S-D-RSM: Stochastic Distributed Regularized Splitting Method for Large-Scale Convex Optimization Problems

Maoran Wang[1], **Xingju Cai**[1,2], **Yongxin Chen**[3*]

[1]School of Mathematical Sciences, Nanjing Normal University, P.R. China
[2]Key Laboratory of NSLSCS (NNU), Ministry of Education, P.R. China
[3]School of Mathematics and Statistics, Nanjing University of Science and Technology, P.R. China
{230901010, caixingju}@njnu.edu.cn, chenyongxin@buaa.edu.cn

Abstract

This paper investigates problems of large-scale distributed composite convex optimization, with motivations from a broad range of applications, including multi-agent systems, federated learning, smart grids, wireless sensor networks, compressed sensing, and so on. Stochastic gradient descent (SGD) and its variants are commonly employed to solve such problems. However, existing algorithms often rely on vanishing step sizes, strong convexity assumptions, or entail substantial computational overhead to ensure convergence or obtain favorable complexity. To bridge the gap between theory and practice, we integrate consensus optimization and operator splitting techniques (see Problem Reformulation) to develop a novel stochastic splitting algorithm, termed the stochastic distributed regularized splitting method (S-D-RSM). In practice, S-D-RSM performs parallel updates of proximal mappings and gradient information for only a randomly selected subset of agents at each iteration. By introducing regularization terms, it effectively mitigates consensus discrepancies among distributed nodes. In contrast to conventional stochastic methods, our theoretical analysis establishes that S-D-RSM achieves global convergence without requiring diminishing step sizes or strong convexity assumptions. Furthermore, it achieves an iteration complexity of 1/epsilon with respect to both the objective function value and the consensus error. Numerical experiments show that S-D-RSM achieves up to two to three times speedup compared with state-of-the-art baselines, while maintaining comparable or better accuracy. These results not only validate the algorithm's theoretical guarantees but also demonstrate its effectiveness in practical tasks such as compressed sensing and empirical risk minimization.

Introduction

In this work, we consider a class of large-scale distributed composite convex optimization problems:

$$\min_{x\in\mathbb{R}^n}\left\{\Phi(x):=\sum_{i=1}^{m}\left(f_i(x)+g_i(x)\right)\right\}, \tag{1}$$

where m is the number of nodes, $\{f_i\}_{i=1}^m$ is a sequence of proper, lower semicontinuous convex functions (not necessarily differentiable), $\{g_i\}_{i=1}^m$ is a sequence of convex functions that are Fréchet differentiable on $\mathbb{R}^n$, and each gradient ∇g_i is $\frac{1}{\beta_i}$-Lipschitz continuous. Throughout this paper, the usual restrictive requirement of strong convexity of f_i or g_i is not needed (Pathak and Wainwright 2020; Li, Chang, and Chi 2020; Li, Acharya, and Richtárik 2024; Sadiev, Condat, and Richtárik 2024). Problem (1) arises in a wide range of applications, including economics and traffic theory (Cornuejols and Tütüncü 2006; Gu et al. 2019), image processing (Chambolle and Pock 2016; Ehrhardt et al. 2025), machine learning (Philippenko and Dieuleveut 2024), and other fields.

Some "full participation" optimization methods—where all nodes are involved in computation at per iteration—have been proposed to solve problem (1); see, for example, (Raguet, Fadili, and Peyré 2013; Briceño Arias 2015; Pathak and Wainwright 2020; Aragón-Artacho et al. 2023). It is worth noting that in (Wu et al. 2025), the authors proposed a new algorithm that unifies several commonly used full-participation schemes and provides a unified framework for their theoretical analysis. Although these algorithms admit global convergence under general convexity assumptions, their per-iteration cost remains high due to the need to compute all proximal mappings prox_{f_i} and evaluate all gradients ∇g_i for large-scale problems. As a result, stochastic (i.e., partial participation) optimization methods have attracted increasing attention.

Related Works

Gradient-Based Methods for Smooth Problems. Stochastic gradient descent (SGD) (Robbins and Monro 1951) is a foundational algorithm widely used in machine learning. Since its introduction by Robbins and Monro (1951), SGD has undergone numerous developments, giving rise to variants such as stochastic batch gradient descent (Nemirovski et al. 2009) and compressed gradient descent (Alistarh et al. 2017). Recently, Gower et al. (2019) proposed a general framework for analyzing SGD with arbitrary sampling strategies in the strongly convex setting. Overall, while these methods are effective in many practical settings, their theoretical convergence guarantees typically rely on restrictive conditions such as vanishing step sizes and strong convexity.

Proximal Point Algorithms for Non-Smooth Problems. For non-smooth optimization problems, the proximal point

*Corresponding author

algorithm (PPA) (Rockafellar 1976) and its variants have been extensively investigated. Compared to gradient-based methods, PPA exhibits greater robustness to inaccuracies in step size selection, as evidenced by the analyses in (Ryu and Boyd 2014; Parikh and Boyd 2014). For large-scale non-smooth problems, stochastic variants of PPA (S-PPA) are more commonly employed in practice (Bertsekas 2011; Bianchi 2016; Patrascu and Necoara 2018). Under random sampling of component functions, vanishing step sizes, and suitable measurability and boundedness assumptions, Bianchi (2016) established the almost sure convergence of S-PPA in the ergodic sense. Recently, Li et al. (2024) proposed an extrapolated version of S-PPA (also known as FedExProx) for federated learning, which incorporates mini-batch sampling and an extrapolation step to accelerate convergence. Under convexity, Lipschitz continuity, and interpolation regimes (Montanari and Zhong 2022)—which are satisfied in overparameterized deep learning models—they established an iteration complexity of $\mathcal{O}(\epsilon^{-1})$. Sadiev et al. (2024) further established linear convergence under additional structural assumptions.

Proximal Gradient Methods for Composite Problems. For composite problems with *multiple* smooth components and a *single* non-smooth convex function, the stochastic proximal gradient (S-PG) method—originating from a combination of SGD (Robbins and Monro 1951) and proximal gradient methods (Beck and Teboulle 2009)—has been extensively investigated (Rosasco, Villa, and Vũ 2016; Atchadé, Fort, and Moulines 2017; Rosasco, Villa, and Vũ 2020). In (Rosasco, Villa, and Vũ 2016), the almost sure convergence of S-PG was established under strong convexity and vanishing step sizes. Under the general convex setting, Atchadé et al. (2017) developed a unified analytical framework for both unbiased and biased gradient estimators in S-PG and derived an $\mathcal{O}(\epsilon^{-2})$ complexity bound under the assumption that the non-smooth component of the objective function is nonnegative and vanishing step sizes. Recently, Rosasco et al. (2020) refined the complexity result of Atchadé et al. (2017) by establishing an improved bound of $\mathcal{O}\left(\epsilon^{1/(t-1)}\right)$ without requiring the non-negativity assumption, with a vanishing step size of the form $\mathcal{O}(1/k^t)$, where $t \in (1/2, 1)$ and k denotes the iteration index. However, no convergence guarantees for S-PG are available in the absence of either strong convexity or vanishing step sizes.

