le \frac{1}{s+r}R_i^2 + (1 + \frac{s+r}{\gamma^2 p})\frac{2\alpha^2}{p}.$$

The bound holds with probability at least $1 - 3\exp(-w_i^2/12)$. The same is true for $\|\Delta_L^*(T_i)\|_{\mathbb{F}}^2$. Recall that $R_i^2 = R_1^2 2^{-(i-1)}$. Since $w_i^2 = w^2 + 24\log i$, we can apply union bound to simplify the error probability as $1 - 6\exp(-w^2/12)$. Let $\delta = 6\exp(-w^2/12)$, we write the bound in terms of δ, using $w^2 = 12\log(6/\delta)$.

Next we convert the error bound from its dependence on the number of epochs k_T to the number of iterations needed to complete k_T epochs, i.e. $T(K) = \sum_{i=1}^{k} T_i$. Using the same approach as in proof of Theorem 3, we get

$$k_T \simeq -\log \frac{(s+r+(s+r)/\gamma)^2}{R_1^2 T} - \log\left[\log p + 12\beta^2(p)\sigma^2 w^2\right].$$

As a result

$$\|\Delta_S^*(T_i)\|_{\mathbb{F}}^2 \leq \frac{C(s+r)}{T}\left[\log p + \beta^2(p)\sigma^2\left(w^2 + \log k_T\right)\right]\right] + \frac{\alpha^2}{p}.$$

For the low-rank part, we proved feasibility in proof of Equation $(36b)$, consequently The same bound holds for $\|\Delta_L^*(T_i)\|_{\mathbb{F}}^2$.

D.1 Proofs for Convergence within a Single Epoch for Algorithm 2

We showed that our method is equivalent to running Bregman ADMM on M and $W = [S; L]$. Consequently, our previous analysis for sparse case holds true for the error bound on sum of loss function and regularizers within a single epoch. With $\rho = c_2\sqrt{T_i}, \tau = \rho, c_2 = \frac{\sqrt{\log p}}{\sqrt{R_i^2 + \tilde{R}_i^2}}$. We use the same approach as in Section B.1 for bounds on dual variable Z_k. Hence,

$$\begin{aligned}
&f(\bar{M}(T_i)) + \lambda_i\phi(\bar{W}(T_i)) - f(\hat{M}_i) - \lambda_i\phi(\hat{W}(T_i)) \\
&\leq \frac{c_2\|A\hat{W}(T_i) - AW_0\|_{\mathbb{F}}^2}{\sqrt{T_i}} + \frac{\rho_x\|\hat{M}(T_i) - M_0\|_{\mathbb{F}}^2}{T_i} + \frac{GR_i}{T_i} + \frac{R_i\sqrt{\log p}}{\sqrt{T_i}} \\
&\quad + \frac{\sum_{k=1}^{T_i} \operatorname{Tr}(E_k, \hat{M}_i - M_k)}{T_i} \\
&\leq \left[\frac{c_2}{\sqrt{T_i}} + \frac{\rho_x}{T_i}\right]\|\hat{S}_i - \tilde{S}_i + \hat{L}_i - \tilde{L}_i\|_{\mathbb{F}}^2 + \frac{GR_i}{T_i} + \frac{R_i\sqrt{\log p}}{\sqrt{T_i}} \\
&\quad + \frac{\sum_{k=1}^{T_i} \operatorname{Tr}(E_k, \hat{M}_i - M_k)}{T_i}.
\end{aligned}$$

By the constraints enforced in the algorithm, we have

$$\begin{aligned}
&f(\bar{M}(T_i)) + \lambda_i\phi(\bar{W}(T_i)) - f(\hat{M}_i) - \lambda_i\phi(\hat{W}(T_i)) \\
&\leq \sqrt{\frac{R_i^2 + \tilde{R}_i^2}{T_i}}\sqrt{\log p} + \frac{R_i^2 + \tilde{R}_i^2}{T_i}\rho_x + \frac{G\sqrt{R_i^2 + \tilde{R}_i^2}}{T_i} + \frac{\sum_{k=1}^{T_i} \operatorname{Tr}(E_k, \hat{M}_i - M_k)}{T_i}.
\end{aligned}$$

Lemma 10. *The dual variable in REASON 2 is bounded. i.e.,*

$$\|Z_k\|_1 \leq G + 2\rho_0 R_i, \quad \textit{where} \quad \rho_0 := \rho_x + \rho.$$

Proof. The proof follows the same line as in proof of Lemma 5 and replacing θ, y by M, W where $W = [S; L]$. Hence,

$$\|Z_k\|_1 \leq G + 2\rho_0 R_i, \quad \text{where} \quad \rho_0 := \rho_x + \rho.$$

