Distribution Agreement

Tianyang Hu December 11, 2024

Optimal Transportation: A Comprehensive Review and Novel Approaches

by

Tianyang Hu

Levon Nurbekyan

Adviser

Mathematics Department

Levon Nurbekyan

Adviser

Elizabeth Newman

Committee Member

Ruoxuan Xiong

Committee Member

2024

Optimal Transportation: A Comprehensive Review and Novel Approaches

By

Tianyang Hu

Levon Nurbekyan

Adviser

An abstract of
a thesis submitted to the Faculty of Emory College of Arts and Sciences
of Emory University in partial fulfillment
of the requirements of the degree of
Bachelor of Science with Honors

Mathematics Department

2024

Abstract

Optimal Transportation: A Comprehensive Review and Novel Approaches

By Tianyang Hu

This thesis explores optimal transport (OT) with a focus on improving computational efficiency and generalization through novel partitioning strategies in no-collision transportation maps. The first part provides a comprehensive review of the theoretical foundations and computational techniques in classical OT, including the Monge and Kantorovich problems, Kantorovich duality, and various transportation distances such as the Wasserstein metric and Linearized Optimal Transport (LOT). It critically examines established computational methods like the North-West Corner Method, Network Simplex Algorithm, and Auction Algorithm, highlighting their strengths and limitations.

The second part introduces alternative partitioning strategies that challenge the traditional vertical-horizontal partitioning methods in no-collision transportation maps. It proposes and rigorously tests two new partitioning techniques—random and PCA-based partitioning—analyzing their computational efficiency and generalization capabilities. The results demonstrate the potential of these strategies to improve both the scalability and robustness of OT solutions, offering new avenues for their application in complex fields such as finance and economics.

Optimal Transportation: A Comprehensive Review and Novel Approaches

By

Tianyang Hu

Levon Nurbekyan

Adviser

A thesis submitted to the Faculty of Emory College of Arts and Sciences
of Emory University in partial fulfillment
of the requirements of the degree of
Bachelor of Science with Honors

Mathematics Department

2024

Acknowledgements

This thesis marks the culmination of a deeply enriching and challenging journey, one that I could not have completed without the steadfast support of my supervisors, committee members, and everyone who has nurtured and believed in me throughout my academic and personal endeavors.

I am profoundly grateful to my supervisor, Prof. Levon Nurbekyan, for providing me with the opportunity to delve into the captivating mathematical topics that inspire me. His guidance extended beyond the academic rigors of this thesis, aiding significantly in the shaping of my life goals. I wish to express my heartfelt thanks to Prof. Nurbekyan's passion for research and invaluable advising to my professional and personal growth.

Additionally, I extend my sincerest thanks to Prof. Liz Newman and Prof. Ruoxuan Xiong for their roles on my honor committee. Their insightful feedback and probing questions have been crucial in refining my thesis. I am equally thankful for their lucid and profound instruction in courses that laid the robust academic foundation upon which this thesis was built.

Immense gratitude goes to my parents; thank you for giving me life and strength. I am blessed to grow in a loving and supportive home my parents together built for me. I will not be the person I am today without my parents who support every choice I made, enlighten my gloomy days, and unwaveringly believe in me.

A special thank you to Leslie, who has been my steadfast companion through both triumphant and challenging times. Your enduring support and affirmation of

my worth have been a pillar of strength.

Finally, I am grateful to all who have embraced my imperfections and supported me unconditionally. Your love and encouragement have been integral to my journey.

Contents

List of Tables

List of Figures

Introduction

In the pursuit of efficiency, the path of least resistance has long been a guiding principle for both nature and human endeavor: In nature, ants forge pheromone trails towards food source, gravitating along paths marked by the strongest scents which indicates the shortest or most resource-efficient route [1]; in human-engineered systems, airlines companies often calculate wind patterns and earth's curvature to plan routes along the shortest path between two locations to minimize distance traveled and hence fuel consumption.

In mathematical terms, this pursuit translates into the optimal transport problem, where the objective is to determine the most efficient way to move resources from one configuration to another while minimizing a given cost function. While the framework appears straightforward, the complexity of real-world applications introduces a multitude of challenges. These challenges stem from the high-dimensional nature of the data, the complexity of constraints that can be both nonlinear and non-convex, and the sensitivity to the accuracy of input data. Even simplified models often require advanced numerical methods and substantial computational resources to find solutions that are close to optimal. Moreover, the increasing application of optimal transport in machine learning—for tasks such as domain adaptation and improving generative models—further motivates mathematicians to address these complex problems.

