

Sampling-Based Methods for Multi-Block Optimization Problems over Transport Polytopes*

Yukuan Hu[†] Mengyu Li[‡] Xin Liu[†] Cheng Meng[§]

Abstract

This paper focuses on multi-block optimization problems over transport polytopes, which underlie various applications including strongly correlated quantum physics and machine learning. Conventional block coordinate descent-type methods for the general multi-block problems store and operate on the matrix variables directly, resulting in formidable expenditure for large-scale settings. On the other hand, optimal transport problems, as a special case, have attracted extensive attention and numerical techniques that waive the use of the full matrices have recently emerged. However, it remains nontrivial to apply these techniques to the multi-block, possibly nonconvex problems with theoretical guarantees. In this work, we leverage the benefits of both sides and develop novel sampling-based block coordinate descent-type methods, which are equipped with either entropy regularization or Kullback-Leibler divergence. Each iteration of these methods solves subproblems restricted on the sampled degrees of freedom. Consequently, they involve only sparse matrices, which amounts to considerable complexity reductions. We explicitly characterize the sampling-induced errors and establish convergence and asymptotic properties for the methods equipped with the entropy regularization. Numerical experiments on typical strongly correlated electron systems corroborate their superior scalability over the methods utilizing full matrices. The advantage also enables the first visualization of approximate optimal transport maps between electron positions in three-dimensional contexts.

1 Introduction

In this work, we consider multi-block optimization problems over transport polytopes as follows:

$$\begin{aligned} & \text{minimize} && f(X_1, \dots, X_N), \\ & \text{subject to} && X_i \in \mathcal{U}(\mathbf{a}_i, \mathbf{b}_i), \quad i = 1, \dots, N, \end{aligned} \tag{1.1}$$

where, for any $i \in \{1, \dots, N\}$ ($1 \leq N \in \mathbb{N}$), $\mathbf{a}_i \in \mathbb{R}_+^{m_i}$, $\mathbf{b}_i \in \mathbb{R}_+^{n_i}$,

$$\mathcal{U}(\mathbf{a}_i, \mathbf{b}_i) := \{T \in \mathbb{R}_+^{m_i \times n_i} \mid T\mathbf{1}_{n_i} = \mathbf{a}_i, \quad T^\top \mathbf{1}_{m_i} = \mathbf{b}_i\}$$

***Funding:** The work of the first author was supported by the National Key R&D Program of China (2020YFA0711900, 2020YFA0711904). The work of the second author was supported by the Outstanding Innovative Talents Cultivation Funded Programs 2021 of Renmin University of China. The work of the third author was supported in part by the National Natural Science Foundation of China (12125108, 11971466, 12226008, 11991021, 11991020, 12021001, 12288201), Key Research Program of Frontier Sciences, Chinese Academy of Sciences (ZDBS-LY-7022), and CAS-Croucher Funding Scheme for Joint Laboratories “CAS AMSS-PolyU Joint Laboratory of Applied Mathematics: Nonlinear Optimization Theory, Algorithms and Applications”. The work of the fourth author was supported by Beijing Municipal Natural Science Foundation (1232019), the National Natural Science Foundation of China (12101606), and Renmin University of China research fund program for young scholars.

[†]State Key Laboratory of Scientific and Engineering Computing, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, and University of Chinese Academy of Sciences, Beijing, China (ykhu@lsec.cc.ac.cn, liuxin@lsec.cc.ac.cn).

[‡]Institute of Statistics and Big Data, Renmin University of China, Beijing, China (limengyu516@ruc.edu.cn).

[§]Center for Applied Statistics, Institute of Statistics and Big Data, Renmin University of China, Beijing, China (chengmeng@ruc.edu.cn).

is called the transport polytope. The notation “ $\mathbf{1}_n$ ” refers to the all-ones vector in \mathbb{R}^n . The unknown matrix variables are $X_i \in \mathbb{R}^{m_i \times n_i}$ ($i = 1, \dots, N$). The objective function $f : \bigtimes_{i=1}^N \mathbb{R}^{m_i \times n_i} \rightarrow \mathbb{R}$ is assumed to be block Lipschitz smooth over $\bigtimes_{i=1}^N \mathcal{U}(\mathbf{a}_i, \mathbf{b}_i)$ (for the definition see section 4), possibly nonconvex. Problem (1.1) finds its applications in several fields. For example, in quantum physics, it provides a promising route for treating the elusive strongly correlated electron systems (e.g., transition metal oxides [29]), describing the electron-electron correlation explicitly [23, 42]. It can also act as a subproblem in finding the Wasserstein barycenter among several discrete probability distributions [21], which has gained popularity so far in statistics [13] and machine learning [28], as well as in label distribution learning [84], which reflects the relative importance of different labels in supervised learning [39].

For solving general multi-block optimization problems, block coordinate descent (BCD)-type methods rank among the top choices. These methods fully exploit the separability of the feasible region, in that each subproblem involves only one variable block and is much easier to solve than the original problem. Representatives of the BCD-type methods are the block coordinate descent (BCD) methods [12, 35, 78], block conditional gradient (BCG) methods [9, 16], proximal alternating linearized minimization (PALM) methods [14, 43], as well as their stochastic versions [9, 22, 31, 40, 51, 74], where randomness is introduced to the gradient calculations or update order. Nevertheless, for the specific problem (1.1), all the existing BCD-type methods store and operate on the matrix variables directly, requiring at least quadratically growing complexities per iteration. This forms formidable memory and computation burdens when $\{m_i\}_{i=1}^N$ and/or $\{n_i\}_{i=1}^N$ are of large magnitude. Taking the aforementioned quantum physics application [42] for instance, m_i ($= n_i$) stands for the number of grid points and can be of order 10^4 or 10^5 even for crude discretization.

When N equals one ($m_i = m$, $n_i = n$) and f is affine, problem (1.1) reduces to the Kantorovich formulation of the classical optimal transport (OT) problem [75]. The exploration of this problem dates back to Monge’s pioneering work in the 18th century [64], followed by Kantorovich’s relaxation in the 20th century [45]. Since then, a plethora of numerical methods for solving OT problems have been constantly emerging. Traditional ones solve differential equations [10, 19] or turn to linear programming solvers [66, 68], resulting in unacceptable cubic complexities. Nowadays, the widest usage may go to the entropy regularization-based methods [27, 67], which allow for the approximations of solutions in $\mathcal{O}(t_{\max} mn)$ scaling time with the Sinkhorn algorithm [73], where t_{\max} is the number of iterations; see section 2.3 for more discussions. In recent years, motivated by the need in large-scale contexts, there have been works dedicated to alleviating the per-iteration quadratic costs by the conventional Sinkhorn algorithm; for example, the low-rank approximation-based [7] and the entrywise sampling-based [53] variants of the Sinkhorn algorithm. Remarkably, the latter variant essentially deals with a *restricted* OT problem:

$$\min_X \quad \langle \hat{C}, X \rangle, \quad \text{s.t. } X \in \mathcal{U}(\mathbf{a}, \mathbf{b}), \quad X_{\mathcal{I}^c} = 0, \quad (1.2)$$

where $\hat{C} \in \mathbb{R}^{m \times n}$ is an effective cost matrix (defined later), $X \in \mathbb{R}^{m \times n}$, $\mathbf{a} \in \mathbb{R}^m$, $\mathbf{b} \in \mathbb{R}^n$, $\mathcal{I} \subseteq \{(j, k) \mid j = 1, \dots, m, k = 1, \dots, n\}$ contains the indices sampled from the beginning according to some probability distribution related to \mathbf{a} and \mathbf{b} , and \mathcal{I}^c denotes its complementary set. The constraint “ $X_{\mathcal{I}^c} = 0$ ” enforces the entries in X indexed by \mathcal{I}^c to be zero, which distinguishes the algorithm from the well-known stochastic optimization methods. As a result, only $|\mathcal{I}|$ entries in X get involved in the calculations and updates, leading to a nice scaling when $|\mathcal{I}| = o(mn)$. However, it remains unclear whether the sampling technique can be adapted to handle the multi-block, possibly nonconvex problem (1.1), while maintaining favorable convergence properties. One possible way is to integrate the sampling technique into the BCD-type methods and to solve restricted subproblems like (1.2) in each iteration. Analyzing the accumulation of errors induced by sampling will then become subtle.

1.1 Contributions and organization.

We develop in this paper sampling-based BCD-type methods for problem (1.1), which are equipped with either entropy regularization or Kullback-Leibler divergence. In particular, importance samplings are performed conforming to the probability distributions associated with the *previous iterates* and each iteration solves subproblems restricted over sampled supports. Consequently, only $o(m_i n_i)$ entries in X_i ($i = 1, \dots, N$) take part in the updates and derivatives calculations, which amounts to considerable computational saving in large-scale contexts.

Following the theoretical results about randomized matrix sparsification, we analyze the convergence and asymptotic properties for the methods equipped with entropy regularization. We explicitly characterize the sampling-induced errors and establish upper bounds for the average stationarity violations. The average violation is further shown to vanish in the limit $\sum_{i=1}^N (m_i + n_i) \rightarrow +\infty$ (with probability going to 1). Notably, to the best of our knowledge, our work is the first attempt in applying the matrix entrywise sampling technique to multi-block nonconvex settings with theoretical guarantees.

We demonstrate the efficiency of the newly designed methods via numerical simulations of typical strongly correlated electron systems. Their better scalability enables the first visualization of the approximate OT maps between electron positions in three-dimensional contexts.

The paper is organized as follows. We provide preliminaries in section 2 and elaborate on the algorithmic developments in section 3. Section 4 contains our theoretical results, whose proofs are deferred to the appendix. Numerical experiments and results are described in section 5. Finally, we conclude in section 6.

2 Preliminaries

This section offers some preliminaries, including notations, tools from OT, and bibliographical notes on entrywise matrix sparsification.

2.1 Notations

This paper presents scalars, vectors, and matrices by regular-font, bold lower-case, and upper-case letters, respectively. We denote the rounding down operation by “ $\lfloor \cdot \rfloor$ ”. The notation “ $\mathbf{1}_n$ ” stands for the all-ones vector in \mathbb{R}^n . The notations “ $\langle \cdot, \cdot \rangle$ ” and “ $\|\cdot\|$ ” calculate, respectively, the standard inner product and norm of vectors or matrices in the ambient Euclidean space. We use “ $\|\cdot\|_2$ ” particularly for the 2-norm of matrices. The notation “ $\kappa(\cdot)$ ” refers to the spectral condition number of a matrix. We use “ $\text{Diag}(\cdot)$ ” to form a diagonal matrix with the input vector. We denote the entries or sub-blocks of vectors by single subscripts (e.g., ϱ_k or \mathbf{a}_i), the sub-blocks of matrices by single subscripts (e.g., X_i), and the entries of matrices by double subscripts (e.g., $x_{i,jk}$). A matrix with a set subscript refers to the entries indexed by the set (e.g., $X_{\mathcal{I}}$). Sometimes, we make abbreviations for the aggregation of the sub-blocks of a matrix (e.g., $X_{\leq i} := (X_1, \dots, X_i)$, $X_{>i} := (X_{i+1}, \dots, X_N)$). These abbreviations become null if the index sets in the subscripts are empty. The entrywise product and division of two vectors or matrices are denoted by “ \odot ” and “ \oslash ”, respectively. Univariate functions, such as “ $\exp(\cdot)$ ” and “ $\log(\cdot)$ ”, are extended to vectors and matrices as entrywise operations.

For a multivariate function g , ∇g is the gradient of g at the points where g is differentiable. We add a subscript to indicate the block with respect to which the derivative is taken (e.g., $\nabla_i g$).

We use \mathbb{R}_{++} to denote the set of positive real numbers. Given a set, its measure or cardinality is represented using “ $|\cdot|$ ”. We use “ \times ” or “ \times ” to refer to the Cartesian product of sets or spaces (e.g., $\bigtimes_{i=1}^N \mathbb{R}^{m_i \times n_i}$ or $\mathbb{R}^{m_i} \times \mathbb{R}^{n_i}$), use exponents to represent the Cartesian product of identical sets or spaces (e.g., $(\mathbb{R}^{K \times K})^{N_e - 1}$). The complementary set is noted with a superscript “ c ” (e.g., \mathcal{I}^c).

When describing algorithms, we use superscripts within brackets to refer to the iteration numbers (e.g., $X_i^{(t)}$).

2.2 Negative entropy and Kullback-Leibler divergence

Definition 1 ([49, 72]). *For any $T = (t_{ij}) \in \mathbb{R}_+^{m \times n}$, its negative entropy is defined as $h(T) := \sum_{ij} t_{ij}(\log t_{ij} - 1)$. Given any $T = (t_{ij})$, $T' = (t'_{ij}) \in \mathbb{R}_+^{m \times n}$, the Kullback-Leibler (KL) divergence between T and T' is defined as*

$$\text{KL}(T; T') := \sum_{i,j} [t_{ij}(\log t_{ij} - \log t'_{ij}) - (t_{ij} - t'_{ij})]. \quad (2.1)$$

If $t_{ij} > 0$ and $t'_{ij} = 0$ for some pair (i, j) , then $\text{KL}(T; T') = +\infty$.

The negative entropy has been adopted in thermodynamics as a measure of disorder in a system, or a measure of uncertainty in information theory. The KL divergence can be treated as the Bregman distance [18] associated with the negative entropy; it has been used as a measure of the disparity between probability distributions [67].

2.3 Entropy regularized optimal transport

The Kantorovich formulation of the discrete OT problem is in general

$$\min_T \langle W, T \rangle, \quad \text{s.t. } T \in \mathcal{U}(\mathbf{p}, \mathbf{q}), \quad (2.2)$$

where $T = (t_{ij}) \in \mathbb{R}^{m \times n}$ is the transport plan, $W = (w_{ij}) \in \mathbb{R}^{m \times n}$ is the cost matrix, and $\mathbf{p} \in \mathbb{R}^m$, $\mathbf{q} \in \mathbb{R}^n$ are discrete probability distributions. The solution to problem (2.2) is referred to as the OT plan, which achieves minimal transportation efforts. Nowadays, OT has attracted extensive attention from applications (e.g., [8, 60, 63, 81]).

The computational complexity of directly solving problem (2.2) as a linear programming usually grows cubically as m and n increase, which severely hinders the wide applications of OT. To approximate the solution efficiently within certain tolerance, the author of [27] adds a (negative) entropy regularization penalty, obtaining

$$\min_T \langle W, T \rangle + \lambda h(T), \quad \text{s.t. } T\mathbf{1}_n = \mathbf{p}, \quad T^\top \mathbf{1}_m = \mathbf{q}, \quad (2.3)$$

where $\lambda > 0$ is the regularization parameter. Note that the nonnegativity requirement is unnecessary due to the definition of h .

On account of the entropy regularizer, problem (2.3) becomes strongly convex and thus admits a unique optimal solution. Furthermore, as $\lambda \rightarrow 0$, this optimal solution converges to an optimal solution of problem (2.2) [67]. Computationally, the dual of problem (2.3) can be solved by an alternating minimization scheme, known as the Sinkhorn algorithm in the OT community [73]. The alternating scheme involves only matrix-vector multiplications and entrywise divisions between vectors (see section 3.1), particularly suited for GPU executions [27].

2.4 Entrywise matrix sparsification

Entrywise matrix sparsification is pioneered by Achlioptas and McSherry [2], later developed in [17, 32, 50], where sampling-based algorithms are described to select a small number of entries from an input matrix to construct a sparse sketch; the sketch is close to the original one in the operator norm with a high probability guarantee. The entries are sampled following some probability distribution associated with the original matrix. Among others, importance sampling, as a statistical technique, constructs the sampling probability distribution in the spirit of variance reduction

Algorithm 1 The ERALM method for solving problem (1.1).

Input: $X_i^{(0)} \in \mathbb{R}^{m_i \times n_i}$, $\mathbf{a}_i \in \mathbb{R}^{m_i}$, $\mathbf{b}_i \in \mathbb{R}^{n_i}$ ($i = 1, \dots, N$), $t_{\max} \in \mathbb{N}$.

- 1: Set $t := 0$.
- 2: **while** certain conditions are not satisfied **and** $t < t_{\max}$ **do**
- 3: **for** $i = 1, \dots, N$ **do**
- 4: Select a regularization parameter $\lambda_i^{(t)} > 0$ and a step size $\alpha_i^{(t)} \in (0, 1]$.
- 5: Compute $C_i^{(t)} \in \mathbb{R}^{m_i \times n_i}$ as in the formula (3.2).
- 6: Solve subproblem (3.3) or (3.4) to obtain $\tilde{X}_i^{(t+1)} \in \mathbb{R}^{m_i \times n_i}$.
- 7: Update $X_i^{(t+1)} := (1 - \alpha_i^{(t)})X_i^{(t)} + \alpha_i^{(t)}\tilde{X}_i^{(t+1)} \in \mathbb{R}^{m_i \times n_i}$.
- 8: **end for**
- 9: Set $t := t + 1$.
- 10: **end while**

Output: Approximate solution $(X_1^{(t)}, \dots, X_N^{(t)}) \in \bigtimes_{i=1}^N \mathbb{R}^{m_i \times n_i}$.

[34, 56, 57, 65]. Owing to the substantial computational complexity reduction thereby, entrywise matrix sparsification has been adopted in various scenarios, e.g., for computing approximate eigenvectors [2], solving OT problems [53], and computing Gromov-Wasserstein distances [54].

3 Algorithmic developments

In this part, we develop two classes of methods for the multi-block problem over the transport polytopes (1.1), with the BCG and PALM methods as starting points. In particular, we add entropy regularizers to the subproblems in the BCG method, while replacing the proximal term with the KL divergence in the PALM method. We further equip them with importance sampling-based entrywise matrix sparsification.

3.1 Entropy regularized alternating linearized minimization

In each iteration, the BCG method obtains search directions via solving subproblems as

$$\min_{X_i} \left\langle C_i^{(t)}, X_i - X_i^{(t)} \right\rangle, \text{ s.t. } X_i \in \mathcal{U}(\mathbf{a}_i, \mathbf{b}_i), \quad (3.1)$$

where

$$C_i^{(t)} := \nabla_i f(X_{<i}^{(t+1)}, X_{\geq i}^{(t)}) \in \mathbb{R}^{m_i \times n_i}. \quad (3.2)$$

Subproblem (3.1) can be identified as an OT problem with $C_i^{(t)}$ being the cost matrix. As mentioned in section 2.3, solving subproblem (3.1) entails cubic complexities using linear programming methods, which forms an unacceptable computational burden. To this end, we add an entropy regularization and instead resort to

$$\min_{X_i} \left\langle C_i^{(t)}, X_i - X_i^{(t)} \right\rangle + \lambda_i^{(t)} h(X_i), \text{ s.t. } X_i \mathbf{1}_{n_i} = \mathbf{a}_i, X_i^\top \mathbf{1}_{m_i} = \mathbf{b}_i \quad (3.3)$$

in each iteration, where $\lambda_i^{(t)} > 0$ is the regularization parameter and h is the negative entropy in Definition 1. Consequently, we obtain the entropy regularized alternating linearized minimization (ERALM) method; see Algorithm 1.

Observe that the number of variables in subproblem (3.3) scales quadratically, while the number of equality constraints grows linearly. Therefore, it is more advantageous to work from the dual

perspective, especially when m_i and/or n_i is of large magnitude. From [67], the dual of subproblem (3.3) is

$$\min_{\tilde{\mathbf{u}}_i, \tilde{\mathbf{v}}_i} q_i(\tilde{\mathbf{u}}_i, \tilde{\mathbf{v}}_i; \lambda_i^{(t)}, \Psi_i^{(t)}) := \lambda_i^{(t)} \exp\left(\frac{\tilde{\mathbf{u}}_i}{\lambda_i^{(t)}}\right)^\top \Psi_i^{(t)} \exp\left(\frac{\tilde{\mathbf{v}}_i}{\lambda_i^{(t)}}\right) - \tilde{\mathbf{u}}_i^\top \mathbf{a}_i - \tilde{\mathbf{v}}_i^\top \mathbf{b}_i, \quad (3.4)$$

where $\tilde{\mathbf{u}}_i \in \mathbb{R}^{m_i}$, $\tilde{\mathbf{v}}_i \in \mathbb{R}^{n_i}$ are the dual variables associated with the equality constraints, and

$$\Psi_i^{(t)} := \exp\left(-C_i^{(t)}/\lambda_i^{(t)}\right) \in \mathbb{R}^{m_i \times n_i} \quad (3.5)$$

is called the kernel matrix. To tackle the block-structured problem (3.4), one natural choice is the BCD method, which is also known as the *Sinkhorn algorithm* [73] in the context of OT. Starting from a given $\tilde{\mathbf{v}}_i^{(t,0)} \in \mathbb{R}^{n_i}$, the Sinkhorn algorithm repeats the following two steps until fulfilling certain criteria:

$$\begin{aligned} \tilde{\mathbf{u}}_i^{(t,s+1)} &:= \lambda_i^{(t)} \log\left(\mathbf{a}_i \oslash \left(\Psi_i^{(t)} \exp\left(\tilde{\mathbf{v}}_i^{(t,s)}/\lambda_i^{(t)}\right)\right)\right), \\ \tilde{\mathbf{v}}_i^{(t,s+1)} &:= \lambda_i^{(t)} \log\left(\mathbf{b}_i \oslash \left(\Psi_i^{(t)\top} \exp\left(\tilde{\mathbf{u}}_i^{(t,s+1)}/\lambda_i^{(t)}\right)\right)\right), \end{aligned}$$

where s indicates the subiteration number. After letting $\check{\mathbf{u}}_i^{(t,s)} := \exp\left(\tilde{\mathbf{u}}_i^{(t,s)}/\lambda_i^{(t)}\right) \in \mathbb{R}^{m_i}$ and $\check{\mathbf{v}}_i^{(t,s)} := \exp\left(\tilde{\mathbf{v}}_i^{(t,s)}/\lambda_i^{(t)}\right) \in \mathbb{R}^{n_i}$, the above schemes can be rewritten as

$$\check{\mathbf{u}}_i^{(t,s+1)} := \mathbf{a}_i \oslash \left(\Psi_i^{(t)} \check{\mathbf{v}}_i^{(t,s)}\right), \quad \check{\mathbf{v}}_i^{(t,s+1)} := \mathbf{b}_i \oslash \left(\Psi_i^{(t)\top} \check{\mathbf{u}}_i^{(t,s+1)}\right), \quad (3.6)$$

involving matrix-vector multiplications and entrywise divisions between vectors, thus favoring high parallel scalability [27]. The linear convergence rate of the Sinkhorn algorithm has been established [59]. Further acceleration can be gained via warm starts [80].

