PRUNE AND SPLIT: A SCALABLE APPROACH TO UNBALANCED OPTIMAL TRANSPORT

Anonymous authors
Paper under double-blind review

ABSTRACT

In this paper, we analyze the T-Sparse Unbalanced Optimal Transport (TSUOT) problem, a variant of the standard Unbalanced Optimal Transport (UOT) problem that enables customizable sparsity, i.e., *Pruning* of long-range transport arcs, via a tunable parameter T. After showing that the TSUOT is well defined for most of the Csiszár divergences, we propose the Main Diagonal Block method, an *ad hoc* Block Coordinate Descent algorithm for UOT problems. At the core of the method is a structure-aware coordinate partition that, by *Splitting* the transport variables into suitably chosen blocks, allows for efficient parallel optimization. The interaction between Pruning and Splitting yields dual advantages. On the one hand, Pruning bounds the number of relevant blocks; on the other hand, Splitting decreases the local Hölder constant. These effects combined improve the overall complexity of the method. Experiments on heterogeneous real-world benchmarks show that the proposed approach achieves comparable numerical accuracy to standard UOT baselines while substantially reducing wall-clock computation time, with speedups of up to two orders of magnitude in the tested regimes. These results position the proposed framework as a competitive approach to large-scale UOT computation.

1 INTRODUCTION

Optimal Transport (OT) has emerged as a powerful mathematical framework to compare probability distributions, with widespread applications in machine learning and computer vision (Arjovsky et al., 2017; Kusner et al., 2015; Rolet et al., 2016). Indeed, OT-based metrics, also known as Wasserstein distances, naturally lift the geometry of the inhabited space X to the space of probability distributions over X (Villani et al., 2008). However, Wasserstein distances can compare only distributions with equal mass, while many objects in machine learning cannot be represented as normalized distributions. To overcome this limitation, Liero et al. (2018) proposed an unbalanced version of optimal transport, known as the Unbalanced Optimal Transport (UOT), where the hard constraint on masses is relaxed via a convex penalty term on the marginals. Given two discrete positive measures μ and ν and a convex function ϕ, the UOT problem reads as follows

$$\mathrm{UOT}(\mu,\nu) = \min_{\pi_{i,j}\geq 0} \left\{ \sum_{i,j=1}^{N} c_{i,j}\pi_{i,j} + \rho\Big(\mathrm{D}_\phi(\pi^{(1)}|\mu) + \mathrm{D}_\phi(\pi^{(2)}|\nu) \Big) \right\}, \tag{1}$$

where $c_{i,j}$ is the cost of moving one unit of mass from x_i to y_j, $\pi^{(\ell)}$ is the ℓ-th marginal of π, $\mathrm{D}_\phi(\cdot|\mu)$ is a Csiszár divergence induced by ϕ with respect to μ, i.e., a convex functional that penalizes mass destruction and mass creation, and $\rho > 0$ is a gauging term.

While offering a theoretically sound framework to compare general positive measures, computing the solution to the UOT problem remains challenging. The number of transport variables grows quadratically with the size of the supports as every transportation plan is represented by a two-dimensional matrix, and the marginal divergence terms couple all variables belonging to the same row or column of the transport plan. Existing approaches often rely on entropic regularizations or variational formulations, which may introduce bias or require careful tuning (Chizat et al., 2018b). In this work, we take a different approach and address these computational bottlenecks through two complementary well-established mechanisms: *Pruning* and *Splitting*. First, we analyze a sparsity-controlled variant of UOT, namely T-Sparse UOT (TSUOT), that prunes variables whose cost is

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

larger than a fixed threshold T, and study the conditions under which the problem is feasible. Sparsification techniques for UOT are relatively under-explored, with existing approaches being restricted to iterative routines that employ an iteration-dependent adaptive pruning (Su et al., 2025; Schmitzer, 2019). To the best of our knowledge, a study of fixed pruning strategies for UOT is still lacking.

Second, we exploit the structural properties of the TSUOT to design a tailored partition of the remaining transport variables, enabling an efficient Block Coordinate Descent scheme to solve the TSUOT problem. Our proposal splits the variables into blocks so that no two variables in the same block share a row/column of the transportation plan. As a result, the objective function restricted to each block becomes fully separable into independent one-dimensional terms. From a practical perspective, this decomposition enables cheap block minimization, simple updates, and easy parallel implementation. This gain is especially evident when the block is selected at random and the resulting one-dimensional subproblems admit a closed-form update: in this case, the method is able to reach convergence without explicitly using first-order information. From a theoretical perspective, the partition yields a small Hölder constant for each block. Moreover, the Pruning step reduces the number of active blocks, leading to better constants in the convergence-rate bounds.

1.1 Related Works

Wasserstein distances have proven effective across machine learning, computer vision, and natural language processing (Arjovsky et al., 2017; Aslam et al., 2024; Cuturi & Doucet, 2014; Frogner et al., 2015; Kusner et al., 2015; Pele & Werman, 2009; Rolet et al., 2016; Rubner et al., 2000; Auricchio et al., 2018). However, these distances are able to compare only measures with equal total mass, which restricts their applicability to settings such as shape analysis and persistence diagrams (Chizat et al., 2018a; Lacombe et al., 2018; Solomon et al., 2015; Vidal et al., 2019). Moreover, the mass conservation constraint makes Wasserstein distances sensitive to noise, limiting their robustness in applications including domain adaptation, imaging, biology, change detection, and text matching (Balaji et al., 2020; Fatras et al., 2021; Lellmann et al., 2014; Sato et al., 2020; Schiebinger et al., 2019; Tang et al., 2023; Fiorucci et al., 2023). To overcome these shortcomings, several notions of Unbalanced Optimal Transport (UOT) have been proposed, including partial transport (Figalli, 2010), the Wasserstein-Fisher-Rao metrics (Chizat et al., 2018a), and other Kantorovich-Rubinstein distances inspired variants (Lellmann et al., 2014).

In this paper, we adopt the UOT framework proposed by Liero et al. (2018), where the marginal constraints are replaced with convex Csiszár divergence penalties. As per the classic OT problem, the majority of approaches make use of the entropic regularization (Chizat et al., 2018b), technique introduced by Cuturi (2013), which enables the use of Sinkhorn-type algorithms (Sinkhorn & Knopp, 1967). In the one-dimensional setting, these relaxations exploit the structure of the problem (Séjourné et al., 2022), improving the overall performance and motivating the usage of sliced UOT variants (Bonet et al., 2024). Alternative strategies include low-rank approximations of the transport plan (Scetbon et al., 2023), ℓ_2-regularized formulations (Blondel et al., 2018; Nguyen et al., 2023), reformulations as convex regression problems (Chapel et al., 2021), and semi-unbalanced models (Fukunaga & Kasai, 2022). We study a sparsified formulation of the UOT problem obtained by pruning variables associated with high-cost edges. Unlike classical OT, whose mass constraints preclude arbitrary pruning (Pele & Werman, 2009), UOT allows mass to be created and destroyed, making pruning a viable and relatively under-explored strategy (Su et al., 2024). To the best of our knowledge, the TSUOT formulation was introduced by Schmitzer (2019) as an auxiliary problem to justify the adoption of a kernel sparsification to reduce the complexity of the Sinkhorn iterates. For this reason, in (Schmitzer, 2019) the analysis of the TSUOT is focused on estimating the error induced by such kernel sparsification. This analysis relies on assumptions that are not stated explicitly, most notably the finiteness of the marginal penalty functions which might compromise the error estimation. Moreover, as the author notes, the Sinkhorn algorithm may fail to converge when applied to a general TSUOT problem. To overcome these issues, we lay out a set of conditions under which the TSUOT problem with fixed truncation parameter is well-defined and pair it with a highly scalable *ad hoc* Block Coordinate Descent method that guarantees fast convergence.

Block Coordinate Descent (BCD) methods represent a classic and well-known family of optimization algorithms for large-scale problems that, by suitably partitioning the variables into smaller blocks, optimize one block at a time while keeping the others fixed, leading to low memory requirements, cheap per-iteration cost, and high scalability (Beck, 2017; Bertsekas, 1999). The efficiency of

these methods depends on how variables are split and on how the block is selected at each iteration. Randomized (Nesterov, 2012) and greedy (aka Gauss-Southwell-type) rules have good practical and theoretical behavior under minimal regularity assumptions (Karimi et al., 2016; Nutini et al., 2015). For a complete introduction to BCD methods, we refer to (Nutini et al., 2022) and the references therein. Block-coordinate and alternating-update schemes also appear frequently in OT problems. For instance, Sinkhorn-type algorithms for entropically regularized OT can be seen as alternating updates of dual scaling variables (Peyré & Cuturi, 2019); Wasserstein barycenter algorithms alternate between transport plans and barycenter measures (Cuturi & Doucet, 2014); moreover, blockwise updates naturally arise in multi-marginal, unbalanced, and low-rank OT (Chizat et al., 2018b; Fukunaga & Kasai, 2022; Scetbon et al., 2023). These methods consider variable blocks that correspond to functionally different components of the model, or to naturally separable/semi-relaxed formulations. Our approach is conceptually different as it splits the primal transportation variables of the UOT problem, which is a non-trivial task due to the coupling of all variables related to entries lying in the same row/column of the transport plan induced by the Csiszár divergences.

1.2 Our Contribution

Our contribution is threefold.

First, we focus on the TSUOT problem, which restricts the formulation introduced in (Liero et al., 2018) by imposing that no mass can be transported beyond a prescribed threshold T. This allows us to prune high-cost transport arcs, reducing the number of variables and promoting sparse transport plans. We characterize the feasibility of the resulting problem and provide an explicit procedure to recover a feasible TSUOT problem whenever the original one is not feasible. Our approach can be viewed as an extension of the framework presented in (Schmitzer, 2019), where the TSUOT formulation is introduced to evaluate the error incurred by truncating the kernel involved in the Sinkhorn iterations. In this paper, we demonstrate the viability and scalability potential of fixed pruning strategies for the UOT problem when combined with an *ad hoc* block partitioning technique.

Second, we propose a tailored Block Coordinate method that fully exploits the structure of the UOT problem. The resulting algorithm, namely the Main Diagonal Block (MDB) method, relies on a structure-aware partition of the transport variables and has two main advantages over standard techniques: (i) each block update decomposes into several independent one-dimensional problems, which give better descent guarantees at each iteration step; (ii) owing to the TSUOT definition, only the blocks containing admissible transport variables need to be considered, so that the number of active blocks only depends on the threshold T. The *Prune and Split* ideas are thus combined to give an efficient and highly scalable optimization scheme for large-scale TSUOT problems. Our algorithm is backed by a rigorous convergence analysis, which ensures the same convergence speed of classic gradient methods with better constants and an overall reduced complexity.

Third, we numerically validate the proposed framework on three different real-world applications: (i) image transfer, (ii) unsupervised change detection on airborne LiDAR point clouds, and (iii) domain adaptation on deep visual features. Across these tasks, our approach is able to attain reliable solutions or exceeds state-of-the-art performance while significantly reducing computation time.

Due to space limits, all the proofs of our statements are deferred to Appendices A.1-A.2.

2 Preliminaries and UOT Model Description

In what follows, we denote by $\mathcal{M}(X)$ the set of positive measures supported over a finite set $X = \{x_i\}_{i\in[N]}$, where $[N] := \{1, \dots, N\}$. Notice that every element of $\mathcal{M}(X)$ can be identified by a collection of positive values $\{\mu_i\}_{i\in[N]}$ through the identity $\mu = \sum_{x_i\in X} \mu_i \delta_{x_i}$, with δ_{x_i} being the Dirac delta centered in x_i. For the sake of simplicity, we assume that the support of μ and ν have the same cardinality. Finally, for every $\gamma \in \mathcal{M}(X \times X)$, we define as $\gamma^{(\ell)}$ the ℓ-th marginal of γ.

Given two positive measures $\mu, \nu \in \mathcal{M}(X)$, a cost function $c : X \times X \to [0, \infty)$, a convex function $\phi : [0, \infty) \to [0, \infty]$ such that $\phi(1) = 0$, and a gauge term $\rho > 0$, the UOT problem is defined as

$$\mathrm{UOT}_{\phi,\mathrm{c},\rho}(\mu, \nu) = \min_{\pi_{i,j}\geq 0} \mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi) = \min_{\pi_{i,j}\geq 0} \left\{ \sum_{i,j=1}^{N} c_{i,j}\pi_{i,j} + \rho(\mathrm{D}_\phi(\pi^{(1)}|\mu) + \mathrm{D}_\phi(\pi^{(2)}|\nu)) \right\}, \quad (2)$$

where $\mathrm{D}_\phi(\alpha|\mu) = \sum_{\substack{i\in[N]\\ \mu_i>0}} \phi\big(\frac{\alpha_i}{\mu_i}\big)\mu_i + \sum_{\substack{i\in[N]\\ \mu_i=0}} \phi'_\infty \alpha_i$, $\alpha = \sum_{i\in[N]} \alpha_i \delta_{x_i}$ and $\phi'_\infty = \lim_{s\to\infty} \frac{1}{s}\phi(s)$.

Notice that $\phi'_\infty = \infty$ if and only if ϕ has superlinear growth for $s \to \infty$ in which case, any solution to (2) satisfies $\pi_i^{(1)} = 0$ if $\mu_i = 0$ and $\pi_j^{(2)} = 0$ if $\nu_j = 0$ (Liero et al., 2018). In what follows, we assume ϕ is continuously differentiable over $(0, \infty)$, $\phi'_\infty = \infty$, and ϕ' is α-Hölder continuous with Hölder constant L'. Under these conditions, $\lambda\phi(s/\lambda)$ has α-Hölder continuous derivative with Hölder constant $\frac{L'}{\lambda^\alpha}$ and hence $\nabla \mathrm{UT}_{\phi,\mathrm{c},\rho}$ is α-Hölder continuous with Hölder constant $L = 2\rho L' N^{\frac{3-\alpha}{2}} \tau^{-\alpha}$, where $\tau = \min_{i,j\in[N]\,:\,\mu_i,\nu_j>0}\{\mu_i, \nu_j\}$.

Given $T > 0$, the TSUOT problem is attained by pruning variables associated with a cost greater than T. In what follows, we denote by $\mathcal{E}_{c,T} := \{(i,j) \in [N] \times [N];\ \ c_{i,j} \le T\}$ the set of indices associated with the active variables of the TSUOT. The selected threshold T plays the role of both a sparsity parameter and an application-dependent tolerance on admissible spatial displacement.

Definition 1 (TSUOT) *Given a cost function c, a convex function $\phi : [0,\infty) \to [0,\infty]$, and a parameter $\rho > 0$, let $T > 0$ be a sparsity parameter. Then, for any pair of positive measures μ and ν supported over X, we define the T-Sparse UOT (TSUOT) problem as*

$$\mathrm{TSUOT}_{\phi,\mathrm{c},\rho,\mathrm{T}}(\mu,\nu) := \min_{\{\pi\ge 0;\ \pi_{i,j}=0\ (i,j)\notin\mathcal{E}_{c,T}\}} \mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi). \tag{3}$$

In general, imposing $\pi_{i,j} = 0$ whenever $c_{i,j} > T$ can render the problem unbounded depending on ϕ. In the following theorem, we characterize this behaviour: (i) We show that if $\phi(0)$ is bounded, the TSUOT problem is well-defined, i.e., it attains finite objective value regardless of T and the positive measures μ and ν. (ii) When $\phi(0) = \infty$, we outline under which conditions on the measures μ, ν and the sparsity parameter T the TSUOT admits bounded minimum.

Theorem 1 *Let ϕ be such that $\phi'_\infty = \infty$. Then, we have the following. If $\phi(0) < \infty$, for every $\mu, \nu \in \mathcal{M}(X)$, cost function c, $\rho > 0$, and $T > 0$ it holds $\mathrm{TSUOT}_{\phi,\mathrm{c},\rho,\mathrm{T}}(\mu,\nu) < \infty$. If $\phi(0) = \infty$, then $\mathrm{TSUOT}_{\phi,\mathrm{c},\rho,\mathrm{T}}(\mu,\nu) < \infty$ if and only if the following two conditions hold: (i) for every x_i in the support of μ, there exists an element y_j in the support of ν for which $c_{i,j} \le T$; and (ii) for every y_j in the support of ν, there exists an element x_i in the support of μ for which $c_{i,j} \le T$.*

We conclude by showing that if the conditions in Theorem 1 are not satisfied, we can prune the positive measures μ and ν to achieve a TSUOT problem with a bounded minimum value.