□

D.2 Proof of Proposition 2: Equation $(36a)$

In this section we bound the term $\frac{\sum_{k=1}^{T_i} \mathrm{Tr}(E_k, \hat{M}_i - M_k)}{T_i}$. We have

$$M_k - \hat{M}_i = S_k - \hat{S}_i + L_k - \hat{L}_i + (Z_{k+1} - Z_k)/\tau.$$

Hence,

$$\begin{aligned}
&[\mathrm{Tr}(E_k, \hat{M}_i - M_k)]^2 \\
&\leq [\|E_k\|_\infty \|S_k - \hat{S}_i\|_1 + \|E_k\|_2^2 \|L_k - \hat{L}_i\|_* + \|E_k\|_\infty \|(Z_{k+1} - Z_k)/\tau\|_1]^2 \\
&\leq [2R_i \|E_k\|_\infty + 2\tilde{R}_i \|E_k\|_2 + (G + 2\rho_0 R_i)/\tau \|E_k\|_\infty]^2 \\
&\leq \|E_k\|_2^2 [2R_i + 2\tilde{R}_i + (G + 2\rho_0 R_i)/\tau]^2.
\end{aligned}$$

Consider the term $\|E_k\|_2$. Using Assumption A4, our previous approach in proof of Equation $(24a)$, holds true with addition of a $\beta(p)$ term. Consequently,

$$\begin{aligned}
&f(\bar{M}(T_i)) + \lambda_i \phi(\bar{W}(T_i)) - f(\hat{M}_i) - \lambda_i \phi(\hat{W}(T_i)) \\
&\leq \sqrt{\frac{R_i^2 + \tilde{R}_i^2}{T_i}} \sqrt{\log p} + \frac{R_i^2 + \tilde{R}_i^2}{T_i} \rho_x + \frac{G\sqrt{R_i^2 + \tilde{R}_i^2}}{T_i} \\
&\quad + \frac{\beta(p)(R_i + \tilde{R}_i)\sigma_i \sqrt{12 \log(3/\delta_i)}}{\sqrt{T_i}} + \frac{\beta(p) G (R_i + \tilde{R}_i)\sigma_i \sqrt{12 \log(3/\delta_i)}}{T_i \sqrt{\log p}}.
\end{aligned}$$

with probability at least $1 - \delta_i$.

D.3 Proof of Lemma 8

We use Lemma 1 (Negahban et al., 2012) for designing λ_i and μ_i. This Lemma requires that for optimization problem $\min\limits_\Theta \{L(\Theta) + \lambda_i Q(\Theta)\}$, we design the regularizer coefficient $\lambda_i \geq 2Q^*(\nabla L(\Theta^*))$, where L is the loss function, Q is the regularizer and Q^* is the dual regularizer. For our case Θ stands for $[S; L]$.

$$L(\Theta) = \frac{1}{n} \sum_{k=1}^{n} f_k(\Theta, x),$$

and

$$\begin{aligned}
Q^*(\nabla L(\Theta^*)) &= Q^* \left[\mathbb{E}(\nabla f(\Theta^*) + \frac{1}{n} \sum_{k=1}^{n} \{\nabla f_k(\Theta^*)) - \mathbb{E}(\nabla f(\Theta^*))\} \right] \\
&= Q^*(\frac{1}{n} \sum_{k=1}^{n} E_k),
\end{aligned}$$

where $E_k = g_k - \mathbb{E}(g_k)$ is the error in gradient estimation as defined earlier.
Using Theorem 1 (Agarwal et al., 2012a) in this case, if we design

$$\lambda_i \geq 4 \left\| \frac{1}{n} \sum_{k=1}^{n} E_k \right\|_\infty + \frac{4\gamma\alpha}{p} \quad \text{and} \quad \mu_i \geq 4 \left\| \frac{1}{n} \sum_{k=1}^{n} E_k \right\|_2, \tag{37}$$

then we have

$$\|\hat{S}_i - S^*\|_{\mathbb{F}}^2 + \|\hat{L}_i - L^*\|_{\mathbb{F}}^2 \leq c\{s\frac{\lambda_i^2}{\gamma^2} + r\frac{\mu_i^2}{\gamma^2}\}. \tag{38}$$

Lemma 11. *Assume $X \in \mathbb{R}^{p\times p}$. If $\|X\|_2 \leq B$ almost surely then with probability at least $1-\delta$ we have*

$$\left\| \frac{1}{n}\sum_{k=1}^{n} X_k - \mathbb{E}(X_k) \right\|_2 \leq \frac{6B}{\sqrt{n}}\left(\sqrt{\log p} + \sqrt{\log\frac{1}{\delta}}\right).$$

Note that this lemma is matrix Hoeffding bound and provides a loose bound on matrix. Whereas using matrix Bernstein provided tighter results using $\mathbb{E}(E_k E_k^\top)$. Moreover, since the elementwise max norm $\|X\|_\infty \leq \|X\|_2$, we use the same upper bound for both norms.

By definition $\mathbb{E}(E_k) = 0$. According to Assumption A4, $\|E_k\|_2 \leq \beta(p)\sigma$. Thus it suffices to design

$$\lambda_i \geq \frac{24\beta(p)\sigma_i}{\sqrt{T_i}}\left(\sqrt{\log p} + \sqrt{\log\frac{1}{\delta_i}}\right) + \frac{4\gamma\alpha}{p}$$

and

$$\mu_i \geq \frac{24\beta(p)\sigma_i}{\sqrt{T_i}}\left(\sqrt{\log p} + \sqrt{\log\frac{1}{\delta_i}}\right).$$

Then, we can use Equation (38).