The foundational work by Monge and later Kantorovich provides a formal structure to approach these problems, setting the stage for a variety of mathematical strategies aimed at finding solutions. However, as comprehensive as the Monge-Kantorovich framework is, it often necessitates simplifications or approximations when applied to complex or large-scale problems.

In response to these challenges, the field has begun to explore alternative methodologies. Among these, no-collision transportation maps are a promising direction providing an alternative notion of transportation maps that have some of the advantageous properties of the optimal ones but are much cheaper to compute.

This thesis seeks to provide a thorough examination of the foundational theories of optimal transportation, alongside a critical review of alternative methodologies that have emerged to address the computational complexities inherent in these theories. It will first establish the mathematical underpinnings of optimal transportation and then explore their computational applications. The primary focus of this study is an in-depth analysis of no-collision transportation maps, which present a promising alternative due to their computational tractability. Additionally, this thesis will introduce a novel partitioning method, offering a new perspective on the practical implementation and potential benefits of no-collision maps.

To achieve the objectives outlined, the structure of this thesis is organized as follows. Section 2 begins by introducing the essential mathematical frameworks, including measure theory, the Monge problem, the Kantorovich problem, and their dual formulations. This is followed by a review of computational techniques used for these theories, assessing their strengths, weaknesses, and computational intensiveness. This thesis then explores alternative transportation distances, especially Sliced Wassrestein Distances and Linearized Optimal Transportation Distance, and critically examines the limitations and challenges of existing partitioning methods. The concept of no-collision transportation maps is discussed next, with a focus on partitioning data into equal-weight parts and their connections with k-d trees. Moving on, this thesis aims to propose two new partition methods on no-collision transportation. Section 4 details the methodology, including the proposed partitioning methods and computational strategies, followed by a theoretical analysis and comparative evaluation of different approaches. The thesis concludes with the presentation of computational results and a discussion of the findings, along with suggestions for future research directions.

State of the Art

This section provides a comprehensive review of the existing literature and foundational concepts in the field of optimal transportation. It serves to contextualize the current research by summarizing key definitions, problems, dualities, and computational techniques that have been developed and studied in past works. All definitions and discussions presented herein are derived from established sources and are not original contributions of this thesis.

0.1 Optimal Transportation Preliminaries

This section delves into some of the essential mathematical frameworks that underpin optimal transportation (OT) theory, including basic measure theory, the Monge problem formulation, and the Kantorovich relaxation. Furthermore, we discuss the dual problem, which plays a critical role in providing insights into the cost minimization strategies inherent in OT.

0.1.1 Measures

Measure theory serves as the formalism for the mathematically rigorous definition of the transportation of goods. Here, we briefly discuss the fundamentals of the measure theory; see [11, 2] for an in-depth exposition of the subject.

Informally speaking, *measure* is a systematic way to assign a size to various mathematical objects. Hence, measure is a mathematical formalization and extension of real life concepts of length, area, and volume.

Definition 0.1.1 (σ-algebra)**.** *Let X be a set, and $\mathcal{A}$ be a collection of subsets of A. $\mathcal{A}$ is called a σ-algebra on X if the following three properties hold:*

1. $\emptyset \in \mathcal{A}$.

2. *For all $A \in \mathcal{A}$ one has that $A^c = X \setminus A \in \mathcal{A}$.*

3. *For all $A_1, A_2, A_3, \dots \in \mathcal{A}$, one has that $\bigcup_{i=1}^{\infty} A_i \in \mathcal{A}$.*

A σ**-algebra** specifies the subsets of a set X that can consistently be assigned a size by a measure. Not all subsets of X are guaranteed to behave well with a measure, especially in more complex settings. The σ-algebra ensures that operations like unions, intersections, and complements remain well-defined, making it possible to assign sizes to subsets in a consistent and mathematically rigorous way.