Corollary 1 *Given a cost function c, a positive parameter ρ and two positive measures $\mu, \nu \in \mathcal{M}(X)$, let ϕ be a convex function such that $\phi(0) = \infty$ and $\phi'_\infty = \infty$. Then, for every $T > 0$, there exists a subset S_μ of the support of μ and a subset S_ν of the support of ν for which the following conditions hold (i) $\mathrm{TSUOT}_{\phi,\mathrm{c},\rho,\mathrm{T}}(\mu',\nu') < \infty$, where μ' and ν' are the positive measures attained by restricting μ on S_μ and ν on S_ν, respectively; and (ii) every point x_i in the support of μ but not in S_μ is such that $c_{i,j} > T$ for every y_j in the support of ν and, vice versa, every element y_j in the support of ν but not in S_ν is such that $c_{i,j} > T$ for every x_i in the support of μ.*

3 The Main-Diagonal Block Method

In this section, we introduce the Main-Diagonal Block (MDB) method, a structure-aware Block Coordinate Descent scheme tailored to the TSUOT problem (see Algorithm 1). The main idea of the MDB method is to exploit the row-column coupling induced by the marginal divergence terms in $\mathrm{UT}_{\phi,\mathrm{c},\rho}$. Rather than using an arbitrary partition of the transport variables, MDB groups coordinates along admissible main-diagonal blocks so that no two variables in the same block share the same row or column. This construction has two key consequences: (i) the restriction of the UOT objective to a selected block separates into independent univariate functions, allowing to parallelize the update step; and (ii) in the TSUOT setting, only blocks containing admissible transport arcs are considered, so the number of active blocks is controlled by the pruning threshold T.

3.1 A Good Block-Splitting Strategy

In what follows, we identify a block of coordinates by the indices of the coordinates that belong to the block, hence $B = \{(i,j)\} \subset [N] \times [N]$ describes the set of coordinates whose indices

Algorithm 1 Main Diagonal Block (MDB) Method

Require: Two positive measures μ and ν over a set of N points, a cost function c, a Csiszár divergence D_ϕ, and an initial point $\pi^{(0)} \geq 0$,

$\pi \leftarrow \pi^{(0)} \in \mathbb{R}^{N\times N}$, $t \leftarrow 0$
while $\pi^{(t)}$ does not satisfy `stopping criterion` **do**
  Select block $B_k \in \mathcal{B}$ according to a `rule`
  for $(i,j) \in B_k$ **do**
    Get $\partial_{i,j}\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)})$
    Compute $\gamma_{i,j}^{(t)} \in \arg\min_{\pi_{i,j}^{(t)}+\gamma_{i,j}\geq 0} \left\{\partial_{i,j}\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)})\gamma_{i,j} + \frac{L^\star}{1+\alpha}|\gamma_{i,j}|^{1+\alpha}\right\}$
    Set $\pi_{i,j}^{(t+1)} \leftarrow \pi_{i,j}^{(t)} + \gamma_{i,j}^{(t)}$
  end for
  Set $\pi_{i,j}^{(t+1)} = \pi_{i,j}^{(t)} \quad \forall\,(i,j) \in B_k^c$, $t \leftarrow t{+}1$
end while
return π

belong to B. Given $T > 0$ and a cost function c, let $\mathcal{E}_{c,T}$ be the set of indices associated with the active variables for the TSUOT. We say that $\mathcal{B} := \{B_k\}_{k\in[K]}$ is a block partition for the TSUOT if $\bigcup_{k\in[K]} B_k = \mathcal{E}_{c,T}$ and $B_k \bigcap B_j = \emptyset$ if $k \neq j$. Given a block of coordinates B_k, we denote by B_k^c the complementary set of indices, that is the set of indices that do not belong to B_k.

Definition 2 *A coordinate partition $\mathcal{B} = \{B_k\}_{k\in[K]}$ is a good partition if, for every $k \in [K]$,*

$$(i,j) \in B_k \implies (i,j') \notin B_k \qquad \forall j' \neq j \in [N], \quad \textit{and} \quad (i',j) \notin B_k \qquad \forall i' \neq i \in [N]. \tag{4}$$

It is always possible to define a good partition of the coordinates in $\mathcal{E}_{c,T} \subseteq [N] \times [N]$. In what follows, we consider the Block-Diagonal Partition (BDP), which is a partition containing N blocks and it is indexed by $k \in [N]$. Indeed, given $k \in [N]$, we define

$$B_k := \left\{(i, 1 + (i+k) \mod N); \quad \text{for all} \quad i \in [N]\right\} \cap \mathcal{E}_{c,T}. \tag{5}$$

By definition, $\{B_k\}_{k\in[N]}$ is a partition of $\mathcal{E}_{c,T}$. Moreover, the BDP is a good partition as $i \to 1 + (i+k) \mod N$ is a bijection from $[N]$ to $[N]$, thus each B_k satisfies the condition in (4). When the objective of the TSUOT is restricted to a block B_k of a good partition, it decomposes into $|B_k|$ independent univariate functions. Indeed, since variables within the same block do not share rows or columns, each variable can be updated independently of the others. This allows for a simpler and efficient implementation of the update step of the Block Coordinate Descent algorithm.

Theorem 2 *Given two positive measures $\mu, \nu \in \mathcal{M}(X)$, a convex and differentiable function ϕ with α-Hölder derivative, a cost function c, and $T, \rho > 0$. Let $\mathcal{B}$ be a good partition. Then, given a block $B_k \in \mathcal{B}$, the function $\mathrm{UT}_{\phi,\mathrm{c},\rho}$ restricted to the variables in B_k, namely $(\mathrm{UT}_{\phi,\mathrm{c},\rho})_{|B_k}$, is the sum of $|B_k|$ fully separable one dimensional functions. In particular, the minimum of $(\mathrm{UT}_{\phi,\mathrm{c},\rho})_{|B_k}$ can be retrieved by independently minimizing $|B_k|$ one-dimensional functions. Moreover, each $(\mathrm{UT}_{\phi,\mathrm{c},\rho})_{|B_k}$ has α-Hölder continuous gradient with Hölder constant $L^\star \leq LN^{-1}$.*

Theorem 2 becomes especially meaningful when the probability measures are supported over regular d-dimensional grids with $N = n^d$ endowed with the ℓ_p norm. In this setting, the structure of the set X and the use of an ℓ_p cost ensure that the minimum number of blocks contained in the partition (5) is determined solely by T and d, and does not depend on the grid side n, as formalized next.

Theorem 3 *Given $T > 0$, let μ and ν be two positive measures on an integer regular d-dimensional grid with n points per side. If $c = \ell_p$, there exists a good partition of $\mathcal{E}_{c,T}$ with $O(T^d)$ blocks.*

3.2 The Block-Selection Scheme

We now describe how to select the block at each iteration of the routine. In what follows, we assume that $\mathrm{UT}_{\phi,\mathrm{c},\rho}$ is differentiable and that $\nabla\,\mathrm{UT}_{\phi,\mathrm{c},\rho}$ is α-Hölder continuous with Hölder constant $L >$

0. Notice that these conditions are satisfied whenever ϕ is differentiable and its derivative is α-Hölder continuous. First, we define

$$\mathcal{H}^{(i,j)}(\pi,\lambda) := -(1+\alpha)\lambda \min_{\pi_{i,j}+\gamma_{i,j}\geq 0} \Big(\partial_{i,j}\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi)\gamma_{i,j} + \frac{\lambda}{1+\alpha}|\gamma_{i,j}|^{1+\alpha}\Big), \tag{6}$$

where $\alpha \in (0,1]$, so that $\mathcal{H}^{(i,j)}(\pi,\lambda)$ is the α-Hölder generalization of the operator introduced in (Karimi et al., 2016). Given a good partition $\mathcal{B} := \{B_k\}_{k\in[K]}$, we set $\mathcal{H}^{(k)}(\pi,\lambda) := \sum_{(i,j)\in B_k} \mathcal{H}^{(i,j)}(\pi,\lambda)$. Owing to the Descent Lemma for functions with Hölder continuous gradient and Theorem 2, $\mathcal{H}^{(k)}(\pi, L^\star)$ quantifies the maximal guaranteed descent on any block $B_k \in \mathcal{B}$.

Given the t-th iterate $\pi^{(t)}$, the MDB selects the block in step 3 of Algorithm 1 according to two `rules`: (i) the Random (R-MDB) selection scheme, which selects the block $k \in [K]$ uniformly at random; and (ii) a Gauss-Southwell (GS-MDB) selection scheme, which selects the block $k \in [K]$ whose associated value $\mathcal{H}^{(k)}(\pi^{(t)}, L^\star)$ is maximal. Notice that the R-MDB does not require the computation of the full gradient, while the GS-MDB does, however, the GS-MDB selects the block with the largest guaranteed decrease according to the local Hölder model.

Theorem 4 *Given $\mu,\nu \in \mathcal{M}(X)$, a cost function c, a convex function ϕ with α Hölder derivative, a gauge parameter ρ, $T > 0$, a good partition $\mathcal{B}$, and $\pi^{(0)}$ a feasible starting point, let $\pi^{(t,\texttt{GS-MDB})}$ be the t-th iterate generated by the GS-MDB and let $\pi^{(t,\texttt{R-MDB})}$ be the random variable describing the t-th iterate of the R-MDB. If $\phi(0) < \infty$ and there exists $\tilde{\mu} > 0$ for which the following holds*

$$\min_{\pi'\in\Pi^\star} \|\pi - \pi'\|_2^2 \leq \tilde{\mu}\,\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi) \qquad \textit{with} \qquad \Pi^\star = \operatorname*{arg\,min}_{\pi\geq 0\,:\,\pi^\star_{i,j}=0\ (i,j)\notin\mathcal{E}_{c,T}} \mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi), \tag{7}$$

for every $\pi \in \mathcal{L}_0 := \{\pi \ : \ \mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi) \leq \mathrm{UT}_{\phi,\mathrm{c}}(\pi^{(0)})\}$, we have the following two convergence guarantees depending on $\alpha \in (0,1]$

- *If $\alpha = 1$, let $L^\star$ be the largest Lipschitz constant of $\nabla\,\mathrm{UT}_{\phi,\mathrm{c},\rho}$ over the blocks in $\mathcal{B}$, then, by setting $\eta = \min\{\frac{1}{2}, \frac{1}{2L^\star\tilde{\mu}}\} > 0$ the following two inequalities hold. For the R-MDB*

$$\mathbb{E}\big[\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t+1,\texttt{R-MDB})})\big] \leq \big(1 - \frac{\eta}{K}\big)^{t+1}\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(0)}), \tag{8}$$

  where $\mathbb{E}$ is the expectation with respect to the random block selections. For the GS-MDB,

$$\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t+1,\texttt{GS-MDB})}) \leq \Big(1 - \frac{\eta}{K}\Big)^{t+1}\,\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(0)}). \tag{9}$$

- *If $\alpha \in (0,1)$, let $r_\alpha = \frac{1-\alpha}{2\alpha}$, $L^\star$ be the largest Hölder constant of $\nabla\,\mathrm{UT}_{\phi,\mathrm{c},\rho}$ over the blocks in $\mathcal{B}$, and m be the cardinality of $\mathcal{E}_{c,T}$. Then, for $\eta_\alpha = \frac{\alpha}{1+\alpha}\min\big\{(L^\star\tilde{\mu}^{\frac{1+\alpha}{2\alpha}}m^{\frac{1-\alpha}{2}})^{-\frac{1}{\alpha}}, \Delta_0^{\frac{\alpha-1}{2\alpha}}\big\}$ the following two inequalities hold. For the R-MDB*

$$\mathbb{E}\big[\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t+1,\texttt{R-MDB})})\big] \leq \big(\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(0)})^{-r_\alpha} + \frac{r_\alpha\eta_\alpha}{K}(t+1)\big)^{-1/r_\alpha}, \tag{10}$$

  where $\mathbb{E}$ is the expectation with respect to the random block selections. For the GS-MDB, it holds

$$\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t+1,\texttt{GS-MDB})}) \leq \big(\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(0)})^{-r_\alpha} + \frac{r_\alpha\eta_\alpha}{K}(t+1)\big)^{-1/r_\alpha}. \tag{11}$$

Remark 1 *The growth condition* (7) *is satisfied whenever ϕ is strongly convex. In this case, D_ϕ is also strongly convex, which allows the objective function of the TSUOT problem to be expressed as the sum of a linear function and a strongly convex function composed with a linear map. The growth condition* (7) *then follows from the argument presented in (Lacoste-Julien & Jaggi, 2015).*

Building on Theorem 4, we obtain the method complexity through straightforward mathematical steps. In Table 1, we report the complexities obtained by considering the MDB with the Gauss-Southwell block selection rule applied to the framework of Theorem 3 for a given primal gap tolerance $\varepsilon > 0$. We notice that the Pruning and Splitting procedure both have a positive effect on the total complexity of the method by reducing the Hölder constant and the number of blocks. Moreover, the splitting procedure reduces the computational and memory costs at each iteration, requiring only n^d memory to store the gradient. It also enables the parallelization of the optimization step, reducing the cost of updating the transport plan to $\frac{n^d}{r}$, where r is the number of available kernels.

Table 1: Effect of *Pruning* and *Splitting* on the complexity of block-coordinate optimization under the assumptions of Theorem 3 and $0 < \alpha \leq 1$. Here n denotes the number of points per grid side, d the dimension, $T \ll n$ the pruning parameter, K the number of blocks, r the number of kernels available to parallelize, OC the Oracle Complexity, HC the Hölder (or Lipschitz for $\alpha = 1$) constant of $\mathrm{UT}_{\phi,\mathrm{c},\rho}$ or $(\mathrm{UT}_{\phi,\mathrm{c},\rho})_{|B_k}$, m is the number of active transport variables and TC the total complexity of the method: if $\alpha = 1$ then $\mathcal{K}_{\alpha,\varepsilon} = \log(\varepsilon^{-1})$, otherwise $\mathcal{K}_{\alpha,\varepsilon} = \varepsilon^{-\frac{1-\alpha}{2\alpha}}$ where ε is the target accuracy on the primal gap. All values are computed under the assumptions that $\eta_\alpha = \frac{\alpha}{1+\alpha}(L\tilde{\mu}^{\frac{1+\alpha}{2}} m^{\frac{1-\alpha}{2}})^{-\frac{1}{\alpha}}$.

Pruning	Splitting	OC	K	HC	m	TC
No	No	$\mathcal{O}(n^{2d})$	1	$\mathcal{O}(n^{\frac{3-\alpha}{2}d})$	n^{2d}	$\mathcal{O}(n^{\frac{5+\alpha}{2\alpha}d}\mathcal{K}_{\alpha,\varepsilon})$
No	Yes	$\mathcal{O}(n^d/r)$	$\mathcal{O}(n^d)$	$\mathcal{O}(n^{\frac{1-\alpha}{2}d})$	n^{2d}	$\mathcal{O}(n^{\frac{3+\alpha}{2\alpha}d}\mathcal{K}_{\alpha,\varepsilon}/r)$
Yes	No	$\mathcal{O}(n^d T^d)$	1	$\mathcal{O}(n^{\frac{1-\alpha}{2}d}T^d)$	$n^d T^d$	$\mathcal{O}(n^{\frac{d}{\alpha}}T^{\frac{3+\alpha}{2\alpha}d}\mathcal{K}_{\alpha,\varepsilon})$
Yes	Yes	$\mathcal{O}(n^d/r)$	$\mathcal{O}(T^d)$	$\mathcal{O}(n^{\frac{1-\alpha}{2}d})$	$n^d T^d$	$\mathcal{O}(n^{\frac{d}{\alpha}}T^{\frac{1+\alpha}{2\alpha}d}\mathcal{K}_{\alpha,\varepsilon}/r)$

4 Numerical Validation

In this section, we validate our proposal via numerical experiments. Our aim is to (i) isolate the contribution of Pruning and Splitting to the efficiency of MDB, and (ii) to show that MDB is competitive and outperforms in terms of task performance or wall-clock time widely used UOT solvers. Appendices A.3-A.6 provide the hardware configuration, detailed experimental settings, and additional quantitative and qualitative results for the three applications considered: image transfer, change detection, and domain adaptation.