D.4 Proof of Lemma 9

By LSC condition on $X = S + L$

$$\frac{\gamma}{2}\|\hat{S}_i - \bar{S}(T_i) + \hat{L}_i - \bar{L}(T_i)\|_{\mathbb{F}}^2$$
$$\leq f(\bar{X}(T_i)) + \lambda_i\|\bar{S}(T_i)\|_1 + \mu_i\|\bar{L}(T_i)\|_* - f(\hat{X}_i) - \lambda_i\|\hat{S}(T_i)\|_1 - \mu_i\|\hat{L}(T_i)\|_*$$

We want to use the following upper bound for the above term.

$$f(\bar{M}(T_i)) + \lambda_i\phi(\bar{X}(T_i)) - f(\hat{M}_i) - \lambda_i\phi(\hat{X}(T_i)) \leq$$
$$\sqrt{\frac{R_i^2 + \tilde{R}_i^2}{T_i}}\sqrt{\log p} + \frac{R_i^2 + \tilde{R}_i^2}{T_i}\rho_x + \frac{G\sqrt{R_i^2 + \tilde{R}_i^2}}{T_i}$$
$$+\frac{\beta(p)(R_i + \tilde{R}_i)\sigma_i\sqrt{12\log\frac{3}{\delta_i}}}{\sqrt{T_i}} + \frac{\beta(p)G(R_i + \tilde{R}_i)\sigma_i\sqrt{12\log\frac{3}{\delta_i}}}{T_i},$$

$\hat{M}_i = \hat{X}_i$, i.e., all the terms are the same except for $f(\bar{M}(T_i)), f(\bar{X}(T_i))$. We have $\bar{M}(T_i) = \bar{X}(T_i) + \frac{Z_T}{\tau T_i}$. This is a bounded and small term $\mathcal{O}(R_i/(T_i\sqrt{T_i}))$. We accept this approximation giving the fact that this is a higher order term compared to $\mathcal{O}(1/\sqrt{T_i})$. Hence, it will not play a role in the final bound on the convergence rate. Therefore,

$$\begin{aligned}
&\frac{\gamma}{2}\|\hat{S}_i - \bar{S}(T_i) + \hat{L}_i - \bar{L}(T_i)\|_{\mathbb{F}}^2 \\
&\leq \sqrt{\frac{R_i^2 + \tilde{R}_i^2}{T_i}}\sqrt{\log p} + \frac{R_i^2 + \tilde{R}_i^2}{T_i}\rho_x + \frac{G\sqrt{R_i^2 + \tilde{R}_i^2}}{T_i} \\
&\quad + \frac{\beta(p)(R_i + \tilde{R}_i)\sigma_i\sqrt{12\log\frac{3}{\delta_i}}}{\sqrt{T_i}} + \frac{\beta(p)G(R_i + \tilde{R}_i)\sigma_i\sqrt{12\log\frac{3}{\delta_i}}}{T_i\sqrt{\log p}},
\end{aligned} \tag{39}$$

with probability at least $1 - \delta_i$.

For simplicity, we use

$$\begin{aligned}
H_1 = &\sqrt{\frac{R_i^2 + \tilde{R}_i^2}{T_i}}\sqrt{\log p} + \frac{R_i^2 + \tilde{R}_i^2}{T_i}\rho_x + \frac{G\sqrt{R_i^2 + \tilde{R}_i^2}}{T_i} \\
&+ \frac{\beta(p)(R_i + \tilde{R}_i)\sigma_i\sqrt{12\log\frac{3}{\delta_i}}}{\sqrt{T_i}} + \frac{\beta(p)G(R_i + \tilde{R}_i)\sigma_i\sqrt{12\log\frac{3}{\delta_i}}}{T_i\sqrt{\log p}}.
\end{aligned}$$

We have,

$$-\frac{\gamma}{2}\,\mathrm{Tr}(\hat{\Delta}_S\hat{\Delta}_L) = \frac{\gamma}{2}\{\|\hat{\Delta}_S\|_{\mathbb{F}}^2 + \|\hat{\Delta}_L\|_{\mathbb{F}}^2\} - \frac{\gamma}{2}\{\|\hat{\Delta}_S + \hat{\Delta}_L\|_{\mathbb{F}}^2\},$$

In addition,

$$\gamma\|\,\mathrm{Tr}(\hat{\Delta}_S(T_i)\hat{\Delta}_L(T_i))| \leq \gamma\|\hat{\Delta}_S(T_i)\|_1\|\hat{\Delta}_L(T_i)\|_\infty.$$

We have,

$$\|\hat{\Delta}_L(T_i)\|_\infty \leq \|\hat{L}_i\|_\infty + \|\bar{L}(T_i)\|_\infty$$

$$\begin{aligned}
\|\bar{L}(T_i)\|_\infty &\leq \|\bar{Y}(T_i)\|_\infty + \|\bar{L}(T_i) - \bar{Y}(T_i)\|_\infty \\
&\leq \|\bar{Y}(T_i)\|_\infty + \|\frac{\sum_{k=0}^{T_i-1}(L_k - Y_k)}{T_i}\|_\infty \\
&= \|\bar{Y}(T_i)\|_\infty + \|\frac{\sum_{k=0}^{T_i-1}(U_k - U_{k+1})}{\tau T_i}\|_\infty \\
&= \|\bar{Y}(T_i)\|_\infty + \|\frac{-U_{k+1}}{\tau T_i}\|_\infty \\
&\leq \frac{\alpha}{p} + \frac{\sqrt{p}}{\tau T_i}.
\end{aligned}$$