Operator Splitting Methods. When both smooth and non-smooth components are present in *multiple* blocks, *operator splitting techniques* provide a powerful algorithmic framework for designing deterministic algorithms. Some methods have been successfully extended to stochastic settings in recent works. Cevher et al. (2016; 2018) introduced the stochastic forward Douglas-Rachford (S-FDR) splitting method, establishing a stochastic extension of deterministic FDR (Briceño Arias 2015). Although S-FDR adopts the SGD-style gradient estimate, its requirement to compute all proximal mappings per iteration raises scalability concerns for large-scale problems. Furthermore, by inheriting SGD's framework, S-FDR inherits similar theoretical requirements, including vanishing step sizes and strong convexity assumptions (Yurtsever, Vũ, and Cevher 2016). More recently, a broader algorithmic framework was introduced by Combettes et al. (Combettes and Pesquet 2015; Bùi, Combettes, and Woodstock 2022; Combettes and Madariaga 2025), who developed the stochastic generalized forward–backward (S-GFB) method as a stochastic extension of the deterministic GFB (Raguet, Fadili, and Peyré 2013). In contrast to S-FDR, this approach reduces the per iteration computational burden by updating only a subset of the proximal mappings $\{\mathrm{prox}_{\gamma f_i}\}_{i=1}^m$, though it may still pose computational challenges due to the need for full gradient evaluations, particularly in large-scale applications. For a comprehensive overview of operator splitting algorithms, we refer the reader to (Condat et al. 2023; Han 2022; Cai et al. 2022).

For clarity and ease of comparison, the properties of the aforementioned algorithms are summarized in Table 1. In the table, S_k denotes the index set sampled at iteration k. The notions of "Convergence" and "Complexity" are analyzed under standard convexity assumptions, without imposing strong convexity or diminishing step-size conditions.

Theoretical and Practical Trade-offs. Compared to full participation approaches, stochastic methods significantly reduce computational costs by involving only a subset of nodes in each iteration. However, they also entail inherent trade-offs in step size policies, strong convexity requirements, gradient approximation accuracy, convergence guarantees, and complexity analysis. Vanishing step sizes are commonly used to establish almost sure convergence (Bianchi 2016; Rosasco, Villa, and Vũ 2020), but they may degrade practical performance. Conversely, constant step sizes typically require more accurate gradient estimates (Combettes and Madariaga 2025; Bùi, Combettes, and Woodstock 2022; Combettes and Pesquet 2015), which can increase computational overhead. Furthermore, complexity analysis often relies on additional structural assumptions about the objective function, such as strong convexity or interpolation regimes (Yurtsever, Vũ, and Cevher 2016; Gower et al. 2019; Rosasco, Villa, and Vũ 2020; Li, Acharya, and Richtárik 2024). These challenges highlight the need for algorithms that are both theoretically robust and computationally efficient.

Contributions. Motivated by the unresolved theoretical-practical trade-offs in stochastic optimization, we propose a novel framework addressing three persistent limitations in state-of-the-art methods:

- **Practical Limitations of Vanishing Step Sizes**
  While vanishing step sizes ($\gamma_k \to 0$) ensure theoretical convergence, empirical evidence consistently highlights adverse effects: asymptotic slowdown preventing ϵ-optimal solutions and hyperparameter sensitivity causing sharp convergence deterioration (Bottou, Curtis, and Nocedal 2018). This bottleneck acutely impacts cross-device federated learning with heterogeneous compute capabilities.
- **Restrictive Functional Assumptions**
  Existing $\mathcal{O}(\epsilon^{-1})$ guarantees rely on structurally convenient but impractical conditions: strong convexity vio-

Algorithm	f_i, g_i	Step size	Computational cost	Convergence	Complexity
S-PPA (Bianchi 2016)	$g_i = 0$	vanishing	$\mathrm{prox}_{\gamma_k f_{i_k}}$	No	No
FedExProx (Li, Acharya, and Richtárik 2024)	$g_i = 0$	constant	$\left\{\mathrm{prox}_{\gamma f_i}\right\}_{i\in S_k}$	No	No
S-PG (Rosasco, Villa, and Vũ 2020)	$f_i = 0,\ i \geq 2$	vanishing	$\{\nabla g_i\}_{i\in S_k}$	No	$\mathcal{O}(\epsilon^{1/(t-1)})$ $t \in (1/2, 1)$
S-GFB (Combettes and Madariaga 2025)	$f_i, g_i \neq 0$	constant	$\{\mathrm{prox}_{\gamma f_i}\}_{i\in S_k}; \{\nabla g_i\}_{i\in[m]}$	Yes	No
S-FDR (Cevher, Vũ, and Yurtsever 2018)	$f_i, g_i \neq 0$	vanishing	$\{\mathrm{prox}_{\gamma f_i}\}_{i\in[m]}; \{\nabla g_i\}_{i\in S_k}$	Yes	No
This paper	$\boldsymbol{f_i, g_i \neq 0}$	**constant**	$\{\mathrm{prox}_{\gamma f_i}, \nabla g_i\}_{i\in \boldsymbol{S_k}}$	**Yes**	$\boldsymbol{\mathcal{O}(\epsilon^{-1})}$

Table 1: Comparison of the properties of S-D-RSM (Algorithm 1) and several state-of-the-art methods.

lated by large machine learning models, and interpolation regimes implausible under non-IID data (e.g., recommendation systems) (Zhang et al. 2024).

- **Large-scale computing bottlenecks**
  Despite their stochastic formulations, prevalent operator-splitting methods still inherit deterministic burdens: S-FDR-type algorithms require $\mathcal{O}(m)$ proximal evaluations per iteration, while S-GFB-type methods necessitate $\mathcal{O}(m)$ gradient computations. These computational demands can become prohibitive in large-scale distributed systems, especially when m is large or when proximal or gradient evaluations are costly.