Settings. In our experiments, we set the cost function to be the Euclidean distance, i.e., $c(x, y) = \ell_2(x, y) = \|x - y\|_2$ for any couple $x, y \in \mathbb{R}^d$. As per the Csiszár divergence, we consider D_p induced by $\phi_p(s) = \frac{1}{p(p-1)}(s^p - p(s-1) - 1)$ with $1 < p \leq 2$. Owing to Theorem 1, the TSUOT is well defined for every T and the function ϕ_p has α-Hölder continuous gradient with $\alpha = p - 1$. We fix $\rho = 1$ and select T via hyper-parameter tuning.

Baselines. We compare with Sinkhorn (Sink; Chizat et al. (2018b)), translation-invariant Sinkhorn (SinkTI; Séjourné et al. (2022)), low-rank Sinkhorn (SinkLR; Scetbon et al. (2023)), and the majorization-minimization solver for UOT (MM; Chapel et al. (2021)), covering the main UOT solvers. By definition, these methods only solve the UOT problem (intended as in (1)) when ϕ is the Kullback-Leibler divergence.

4.1 Image Transfer

We begin by performing an ablation study on the DOTmark (Schrieber et al., 2017), a benchmark of grayscale images supported over regular grids, with various resolutions. Following Rubner et al. (2000), we cast the comparison of two images as an Image Transfer problem, in which each pixel is assigned with a mass proportional to its gray intensity.

Pruning and splitting. We isolate the effect of pruning and splitting by comparing the four configurations obtained by switching each one on and off independently: no pruning sets $T = \infty$, while no splitting treats the whole support as a single block, updated by projected gradient descent. We run this comparison at resolution 32 because it is small enough to keep both the unsplit and the unpruned configurations tractable. Figure 1 shows that removing either mechanism slows convergence, in line with Table 1. All four configurations converge to the same cost on average: radius $T = 4$ already matches the unpruned optimum to within 0.01%, see Table 3 in Appendix A.4.

Partitioning. We compare our MDB partition against two alternatives, using the same Random/Gauss-Southwell rules of Section 3.2: Coordinate Descent (BCD) (Bertsekas, 1999), whose blocks are single entries, and RowBlock, which groups each source point with all its reachable targets into blocks, as in (Fukunaga & Kasai, 2022). We report this comparison at resolution

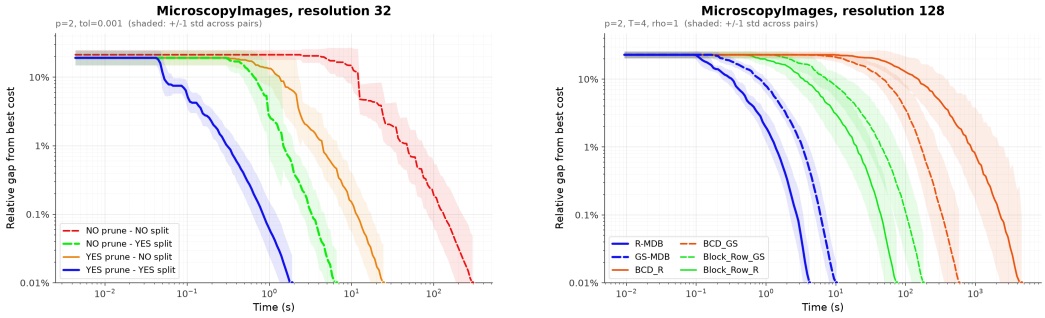


Figure 1: MicroscopyImages, $p = 2.0$, $T = 4$, $\rho = 1$: relevance of pruning and splitting (left), relevance of MDB partitioning (right).

Table 2: mIoU (%) and average wall-clock time (s) across OT-based change-detection methods on the five Urb3DCD test configurations. For each column, we report in **bold** the best results attained and underline the ones attaining the second best result. *Did not complete within 500,000 seconds.

	(1)		(2)		(3)		(4)		(5)		Avg	
Method	Time	mIoU	Time	mIoU	Time	mIoU	Time	mIoU	Time	mIoU	Time	mIoU
Sink	3240	43.8	141715	50.0	3617	42.7	3280	43.1	**10198**	47.3	32410	45.4
SinkTI	8174	45.1	477961	49.9	14890	43.5	8188	45.8	19053	46.7	105653	46.2
SinkLR	17851	7.7	−−*	−−*	17805	7.0	17971	6.6	53564	26.4	26798*	11.9*
MM	162850	35.2	−−*	−−*	163966	32.7	165273	35.0	−−*	−−*	164030*	34.3*
GS-MDB	1132	**63.3**	122237	68.1	1495	**57.3**	1556	**60.1**	19525	**55.5**	29189	**60.9**
R-MDB	**1082**	**63.3**	**117434**	**68.4**	**1452**	**57.3**	**1495**	**60.1**	17054	**55.5**	**27703**	**60.9**

128: a larger support means more blocks for every partition, and MDB's advantage over the alternatives is clearly visible. Figure 1 shows that our partitioning yields a clear advantage over alternatives at the matched pruning radius.

4.2 Change Detection

We evaluate our framework on the Urb3DCD change detection benchmark (de Gélis et al., 2021), comprising bi-temporal LiDAR point clouds of Lyon across five settings: *Lidar05* (baseline), *Lidar10* (high resolution), *Lidar05Noisy* (high noise), *Photogrammetry* (low resolution, high noise), and *MultiSensor* (unmatched point counts). We model point clouds pc_0 and pc_1 acquired at times t_0, t_1 as empirical distributions $\mu_0 = \frac{1}{n_0}\sum_{i=1}^{n_0}\delta_{x_i}$ and $\mu_1 = \frac{1}{n_1}\sum_{j=1}^{n_1}\delta_{y_j}$ over supports $\mathcal{X}_0 = \{x_i\}_{i\in[n_0]}, \mathcal{X}_1 = \{y_j\}_{j\in[n_1]} \subset \mathbb{R}^3$. Bi-temporal alignment is cast as a UOT problem yielding an optimal plan $\hat{P} \in \mathbb{R}^{n_0\times n_1}$. The source geometry pc_0 is mapped onto pc_1 via barycentric projection $\hat{\mathcal{X}}_1 = \text{diag}(\hat{P}^T\mathbf{1}_{n_0})^{-1}\hat{P}^T\mathcal{X}_0$, where pointwise Euclidean distances between $\hat{\mathcal{X}}_1$ and $\mathcal{X}_1$ quantify structural changes over time. We compute the signed squared displacement $s_j = \|y_j - \hat{y}_j\|_2^2\,\text{sign}(y_{j,z} - \hat{y}_{j,z})$ between each target point y_j and its barycentric projection $\hat{y}_j$: a point is labeled as changed if $|s_j|$ exceeds the threshold h, and as a new building ($s_j > 0$) or demolition ($s_j < 0$) according to the sign.

The hyper-parameters of all methods were selected via grid search on Lidar05, and the best configuration was then used in all five test settings to obtain Table 2. Our findings show that GS-MDB and R-MDB deliver both the best average mIoU and attain the lowest runtimes across all but one of the considered datasets.

4.3 Domain Adaptation

We further evaluate our framework on unsupervised domain adaptation (Ben-David et al., 2010), following the mini-batch UOT pipeline of Fatras et al. (2021): a UOT penalty-term between source and target embeddings is added to the classification loss of ResNet-50 and minimized jointly. We

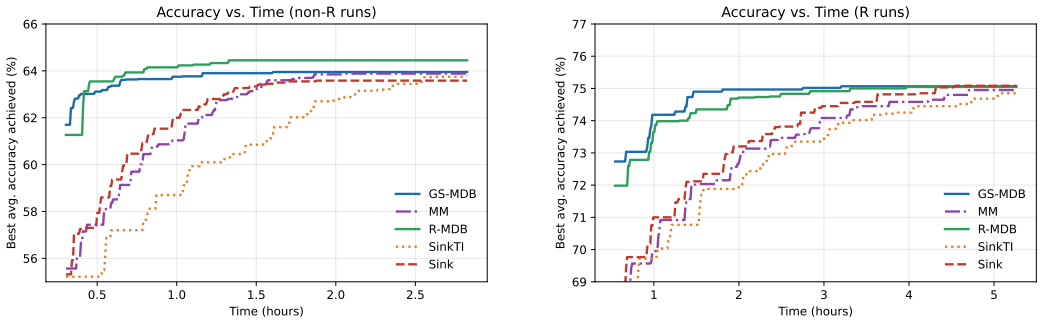


Figure 2: Best average accuracy achievable within a given iteration budget on Office-Home, on the left, averaged over the 6 non-R task pairs and, on the right, averaged over the 6 R task pairs. The pipelines adopting R-MDB and GS-MDB reach their near-final accuracy within roughly 1.5 hours, while the others need longer to reach the same level. SinkLR is omitted because of its low performances (see Table 7).

adopt this pipeline unchanged, replacing only its UOT penalty-term. We evaluate on Office-Home (Venkateswara et al., 2017), whose four domains yield 12 adaptation tasks. Each task is a 65-way classification of everyday objects shared across the four domains (Art, Clipart, Product, and Real-World), which differ in visual style and background.

For every method, we give a budget of 5000 iterations and evaluate the accuracy every 500. Figure 2 reports how the best average accuracy evolves through time. Experiments involving the Real-World domain (R) are plotted separately, as they take twice as long to run than the others. Since solving 65×65 UOT problems takes a negligible amount of time compared to network training, we attribute the faster accuracy gains of R-MDB and GS-MDB to the sparsity of the solutions to the TSUOT. Indeed, unlike Sinkhorn-type and MM solvers which maintain denser couplings, MDB methods give sparse plans concentrating mass early on a few likely class-consistent source-target pairs. This might reduce spurious cross-class couplings and the associated transfer of incorrect labels.

5 Conclusion and Limitations

In this work, we introduced a computational framework for scalable UOT based on two mechanisms, that is Pruning and Splitting. Our TSUOT formulation reduces the dimensionality of the transport problem by pruning the set of variables whose associated cost is larger than a given parameter $T > 0$. Building on this sparse structure, we designed the MDB method, a tailor-made implementation of the Block Coordinate Descent scheme in which we split the variables into blocks that make the function fully separable. Beyond the practical computational gains, our analysis highlights how the proposed pruning and splitting mechanisms jointly improve the optimization complexity. In particular, the sparsity induced by the pruning step reduces the number of relevant blocks, while the proposed splitting yields better smoothness properties through smaller block Hölder constants, which reduces the complexity of the method. Together, these properties lead to scalable optimization procedures that are parallelizable and well-suited for large-scale UOT problems. Finally, we validated the approach on different real-world problems: change detection and domain adaptation. Our results confirm that the TSUOT formulation combined with the *ad hoc* MDB method leads to an efficient and scalable alternative to the standard UOT problem. Moreover, the sparsity threshold T is an easy to interpret parameter as it controls the spatial range of admissible transport interactions.

Limitations: While the MDB provides a competitive and scalable approach to UOT, the method yields best performance when $\phi(0)$ is bounded, as otherwise it requires to adapt the input measures depending on the cost between support points. Moreover, for specific tasks, the pruning might discard high-cost arcs needed to connect two relevant points in the problem at hand.

AI USE STATEMENT

In this work, we used generative AI tools for providing feedback on research methodology or experiments. We have not used generative AI tools for generating synthetic data sets; helping develop theoretical models or conceptual frameworks; formulating mathematical claims; providing critical ingredients for proving mathematical claims; assisting in the writing of proofs; proposing or refining hypotheses; implementing methods; cleaning and reformatting datasets; or interpreting results. The remaining required disclosure tasks, assisting with translation and supporting qualitative and thematic data analysis, are not applicable to this work. Additionally, we used generative AI tools for editing the research paper to improve readability. We have reviewed all AI-assisted work. Each review comment and editorial suggestion was assessed by the authors against the original manuscript, its mathematical derivations, and the reported experimental results. No AI-generated mathematical arguments, proofs, claims, or interpretations of results were incorporated into the paper. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The paper discloses the experimental setup in the "Numerical Validation" section, including the information needed to reproduce the reported results. The data are open access, whereas the code is provided as supplementary material.

REFERENCES

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. *Proceedings of Machine Learning Research*, 70:214–223, 2017. doi: 10.48550/arXiv.1701.07875.

Muhammad Haseeb Aslam, Muhammad Osama Zeeshan, Soufiane Belharbi, Marco Pedersoli, Alessandro Lameiras Koerich, Simon Bacon, and Eric Granger. Distilling Privileged Multimodal Information for Expression Recognition using Optimal Transport. In *2024 IEEE 18th International Conference on Automatic Face and Gesture Recognition (FG)*, pp. 1–10, 2024. doi: 10.1109/FG59268.2024.10582047.

Gennaro Auricchio, Federico Bassetti, Stefano Gualandi, and Marco Veneroni. Computing Kantorovich-Wasserstein distances on d-dimensional histograms using $(d+1)$-partite graphs. In *Advances in Neural Information Processing Systems*, pp. 5793–5803, 2018. doi: 10.48550/arXiv.1805.07416.

Yogesh Balaji, Rama Chellappa, and Soheil Feizi. Robust optimal transport with applications in generative modeling and domain adaptation. *Advances in Neural Information Processing Systems*, 33:12934–12944, 2020. doi: 10.48550/arXiv.2010.05862.

Amir Beck. *First-Order Methods in Optimization*. MOS-SIAM Series on Optimization. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2017. ISBN 978-1-61197-498-0.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine Learning*, 79(1-2):151–175, 2010. doi: 10.1007/s10994-009-5152-4.

Dimitri P. Bertsekas. *Nonlinear Programming*. Athena Scientific, Belmont, MA, 2nd edition, 1999. ISBN 978-1886529007.

Mathieu Blondel, Vivien Seguy, and Antoine Rolet. Smooth and sparse optimal transport. In *International Conference on Artificial Intelligence and Statistics*, pp. 880–889. PMLR, 2018. doi: 10.48550/arXiv.1710.06276.

Clément Bonet, Kimia Nadjahi, Thibault Séjourné, Kilian Fatras, and Nicolas Courty. Slicing unbalanced optimal transport. *Transactions on Machine Learning Research*, 2024. doi: 10.48550/arXiv.2306.07176.

Kristian Bredies. A forward–backward splitting algorithm for the minimization of non-smooth convex functionals in Banach space. *Inverse Problems*, 25(1):015005, 2009. doi: 10.1088/0266-5611/25/1/015005.

Laetitia Chapel, Rémi Flamary, Haoran Wu, Cédric Févotte, and Gilles Gasso. Unbalanced optimal transport through non-negative penalized linear regression. *Advances in Neural Information Processing Systems*, 34:23270–23282, 2021. doi: 10.48550/arXiv.2106.04145.

Lenaic Chizat, Gabriel Peyré, Bernhard Schmitzer, and François-Xavier Vialard. An interpolating distance between optimal transport and Fisher–Rao metrics. *Foundations of Computational Mathematics*, 18(1):1–44, 2018a. doi: 10.1007/s10208-016-9331-y.

Lenaic Chizat, Gabriel Peyré, Bernhard Schmitzer, and François-Xavier Vialard. Scaling algorithms for unbalanced optimal transport problems. *Mathematics of Computation*, 87(314):2563–2609, 2018b. doi: 10.1090/mcom/3303.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems*, pp. 2292–2300, 2013. doi: 10.5555/2999792.2999868.

Marco Cuturi and Arnaud Doucet. Fast computation of Wasserstein barycenters. *Proceedings of Machine Learning Research*, 32(2):685–693, 2014. doi: 10.48550/arXiv.1310.4375.

Bharath Bhushan Damodaran, Benjamin Kellenberger, Rémi Flamary, Devis Tuia, and Nicolas Courty. DeepJDOT: Deep joint distribution optimal transport for unsupervised domain adaptation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 447–463, 2018. doi: 10.1007/978-3-030-01225-0_28.

Iris de Gélis, Sébastien Lefèvre, and Thomas Corpetti. Urb3DCD: Urban point clouds simulated dataset for 3D change detection, 2021.

Kilian Fatras, Thibault Séjourné, Rémi Flamary, and Nicolas Courty. Unbalanced minibatch optimal transport; applications to domain adaptation. In *International Conference on Machine Learning*, pp. 3186–3197. PMLR, 2021. doi: 10.48550/arXiv.2103.03606.

Alessio Figalli. The optimal partial transport problem. *Archive for Rational Mechanics and Analysis*, 195(2):533–560, 2010. doi: 10.1007/s00205-008-0212-7.