Example 0.1.1 (σ-Algebra of Weekdays)**.** *Let X be the set of days in a week:*

$$X = \{\textit{Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday}\}.$$

The σ-algebra $\mathcal{A}$ can be chosen as the collection of all subsets of X: $\mathcal{A} = 2^X$, which includes:

- *The empty set: $\emptyset$,*
- *Single-day subsets: e.g., $\{$Monday$\}$,*
- *Multi-day subsets: e.g., $\{$Saturday, Sunday$\}$,*
- *The entire set X.*

This σ-algebra satisfies all the properties:

- *The empty set and the full set X are included.*
- *For any subset $A \in \mathcal{A}$, its complement $A^c = X \setminus A$ is also in $\mathcal{A}$.*
- *The union of any countable collection of subsets in $\mathcal{A}$ is also in $\mathcal{A}$.*

Definition 0.1.2 (Measures)**.** *Let X be a set and $\mathcal{A}$ a σ-algebra on X. A measure on $(X, \mathcal{A})$ is a function $\mu : \mathcal{A} \to [0, \infty]$ that satisfies the following properties:*

1. $\forall A \in \mathcal{A}, \mu(A) \geq 0.$
2. $\mu(\emptyset) = 0.$
3. *For all $A_1, A_2, \cdots \in \mathcal{A}$ such that $A_i \cap A_j = \emptyset$ for $i \neq j$, one has that $\mu(\bigcup_{i=1}^{\infty} A_i) = \sum_i^{\infty} \mu(A_i)$.*

Example 0.1.2 (Measure of Rainfall Probability on Weekdays)**.** *Given the σ-algebra $\mathcal{A}$ from the previous example, define a measure μ that assigns probabilities to subsets of X, representing the likelihood of rainfall on those days. Specifically:*

$$\mu(\textit{Monday}) = \mu(\textit{Tuesday}) = \mu(\textit{Wednesday}) = \mu(\textit{Thursday}) = 0.1,$$

$$\mu(\textit{Friday}) = \mu(\textit{Saturday}) = \mu(\textit{Sunday}) = 0.2.$$

We can quickly check if it satisfies the measures' properties:

1. *Non-negativity:* $\mu(A) \geq 0$ *for all subsets* $A \in \mathcal{A}$*.*

2. *Null Empty Set:* $\mu(\emptyset) = 0$ *as the probability of rainfall on no-day is zero.*

3. *Countable Additivity: If subsets are disjoint, their measures add up (Under the assumption that the events of rainfall on each day are independent).*

The concept of a measure is foundational in various mathematical fields due to its generality and flexibility. In optimal transport theory, measures allow for the representation of distributions of mass or resources, whether they are spread continuously across a space or concentrated at discrete points.

Digression (Banach-Tarski Paradox). Let $X = \mathbb{R}^d$. Then $d = 3$ corresponds to our usual 3-dimensional space. The larger $\mathcal{A}$ in Definition 0.1.2 the better – indeed, one can quantify more subsets of X. In particular, the following natural question arises:

Question. Can we assign volumes consistently to all subsets of $\mathbb{R}^3$?

Any measure that is invariant under congruence (translation, rotations, and reflections) and assigns a non-zero measure to the unit ball A can be a candidate for a volume. The answer to this question is *No* and a consequence of the following theorem.

Theorem 0.1.1 (Banack-Tarski Paradox)**.** *Let $d \geq 3$, and A be the unit ball in $\mathbb{R}^d$. Furthermore, let B be the set consisting of two identical disjoint copies of A. There exist disjoint sets $C_1, C_2, \cdots, C_k$ and $D_1, D_2, \cdots, D_k$ such that*

- *$A = C_1 \cup C_2 \cup C_3 \cup \cdots \cup C_k$, and $B = D_1 \cup D_2 \cup D_3 \cup \cdots \cup D_k$,*

- *C_i and D_i are congruent for $1 \leq i \leq k$.*

If we were able to assign a volume to all subsets of $\mathbb{R}^3$; that is, if there were μ defined on all subsets of $\mathbb{R}^3$ invariant under congruence then we would have

$$\mu(A) = \sum_{i=1}^{k} \mu(C_i) = \sum_{i=1}^{k} \mu(D_i) = \mu(B) = 2\mu(A),$$

which leads to a contradiction.

In summary, the paradox demonstrates that a set can be split into pieces, rearranged, and reassembled to form two identical copies of the original set, violating the usual principles of volume.

This paradox highlights the importance of properly defining a measure using σ-algebras. A natural collection of sets where we can consistently assign volumes and other useful measures is the **Borel σ-algebra**, which consists of sets that can be generated from open sets through countable unions, intersections, and complements. The Borel σ-algebra provides a framework for defining well-behaved measures, such as volume, that avoid the paradoxical inconsistencies encountered in the Banach-Tarski construction.