The main contributions are summarized as follows:

- The proposed method integrates *consensus optimization* with *operator splitting* and exploits *parallelism* by evaluating only a *subset* of proximal mappings $\{\mathrm{prox}_{\gamma f_i}\}_{i=1}^m$ and gradients $\{\nabla g_i\}_{i=1}^m$ at each iteration. In addition, regularization is introduced into each subproblem to mitigate consensus discrepancies among distributed nodes, as confirmed by numerical experiments.
- We provide a rigorous convergence analysis showing that S-D-RSM achieves global convergence under general convexity assumptions, *without* requiring strong convexity, interpolation, or vanishing step sizes. The method attains a sublinear ergodic convergence rate of $\mathcal{O}(1/K)$ with respect to both the objective gap and consensus violation, leading to an iteration complexity of $\mathcal{O}(\epsilon^{-1})$. Notably, we establish almost sure convergence of the iterate sequence, further reinforcing the algorithm's reliability in practice.
- Since the theoretical guarantees of our algorithm are established solely based on the objective function and do not depend on the underlying data distribution across devices, it retains global convergence and an $\mathcal{O}(\epsilon^{-1})$ complexity under heterogeneous settings, provided that the loss function is convex.

Notations

We denote by $\Gamma_0(\mathbb{R}^n)$ the set of all proper, lower semicontinuous, and convex functions on $\mathbb{R}^n$. Given $f \in \Gamma_0(\mathbb{R}^n)$, the subdifferential of f is defined as

$$\partial f : x \mapsto \{u \in \mathbb{R}^n \mid f(y) \geq f(x) + \langle u, y - x\rangle,\ \forall y \in \mathbb{R}^n\},$$

and its proximal mapping is defined by

$$\mathrm{prox}_f : x \mapsto \arg\min_{y\in\mathbb{R}^n} \left\{ f(y) + \frac{1}{2}\|y - x\|^2 \right\}.$$

From the definition of prox_f, it can be verified that for all $u, x \in \mathbb{R}^n$ and $\delta > 0$,

$$x = \mathrm{prox}_f(u - \delta x) \Leftrightarrow x = \mathrm{prox}_{\frac{f}{1+\delta}}\left(\frac{u}{1+\delta}\right). \quad (2)$$

Let $(\Omega, \mathcal{F}, \mathbb{P})$ denote a probability space, with Ω the sample space, $\mathcal{F}$ the σ-algebra, and $\mathbb{P}$ the probability measure. The abbreviation "$\mathbb{P}$-a.s." refers to "$\mathbb{P}$-almost surely". A sequence of random variables $\{\xi^k\}_{k=1}^\infty$ is said to converge $\mathbb{P}$-a.s. to a random variable ξ, if

$$\mathbb{P}\left(\left\{\omega \in \Omega \mid \lim_{k\to\infty} \xi^k(\omega) = \xi(\omega)\right\}\right) = 1,$$

which is denoted as $\lim_{k\to\infty} \xi^k = \xi$, $\mathbb{P}$-a.s. Unless otherwise specified, all inequalities involving random variables are understood to hold $\mathbb{P}$-almost surely. The bold symbol $\boldsymbol{x}$ represents a vector of $m-1$ stacked vectors, $\{x_i\}_{i=1}^{m-1} \subset \mathbb{R}^n$, i.e., $\boldsymbol{x} = (x_1, x_2, \ldots, x_{m-1})$. Moreover, we define $\sigma(x^0, x^1, \ldots, x^k) \subset \mathcal{F}$ as the smallest σ-algebra generated by the set of random variables $\{x^0, x^1, \ldots, x^k\}$. For a random variable v and a σ-algebra $\mathcal{J} \subset \mathcal{F}$, we denote by $\mathbb{E}(v|\mathcal{J})$ the conditional expectation of v given $\mathcal{J}$, and write $v \perp\!\!\!\perp \mathcal{J}$ to denote that v is independent of $\mathcal{J}$. Finally, for any real number r, the largest integer not greater than r is denoted by $\lfloor r \rfloor$, and we define $\frac{r}{0} = \infty$ in this paper.

Problem Reformulations and the Proposed Algorithm

In this section, we introduce two reformulations of problem (1) that serve as the foundation of our approach. The first reformulation characterizes a system of equations satisfied by the solutions of problem (1), offering theoretical guidance for the algorithm design and global convergence analysis. The second reformulation gives rise to the definition of ϵ-optimal solutions, laying the foundation for the complexity analysis of the proposed algorithm. Since both reformulations are equivalent to problem (1), their interrelationship is further clarified in Lemma 3. Due to space constraints, all technical details and proofs are provided in the extended version: https://arxiv.org/abs/2511.10133.

Problem Reformulation I

Assumption 1 *Assume that problem* (1) *admits at least one solution and problem* (1) *satisfies*

$$\bigcap_{i=1}^{m} \mathrm{ri}\left(\mathrm{dom} f_i\right) \neq \emptyset,$$

*where "*ri*" denotes the set of relative interior points.*

Under Assumption 1, we obtain (Rockafellar 1970)

$$\arg\min_{x\in\mathbb{R}^n} \Phi(x) = \mathrm{zer}\left(\sum_{i=1}^{m}(\partial f_i + \nabla g_i)\right). \tag{3}$$

Based on (3), we derive an alternative reformulation of the solution set of problem (1).

Lemma 1 (Reformulation I) *Let $\mathcal{S}$ be the set of all $(z_1, z_2, \ldots, z_{m-1}, x)$ satisfying the following system:*

$$\begin{cases} x = \mathrm{prox}_{\frac{\gamma f_m}{m-1}}\left(\frac{1}{m-1}\sum_{i=1}^{m-1}\left(z_i - \sigma\gamma\nabla g_i(x)\right) - \frac{\gamma}{m-1}\nabla g_m(x)\right), \\ x = \mathrm{prox}_{\gamma f_1}(2x - z_1 - (1-\sigma)\gamma\nabla g_1(x)), \\ \quad \vdots \\ x = \mathrm{prox}_{\gamma f_{m-1}}(2x - z_{m-1} - (1-\sigma)\gamma\nabla g_{m-1}(x)). \end{cases}$$

Then the following assertions hold:

- *If $x^\star$ minimizes Φ, then there exist $z_1^\star, \ldots, z_{m-1}^\star \in \mathbb{R}^n$ such that $(z_1^\star, \ldots, z_{m-1}^\star, x^\star) \in \mathcal{S}$.*
- *Conversely, if $(z_1^\star, \ldots, z_{m-1}^\star, x^\star) \in \mathcal{S}$, then $x^\star$ minimizes Φ.*

Problem Reformulation II

By introducing the constraint $x_1 = x_2 = \cdots = x_m$, problem (1) can be reformulated as:

$$\begin{aligned} &\min_{x_i} \sum_{i=1}^{m} f_i(x_i) + \sum_{i=1}^{m-1}\{(1-\sigma)g_i(x_i) + \sigma g_i(x_m)\} + g_m(x_m) \\ &\text{s.t. } x_1 = x_2 = \cdots = x_m. \end{aligned} \tag{4}$$

By leveraging the equivalence between problem (1) and problem (4), the ϵ-optimal solution of problem (1) is defined as follows.