3.2 Sampling-based variant of the ERALM method

The ERALM method still works on matrix variables and requires calculating $C_i^{(t)}$ (3.5) explicitly. Below, we use a sparse matrix $\hat{\Psi}_i^{(t)} \in \mathbb{R}^{m_i \times n_i}$ to approximate $\Psi_i^{(t)} = (\psi_{i,jk}^{(t)})$, rendering most of the computational costs dispensable.

The idea of the sparse approximation largely originates from the following multiplicative expression for the unique optimal solution of subproblem (3.3):

$$\tilde{X}_i^{(t+1,\star)} := \text{Diag}\left(\exp\left(\frac{\tilde{\mathbf{u}}_i^{(t,\star)}}{\lambda_i^{(t)}}\right)\right) \Psi_i^{(t)} \text{Diag}\left(\exp\left(\frac{\tilde{\mathbf{v}}_i^{(t,\star)}}{\lambda_i^{(t)}}\right)\right) \in \mathbb{R}^{m_i \times n_i}, \quad (3.7)$$

where $(\tilde{\mathbf{u}}_i^{(t,\star)}, \tilde{\mathbf{v}}_i^{(t,\star)}) \in \mathbb{R}^{m_i} \times \mathbb{R}^{n_i}$ is an optimal solution of the dual (3.4). The expression (3.7) indicates that $\tilde{x}_{i,jk}^{(t+1,\star)} = 0$ whenever $\psi_{i,jk}^{(t)} = 0$. For another, it has been theoretically established in [41] that the solutions of subproblem (3.1) can be sparse. When f is multi-affine, there even exist sparse solutions with $\mathcal{O}(\sum_{i=1}^N (m_i + n_i))$ nonzero entries to problem (1.1) [42, 44]. These two points together motivate us to compute only a small portion of the entries in $\Psi_i^{(t)}$ to eliminate most of the storage and computation overhead.

For this purpose, it suffices to estimate the sparsity pattern along iterations. In this work, we make an attempt through entrywise sampling [1, 17, 32, 50, 54]; that is, randomly pick a small portion from $\{(j, k) : j = 1, \dots, m_i, k = 1, \dots, n_i\}$ according to certain sampling probability distribution. Specifically, the entrywise sampling is implemented via Poisson sampling framework following the

recent works [4, 17, 53], which independently evaluate each index for inclusion in the sampled set. Compared to the sampling with replacement, the Poisson sampling has been shown to provide higher accuracy in some scenarios and is more practical for distributed systems [77].

In light of the importance sampling (see section 2.4), the optimal sampling probabilities should be $p_{i,jk}^{(t,\star)} \propto \tilde{x}_{i,jk}^{(t+1,\star)}$ [54]. However, $\tilde{X}_i^{(t+1,\star)}$ is completely unknown without the knowledge of $\tilde{\mathbf{u}}_i^{(t,\star)}$ and $\tilde{\mathbf{v}}_i^{(t,\star)}$ (see the formula (3.7)). An alternative is to sample conforming to the values in the previous iteration, i.e., $p_{i,jk}^{(t)'} \propto x_{i,jk}^{(t)}$. This becomes reasonable when the procedure gets close to the optimum. In addition, to recover entries (j, k) from the optimal sparsity pattern that are missing in $X_i^{(t)}$, in the sense that $x_{i,jk}^{(t)} = 0$, we linearly interpolate between $p_{i,jk}^{(t)'}$ and the sampling probability proposed by [54], i.e., $p_{i,jk}'' \propto \sqrt{a_{i,j} b_{i,k}}$. Specifically,

$$p_{i,jk}^{(t)} := \gamma p_{i,jk}^{(t)'} + (1 - \gamma)p_{i,jk}'' = \gamma \frac{x_{i,jk}^{(t)}}{\sum_{j',k'} x_{i,j'k'}^{(t)}} + (1 - \gamma) \frac{\sqrt{a_{i,j} b_{i,k}}}{\sum_{j',k'} \sqrt{a_{i,j'} b_{i,k'}}} \quad (3.8)$$

for $j = 1, \dots, m_i$, $k = 1, \dots, n_i$. Here, $\gamma \in [0, 1]$ stands for the interpolation factor. Such a shrinkage strategy is widely adopted in the subsampling literature [61, 83].

Given a sampling parameter $n_{s,i} \in \mathbb{N}$, the sparse approximation $\hat{\Psi}_i^{(t)} = (\hat{\psi}_{i,jk}^{(t)}) \in \mathbb{R}^{m_i \times n_i}$ for the kernel matrix $\Psi_i^{(t)}$ is constructed in accordance with the Poisson sampling principle, i.e.,

$$\hat{\psi}_{i,jk}^{(t)} := \begin{cases} \psi_{i,jk}^{(t)}/p_{i,jk}^{(t)*}, & \text{with probability } p_{i,jk}^{(t)*} := \min \left\{ 1, n_{s,i} \cdot p_{i,jk}^{(t)} \right\}, \\ 0, & \text{otherwise,} \end{cases} \quad (3.9)$$

where the denominators ensure the unbiasedness of the random approximation (see Appendix B). We denote the sampled set of indices as $\mathcal{I}_i^{(t)} \subseteq \{(j, k) : j = 1, \dots, m_i, k = 1, \dots, n_i\}$. Note that $\mathbb{E}(|\mathcal{I}_i^{(t)}|) = \sum_{j,k} p_{i,jk}^{(t)*} \leq n_{s,i} \sum_{j,k} p_{i,jk}^{(t)} = n_{s,i}$, which indicates that $n_{s,i}$ is an upper bound for the expected number of nonzero entries in $\hat{\Psi}_i^{(t)}$. Although $|\mathcal{I}_i^{(t)}|$ fluctuates, it still concentrates near its expectation with high probability [4], ensuring that the computational cost remains manageable.

Replacing $\Psi_i^{(t)}$ in subproblem (3.4) with $\hat{\Psi}_i^{(t)}$, we obtain

$$\min_{\tilde{\mathbf{u}}_i, \tilde{\mathbf{v}}_i} q_i(\tilde{\mathbf{u}}_i, \tilde{\mathbf{v}}_i; \lambda_i^{(t)}, \hat{\Psi}_i^{(t)}), \quad (3.10)$$

which is the dual of¹⁾

$$\begin{aligned} \min_{X_i} \quad & \left\langle \hat{C}_i^{(t)}, X_i - X_i^{(t)} \right\rangle + \lambda_i^{(t)} h(X_i), \\ \text{s.t.} \quad & X_i \mathbf{1}_{n_i} = \mathbf{a}_i, \quad X_i^\top \mathbf{1}_{m_i} = \mathbf{b}_i, \quad (X_i)_{(\mathcal{I}_i^{(t)})^c} = 0. \end{aligned} \quad (3.11)$$

Here, $\hat{C}_i^{(t)} = (\hat{c}_{i,jk}^{(t)}) \in \mathbb{R}^{m_i \times n_i}$ is the *effective cost matrix*, defined as

$$\hat{c}_{i,jk}^{(t)} := \begin{cases} c_{i,jk}^{(t)} + \lambda_i^{(t)} \log(p_{i,jk}^{(t)*}), & \text{if } (j, k) \in \mathcal{I}_i^{(t)}, \\ c_{i,jk}^{(t)}, & \text{otherwise.} \end{cases} \quad (3.12)$$

Note that in $\hat{C}_i^{(t)}$ only the entries indexed by $\mathcal{I}_i^{(t)}$ are required to form $\hat{\Psi}_i^{(t)}$. To solve subproblem (3.10) and (3.11), it suffices to substitute $\Psi_i^{(t)}$ with $\hat{\Psi}_i^{(t)}$ in the Sinkhorn algorithm (3.6).

We call the ERALM method equipped with sampling the S-ERALM method, which is summarized in Algorithm 2. The maximum expected numbers of sampled indices are controlled by $\{n_{s,i}\}_{i=1}^N \subseteq \mathbb{N}$.

¹⁾The strong duality holds if subproblem (3.11) is feasible with $\mathcal{I}_i^{(t)}$. The same will be true later, for subproblem (3.20).

Algorithm 2 The S-ERALM method for solving problem (1.1).

Input: $X_i^{(0)} \in \mathbb{R}^{m_i \times n_i}$, $\mathbf{a}_i \in \mathbb{R}^{m_i}$, $\mathbf{b}_i \in \mathbb{R}^{n_i}$ ($i = 1, \dots, N$), $\gamma \in [0, 1]$, $\{n_{s,i}\}_{i=1}^N \subseteq \mathbb{N}$, $t_{\max} \in \mathbb{N}$.

- 1: Set $t := 0$.
- 2: **while** certain conditions are not satisfied **and** $t < t_{\max}$ **do**
- 3: **for** $i = 1, \dots, N$ **do**
- 4: Select a regularization parameter $\lambda_i^{(t)} > 0$ and a step size $\alpha_i^{(t)} \in (0, 1]$.
- 5: Randomly pick a subset $\mathcal{I}_i^{(t)} \subseteq \{(j, k) : j = 1, \dots, m_i, k = 1, \dots, n_i\}$ following the Poisson sampling with $P_i^{(t)} = (p_{i,jk}^{(t)}) \in \mathbb{R}_+^{m_i \times n_i}$ given by (3.8) and $n_{s,i}$.
- 6: Construct $\hat{\Psi}_i^{(t)} = (\hat{\psi}_{i,jk}^{(t)}) \in \mathbb{R}^{m_i \times n_i}$ as in the formula (3.9).
- 7: Solve subproblem (3.10) or (3.11) to obtain $\tilde{X}_i^{(t+1)} \in \mathbb{R}^{m_i \times n_i}$.
- 8: Update $X_i^{(t+1)} := (1 - \alpha_i^{(t)}) X_i^{(t)} + \alpha_i^{(t)} \tilde{X}_i^{(t+1)} \in \mathbb{R}^{m_i \times n_i}$.
- 9: **end for**
- 10: Set $t := t + 1$.
- 11: **end while**

Output: Approximate solution $(X_1^{(t)}, \dots, X_N^{(t)}) \in \bigtimes_{i=1}^N \mathbb{R}^{m_i \times n_i}$.

3.3 KL divergence-based alternating linearized minimization

We now turn to methods involving a direct penalization of the distance between X_i and $X_i^{(t)}$. In each iteration, the PALM method solves several proximal linearized subproblems of the form

$$\min_{X_i} \left\langle C_i^{(t)}, X_i - X_i^{(t)} \right\rangle + \frac{\mu_i^{(t)}}{2} \|X_i - X_i^{(t)}\|^2, \text{ s.t. } X_i \in \mathcal{U}(\mathbf{a}_i, \mathbf{b}_i), \quad (3.13)$$

where $\mu_i^{(t)} > 0$ refers to the proximal parameter and $C_i^{(t)}$ is defined in equation (3.2). Solving subproblem (3.13) is equivalent to projecting $X_i^{(t)} - C_i^{(t)}/\mu_i^{(t)}$ onto $\mathcal{U}(\mathbf{a}_i, \mathbf{b}_i)$. In this part, we replace the proximal term $\|X_i - X_i^{(t)}\|^2/2$ by the KL divergence $\text{KL}(X_i; X_i^{(t)})$, which is the Bregman distance induced by the negative entropy (see section 2.2), and obtain

$$\min_{X_i} \left\langle C_i^{(t)}, X_i - X_i^{(t)} \right\rangle + \mu_i^{(t)} \text{KL}(X_i; X_i^{(t)}), \text{ s.t. } X_i \mathbf{1}_{n_i} = \mathbf{a}_i, X_i^\top \mathbf{1}_{m_i} = \mathbf{b}_i. \quad (3.14)$$

This is motivated by the Bregman proximal point algorithms for convex optimization [80, 82] and the Bregman distance-based PALM methods for nonconvex optimization with relatively smooth objectives [3, 55]. We describe the KL divergence-based alternating linearized minimization (KLALM) method in Algorithm 3.

As in the previous subsection, we write out the dual of subproblem (3.14):

$$\min_{\mathbf{u}_i, \mathbf{v}_i} q_i(\mathbf{u}_i, \mathbf{v}_i; \mu_i^{(t)}, \Phi_i^{(t)}), \quad (3.15)$$

where q_i is defined in problem (3.4), $\mathbf{u}_i \in \mathbb{R}^{m_i}$, $\mathbf{v}_i \in \mathbb{R}^{n_i}$ are the dual variables associated with the equality constraints and the kernel matrix

$$\Phi_i^{(t)} := \exp \left(-C_i^{(t)} / \mu_i^{(t)} \right) \odot X_i^{(t)} \in \mathbb{R}^{m_i \times n_i}. \quad (3.16)$$

The Sinkhorn algorithm (3.6) thus still applies, yet with $\Psi_i^{(t)}$ changed to $\Phi_i^{(t)}$.

Algorithm 3 The KLALM method for solving problem (1.1).

Input: $X_i^{(0)} \in \mathbb{R}^{m_i \times n_i}$, $\mathbf{a}_i \in \mathbb{R}^{m_i}$, $\mathbf{b}_i \in \mathbb{R}^{n_i}$ ($i = 1, \dots, N$), $t_{\max} \in \mathbb{N}$.

- 1: Set $t := 0$.
- 2: **while** certain conditions are not satisfied **and** $t < t_{\max}$ **do**
- 3: **for** $i = 1, \dots, N$ **do**
- 4: Select a proximal parameter $\mu_i^{(t)} > 0$.
- 5: Compute $C_i^{(t)} \in \mathbb{R}^{m_i \times n_i}$ as in the formula (3.2).
- 6: Solve subproblem (3.14) or (3.15) to obtain $X_i^{(t+1)} \in \mathbb{R}^{m_i \times n_i}$.
- 7: **end for**
- 8: Set $t := t + 1$.
- 9: **end while**

Output: Approximate solution $(X_1^{(t)}, \dots, X_N^{(t)}) \in \times_{i=1}^N \mathbb{R}^{m_i \times n_i}$.

3.4 Sampling-based variant of the KLALM method

Since subproblem (3.14) in the KLALM method admits the following multiplicative expression for the unique optimal solution:

$$X_i^{(t+1,\star)} := \text{Diag} \left(\exp \left(\frac{\mathbf{u}_i^{(t,\star)}}{\mu_i^{(t)}} \right) \right) \Phi_i^{(t)} \text{Diag} \left(\exp \left(\frac{\mathbf{v}_i^{(t,\star)}}{\mu_i^{(t)}} \right) \right) \in \mathbb{R}^{m_i \times n_i}, \quad (3.17)$$

where $(\mathbf{u}_i^{(t,\star)}, \mathbf{v}_i^{(t,\star)}) \in \mathbb{R}^{m_i} \times \mathbb{R}^{n_i}$ is an optimal solution of the dual (3.15), we could likewise employ sparse approximation on $\Phi_i^{(t)}$ for better scalability. We define the sparse approximation $\hat{\Phi}_i^{(t)} = (\hat{\varphi}_{i,jk}^{(t)}) \in \mathbb{R}^{m_i \times n_i}$ for $\Phi_i^{(t)} = (\varphi_{i,jk}^{(t)})$ as

$$\hat{\varphi}_{i,jk}^{(t)} := \begin{cases} \varphi_{i,jk}^{(t)}/p_{i,jk}^{(t)*}, & \text{with probability } p_{i,jk}^{(t)*}, \\ 0, & \text{otherwise,} \end{cases} \quad (3.18)$$

where $p_{i,jk}^{(t)*}$ is given as in equation (3.9). Replacing $\Phi_i^{(t)}$ in subproblem (3.15) with $\hat{\Phi}_i^{(t)}$, we obtain

$$\min_{\mathbf{u}_i, \mathbf{v}_i} q_i(\mathbf{u}_i, \mathbf{v}_i; \mu_i^{(t)}, \hat{\Phi}_i^{(t)}), \quad (3.19)$$

which is the dual of

$$\begin{aligned} \min_{X_i} \quad & \langle \hat{C}_i^{(t)}, X_i - X_i^{(t)} \rangle + \mu_i^{(t)} \text{KL}(X_i; X_i^{(t)}), \\ \text{s.t.} \quad & X_i \mathbf{1}_{n_i} = \mathbf{a}_i, \quad X_i^\top \mathbf{1}_{m_i} = \mathbf{b}_i, \quad (X_i)_{(\mathcal{I}_i^{(t)})^c} = 0. \end{aligned} \quad (3.20)$$

The Sinkhorn algorithm (3.6) then applies with $\Psi_i^{(t)}$ replaced by $\hat{\Phi}_i^{(t)}$.

While everything seems to go smoothly, we shall point out some distinctions: samplings over iterations will not improve the supports to be optimized and can even result in infeasible subproblems. Recalling the definition (3.16) of $\Phi_i^{(t)}$, one knows from relation (3.17) that $x_{i,jk}^{(t)} = 0$ implies $x_{i,jk}^{(t+1,\star)} = 0$. If the subproblems are exactly solved and we perform samplings in two successive iterations (say, t and $t+1$), the i th support to be optimized in the $(t+1)$ th iteration is $\mathcal{I}_i^{(t-1)} \cap \mathcal{I}_i^{(t)}$, which is a subset of $\mathcal{I}_i^{(t-1)}$. In implementation, we adopt the Sinkhorn algorithm to inexactly solve the subproblems. Its iterative schemes (3.6) also imply the inheritance of zero entries. In both cases, samplings over iterations do not improve the supports to be optimized and can lead to infeasibility due to repeated intersections. To this end, we choose to perform sampling only in some critical iteration, say $\hat{t} \in \mathbb{N}$. For $t < \hat{t}$, the kernel matrices $\Phi_i^{(t)}$ are fully computed. With properly chosen

Algorithm 4 The S-KLALM method for solving problem (1.1).

Input: $X_i^{(0)} \in \mathbb{R}^{m_i \times n_i}$, $\mathbf{a}_i \in \mathbb{R}^{m_i}$, $\mathbf{b}_i \in \mathbb{R}^{n_i}$ ($i = 1, \dots, N$), $\gamma \in [0, 1]$, $\{n_{s,i}\}_{i=1}^N \subseteq \mathbb{N}$, $\hat{t}, t_{\max} \in \mathbb{N}$.

- 1: Set $t := 0$.
- 2: **while** certain conditions are not satisfied **and** $t < t_{\max}$ **do**
- 3: **for** $i = 1, \dots, N$ **do**
- 4: Select a proximal parameter $\mu_i^{(t)} > 0$.
- 5: **if** $t = \hat{t}$ **then**
- 6: Randomly pick a subset $\mathcal{I}_i^{(t)} \subseteq \{(j, k) : j = 1, \dots, m_i, k = 1, \dots, n_i\}$ following the Poisson sampling with $P_i^{(t)} = (p_{i,jk}^{(t)}) \in \mathbb{R}_+^{m_i \times n_i}$ given by (3.8) and $n_{s,i}$.
- 7: **end if**
- 8: **if** $t < \hat{t}$ **then**
- 9: Let $\hat{\Phi}_i^{(t)} := \Phi_i^{(t)}$ defined in the formula (3.16).
- 10: **else**
- 11: Construct $\hat{\Phi}_i^{(t)} \in \mathbb{R}^{m_i \times n_i}$ as in the formula (3.18) with $\mathcal{I}_i^{(\hat{t})}$ and $P_i^{(\hat{t})}$.
- 12: **end if**
- 13: Solve subproblem (3.19) or (3.20) to obtain $X_i^{(t+1)} \in \mathbb{R}^{m_i \times n_i}$.
- 14: **end for**
- 15: Set $t := t + 1$.
- 16: **end while**

Output: Approximate solution $(X_1^{(t)}, \dots, X_N^{(t)}) \in \bigtimes_{i=1}^N \mathbb{R}^{m_i \times n_i}$.

\hat{t} , we can expect $X_i^{(\hat{t})}$ to capture well the sparsity pattern. Then, for $t > \hat{t}$, the selected indices are fixed and no sampling occurs.

We summarize the KLALM method with sampling (the S-KLALM method) in Algorithm 4.

3.5 Computational complexities

We compare the single-iteration computational complexities of the ERALM, S-ERALM, KLALM, and S-KLALM methods. In particular, we focus on the cost of computing the (sparse) kernel matrices, performing importance sampling, and the subiterations within the Sinkhorn algorithm (3.6). For a summary, see Table 1, where $s_{\max} \in \mathbb{N}$ is the maximum subiteration number, and we assume $m_i \equiv m \in \mathbb{N}$, $n_i \equiv n \in \mathbb{N}$, $n_{s,i} \equiv n_s \in \mathbb{N}$ ($i = 1, \dots, N$) for better readability.

Armed with warm starts, the Sinkhorn algorithm usually terminates after few subiterations. As such, the S-KLALM method enjoys the lowest complexity per iteration when $t > \hat{t}$. Given $n_s \sim (m+n)^{1+\tau}$ with $\tau \in (0, 1)$, this advantage becomes more evident as τ tends to 0 or $m+n$ goes to $+\infty$.

4 Convergence analysis

In this section, we present the convergence and asymptotic properties of both the ERALM and S-ERALM methods. Since the KL divergence lacks local Lipschitz smoothness, the analysis for the KLALM and S-KLALM methods can be highly nontrivial and is left over as future work. Nevertheless, the theoretical results obtained for the ERALM and S-ERALM methods already deserve a whistle. To the best of our knowledge, our work is the first attempt in incorporating randomized matrix sparsification into numerical methods for multi-block nonconvex optimization, while establishing the convergence and asymptotic properties.

All the results are obtained by assuming that the subproblems are exactly solved. Furthermore, the subproblems of the S-ERALM method are assumed to be feasible in their primal forms in all iter-

Table 1: A comparison of computational complexities.