Marco Fiorucci, Peter Naylor, and Makoto Yamada. Optimal transport for change detection on lidar point clouds. In *IGARSS 2023 - 2023 IEEE International Geoscience and Remote Sensing Symposium*, pp. 982–985, 2023. doi: 10.1109/IGARSS52108.2023.10283101.

Charlie Frogner, Chiyuan Zhang, Hossein Mobahi, Mauricio Araya, and Tomaso A. Poggio. Learning with a Wasserstein loss. In *Advances in Neural Information Processing Systems*, pp. 2053–2061, 2015. doi: 10.5555/2969442.2969469.

Takumi Fukunaga and Hiroyuki Kasai. Block-coordinate Frank-Wolfe algorithm and convergence analysis for semi-relaxed optimal transport problem. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5433–5437. IEEE, 2022. doi: 10.1109/ICASSP43922.2022.9746032.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016. doi: 10.1109/CVPR.2016.90.

Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the Polyak-Lojasiewicz condition. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 795–811. Springer, 2016. doi: 10.1007/978-3-319-46128-1_50.

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. From word embeddings to document distances. In *International Conference on Machine Learning*, pp. 957–966. PMLR, 2015. doi: 10.5555/3045118.3045221.

Théo Lacombe, Marco Cuturi, and Steve Oudot. Large scale computation of means and clusters for persistence diagrams using optimal transport. *Advances in Neural Information Processing Systems*, 31, 2018. doi: 10.48550/arXiv.1805.08331.

Simon Lacoste-Julien and Martin Jaggi. On the global linear convergence of Frank-Wolfe optimization variants. *Advances in Neural Information Processing Systems*, 28, 2015. doi: 10.48550/arXiv.1511.05932.

Dimitri Lague, Nicolas Brodu, and Jérôme Leroux. Accurate 3D comparison of complex topography with terrestrial laser scanner: Application to the rangitikei canyon (nz). *ISPRS Journal of Photogrammetry and Remote Sensing*, 82:10–26, 2013. doi: 10.1016/j.isprsjprs.2013.04.009.

Jan Lellmann, Dirk A. Lorenz, Carola Schonlieb, and Tuomo Valkonen. Imaging with Kantorovich–Rubinstein discrepancy. *SIAM Journal on Imaging Sciences*, 7(4):2833–2859, 2014. doi: 10.1137/140975528.

Matthias Liero, Alexander Mielke, and Giuseppe Savaré. Optimal entropy-transport problems and a new Hellinger–Kantorovich distance between positive measures. *Inventiones Mathematicae*, 211 (3):969–1117, 2018. doi: 10.1007/s00222-017-0759-8.

Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I. Jordan. Conditional adversarial domain adaptation. In *Advances in Neural Information Processing Systems*, volume 31, 2018. doi: 10.48550/arXiv.1705.10667.

Peter Naylor, Diego Di Carlo, Arianna Traviglia, Makoto Yamada, and Marco Fiorucci. Implicit neural representation for change detection. *arXiv preprint arXiv:2307.15428*, 2023. doi: 10.48550/arXiv.2307.15428.

Yurii Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22(2):341–362, 2012. doi: 10.1137/100802001.

Quang Minh Nguyen, Hoang H. Nguyen, Yi Zhou, and Lam M. Nguyen. On unbalanced optimal transport: Gradient methods, sparsity and approximation error. *Journal of Machine Learning Research*, 24(384):1–41, 2023. doi: 10.5555/3648699.3649083.

Julie Nutini, Mark Schmidt, Issam Laradji, Michael Friedlander, and Hoyt Koepke. Coordinate descent converges faster with the Gauss-Southwell rule than random selection. In *Proceedings of the International Conference on Machine Learning*, pp. 1632–1641, 2015. doi: 10.5555/3045118.3045292.

Julie Nutini, Issam Laradji, and Mark Schmidt. Let's make block coordinate descent converge faster: Faster greedy rules, message-passing, active-set complexity, and superlinear convergence. *Journal of Machine Learning Research*, 23(131):1–74, 2022. doi: 10.5555/3586589.3586720.

Ofir Pele and Michael Werman. Fast and robust Earth Mover's distances. In *2009 IEEE 12th International Conference on Computer Vision*, pp. 460–467. IEEE, 2009. doi: 10.1109/ICCV.2009.5459199.

Gabriel Peyré and Marco Cuturi. *Computational Optimal Transport*. Now Publishers, 2019. doi: 10.1561/2200000073.

Antoine Rolet, Marco Cuturi, and Gabriel Peyré. Fast dictionary learning with a smoothed Wasserstein loss. In *Artificial Intelligence and Statistics*, pp. 630–638. PMLR, 2016.

Yossi Rubner, Carlo Tomasi, and Leonidas J. Guibas. The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision*, 40:99–121, 2000. doi: 10.1023/A:1026543900054.

Ryoma Sato, Makoto Yamada, and Hisashi Kashima. Fast unbalanced optimal transport on a tree. *Advances in Neural Information Processing Systems*, 33:19039–19051, 2020. doi: 10.48550/arXiv.2006.02703.

Meyer Scetbon, Michal Klein, Giovanni Palla, and Marco Cuturi. Unbalanced low-rank optimal transport solvers. *Advances in Neural Information Processing Systems*, 36:52312–52325, 2023. doi: 10.48550/arXiv.2305.19727.

Geoffrey Schiebinger, Jian Shu, Marcin Tabaka, Brian Cleary, Vidya Subramanian, Aryeh Solomon, Joshua Gould, Siyan Liu, Stacie Lin, Peter Berube, et al. Optimal-transport analysis of single-cell gene expression identifies developmental trajectories in reprogramming. *Cell*, 176(4):928–943, 2019. doi: 10.1016/j.cell.2019.01.006.

Bernhard Schmitzer. Stabilized sparse scaling algorithms for entropy regularized transport problems. *SIAM Journal on Scientific Computing*, 41(3):A1443–A1481, 2019. doi: 10.1137/16M1106018.

Jörn Schrieber, Dominic Schuhmacher, and Carsten Gottschlich. DOTmark – a benchmark for discrete optimal transport. *IEEE Access*, 5:271–282, 2017. doi: 10.1109/ACCESS.2016.2639065.

Thibault Séjourné, François-Xavier Vialard, and Gabriel Peyré. Faster unbalanced optimal transport: Translation invariant Sinkhorn and 1-d Frank-Wolfe. In *International Conference on Artificial Intelligence and Statistics*, pp. 4995–5021. PMLR, 2022. doi: 10.48550/arXiv.2201.00730.

Richard Sinkhorn and Paul Knopp. Concerning nonnegative matrices and doubly stochastic matrices. *Pacific Journal of Mathematics*, 21(2):343–348, 1967. doi: 10.2140/pjm.1967.21.343.

Justin Solomon, Fernando De Goes, Gabriel Peyré, Marco Cuturi, Adrian Butscher, Andy Nguyen, Tao Du, and Leonidas Guibas. Convolutional Wasserstein distances: Efficient optimal transportation on geometric domains. *ACM Transactions on Graphics (TOG)*, 34(4):66, 2015. doi: 10.1145/2766963.

Xun Su, Zhongxi Fang, and Hiroyuki Kasai. Safe screening for ℓ_2-penalized unbalanced optimal transport problem. In *2024 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8. IEEE, 2024. doi: 10.1109/IJCNN60899.2024.10650283.

Xun Su, Zhongxi Fang, and Hiroyuki Kasai. Safe screening for unbalanced optimal transport. *Neurocomputing*, pp. 132308, 2025. doi: 10.1016/j.neucom.2025.132308.

Wei Tang, Zuyao Ma, Haifeng Sun, and Jingyu Wang. Learning sparse alignments via optimal transport for cross-domain fake news detection. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. IEEE, 2023. doi: 10.1109/ICASSP49357.2023.10095281.

Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5018–5027, 2017. doi: 10.1109/CVPR.2017.572.

Jules Vidal, Joseph Budin, and Julien Tierny. Progressive Wasserstein barycenters of persistence diagrams. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):151–161, 2019. doi: 10.1109/TVCG.2019.2934256.

Cédric Villani et al. *Optimal Transport: Old and New*, volume 338. Springer, 2008. doi: 10.1007/978-3-540-71050-9.

A TECHNICAL APPENDIX

In this appendix we report the missing proofs of our statements and additional numerical results.

A.1 MISSING PROOFS

Proof 1 (Proof of Theorem 1) *Let us denote by $\pi^{(0)}$ the null transportation plan, that is $\pi^{(0)}_{i,j} = 0$ for every i and for every j. We observe that if $\phi(0) < \infty$, then*

$$\mathrm{UOT}_{\phi,K^{[T]},\rho}(\mu,\nu) \leq \mathrm{UT}_{\phi,K^{[T]},\rho}(\pi^{(0)}) = \rho\Big(\sum_{i\in[N]}\phi(0)\mu_i + \sum_{j\in[N]}\phi(0)\nu_j\Big) < \infty, \tag{12}$$

where $K^{[T]}$ is defined as

$$K^{[T]}_{i,j} := K^{[T]}(x_i,y_j) = c_{i,j} + \iota_{\mathcal{D}_T}(x_i,y_j), \tag{13}$$

where $(x_i,y_j) \in X\times X$ and $\iota_{\mathcal{D}_T}$ is the indicator function of the set $\mathcal{D}_T := \{c_{i,j} \leq T\}$ see (Schmitzer, 2019). Since the problem defining $\mathrm{UOT}_{\phi,K^{[T]},\rho}$ attains a minimum (Liero et al., 2018), we then conclude that the problem attains a minimum and, on such minimum, the objective function has bounded value, which concludes the first part of the statement.

Let us now assume that $\phi(0) = \infty$ and let us define π as follows

$$\pi_{i,j} = \begin{cases} 1 & \text{if } c_{i,j} \leq T \text{ and } 0 < \mu_i,\nu_j, \\ 0 & \text{otherwise.} \end{cases} \tag{14}$$

Then, we have that $\pi^{(1)}_i > 0$ for every i for which $\mu_i > 0$ if and only if for every x_i in the support of μ there exists at least one y_j in the support of ν for which $c_{i,j} \leq T$. Likewise, $\pi^{(2)}_j > 0$ if and only if for every y_j in the support of ν there exists at least one x_i in the support of μ for which $c_{i,j} \leq T$.

Then, if $\pi^{(1)}_i > 0$ and $\pi^{(2)}_j > 0$ for every i and j then it holds

$$\begin{aligned}\mathrm{TSUOT}_{\phi,\mathrm{c},\rho,\mathrm{T}}(\mu,\nu) \leq\ & T\sum_{i\in[N],j\in[N]}\pi_{i,j} + \rho\sum_i \max\left\{\phi\Big(\frac{N_\nu}{\mu_i}\Big),\phi\Big(\frac{1}{\mu_i}\Big)\right\}\mu_i \\ & + \rho\sum_{j\in[N]}\max\left\{\phi\Big(\frac{N_\mu}{\nu_j}\Big),\phi\Big(\frac{1}{\nu_j}\Big)\right\}\nu_j \\ & < \infty,\end{aligned}$$

where N_ν is the number of points in the support of ν and N_μ is the number of points in the support of μ.

To conclude, we notice that if x_i is an element in the support of μ such that $c_{i,j} > T$ for every y_j in the support of ν then every feasible plan π for the TSUOT is such that $\pi^{(1)}_i = 0$, thus $\mathrm{TSUOT}_{\phi,\mathrm{c},\rho,\mathrm{T}}(\mu,\nu) = \infty$, which concludes the proof.

Proof 2 (Proof of Corollary 1) *It follows from Theorem 1: given two positive measures μ and ν, we define S_μ and S_ν as follows*

$$S_\mu := \{x_i \in X,\ \text{s.t. } \mu_i > 0 \text{ and } \exists y_j \in X;\ \nu_j > 0 \text{ and } c_{i,j} \leq T\} \tag{15}$$

and, likewise

$$S_\nu := \{y_j \in X,\ \text{s.t. } \nu_j > 0 \text{ and } \exists x_i \in X;\ \mu_i > 0 \text{ and } c_{i,j} \leq T\}. \tag{16}$$

Let us now define as μ' and ν' the measures obtained by restricting μ to S_μ and ν to S_ν respectively. Owing to the definition of S_μ and S_ν, μ' and ν' satisfy the conditions of Theorem 1, so that $\mathrm{TSUOT}_{\phi,\mathrm{c},\rho,\mathrm{T}}(\mu',\nu') < \infty$, which allows us to conclude the proof.

Proof 3 (Proof of Theorem 2) *Let $\mathcal{B} := \{B_k\}_{k\in[K]}$ be a good partition. Given B_k the block of coordinates associated with $k \in [K]$, we define the set $\mathcal{I}_k$ and $\mathcal{J}_k$ as follows*

$$\mathcal{I}_k := \{i \in [N] \text{ s.t. } \exists j \in [N] \quad (i,j) \in B_k\} \tag{17}$$

$$\mathcal{J}_k := \{j \in [N] \text{ s.t. } \exists i \in [N] \quad (i,j) \in B_k\}. \tag{18}$$

By definition of good partition, we have that $|\mathcal{I}_k| = |\mathcal{J}_k|$. Moreover, there exists a bijective function $F_k : \mathcal{I}_k \to \mathcal{J}_k$ that assigns every $i \in \mathcal{I}_k$ the unique element in $\mathcal{J}_k$ for which it holds $(i, j) \in B_k$.

Let us now consider the function $\mathrm{UT}_{\phi,\mathrm{c},\rho}$. By definition, we can write

$$\begin{aligned}
\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi) &= \sum_{i\in[N],j\in[N]} c_{i,j}\pi_{i,j} + \rho\Big(\mathrm{D}_\phi(\pi^{(1)}|\mu) + \mathrm{D}_\phi(\pi^{(2)}|\nu)\Big) \\
&= \sum_{i\in\mathcal{I}_k} \Big(c_{i,F_k(i)}\pi_{i,F_k(i)} + \rho\phi\Big(\frac{\sum_{j\in[N]}\pi_{i,j}}{\mu_i}\Big)\mu_i + \rho\phi\Big(\frac{\sum_{i\in[N]}\pi_{i,F_k(i)}}{\nu_{F_k(i)}}\Big)\nu_{F_k(i)}\Big) \\
&\quad + \sum_{(i,j)\in B_k^c} c_{i,j}\pi_{i,j} + \rho\Big(\sum_{i\in\mathcal{I}_k^c}\phi\Big(\frac{\sum_{j\in[N]}\pi_{i,j}}{\mu_i}\Big)\mu_i + \sum_{j\in\mathcal{J}_k^c}\phi\Big(\frac{\sum_{i\in[N]}\pi_{i,j}}{\nu_j}\Big)\nu_j\Big) \\
&= \sum_{i\in\mathcal{I}_k} G_{i,j}(\pi_{i,j}) + K(\{\pi_{i,j}\}_{(i,j)\in B_k^c}),
\end{aligned}$$

where $G_{i,j}$ is defined as

$$G_{i,j} : \gamma \to c_{i,j}\gamma + \rho\Big(\mu_i\phi\Big(\frac{\gamma + \sum_{l\neq j}\pi_{i,l}}{\mu_i}\Big) + \nu_j\phi\Big(\frac{\gamma + \sum_{l\neq i}\pi_{l,j}}{\nu_j}\Big)\Big) \qquad \gamma \geq 0. \tag{19}$$

and K is defined as

$$K(\{\pi_{i,j}\}_{i,j\in B_k^c}) := \sum_{(i,j)\in B_k^c} c_{i,j}\pi_{i,j} + \rho\Big(\sum_{i\in\mathcal{I}_k^c}\phi\Big(\frac{\sum_{j\in[N]}\pi_{i,j}}{\mu_i}\Big)\mu_i + \sum_{j\in\mathcal{J}_k^c}\phi\Big(\frac{\sum_{i\in[N]}\pi_{i,j}}{\nu_j}\Big)\nu_j\Big) \tag{20}$$

and thus is a function that does not depend on the coordinates $\{\pi_{i,j}\}_{i,j\in B_k}$. This proves that the restriction of $\mathrm{UT}_{\phi,\mathrm{c},\rho}$ to B_k is fully separable. Moreover, since the constraint $\pi_{i,j} \geq 0$ is separable along every variable and thus minimizing $\mathrm{UT}_{\phi,\mathrm{c},\rho}$ with respect to the coordinates in B_k is equivalent to minimize each $G_{i,j}$ individually.