Definition 1 *Let the tuple $(x_1, x_2, \ldots, x_m)$ consist of random variables generated by a stochastic algorithm over a probability space $(\Omega, \mathcal{F}, \mathbb{P})$. The tuple $(x_1, x_2, \ldots, x_m)$ is said to be* ϵ-optimal in expectation *if the random variables $(x_1, x_2 \ldots, x_m)$ satisfy the following two conditions for all $i, j \in [m]$*

$$\|\mathbb{E}[x_j - x_i]\| \leq \epsilon \quad \textit{and} \quad |\mathbb{E}\left[H(x_1, \ldots, x_m)\right] - \Phi^\star| \leq \epsilon,$$

where H denotes the objective function of the reformulated problem (4)*, and $\Phi^\star$ denotes the global optimal value of problem* (1)*.*

Algorithm 1: S-D-RSM for solving problem (1)

Input: $K > 0$; $\alpha_i \geq 0$; $\sigma \in [0, 1]$; $\alpha_i + \lfloor 1 - \sigma \rfloor \neq 0$; initial point $y_i^0, z_i^0 \in \mathbb{R}^n$, $i \in [m-1]$.

Parameter: $\gamma \in \left(0, \min_{i\in[m-1]}\left\{\frac{2\alpha_i}{\frac{1}{\beta_m(m-1)} + \frac{\sigma}{\beta_i}}, \frac{2(2+\alpha_i)\beta_i}{1-\sigma}\right\}\right)$; $\lambda_i \in \left(0, 2 + \alpha_i - \frac{(1-\sigma)\gamma}{2\beta_i}\right)$; error $= 1$ and the error tolerance $\varepsilon > 0$.

Output: Approximate solution x^k.

Process:

1: Let $k = 0$, error $= 1$.

2: **while** error $> \varepsilon$ or $k \leq K$ **do**

3: Server update

$$x^{k+1} = \mathrm{prox}_{\frac{\gamma f_m}{m-1}}\left(\frac{1}{m-1}\sum_{i=1}^{m-1}\left(z_i^k + \alpha_i(y_i^k - x^{k+1}) - \frac{\gamma}{m-1}\nabla g_m(y_i^k)\right) - \frac{\sigma\gamma}{m-1}\sum_{i=1}^{m-1}\nabla g_i(y_i^k)\right).$$

4: Randomly select users $S_k \subseteq [m-1]$.

5: For user $i \in S_k$, compute

$$y_i^{k+1} = \mathrm{prox}_{\gamma f_i}\Big(2x^{k+1} - z_i^k + \alpha_i(x^{k+1} - y_i^{k+1}) - (1-\sigma)\gamma\nabla g_i(x^{k+1})\Big),$$
$$z_i^{k+1} = z_i^k + \lambda_i\left(y_i^{k+1} - x^{k+1}\right).$$

6: For user $i \notin S_k$, set

$$y_i^{k+1} = y_i^k,$$
$$z_i^{k+1} = z_i^k.$$

7: Update error $\leftarrow \frac{\sum_{i=1}^{m-1}\|y_i^k - x^k\|^2}{\|x^k\|^2}$ and $k \leftarrow k + 1$.

8: **end while**

The Proposed Algorithm

Based on the problem reformulation I, we introduce the stochastic distributed regularized splitting method (S-D-RSM) for addressing (1).

Remark 1 • *Based on the definition of the proximal mapping, the subproblems in step 3 and step 5 contain regularization terms, specifically $\frac{1}{2\gamma}\sum_{i=1}^{m-1}\alpha_i\|x - y_i^k\|^2$ and $\frac{\alpha_i}{2\gamma}\|y_i - x^{k+1}\|^2$, which are introduced to balance the discrepancy between x^{k+1} and y_i^k, with the balancing strength controlled by the parameter α_i.*

- *If $\sigma > 0$, the computation of x^{k+1} requires all $\nabla g_i(y_i^k)$ and $\nabla g_m(y_i^k)$ only at the initial iteration $k = 0$, while for $k > 0$, only a subset of these gradients needs to be computed.*
- *For computational convenience, we explicitly express*

x^{k+1} *and* y_i^{k+1} *(for* $i \in S_k$*) based on* (2) *as follows:*

$$\begin{cases} x^{k+1} = \operatorname{prox}_{\frac{\gamma f_m}{(1+\bar{\alpha})(m-1)}}\Big(\frac{1}{(m-1)(1+\bar{\alpha})}\sum_{i=1}^{m-1}\Big(z_i^k + \alpha_i y_i^k \\ \qquad - \frac{\gamma}{m-1}\nabla g_m(y_i^k)\Big) - \frac{\sigma\gamma}{(m-1)(1+\bar{\alpha})}\sum_{i=1}^{m-1}\nabla g_i(y_i^k)\Big), \\ y_i^{k+1} = \operatorname{prox}_{\frac{\gamma f_i}{1+\alpha_i}}\Big(\frac{2+\alpha_i}{1+\alpha_i}x^{k+1} - \frac{z_i^k}{1+\alpha_i} - \frac{(1-\sigma)\gamma}{1+\alpha_i}\nabla g_i(x^{k+1})\Big), \end{cases}$$

where $\bar{\alpha} = \frac{1}{m-1}\sum_{i=1}^{m-1}\alpha_i$.

Assumption 2 *Select* $S_k \subset [m-1]$ *such that* $S_k \perp\!\!\!\perp \mathcal{F}_k$ *with* $\mathbb{P}(i \in S_k) = p_i > 0, i \in [m-1]$*, where the* σ*-algebra* $\mathcal{F}_k$ *is defined as*

$$\mathcal{F}_k = \sigma\left(\left\{\boldsymbol{y}^j, \boldsymbol{z}^j\right\}_{j=0}^k\right),$$

where $\boldsymbol{y}^k = \left(y_i^k\right)_{i=1}^{m-1}$ *and* $\boldsymbol{z}^k = \left(z_i^k\right)_{i=1}^{m-1}$.