Ingredients in one iteration	ERALM	KLALM
Kernel matrices	Nmn entries	Nmn entries
Subiterations	$s_{\max} \times Nmn$	$s_{\max} \times Nmn$
Ingredients in one iteration	S-ERALM	S-KLALM
Sampling	$\mathcal{O}(Nmn)$	$\mathcal{O}(Nmn)$ ($t = \hat{t}$)
Kernel matrices	Nn_s entries	Nmn entries ($t < \hat{t}$) Nn_s entries ($t \geq \hat{t}$)
Subiterations	$s_{\max} \times Nn_s$	$s_{\max} \times Nmn$ ($t < \hat{t}$) $s_{\max} \times Nn_s$ ($t \geq \hat{t}$)

ations, so that the strong duality holds. Although at present no analysis is present on the conditions under which the latter assumption is fulfilled, we remark that sampling parameters without careful selection (say, $n_{s,i} = \lfloor (m_i + n_i)^{1.5} \rfloor$) already meets the demand in numerical simulations (see section 5).

To characterize the stationarity violation for problem (1.1), we define the residual functions: for any $X := (X_1, \dots, X_N) \in \bigtimes_{i=1}^N \mathbb{R}^{m_i \times n_i}$ and $i = 1, \dots, N$,

$$R_i(X) := \max_{T \in \mathcal{U}(\mathbf{a}_i, \mathbf{b}_i)} \langle \nabla_i f(X), X_i - T \rangle. \quad (4.1)$$

Moreover, let $R := \sum_{i=1}^N R_i$. It is not hard to verify that $R(X) \geq 0$ for any $X \in \bigtimes_{i=1}^N \mathcal{U}(\mathbf{a}_i, \mathbf{b}_i)$ and that $X \in \bigtimes_{i=1}^N \mathcal{U}(\mathbf{a}_i, \mathbf{b}_i)$ is a Karush-Kuhn-Tucker point of problem (1.1) if and only if $R(X) = 0$ holds. For the iterate $X^{(t)}$ generated by the ERALM or S-ERALM method, $R(X^{(t)})$ can thus characterize the stationarity violation at $X^{(t)}$.

We assume Lipschitz smoothness of f over the feasible region, which holds automatically, e.g., for the quantum physics application (see section 5.1).

Assumption 1. *The gradient of the function f is Lipschitz continuous over $\bigtimes_{i=1}^N \mathcal{U}(\mathbf{a}_i, \mathbf{b}_i)$, i.e., there exists an $L \geq 0$ such that, for $i = 1, \dots, N$,*

$$\|\nabla_i f(X) - \nabla_i f(X')\| \leq L \|X - X'\| \quad \text{for all } X, X' \in \bigtimes_{i=1}^N \mathcal{U}(\mathbf{a}_i, \mathbf{b}_i).$$

We first give the convergence and asymptotic properties of the ERALM method. The proofs of the theorem and corollary below can be found in Appendix A.

Theorem 1. *Suppose that Assumption 1 holds. Let $\{X^{(t)}\}$ be the iterate sequence generated by the ERALM method when*

$$t_{\max} \geq \frac{f(X^{(0)}) - \underline{f}}{2\bar{d}^2 LN(2\sqrt{N} + 1)}, \quad \alpha_1^{(t)} = \dots = \alpha_N^{(t)} \equiv \alpha := \frac{1}{d} \sqrt{\frac{f(X^{(0)}) - \underline{f}}{2LN(2\sqrt{N} + 1)t_{\max}}}, \quad (4.2)$$

and $\lambda_1^{(t)} = \dots = \lambda_N^{(t)} \equiv \lambda$ for $0 \leq t \leq t_{\max}$ and subproblems (3.3) are exactly solved, where $\underline{f} \in \mathbb{R}$ is less than or equal to the optimal value of problem (1.1),

$$d_i := \min \{ \sqrt{m_i} \|\mathbf{a}_i\|_\infty, \sqrt{n_i} \|\mathbf{b}_i\|_\infty \} \quad (i = 1, \dots, N), \quad \bar{d} := \max_{i=1}^N d_i.$$

Then

$$0 \leq \frac{1}{t_{\max}} \sum_{t=0}^{t_{\max}-1} R(X^{(t)}) \leq 2\bar{d}(2N+1) \sqrt{\frac{L(f(X^{(0)}) - \underline{f})}{t_{\max}}} + N\lambda\bar{h}, \quad (4.3)$$

where $\bar{h} := -\min_{i=1}^N h(\mathbf{a}_i \mathbf{b}_i^\top)$.

Remark 1. In Theorem 1, as well as the subsequent Corollary 1, Theorem 2 and Corollary 2, the requirements $\alpha_1^{(t)} = \dots = \alpha_N^{(t)}$ and $\lambda_1^{(t)} = \dots = \lambda_N^{(t)}$ are assumed for better readability. The theorems and corollaries can be extended without too much difficulty to the case where $\alpha_i^{(t)}$ and $\lambda_i^{(t)}$ vary across $i \in \{1, \dots, N\}$ and $0 \leq t \leq t_{\max}$.

The errors related to the entropy terms are inevitable because the objective function in the subproblem of the ERALM method is not a local approximation for f . Nevertheless, the right-hand side of inequality (4.3) vanishes in the limit $\sum_{i=1}^N (m_i + n_i) \rightarrow +\infty$ after choosing proper t_{\max} and λ and imposing Assumption 2 below. This is of particular importance for large-scale applications.

- Assumption 2.**
- (i) There exists an $\underline{f} \in \mathbb{R}$ such that the optimal value of problem (1.1) is lower bounded by \underline{f} for any $\{m_i\}_{i=1}^N, \{n_i\}_{i=1}^N \subseteq \mathbb{N}$.
 - (ii) There exists a $q > 0$ such that, for any $\{m_i\}_{i=1}^N, \{n_i\}_{i=1}^N \subseteq \mathbb{N}$, $\mathbf{a}_i^\top \mathbf{1}_{m_i} = \mathbf{b}_i^\top \mathbf{1}_{n_i} = 1$, $\max_j a_{i,j} \leq q \cdot \min_j a_{i,j}$, and $\max_k b_{i,k} \leq q \cdot \min_k b_{i,k}$.
 - (iii) There exists a $\theta \geq 0$ such that, for any $\{m_i\}_{i=1}^N, \{n_i\}_{i=1}^N \subseteq \mathbb{N}$, the block Lipschitz constant $L = \mathcal{O}(\sum_{i=1}^N (m_i + n_i)^\theta)$.
 - (iv) There exists a $\xi \geq 0$ such that $\max_{i=1}^N (m_i + n_i) / \min_{i=1}^N (m_i + n_i) \leq \xi$ for any $\{m_i\}_{i=1}^N, \{n_i\}_{i=1}^N \subseteq \mathbb{N}$.

Remark 2. Items (i) and (ii) in Assumption 2 are motivated by the application of interest, where a continuous problem arises and one seeks to solve its discretized version. For example, in the application of strongly correlated quantum physics (see section 5.1 and also [23, 42]), the discretized problem has a natural objective lower bound $\underline{f} = 0$ due to the nonnegativity of energy, \mathbf{a}_i and \mathbf{b}_i are discretization of the so-called single-particle density, whose integral is a prescribed constant. Incidentally, up to normalizing f , we assume the total mass of marginals to be 1 in item (ii). For simplicity in theoretical analysis, we adopt items (iii) and (iv), which can be further relaxed to some extent.

Corollary 1. Suppose that Assumptions 1 and 2 (i)-(iii) hold. Let $\{X^{(t)}\}$ be the iterate sequence generated by the ERALM method when

$$\begin{aligned} t_{\max} &\geq \max \left\{ \Omega \left(\sum_{i=1}^N (m_i + n_i)^\eta \right), \frac{f(X^{(0)}) - \underline{f}}{2\bar{d}^2 L N (2\sqrt{N} + 1)} \right\}, \quad f(X^{(0)}) \leq M, \\ \alpha_1^{(t)} &= \dots = \alpha_N^{(t)} \equiv \alpha, \quad \lambda_1^{(t)} = \dots = \lambda_N^{(t)} \equiv \lambda = o \left(\frac{1}{\sum_{i=1}^N \log m_i n_i} \right) \end{aligned}$$

for $0 \leq t \leq t_{\max}$ and subproblems (3.3) are exactly solved, where $\eta (> \theta)$ and M are constants independent from $\{m_i\}_{i=1}^N$ and $\{n_i\}_{i=1}^N$ and α is defined in equation (4.2). Then $\sum_{t=0}^{t_{\max}-1} R(X^{(t)}) / t_{\max} \rightarrow 0$ as $\sum_{i=1}^N (m_i + n_i) \rightarrow +\infty$.

With sampling, the analysis for the S-ERALM method is much more involved, in that the entrywise matrix sparsification results in additional errors. We establish the convergence and asymptotic results with the help of the theory of randomized matrix sparsification as well as the following Assumption 3; details are provided in Appendix B.

Assumption 3. (i) There exist constants $\nu \in (1/2, 1]$, $c_1, c_2, \hat{c}_2 > 0$ such that, for any $0 \leq t \leq t_{\max}$ and $i = 1, \dots, N$,

$$\|\Psi_i^{(t)}\|_2 \geq \frac{(m_i + n_i)^\nu}{c_1}, \quad \kappa(\Psi_i^{(t)}) \leq c_2, \quad \kappa(\hat{\Psi}_i^{(t)}) \leq \hat{c}_2.$$

(ii) The interpolation factor γ is less than 1 and there exists an $\varepsilon > 0$ such that, for $i = 1, \dots, N$,

$$\frac{1}{\max_{j,k,t} p_{i,jk}^{(t)}} \geq n_{s,i} \geq \frac{8(m_i + n_i)^{1-2\nu} \log^4(m_i + n_i)}{(1-\gamma)w_i \log^4(1+\varepsilon)},$$

where $w_i := \min_{j,k} \sqrt{a_{i,j} b_{i,k}} / \sum_{j',k'} \sqrt{a_{i,j'} b_{i,k'}}$.

Remark 3. Assumption 3 is adopted for simplicity in theoretical analysis. In particular, since $w_i \leq 1/m_i n_i$, item (ii) implies the following lower bound for $n_{s,i}$:

$$\frac{8(m_i + n_i)^{1-2\nu} \log^4(m_i + n_i)}{(1-\gamma)w_i \log^4(1+\varepsilon)} \geq \frac{8}{(1-\gamma) \log^4(1+\varepsilon)} \frac{m_i n_i}{(m_i + n_i)^{2\nu-1}} \log^4(m_i + n_i),$$

which is of lower order than $m_i n_i$ because $\nu \in (1/2, 1]$. The condition $n_{s,i} p_{i,jk}^{(t)} \leq 1$ is a convention prevalent in the literature of Poisson sampling [76, 77]. Note that under Assumption 2 (ii), the entries in \mathbf{a}_i or \mathbf{b}_i are of the same order. Therefore, $w_i = \Theta(1/m_i n_i)$ and $p_{i,jk}^{(t)} = \Theta(1/m_i n_i)$ after choosing a proper value for γ . Item (ii) then holds if we choose $n_{s,i} = \Theta(m_i n_i \log^4(m_i + n_i)/(m_i + n_i)^{2\nu-1})$ and the values of ε and γ are independent from $\{m_i\}_{i=1}^N$ and $\{n_i\}_{i=1}^N$.

Theorem 2. Suppose that Assumption 1 holds. Let $\{X^{(t)}\}$ be the iterate sequence generated by the S-ERALM method when

$$t_{\max} \geq \frac{f(X^{(0)}) - \underline{f}}{2\bar{d}^2 L N (2\sqrt{N} + 1)},$$

$\alpha_1^{(t)} = \dots = \alpha_N^{(t)} \equiv \alpha$, and $\lambda_1^{(t)} = \dots = \lambda_N^{(t)} \equiv \hat{\lambda}$ for $0 \leq t \leq t_{\max}$, subproblems (3.11) are feasible and exactly solved, and Assumption 3 is fulfilled, where $\underline{f} \in \mathbb{R}$ is less than or equal to the optimal value of problem (1.1) and α is defined in equation (4.2). Then, for any $m_i + n_i > \max\{152, e^{\sqrt{c_3}}\}$ ($i = 1, \dots, N$), $\zeta > 0$, and $\iota > 0$, with probability no less than

$$\prod_{i=1}^N \left\{ \left[1 - 2 \exp \left(-\frac{16\zeta^2}{\varepsilon^4} \log^4(m_i + n_i) \right) \right] [1 - \exp(-2\iota^2 m_i n_i)] \right\}^{t_{\max}},$$

there holds

$$\begin{aligned} 0 \leq \frac{1}{t_{\max}} \sum_{t=0}^{t_{\max}-1} R(X^{(t)}) &\leq 2\bar{d}(2N+1) \sqrt{\frac{L(f(X^{(0)}) - \underline{f})}{t_{\max}}} + 2N\hat{\lambda}\bar{h} \\ &+ \hat{\lambda}\bar{d} \sum_{i=1}^N \sqrt{n_{s,i} + \iota \cdot m_i n_i} \log \frac{1}{(1-\gamma)w_i n_{s,i}} + \hat{\lambda} \sum_{i=1}^N \frac{\hat{c}_2 c_3}{\log^2(m_i + n_i) - c_3}, \end{aligned} \tag{4.4}$$

where $c_3 := c_1(1 + \varepsilon + \zeta) \log^2(1 + \varepsilon)$.

Corollary 2. Suppose that Assumptions 1 and 2 hold. Let $\{X^{(t)}\}$ be the iterate sequence generated by the S-ERALM method when

$$t_{\max} = \Theta \left(\sum_{i=1}^N (m_i + n_i)^\eta \right) \text{ satisfying } t_{\max} \geq \frac{f(X^{(0)}) - \underline{f}}{2\bar{d}^2 L N (2\sqrt{N} + 1)}, \quad f(X^{(0)}) \leq M,$$

$$\alpha_1^{(t)} = \dots = \alpha_N^{(t)} \equiv \alpha, \quad n_{s,i} = \Theta\left(\frac{m_i n_i}{(m_i + n_i)^{2\nu-1}} \log^4(m_i + n_i)\right),$$

$$\lambda_1^{(t)} = \dots = \lambda_N^{(t)} \equiv \hat{\lambda} = o\left(\frac{1}{\sum_{i=1}^N \sqrt{m_i n_i} \log(m_i + n_i)}\right)$$

for $0 \leq t \leq t_{\max}$, subproblems (3.11) are feasible and exactly solved, and Assumption 3 is fulfilled, where $c_1, c_2, \hat{c}_2, \varepsilon, \eta(> \theta), \gamma, \nu, \xi$, and M are independent from $\{m_i\}_{i=1}^N$ and $\{n_i\}_{i=1}^N$ and α is defined in equation (4.2). Then $\sum_{t=0}^{t_{\max}-1} R(X^{(t)})/t_{\max} \rightarrow 0$ as $\sum_{i=1}^N (m_i + n_i) \rightarrow +\infty$ with probability going to 1.

5 Numerical experiments

We demonstrate the efficiency of the proposed methods via numerical results on model one/two/three-dimensional strongly correlated electron systems. We first describe the related optimization problem of the form (1.1) mathematically and provide experimental details, including systems to be simulated and default algorithmic settings. Then numerical comparisons are conducted among the PALM method [14, 43] and the newly designed four methods on the one-dimensional systems. We integrate those with favorable performances into a cascadic multigrid optimization framework for the simulations of two/three-dimensional systems. A first visualization of approximate OT maps between electron positions in three-dimensional contexts is provided. Finally, we test the scalability of methods with respect to the problem size as well as the number of variable blocks.

5.1 Problem description

From the strong-interaction limit of density functional theory [37], the strongly correlated quantum systems in the strictly correlated regime can be well understood by solving the multimarginal optimal transport problems with Coulomb cost (MMOT) [5, 6, 11, 20, 23, 26, 38, 47, 48, 62]:

$$\begin{aligned} \min_{\pi} \quad & \int_{(\mathbb{R}^d)^{N_e}} c_{ee}(\mathbf{r}_1, \dots, \mathbf{r}_{N_e}) d\pi(\mathbf{r}_1, \dots, \mathbf{r}_N), \\ \text{s.t.} \quad & \int_{(\mathbb{R}^d)^{N_e-1}} d\pi(\mathbf{r}_1, \dots, \mathbf{r}_{i-1}, \cdot, \mathbf{r}_{i+1}, \dots, \mathbf{r}_{N_e}) = \frac{\rho}{N_e}, \quad i = 1, \dots, N_e. \end{aligned} \quad (5.1)$$

Here, $d \in \{1, 2, 3\}$ is the system dimension, $N_e \in \mathbb{N}$ is the number of electrons, $\mathbf{r}_i \in \mathbb{R}^d$ refers to the position of the i th electron ($i = 1, \dots, N_e$), $\rho : \mathbb{R}^d \rightarrow \mathbb{R}_+$ is the single-particle density, satisfying $\int \rho = N_e$, $c_{ee}(\mathbf{r}_1, \dots, \mathbf{r}_{N_e}) := \sum_{1 \leq i < j} 1/\|\mathbf{r}_i - \mathbf{r}_j\|$ stands for the N -particle Coulomb potential, and π is a joint probability measure of N_e electron positions.

Since the dimension of the search space in the MMOT (5.1) scales exponentially with the number of electrons, one could adopt a Monge-like ansatz [23, 42], which characterizes the electron-electron couplings explicitly, to transform the MMOT into the following problem:

$$\begin{aligned} \min_{\{\gamma_i\}_{i=2}^{N_e}} \quad & \sum_{i=2}^{N_e} \int_{(\mathbb{R}^d)^2} \frac{\rho(\mathbf{r}) \gamma_i(\mathbf{r}, \mathbf{r}')}{\|\mathbf{r} - \mathbf{r}'\|} d\mathbf{r} d\mathbf{r}' + \sum_{2 \leq i < j} \int_{(\mathbb{R}^d)^3} \frac{\rho(\mathbf{r}) \gamma_i(\mathbf{r}, \mathbf{r}') \gamma_j(\mathbf{r}, \mathbf{r}'')}{\|\mathbf{r}' - \mathbf{r}''\|} d\mathbf{r} d\mathbf{r}' d\mathbf{r}'', \\ \text{s.t.} \quad & \int_{\mathbb{R}^d} \gamma_i(\cdot, \mathbf{r}_i) d\mathbf{r}_i = 1, \quad \int_{\mathbb{R}^d} \gamma_i(\mathbf{r}_1, \cdot) \rho(\mathbf{r}_1) d\mathbf{r}_1 = \rho, \quad \gamma_i \geq 0, \quad i = 2, \dots, N_e. \end{aligned} \quad (5.2)$$

Here, $\gamma_i : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}_+$ encodes the coupling between the first and i th electrons, $\rho \gamma_i$ can be understood as the joint probability density between their positions ($i = 2, \dots, N_e$).

To numerically solve problem (5.2), we confine the integral domain to some bounded $\Omega \subseteq \mathbb{R}^d$ and adopt finite elements-like discretization. In particular, we first define a mesh $\mathcal{T} = \{e_k\}_{k=1}^K$ to

divide Ω into K non-overlapping elements, i.e., $\cup_{k=1}^K e_k = \Omega$ and $e_k \cap e_{k'} = \emptyset$ whenever $k \neq k'$. Then we use a finite summation of Dirac measures to approximate ρ as $\rho \approx \sum_{k=1}^K \varrho_k \delta_{\mathbf{d}_k}$, where $\varrho_k := \int_{e_k} \rho$, $\mathbf{d}_k \in \mathbb{R}^d$ is the barycenter of the element e_k ($k = 1, \dots, K$). Let $\boldsymbol{\varrho} := [\varrho_1, \dots, \varrho_K] \in \mathbb{R}_{++}^K$ and $\Lambda := \text{Diag}(\boldsymbol{\varrho}) \in \mathbb{R}^{K \times K}$. The two-particle Coulomb potential and couplings are respectively discretized into $K \times K$ matrices $C = (c_{kl}) \in \mathbb{R}^{K \times K}$ and $X_i = (x_{i,kl})_{kl} \in \mathbb{R}^{K \times K}$ ($i = 2, \dots, N_e$), where, for $i = 2, \dots, N_e$ and $k, l = 1, \dots, K$,

$$c_{kl} := \begin{cases} \|\mathbf{d}_k - \mathbf{d}_l\|^{-1}, & \text{if } k \neq l, \\ 0, & \text{otherwise,} \end{cases} \quad x_{i,kl} := \frac{1}{|e_k|} \int_{e_k} \int_{e_l} \gamma_i(\mathbf{r}, \mathbf{r}') d\mathbf{r}' d\mathbf{r}.$$

Note that the diagonal entries in C are set to 0 to avoid numerical instability. We impose the following extra constraints on $\{X_i\}_{i=2}^{N_e}$ to maintain the model equivalence:

$$\text{Tr}(X_i) = 0, \quad i = 2, \dots, N_e; \quad \langle X_i, X_j \rangle = 0, \quad i, j = 2, \dots, N_e : i \neq j. \quad (5.3)$$

Intuitively, the constraints (5.3) exclude the cases where two electrons collide. After penalizing the constraints (5.3) in ℓ_1 form and transforming X_i to $Y_i := \Lambda X_i \in \mathbb{R}^{K \times K}$ ($i = 2, \dots, N_e$), we obtain a multi-block optimization problem over the transport polytopes of the form (1.1) (with $N = N_e - 1$):

$$\begin{aligned} \min_{\{Y_i\}_{i=2}^{N_e}} \quad & \sum_{i=2}^{N_e} \langle Y_i, C + \beta \Lambda^{-1} \rangle + \sum_{2 \leq i < j} \langle Y_i, \Lambda^{-1} Y_j C + \beta \Lambda^{-2} Y_j \rangle, \\ \text{s.t.} \quad & Y_i \in \mathcal{U}(\boldsymbol{\varrho}, \boldsymbol{\varrho}) \subseteq \mathbb{R}^{K \times K}, \quad i = 2, \dots, N_e. \end{aligned} \quad (5.4)$$

The matrix variable Y_i can be understood as the transport plan between the positions of the first and i th electrons ($i = 2, \dots, N_e$). It has been shown in [44] that there exists a $\hat{\beta} \geq 0$ such that the optimal solutions of (5.4) satisfy the constraints (5.3) whenever $\beta \geq \hat{\beta}$.