To conclude, we study the Hölder continuity of the gradient of the function $\mathrm{UT}_{\phi,\mathrm{c},\rho}$ restricted to B_k, which we denote by $(\mathrm{UT}_{\phi,\mathrm{c},\rho})_{|B_k}$. Without loss of generality, we assume that $\mu_i > 0$ and $\nu_j > 0$ for every $i, j \in [N]$, the general case can be recovered by suitably restricting the support of μ and ν. Owing to the previous point, we have that the regularity of $(\mathrm{UT}_{\phi,\mathrm{c},\rho})_{|B_k}$ is the same as the regularity of $\sum_{(i,j)\in B_k} G_{i,j}$ since

$$\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi) = \sum_{i\in\mathcal{I}_k} G_{i,j}(\pi_{i,j}) + K(\{\pi_{i,j}\}_{(i,j)\in B_k^c}) \tag{21}$$

with K being constant with respect to the variables $\{\pi_{i,j}\}_{(i,j)\in B_k}$. By definition, we have

$$\partial_{i,j}(\mathrm{UT}_{\phi,\mathrm{c},\rho})_{|B_k}(\{\pi_{i,j}\}_{(i,j)\in B_k}) = c_{i,j} + \rho\phi'\left(\frac{\pi_{i,j} + \sum_{\ell\neq j}\pi_{i,\ell}}{\mu_i}\right) + \rho\phi'\left(\frac{\pi_{i,j} + \sum_{\ell\neq i}\pi_{\ell,j}}{\nu_j}\right).$$

Owing to the fact that ϕ' is α-Hölder continuous with Hölder constant equal to L', we infer that $\partial_{i,j}(\mathrm{UT}_{\phi,\mathrm{c},\rho})_{|B_k}$ is α-Hölder continuous with Hölder constant bounded above by $\frac{2\rho L'}{\min\{\mu_i,\nu_j\}^\alpha}$. Then, given $\gamma^{(1)} = \{\gamma_{i,j}^{(1)}\}_{(i,j)\in B_k}$ and $\gamma^{(2)} = \{\gamma_{i,j}^{(2)}\}_{(i,j)\in B_k}$ we have

$$\begin{aligned}
&\sum_{(i,j)\in B_k} \Big(\partial_{i,j}(\mathrm{UT}_{\phi,\mathrm{c},\rho})_{|B_k}(\gamma^{(1)}) - \partial_{i,j}(\mathrm{UT}_{\phi,\mathrm{c},\rho})_{|B_k}(\gamma^{(2)})\Big)^2 \\
&\qquad \leq \sum_{(i,j)\in B_k} \Big(\frac{2\rho L'}{\min\{\mu_i,\nu_j\}^\alpha}|\gamma_{i,j}^{(1)} - \gamma_{i,j}^{(2)}|^\alpha\Big)^2 \\
&\qquad \leq \Big(\frac{2\rho L'}{\min\{\mu_i,\nu_j\}^\alpha}\Big)^2 \sum_{(i,j)\in B_k} |\gamma_{i,j}^{(1)} - \gamma_{i,j}^{(2)}|^{2\alpha}.
\end{aligned}$$

Finally, we notice that

$$\sum_{(i,j)\in B_k} |\gamma_{i,j}^{(1)} - \gamma_{i,j}^{(2)}|^{2\alpha} = |B_k|\Big(\Big(\sum_{(i,j)\in B_k} |\gamma_{i,j}^{(1)} - \gamma_{i,j}^{(2)}|^{2\alpha} \frac{1}{|B_k|}\Big)^{\frac{1}{\alpha}}\Big)^{\alpha}$$
$$\leq |B_k|^{1-\alpha}\Big(\sum_{(i,j)\in B_k} |\gamma_{i,j}^{(1)} - \gamma_{i,j}^{(2)}|^{2}\Big)^{\alpha}$$
$$\leq N^{1-\alpha}\Big(\sum_{(i,j)\in B_k} |\gamma_{i,j}^{(1)} - \gamma_{i,j}^{(2)}|^{2}\Big)^{\alpha}, \tag{22}$$

Since every block B_k contains at most N variables. We then conclude that

$$\sum_{(i,j)\in B_k} \Big(\partial_{i,j}(\mathrm{UT}_{\phi,\mathrm{c},\rho})_{|B_k}(\gamma^{(1)}) - \partial_{i,j}(\mathrm{UT}_{\phi,\mathrm{c},\rho})_{|B_k}(\gamma^{(2)})\Big)^2$$
$$\leq N^{1-\alpha}\Big(\frac{2\rho L'}{\min_{(i,j)\in B_k}\{\mu_i,\nu_j\}^{\alpha}}\Big)^2\Big(\sum_{(i,j)\in B_k} |\gamma_{i,j}^{(1)} - \gamma_{i,j}^{(2)}|^{2}\Big)^{\alpha},$$

by taking the square root on both sides, we find

$$\|\nabla_{B_k}\mathrm{UT}_{\phi,\mathrm{c},\rho}(\gamma^{(1)}) - \nabla_{B_k}\mathrm{UT}_{\phi,\mathrm{c},\rho}(\gamma^{(2)})\|_2 \leq N^{\frac{1-\alpha}{2}}\frac{2\rho L'}{\min_{(i,j)\in B_k}\{\mu_i,\nu_j\}^{\alpha}}\|\gamma^{(1)} - \gamma^{(2)}\|_2^{\alpha}. \tag{23}$$

Lastly, we recall that

$$L = \frac{2\rho L' N^{\frac{3-\alpha}{2}}}{\min_{i,j\in[N]\,:\,\mu_i,\nu_j>0}\{\mu_i,\nu_j\}^{\alpha}} \tag{24}$$

so that, denoted by $L^\star$ the Hölder constant of $(\mathrm{UT}_{\phi,\mathrm{c},\rho})_{|B_k}$, we have $L^\star \leq \frac{L}{N}$.

Proof 4 (Proof of Theorem 3) *It follows from the fact that for every couple of indices $i,j \in [N]$ it holds $\ell_p(x_i,y_j) \leq T$ and, consequently, $\max_{\ell=1,\dots,d}\{|x_{i_\ell} - y_{j_\ell}|\} \leq T$. Since on regular grids there are less than $(2T)^d$ indices for which $\max_{\ell=1,\dots,d}\{|x_{i_\ell} - y_{j_\ell}|\} \leq T$, we conclude the thesis.*

Proof 5 (Proof of Theorem 4) *For the sake of simplicity, we split the proof into two cases, depending on the value of α. The proof of both cases follows a similar structure and can be summarized into three parts: (i) first, we quantify the objective decrease after each iteration of the MDB method, (ii) second, we show that the quadratic-growth condition (7) implies a Polyak-Łojasiewicz -type inequality, and (iii) finally, we combine the results from the previous two points to retrieve the convergence speed of the MDB method.*

In what follows, we denote by

$$\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi) = \mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi) - \min_{\pi^\star\geq 0\,:\,\pi^\star_{i,j}=0\ (i,j)\notin\mathcal{E}_{c,T}} \mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^\star),$$

and

$$\mathcal{L}_0 := \{\pi\in\Pi\ :\ \mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi) \leq \mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(0)})\}$$

with $\Delta_0 = \mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(0)}) - \mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^\star)$ and $\mathrm{UT}_{\phi,\mathrm{c},\rho}{}^\star := \min_{\pi\geq 0\,:\,\pi_{i,j}=0\ (i,j)\notin\mathcal{E}_{c,T}} \mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi)$.

***Case** $\alpha = 1$. We start by noticing that our function has Lipschitz continuous gradient in this case.*

***Part 1.** We now show that at every step we can quantify the decrease in the objective function in terms of $\mathcal{H}^{(i,j)}(\pi^{(t)}, L^\star)$. Given a good partition of the coordinates $\mathcal{B}$, let us assume that block B_k is selected at iterate t. Owing to Algorithm 1, given a block B_k and an iterate $\pi^{(t)}$ of the MDB, the $(t+1)$-iterate of the MDB is given by*

$$\pi_{i,j}^{(t+1)} := \pi_{i,j}^{(t)} + \operatorname*{arg\,min}_{\pi_{i,j}^{(t)}+\gamma_{i,j}\geq 0}\Big(\partial_{i,j}\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)})\gamma_{i,j} + \frac{L^\star}{2}|\gamma_{i,j}|^2\Big) \tag{25}$$

for every $(i,j) \in B_k$, and $\pi_{i,j}^{(t+1)} = \pi_{i,j}^{(t)}$ for every $(i,j) \in B_k^c$. Since ϕ has Lipschitz continuous derivative, by the definition of the update rule in (25) *and the Descent Lemma, we get*

$$\begin{aligned}\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t+1)}) - \mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)}) &\leq \sum_{(i,j)\in B_k} \partial_{i,j}\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)})\big(\pi_{i,j}^{(t+1)} - \pi_{i,j}^{(t)}\big) \\ &\quad + \frac{L^\star}{2}\sum_{(i,j)\in B_k}\left|\pi_{i,j}^{(t+1)} - \pi_{i,j}^{(t)}\right|^2 \\ &= -\frac{1}{2L^\star}\mathcal{H}^{(k)}(\pi^{(t)}, L^\star), \end{aligned} \tag{26}$$

where we have used the fact that

$$\mathcal{H}^{(k)}(\pi, L^\star) = -2L^\star \min_{\{\pi_{i,j}+\gamma_{i,j}\geq 0\}_{(i,j)\in B_k}} \sum_{(i,j)\in B_k}\left(\partial_{i,j}\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi)\gamma_{i,j} + \frac{L^\star}{2}|\gamma_{i,j}|^2\right).$$

Finally, by rearranging (26)*, we get*

$$\frac{1}{2L^\star}\mathcal{H}^{(k)}(\pi^{(t)}, L^\star) \leq \mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)}) - \mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t+1)}), \tag{27}$$

for every block. Notice that this condition also ensures that $\pi^{(t)}$ belongs to $\mathcal{L}_0$ for every $t \geq 0$.

Part 2. *Next, we show that quadratic growth condition implies the following Polyak-Łojasiewicz condition on* $\mathrm{UT}_{\phi,\mathrm{c},\rho}$

$$\eta\Big(\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi) - \mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^\star)\Big) \leq \frac{1}{2L^\star}\sum_{k\in[K]}\mathcal{H}^{(k)}(\pi, L^\star), \tag{28}$$

where $\eta > 0$. From our hypothesis, there exists a constant $\tilde{\mu} > 0$ such that

$$\mathrm{dist}(\pi, \Pi^\star)^2 \leq \tilde{\mu}\big(\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi) - \mathrm{UT}_{\phi,\mathrm{c},\rho}{}^\star\big) \qquad \forall \pi \in \mathcal{L}_0, \tag{29}$$

where by $\Pi^\star$ we denote the set of optimal solutions for our problem and $\mathrm{dist}(\pi, \Pi^\star) := \min_{\bar{\pi}\in\Pi^\star}\|\pi - \bar{\pi}\|_2$. Let $\pi^\star \in \Pi^\star$ be an optimal positive measure for which

$$\mathrm{dist}(\pi, \Pi^\star) = \|\pi - \pi^\star\|_2.$$

For any $\beta \in [0,1]$, let us now consider the direction

$$\gamma = \beta(\pi^\star - \pi),$$

which is a feasible direction, due to the convexity of the feasible set. Hence, by the definition of $\mathcal{H}^{(k)}(\pi, L^\star)$, we have

$$\frac{1}{2L^\star}\sum_{k\in[K]}\mathcal{H}^{(k)}(\pi, L^\star) \geq \beta\,\langle \nabla\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi), \pi - \pi^\star\rangle - \frac{L^\star}{2}\beta^2\|\pi - \pi^\star\|_2^2, \tag{30}$$

for every $\beta \in [0,1]$. Owing to the convexity of $\mathrm{UT}_{\phi,\mathrm{c},\rho}$

$$\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi) - \mathrm{UT}_{\phi,\mathrm{c},\rho}{}^\star \leq \langle \nabla\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi), \pi - \pi^\star\rangle \tag{31}$$

and the growth condition,

$$\|\pi - \pi^\star\|_2^2 = \mathrm{dist}(\pi, \Pi^\star)^2 \leq \tilde{\mu}\Big(\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi) - \mathrm{UT}_{\phi,\mathrm{c},\rho}{}^\star\Big), \tag{32}$$

if we substitute (32) *and* (31) *into* (30)*, we obtain*

$$\frac{1}{2L^\star}\sum_{k\in[K]}\mathcal{H}^{(k)}(\pi, L^\star) \geq \left(\beta - \frac{L^\star\tilde{\mu}}{2}\beta^2\right)\Big(\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi) - \mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^\star)\Big).$$

If $\frac{1}{L^\star\tilde{\mu}} \leq 1$, we set $\beta = \frac{1}{L^\star\tilde{\mu}}$ and infer that

$$\frac{1}{2L^\star}\sum_{k\in[K]}\mathcal{H}^{(k)}(\pi, L^\star) \geq \frac{1}{2L^\star\tilde{\mu}}\Big(\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi) - \mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^\star)\Big).$$

If $\frac{1}{L^\star\tilde{\mu}} > 1$*, that is* $L^\star\tilde{\mu} < 1$*, by setting* $\beta = 1$ *we infer*

$$\frac{1}{2L^\star}\sum_{k\in[K]}\mathcal{H}^{(k)}(\pi, L^\star) \geq \left(1 - \frac{L^\star\tilde{\mu}}{2}\right)\Big(\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi) - \mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^\star)\Big)$$
$$\geq \frac{1}{2}\Big(\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi) - \mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^\star)\Big).$$

Putting everything together, we infer that

$$\min\left\{\frac{1}{2L^\star\tilde{\mu}}, \frac{1}{2}\right\}\Big(\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi) - \mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^\star)\Big) \leq \frac{1}{2L^\star}\sum_{k\in[K]}\mathcal{H}^{(k)}(\pi, L^\star).$$

This proves the projected Polyak-Łojasiewicz inequality (28)*, with* $\eta = \min\left\{\frac{1}{2L^\star\tilde{\mu}}, \frac{1}{2}\right\}$.

Part 3. *We now put together the two previous parts to retrieve the convergence speed of the method. For the sake of simplicity, we split the proof depending on whether we consider the Randomized MDB or the Gauss-Southwell MDB.*

First, we consider the Randomized MDB. By taking the average on both sides of (27) *over the* K *possible blocks from which we can choose, we get*

$$\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)}) - \mathbb{E}_t\big[\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t+1)})\big] \geq \frac{1}{2L^\star K}\sum_{k\in[K]}\mathcal{H}^{(k)}(\pi^{(t)}, L^\star), \tag{33}$$

where $\mathbb{E}_t$ *is expected value conditioned to* $\pi^{(t)}$*, which is uniform over* $[K]$*. By adding and subtracting* $\mathrm{UT}_{\phi,\mathrm{c},\rho^\star}$ *in the right hand side and by combining the resulting inequality with the projected Polyak-Łojasiewicz inequality we get*

$$\eta\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)}) \leq K\left(\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)}) - \mathbb{E}_\xi[\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t+1)})]\right). \tag{34}$$

Equivalently,

$$\mathbb{E}_\xi[\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t+1)})] \leq \left(1 - \frac{\eta}{K}\right)\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)}). \tag{35}$$

By iterating the argument and noticing that $\frac{\eta}{K} < 1$ *since* $K \geq 1$ *and* $\eta \leq \frac{1}{2}$*, we conclude the proof for the Randomized MDB.*

We then consider the Gauss-Southwell MDB. First, we observe that

$$\frac{1}{K}\sum_{k\in[K]}\mathcal{H}^{(k)}(\pi^{(t)}, L^\star) \leq \max_{k\in[K]}\mathcal{H}^{(k)}(\pi^{(t)}, L^\star). \tag{36}$$

By the sufficient decrease inequality,

$$\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)}) - \mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t+1)}) \geq \frac{1}{2L^\star}\mathcal{H}^{(k)}(\pi^{(t)}, L^\star).$$

Using (36)*, we have*

$$\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)}) - \mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t+1)}) \geq \frac{1}{2L^\star}\mathcal{H}^{(k)}(\pi^{(t)}, L^\star) \geq \frac{1}{2L^\star K}\sum_{k\in[K]}\mathcal{H}^{(k)}(\pi^{(t)}, L^\star).$$

By the Polyak-Łojasiewicz inequality, we have

$$\eta\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)}) \leq K\left(\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)}) - \Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t+1)})\right). \tag{37}$$

Rearranging the terms, we obtain

$$\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t+1)}) \leq \left(1 - \frac{\eta}{K}\right)\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)}), \tag{38}$$

which allows us to conclude the proof as done per the R-MDB.