Consequently, the iterates x^k, $\boldsymbol{y}^k$, and $\boldsymbol{z}^k$ are random variables, and the random set S_k is independent of the history $\left\{\boldsymbol{y}^j, \boldsymbol{z}^j\right\}_{j\le k}$ and $\left\{x^j\right\}_{j\le k+1}$. Furthermore, since x^{k+1} is generated through a continuous mapping of $\boldsymbol{y}^k$ and $\boldsymbol{z}^k$, it follows that x^{k+1} is $\mathcal{F}_k$-measurable.

Main Theory Results

In this section, we present several convergence results for Algorithm 1. All theoretical results concerning Algorithm 1 are derived under Assumptions 1–2 and the parameter settings for γ, σ, α_i, and λ_i as specified in Algorithm 1. To facilitate the convergence analysis of Algorithm 1, we introduce the following auxiliary variables, which are not computed in practice:

$$\begin{cases} \tilde{y}_i^{k+1} = \operatorname{prox}_{\gamma f_i}\Big(2x^{k+1} - z_i^k + \alpha_i(x^{k+1} - \tilde{y}_i^{k+1}) \\ \qquad - (1-\sigma)\gamma\nabla g_i(x^{k+1})\Big),\ \forall i \in [m-1], \\ \tilde{z}_i^{k+1} = z_i^k + \lambda_i\left(\tilde{y}_i^{k+1} - x^{k+1}\right),\ \forall i \in [m-1]. \end{cases} \tag{5}$$

The following result demonstrates the decreasing properties of the random variables generated by Algorithm 1.

Lemma 2 (Decreasing properties) *The random sequence* $\left\{x^k, (y_i^k, z_i^k)_{i=1}^{m-1}\right\}_{k=0}^{\infty}$ *generated by Algorithm 1 and the virtual user variables* $\left\{(\tilde{y}_i^k)_{i=1}^{m-1}\right\}_{k=1}^{\infty}$ *defined by* (5) *satisfy that*

$$\begin{aligned} &\mathbb{E}\left\{\sum_{i=1}^{m-1}\left(\frac{1}{\lambda_i p_i}\|z_i^{k+1} - z_i^\star\|^2 + \frac{\alpha_i}{p_i}\|y_i^{k+1} - x^\star\|^2\right)\Big|\mathcal{F}_k\right\} \\ &\le \sum_{i=1}^{m-1}\left(\frac{1}{\lambda_i p_i}\|z_i^k - z_i^\star\|^2 + \frac{\alpha_i}{p_i}\|y_i^k - x^\star\|^2\right) \\ &\quad - \sum_{i=1}^{m-1}\left(\alpha_i - \frac{\gamma}{2\beta_m(m-1)} - \frac{\sigma\gamma}{2\beta_i}\right)\|x^{k+1} - y_i^k\|^2 \\ &\quad - \sum_{i=1}^{m-1}\left(2 + \alpha_i - \frac{(1-\sigma)\gamma}{2\beta_i} - \lambda_i\right)\|x^{k+1} - \tilde{y}_i^{k+1}\|^2, \end{aligned} \tag{6}$$

for any $(z_1^\star, z_2^\star, \ldots, z_{m-1}^\star, x^\star) \in \mathcal{S}$.

Based on the reformulation I (Lemma 1) and the decreasing properties of Algorithm 1 (Lemma 2), we are now able to demonstrate the **global convergence** of the sequence produced by Algorithm 1 **without** requiring diminishing step sizes or strong convexity assumptions.

Theorem 1 (Convergence) *Let* $\left\{x^k, (y_i^k, z_i^k)_{i=1}^{m-1}\right\}_{k=0}^{\infty}$ *denote the sequence generated by Algorithm 1. Then, the following hold:*

- $\lim_{k\to\infty}\|x^{k+1} - y_i^k\| = \lim_{k\to\infty}\|x^k - \tilde{y}_i^k\| = 0$*,* $\mathbb{P}$*-a.s.,* $\forall i \in [m-1]$.
- *There exists a random variable* $\tilde{x}$ *taking values in* $\arg\min_{x\in\mathbb{R}^n}\{\Phi(x)\}$ *such that* $\lim_{k\to\infty} x^k = \tilde{x}$*,* $\mathbb{P}$*-a.s.*

The following lemma establishes a connection between two equivalent formulations of problem (1). Specifically, it presents properties of the objective function in problem (4) associated with the solution set $\mathcal{S}$ defined in Lemma 1.

Lemma 3 *For any* $(z_1^\star, z_2^\star, \ldots, z_{m-1}^\star, x^\star) \in \mathcal{S}$ *and any* $x_i \in \mathbb{R}^n$ *for* $i \in [m]$*, the following inequality holds:*

$$H(x_1, \cdots, x_m) - \Phi^\star \ge \frac{1}{\gamma}\sum_{i=1}^{m-1}\langle x^\star - z_i^\star, x_m - x_i\rangle, \tag{7}$$

where $\Phi^\star$ *is the global optimal value of problem* (1)*.*

Next, we analyze the evolution of the objective function of problem (1) along the sequence of iterates generated by Algorithm 1.

Lemma 4 *Let* $\left\{x^k, (y_i^k, z_i^k)_{i=1}^{m-1}\right\}_{k=0}^{\infty}$ *be the sequence generated by Algorithm 1, and let the virtual variables* $\left\{(\tilde{y}_i^k)_{i=1}^{m-1}\right\}_{k=1}^{\infty}$ *be defined by* (5)*. Then, for any* $(z_1^\star, z_2^\star, \ldots, z_{m-1}^\star, x^\star) \in \mathcal{S}$*, the following inequality holds:*

$$\begin{aligned} &2\gamma\left(H(\tilde{y}_1^{k+1}, \cdots, \tilde{y}_{m-1}^{k+1}, x^{k+1}) - H(x^\star, \cdots, x^\star)\right) \\ &\le 2\sum_{i=1}^{m-1}\left\langle x^\star - z_i^\star, x^{k+1} - \tilde{y}_i^{k+1}\right\rangle + a_k - \mathbb{E}[a_{k+1} \mid \mathcal{F}_k] \\ &\quad - \sum_{i=1}^{m-1}\left(2 + \alpha_i - \lambda_i - \frac{(1-\sigma)\gamma}{\beta_i}\right)\|\tilde{y}_i^{k+1} - x^{k+1}\|^2 \\ &\quad - \sum_{i=1}^{m-1}\left(\alpha_i - \frac{\gamma}{(m-1)\beta_m} - \frac{\sigma\gamma}{\beta_i}\right)\|x^{k+1} - y_i^k\|^2, \end{aligned} \tag{8}$$

where

$$a_k = \sum_{i=1}^{m-1}\left(\frac{\alpha_i}{p_i}\|y_i^k - x^\star\|^2 + \frac{1}{\lambda p_i}\|z_i^k - z_i^\star\|^2\right).$$

Building on the previously established descent properties of Algorithm 1 and the structural characteristics of the original problem, we now establish the convergence rate of Algorithm 1 under a **constant** step size and **general convexity** assumptions.