In the last step we incorporated the constraint $\|Y\|_\infty \leq \frac{\alpha}{p}$, and the fact that $U_0 = 0$. Moreover, we used

$$\|U_{k+1}\|_\infty = \|\nabla\{\|L\|_*\}\|_\infty \leq \sqrt{\mathrm{rank}(L)} \leq \sqrt{p}.$$

Last step is from the analysis of Watson (1992). Therefore,

$$\gamma\| \operatorname{Tr}(\hat{\Delta}_S(T_i)\hat{\Delta}_L(T_i))| \leq \gamma(\frac{2\alpha}{p} + \frac{\sqrt{p}}{\tau T_i})\|\hat{\Delta}_S(T_i)\|_1.$$

Consequently,

$$\frac{\gamma}{2}\|\hat{\Delta}_S(T_i) + \hat{\Delta}_L(T_i)\|_{\mathbb{F}}^2 \geq \frac{\gamma}{2}\{\|\hat{\Delta}_S(T_i)\|_{\mathbb{F}}^2 + \|\hat{\Delta}_L(T_i)\|_{\mathbb{F}}^2\} - \frac{\gamma}{2}(\frac{2\alpha}{p} + \frac{\sqrt{p}}{\tau T_i})\|\hat{\Delta}_S(T_i)\|_1.$$

Combining the above equation with (39), we get

$$\frac{\gamma}{2}\{\|\hat{\Delta}_S(T_i)\|_{\mathbb{F}}^2 + \|\hat{\Delta}_L(T_i)\|_{\mathbb{F}}^2\} - \frac{\gamma}{2}(\frac{2\alpha}{p} + \frac{\sqrt{p}}{\tau T_i})\|\hat{\Delta}_S(T_i)\|_1 \leq H_1.$$

Using $\|S\|_1 \leq \sqrt{p}\|S\|_{\mathbb{F}}$,

$$\begin{aligned}
&\|\hat{\Delta}_S(T_i)\|_{\mathbb{F}}^2 + \|\hat{\Delta}_L(T_i)\|_{\mathbb{F}}^2 \\
&\leq \frac{2}{\gamma}\{\sqrt{\frac{R_i^2 + \tilde{R}_i^2}{T_i}}\sqrt{\log p} + \frac{R_i^2 + \tilde{R}_i^2}{T_i}\rho_x + \frac{G\sqrt{R_i^2 + \tilde{R}_i^2}}{T_i} \\
&\quad + \frac{\beta(p)(R_i + \tilde{R}_i)\sigma_i\sqrt{12\log(3/\delta_i)}}{\sqrt{T_i}} \\
&\quad + \frac{\beta(p)G(R_i + \tilde{R}_i)\sigma_i\sqrt{12\log(3/\delta_i)}}{T_i\sqrt{\log p}}\} + (\frac{2\alpha}{\sqrt{p}} + \frac{p}{\tau T_i})^2,
\end{aligned}$$

with probability at least $1 - \delta_i$.

D.5 Proof of Proposition 2: Equation $(36b)$

Now we want to convert the error bound in $(36a)$ from function values into vectorized ℓ_1 and Frobenius-norm bounds. Since the error bound in $(36a)$ holds for the minimizer $\hat{M}_i$, it also holds for any other feasible matrix. In particular, applying it to M^* leads to,

$$\begin{aligned}
&f(\bar{M}(T_i)) - f(M^*) + \lambda_i\phi(\bar{W}(T_i)) - \lambda_i\phi(W^*) \\
&\leq \sqrt{\frac{R_i^2 + \tilde{R}_i^2}{T_i}}\sqrt{\log p} + \frac{R_i^2 + \tilde{R}_i^2}{T_i}\rho_x + \frac{G\sqrt{R_i^2 + \tilde{R}_i^2}}{T_i} \\
&\quad + \frac{\beta(p)(R_i + \tilde{R}_i)\sigma_i\sqrt{12\log(3/\delta_i)}}{\sqrt{T_i}} + \frac{\beta(p)G(R_i + \tilde{R}_i)\sigma_i\sqrt{12\log(3/\delta_i)}}{T_i\sqrt{\log p}},
\end{aligned}$$

with probability at least $1 - \delta_i$.

For the next step, we find a lower bound on the left hand side of this inequality.

$$\begin{aligned}
&f(\bar{M}(T_i)) - f(M^*) + \lambda_i\phi(\bar{W}(T_i)) - \lambda_i\phi(W^*) \geq \\
&f(M^*) - f(M^*) + \lambda_i\phi(\bar{W}(T_i)) - \lambda_i\phi(W^*) = \\
&\lambda_i\phi(\bar{W}(T_i)) - \lambda_i\phi(W^*),
\end{aligned}$$

where the first inequality results from the fact that M^* optimizes M.