Case $0 < \alpha < 1$***.*** *We now consider the case in which α is positive and strictly less than 1. In this case the gradient is α-Hölder continuous.*

Part 1. *First, we show that at every step we can quantify the decrease in the objective function in terms of $\mathcal{H}^{(i,j)}(\pi^{(t)}, L^\star)$. Given a good partition of the coordinates $\mathcal{B}$, let us assume that block B_k is selected at iterate t. Owing to Algorithm 1, given a block B_k and an iterate $\pi^{(t)}$ of the MDB, the $(t+1)$-iterate of the MDB is given by*

$$\pi_{i,j}^{(t+1)} := \pi_{i,j}^{(t)} + \underset{\pi_{i,j}^{(t)}+\gamma_{i,j}\geq 0}{\arg\min} \left(\partial_{i,j} \operatorname{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)})\gamma_{i,j} + \frac{L^\star}{1+\alpha}|\gamma_{i,j}|^{1+\alpha} \right) \tag{39}$$

for every $(i,j) \in B_k$, and $\pi_{i,j}^{(t+1)} = \pi_{i,j}^{(t)}$ for every $(i,j) \in B_k^c$. Since ϕ has α-Hölder continuous derivative, by the definition of the update rule in (39) *and the Descent Lemma for functions with α Hölder continuous gradient, we get*

$$\begin{aligned} \operatorname{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t+1)}) - \operatorname{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)}) &\leq \sum_{(i,j)\in B_k} \partial_{i,j} \operatorname{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)})\big(\pi_{i,j}^{(t+1)} - \pi_{i,j}^{(t)}\big) \\ &\quad + \frac{L^\star}{1+\alpha} \sum_{(i,j)\in B_k} \left|\pi_{i,j}^{(t+1)} - \pi_{i,j}^{(t)}\right|^{1+\alpha} \\ &= -\frac{1}{(1+\alpha)L^\star}\mathcal{H}^{(k)}(\pi^{(t)}, L^\star). \end{aligned} \tag{40}$$

Finally, by rearranging (40)*, we get*

$$\frac{1}{(1+\alpha)L^\star}\mathcal{H}^{(k)}(\pi^{(t)}, L^\star) \leq \operatorname{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)}) - \operatorname{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t+1)}), \tag{41}$$

for every block. Notice that this condition also ensures that $\pi^{(t)}$ belongs to $\mathcal{L}_0$ for every $t \geq 0$.

Part 2. *We now show that the quadratic-growth condition* (7) *implies the following nonlinear Polyak-Łojasiewicz -type inequality*

$$\eta_\alpha \big(\Delta \operatorname{UT}_{\phi,\mathrm{c},\rho}(\pi)\big)^{\frac{1+\alpha}{\alpha}} \leq \frac{1}{(1+\alpha)L^\star} \sum_{k\in[K]} \mathcal{H}^{(k)}(\pi, L^\star) \tag{42}$$

for a suitable $\eta_\alpha > 0$.

Let $\pi \in \mathcal{L}_0$, and let $\pi^\star \in \Pi^\star$ be an optimal positive measure for which

$$\operatorname{dist}_2(\pi, \Pi^\star) = \min_{\pi'\in\Pi^\star} \|\pi - \pi'\|_2 = \|\pi - \pi^\star\|_2.$$

For any $\beta \in [0,1]$, consider the direction

$$\gamma = \beta(\pi^\star - \pi),$$

which is a feasible direction owing to the convexity of the feasible set. Hence, by the definition of $\mathcal{H}^{(k)}(\pi, L^\star)$, for every $\beta \in [0,1]$ we have

$$\begin{aligned} \frac{1}{(1+\alpha)L^\star} \sum_{k\in[K]} \mathcal{H}^{(k)}(\pi, L^\star) &\geq \beta \left\langle \nabla \operatorname{UT}_{\phi,\mathrm{c},\rho}(\pi), \pi - \pi^\star \right\rangle - \frac{L^\star}{1+\alpha}\beta^{1+\alpha}\|\pi - \pi^\star\|_{1+\alpha}^{1+\alpha} \\ &\geq \beta \left\langle \nabla \operatorname{UT}_{\phi,\mathrm{c},\rho}(\pi), \pi - \pi^\star \right\rangle - \frac{L^\star c_\alpha^{1+\alpha}}{1+\alpha}\beta^{1+\alpha}\|\pi - \pi^\star\|_2^{1+\alpha}, \end{aligned} \tag{43}$$

where c_α is the equivalence constant between $\|\cdot\|_2$ and $\|\cdot\|_{1+\alpha}$. Owing to the convexity of $\operatorname{UT}_{\phi,\mathrm{c},\rho}$*,*

$$\operatorname{UT}_{\phi,\mathrm{c},\rho}(\pi) - \operatorname{UT}_{\phi,\mathrm{c},\rho}(\pi^\star) \leq \left\langle \nabla \operatorname{UT}_{\phi,\mathrm{c},\rho}(\pi), \pi - \pi^\star \right\rangle \tag{44}$$

and to the quadratic growth condition (7)

$$\|\pi - \pi^\star\|_2^{1+\alpha} = \big(\|\pi - \pi^\star\|_2^2\big)^{\frac{1+\alpha}{2}} \leq \tilde{\mu}^{\frac{1+\alpha}{2}} \Big(\operatorname{UT}_{\phi,\mathrm{c},\rho}(\pi) - \operatorname{UT}_{\phi,\mathrm{c},\rho}(\pi^\star)\Big)^{\frac{1+\alpha}{2}} \tag{45}$$

we obtain

$$\frac{1}{(1+\alpha)L^\star}\sum_{k\in[K]}\mathcal{H}^{(k)}(\pi,L^\star)\geq\beta\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi)-\frac{L^\star c_\alpha^{1+\alpha}}{1+\alpha}\tilde{\mu}^{\frac{1+\alpha}{2}}\beta^{1+\alpha}\big(\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi)\big)^{\frac{1+\alpha}{2}}. \quad (46)$$

Let $A:=L^\star c_\alpha^{1+\alpha}\tilde{\mu}^{\frac{1+\alpha}{2}}$. *By taking the derivative of the right-hand side of* (46) *with respect to* β, *we find that its unconstrained maximum is attained at*

$$\beta=A^{-\frac{1}{\alpha}}\big(\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi)\big)^{\frac{1-\alpha}{2\alpha}}. \quad (47)$$

If $\beta\leq 1$, *substituting* (47) *into* (46), *we obtain*

$$\frac{1}{(1+\alpha)L^\star}\sum_{k\in[K]}\mathcal{H}^{(k)}(\pi,L^\star)\geq\frac{\alpha}{1+\alpha}\frac{\big(\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi)\big)^{\frac{1+\alpha}{2\alpha}}}{(L^\star)^{1/\alpha}\tilde{\mu}^{\frac{1+\alpha}{2\alpha}}c_\alpha^{\frac{1+\alpha}{\alpha}}}. \quad (48)$$

If $\beta>1$, *that is* $\big(\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi)\big)^{\frac{1-\alpha}{2\alpha}}>A^{\frac{1}{\alpha}}$ *and thus* $\big(\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi)\big)^{\frac{1-\alpha}{2}}>A$, *we set* $\beta=1$ *and attain*

$$\begin{aligned}\frac{1}{(1+\alpha)L^\star}\sum_{k\in[K]}\mathcal{H}^{(k)}(\pi,L^\star)&\geq\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi)-\frac{A}{1+\alpha}\big(\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi)\big)^{\frac{1+\alpha}{2}}\\&=\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi)\Big(1-\frac{A}{1+\alpha}\big(\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi)\big)^{\frac{\alpha-1}{2}}\Big)\\&\geq\frac{\alpha}{1+\alpha}\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi)\\&=\frac{\alpha}{1+\alpha}\frac{\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi)^{\frac{1+\alpha}{2\alpha}}}{\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi)^{\frac{1-\alpha}{2\alpha}}}\\&\geq\left(\frac{\alpha}{1+\alpha}\Delta_0^{\frac{\alpha-1}{2\alpha}}\right)\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi)^{\frac{1+\alpha}{2\alpha}},\end{aligned}$$

since $\pi\in\mathcal{L}_0$ *and hence* $\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi)\leq\Delta_0$. *Proving* (42) *with*

$$\eta_\alpha=\frac{\alpha}{1+\alpha}\min\big\{(L^\star c_\alpha^{1+\alpha}\tilde{\mu}^{\frac{1+\alpha}{2}})^{-\frac{1}{\alpha}},\Delta_0^{\frac{\alpha-1}{2\alpha}}\big\}.$$

Part 3. *We now combine the previous two steps to attain the convergence estimate. For the sake of clarity, we split the proof depending on whether we consider the Randomized MDB or the Gauss–Southwell MDB.*

First, we consider the Randomized MDB. By taking the average over the K *possible blocks from which we can choose, we get*

$$\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)})-\mathbb{E}_t\big[\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t+1)})\big]\geq\frac{1}{(1+\alpha)L^\star K}\sum_{k\in[K]}\mathcal{H}^{(k)}(\pi^{(t)},L^\star), \quad (49)$$

where $\mathbb{E}_t$ *denotes the expected value conditioned with respect to the previous iterate* $\pi^{(t)}$. *Combining this inequality with* (42) *yields*

$$\mathbb{E}_t\left[\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t+1)})\right]\leq\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)})-\frac{\eta_\alpha}{K}\big(\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)})\big)^{\frac{1+\alpha}{2\alpha}}. \quad (50)$$

Let

$$u_t:=\mathbb{E}\left[\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)})\right].$$

Since $\frac{1+\alpha}{2\alpha}>1$, *if we take the expected value with respect to* $\pi^{(t)}$ *on both sides and apply Jensen's inequality along with* (50), *we get*

$$u_{t+1}\leq u_t-\frac{\eta_\alpha}{K}u_t^{\frac{1+\alpha}{2\alpha}}. \quad (51)$$

Let us now consider the Gauss–Southwell MDB. First, we observe that

$$\frac{1}{K}\sum_{k\in[K]}\mathcal{H}^{(k)}(\pi^{(t)},L^\star)\le\max_{k\in[K]}\mathcal{H}^{(k)}(\pi^{(t)},L^\star). \tag{52}$$

Combining (52) *with the sufficient decrease inequality* (41)*, we have*

$$\begin{aligned}\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)})-\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t+1)})&\ge\frac{1}{(1+\alpha)L^\star}\mathcal{H}^{(k)}(\pi^{(t)},L^\star)\\&\ge\frac{1}{(1+\alpha)L^\star K}\sum_{k\in[K]}\mathcal{H}^{(k)}(\pi^{(t)},L^\star).\end{aligned}$$

By the projected Polyak-Łojasiewicz -type inequality (42)*, we obtain*

$$\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t+1)})\le\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)})-\frac{\eta_\alpha}{K}\big(\Delta\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi^{(t)})\big)^{\frac{1+\alpha}{2\alpha}}. \tag{53}$$

Finally, let us set $r_\alpha:=\frac{1-\alpha}{2\alpha}>0$*. Since* $\frac{1+\alpha}{2\alpha}=1+r_\alpha$*, both* (51) *and* (53) *have the form*

$$z_{t+1}\le z_t-\frac{\eta_\alpha}{K}z_t^{1+r_\alpha}.$$

The standard recursion estimate (Bredies, 2009) therefore gives

$$z_{t+1}\le\Big(z_0^{-r_\alpha}+\frac{r_\alpha\eta_\alpha}{K}(t+1)\Big)^{-1/r_\alpha},$$

which, combined with $c_\alpha=m^{\frac{1-\alpha}{2(1+\alpha)}}$*, where* m *is the number of variables in* $\mathcal{E}_{c,T}$*, concludes the proof.*

A.2 Derivation of the Hölder constant for the TSUOT

In this section, we recover the Hölder constant for the objective function of the TSUOT under the assumptions of Theorem 3, so that for each i there are at most T^d indices j for which $(i,j)\in\mathcal{E}_{c,T}$. Let us consider two feasible points for the TSUOT problem, namely $\{\pi'_{i,j}\}_{(i,j)\in\mathcal{E}_{c,T}}$ and $\{\pi_{i,j}\}_{(i,j)\in\mathcal{E}_{c,T}}$. Denoted by R_i the set of indices ℓ for which $(i,\ell)\in\mathcal{E}_{c,T}$ and by R_j the set of indices ℓ for which $(\ell,j)\in\mathcal{E}_{c,T}$, we have

$$\begin{aligned}&\sum_{(i,j)\in\mathcal{E}_{c,T}}|\partial_{i,j}\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi')-\partial_{i,j}\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi)|\\&\qquad\le\Big(\frac{2\rho L'}{\tau^\alpha}\Big)^2\sum_{(i,j)\in\mathcal{E}_{c,T}}\left(\Big|\sum_{\ell\in R_i}(\pi'_{i,\ell}-\pi_{i,\ell})\Big|^\alpha+\Big|\sum_{\ell\in R_j}(\pi'_{\ell,j}-\pi_{\ell,j})\Big|^\alpha\right)^2.\end{aligned}$$

Denoted by $K=\frac{2\rho L'}{\tau^\alpha}$, we have that

$$\begin{aligned}&\sum_{(i,j)\in\mathcal{E}_{c,T}}|\partial_{i,j}\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi')-\partial_{i,j}\,\mathrm{UT}_{\phi,\mathrm{c},\rho}(\pi)|^2\\&\qquad\le 2K^2\sum_{(i,j)\in\mathcal{E}_{c,T}}\left(\Big|\sum_{\ell\in R_i}(\pi'_{i,\ell}-\pi_{i,\ell})\Big|^{2\alpha}+\Big|\sum_{\ell\in R_j}(\pi'_{\ell,j}-\pi_{\ell,j})\Big|^{2\alpha}\right)\\&\qquad\le 2K^2\sum_{(i,j)\in\mathcal{E}_{c,T}}\left(|R_i|^\alpha\Big|\sum_{\ell\in R_i}(\pi'_{i,\ell}-\pi_{i,\ell})^2\Big|^\alpha+|R_j|^\alpha\Big|\sum_{\ell\in R_j}(\pi'_{\ell,j}-\pi_{\ell,j})\Big|^{2\alpha}\right)\\&\qquad\le 2K^2T^{\alpha d}\sum_{(i,j)\in\mathcal{E}_{c,T}}\left(\Big|\sum_{\ell\in R_i}(\pi'_{i,\ell}-\pi_{i,\ell})^2\Big|^\alpha+\Big|\sum_{\ell\in R_j}(\pi'_{\ell,j}-\pi_{\ell,j})\Big|^{2\alpha}\right)\\&\qquad\le 2K^2T^{\alpha d}\sum_{(i,j)\in\mathcal{E}_{c,T}}\left(\sum_{\ell\in R_i}|\pi'_{i,\ell}-\pi_{i,\ell}|^{2\alpha}+\sum_{\ell\in R_j}|\pi'_{\ell,j}-\pi_{\ell,j}|^{2\alpha}\right)\\&\qquad\le 4K^2T^{(1+\alpha)d}\sum_{(i,j)\in\mathcal{E}_{c,T}}|\pi'_{i,\ell}-\pi_{i,\ell}|^{2\alpha},\end{aligned}$$

where we have used the fact that $\max\{|R_i|, |R_j|\} \leq T^d$. To conclude, we notice that

$$\sum_{(i,j)\in\mathcal{E}_{c,T}} |\pi'_{i,\ell} - \pi_{i,j}|^{2\alpha} \leq |\mathcal{E}_{c,T}|^{1-\alpha} \Big(\sum_{(i,j)\in\mathcal{E}_{c,T}} |\pi'_{i,\ell} - \pi_{i,j}|^2 \Big)^{\alpha} \tag{54}$$

$$= (nT)^{(1-\alpha)d} \Big(\sum_{(i,j)\in\mathcal{E}_{c,T}} |\pi'_{i,j} - \pi_{i,j}|^2 \Big)^{\alpha}. \tag{55}$$

We thus infer that

$$\sum_{(i,j)\in\mathcal{E}_{c,T}} |\partial_{i,j} \operatorname{UT}_{\phi,\mathrm{c},\rho}(\pi') - \partial_{i,j} \operatorname{UT}_{\phi,\mathrm{c},\rho}(\pi)|^2$$

$$\leq 4K^2 n^{(1-\alpha)d} T^{2d} \Big(\sum_{(i,j)\in\mathcal{E}_{c,T}} |\pi'_{i,j} - \pi_{i,j}|^2 \Big)^{\alpha},$$

which allows us to conclude the proof by taking the square root on both sides.