Definition 0.1.3 (Borel σ-algebra)**.** *The Borel σ-algebra in $\mathbb{R}^d$ is the smallest σ-algebra containing all open sets in $\mathbb{R}^d$. We denote the Borel σ-algebra by $\mathcal{B}(\mathbb{R}^d)$ and say that a measure μ is a Borel measure if it is defined on $\mathcal{B}(\mathbb{R}^d)$.*

Borelσ-algebra focuses specially on $\mathbb{R}^d$, and a special measure here is **Borel Probability Measures**, which allow us to model distributions of random phenomena, assigning probabilities to subsets of $\mathbb{R}^d$. For example, we can use them to describe the likelihood of events in probability theory or to integrate functions for expected values.

Definition 0.1.4 (Borel Probability Measures)**.** *A Borel measure μ on $\mathcal{B}(\mathbb{R}^d)$ is called a probability measure if $\mu(\mathbb{R}^d) = 1$. The set of all Borel probability measures on $\mathbb{R}^d$ is denoted by $\mathcal{P}(\mathbb{R}^d)$.*

Question. So far we are only focusing on measures on sets, but can we extend measures to act not just on subsets of $\mathbb{R}^d$ but also on functions?

Theorem 0.1.2 (Riesz Representation Theorem)**.** *For any bounded linear functional $L : C_b(\mathbb{R}^d) \to \mathbb{R}$, there exists a unique Borel measure $\mu \in \mathcal{P}(\mathbb{R}^d)$ such that:*

$$L(f) = \int_{\mathbb{R}^d} f(x)\, d\mu(x), \quad \forall f \in C_b(\mathbb{R}^d).$$

Several useful definitions here:

- μ: A measure on a space X.
- $f(x)$: A continuous, bounded function on X ($f \in C_b(X)$).
- $L(f)$: A functional that applies μ to $f(x)$.

Example 0.1.3 (Aggregating a Linear Function under a Uniform Measure)**.** *Let $X = [0,1]$, $f(x) = 2x$, and μ be the uniform measure on $[0,1]$. The functional:*

$$L(f) = \int_0^1 f(x)\, d\mu(x) = \int_0^1 2x\, dx = \left[x^2\right]_0^1 = 1.$$

Interpretation: $L(f)$ aggregates the values of $f(x)$ over $[0,1]$, weighted by μ.

Continuous and Discrete Measures: The behavior of the integral depends on whether the measure μ is continuous or discrete.

- **Absolutely Continuous Measures:** If μ is absolutely continuous with respect to the Lebesgue measure, there exists a density function $\rho(x) \geq 0$ such that:

  $$\int_{\mathbb{R}^d} f(x)\, d\mu(x) = \int_{\mathbb{R}^d} f(x)\rho(x)\, dx.$$

  Here, $\rho(x)$ represents the "weight" or density of the measure at each point.

- **Discrete Measures:** For discrete measures, μ assigns weights to specific points $x_i \in \mathbb{R}^d$:

$$\int_{\mathbb{R}^d} f(x)\, d\mu(x) = \sum_i f(x_i)\mu(\{x_i\}),$$

where the measure acts only on these points.

Example 0.1.4 (Insurance Risk Assessment: Continuous and Discrete Measures)**.** *Consider an insurance company assessing the risk of natural disasters across various geographical regions. Let X represent the entire geographical region under consideration. The company models disaster risk using a probability measure μ on X.*

Probability of Disaster in Subareas of X: *The company needs to evaluate the likelihood of a natural disaster occurring in specific subareas of X. For any subset $A \subseteq X$, the probability measure μ assigns a value $\mu(A)$, where:*

$$\mu(X) = 1, \quad \mu(A) \geq 0 \quad \textit{for all } A \subseteq X.$$

For instance, $\mu(\textit{Coastal Region})$ might represent the probability of a disaster occurring in the coastal areas of the region.

Probability of Disaster at Specific Locations: *To model disaster likelihood at a specific location $x \in X$, the company introduces the probability density function (PDF) $\rho_\mu(x)$. This PDF describes the distribution of disaster risk across the region. The relationship between the probability measure μ and the PDF $\rho_\mu(x)$ is* *given by:*

$$d\mu(x) = \rho_\mu(x)\,dx.$$

This means that the infinitesimal probability assigned to a small area around x is proportional to the density $\rho_\mu(x)$ at that point.