Theorem 2 (Rate) *Let* $\left\{(z_i^k)_{i=1}^{m-1}, x^k, (y_i^k)_{i=1}^{m-1}\right\}_{k=0}^{\infty}$ *be the sequence generated by Algorithm 1. Then for every* $K \in \mathbb{N}$ *and* $i \in [m-1]$*, define*

$$x_{\mathrm{av}}^K = \frac{1}{K}\sum_{k=0}^{K-1} x^{k+1},\ y_{\mathrm{av},i}^K = \frac{1}{K}\sum_{k=0}^{K-1} y_i^k.$$

Then the following hold:

- $\left\|\mathbb{E}\left[x_{\text{av}}^K - y_{\text{av},i}^K\right]\right\| = \mathcal{O}\left(1/K\right)$, *for all* $i \in [m-1]$.
- $\left|\mathbb{E}\left[H(y_{\text{av},1}^K, \cdots, y_{\text{av},m-1}^K, x_{\text{av}}^K)\right] - \Phi^\star\right| = \mathcal{O}\left(1/K\right)$.

As a consequence of Theorem 2, Algorithm 1 achieves an ϵ-optimal solution in expectation within at most $\mathcal{O}(\epsilon^{-1})$ iterations.

Numerical Experiments

In this section, we apply the proposed S-D-RSM to solve the compressed sensing problem and the logistic regression problem. We compare the performance of S-D-RSM with four state-of-the-art methods: Split-Douglas-Rachford method (SDR)(Briceño-Arias and Roldán 2021; Wang, Cai, and Chen 2024), S-GFB (Combettes and Madariaga 2025), FedDR(FedADMM)(Tran Dinh et al. 2021; Wang, Marella, and Anderson 2022), and S-FDR (Cevher, Vũ, and Yurtsever 2018). All algorithms were implemented in MATLAB 2021b, and experiments were conducted on a desktop computer equipped with an Intel(R) Core(TM) i5-10210U CPU @ 1.60GHz, 2112 MHz, and 8 GB RAM.

For all tested algorithms, each numerical experiment is repeated 20 times, and the average performance is reported. In each iteration, 30% of the users are activated according to a uniform sampling strategy in the stochastic method. The initial points are set to zero vectors and are identical across all algorithms. The regularization parameters α_i, for all $i \in [m-1]$, are set to 1, and the parameter σ is set to $1/2$ in the proposed S-D-RSM algorithm. To ensure a fair comparison, the parameters of each algorithm are tuned as large as possible while still guaranteeing convergence.

Compressed Sensing

We begin by evaluating the empirical performance of the proposed method on the compressed sensing problem:

$$\begin{aligned} &\min_{x\in\mathbb{R}^n} \ \|x\|_1 \\ &\text{s.t. } Ax = b, \end{aligned} \tag{9}$$

where $A \in \mathbb{R}^{p\times n}$ is the sensing matrix and $b \in \mathbb{R}^p$ is the observed measurement vector. Let a_i denote the i-th row of A, and b_i denote the i-th entry of b. By incorporating indicator functions for affine constraints, which equal 0 on the constraint set and ∞ otherwise, problem (9) can be reformulated in the structure of problem (1), with $g_i \equiv 0$:

$$\min_{x\in\mathbb{R}^n} \ \|x\|_1 + \sum_{i=1}^{p} I_{C_i}(x), \tag{10}$$

where each $C_i = \{x \mid a_i^\top x = b_i\}$ is a affine constraint set.

We set $n = 2500$ and $p = 0.25n$, and construct the sensing matrix A using the discrete cosine transform (DCT) or discrete Fourier transform (DFT). The ground-truth signal $x^* \in \mathbb{R}^{2500}$ is generated from the standard normal distribution, with a sparsity level of 1%. The observed vector is then computed as $b = Ax^*$. Figure 1 shows the relative consensus error $\max_i \left\{\|y_i^k - x^k\|/\|x^k\|\right\}$ of all the algorithms for the two sensing matrices A (DCT, DFT). These figures clearly indicate that S-D-RSM converges much faster than SDR and S-GFB in terms of the objective value and relative error under all settings. Compared with S-GFB and FedADMM, the regularization term in S-D-RSM helps reduce the consensus error during the iteration process. Compared with SDR, the parallel update mechanism in S-D-RSM enhances computational efficiency, while the randomized selection of constraint sets C_i for projection reduces the frequency of redundant constraint processing in the linear system $Ax = b$.

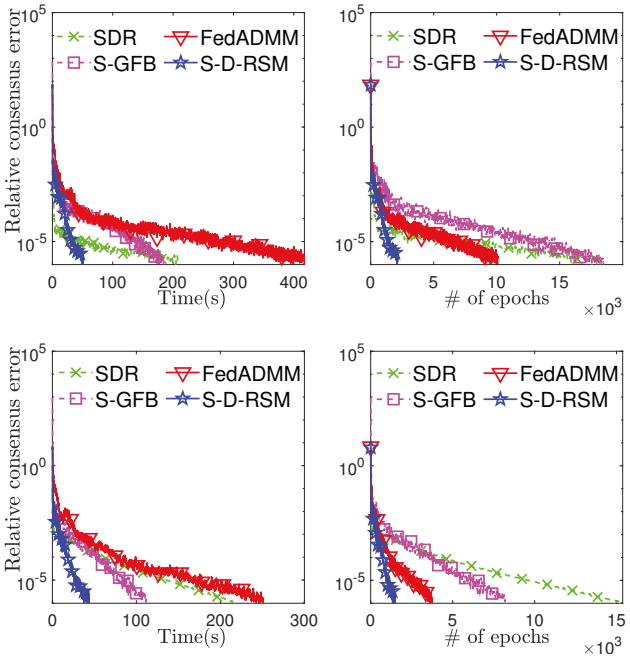

Figure 1: Comparison of different methods for compressed sensing problems: DCT (top) and DFT (bottom).