From here onward all equations hold with probability at least $1-\delta_i$. We have

$$\phi(\bar{W}(T_i)) - \phi(W^*) \leq H_1/\lambda_i. \tag{40}$$

i.e.

$$\|\bar{S}(T_i)\|_1 + \frac{\mu_i}{\lambda_i}\|\bar{L}(T_i)\|_* \leq \|S^*\|_1 + \frac{\mu_i}{\lambda_i}\|L^*\|_* + H_1/\lambda_i$$

Using $\bar{S}(T_i) = \Delta_S^* + S^*$, $\bar{L}(T_i) = \Delta_L^* + L^*$. We split Δ_S^* into its on-support and off-support part. We also divide Δ_L^* into its projection onto V and $V^\perp$. V is range of L^*. Meaning $\forall X \in V, \|X\|_* \leq r$. Therefore,

$$\begin{aligned}\|(\bar{S}(T_i))_{supp}\|_1 &\geq \|(S^*)_{supp}\|_1 - \|(\Delta_S^*)_{supp}\|_1\\ \|(\bar{S}(T_i))_{supp^c}\|_1 &\geq -\|(S^*)_{supp^c}\|_1 + \|(\Delta_S^*)_{supp^c}\|_1,\end{aligned}$$

and

$$\begin{aligned}\|(\bar{L}(T_i))_V\|_* &\geq \|(L^*)_V\|_* - \|(\Delta_L^*)_V\|_*\\ \|(\bar{L}(T_i))_{V^\perp}\|_* &\geq -\|(L^*)_{V^\perp}\|_* + \|(\Delta_L^*)_{V^\perp}\|_*.\end{aligned}$$

Consequently,

$$\|(\Delta_S^*)_{supp^c}\|_1 + \frac{\mu_i}{\lambda_i}\|(\Delta_L^*)_{V^\perp}\|_* \leq \|(\Delta_S^*)_{supp}\|_1 + \frac{\mu_i}{\lambda_i}\|(\Delta_L^*)_V\|_* + H_1/\lambda_i. \tag{41}$$

$\Delta_S^*(T_i) - \hat{\Delta}_S(T_i) = \hat{S}_i - S^*$. Therefore,

$$\begin{aligned}&\|\hat{S}_i - S^*\|_1 =\\ &\|(\Delta_S^*(T_i))_{supp} - (\hat{\Delta}_S(T_i))_{supp}\|_1 + \|(\Delta_S^*(T_i))_{supp^c} - (\hat{\Delta}_S(T_i))_{supp^c}\|_1 \geq\\ &\left\{\|(\Delta_S^*(T_i))_{supp}\|_1 - \|(\hat{\Delta}_S(T_i))_{supp}\|_1\right\} - \left\{\|(\Delta_S^*(T_i))_{supp^c}\|_1 - \|(\hat{\Delta}_S(T_i))_{supp^c}\|_1\right\}.\end{aligned}$$

Hence,

$$\begin{aligned}&\|(\hat{\Delta}_S(T_i))_{supp^c}\|_1 - \|(\hat{\Delta}_S(T_i))_{supp}\|_1\\ &\leq \|(\Delta_S^*(T_i))_{supp^c}\|_1 - \|(\Delta_S^*(T_i))_{supp}\|_1 + \|\hat{S}_i - S^*\|_1.\end{aligned}$$

As Equation (37) is satisfied, we can use Lemma 1 (Negahban et al., 2012). Combining the result with Lemma 8, we have $\|\hat{S}_i - S^*\|_1^2 \leq (4s+3r)(s\frac{\lambda_i^2}{\gamma^2} + r\frac{\mu_i^2}{\gamma^2})$. Consequently, further use of Lemma 8 and the inequality $\|(\hat{\Delta}_S(T_i))_{supp}\|_1 \leq \sqrt{s}\|\hat{\Delta}(T_i)\|_{\mathbb{F}}$ allows us to conclude that there exists a universal constant c such that

$$\begin{aligned}\|\hat{\Delta}_S(T_i)\|_1^2 &\leq 4s\|\hat{\Delta}_S(T_i)\|_{\mathbb{F}}^2 + (H_1/\lambda_i)^2 + c(s+r)(s\frac{\lambda_i^2}{\gamma^2} + r\frac{\mu_i^2}{\gamma^2})\\ &\quad + cr\frac{\mu_i^2}{\lambda_i^2}\left[\frac{2}{\gamma}H_1 + (\frac{\alpha}{\sqrt{p}} + \frac{p}{\tau T_i})^2 + s\frac{\lambda_i^2}{\gamma^2} + r\frac{\mu_i^2}{\gamma^2}\right]\\ &\leq 4s\left[\frac{2}{\gamma}H_1 + (\frac{\alpha}{\sqrt{p}} + \frac{p}{\tau T_i})^2\right] + (H_1/\lambda_i)^2 + c(s+r)(s\frac{\lambda_i^2}{\gamma^2} + r\frac{\mu_i^2}{\gamma^2})\\ &\quad + cr\frac{\mu_i^2}{\lambda_i^2}\left[\frac{2}{\gamma}H_1 + (\frac{\alpha}{\sqrt{p}} + \frac{p}{\tau T_i})^2 + s\frac{\lambda_i^2}{\gamma^2} + r\frac{\mu_i^2}{\gamma^2}\right],\end{aligned}$$

with probability at least $1-\delta_i$. Optimizing the above bound with choice of λ_i and complying with the conditions in Lemma 11, leads to

$$\lambda_i^2 = \frac{\gamma}{s+r}H_1 + \frac{\alpha^2}{p^2} + \frac{\beta^2(p)\sigma^2}{T_i}\left(\log p + \log\frac{1}{\delta}\right).$$