5.2 Systems under simulations

We consider eight one/two/three-dimensional (1D/2D/3D) systems. Table 2 contains their single-particle densities, domains of interest, and numbers of electrons. The component function $\rho_\alpha(\cdot; \mathbf{c})$ ($\alpha > 0$, $\mathbf{c} \in \mathbb{R}^d$) is defined as

$$\rho_\alpha(\mathbf{r}; \mathbf{c}) := \exp(-\alpha \|\mathbf{r} - \mathbf{c}\|^2), \quad \forall \mathbf{r} \in \mathbb{R}^d.$$

We illustrate the single-particle densities in Figure 1.

Remark 4. *Nearly all the single-particle densities in Table 2 comprise of Gaussian functions. These settings are reasonable because electrons tend to concentrate around the nuclei, which are represented by the potential wells. For example, the single-particle density of system 8 comprises of two Gaussian functions with weights 1 and 3, respectively. It can describe a dissociating lithium hydride [36], which has two nuclei with charge numbers 1 and 3, respectively. We shall note that the applicability of the proposed methods is independent from the constructions of the single-particle densities.*

5.3 Default settings

We employ either equimass or equisize discretization (to be fixed later) for problem (5.4). We set $\beta = 1$ for any K following [42], which is reasonable because the nonzero entries in both C and Λ^{-1} are of order K . For both the ERALM and S-ERALM methods, we adopt decreasing step sizes $\alpha_i^{(t)} = 1/(t+1)^{0.75}$, $i = 2, \dots, N_e$, to pursue convergence. In the S-ERALM and S-KLALM methods, we let $n_{s,i} = \lfloor K^{1.5} \rfloor$ ($i = 2, \dots, N_e$). All the proposed methods set the regularization or proximal parameters adaptively as

$$\lambda_i^{(t)} = \sigma \|\tilde{\mathbf{v}}_i^{(t)}\|_\infty / (20 \log(K)), \quad \mu_i^{(t)} = \sigma \|\mathbf{v}_i^{(t)}\|_\infty / (20 \log(K)), \quad i = 2, \dots, N_e, \quad (5.5)$$

Table 2: 1D/2D/3D systems used for simulations. The second column lists the unnormalized single-particle densities ρ , the third gives the domains Ω , and the last indicates the numbers of electrons N_e in systems.

System No.	ρ	Ω	N_e
1D systems			
1	$\cos(\pi r) + 1$	$[-1, 1]$	3
2	$2\rho_6(r; -0.5) + 1.5\rho_4(r; 0.5)$	$[-1.5, 1.5]$	3
3	$\rho_{1/\sqrt{\pi}}(r)$	$[-2, 2]$	7
4	$\rho_4(r; -2) + \rho_4(r; -1.5) + \rho_4(r; -1) + \rho_4(r; -0.5) + \rho_4(r; 2/3) + \rho_4(r; 4/3) + \rho_4(r; 2)$	$[-3, 3]$	7
2D systems			
5	$\rho_3(\mathbf{r}; [0, 0.96]^\top) + \rho_3(\mathbf{r}; [1.032, -0.84]^\top) + \rho_3(\mathbf{r}; [-1.032, -0.84]^\top)$	$[-3, 3]^2$	3
6	$2\rho_3(\mathbf{r}; [0, 1.2]^\top) + \rho_3(\mathbf{r}; [1.29, -1.05]^\top) + \rho_3(\mathbf{r}; [-1.29, -1.05]^\top)$	$[-3, 3]^2$	4
3D systems			
7	$\rho_3(\mathbf{r}; [-1, -1, -1]^\top) + \rho_3(\mathbf{r}; [1, 1, -1]^\top) + \rho_3(\mathbf{r}; [-1, 1, 1]^\top)$	$[-2, 2]^3$	3
8	$3\rho_4(\mathbf{r}; [-1, 0, 0]^\top) + \rho_4(\mathbf{r}; [1, 0, 0]^\top)$	$[-2, 2] \times [-1, 1]^2$	4

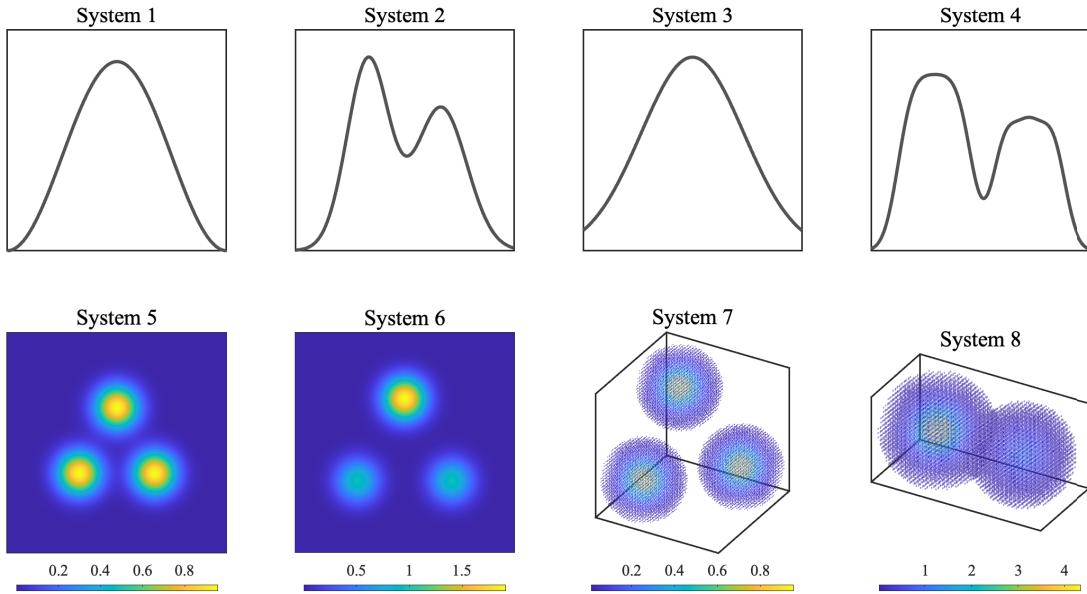


Figure 1: The illustrations of the single-particle densities listed in Table 2. For the 3D systems, we only show the regions where the values of the single-particle densities are larger than 0.01.

where $\sigma = 1$. The selections of γ and \hat{t} will be presented in section 5.4. We shall point out that better performances of the proposed methods can be expected with more careful parameter tuning. All the subproblems are solved using the Sinkhorn algorithm (3.6), with warm starts for acceleration. Incidentally, to circumvent possible underflow and achieve better convergence rates

[33] when using the Sinkhorn algorithm, we discard the entries in ϱ that are smaller than 0.1% of $\|\varrho\|_\infty$; this is also practically reasonable, in that the regions of low probabilities are far less important. With the abuse of notations, we still denote the truncated vector by ϱ so that other symbols remain unchanged. Regarding the stopping criteria, we terminate the Sinkhorn algorithm whenever the feasibility violation $\|Y_i^{(t+1)} \mathbf{1}_K - \varrho\|_\infty$ is less than 10^{-6} or the subiteration number arrives at $s_{\max} = 20$ ($i = 2, \dots, N_e$). We stop the outer loop if

$$\Delta^{(t)} := \frac{1}{N_e - 1} \sum_{i=2}^{N_e} \left\| \Lambda^{-1}(Y_i^{(t)} - Y_i^{(t-1)}) \right\|$$

falls below a prescribed $tol > 0$ or the iteration number reaches a prescribed $t_{\max} \in \mathbb{N}$. The specific values of tol and t_{\max} are detailed in the subsequent subsections. All the experiments presented are run in a platform with Intel(R) Xeon(R) Gold 6242R CPU @ 3.10GHz and 510GB RAM running MATLAB R2019b under Ubuntu 20.04.

For quantities of interest, we monitor the converged objective value (obj) and approximate the so-called strictly-correlated-electrons (SCE) potential [23, 30, 42] with the output dual variables. The SCE potential is the functional derivative of the optimal value of the MMOT with respect to ρ and is important for electronic structure calculations [23]. Taking the ERALM method for example, we approximate the SCE potential by $\mathbf{v} := \tilde{\mathbf{v}} - \min_{k=1}^K \{\tilde{v}_k\} \cdot \mathbf{1}_K \in \mathbb{R}^K$, where

$$\tilde{\mathbf{v}} := \frac{1}{N_e - 1} \sum_{i=2}^{N_e} \tilde{\mathbf{v}}_i \in \mathbb{R}^K,$$

and $\{\tilde{\mathbf{v}}_i\}_{i=2}^{N_e}$ are the dual solutions yielded by the Sinkhorn algorithm. In addition, for the cases where explicit constructions of the optimal solutions to problem (5.4) and the SCE potentials of the MMOT are available (e.g., in one-dimensional settings [25, 44]), we also evaluate the qualities of the converged solutions via the relative errors of the objective values (err_obj) and SCE potentials (err_sce). They are defined respectively as

$$\text{err_obj} := \left| \frac{\text{obj} - \text{obj}^*}{\text{obj}^*} \right|, \quad \text{err_sce} := \frac{\|\mathbf{v} - \mathbf{v}^*\|_\infty}{\|\mathbf{v}^*\|_\infty}.$$

Here, $\text{obj}^* \in \mathbb{R}$ denotes the optimal objective value of problem (5.4) and $\mathbf{v}^* \in \mathbb{R}^K$ refers to the vector made up by the values of the SCE potential at barycenters. For the efficiency comparison, we record the CPU time in seconds (T).

5.4 Algorithm comparisons

We conduct comparisons among the PALM method [14, 43] and the proposed four methods on the 1D systems in Table 2. In particular, we first test the S-ERALM methods with different sampling probabilities and the S-KLALM methods with different choices of \hat{t} . As a byproduct, we select default values of γ and \hat{t} . Secondly, we compare the proposed four methods. Those with favorable performances are tested against the PALM method in the third part and used for the simulations of 2D and 3D systems in section 5.6.

5.4.1 Comparisons among the S-ERALM methods with different sampling probabilities

We consider randomly generated sampling probabilities and importance sampling-based probabilities (3.8) ($\gamma \in \{0.1, 0.3, 0.5, 0.7, 0.9, 0.99, 0.999\}$) on system 1 with $K = 90$ (equimass discretization). For each setting of sampling probability, 10 random trials are generated by the built-in function “rand” in MATLAB. The stopping parameters are $tol = 5 \times 10^{-3}$ and $t_{\max} = +\infty$. We record the achieved err_obj, err_sce, and required T averaged over 10 trials for each setting in Table 3.

Table 3: The achieved err_obj, err_sce, and required T averaged over 10 trials given by the S-ERALM methods with different sampling probabilities on system 1 with $K = 90$ (equimass discretization).

Sampling Prob.	err_obj	err_sce	T
Random	0.4184	0.86	128.84
$\gamma = 0.1$	0.3098	0.69	120.16
$\gamma = 0.3$	0.1729	0.54	93.30
$\gamma = 0.5$	0.1044	0.42	65.10
$\gamma = 0.7$	0.0693	0.39	48.13
$\gamma = 0.9$	0.0597	0.37	35.05
$\gamma = 0.99$	0.0525	0.36	21.76
$\gamma = 0.999$	0.1118	0.34	5.66

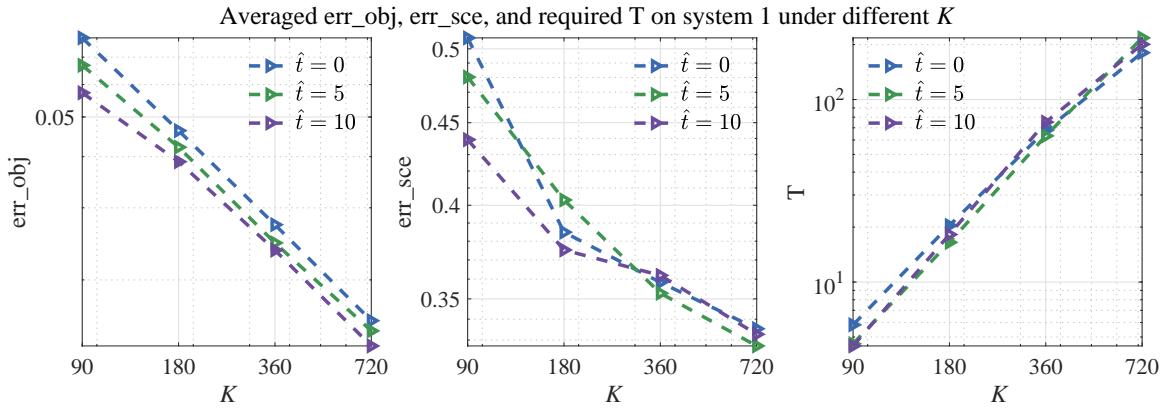


Figure 2: The achieved err_obj, err_sce, and required T averaged over 10 trials for each pair of (K, \hat{t}) given by the S-KLALM method with different values of \hat{t} on system 1 (equimass discretization). The blue, green, and purple dashed lines with right-pointing triangle markers are the results of the S-KLALM method with $\hat{t} = 0, 5, 10$, respectively. Left: err_obj. Middle: err_sce. Right: T.

Though lacking theoretical justifications, the sampling probabilities incorporating information of previous iterates are found to yield lower errors within less CPU time than randomly generated ones. Increasing the value of γ contributes to less CPU time for fulfilling the stopping criterion, yet worsening the accuracy once surpassing some threshold. We select $\gamma = 0.99$ in the S-ERALM and S-KLALM methods for a compromise between accuracy and efficiency in the subsequent experiments.

5.4.2 Comparisons among the S-KLALM methods with different \hat{t}

We conduct numerical comparisons on system 1 in Table 2 with equimass discretization and $K \in \{90, 180, 360, 720\}$. We call the S-KLALM method with $\hat{t} \in \{0, 5, 10\}$. For each pair of (K, \hat{t}) , 10 random trials are performed. The stopping parameters are $tol = 10^{-3} \times \sqrt{2}^{\log_2(K/90)}\text{2}$ and $t_{\max} = +\infty$. We depict the achieved err_obj, err_sce, and required T averaged over 10 trials for each pair of (K, \hat{t}) in Figure 2.

From Figure 2, we observe that the S-KLALM method with a positive \hat{t} does yield higher-quality solutions within comparable CPU time. But the advantage becomes less obvious as K grows large. Since large-scale problems are common in the applications of interest and a positive \hat{t} indicates

²⁾We increase tol since the size of $\mathcal{U}(\boldsymbol{\varrho}, \boldsymbol{\varrho})$ decreases as $\mathcal{O}(1/\sqrt{K})$; see Lemma 2.

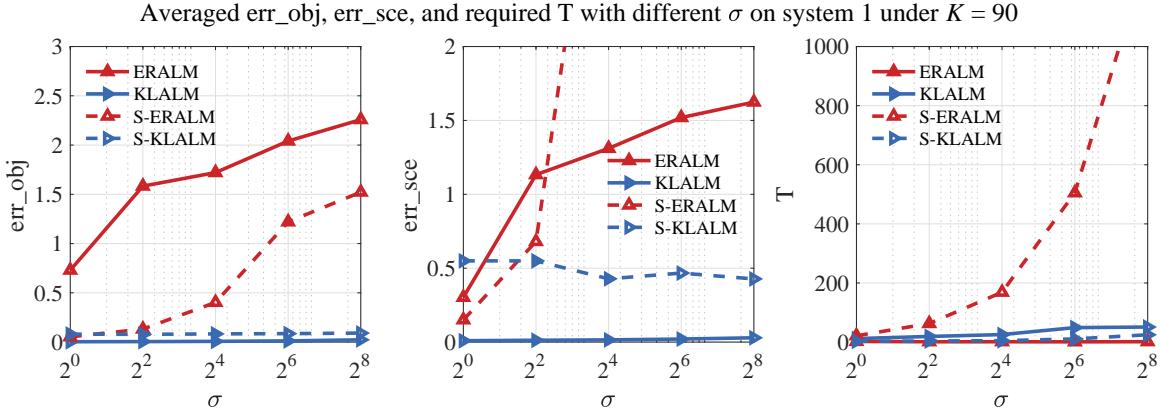


Figure 3: The achieved err_obj, err_sce, and required T averaged over 10 trials for each value of σ given by the ERALM, KLALM, S-ERALM, and S-KLALM methods on system 1 with $K = 90$ (equimass discretization). The red solid and dashed lines with triangle markers represent the results of the ERALM and S-ERALM methods, respectively. The blue solid and dashed lines with right-pointing triangle markers represent the results of the KLALM and S-KLALM methods, respectively. Left: err_obj. Middle: err_sce. Right: T.

the computations and storage of full matrices during the first iterations, we choose $\hat{t} = 0$ in the S-KLALM method for the ensuing numerical experiments.

5.4.3 Comparisons among the S-ERALM and S-KLALM methods

We first test the proposed four methods with σ taking its value in $\{2^0, 2^2, 2^4, 2^6, 2^8\}$ on system 1 with $K = 90$ and equimass discretization. For each value of σ , 10 random trials are performed. The stopping parameters are $tol = 5 \times 10^{-3}/\sqrt{\sigma}$ and $t_{max} = +\infty$. We record the achieved err_obj, err_sce and required T averaged over 10 random trials for each value of σ in Figure 3.

From Figure 3, we observe that (i) the objective errors of the ERALM and S-ERALM methods rise quickly as σ increases, which conforms to the theoretical results in section 4; (ii) the KLALM and S-KLALM methods yield high-quality solutions regardless of the choices of σ , demonstrating their robustness to the choices of proximal parameters. The robustness is practically desirable because a tiny σ , as is needed by the (S-)ERALM methods to achieve high accuracy, can result in numerical underflow.

In the above settings, we also notice that the S-ERALM method arrives at better solutions than those given by the S-KLALM method when a small σ is used. But we shall point out that the advantage no longer persists as K increases. We test the S-ERALM and S-KLALM methods with $\sigma = 1$ on system 1 with $K \in \{90, 180, 360, 720\}$ (equimass discretization). For each value of K , 10 random trials are performed. The stopping parameters are $tol = 10^{-3} \times \sqrt{2^{\log_2(K/90)}}$ and $t_{max} = +\infty$. We depict the achieved err_obj, err_sce, and required T averaged over 10 random trials for each value of K in Figure 4.

From Figure 4, we conclude that the advantage of the S-ERALM method over the S-KLALM method in terms of accuracy disappears as the problem size increases, along with fast growing computational time. This is in line with Table 1, i.e., the matrix entrywise sampling consumes quadratic complexity in each iteration of the S-ERALM method.

Finally, we compare the KLALM and S-KLALM methods. Aiming at a problem restricted on the sampled support, it is impractical for the S-KLALM method to outperform the KLALM method in accuracy. However, if the computational budget is limited, the S-KLALM method can yield relatively high-quality solutions within much less CPU time, particularly when the problem size goes large. We

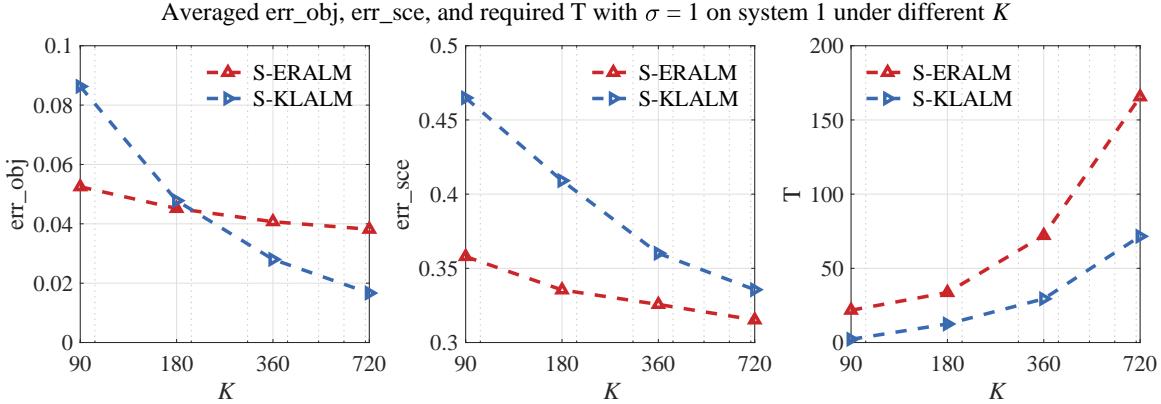


Figure 4: The achieved err_obj, err_sce, and required T averaged over 10 trials for each value of K given by the S-ERALM and S-KLALM methods with $\sigma = 1$ on system 1 (equimass discretization). The red dashed lines with triangle markers represent the results of the S-ERALM method. The blue dashed lines with right-pointing triangle markers represent the results of the S-KLALM method. Left: err_obj. Middle: err_sce. Right: T.

demonstrate this point by numerical comparisons on system 1. We employ equimass discretization with $K \in \{90, 180, 360, 720, 1440, 2880\}$. We call the KLALM and S-KLALM methods for each value of K with 10 random trials. The stopping parameters are $tol = 10^{-3} \times \sqrt{2}^{\log_2(K/90)}$ and $t_{\max} = +\infty$. We depict the convergence curves of err_obj along with the CPU time averaged over 10 trials for each value of K in Figure 5. The notation T_{inter} refers to the CPU time where the curves of two methods intersect for the last time.

From Figure 5, we observe that the KLALM method attains worse accuracy than the S-KLALM method until the CPU time touches T_{inter} . Moreover, as we increase the value of K , T_{inter} grows at a cubic rate; see Figure 6 for an illustration of cubic polynomial fitting. These results imply that (i) if the problem size is small (e.g., of order 10^2), the KLALM method achieves high accuracy within acceptable CPU time; (ii) if the problem size is relatively large (e.g., of order 10^3 or higher), as is usually the case in the applications of interest, and the computational budget is limited, the S-KLALM method is more preferable from the practical perspective.

Based on the above numerical findings, we choose the KLALM and S-KLALM methods for the comparisons with the PALM method as well as the simulations of the 2D and 3D systems in section 5.6.