Discrete vs. Continuous Measures:

- *If the measure μ is discrete, the disaster risk is concentrated at specific points (e.g., urban centers or critical infrastructure). The probability measure is a sum of weights at these points:*

$$\mu = \sum_i \mu(\{x_i\}), \quad \textit{and} \int_X f(x)\,d\mu(x) = \sum_i f(x_i)\mu(\{x_i\}).$$

- *If the measure μ is continuous, the disaster risk is distributed across the region, with the density function $\rho_\mu(x)$ describing the likelihood at each point:*

$$\mu(A) = \int_A \rho_\mu(x)\,dx, \quad \textit{and} \int_X f(x)\,d\mu(x) = \int_X f(x)\rho_\mu(x)\,dx.$$

Expected Financial Impact of Disasters: *The company is also concerned with the financial consequences of disasters. Let $c(x)$ represent the financial cost of a disaster occurring at location x. The expected total cost is given by:*

$$\int_X c(x)\,d\mu(x),$$

where $c(x)$ is weighted by the probability of a disaster occurring at x. If the measure μ is continuous, this becomes:

$$\int_X c(x)\rho_\mu(x)\,dx.$$

Connection to the Riesz Representation Theorem: *The functional $L(f) = \int_X f(x)\,d\mu(x)$ aggregates information about the function $f(x)$ (e.g., cost or risk) over the region X, weighted by the probability measure μ. By the Riesz Representation Theorem, this functional corresponds uniquely to the measure μ, demonstrating how measures can operate on functions, not just subsets.*

0.1.2 Monge Problem

Optimal transportation theory starts with Gaspard Monge [7]. A simplified version of his problem is as follows.

Monge problem between discrete measures. Let α, β be discrete reference (source) and target measures, respectively, defined as follows:

$$\alpha = \sum_{i=1}^{n} a_i \delta_{x_i}, \quad \beta = \sum_{j=1}^{m} b_j \delta_{y_j}$$

One can think of α representing piles of construction rubble located at $\{x_i\}$ with masses $\{a_i\}$, and β representing holes at locations $\{y_j\}$ with capacities $\{b_j\}$, where the rubble should be transported.

Thus, Monge problem searches a (transportation) map $\sigma : \{1, \ldots, n\} \rightarrow$

$\{1, \ldots, m\}$ that minimizes the total transportation cost. Mathematically, the problem reads as

$$\min_{\sigma} \sum_{i=1}^{n} c(x_i, y_{\sigma(i)}) a_i, \tag{1}$$

where $c(x_i, y_j) \geq 0$ represents the cost of transporting a unit mass from source x_i to destination y_j, which could be measured by various metrics such as Euclidean distance or other relevant cost functions.

Example 0.1.5 (Transporting Rubble to Construction Sites)**.** *Suppose* $\{x_1, x_2\}$ *are locations with rubble piles of* $10\,\mathrm{kg}$ *and* $15\,\mathrm{kg}$*, respectively. Let* $\{y_1, y_2\}$ *be construction sites with capacities* $15\,\mathrm{kg}$ *and* $10\,\mathrm{kg}$*. The Monge problem seeks to assign a transport map* σ *that minimizes the total transportation cost:*

$$\min_{\sigma} \sum_{i=1}^{n} c(x_i, y_{\sigma(i)}) a_i,$$

where $c(x_i, y_j)$ *could represent the distance between* x_i *and* y_j*. For instance, if* $c(x_1, y_1) = 2$*,* $c(x_1, y_2) = 5$*,* $c(x_2, y_1) = 4$*, and* $c(x_2, y_2) = 1$*, the optimal transport plan minimizes total cost:*

$$\textit{Optimal plan: } x_1 \to y_1,\ x_2 \to y_2 \textit{ with cost } 10 \cdot 2 + 15 \cdot 1 = 35.$$

Note that we require all mass being transported and no surplus left in the reference or target measures; that is,

$$b_j = \sum_{\sigma(i)=j} a_i, \quad \forall 1 \leq j \leq m. \tag{2}$$

In particular, we require that the total masses are equal; that is,

$$\sum_{j=1}^{m} b_j = \sum_{j=1}^{m} \sum_{\sigma(i)=j} a_i = \sum_{i=1}^{n} a_i.$$

Equation (2) means that σ pushes α forward to β. Below, we define the push-forward operation rigorously to consider more general measures.