Repeating the same calculations for $\|\hat{\Delta}_L(T_i)\|_*$ results in

$$\mu_i^2 = c_\mu \lambda_i^2,$$

we have

$$\|\hat{\Delta}_S(T_i)\|_1^2 \leq c(s+r+\frac{s+r}{\gamma})H_1 + c(s+r)(1+\frac{s+r}{p\gamma^2})\frac{\alpha^2}{p} + (s+r)(\frac{p^2}{\tau T_i^2}+\frac{\alpha}{\tau T_i}).$$

Therefore,

$$\begin{aligned}\|\Delta_S^*(T_i)\|_1^2 &\leq 2\|\hat{\Delta}_S(T_i)\|_1^2 + 2\|S^*-\hat{S}_i\|_1^2 \qquad (42)\\ &\leq 2\|\hat{\Delta}(T_i)\|_1^2 + 8c(s+r)(s\frac{\lambda_i^2}{\gamma^2}+r\frac{\mu_i^2}{\gamma^2})\\ &\leq c(s+r+\frac{s+r}{\gamma})H_1 + c(s+r)(1+\frac{s+r}{p\gamma^2})\frac{\alpha^2}{p} + (s+r)(\frac{p^2}{\tau T_i^2}+\frac{\alpha}{\tau T_i}).\end{aligned}$$

Finally, in order to use $\bar{S}(T_i)$ as the next prox center $\tilde{S}_{i+1}$, we would also like to control the error $\|\bar{S}(T_i)-\hat{S}_{i+1}\|_1^2$. Without loss of generality, we can design $\tilde{R}_i = c_r R_i$ for any $0 \leq c_r \leq 1$. The result only changes in a constant factor. Hence, we use $\tilde{R}_i = R_i$. Since $\lambda_{i+1} \leq \lambda_i$ by assumption, we obtain the same form of error bound as in (42). We want to run the epoch till all these error terms drop to $R_{i+1}^2 := R_i^2/2$. It suffices to set the epoch length T_i to ensure that sum of all terms in (42) is not greater that $R_i^2/2$. All above conditions are met if we choose the epoch length

$$\begin{aligned}T_i \simeq C(s+r+\frac{s+r}{\gamma})^2\left[\frac{\log p + 12\beta^2(p)\sigma_i^2\log\frac{6}{\delta}}{R_i^2}\right]\\ + C(s+r+\frac{s+r}{\gamma})\left[\frac{\beta(p)G\sigma_i\sqrt{12\log\frac{6}{\delta}}}{R_i\sqrt{\log p}}+\frac{G}{R_i}+\rho_x\right],\end{aligned}$$

for a suitably large universal constant C. Then, we have that

$$\|\Delta_S^*(T_i)\|_1^2 \leq \frac{c'}{\sqrt{C}}R_i^2 + c(s+r)(1+\frac{s+r}{p\gamma^2})\frac{\alpha^2}{p}.$$

Since the second part of the upper bound does not shrink in time, we stop where two parts are equal. Namely, $R_i^2 = c(s+r)(1+\frac{s+r}{p\gamma^2})\frac{\alpha^2}{p}$.

With similar analysis for L, we get

$$\|\Delta_L^*(T_i)\|_*^2 \leq \frac{c'}{\sqrt{C}}\frac{1}{1+\gamma}R_i^2 + c\frac{(s+r)^2}{p\gamma^2}\frac{\alpha^2}{p}.$$

D.6 Proof of Guarantees with Fixed Epoch Length, Sparse + Low Rank Case

This is a special case of Theorem 4 (Appendix). Note that this fixed epoch length results in a convergence rate that is worse by a factor of $\log p$. The key difference between this case and optimal epoch length setting of Theorem 4 is that in the latter we guaranteed error halving by the end of each epoch whereas with fixed epoch length that statement may not be possible after the number of epochs becomes large enough. Therefore, we need to show that in such case the error does not increase much to invalidate our analysis. Let k^* be the epoch number such that error halving holds true until then. Next we demonstrate that error does not increase much for $k > k^*$. The proof follows the same nature as that of Theorem 1 (in the main text), Section B.6, with

$$k^* := \sup\left\{ i : 2^{\frac{j}{2}+1} \leq \frac{cR_1\gamma}{s+r}\sqrt{\frac{T_0}{\log p + \beta^2(p)\sigma_i^2 w^2}} \right\},$$

for all epochs $j \leq i$ and

$$k_0 = \log\left(\frac{R_1\gamma}{s+r}\sqrt{\frac{T}{\log p + \beta^2(p)\sigma^2 w^2}} \right).$$

D.7 Proof of Guarantees for Sparse + Low Rank Graphical Model selection Problem

Here we prove Corollary 2. Proof follows by using the bounds derived in Appendix B.7 for Taylor series expansion and following the lines of Theorem 4 proof as in Appendix D.