A.3 Hardware.

The experiments in Section 4.3 were run on a workstation equipped with Intel(R) Xeon(R), one NVIDIA RTX A5000 GPU (24 GB), and 60 GB of system RAM. All remaining experiments were run on an Intel Xeon E5-2680 CPU at 2.70 GHz with 96 GB of system RAM. The code is available in the supplementary material.

A.4 Image Transfer

Following Rubner et al. (2000), we cast the comparison of two grayscale images as an optimal transport problem: each image is treated as a positive measure supported on its pixel grid, with the mass at every pixel proportional to its gray intensity. Given a pair of images, we solve the TSUOT problem between their two measures and obtain an optimal plan π together with its cost. The plan itself has a direct interpretation: entry π_{ij} is the amount of gray-level mass moved from pixel x_i of the source image to pixel y_j of the target image, so π describes the best way, under the chosen ground cost and marginal penalty, to redistribute one image's intensity pattern onto the other's.

R-MDB and GS-MDB share the parameters T, p, ρ, and the image resolution, which fixes the problem scale. Table 3 reports time and cost precision across resolutions $\{32, 64, 128, 256, 512\}$, $p \in \{1.1, 1.5\}$, and $T \in \{1, \ldots, 4\}$, keeping $\rho = 1.0$. For the stopping rule, every run is stopped once it reaches `max_iter` iterations (default 10^5, large enough in practice never to bind), or once the relative change of the block updated at each iteration has stayed below `tol` (default 10^{-2}) for a window of $10K$ iterations, with K being the number of blocks, large enough that R-MDB's random selection has, in expectation, visited every block at least once within the window. GS-MDB additionally stops as soon as the largest projected-gradient ℓ_2 norm over the K blocks drops below the same `tol`.

The cost gap of a single image pair k is the difference between the cost found, c_k, and the best cost found for that pair, $c_k^\star$, normalized by c_k: gap $= (c_k - c_k^\star)/c_k$. The reported cost gap is averaged over the 45 image pairs, with $c_k^\star$ computed with $T = \infty$. Both variants always reach the same cost, confirming that they converge to approximately the same objective value. Although R-MDB avoids computing the full gradient at every step, it is not uniformly faster than GS-MDB: random block selection reduces the cost per iteration, whereas the more informed GS selection may require fewer iterations, resulting in configuration-dependent runtimes.

The pruning radius T trades off time against accuracy: increasing T from 1 to 4 raises the running time, since larger blocks admit more arcs, but also closes the gap to the best cost found among the tested radii, which is already below 0.01% at $T = 4$, for every resolution and both values of p. Even the smallest radius, $T = 1$, already keeps the gap under 5% in every configuration (the largest, 4.94%, at resolution 64 and $p = 1.1$), and under 2% in most others (12 of the 20 tested configurations). This confirms that pruning is especially effective on this benchmark, where the two measures share the same regularly-spaced support: a small T discards only long-range, low-probability arcs, so little accuracy is sacrificed for a large reduction in computation. We stress that

Table 3: MicroscopyImages: R-MDB vs. GS-MDB across resolutions, $p \in \{1.1, 1.5\}$, and truncation radius T. Time and gap are reported as mean $\pm$ standard deviation across the 45 image pairs of each (size, p) block. Gap is calculated as described in the text with respect to the unpruned cost achievable at $T = \infty$, so that 0% denotes the best cost.

		$p = 1.1$				$p = 1.5$			
		R-MDB		GS-MDB		R-MDB		GS-MDB	
Size	T	Time (s)	Gap (%)	Time (s)	Gap (%)	Time (s)	Gap (%)	Time (s)	Gap (%)
32	1	1.58 ± 0.21	1.26 ± 0.70	2.21 ± 0.79	1.26 ± 0.70	1.52 ± 0.22	0.47 ± 0.24	2.15 ± 0.76	0.47 ± 0.24
	2	2.13 ± 0.24	0.06 ± 0.07	2.71 ± 0.89	0.07 ± 0.07	2.11 ± 1.08	0.03 ± 0.05	2.65 ± 0.99	0.03 ± 0.05
	3	3.29 ± 0.43	0.01 ± 0.02	3.23 ± 0.88	0.01 ± 0.02	2.83 ± 0.32	0.02 ± 0.06	3.10 ± 0.98	0.01 ± 0.04
	4	4.64 ± 0.58	0.00 ± 0.01	4.52 ± 0.91	0.01 ± 0.01	3.89 ± 0.49	0.01 ± 0.04	3.80 ± 0.86	0.01 ± 0.04
64	1	2.01 ± 0.32	4.94 ± 1.69	2.46 ± 0.78	4.94 ± 1.69	1.81 ± 0.27	1.55 ± 0.55	2.39 ± 0.79	1.55 ± 0.55
	2	3.16 ± 0.72	0.59 ± 0.39	3.21 ± 0.83	0.59 ± 0.39	2.65 ± 0.58	0.06 ± 0.05	2.59 ± 0.73	0.06 ± 0.05
	3	5.95 ± 1.88	0.03 ± 0.03	5.68 ± 1.34	0.03 ± 0.03	4.69 ± 1.64	0.01 ± 0.01	4.89 ± 3.94	0.01 ± 0.01
	4	8.86 ± 2.86	0.00 ± 0.00	8.69 ± 3.27	0.00 ± 0.00	7.00 ± 2.19	0.00 ± 0.02	7.74 ± 5.16	0.01 ± 0.02
128	1	6.14 ± 1.97	3.21 ± 1.61	4.62 ± 3.32	3.21 ± 1.61	5.46 ± 1.63	0.96 ± 0.51	3.96 ± 0.92	0.96 ± 0.51
	2	12.92 ± 3.85	0.24 ± 0.22	9.17 ± 2.50	0.24 ± 0.22	11.06 ± 5.21	0.02 ± 0.02	7.30 ± 1.86	0.02 ± 0.02
	3	26.52 ± 7.88	0.01 ± 0.01	22.44 ± 8.16	0.01 ± 0.01	21.81 ± 10.86	0.00 ± 0.00	13.88 ± 3.55	0.00 ± 0.00
	4	40.80 ± 18.15	0.00 ± 0.00	35.60 ± 10.97	0.00 ± 0.00	31.47 ± 13.14	0.00 ± 0.00	20.69 ± 6.23	0.00 ± 0.00
256	1	21.10 ± 6.13	2.42 ± 1.20	11.49 ± 3.39	2.42 ± 1.20	17.56 ± 5.70	0.68 ± 0.37	8.88 ± 2.09	0.68 ± 0.37
	2	41.78 ± 11.87	0.16 ± 0.13	28.17 ± 6.94	0.16 ± 0.13	38.67 ± 18.72	0.01 ± 0.01	23.27 ± 7.54	0.01 ± 0.01
	3	92.05 ± 24.24	0.00 ± 0.00	65.78 ± 17.13	0.01 ± 0.00	69.79 ± 23.76	0.00 ± 0.00	43.83 ± 12.44	0.00 ± 0.00
	4	138.95 ± 38.09	0.00 ± 0.00	128.86 ± 34.33	0.00 ± 0.00	102.04 ± 28.25	0.00 ± 0.00	82.13 ± 28.40	0.00 ± 0.00
512	1	62.99 ± 18.02	2.13 ± 1.11	48.97 ± 13.85	2.13 ± 1.11	49.87 ± 13.36	0.61 ± 0.37	47.14 ± 15.95	0.61 ± 0.37
	2	153.47 ± 41.43	0.17 ± 0.14	198.87 ± 68.66	0.17 ± 0.14	123.57 ± 34.86	0.01 ± 0.02	168.56 ± 64.49	0.01 ± 0.02
	3	330.76 ± 86.38	0.00 ± 0.01	557.07 ± 214.98	0.00 ± 0.01	258.70 ± 69.60	0.00 ± 0.00	453.48 ± 160.30	0.00 ± 0.00
	4	531.62 ± 147.36	0.00 ± 0.00	1076.57 ± 478.31	0.00 ± 0.00	425.45 ± 117.11	0.00 ± 0.00	862.26 ± 301.60	0.00 ± 0.00

Table 4: MicroscopyImages: assessing the effect of the proposed MDB technique, comparing the convergence times for R-MDB and GS-MDB with R-BCD, GS-BCD, RowBlock-R, and RowBlock-GS for size 128, $p = 2$, and different truncation radii T. We report the mean $\pm$ standard deviation, over the 45 image pairs, of the computation time (s) needed to reach a cost within 0.01% of the best cost found for each pair.

	T	R-MDB	GS-MDB	R-BCD	GS-BCD	RowBlock-R	RowBlock-GS
128	1	1.47 ± 0.22	1.90 ± 0.23	31.71 ± 23.54	39.05 ± 18.04	9.28 ± 4.61	21.30 ± 9.00
	2	1.82 ± 0.32	3.29 ± 0.83	175.92 ± 134.68	128.70 ± 53.91	16.97 ± 8.81	50.66 ± 22.10
	3	2.84 ± 0.51	4.42 ± 0.65	770.50 ± 643.10	257.44 ± 110.97	27.63 ± 14.18	76.70 ± 36.21
	4	3.82 ± 0.87	9.22 ± 2.03	2336.57 ± 2016.66	417.88 ± 193.68	46.54 ± 29.61	126.73 ± 63.38

this favorable trade-off is data-dependent: it is most pronounced when the cost structure is regular, as on a grid, or when a physical constraint already bounds how far mass can move, as in the Change Detection application.

Table 4 extends the comparison performed in Figure 1 to $T \in \{1, \dots, 4\}$, reporting the cost gap to the optimum over time for R-MDB, GS-MDB, and two non-MDB partitions: Coordinate Descent (BCD), whose blocks are single coordinates, and BlockRow, whose blocks each group one source point with all its reachable targets. Note that we implement only this row-based variant because the column-based analogue, which would consist in grouping each target point with all the sources that reach it, would be the same algorithm with the roles of μ and ν swapped. Both alternatives reuse MDB's per-entry update rule where possible, but differ in what their blocks allow. BCD's blocks contain a single entry, so, like MDB's, they carry no internal coupling: each update is the same exact closed-form (for $p = 1, 2$) or Newton step used by MDB, only applied to one entry at a time rather than to an entire conflict-free block in parallel. Coordinate selection is performed as in MDB. BlockRow's blocks have no such guarantee: the block's own entries all share the same row marginal, so they remain mutually coupled and the block subproblem has no closed-form solution in general. Each BlockRow update is instead a single projected-gradient step. All methods share the same T, p, ρ as R-MDB/GS-MDB, and the same default tolerance 10^{-2} for their own internal stopping rule.

The table reports the time each method needs to reach a cost within 0.01% of the optimum. The ranking matches the one observed at $T = 4$: R-MDB is the fastest, followed by GS-MDB, while BCD and BlockRow are between $6\times$ and over $600\times$ slower. The gap widens with T for the Random-selection variants, from $21\times$ to $612\times$ for BCD and from $6\times$ to $12\times$ for BlockRow.

A.5 CHANGE DETECTION

Detailed setting. The hyper-parameters of all methods were selected via grid search on scene *LyonS* of *Lidar05*, and the configuration achieving the best Mean Intersection Over Union (mIoU) was then used on all five test settings to obtain Table 2 (the grids with the tuning parameters are detailed in the code provided as supplementary material). We point out that mIoU is computed over the two change classes only (New Building, Destroyed) because unchanged points make up the vast majority of any scene, so including them would inflate the score without reflecting what change detection is actually meant to measure. For R-MDB and GS-MDB, we set $T = 2$ m, $\rho = 1$, $p = 2$, a tolerance of 10^{-2}, and at most 10^6 iterations. For the baselines, we use $\varepsilon = 3$ and $\tau = 0.9$ for Sink; `reg` $= 2$ and `reg_m` $= 20$ for SinkTI; $\varepsilon = 10^{-2}$, $\tau = 0.99$, rank 500 and $\gamma = 10^3$ for SinkLR; `reg_m` $= 10$ with the KL divergence for MM; and normal radius 500, cylinder radius 2 and maximum distance 100 for M3C2. Each algorithm also has to tune a threshold parameter h used to classify the points. Points are classified according to how much their height z differs from that of the source cloud. We compute the signed squared displacement $s_j = \|y_j - \hat{y}_j\|_2^2 \operatorname{sign}(y_{j,z} - \hat{y}_{j,z})$ between each target point y_j and its barycentric projection $\hat{y}_j$: a point is labeled as changed if $|s_j|$ exceeds the threshold h, and as a new building ($s_j > 0$) or demolition ($s_j < 0$) according to the sign.

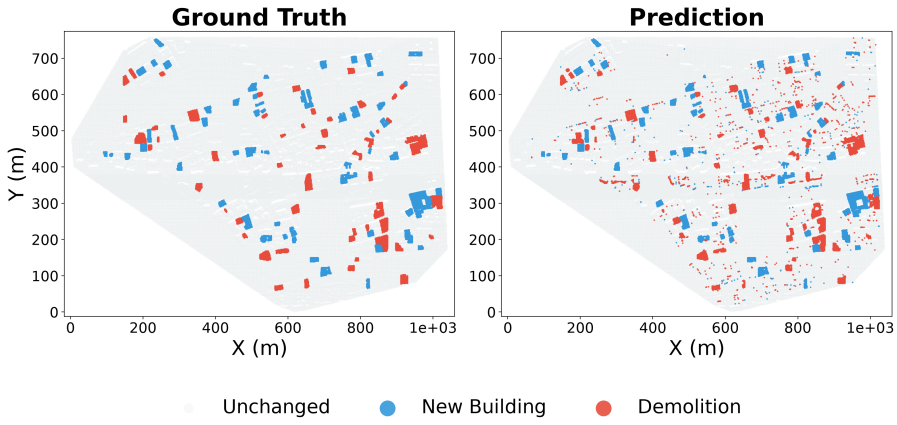
The point clouds are too large to be processed as a single UOT problem, so the scene is split into chunks that are solved independently. We recursively bisect the scene along the median of the horizontal coordinates of the two clouds, alternating between the two axes, until each chunk contains at most $30{,}000$ points per point cloud. Each chunk is then extended with a halo of width 5 m in the horizontal plane, made of the points of both epochs lying in the enlarged bounding box, so that points close to a tile boundary keep their neighbours across it. Only the target points of the original tile are then retained, so that each target point is classified exactly once. All OT-based methods use the same chunks.

Additional results. Beyond the UOT solvers of Table 2, we compare MDB with two non-UOT state-of-the-art methods evaluated on the Lyon LiDAR data, M3C2 (Lague et al., 2013) and RFF (Naylor et al., 2023). Table 5 shows that both R-MDB and GS-MDB outperform them on all five settings, with an average mIoU of 60.9% against 52.7% for RFF ($+8.2$) and 44.9% for M3C2 ($+16.0$); the gain over RFF ranges from $+4.0$ points on Lidar05Noisy to $+13.6$ on Lidar05. Notably, no UOT baseline of Table 2 reaches the average mIoU of RFF, so the improvement cannot be ascribed to the UOT formulation alone. Table 6 reports the IoU and F1 score of each class, averaged over the five settings. MDB outperforms all the methods in the table on every class, with the largest gains on the two change classes: R-MDB raises the IoU on New Building from 54.5% (SinkTI) to 74.1% and on Destroyed from 39.6% (M3C2) to 47.7%, while No Change is accurately classified by all methods (IoU $\geq 93.4\%$). Since the mIoU averages the IoU of the two change classes, these gains account for the improvement in Tables 2 and 5.

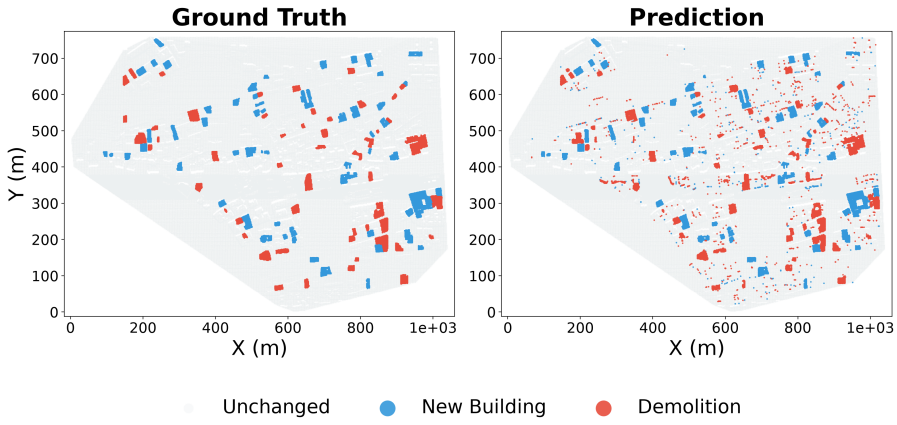
Table 5: mIoU (%) compared against literature baselines M3C2 (Lague et al., 2013) and RFF (Naylor et al., 2023), alongside GS-MDB and R-MDB, on the same five Urb3DCD test configurations.