5.4.4 Comparisons among the PALM, KLALM, and S-KLALM methods

We perform numerical comparisons on the 1D systems in Table 2 with equimass discretization. The implementation of the PALM method follows [42], except that we tune the proximal parameters adaptively as in equation (5.5) for fair comparisons. For systems 1 and 2, we consider $K \in \{90, 180, 360, 720\}$. For each value of K , 10 random trials are performed. The stopping parameters are $tol = 10^{-3} \times \sqrt{2}^{\log_2(K/90)}$ and $t_{\max} = +\infty$. For systems 3 and 4, we consider $K \in \{140, 280, 560, 1120\}$. For each value of K , 10 random trials are performed. The stopping parameters are $tol = 10^{-3} \times \sqrt{2}^{\log_2(K/140)}$ and $t_{\max} = +\infty$. The achieved err_obj, err_sce, and required T averaged over 10 random trials given by the three methods are gathered in Figures 7 and 8.

Under the same settings of proximal and stopping parameters, the PALM method suffers from premature convergence, while the KLALM method yields much smaller objective errors across the 1D systems. The multiplicative expression (3.17) further enables the use of randomized matrix

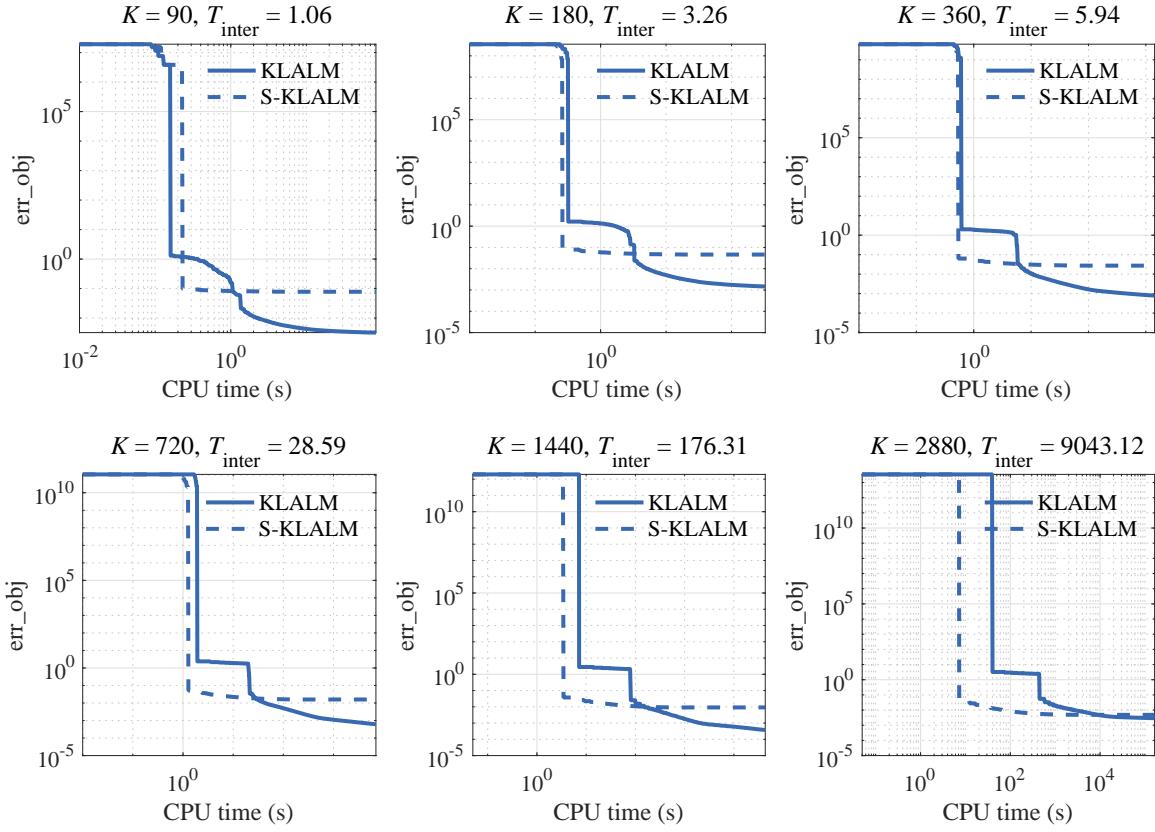


Figure 5: The convergence curves of err_obj along with the CPU time averaged over 10 trials for each value of K given by the KLALM and S-KLALM methods on system 1 (equimass discretization). The blue solid and dashed lines stand for the results of the KLALM and S-KLALM methods, respectively. The notation T_{inter} refers to the CPU time where the curves of two methods intersect for the last time.

sparsification, leading to the S-KLALM method with better scalability.

5.5 A cascadic multigrid optimization framework

Before the simulations of 2D and 3D systems, we introduce a cascadic multigrid (CMG) optimization framework for problem (5.4). The CMG optimization framework has demonstrated its power in accelerating the numerical solution of large-scale scientific problems, especially those with governing partial differential equations; see, e.g., [15, 24, 58, 79]. In general, the framework begins with a coarse mesh, with which a small-scale problem is associated. An accurate solver is then called to solve the small-scale problem and yields a solution. Later, it basically repeats the following three steps: (i) refine the previous coarse mesh to a finer one; (ii) prolongate the previous solution and construct an initial point over the current mesh; (iii) start a local solver from the initial point and obtain a solution over the current mesh. With carefully designed prolongation operator and local solver, the framework can fully utilize the solution information in the coarser meshes and considerably accelerate the numerical solution of the large-scale problems over the finer meshes.

In our context, we call accurate solvers, such as the KLALM method, to tackle the problems over the initial coarse meshes. After uniform or adaptive mesh refinements, we prolongate the previous

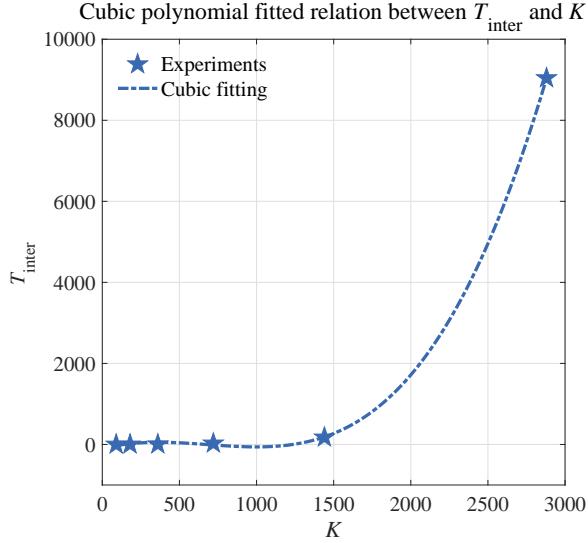


Figure 6: Cubic polynomial fitted relation between T_{inter} and K on system 1 (equimass discretization). The fitted polynomial is $9.09 \times 10^{-7}K^3 - 1.86 \times 10^{-3}K^2 + 1.00K - 106.38$. The blue pentagons are the obtained T_{inter} in experiments and the blue dashdotted line is the fitted relation.

solution (say, $(Y_2^{(\ell-1,\star)}, \dots, Y_{N_e}^{(\ell-1,\star)})$) to the current mesh (say, $\{e_k^{(\ell)}\}_{k=1}^{K^{(\ell)}}$) as follows:

$$y_{i,kl}^{(\ell,0)} := \frac{1}{K_{k'l'}^{(\ell)}} y_{i,k'l'}^{(\ell-1,\star)}, \quad k, l = 1, \dots, K^{(\ell)}, \quad i = 2, \dots, N_e, \quad (5.6)$$

where $k', l' \in \{1, \dots, K^{(\ell-1)}\}$ are such that $e_k^{(\ell)} \subseteq e_{k'}^{(\ell-1)}$, $e_l^{(\ell)} \subseteq e_{l'}^{(\ell-1)}$, and

$$K_{k'l'}^{(\ell)} := \left| \left\{ (k, l) : e_k^{(\ell)} \subseteq e_{k'}^{(\ell-1)}, e_l^{(\ell)} \subseteq e_{l'}^{(\ell-1)}, k, l \in \{1, \dots, K^{(\ell)}\} \right\} \right|.$$

The design of the prolongation operator follows that $\rho\gamma_i$ can be understood as a joint probability density between the positions of the first and i th electrons ($i = 2, \dots, N_e$). Finally, cheap local solvers, such as the S-KLALM method, start from $(Y_2^{(\ell,0)}, \dots, Y_{N_e}^{(\ell,0)})$ and solve the problem over the current mesh. Consequently, apart from the previously mentioned warm starts, the prolongation operator also enables “warm samplings”; please refer to the importance sampling probability (3.8). We summarize our CMG optimization framework in Framework 5.

To demonstrate the utility of Framework 5, we include numerical experiments on the 1D systems. We call the CMG optimization method, which takes respectively the KLALM and S-KLALM methods as the accurate and cheap local solvers, the S-KLALM-CMG method. We compare the numerical performances of the KLALM, S-KLALM, and S-KLALM-CMG methods. For systems 1 and 2, the KLALM and S-KLALM methods directly solve the problems with $K = 720$ and equimass discretization starting from 10 random trials. The stopping parameters are $\text{tol} = 2\sqrt{2} \times 10^{-3}$ and $t_{\max} = +\infty$. The S-KLALM-CMG method starts 10 random trials from $K^{(0)} = 90$ with equimass discretization and reaches the desired the mesh after equimass refinements for three times. The stopping parameters at level $\ell \in \{0, 1, 2, 3\}$ are $\text{tol} = 10^{-3} \times \sqrt{2}^{\log_2(K^{(\ell)}/K^{(0)})}$ and $t_{\max} = +\infty$. The settings for the simulations of systems 3 and 4 are analogous, except that the problem size handled by the KLALM and S-KLALM methods is $K = 1120$ and the initial problem size handled by the S-KLALM-CMG method is $K^{(0)} = 140$. We report the achieved err_obj , err_sce , and required T averaged over 10 random trials given by the three methods on the 1D systems in Table 4. The S-KLALM-CMG method is found to yield relatively high-quality solutions within the least CPU time.

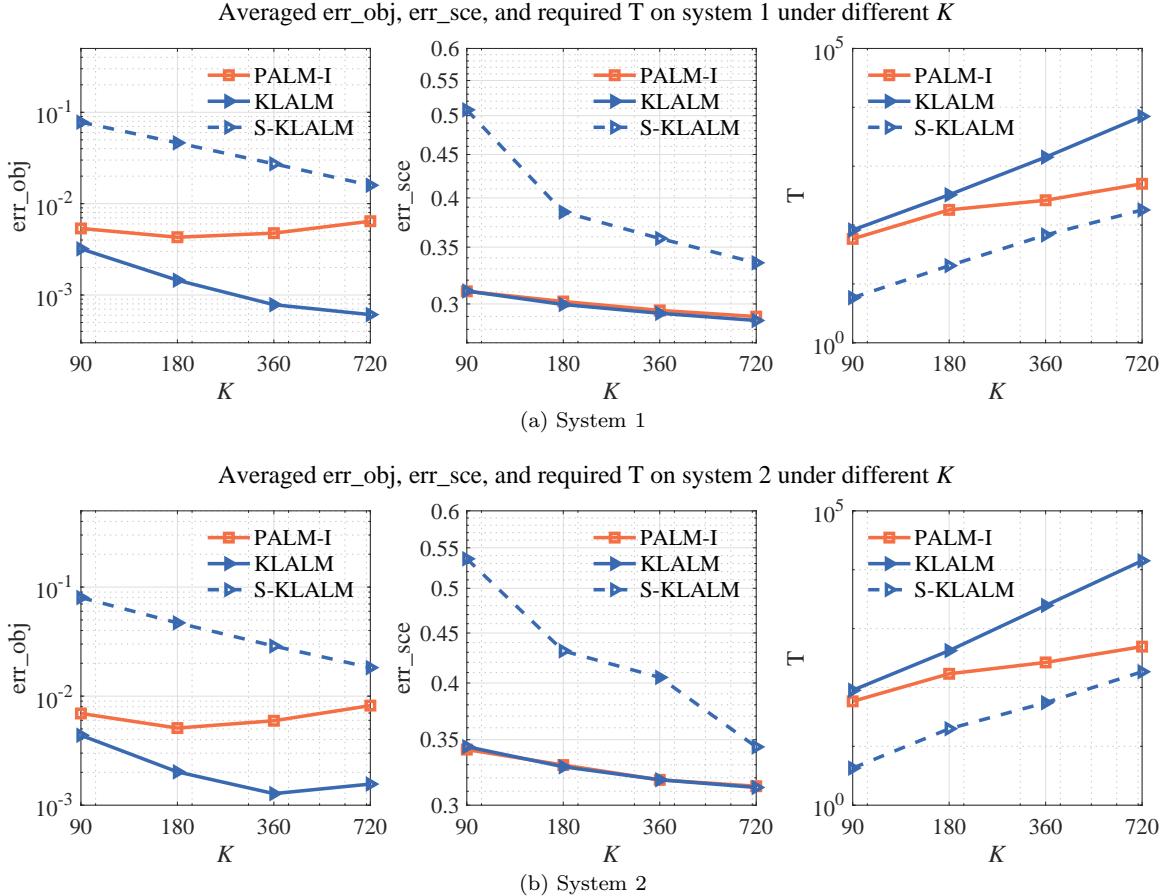


Figure 7: The achieved err_obj , err_sce , and required T averaged over 10 trials for each value of K given by the PALM, KLALM, and S-KLALM methods on the three-electron 1D systems (equimass discretization). The orange solid lines with square markers are the results of the PALM method. The blue solid and dashed lines with right-pointing triangle markers are the results of the KLALM and S-KLALM methods, respectively. From left to right: err_obj , err_sce , and T . (a) System 1. (b) System 2.

5.6 Simulations on 2D and 3D systems

We use the S-KLALM-CMG method for simulating the 2D and 3D systems in Table 2. The initial step employs equisize discretization and the latter ones refine the meshes uniformly. For the 2D systems, the initial mesh size is $K^{(0)} = 900$ before truncation. After grid refinements for three times, we get $K^{(3)} = 57600$ before truncation. For the 3D system 7, we set $K^{(0)} = 1728$ before truncation and perform grid refinements twice to arrive at $K^{(2)} = 110592$ before truncation. For the 3D system 8, we set $K^{(0)} = 1000$ before truncation and perform grid refinements twice to arrive at $K^{(2)} = 64000$ before truncation. The stopping parameters are $t_{\max} = 10^4$ and

$$tol = \begin{cases} 5 \times 10^{-3}, & \ell = 0, \\ 10^{-2} \times (\sqrt{2d})^{\log_2(K/K^{(0)})}, & \ell > 0. \end{cases}$$

Note that for these systems, no explicit constructions of the optimal solutions are available. Hence, we monitor instead the evolution of objective values and the approximate SCE potentials. Moreover, following [42], we approximate the OT maps between electron positions in the so-called Monge ansatz

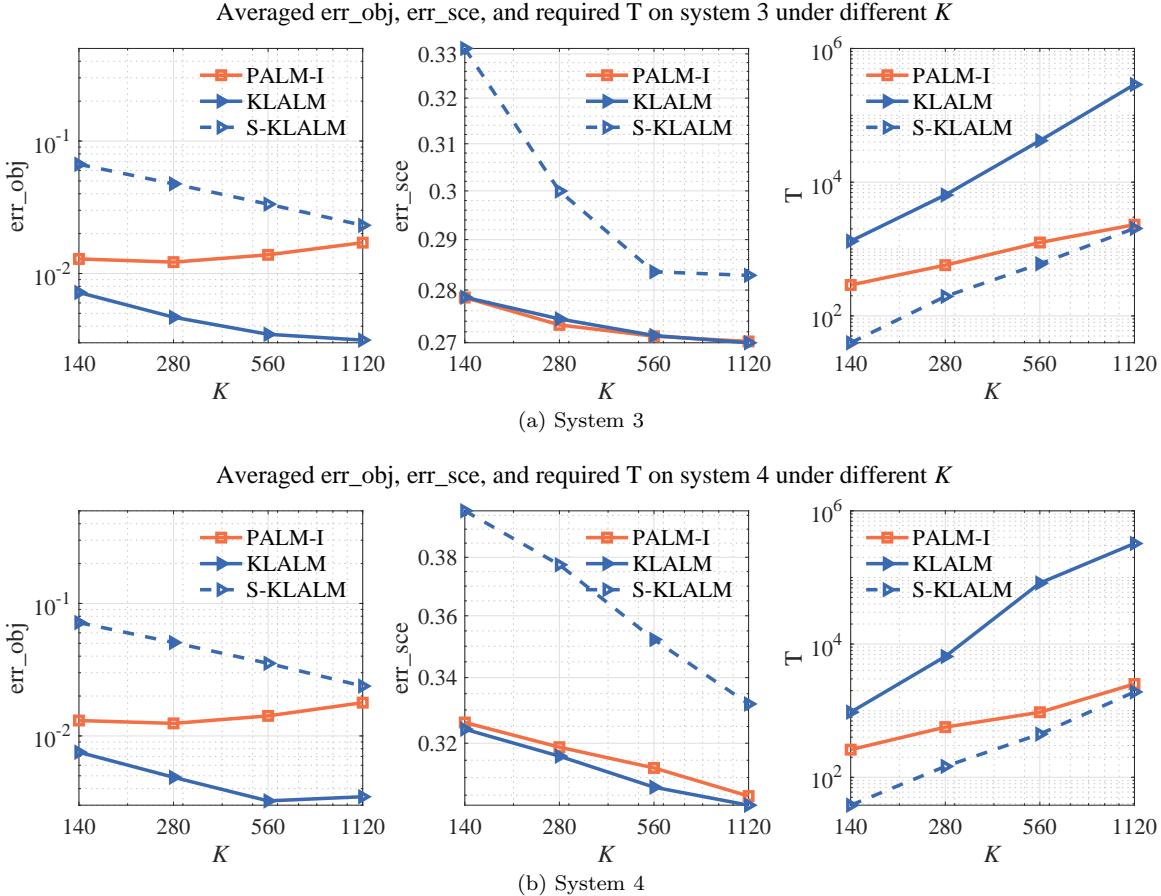


Figure 8: The achieved err_obj , err_sce , and required T averaged over 10 trials for each value of K given by the PALM, KLALM, and S-KLALM methods on the seven-electron 1D systems (equimass discretization). The orange solid lines with square markers are the results of the PALM method. The blue solid and dashed lines with right-pointing triangle markers are the results of the KLALM and S-KLALM methods, respectively. From left to right: err_obj , err_sce , and T . (a) System 3. (b) System 4.

[69–71], which are of particular physical interest, as

$$\mathcal{T}_i(\mathbf{d}_j) := \sum_{1 \leq k \leq K} y_{i,j,k} \mathbf{d}_k / \varrho_j, \quad j = 1, \dots, K, \quad i = 2, \dots, N_e,$$

where $\mathcal{T}_i : \mathbb{R}^d \rightarrow \mathbb{R}^d$ represents the transport map between the positions of the first and i th electrons. We collect the results in Table 5 as well as Figures 9 and 10, where $K_{\text{trunc}} \in \mathbb{N}$ refers to the dimension of the truncated $\boldsymbol{\varrho}$. We illustrate the approximate OT maps $\{\mathcal{T}_i\}_{i=2}^{N_e}$ through their images of the barycenters of the finite elements within some given subregion $\omega \subseteq \Omega$.

The results in Table 5 and Figure 9 showcase the convergence of the CMG optimization framework, also manifesting the effectiveness of the prolongation operator therein. The approximate OT maps in Figure 10 are in line with physical intuitions. In particular, for the three-Gaussian 2D system 5, Figure 10 (a) implies that if the first electron is around one Gaussian center, the other two electrons will go near the other two Gaussian centers, respectively. For the four-Gaussian 2D system 6, Figure 10 (b) shows that if one electron revolves around but stays away from the top Gaussian center,

Framework 5 The CMG optimization framework for problem (5.4).

Input: Discretization and refinement oracles, accurate and cheap local solvers, initial number of finite elements $K^{(0)} \in \mathbb{N}$.

- 1: Set $\ell := 0$.
- 2: **while** certain conditions are not satisfied **do**
- 3: **if** $\ell = 0$ **then**
- 4: **Discretization:** discretize the MMOT into problem (5.4) with $K^{(0)}$ finite elements $\{e_k^{(0)}\}_{k=1}^{K^{(0)}} \subseteq \mathbb{R}^d$.
- 5: Construct a random initial point $(Y_2^{(0,0)}, \dots, Y_{N_e}^{(0,0)}) \in (\mathbb{R}^{K^{(\ell)} \times K^{(\ell)}})^{N_e-1}$.
- 6: **Local solution:** start the accurate local solver from $(Y_2^{(0,0)}, \dots, Y_{N_e}^{(0,0)})$ for problem (5.4) at level 0 and obtain $(Y_2^{(0,*)}, \dots, Y_{N_e}^{(0,*)})$.
- 7: **else**
- 8: **Grid refinement:** refine $\{e_k^{(\ell-1)}\}_{k=1}^{K^{(\ell-1)}}$ to $\{e_k^{(\ell)}\}_{k=1}^{K^{(\ell)}} \subseteq \mathbb{R}^d$ with $K^{(\ell)} \in \mathbb{N}$.
- 9: **Prolongation:** prolongate $(Y_2^{(\ell-1,*)}, \dots, Y_{N_e}^{(\ell-1,*)})$ to the current mesh through equation (5.6) and obtain $(Y_2^{(\ell,0)}, \dots, Y_{N_e}^{(\ell,0)}) \in (\mathbb{R}^{K^{(\ell)} \times K^{(\ell)}})^{N_e-1}$.
- 10: **Local solution:** start the cheap local solver from $(Y_2^{(\ell,0)}, \dots, Y_{N_e}^{(\ell,0)})$ for problem (5.4) at level ℓ and obtain $(Y_2^{(\ell,*)}, \dots, Y_{N_e}^{(\ell,*)})$.
- 11: **end if**
- 12: Set $\ell := \ell + 1$.
- 13: **end while**

Output: Approximate solution $(Y_2^{(\ell-1,*)}, \dots, Y_{N_e}^{(\ell-1,*)})$.

Table 4: The achieved err_obj, err_sce, and required T averaged over 10 trials given by the KLALM, S-KLALM, and S-KLALM-CMG methods on the 1D systems (equimass discretization).