According to D.1, in order to prove guarantees, we first need to bound $\|z_{k+1} - z_k\|_1$ and $\|z_k\|_\infty$. According to Equation (30) and considering the imposed ℓ_1 bound, this is equivalent to bound $\|g_{k+1} - g_k\|_1$ and $\|g_k\|_\infty$.$\|g_{k+1} - g_k\|_1$ and $\|g_k\|_\infty$. The rest of the proof follows on lines of Theorem 2 proof. On the other hand, Lipschitz property requires a bound on $\|g_k\|_1$, which is much more stringent.

Assuming we are in a close proximity of M^*, we can use Taylor approximation to locally approximate M^{-1} by M^{*-1} as in (Ravikumar et al., 2011)

$$M^{-1} = M^{*-1} - M^{*-1}\Delta M^{*-1} + \mathcal{R}(\Delta),$$

where $\Delta = M - M^*$ and $\mathcal{R}(\Delta)$ is the remainder term. We have

$$\|g_{k+1} - g_k\|_1 \leq |\!|\!|\Gamma^*|\!|\!|_\infty \|M_{k+1} - M_k\|_1,$$

and

$$\begin{aligned}
\|g_k\|_\infty &\leq \|g_k - \mathbb{E}(g_k)\|_\infty + \|\mathbb{E}(g_k)\|_\infty \\
&\leq \|e_k\|_\infty + \|\Sigma^* - M_k^{-1}\|_\infty \\
&\leq \sigma + \|\Gamma^*\|_\infty \|M_{k+1} - M_k\|_1.
\end{aligned}$$

The term $\|M_{k+1} - M_k\|_1$ is bounded by $2\breve{R}$ by construction. We assume $|\!|\!|\Gamma^*|\!|\!|_\infty$ and $\|\Gamma^*\|_\infty$ are bounded.

The error Δ needs to be "small enough" for the $\mathcal{R}(\Delta)$ to be negligible, and we now provide the conditions for this. By definition, $\mathcal{R}(\Delta) = \sum_{k=2}^{\infty}(-1)^k(M^{*-1}\Delta)^k M^{*-1}$. Using triangle inequality and sub-multiplicative property for Frobenious norm,

$$\|\mathcal{R}(\Delta)\|_{\mathbb{F}} \leq \frac{\|M^{*-1}\|_{\mathbb{F}}\|\Delta M^{*-1}\|_{\mathbb{F}}^2}{1-\|\Delta M^{*-1}\|_{\mathbb{F}}}.$$

For $\|\Delta\|_{\mathbb{F}} \leq 2\breve{R} \leq \frac{0.5}{\|M^{*-1}\|_{\mathbb{F}}}$, we get

$$\|\mathcal{R}(\Delta)\|_{\mathbb{F}} \leq \|M^{*-1}\|_{\mathbb{F}}.$$

We assume $\|\Sigma^*\|_{\mathbb{F}}$ is bounded.

Therefore, if the variables are closely-related we need to start with a small $\breve{R}$. For weaker correlations, we can start in a bigger ball. The rest of the proof follows the lines of proof for Theorem 4, by replacing G^2 by $\|\Gamma^*\|_\infty \breve{R}(\sigma + \|\Gamma^*\|_\infty \breve{R})$.

E Implementation

Here we discuss the updates for REASON 1 and REASON 2. Note that for any vector v, $v_{(j)}$ denotes the j-th entry.

E.1 Implementation details for REASON 1

Let us start with REASON 1. We have already provided closed form solution for y and z. The update rule for θ can be written as

$$\begin{aligned} &\min_{w} \ \|w-v\|_2^2 \ \ s.t. \ \ \|w\|_1 \leq R, \\ &w = \theta - \tilde{\theta}_i, \\ &R = R_i, \\ &v = \frac{1}{\rho+\rho_x}[y_k - \tilde{\theta}_i - \frac{f(\theta_k)}{\rho} + \frac{z_k}{\rho} + \frac{\rho_x}{\rho}(\theta_k - \tilde{\theta}_i)]. \end{aligned} \tag{43}$$

We note that if $\|v\|_1 \leq R$, the answer is $w = v$. Else, the optimal solution is on the boundary of the constraint set and we can replace the inequality constraint with $\|w\|_1 = R$. Similar to (Duchi et al., 2008), we perform Algorithm 3 for solving (43). The complexity of this Algorithm is $\mathcal{O}(d \log d)$, $d = p^2$.