Data	M3C2	RFF	GS-MDB	R-MDB
(1)	52.1	49.7	**63.3**	**63.3**
(2)	48.0	59.5	<u>68.1</u>	**68.4**
(3)	45.1	53.3	**57.3**	**57.3**
(4)	46.7	54.0	**60.1**	**60.1**
(5)	32.7	47.2	**55.5**	**55.5**
Avg	44.9	52.7	**60.9**	**60.9**

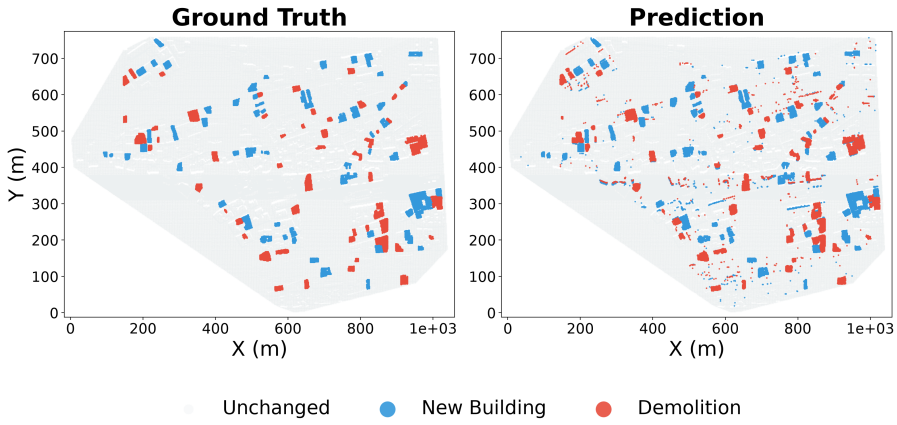
Qualitative results. Figures 3–4 confirm the quantitative results on scene *LyonS* of *Lidar05*, and Figure 5 shows its height maps. R-MDB and GS-MDB recover the footprints of both new and demolished buildings, with errors limited to sparse, isolated points, and are visually almost indistinguishable. Sinkhorn and SinkTI produce more false alarms; SinkLR labels most of the scene as changed, and MM recovers only part of the changed structures.



(a) GS-MDB: 2D plot

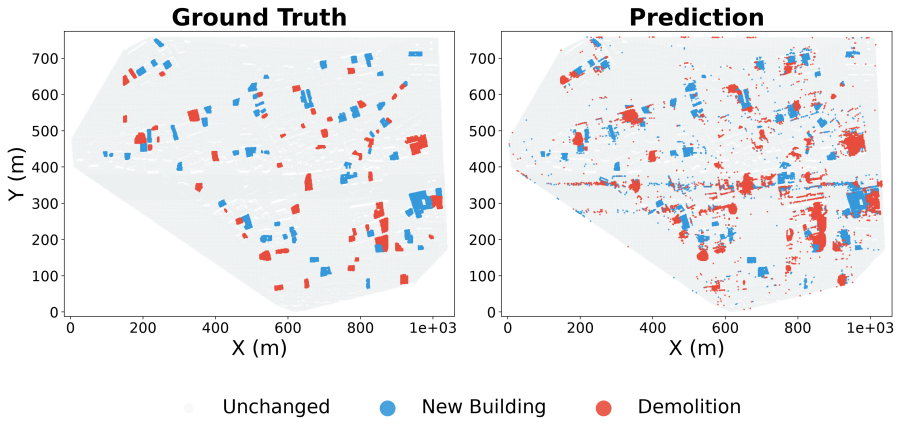


(b) R-MDB: 2D plot

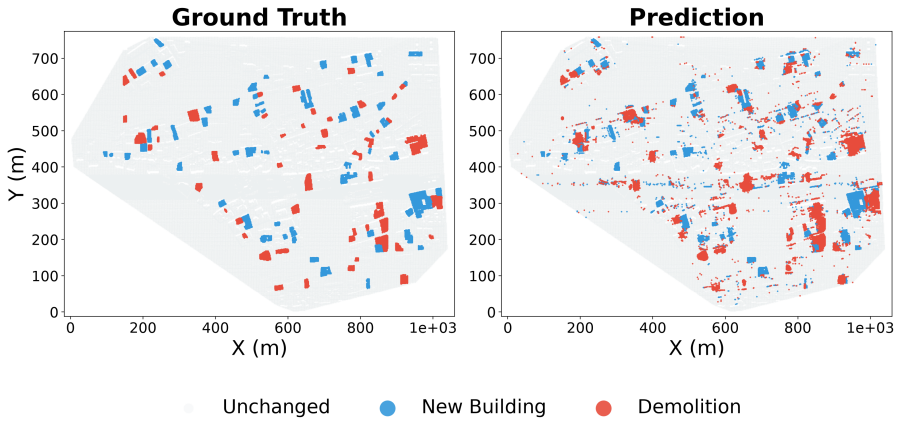


(c) M3C2: 2D plot

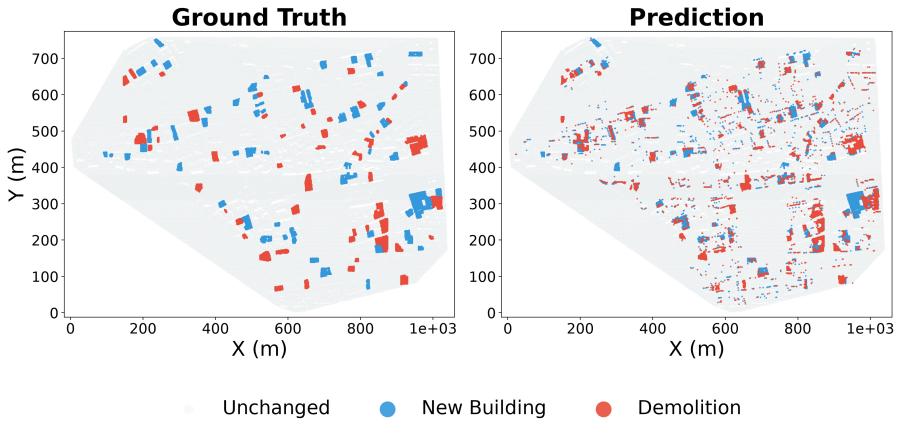
Figure 3: Ground truth vs. prediction comparison (2D) for the three classes on Lidar05 (scene LyonS), across change-detection methods (1/2).



(a) Sinkhorn: 2D plot



(b) Sinkhorn TI: 2D plot



(c) MM Unbalanced: 2D plot

Figure 4: Ground truth vs. prediction comparison (2D) for the three classes on Lidar05 (scene LyonS), across change-detection methods (2/2).

Table 6: Per-class IoU and F1 score (%) of R-MDB, GS-MDB, Sink, SinkTI, and M3C2, averaged across all 5 change-detection datasets (mean ± std). MM and SinkLR are not reported because they did not complete within the time limit on some datasets (see Table 2). **Bold** marks the best method per row, underline the second best.

	R-MDB	GS-MDB	Sink	SinkTI	M3C2
IoU (No Change)	**96.2** ± 0.7	**96.2** ± 0.7	93.5 ± 0.4	94.1 ± 0.6	93.4 ± 1.1
F1 (No Change)	**98.1** ± 0.3	**98.1** ± 0.3	96.6 ± 0.2	96.8 ± 0.5	96.6 ± 0.6
IoU (New Building)	**74.1** ± 5.6	**74.1** ± 5.5	54.2 ± 4.2	54.5 ± 2.9	49.1 ± 6.9
F1 (New Building)	85.0 ± 3.6	**85.1** ± 3.5	69.9 ± 3.5	68.2 ± 3.7	65.2 ± 6.0
IoU (Destroyed)	**47.7** ± 5.3	**47.7** ± 5.2	36.5 ± 2.8	37.9 ± 3.2	39.3 ± 7.9
F1 (Destroyed)	**63.8** ± 5.1	63.6 ± 5.0	53.1 ± 2.9	53.2 ± 2.2	55.5 ± 8.4

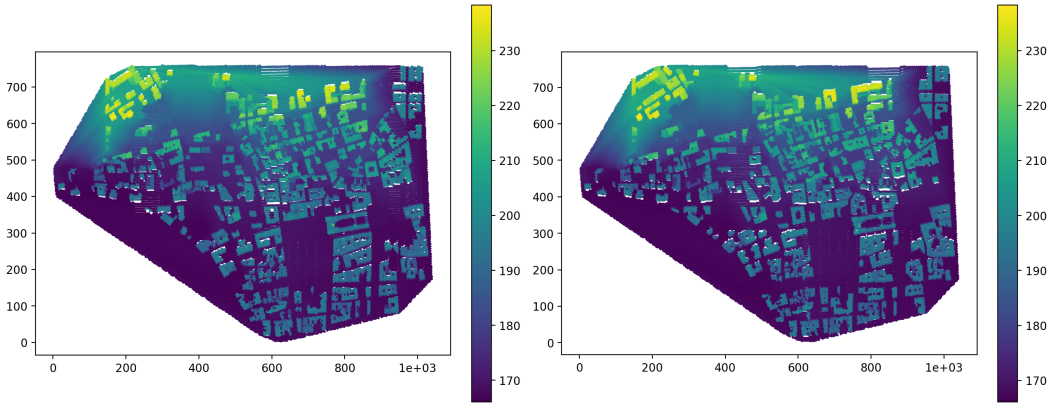


Figure 5: Height maps for scene LyonS (Lidar05).

A.6 Domain Adaptation

Pipeline and settings. To compare MDB with other UOT solvers in a learning setting, we follow the minibatch UOT pipeline of JUMBOT (Fatras et al., 2021), one of the reference approaches to unsupervised domain adaptation (UDA) with unbalanced optimal transport. A classifier is trained on a labeled source domain and evaluated on an unlabeled target domain that shares the same classes but differs in visual style. We use Office-Home (Venkateswara et al., 2017), whose four domains (Art, Clipart, Product and Real-World) contain images of the same 65 object categories and give 12 ordered source-to-target tasks. The network consists of a ResNet-50 backbone pretrained on ImageNet (He et al., 2016), followed by a 512-dimensional bottleneck layer g_θ, which produces the embeddings, and a linear classifier f_λ. At each iteration, a source minibatch $\{(x_i^s, y_i^s)\}_{i=1}^m$ and a target minibatch $\{x_j^t\}_{j=1}^m$ of $m = 65$ images are drawn; as in (Damodaran et al., 2018) and (Fatras et al., 2021), source minibatches are stratified, i.e., they contain one image per class. The ground cost between the i-th source and the j-th target image combines the distance between their embeddings with the cross-entropy $\mathcal{L}$ between the source label and the target prediction,

$$C_{ij} = \eta_1 \left\| g_\theta(x_i^s) - g_\theta(x_j^t) \right\|_2^2 + \eta_2 \, \mathcal{L}\big(y_i^s, f_\lambda(g_\theta(x_j^t))\big),$$

with $\eta_1 = 0.01$ and $\eta_2 = 0.5$ as in Fatras et al. (2021). A UOT solver then computes a transport plan π between the two minibatches, both with uniform weights $1/m$, and the transfer loss $\langle \pi, C \rangle = \sum_{i,j} \pi_{ij} C_{ij}$ is added to the cross-entropy on the source minibatch. The plan is treated as a constant during back-propagation, so that the gradient flows through the cost C only. The network is trained for 5000 iterations with SGD, and the accuracy on the target domain is evaluated every 500 iterations using the ten-crop protocol (Long et al., 2018). With the entropic unbalanced Sinkhorn algorithm (Chizat et al., 2018b) as the solver, this pipeline is exactly JUMBOT, which achieved state-of-the-art accuracy among OT-based UDA methods on Office-Home, consistently outperforming earlier approaches such as DeepJDOT (Damodaran et al., 2018) and ROT (Balaji et al., 2020).

Compared solvers and hyperparameters. We keep the pipeline unchanged and replace only the UOT solver: each solver returns a transport plan, which enters the transfer loss exactly as the Sinkhorn plan does in JUMBOT. We compare R-MDB and GS-MDB with Sink (the original JUMBOT solver), SinkTI (Séjourné et al., 2022), SinkLR (Scetbon et al., 2023) and MM (Chapel et al., 2021). For Sink, we use the hyperparameters of JUMBOT on Office-Home, i.e., marginal penalty $\tau = 0.5$ and entropic regularization $\varepsilon = 0.01$. For the other solvers, the hyperparameters were tuned to maximize the average accuracy over the 12 tasks, leading to $\tau = 0.5$ and $\varepsilon = 0.01$ for SinkTI, $\tau = 0.5$ with the KL divergence for MM, $\tau = 0.5$, rank $r = 10$ and step size $\gamma = 10$ for SinkLR, and $\rho = 1$, $p = 1.1$ and $T = 20$ for both R-MDB and GS-MDB. All other parameters and settings, such as the stopping tolerance and the initialization, are left at each algorithm's default value. Note that MDB uses the power divergence ϕ_p with $p = 1.1$, an approximation of the KL divergence used by the other solvers.

SinkLR is not shown in Figure 2: its accuracy is far below that of all other methods (Table 7), and including it would compress the vertical scale and make the remaining curves hard to distinguish. This behavior is expected. Low-rank solvers are designed to speed up large problems by restricting the plan to a low rank, whereas here each iteration solves a small 65×65 problem whose good plans are close to sparse, class-to-class matchings, which a low-rank plan might not correctly represent. Conversely, because the problems are small, the MM solver, too slow for the large problems of Section 4.2, remains practical here and reaches an accuracy comparable to that of Sink: most of the training time is spent in the forward and backward passes through the network rather than in the UOT solver.

The radius $T = 20$ is larger than in the other experiments because the cost matrix is recomputed at every iteration from embeddings that evolve during training, so the relevant arcs cannot be identified in advance. On 65×65 problems, pruning brings little computational benefit in any case. Since the solver accounts for a small fraction of the training time, the advantage of MDB lies in the plans it computes. After 1000 iterations, R-MDB and GS-MDB reach an average accuracy of 68.5% and 68.4%, against 63.7% for Sink and MM and 63.6% for SinkTI, so the faster progress in Figure 2 reflects a better training signal per iteration rather than a cheaper step. We conjecture that this is due to sparsity: MDB sets entries of the plan exactly to zero, whereas Sink and SinkTI return strictly positive plans and the multiplicative updates of MM keep all entries positive after finitely many

iterations. Dense plans couple images of different classes, transferring incorrect labels to the target domain (Fatras et al., 2021), whereas sparse plans concentrate the mass on the most similar pairs.

Table 7: Best accuracy (%) of R-MDB, GS-MDB, Sink, SinkTI, SinkLR and MM within 5000 iterations on the Office-Home domain adaptation benchmark; **bold** marks the best method for each task, underline the second best.

Method	AC	AP	AR	CA	CP	CR	PA	PC	PR	RA	RC	RP	Avg
R-MDB	**55.1**	74.2	**79.9**	65.0	**74.9**	**75.7**	64.3	**53.2**	79.3	73.1	59.0	**83.3**	**69.8**
GS-MDB	54.1	74.1	79.7	**65.1**	73.9	75.4	64.2	52.3	79.5	73.5	59.4	82.9	69.5
Sink	53.6	73.7	**79.9**	64.8	72.4	73.7	**64.7**	52.3	**79.6**	74.1	**60.0**	83.2	69.3
SinkTI	54.0	**74.9**	79.2	64.9	71.8	73.4	64.2	52.0	79.5	74.3	59.6	83.1	69.2
SinkLR	43.6	63.2	73.9	52.8	57.3	62.6	49.2	34.3	70.0	65.1	43.0	75.0	57.5
MM	53.7	74.8	79.4	64.6	72.2	73.2	64.5	52.5	**79.6**	**74.7**	59.7	83.1	69.3