Algorithms	System 1 ($K = 720$)			System 2 ($K = 720$)		
	err_obj	err_sce	T	err_obj	err_sce	T
KLALM	0.0006	0.29	7014.57	0.0016	0.31	14190.02
S-KLALM	0.0159	0.34	180.51	0.0183	0.34	184.96
S-KLALM-CMG	0.0032	0.34	96.32	0.0044	0.38	109.39
Algorithms	System 3 ($K = 1120$)			System 4 ($K = 1120$)		
	err_obj	err_sce	T	err_obj	err_sce	T
KLALM	0.0031	0.27	288688.75	0.0035	0.30	323734.12
S-KLALM	0.0231	0.28	2034.49	0.0238	0.33	1910.36
S-KLALM-CMG	0.0074	0.32	1204.31	0.0079	0.36	1252.50

the other three electrons will stay near the three Gaussian centers, respectively, with one of which surrounded by the first one but keeping away from each other. For the three-Gaussian 3D system 7, the results depicted in Figure 10 (c) are analogous to those in Figure 10 (a). For the two-Gaussian 3D system 8, Figure 10 (d) indicates that if one electron lies around the Gaussian center $[1, 0, 0]^\top$, the other three will be located around the other center, with their positions trisecting a sphere. In Figure 10, the positions of the electrons determined by the approximate OT maps $\{\mathcal{T}_i\}_{i=2}^{N_e}$ conform to the repulsive law and are improved along our optimization process. Notably in Figure 10, we provide

Table 5: The objective values calculated along the iteration of the S-KLALM-CMG method when solving the 2D and 3D systems. The notation $K_{\text{trunc}} \in \mathbb{N}$ is the dimension of the truncated ρ .

Step	System 5			System 6		
	K	K_{trunc}	obj	K	K_{trunc}	obj
0	900	424	1.1339	900	408	3.0690
1	3600	1622	1.1337	3600	1534	3.0690
2	14400	6410	1.1335	14400	6068	3.0677
3	57600	25562	1.1334	57600	24176	3.0667

Step	System 7			System 8		
	K	K_{trunc}	obj	K	K_{trunc}	obj
0	1728	780	1.0202	1000	720	4.6193
1	13824	5628	1.0209	8000	5272	4.6716
2	110592	42936	1.0209	64000	40764	4.6833

the first visualization of the approximate OT maps between electron positions in 3D contexts.

Remark 5. There have been extensive works dedicated to the numerical solutions of the MMOT (5.1). The authors of [62] consider the Kantorovich dual of the MMOT, whose number of inequality constraints increases exponentially with N_e . The authors of [11] investigate the entropy regularized MMOT, where curse of dimensionality still resides. The authors of [48] derive a convex semidefinite programming relaxation for the discretized N -representability form of the MMOT, where the problem size is independent from the value of N_e but the gap induced by the convex relaxation remains elusive. The authors of [5] exploit a moment-constrained relaxation of the MMOT, which admits sparse optimal solutions but entails careful selections of parameters and test functions to achieve satisfactory approximations. In both [23] and our work, the reformulation of the MMOT under the Monge-like ansatz is adopted, where the problem size increases linearly with respect to N_e . However, at the moment, the ansatz is provably true only for special (e.g., two-electron/one-dimensional/special radially symmetric) cases [37].

The considerable distinctions in models make a fair numerical comparison difficult and go beyond our scope. Nevertheless, we shall note that the model under the Monge-like ansatz explicitly characterizes the electron-electron couplings. Therefore, in contrast with others, it enables the evaluation of solution qualities through approximate OT maps between electron positions; see Figure 10. Additionally, in comparison with [23] where the authors only consider two-electron systems (i.e., H_2 molecule) with discretization size $K \sim 3000$, we solve larger-scale problems ($K \sim 10^5$) and simulate systems with more electrons ($N_e = 3 \sim 7$).

5.7 Scalability tests

Finally, we conduct scalability tests for the KLALM and S-KLALM methods with respect to K and N_e . The system under simulation shares the same normalized single-particle density ρ with system 1 in Table 2, yet with varying N_e . We employ equimass discretization.

For the tests with respect to K , we fix $N_e = 3$ and consider $K \in \{90, 180, 360, 720\}$. For each value of K , the two methods are called with 10 random trials. The proximal parameter is set to $\mu_i^{(t)} \equiv 0.05$. The stopping parameters are $\text{tol} = 10^{-3}$ and $t_{\max} = +\infty$. The achieved err_obj , err_sce , and required T averaged over 10 trials are gathered in Figure 11 (a). In light of Table 1, we fit the linear relation between $\log(T)$ and $\log(K)$ with linear least squares and obtain the following:

$$\text{KLALM: } \log(T) \approx 2.59 \log(K) - 7.22;$$

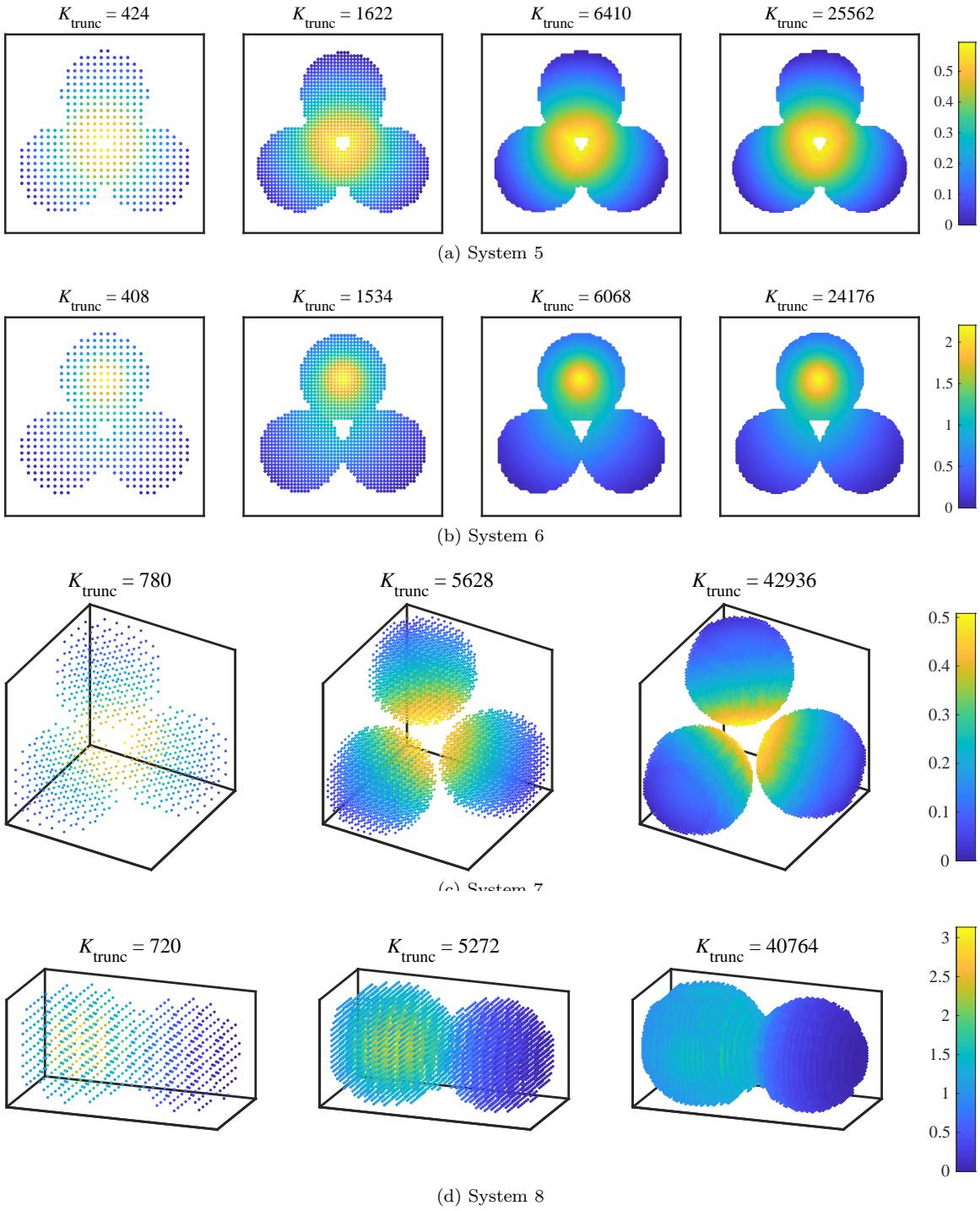


Figure 9: The evolution of the approximate SCE potentials for the 2D and 3D systems given by the S-KLALM-CMG method. (a) System 5. (b) System 6. (c) System 7. (d) System 8.

$$\text{S-KLALM: } \log(T) \approx 2.21 \log(K) - 8.31.$$

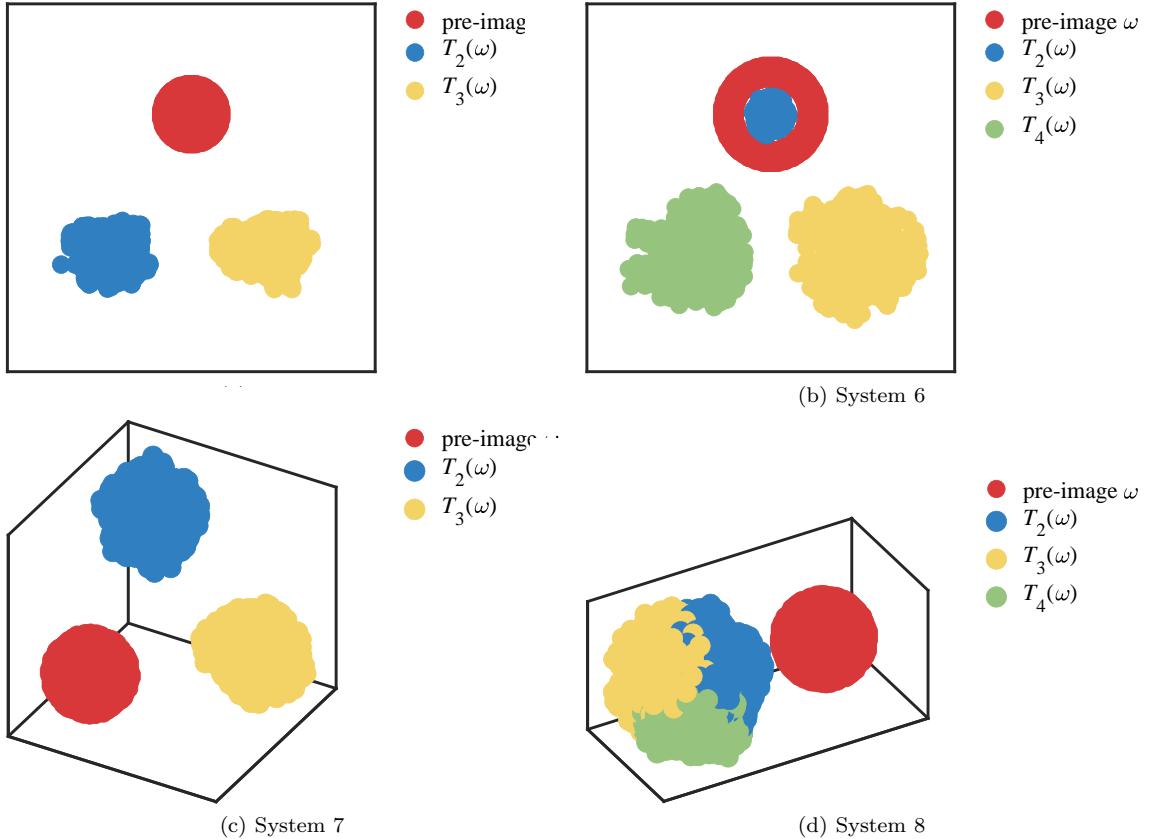


Figure 10: The approximate OT maps between electron positions for the 2D and 3D systems given by the S-KLALM-CMG method. The pre-image ω in red stands for the positions of the first electron, while the areas in other colors are the associated positions of the other electrons, respectively, determined by $\{\mathcal{T}_i\}_{i=2}^{N_e}$. (a) System 5. (b) System 6. (c) System 7. (d) System 8.

For the tests with respect to N_e , we fix $K = 144$ and vary N_e in $\{3, 6, 12, 24, 48\}$. For each value of N_e , the two methods are called with 10 random trials. We fix the proximal parameter $\mu_i^{(t)} \equiv 20/\log(K)$ because $\|\mathbf{v}_i^{(t)}\|_\infty$ grows with N_e . The stopping parameters are $\text{tol} = 5 \times 10^{-3}$ and $t_{\max} = +\infty$. The achieved err_obj , err_sce , and required T averaged over 10 trials are collected in Figure 11 (b). Likewise, we obtain the following relations:

$$\begin{aligned} \text{KLALM: } \log(T) &\approx 1.31 \log(N_e) + 0.32; \\ \text{S-KLALM: } \log(T) &\approx 1.06 \log(N_e) + 0.49. \end{aligned}$$

The obtained scalings of the KLALM and S-KLALM methods with respect to N_e are in rough accordance with the computational complexities summarized in Table 1, where $\tau = 0.5$. The marked deviation in the scalings of the KLALM and S-KLALM methods with respect to K can be ascribed to the implementation in software as well as the difference in the numbers of iterations for each value of K .

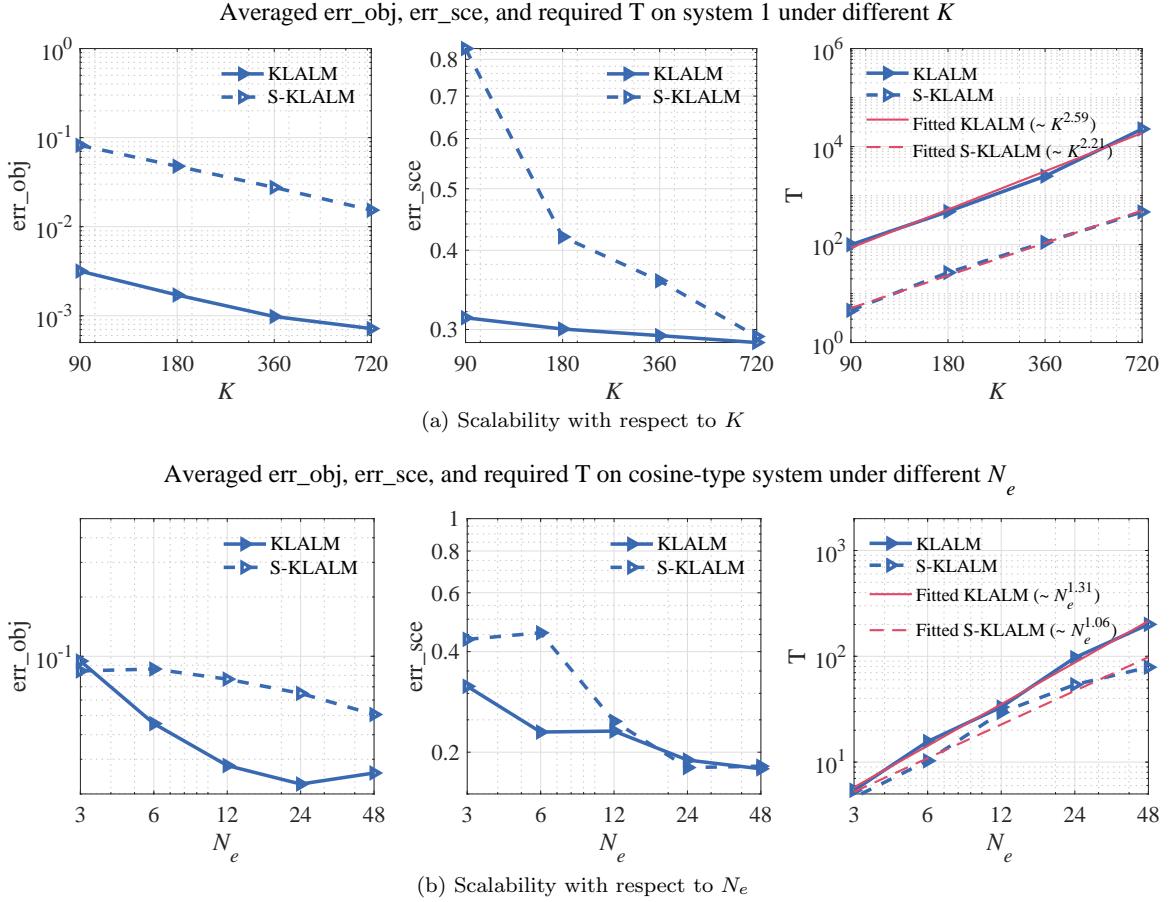


Figure 11: The achieved err_obj, err_sce, and required T averaged over 10 trials for each value of K and N_e given by the KLALM and S-KLALM methods on the cosine-type system (equimass discretization). The blue solid and dashed lines represent the results of the KLALM and S-KLALM methods, respectively. The pink solid and dashed lines denote the fitted relations between T and K (or N_e) when using the KLALM and S-KLALM methods, respectively. Left: err_obj. Middle: err_sce. Right: T. (a) Scalability with respect to K . For the KLALM method, $T \sim K^{2.6}$. For the S-KLALM method, $T \sim K^{2.2}$. (b) Scalability with respect to N_e . For the KLALM method, $T \sim N_e^{1.3}$. For the S-KLALM method, $T \sim N_e^{1.1}$.

6 Conclusions

Leveraging the tools from OT and the importance sampling technique, we introduce novel BCD-type methods, the (S-)ERALM and (S-)KLALM methods, for the multi-block optimization problems over the transport polytopes. These methods enjoy highly scalable schemes for the subproblems and save considerable expenditure for the calculations and storage in large-scale contexts. With the theory of randomized matrix sparsification, we establish for the ERALM and S-ERALM methods that the average stationarity violations tend to 0 as the problem size increases to $+\infty$ (with probability going to 1). To the best of our knowledge, our work is the first attempt in applying the matrix entrywise sampling technique to multi-block nonconvex settings with theoretical guarantees. In the simulations of strongly correlated electron systems, though lacking convergence analysis, the (S-)KLALM methods exhibit desirable robustness to the choices of proximal parameters; compared

with the KLALM method, the S-KLALM method trades accuracy for efficiency. Both the KLALM and S-KLALM methods are further brought together under a cascadic multigrid optimization framework, pursuing a decent balance for large-scale simulations. The numerical results conform to both theoretical predictions and physical intuitions. The better scalability of the sampling-based methods allows the first visualization of the approximate OT maps for the 3D systems.

For future work, we are eager to provide some insights into the theoretical advantages of importance sampling, as evidenced by the numerical results in section 5.4.1. It is also of special interest in optimization to investigate the convergence properties for the proposed KL divergence-based methods by exploring routes that do not rest on the local Lipschitz smoothness of the proximal terms. Computationally, further acceleration can be gained via other techniques, such as support identification [52]. In terms of quantum physics application, it is necessary to study the solution landscape for problem (5.4). Our numerical findings suggest that the objective errors at stationary points decrease as the discretization becomes finer.

A Convergence of the ERALM method

This part includes the proofs of the results concerning the ERALM method. We first adapt a lemma from [46], characterizing the objective error induced by entropy regularization on an arbitrary OT problem.

Lemma 1. *Let $W \in \mathbb{R}^{m \times n}$, $\mathbf{a} \in \mathbb{R}^m$, $\mathbf{b} \in \mathbb{R}^n$, $\lambda > 0$, and $\mathcal{I} \subseteq \{(j, k) : j = 1, \dots, m, k = 1, \dots, n\}$. Suppose that T' , $T'' \in \mathbb{R}^{m \times n}$ are the optimal solutions of*

$$\min_T \langle W, T \rangle, \text{ s.t. } T \in \mathcal{U}(\mathbf{a}, \mathbf{b}), T_{\mathcal{I}^c} = 0$$

and

$$\min_T \langle W, T \rangle + \lambda h(T), \text{ s.t. } T \in \mathcal{U}(\mathbf{a}, \mathbf{b}), T_{\mathcal{I}^c} = 0,$$

respectively. Then $0 \leq \langle W, T'' - T' \rangle \leq -\lambda h(\mathbf{a}\mathbf{b}^\top)$.

Proof. See the proof of [46, Lemma 1], which leverages the fact that $0 \geq h(T) \geq h(\mathbf{a}\mathbf{b}^\top)$ for all $T \in \mathcal{U}(\mathbf{a}, \mathbf{b})$. \square

Next, we provide an upper bound for the size of $\mathcal{U}(\mathbf{a}_i, \mathbf{b}_i)$ ($i \in \{1, \dots, N\}$).

Lemma 2. *For $i \in \{1, \dots, N\}$, there holds*

$$\|T - T'\| \leq 2d_i, \text{ for all } T, T' \in \mathcal{U}(\mathbf{a}_i, \mathbf{b}_i).$$

Proof. It suffices to bound the norm of an arbitrary member in $\mathcal{U}(\mathbf{a}_i, \mathbf{b}_i)$. Note that, for any $T \in \mathcal{U}(\mathbf{a}_i, \mathbf{b}_i)$,

$$\begin{aligned} \|T\| &= \sqrt{\sum_{j=1}^{m_i} \sum_{t=1}^{n_i} t_{jk}^2} \leq \sqrt{\sum_{j=1}^{m_i} \left(\sum_{t=1}^{n_i} t_{jk} \right)^2} = \|T\mathbf{1}_{n_i}\| = \|\mathbf{a}_i\| \\ &= \sqrt{\sum_{j=1}^{m_i} a_{i,j}^2} \leq \sqrt{m_i \|\mathbf{a}_i\|_\infty^2} = \sqrt{m_i} \|\mathbf{a}_i\|_\infty. \end{aligned}$$

Similarly, we have $\|T\| \leq \sqrt{n_i} \|\mathbf{b}_i\|_\infty$. Then by triangle inequality and the definition of d_i in Theorem 1, we complete the proof. \square

Based upon Assumption 1 and Lemma 2, the residual function R_i in equation (4.1) enjoys a Lipschitz-like property, which will be useful for the convergence proof.

Lemma 3. Let $X', X'' \in \bigtimes_{i=1}^N \mathcal{U}(\mathbf{a}_i, \mathbf{b}_i)$. Suppose that Assumption 1 holds and $X'_i = X''_i$ for some $i \in \{1, \dots, N\}$. Then $|R_i(X') - R_i(X'')| \leq 2d_i L \|X' - X''\|$.