Logistic Regression Problem with ℓ_1-norm Regularization Terms

We further evaluate the performance of the proposed S-D-RSM algorithm on a logistic regression problem with ℓ_1-norm regularization terms:

$$\min_{x\in\mathbb{R}^n} \sum_{i=1}^{m} \frac{1}{m}\left(\log(1+\exp(-b_i a_i^\top x)) + \lambda_i \|x\|_1\right),$$

where $\{a_i\}_{i=1}^m \subset \mathbb{R}^n$ and $\{b_i\}_{i=1}^m \subset \{\pm 1\}$ denote the input features and output labels, respectively.

Due to space limitations and the similarity of experimental patterns, we present results only for two commonly used benchmark datasets, namely a7a and mushrooms, obtained from the LIBSVM repository (Chang and Lin 2011). The datasets are randomly partitioned into 75% training and 25% testing sets, and the maximum number of iterations is set to 1000. The regularization parameters λ_i are sampled uniformly from the interval $[10^{-3}, 10^{-2}]$. S-D-RSM demonstrates a clear advantage over competing methods in terms of CPU time, as shown in Figure 2. In particular, S-FDR

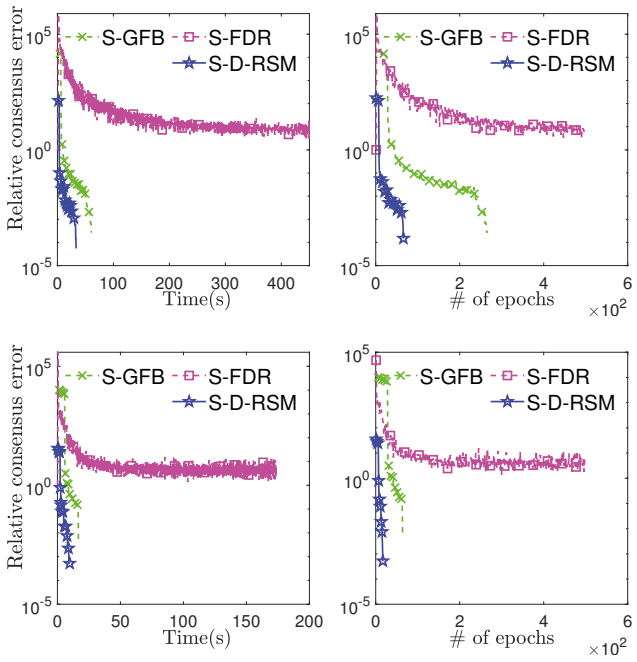

Figure 2: Comparison of different methods for the logistic regression problem with ℓ_1-norm regularizer on the two data sets: a7a (top) and mushrooms (bottom).

employs a vanishing step size, which leads to slower convergence compared to S-GFB and S-D-RSM, both of which use constant step sizes. Although both S-GFB and S-D-RSM involve a comparable number of proximal mapping evaluations per iteration, S-D-RSM updates at least 30% of gradients, whereas S-GFB computes the full gradient in every iteration. Consequently, S-D-RSM can potentially reduce the gradient computation cost by up to 70% per iteration relative to S-GFB.

Conclusions

In this work, we propose a novel stochastic splitting algorithm, S-D-RSM, by integrating consensus optimization with operator splitting techniques. The method enables partial agent participation via parallel updates and incorporates regularization to reduce consensus errors. In contrast to conventional stochastic methods, S-D-RSM is theoretically shown to achieve global convergence and an $\mathcal{O}(\epsilon^{-1})$ complexity for both the objective value and consensus error, under constant step sizes and without strong convexity.

Acknowledgments

The authors would like to express their sincere gratitude to the anonymous referees for their insightful comments and constructive suggestions, which have substantially improved the quality of this paper. This research was supported by the National Natural Science Foundation of China (Grant Nos. 12471290 and 12131004) and the Postgraduate Research & Practice Innovation Program of Jiangsu Province (Grant No. KYCX25_1928).

References

Alistarh, D.; Grubic, D.; Li, J.; Tomioka, R.; and Vojnovic, M. 2017. QSGD: Communication-Efficient SGD via Gradient Quantization and Encoding. In *Advances in Neural Information Processing Systems*, volume 30.

Aragón-Artacho, F. J.; Malitsky, Y.; Tam, M. K.; and Torregrosa-Belén, D. 2023. Distributed Forward-Backward Methods for Ring Networks. *Computational Optimization and Applications*, 86: 845–870.

Atchadé, Y. F.; Fort, G.; and Moulines, E. 2017. On Perturbed Proximal Gradient Algorithms. *Journal of Machine Learning Research*, 18(10): 1–33.

Beck, A.; and Teboulle, M. 2009. A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems. *SIAM Journal on Imaging Sciences*, 2(1): 183–202.

Bertsekas, D. P. 2011. Incremental Proximal Methods for Large Scale Convex Optimization. *Mathematical Programming*, 129(2): 163–195.

Bianchi, P. 2016. Ergodic Convergence of a Stochastic Proximal Point Algorithm. *SIAM Journal on Optimization*, 26(4): 2235–2260.

Bottou, L.; Curtis, F. E.; and Nocedal, J. 2018. Optimization Methods for Large-Scale Machine Learning. *SIAM Review*, 60(2): 223–311.

Briceño Arias, L. M. 2015. Forward-Douglas-Rachford Splitting and Forward-Partial Inverse Method for Solving Monotone Inclusions. *Optimization*, 64(5): 1239–1261.

Briceño-Arias, L. M.; and Roldán, F. 2021. Split-Douglas-Rachford for Composite Monotone Inclusions and Split-ADMM. *SIAM Journal on Optimization*, 31(4): 2987–3013.

Bùi, M. N.; Combettes, P. L.; and Woodstock, Z. C. 2022. Block-Activated Algorithms for Multicomponent Fully Nonsmooth Minimization. In *ICASSP 2022–2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 5428–5432.

Cai, X.; Guo, K.; Jiang, F.; Wang, K.; Wu, Z.; and Han, D. 2022. The Developments of Proximal Point Algorithms. *Journal of the Operations Research Society of China*, 10: 197–239.

Cevher, V.; Vũ, B. C.; and Yurtsever, A. 2018. *Stochastic Forward Douglas-Rachford Splitting Method for Monotone Inclusions*, 149–179. Springer.