E.2 Implementation details for REASON 2

For REASON 2, the update rule for M, Z, Y and U are in closed form. Consider the S-update. It can be written in form of (43) with

$$\min_{W} \ \lambda_i\|W+\tilde{S}_i\|_1 + \frac{\rho}{2\tau_k}\|W - (S_k + \tau_k G_{M_k} - \tilde{S}_i)\|_{\mathbb{F}}^2. \ \ s.t. \ \ \|W\|_1 \leq R,$$

$$W = S - \tilde{S}_i, \quad R = R_i.$$

Algorithm 3 Implementation of θ-update

```
Input: A vector $v = \frac{1}{\rho+\rho_x}[y_k - \tilde{\theta}_i - \frac{\nabla f(\theta_k)}{\rho} + \frac{z_k}{\rho} + \frac{\rho_x}{\rho}(\theta_k - \tilde{\theta}_i)]$ and a scalar $R = R_i > 0$
if $\|v\|_1 \leq R$, then
    Output: $\theta = v + \tilde{\theta}_i$
else
    Sort $v$ into $\mu$: $\mu_1 \geq \mu_2 \geq \cdots \geq \mu_d$.
    Find $\kappa = \max\{j \in [d] : \mu_j - \frac{1}{j}\big(\sum_{i=1}^{j} \mu_i - R\big) > 0\}$.
    Define $\zeta = \frac{1}{\kappa}\big(\sum_{i=1}^{\kappa} \mu_i - R\big)$
    Output: $\theta$, where $\theta_{(j)} = \mathrm{sign}(v_{(j)}) \max\{v_{(j)} - \zeta, 0\} + (\tilde{\theta}_i)_{(j)}$
end if
```

Algorithm 4 Implementation of S-update

```
Input: $W^{(1)} = \mathrm{vector}(S_k - \tilde{S}_i)$ and a scalar $R = R_i > 0$
for $t = 1$ to $t = t_s$ do
    $v = W^{(t)} - \eta_t \Big[\lambda_i \nabla^{(t)} \|W^{(t)} + \mathrm{vector}(\tilde{S}_i)\|_1 + \frac{\rho}{\tau_k}\Big(W^{(t)} - \mathrm{vector}(S_k + \tau_k G_{M_k} - \tilde{S}_i)\Big)\Big]$
    if $\|v\|_1 \leq R$, then
        $W^{(t+1)} = v$
    else
        Sort $v$ into $\mu$: $\mu_1 \geq \mu_2 \geq \cdots \geq \mu_d$.
        Find $\kappa = \max\{j \in [d] : \mu_j - \frac{1}{j}\big(\sum_{i=1}^{j} \mu_i - R\big) > 0\}$.
        Define $\zeta = \frac{1}{\kappa}\big(\sum_{i=1}^{\kappa} \mu_i - R\big)$
        For $1 \leq j \leq d$, $W_{(j)}^{(t+1)} = \mathrm{sign}(v_{(j)}) \max\{v_{(j)} - \zeta, 0\}$
    end if
end for
Output:matrix($W^{(t_s)}$) + $\tilde{S}_i$
```

Therefore, similar to (Duchi et al., 2008), we generate a sequence of $\{W^{(t)}\}_{t=1}^{t_s}$ via

$$W^{(t+1)} = \Pi_1 \left[W^{(t)} - \eta_t \nabla^{(t)} \left(\lambda_i \|W + \tilde{S}_i\|_1 + \frac{\rho}{2\tau_k} \|W - (S_k + \tau_k G_{M_k} - \tilde{S}_i)\|_{\mathbb{F}}^2 \right) \right],$$

where Π_1 is projection on to ℓ_1 norm, similar to Algorithm 3. In other words, at each iteration, vector $\Big(W^{(t)} - \eta_t \Big[\lambda_i \nabla^{(t)} \|W^{(t)} + \tilde{S}_i\|_1 + \frac{\rho}{\tau_k}(W^{(t)} - (S_k + \tau_k G_{M_k} - \tilde{S}_i))\Big]\Big)$ is the input to Algorithm 3 (instead of vector v) and the output is vector$(W^{(t+1)})$. The term $\nabla^{(t)} \|W^{(t)} + \tilde{S}_i\|_1$ stands for subgradient of the ℓ_1 norm $\|W^{(t)} + \tilde{S}_i\|_1$. The S-update is summarized is Algorithm 4. A step size of $\eta_t \propto 1/\sqrt{t}$ guarantees a convergence rate of $\mathcal{O}(\sqrt{\log p/T})$ (Duchi et al., 2008).

The L-update is very similar in nature to the S-update. The only difference is that the projection is on to nuclear norm instead of ℓ_1 norm. It can be done by performing an SVD before the ℓ_1 norm projection.

The code for REASON 1 follows directly from the discussion in Section E.1. For REASON 2 on the other hand, we have added additional heuristic modifications to improve the performance. REASON 2 code is available at `https://github.com/haniesedghi/REASON2`. The first modification is that we do not update the dual variable Z per every iteration on S and L. Instead, we update the dual variable once S and L seem to have converged to some value or after every m iterations on S and L. The reason is that once we start the iteration, S and L can be far from each other which results in a big dual variable and hence, a slower convergence. The value of m can be set based on the problem. For the experiments discussed in the paper we have used $m = 4$.

Further investigation on REASON 2 shows that performing one of the projections (either ℓ_1 or nuclear norm) suffices to reach this performance. The same precision can be reached using only one of the projections. Addition of the second projection improves the performance only marginally. Performing nuclear norm projections are much more expensive since they require SVD. Therefore, it is more efficient to perform the ℓ_1 projection. In the code, we leave it as an option to run both projections.