Proof. Let \bar{X}'_i and \bar{X}''_i be the optimal solutions of

$$\min_{X_i \in \mathcal{U}(\mathbf{a}_i, \mathbf{b}_i)} \langle \nabla_i f(X'), X_i \rangle \quad \text{and} \quad \min_{X_i \in \mathcal{U}(\mathbf{a}_i, \mathbf{b}_i)} \langle \nabla_i f(X''), X_i \rangle,$$

respectively. Simple algebraic calculations yield

$$\begin{aligned} R_i(X') &= \langle \nabla_i f(X'), X'_i - \bar{X}'_i \rangle \quad (\text{equation (4.1)}) \\ &= \langle \nabla_i f(X''), X'_i - \bar{X}'_i \rangle + \langle \nabla_i f(X') - \nabla_i f(X''), X'_i - \bar{X}'_i \rangle \\ &\leq \langle \nabla_i f(X''), X'_i - \bar{X}'_i \rangle + 2d_i L \|X' - X''\| \quad (\text{Assumption 1 and Lemma 2}) \\ &= \langle \nabla_i f(X''), X''_i - \bar{X}''_i \rangle + 2d_i L \|X' - X''\| \quad (\text{since } X'_i = X''_i) \\ &\leq \langle \nabla_i f(X''), X''_i - \bar{X}''_i \rangle + 2d_i L \|X' - X''\| \quad (\text{the definition of } \bar{X}''_i) \\ &= R_i(X'') + 2d_i L \|X' - X''\|. \quad (\text{equation (4.1)}) \end{aligned}$$

Similarly, one can show $R_i(X'') \leq R_i(X') + 2d_i L \|X' - X''\|$. These two together give the desired result. \square

With the above tools in place, we are ready to establish an upper bound for the average residual over the iterate sequence generated by the ERALM method.

Proof of Theorem 1. For any $i \in \{1, \dots, N\}$ and $t \geq 0$,

$$\begin{aligned} f(X_{\leq i}^{(t+1)}, X_{\geq i}^{(t)}) &\leq f(X_{<i}^{(t+1)}, X_{\geq i}^{(t)}) + \left\langle C_i^{(t)}, X_i^{(t+1)} - X_i^{(t)} \right\rangle + \frac{L}{2} \|X_i^{(t+1)} - X_i^{(t)}\|^2 \\ &= f(X_{<i}^{(t+1)}, X_{\geq i}^{(t)}) + \alpha \left\langle C_i^{(t)}, \tilde{X}_i^{(t+1)} - X_i^{(t)} \right\rangle + \frac{\alpha^2 L}{2} \|\tilde{X}_i^{(t+1)} - X_i^{(t)}\|^2 \\ &\leq f(X_{<i}^{(t+1)}, X_{\geq i}^{(t)}) + \alpha \left\langle C_i^{(t)}, \tilde{X}_i^{(t+1)} - X_i^{(t)} \right\rangle + 2(d_i \alpha)^2 L \\ &\leq f(X_{<i}^{(t+1)}, X_{\geq i}^{(t)}) - \alpha \lambda h(\mathbf{a}_i \mathbf{b}_i^\top) - \alpha R_i(X_{<i}^{(t+1)}, X_{\geq i}^{(t)}) + 2(d_i \alpha)^2 L, \end{aligned}$$

where the first inequality uses Assumption 1, the second one uses Lemma 2, the last one relies on Lemma 1 and the definition (4.1) of the residual function R_i . The above relation further gives

$$\alpha R_i(X_{<i}^{(t+1)}, X_{\geq i}^{(t)}) \leq f(X_{<i}^{(t+1)}, X_{\geq i}^{(t)}) - f(X_{\leq i}^{(t+1)}, X_{>i}^{(t)}) - \alpha \lambda h(\mathbf{a}_i \mathbf{b}_i^\top) + 2(d_i \alpha)^2 L. \quad (\text{A.1})$$

Note that

$$\begin{aligned} &\left| R_i(X_{<i}^{(t+1)}, X_{\geq i}^{(t)}) - R_i(X^{(t)}) \right|^2 \\ &\leq 4(d_i L)^2 \|X^{(t+1)} - X^{(t)}\|^2 = 4(d_i L)^2 \sum_{i=1}^N \|X_i^{(t+1)} - X_i^{(t)}\|^2 \\ &= 4(d_i L \alpha)^2 \sum_{i=1}^N \|\tilde{X}_i^{(t+1)} - X_i^{(t)}\|^2 \leq 16 \bar{d}^4 L^2 \alpha^2 N, \end{aligned}$$

where the first inequality follows from Lemma 3 and the last one is due to Lemma 2. Combining the above inequality and relation (A.1) yields

$$\alpha R_i(X^{(t)}) = \alpha \left[R_i(X_{<i}^{(t+1)}, X_{\geq i}^{(t)}) + \left(R_i(X^{(t)}) - R_i(X_{<i}^{(t+1)}, X_{\geq i}^{(t)}) \right) \right]$$

$$\begin{aligned}
&\leq f(X_{<i}^{(t+1)}, X_{\geq i}^{(t)}) - f(X_{\leq i}^{(t+1)}, X_{>i}^{(t)}) - \alpha \lambda h(\mathbf{a}_i \mathbf{b}_i^\top) + 4\bar{d}^2 \alpha^2 L \sqrt{N} + 2(d_i \alpha)^2 L \\
&\leq f(X_{<i}^{(t+1)}, X_{\geq i}^{(t)}) - f(X_{\leq i}^{(t+1)}, X_{>i}^{(t)}) + \alpha \lambda \bar{h} + 2\bar{d}^2 \alpha^2 L (2\sqrt{N} + 1).
\end{aligned}$$

Summing the above inequality over i from 1 to N and dividing both sides by α , one obtains

$$R(X^{(t)}) \leq \frac{f(X^{(t)}) - f(X^{(t+1)})}{\alpha} + N\lambda\bar{h} + 2\bar{d}^2 LN(2\sqrt{N} + 1)\alpha.$$

Summing the above relation over t from 0 to $t_{\max} - 1$ and dividing the both sides by t_{\max} , we complete the proof after noting $f(X^{(t_{\max})}) \geq \underline{f}$, the definition of α in equation (4.2), and $2N(2\sqrt{N} + 1) < (2N + 1)^2$. Incidentally, the lower bound for t_{\max} in equation (4.2) ensures $\alpha \leq 1$. \square

Proof of Corollary 1. By Assumption 2 (i)-(iii), it is not hard to derive that

$$\begin{aligned}
\bar{d} &= \max_{i=1}^N \min\{\sqrt{m_i} \|\mathbf{a}_i\|_\infty, \sqrt{n_i} \|\mathbf{b}_i\|_\infty\} = \Theta\left(\max_{i=1}^N \min\left\{\frac{1}{\sqrt{m_i}}, \frac{1}{\sqrt{n_i}}\right\}\right), \\
\bar{h} &= \max_{i=1}^N \sum_{j,k} a_{i,j} b_{i,k} (1 - \log a_{i,j} b_{i,k}) = \Theta\left(\sum_{i=1}^N \log m_i n_i\right).
\end{aligned}$$

Based on the above bounds, the first term on the right-hand side of inequality (4.3) goes to 0 if $t_{\max} = \Omega(\sum_{i=1}^N (m_i + n_i)^\eta)$ with $\eta > \theta$ and M independent from $\{m_i\}_{i=1}^N$ and $\{n_i\}_{i=1}^N$, and the second one goes to 0 if $\lambda = o(1/\sum_{i=1}^N \log m_i n_i)$. The proof is complete. \square

B Convergence of the S-ERALM method

We define the following auxiliary sequences:

$$\begin{aligned}
\bar{X}_i^{(t+1)} &\in \arg \min_{X_i} \langle C_i^{(t)}, X_i \rangle, \text{ s.t. } X_i \in \mathcal{U}(\mathbf{a}_i, \mathbf{b}_i), i = 1, \dots, N, t \geq 0, \\
\check{X}_i^{(t+1)} &= \arg \min_{X_i} \langle C_i^{(t)}, X_i \rangle + \hat{\lambda} h(X_i), \text{ s.t. } X_i \in \mathcal{U}(\mathbf{a}_i, \mathbf{b}_i), i = 1, \dots, N, t \geq 0.
\end{aligned}$$

Proof of Theorem 2. For any $i \in \{1, \dots, N\}$ and $t \geq 0$,

$$\begin{aligned}
&\left(f(X_{\leq i}^{(t+1)}, X_{>i}^{(t)}) - f(X_{<i}^{(t+1)}, X_{\geq i}^{(t)})\right)/\alpha - 2d_i^2 L \alpha \\
&\leq \left(f(X_{\leq i}^{(t+1)}, X_{>i}^{(t)}) - f(X_{<i}^{(t+1)}, X_{\geq i}^{(t)})\right)/\alpha - L\alpha \|\tilde{X}_i^{(t+1)} - X_i^{(t)}\|^2/2 \\
&= \left(f(X_{\leq i}^{(t+1)}, X_{>i}^{(t)}) - f(X_{<i}^{(t+1)}, X_{\geq i}^{(t)}) - L\|X_i^{(t+1)} - X_i^{(t)}\|^2/2\right)/\alpha \\
&\leq \langle C_i^{(t)}, X_i^{(t+1)} - X_i^{(t)} \rangle / \alpha = \langle C_i^{(t)}, \tilde{X}_i^{(t+1)} - X_i^{(t)} \rangle \\
&= \underbrace{\langle \hat{C}_i^{(t)}, \tilde{X}_i^{(t+1)} \rangle}_{I_1} + \underbrace{\langle C_i^{(t)} - \hat{C}_i^{(t)}, \tilde{X}_i^{(t+1)} \rangle}_{I_2} - \langle C_i^{(t)}, X_i^{(t)} \rangle \\
&= \underbrace{\langle \hat{C}_i^{(t)}, \tilde{X}_i^{(t+1)} \rangle}_{I_1} - \underbrace{\langle C_i^{(t)}, \tilde{X}_i^{(t+1)} \rangle}_{I_2} + \underbrace{\langle C_i^{(t)} - \hat{C}_i^{(t)}, \tilde{X}_i^{(t+1)} \rangle}_{I_2} - R_i(X_{<i}^{(t+1)}, X_{\geq i}^{(t)}),
\end{aligned} \tag{B.1}$$

where the first inequality follows from Lemma 2, the second one uses Assumption 1, and the last equality uses the definition (4.1) of the residual function R_i .

Next, we seek to bound I_1 and I_2 in relation (B.1). For the latter,

$$\langle C_i^{(t)} - \hat{C}_i^{(t)}, \tilde{X}_i^{(t+1)} \rangle \leq d_i \|\hat{C}_i^{(t)} - C_i^{(t)}\| = d_i \hat{\lambda} \sqrt{\sum_{j,k:(j,k) \in \mathcal{I}_i^{(t)}} \log^2(n_{s,i} \cdot p_{i,jk}^{(t)})}$$

$$\leq d_i \hat{\lambda} \sqrt{|\mathcal{I}_i^{(t)}| \log^2 \frac{1}{(1-\gamma)w_i n_{s,i}}} = d_i \hat{\lambda} \sqrt{|\mathcal{I}_i^{(t)}|} \log \frac{1}{(1-\gamma)w_i n_{s,i}},$$

where the first inequality comes from the proof of Lemma 2 and the second one uses the formula (3.8) and Assumption 3 (ii), the first equality follows from equation (3.12) and Assumption 3 (ii). Following Hoeffding's inequality, for any $\iota > 0$,

$$\mathbb{P} \left(|\mathcal{I}_i^{(t)}| \geq n_{s,i} + \iota \cdot m_i n_i \right) \leq \exp(-2\iota^2 m_i n_i).$$

Therefore, with probability no less than $1 - \exp(-2\iota^2 m_i n_i)$, we have

$$\langle C_i^{(t)} - \hat{C}_i^{(t)}, \tilde{X}_i^{(t+1)} \rangle \leq d_i \hat{\lambda} \sqrt{n_{s,i} + \iota \cdot m_i n_i} \log \frac{1}{(1-\gamma)w_i n_{s,i}}. \quad (\text{B.2})$$

Regarding the former term I_1 ,

$$\begin{aligned} & \left\langle \hat{C}_i^{(t)}, \tilde{X}_i^{(t+1)} \right\rangle - \left\langle C_i^{(t)}, \bar{X}_i^{(t+1)} \right\rangle \\ & \leq \left\langle \hat{C}_i^{(t)}, \tilde{X}_i^{(t+1)} \right\rangle - \left\langle C_i^{(t)}, \check{X}_i^{(t+1)} \right\rangle - \hat{\lambda} h(\mathbf{a}_i \mathbf{b}_i^\top) \\ & = \left[\left\langle \hat{C}_i^{(t)}, \tilde{X}_i^{(t+1)} \right\rangle + \hat{\lambda} h(\tilde{X}_i^{(t+1)}) \right] - \left[\left\langle C_i^{(t)}, \check{X}_i^{(t+1)} \right\rangle + \hat{\lambda} h(\check{X}_i^{(t+1)}) \right] \\ & \quad + \hat{\lambda} \left[h(\check{X}_i^{(t+1)}) - h(\tilde{X}_i^{(t+1)}) \right] - \hat{\lambda} h(\mathbf{a}_i \mathbf{b}_i^\top) \\ & \leq q_i(\tilde{\mathbf{u}}_i^{(t,\star)}, \tilde{\mathbf{v}}_i^{(t,\star)}; \hat{\lambda}, \hat{\Psi}_i^{(t)}) - q_i(\tilde{\mathbf{u}}_i^{(t,\star)}, \tilde{\mathbf{v}}_i^{(t,\star)}; \hat{\lambda}, \Psi_i^{(t)}) - 2\hat{\lambda} h(\mathbf{a}_i \mathbf{b}_i^\top) \\ & \leq \hat{c}_2 \hat{\lambda} \frac{\|\hat{\Psi}_i^{(t)} - \Psi_i^{(t)}\|_2}{\|\Psi_i^{(t)}\|_2} \left| 1 - \frac{\|\hat{\Psi}_i^{(t)} - \Psi_i^{(t)}\|_2}{\|\Psi_i^{(t)}\|_2} \right|^{-1} - 2\hat{\lambda} h(\mathbf{a}_i \mathbf{b}_i^\top), \end{aligned} \quad (\text{B.3})$$

where the first inequality uses Lemma 1 (with $\mathcal{I} = \mathcal{I}_i^{(t)}$), the second one leverages $h(\check{X}_i^{(t+1)}) \leq 0$, $h(\tilde{X}_i^{(t+1)}) \geq h(\mathbf{a}_i \mathbf{b}_i^\top)$, and the definition (3.4) together with the strong duality and the optimality of $(\tilde{\mathbf{u}}_i^{(t,\star)}, \tilde{\mathbf{v}}_i^{(t,\star)})$ in problem (3.10), the last one follows from [54]. To bound the first term on the right-hand side of the inequality (B.3), we resort to [2, Theorem 3.1]. Specifically, for any $j \in \{1, \dots, m_i\}$, $k \in \{1, \dots, n_i\}$, it holds by equation (3.8) and Assumption 3 that

$$\mathbb{E} \left(\frac{\hat{\psi}_{i,jk}^{(t)}}{\|\Psi_i^{(t)}\|_2} \right) = n_{s,i} \cdot p_{i,jk}^{(t)} \cdot \frac{\psi_{i,jk}^{(t)}}{n_{s,i} \cdot p_{i,jk}^{(t)}} \cdot \frac{1}{\|\Psi_i^{(t)}\|_2} = \frac{\psi_{i,jk}^{(t)}}{\|\Psi_i^{(t)}\|_2},$$

$$\begin{aligned} \text{Var} \left(\frac{\hat{\psi}_{i,jk}^{(t)}}{\|\Psi_i^{(t)}\|_2} \right) & < \frac{(\psi_{i,jk}^{(t)})^2}{n_{s,i} \cdot p_{i,jk}^{(t)} \|\Psi_i^{(t)}\|_2^2} \leq \frac{1}{(1-\gamma)w_i n_{s,i} \|\Psi_i^{(t)}\|_2^2} \\ & \leq \frac{c_1^2 (m_i + n_i)^{-2\nu}}{(1-\gamma)w_i n_{s,i}} \leq \frac{c_1^2 \log^4(1+\varepsilon)}{8(m_i + n_i) \log^4(m_i + n_i)}. \end{aligned}$$

In addition, $\hat{\psi}_{i,jk}^{(t)} / \|\Psi_i^{(t)}\|_2$ lies in an interval with length no larger than

$$\begin{aligned} \frac{\psi_{i,jk}^{(t)}}{n_{s,i} \cdot p_{i,jk}^{(t)} \|\Psi_i^{(t)}\|_2} & \leq \frac{1}{(1-\gamma)w_i n_{s,i} \|\Psi_i^{(t)}\|_2} \leq \frac{c_1 (m_i + n_i)^{-\nu}}{(1-\gamma)w_i n_{s,i}} \\ & \leq \frac{c_1 \log^4(1+\varepsilon)}{8(m_i + n_i)^{1-\nu} \log^4(m_i + n_i)}. \end{aligned}$$

Then, by Theorem 3.1 in [2],

$$\mathsf{P} \left(\frac{\|\hat{\Psi}_i^{(t)} - \Psi_i^{(t)}\|_2}{\|\Psi_i^{(t)}\|_2} \geq c_1(1 + \varepsilon + \zeta) \frac{\log^2(1 + \varepsilon)}{\log^2(m_i + n_i)} \right) < 2 \exp \left(-\frac{16\zeta^2}{\varepsilon^4} \log^4(m_i + n_i) \right)$$

holds for any $\zeta > 0$ and $m_i + n_i \geq 152$. That is to say, with probability no less than $1 - 2 \exp(-16\zeta^2 \log^4(m_i + n_i)/\varepsilon^4)$,

$$\frac{\|\hat{\Psi}_i^{(t)} - \Psi_i^{(t)}\|_2}{\|\Psi_i^{(t)}\|_2} < c_1(1 + \varepsilon + \zeta) \frac{\log^2(1 + \varepsilon)}{\log^2(m_i + n_i)},$$

and

$$\begin{aligned} & \frac{\|\hat{\Psi}_i^{(t)} - \Psi_i^{(t)}\|_2}{\|\Psi_i^{(t)}\|_2} \left| 1 - \frac{\|\hat{\Psi}_i^{(t)} - \Psi_i^{(t)}\|_2}{\|\Psi_i^{(t)}\|_2} \right|^{-1} \\ & < \frac{c_1(1 + \varepsilon + \zeta) \log^2(1 + \varepsilon)}{\log^2(m_i + n_i) - c_1(1 + \varepsilon + \zeta) \log^2(1 + \varepsilon)}. \end{aligned} \quad (\text{B.4})$$

Combining relations (B.1) to (B.4), we have, with probability no less than

$$\left[1 - 2 \exp \left(-\frac{16\zeta^2}{\varepsilon^4} \log^4(m_i + n_i) \right) \right] \left[1 - \exp(-2\iota^2 m_i n_i) \right],$$

that

$$\begin{aligned} f(X_{\leq i}^{(t+1)}, X_{>i}^{(t)}) & \leq f(X_{<i}^{(t+1)}, X_{\geq i}^{(t)}) - \alpha R_i(X_{<i}^{(t+1)}, X_{\geq i}^{(t)}) + 2(d_i \alpha)^2 L \\ & \quad - 2\alpha \hat{\lambda} h(\mathbf{a}_i \mathbf{b}_i^\top) + d_i \alpha \hat{\lambda} \sqrt{n_{s,i} + \iota \cdot m_i n_i} \log \frac{1}{(1-\gamma) w_i n_{s,i}} \\ & \quad + \alpha \hat{\lambda} \frac{c_1 \hat{c}_2 (1 + \varepsilon + \zeta) \log^2(1 + \varepsilon)}{\log^2(m_i + n_i) - c_1(1 + \varepsilon + \zeta) \log^2(1 + \varepsilon)}, \end{aligned}$$

which implies

$$\begin{aligned} \alpha R_i(X_{<i}^{(t+1)}, X_{\geq i}^{(t)}) & \leq f(X_{<i}^{(t+1)}, X_{\geq i}^{(t)}) - f(X_{\leq i}^{(t+1)}, X_{>i}^{(t)}) + 2(d_i \alpha)^2 L \\ & \quad - 2\alpha \hat{\lambda} h(\mathbf{a}_i \mathbf{b}_i^\top) + d_i \alpha \hat{\lambda} \sqrt{n_{s,i} + \iota \cdot m_i n_i} \log \frac{1}{(1-\gamma) w_i n_{s,i}} \\ & \quad + \alpha \hat{\lambda} \frac{c_1 \hat{c}_2 (1 + \varepsilon + \zeta) \log^2(1 + \varepsilon)}{\log^2(m_i + n_i) - c_1(1 + \varepsilon + \zeta) \log^2(1 + \varepsilon)}. \end{aligned}$$

Following similar arguments for proving Theorem 1 and noticing the definition of c_3 , one obtains the desired result. \square

Proof of Corollary 2. Assumption 2 implies bounds for \bar{d}, \bar{h} (see the proof of Corollary 1 in Appendix A) and also $w_i = \Theta(1/m_i n_i)$ ($i = 1, \dots, N$).