Chambolle, A.; and Pock, T. 2016. An Introduction to Continuous Optimization for Imaging. *Acta Numerica*, 25: 161–319.

Chang, C.-C.; and Lin, C.-J. 2011. LIBSVM: A Library for Support Vector Machines. *ACM Transactions on Intelligent Systems and Technology*, 2: 27:1–27:27. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

Combettes, P.; and Madariaga, J. 2025. A Geometric Framework for Stochastic Iterations. arXiv:2504.02761.

Combettes, P. L.; and Pesquet, J.-C. 2015. Stochastic Quasi-Fejér Block-Coordinate Fixed Point Iterations with Random Sweeping. *SIAM Journal on Optimization*, 25(2): 1221–1248.

Condat, L.; Kitahara, D.; Contreras, A.; and Hirabayashi, A. 2023. Proximal Splitting Algorithms for Convex Optimization: A Tour of Recent Advances, with New Twists. *SIAM Review*, 65(2): 375–435.

Cornuejols, G.; and Tütüncü, R. 2006. *Optimization Methods in Finance*. Cambridge UK.

Ehrhardt, M. J.; Kereta, Ž.; Liang, J.; and Tang, J. 2025. A Guide to Stochastic Optimisation for Large-Scale Inverse Problems. *Inverse Problems*, 41(5): 053001–053062.

Gower, R. M.; Loizou, N.; Qian, X.; Sailanbayev, A.; Shulgin, E.; and Richtárik, P. 2019. SGD: General Analysis and Improved Rates. In *International Conference on Machine Learning*, 5200–5209. PMLR.

Gu, Y.; Cai, X.; Han, D.; and Wang, D. Z. 2019. A Tri-Level Optimization Model for a Private Road Competition Problem with Traffic Equilibrium Constraints. *European Journal of Operational Research*, 273(1): 190–197.

Han, D. 2022. A Survey on Some Recent Developments of Alternating Direction Method of Multipliers. *Journal of the Operations Research Society of China*, 10: 1–52.

Li, H.; Acharya, K.; and Richtárik, P. 2024. The Power of Extrapolation in Federated Learning. In *Advances in Neural Information Processing Systems*, volume 37.

Li, Y.; Chang, T.-H.; and Chi, C.-Y. 2020. Secure Federated Averaging Algorithm with Differential Privacy. In *2020 IEEE 30th International Workshop on Machine Learning for Signal Processing (MLSP)*, 1–6. IEEE.

Montanari, A.; and Zhong, Y. 2022. The Interpolation Phase Transition in Neural Networks: Memorization and Generalization Under Lazy Training. *The Annals of Statistics*, 50(5): 2816–2847.

Nemirovski, A.; Juditsky, A.; Lan, G.; and Shapiro, A. 2009. Robust Stochastic Approximation Approach to Stochastic Programming. *SIAM Journal on Optimization*, 19(4): 1574–1609.

Parikh, N.; and Boyd, S. 2014. Proximal Algorithms. *Foundations and Trends® in Optimization*, 1(3): 127–239.

Pathak, R.; and Wainwright, M. J. 2020. FedSplit: an Algorithmic Framework for Fast Federated Optimization. In *Advances in Neural Information Processing Systems*, volume 33.

Patrascu, A.; and Necoara, I. 2018. Nonasymptotic Convergence of Stochastic Proximal Point Methods for Constrained Convex Optimization. *Journal of Machine Learning Research*, 18(198): 1–42.

Philippenko, C.; and Dieuleveut, A. 2024. Compressed and Distributed Least-Squares Regression: Convergence Rates with Applications to Federated Learning. *Journal of Machine Learning Research*, 25(288): 1–80.

Raguet, H.; Fadili, J.; and Peyré, G. 2013. A Generalized Forward-Backward Splitting. *SIAM Journal on Imaging Sciences*, 6(3): 1199–1226.

Robbins, H.; and Monro, S. 1951. A Stochastic Approximation Method. *Annals of Mathematical Statistics*, 22(3): 400–407.

Rockafellar, R. T. 1970. *Convex Analysis*. Princeton University Press, Princeton, New Jersey.

Rockafellar, R. T. 1976. Monotone Operators and the Proximal Point Algorithm. *SIAM Journal on Control and Optimization*, 14(5): 877–898.

Rosasco, L.; Villa, S.; and Vũ, B. C. 2020. Convergence of Stochastic Proximal Gradient Algorithm. *Journal of Optimization Theory and Applications*, 82: 891–917.

Rosasco, L.; Villa, S.; and Vũ, B. C. 2016. Stochastic Forward–Backward Splitting for Monotone Inclusions. *Journal of Optimization Theory and Applications*, 169(2): 388–406.

Ryu, E. K.; and Boyd, S. 2014. Stochastic Proximal Iteration: a Non-Asymptotic Improvement Upon Stochastic Gradient Descent. *Author website, early draft*, 25.

Sadiev, A.; Condat, L.; and Richtárik, P. 2024. Stochastic Proximal Point Methods for Monotone Inclusions under Expected Similarity. arXiv:2405.14255.

Tran Dinh, Q.; Pham, N. H.; Phan, D.; and Nguyen, L. 2021. FedDR–Randomized Douglas-Rachford Splitting Algorithms for Nonconvex Federated Composite Optimization. In *Advances in Neural Information Processing Systems*, volume 34.

Wang, H.; Marella, S.; and Anderson, J. 2022. Fedadmm: A Federated Primal-Dual Algorithm Allowing Partial Participation. In *2022 IEEE 61st Conference on Decision and Control (CDC)*, 287–294. IEEE.

Wang, M.; Cai, X.; and Chen, Y. 2024. Convergence Analysis of Split-Douglas-Rachford Algorithm and a Novel Preconditioned ADMM with an Improved Condition. *Numerical Mathematics: Theory, Methods and Applications*, 17(3): 658–696.

Wu, R.; Liu, D.; Wang, X.; and Wang, A. 2025. CoCoA Is ADMM: Unifying Two Paradigms in Distributed Optimization. arXiv:2502.00470.

Yurtsever, A.; Vũ, B. C.; and Cevher, V. 2016. Stochastic Three-Composite Convex Minimization. In *Advances in Neural Information Processing Systems*, volume 29.

Zhang, X.; Jia, X.; Liu, H.; Liu, X.; and Zhang, X. 2024. A Goal Interaction Graph Planning Framework for Conversational Recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 19578–19587.