The first term on the right-hand side of inequality (4.4) tends to 0 if $t_{\max} = \Theta(\sum_{i=1}^N (m_i + n_i)^\eta)$ with $\eta > \theta$ and M independent from $\{m_i\}_{i=1}^N$ and $\{n_i\}_{i=1}^N$. For a fixed $\iota > 0$, since

$$\sum_{i=1}^N \sqrt{n_{s,i} + \iota \cdot m_i n_i} \log \frac{1}{(1-\gamma) w_i n_{s,i}}$$

$$= \mathcal{O} \left(\sum_{i=1}^N \left[\frac{\sqrt{m_i n_i}}{(m_i + n_i)^{\nu-1/2}} \log^2(m_i + n_i) + \sqrt{m_i n_i} \right] \log \frac{(m_i + n_i)^{2\nu-1}}{\log^4(m_i + n_i)} \right),$$

with the choices of $\{n_{s,i}\}_{i=1}^N$, the second and third terms tend to 0 if

$$\hat{\lambda} = o \left(\frac{1}{\sum_{i=1}^N \sqrt{m_i n_i} \log(m_i + n_i)} \right)$$

and ε, ν, γ are independent from $\{m_i\}_{i=1}^N$ and $\{n_i\}_{i=1}^N$. Incidentally, the choices of $\{n_{s,i}\}_{i=1}^N$ do not conflict with Assumption 3 (ii) by virtue of Remark 3. Since c_1, c_2 , and \hat{c}_2 are also independent from $\{m_i\}_{i=1}^N$ and $\{n_i\}_{i=1}^N$, the last term vanishes as $\sum_{i=1}^N (m_i + n_i) \rightarrow +\infty$. Finally, the probability is not less than

$$\begin{aligned} & \prod_{i=1}^N \left\{ \left[1 - 2 \exp \left(-\frac{16\zeta^2}{\varepsilon^4} \log^4(m_i + n_i) \right) \right] [1 - \exp(-2\iota^2 m_i n_i)] \right\}^{t_{\max}} \\ & \geq \prod_{i=1}^N \left\{ \left[1 - \frac{2}{(m_i + n_i)^{16\zeta^2/\varepsilon^4}} \right] [1 - \exp(-2\iota^2 m_i n_i)] \right\}^{t_{\max}}, \end{aligned}$$

which, by Assumption 2 (iv), goes to 1 as $\sum_{i=1}^N (m_i + n_i) \rightarrow +\infty$ after choosing $\zeta > 0$ such that $\zeta > \sqrt{\eta\varepsilon^2}/4$. The proof is completed. \square

References

- [1] D. Achlioptas, Z. S. Karnin, and E. Liberty, *Near-optimal entrywise sampling for data matrices*, Advances in Neural Information Processing Systems (C. J. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, eds.), Vol. 26, Curran Associates, Inc., 2013, pp. 1565–1573.
- [2] D. Achlioptas and F. McSherry, *Fast computation of low-rank matrix approximations*, J. ACM **54** (2007), no. 2, Art. 9, 19, DOI 10.1145/1219092.1219097.
- [3] M. Ahookhosh, L. T. K. Hien, N. Gillis, and P. Patrinos, *Multi-Block Bregman proximal alternating linearized minimization and its application to orthogonal nonnegative matrix factorization*, Comput. Optim. Appl. **79** (2021), no. 3, 681–715, DOI 10.1007/s10589-021-00286-3.
- [4] M. Ai, F. Wang, J. Yu, and H. Zhang, *Optimal subsampling for large-scale quantile regression*, J. Complex. **62** (2021), 101512, DOI 10.1016/j.jco.2020.101512.
- [5] A. Alfonsi, R. Coyaud, and V. Ehrlacher, *Constrained overdamped Langevin dynamics for symmetric multimarginal optimal transportation*, Math. Models Methods Appl. Sci. **32** (2022), no. 03, 403–455, DOI 10.1142/S0218202522500105.
- [6] A. Alfonsi, R. Coyaud, V. Ehrlacher, and D. Lombardi, *Approximation of optimal transport problems with marginal moments constraints*, Math. Comp. **90** (2021), no. 328, 689–737, DOI 10.1090/mcom/3568.
- [7] J. Altschuler, F. Bach, A. Rudi, and J. Niles-Weed, *Massively scalable Sinkhorn distances via the Nyström method*, Advances in Neural Information Processing Systems (H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché-Buc, E. Fox, and R. Garnett, eds.), Vol. 32, Curran Associates, Inc., 2019, pp. 4427–4437.
- [8] M. Arjovsky, S. Chintala, and L. Bottou, *Wasserstein generative adversarial networks*, Proceedings of the 34th International Conference on Machine Learning (D. Precup and Y. W. Teh, eds.), Proceedings of Machine Learning Research, vol. 70, PMLR, 2017, pp. 214–223.
- [9] A. Beck, E. Pauwels, and S. Sabach, *The cyclic block conditional gradient method for convex optimization problems*, SIAM J. Optim. **25** (2015), no. 4, 2024–2049, DOI 10.1137/15M1008397.
- [10] J.-D. Benamou, Y. Brenier, and K. Guittet, *The Monge-Kantorovich mass transfer and its computational fluid mechanics formulation*, Internat. J. Numer. Methods Fluids **40** (2002), no. 1-2, 21–30, DOI 10.1002/fld.264. ICFD Conference on Numerical Methods for Fluid Dynamics (Oxford, 2001).
- [11] J.-D. Benamou, G. Carlier, and L. Nenna, *A numerical method to solve multi-marginal optimal transport problems with Coulomb cost*, Splitting Methods in Communication, Imaging, Science, and Engineering (R. Glowinski, S. J. Osher, and W. Yin, eds.), Sci. Comput., Springer, Cham, 2016, pp. 577–601.

- [12] D. P. Bertsekas, *Nonlinear Programming*, 3rd ed., Athena Scientific Optimization and Computation Series, Athena Scientific, Belmont, MA, 2016.
- [13] J. Bigot and T. Klein, *Consistent estimation of a population barycenter in the Wasserstein space*, [arXiv: 1212.2562](#) [math.ST], 2012.
- [14] J. Bolte, S. Sabach, and M. Teboulle, *Proximal alternating linearized minimization for nonconvex and nonsmooth problems*, Math. Program. **146** (2014), no. 1-2, 459–494, DOI 10.1007/s10107-013-0701-9.
- [15] A. Borzì and U. Hohenester, *Multigrid optimization schemes for solving Bose-Einstein condensate control problems*, SIAM J. Sci. Comput. **30** (2008), no. 1, 441–462, DOI 10.1137/070686135.
- [16] G. Braun, A. Carderera, C. W. Combettes, H. Hassani, A. Karbasi, A. Mokhtari, and S. Pokutta, *Conditional gradient methods*, [arXiv: 2211.14103](#) [math.OC], 2022.
- [17] V. Braverman, R. Krauthgamer, A. R. Krishnan, and S. Sapir, *Near-optimal entrywise sampling of numerically sparse matrices*, Proceedings of the 34th Conference on Learning Theory (M. Belkin and S. Kpotufe, eds.), Proceedings of Machine Learning Research, vol. 134, PMLR, 2021, pp. 759–773.
- [18] L. M. Brègman, *A relaxation method of finding a common point of convex sets and its application to the solution of problems in convex programming*, Ž. Vyčisl. Mat i Mat. Fiz. **7** (1967), 620–631 (Russian).
- [19] Y. Brenier, *A homogenized model for vortex sheets*, Arch. Rational Mech. Anal. **138** (1997), no. 4, 319–353, DOI 10.1007/s002050050044.
- [20] G. Buttazzo, L. De Pascale, and P. Gori-Giorgi, *Optimal-transport formulation of electronic density-functional theory*, Phys. Rev. A **85** (2012), no. 6, 062502, DOI 10.1103/PhysRevA.85.062502.
- [21] G. Carlier, A. Oberman, and E. Oudet, *Numerical methods for matching for teams and Wasserstein barycenters*, ESAIM Math. Model. Numer. Anal. **49** (2015), no. 6, 1621–1642, DOI 10.1051/m2an/2015033.
- [22] C. Chen, M. Li, X. Liu, and Y. Ye, *Extended ADMM and BCD for nonseparable convex minimization models with quadratic coupling terms: Convergence analysis and insights*, Math. Program. **173** (2019), no. 1-2, 37–77, DOI 10.1007/s10107-017-1205-9.
- [23] H. Chen, G. Friesecke, and C. B. Mendl, *Numerical methods for a Kohn-Sham density functional model based on optimal transport*, J. Chem. Theory Comput. **10** (2014), no. 10, 4360–4368, DOI 10.1021/ct500586q.
- [24] J. Chen and C. J. García-Cervera, *An efficient multigrid strategy for large-scale molecular mechanics optimization*, J. Comput. Phys. **342** (2017), 29–42, DOI 10.1016/j.jcp.2017.04.035.
- [25] M. Colombo, L. De Pascale, and S. Di Marino, *Multimarginal optimal transport maps for one-dimensional repulsive costs*, Canad. J. Math. **67** (2015), no. 2, 350–368, DOI 10.4153/CJM-2014-011-x.
- [26] C. Cotar, G. Friesecke, and C. Klüppelberg, *Density functional theory and optimal transportation with Coulomb cost*, Comm. Pure Appl. Math. **66** (2013), no. 4, 548–599, DOI 10.1002/cpa.21437.
- [27] M. Cuturi, *Sinkhorn distances: Lightspeed computation of optimal transport*, Advances in Neural Information Processing Systems (C. J. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, eds.), Vol. 26, Curran Associates, Inc., 2013, pp. 2292–2300.
- [28] M. Cuturi and A. Doucet, *Fast computation of Wasserstein barycenters*, Proceedings of the 31st International Conference on Machine Learning (E. P. Xing and T. Jebara, eds.), Proceedings of Machine Learning Research, vol. 32, PMLR, 2014, pp. 685–693.
- [29] E. Dagotto, *Complexity in strongly correlated electronic systems*, Science **309** (2005), no. 5732, 257–262, DOI 10.1126/science.1107559.
- [30] S. Di Marino and A. Gerolin, *Optimal transport losses and Sinkhorn algorithm with general convex regularization*, [arXiv: 2007.00976](#) [math.OC], 2020.
- [31] D. Driggs, J. Tang, J. Liang, M. Davies, and C.-B. Schönlieb, *A stochastic proximal alternating minimization for nonsmooth and nonconvex optimization*, SIAM J. Imaging Sci. **14** (2021), no. 4, 1932–1970, DOI 10.1137/20M1387213.
- [32] P. Drineas and A. Zouzias, *A note on element-wise matrix sparsification via a matrix-valued Bernstein inequality*, Inform. Process. Lett. **111** (2011), no. 8, 385–389, DOI 10.1016/j.ipl.2011.01.010.
- [33] P. Dvurechensky, A. Gasnikov, and A. Kroshnin, *Computational optimal transport: Complexity by accelerated gradient descent is better than by Sinkhorn’s algorithm*, Proceedings of the 35th International Conference on Machine Learning (J. Dy and A. Krause, eds.), Proceedings of Machine Learning Research, vol. 80, PMLR, 2018, pp. 1367–1376.
- [34] V. Elvira and L. Martino, *Advances in importance sampling*, Wiley Statist. Ref. Stat. Ref. Online, posted on 2021, 1–14, DOI 10.1002/9781118445112.stat08284.
- [35] O. Fercoq and P. Richtárik, *Optimization in high dimensions via accelerated, parallel, and proximal coordinate descent*, SIAM Rev. **58** (2016), no. 4, 739–771, DOI 10.1137/16M1085905.

- [36] M. Filatov, *Spin-restricted ensemble-referenced Kohn-Sham method: Basic principles and application to strongly correlated ground and excited states of molecules*, Wiley Interdiscip. Rev. Comput. Mol. Sci. **5** (2015), no. 1, 146–167, DOI 10.1002/wcms.1209.
- [37] G. Friesecke, A. Gerolin, and P. Gori-Giorgi, *The strong-interaction limit of density functional theory*, Density Functional Theory: Modeling, Mathematical Analysis, Computational Methods, and Applications (E. Cancès and G. Friesecke, eds.), Mathematics and Molecular Modeling, Springer, 2023, pp. 183–266, DOI 10.1007/978-3-031-22340-2_4.
- [38] G. Friesecke, A. S. Schulz, and D. Vögler, *Genetic column generation: Fast computation of high-dimensional multimarginal optimal transport problems*, SIAM J. Sci. Comput. **44** (2022), no. 3, A1632–A1654, DOI 10.1137/21M140732X.
- [39] X. Geng, *Label distribution learning*, IEEE Trans. Knowl. Data Eng. **28** (2016), no. 7, 1734–1748, DOI 10.1109/TKDE.2016.2545658.
- [40] J. Hertrich and G. Steidl, *Inertial stochastic PALM and applications in machine learning*, Sampl. Theory Signal Process. Data Anal. **20** (2022), no. 1, Paper No. 4, 33, DOI 10.1007/s43670-022-00021-x.
- [41] B. Hosseini and S. Steinerberger, *Intrinsic sparsity of Kantorovich solutions*, C. R. Math. Acad. Sci. Paris **360** (2022), 1173–1175, DOI 10.5802/crmath.392 (English, with English and French summaries).
- [42] Y. Hu, H. Chen, and X. Liu, *A global optimization approach for multi-marginal optimal transport problems with Coulomb cost*, SIAM J. Sci. Comput. **45** (2023), no. 3, A1214–A1238, DOI 10.1137/21M1455164.
- [43] Y. Hu and X. Liu, *The convergence properties of infeasible inexact proximal alternating linearized minimization*, Sci. China Math. **66** (2023), no. 10, 2385–2410, DOI 10.1007/s11425-022-2074-7.
- [44] ———, *The exactness of the ℓ_1 penalty function for a class of mathematical programs with generalized complementarity constraints*, Fundam. Res., posted on 2023, in press, DOI 10.1016/j.fmre.2023.04.006.
- [45] L. V. Kantorovich, *On the translocation of masses*, C. R. (Doklady) Acad. Sci. URSS (N.S.) (1942), 199–201.
- [46] T. Kerdoncuff, R. Emonet, and M. Sebban, *Sampled Gromov Wasserstein*, Mach. Learn. **110** (2021), no. 8, 2151–2186, DOI 10.1007/s10994-021-06035-1.
- [47] Y. Khoo, L. Lin, M. Lindsey, and L. Ying, *Semidefinite relaxation of multimarginal optimal transport for strictly correlated electrons in second quantization*, SIAM J. Sci. Comput. **42** (2020), no. 6, B1462–B1489, DOI 10.1137/20M1310977.
- [48] Y. Khoo and L. Ying, *Convex relaxation approaches for strictly correlated density functional theory*, SIAM J. Sci. Comput. **41** (2019), no. 4, B773–B795, DOI 10.1137/18M1207478.
- [49] S. Kullback and R. A. Leibler, *On information and sufficiency*, Ann. Math. Statistics **22** (1951), no. 1, 79–86, DOI 10.1214/aoms/1177729694.
- [50] A. Kundu, P. Drineas, and M. Magdon-Ismail, *Recovering PCA and sparse PCA via hybrid-(ℓ_1, ℓ_2) sparse sampling of data elements*, J. Mach. Learn. Res. **18** (2017), no. 75, 1–34.
- [51] S. Lacoste-Julien, M. Jaggi, M. Schmidt, and P. Pletscher, *Block-coordinate Frank-Wolfe optimization for structural SVMs*, Proceedings of the 30th International Conference on Machine Learning (S. Dasgupta and D. McAllester, eds.), Proceedings of Machine Learning Research, vol. 28, PMLR, 2013, pp. 53–61.
- [52] C.-P. Lee, *Accelerating inexact successive quadratic approximation for regularized optimization through manifold identification*, Math. Program. **201** (2023), no. 1-2, 599–633, DOI 10.1007/s10107-022-01916-2.
- [53] M. Li, J. Yu, T. Li, and C. Meng, *Importance sparsification for Sinkhorn algorithm*, J. Mach. Learn. Res. **24** (2023), no. 247, 1–44.
- [54] M. Li, J. Yu, H. Xu, and C. Meng, *Efficient approximation of Gromov-Wasserstein distance using importance sparsification*, J. Comput. Graph. Statist. **32** (2023), no. 4, 1512–1523, DOI 10.1080/10618600.2023.2165500.
- [55] Q. Li, Z. Zhu, G. Tang, and M. B. Wakin, *Provable Bregman-divergence based methods for nonconvex and non-Lipschitz problems*, arXiv: 1904.09712 [math.OC], 2019.
- [56] J. S. Liu, *Metropolized independent sampling with comparisons to rejection sampling and importance sampling*, Stat. Comput. **6** (1996), no. 2, 113–119, DOI 10.1007/BF00162521.
- [57] ———, *Monte Carlo Strategies in Scientific Computing*, Springer Series in Statistics, Springer, New York, 2008.
- [58] J. Liu, W. Yin, W. Li, and Y. T. Chow, *Multilevel optimal transport: A fast approximation of Wasserstein-1 distances*, SIAM J. Sci. Comput. **43** (2021), no. 1, A193–A220, DOI 10.1137/18M1219813.
- [59] Z.-Q. Luo and P. Tseng, *On the convergence rate of dual ascent methods for linearly constrained convex minimization*, Math. Oper. Res. **18** (1993), no. 4, 846–867, DOI 10.1287/moor.18.4.846.
- [60] M. Ma, X. Wang, Y. Duan, S. H. Frey, and X. Gu, *Optimal mass transport based brain morphometry for patients with congenital hand deformities*, Vis. Comput. **35** (2019), 1311–1325, DOI 10.1007/s00371-018-1543-5.

- [61] P. Ma, M. Mahoney, and B. Yu, *A statistical perspective on algorithmic leveraging*, Proceedings of the 31st International Conference on Machine Learning (E. P. Xing and T. Jebara, eds.), Proceedings of Machine Learning Research, vol. 32, PMLR, 2014, pp. 91–99.
- [62] C. B. Mendl and L. Lin, *Kantorovich dual solution for strictly correlated electrons in atoms and molecules*, Phys. Rev. B **87** (2013), no. 12, 125106, DOI 10.1103/PhysRevB.87.125106.
- [63] C. Meng, Y. Ke, J. Zhang, M. Zhang, W. Zhong, and P. Ma, *Large-scale optimal transport map estimation using projection pursuit*, Advances in Neural Information Processing Systems (H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, eds.), Vol. 32, Curran Associates, Inc., 2019, pp. 8118–8129.
- [64] G. Monge, *Mémoire sur la théorie des déblais et des remblais*, Histoire de l’Académie Royale des Sciences (1781), 666–704.
- [65] A. B. Owen, *Monte Carlo Theory, Methods and Examples*, Stanford University, 2013.
- [66] O. Pele and M. Werman, *Fast and robust earth mover's distances*, IEEE 12th International Conference on Computer Vision, IEEE, 2009, pp. 460–467, DOI 10.1109/ICCV.2009.5459199.
- [67] G. Peyré and M. Cuturi, *Computational optimal transport: With applications to data science*, Found. Trends Mach. Learn. **11** (2019), no. 5–6, 355–607, DOI 10.1561/2200000073.
- [68] Y. Rubner, L. J. Guibas, and C. Tomasi, *The earth mover's distance, multi-dimensional scaling, and color-based image retrieval*, Proceedings of the ARPA Image Understanding Workshop, ARPA, 1997, pp. 661–668.
- [69] M. Seidl, *Strong-interaction limit of density-functional theory*, Phys. Rev. A **60** (1999), no. 6, 4387, DOI 10.1103/PhysRevA.60.4387.
- [70] M. Seidl, J. P. Perdew, and S. Kurth, *Simulation of all-order density-functional perturbation theory, using the second order and the strong-correlation limit*, Phys. Rev. Lett. **84** (2000), no. 22, 5070, DOI 10.1103/PhysRevLett.84.5070.
- [71] M. Seidl, J. P. Perdew, and M. Levy, *Strictly correlated electrons in density-functional theory*, Phys. Rev. A **59** (1999), no. 1, 51, DOI 10.1103/PhysRevA.59.51.
- [72] C. E. Shannon, *A mathematical theory of communication*, Bell System Tech. J. **27** (1948), 379–423, 623–656, DOI 10.1002/j.1538-7305.1948.tb01338.x.
- [73] R. Sinkhorn and P. Knopp, *Concerning nonnegative matrices and doubly stochastic matrices*, Pacific J. Math. **21** (1967), no. 2, 343–348, DOI 10.2307/2314570.
- [74] R. Sun, Z.-Q. Luo, and Y. Ye, *On the efficiency of random permutation for ADMM and coordinate descent*, Math. Oper. Res. **45** (2020), no. 1, 233–271, DOI 10.1287/moor.2019.0990.
- [75] C. Villani, *Topics in Optimal Transportation*, Vol. 58, American Mathematical Society, 2003.
- [76] H. Wang and J. Zou, *A comparative study on sampling with replacement vs Poisson sampling in optimal subsampling*, Proceedings of the 24th International Conference on Artificial Intelligence and Statistics (A. Banerjee and K. Fukumizu, eds.), Proceedings of Machine Learning Research, vol. 130, PMLR, 2021, pp. 289–297.
- [77] J. Wang, J. Zou, and H. Wang, *Sampling with replacement vs Poisson sampling: A comparative study in optimal subsampling*, IEEE Trans. Inform. Theory **68** (2022), no. 10, 6605–6630.
- [78] S. J. Wright, *Coordinate descent algorithms*, Math. Program. **151** (2015), no. 1, 3–34, DOI 10.1007/s10107-015-0892-3.
- [79] Q. Xia and T. Shi, *A cascadic multilevel optimization algorithm for the design of composite structures with curvilinear fiber based on Shepard interpolation*, Compos. Struct. **188** (2018), 209–219, DOI 10.1016/j.compstruct.2018.01.013.
- [80] Y. Xie, X. Wang, R. Wang, and H. Zha, *A fast proximal point method for computing exact Wasserstein distance*, Proceedings of the 35th Uncertainty in Artificial Intelligence Conference (R. P. Adams and V. Gogate, eds.), Proceedings of Machine Learning Research, vol. 115, PMLR, Jul. 22, pp. 433–453.
- [81] L. Xu, H. Sun, and Y. Liu, *Learning with batch-wise optimal transport loss for 3D shape recognition*, Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE, Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 3333–3342.
- [82] L. Yang and K.-C. Toh, *Bregman proximal point algorithm revisited: A new inexact version and its inertial variant*, SIAM J. Optim. **32** (2022), no. 3, 1523–1554, DOI 10.1137/20M1360748.
- [83] J. Yu, H. Wang, M. Ai, and H. Zhang, *Optimal distributed subsampling for maximum quasi-likelihood estimators with massive data*, J. Amer. Statist. Assoc. **117** (2022), no. 537, 265–276, DOI 10.1080/01621459.2020.1773832.
- [84] P. Zhao and Z. Zhou, *Label distribution learning by optimal transport*, Proceedings of the AAAI Conference on Artificial Intelligence (S. A. McIlraith and K. Q. Weinberger, eds.), Vol. 32, AAAI Press, 2018, pp. 4506–4513, DOI 10.1609/aaai.v32i1.11609.