The push-forward operator provides a way to transform measures consistently using a measurable map T. Recall that a measure α is defined on a σ-algebra $\mathcal{A}$, which is a collection of subsets of a space X that satisfies certain properties. Let α be a measure on $\mathcal{A}$, and let $T : \mathbb{R}^d \to \mathbb{R}^d$ be a measurable map; that is, $T^{-1}(E) \in \mathcal{A}$ for every $E \in \mathcal{A}$. The push-forward operator transforms α into another measure β on the image space via the map T. Formally, it's defined as:

Definition 0.1.5 (Push-forward)**.** *Let $\alpha \in \mathcal{P}(\mathbb{R}^d)$, and $T : \mathbb{R}^d \to \mathbb{R}^d$ be a measurable map; that is, $T^{-1}(E) \in \mathcal{B}(\mathbb{R}^d)$ for all $E \in \mathcal{B}(\mathbb{R}^d)$. The push-forward of α through T is $\beta = T_{\#}\alpha \in \mathcal{P}(\mathbb{R}^d)$ that satisfies the following identity:*

$$\int_{\mathbb{R}^d} h(y)\, d\beta(y) = \int_{\mathbb{R}^d} h(T(x))\, d\alpha(x), \quad \forall h \in C_0(\mathbb{R}^d). \tag{3}$$

Equation (3) states that the integration against the target measure β is equivalent to the integration against the transported source measure α. Furthermore, the push-forward operator preserves positivity and total mass, ensuring that if α is a probability measure then β is also a probability measure. Finally, note that for discrete α, β identity (3) reduces to

$$\sum_{j=1}^{m} b_j h(y_j) = \sum_{i=1}^{n} a_i h(T(x_i)), \quad \forall h \in C_0(\mathbb{R}^d),$$

which is equivalent to (2), where T and σ are related by

$$T(x_i) = y_j \iff \sigma(i) = j.$$

Alternatively, the push-forward measure $\beta = T_{\#}\alpha$ can be equivalently defined in terms of the **preimage**:

Definition 0.1.6 (Push-Forward via Preimage)**.** *Let $\alpha \in \mathcal{P}(\mathbb{R}^d)$ and $T : \mathbb{R}^d \to \mathbb{R}^d$ be a measurable map. For any measurable set $E \in \mathcal{B}(\mathbb{R}^d)$, the push-forward measure $\beta = T_{\#}\alpha$ satisfies:*

$$\beta(E) = \alpha(T^{-1}(E)),$$

where $T^{-1}(E) = \{x \in \mathbb{R}^d : T(x) \in E\}$ is the preimage of E under the map T.

This definition aligns with the functional definition in Definition 0.1.5 since for any indicator function $h(y) = \chi_E(y)$, we have:

$$\int_{\mathbb{R}^d} \chi_E(y)\, d\beta(y) = \int_{\mathbb{R}^d} \chi_E(T(x))\, d\alpha(x).$$

This equivalence ensures that the measure of a subset in the target space E is consistent with the measure of its preimage $T^{-1}(E)$ in the source space.

Intuitive Analogy: Think of T as a function that maps cities (A) to regions (B) based on population density. For a region in B (e.g., "high-density areas"), the preimage in A represents all the cities contributing to that region.

The preimage condition ensures we can always trace back from B to A:

- If one wants to know the total population in "high-density areas" (B), one must be able to calculate the population of the corresponding cities (A).
- If the cities forming "high-density areas" ($T^{-1}(E)$) are not well-defined or measurable, one cannot consistently compute the population.

This highlights the importance of the preimage condition, ensuring that measurable sets in the target space correspond to measurable sets in the source space, preserving the consistency of operations like integration and measure calculations.

Monge problem between arbitrary measures. Let $\alpha, \beta \in \mathcal{P}(\mathbb{R}^d)$ be arbitrary probability measures. The Monge problem is then formulated as follows:

$$\min_{T:\mathbb{R}^d \to \mathbb{R}^d} \left\{ \int_{\mathbb{R}^d} c(x, T(x)) d\alpha(x) : T_{\#}\alpha = \beta \right\} \tag{4}$$

Note that (4) is an extension of (1).

Push-forward and absolutely continuous measures. Assume that α, β are absolutely continuous measures; that is,

$$d\alpha(x) = \rho_\alpha(x)dx, \quad d\beta(y) = \rho_\beta(y)dy.$$

Then (3) becomes the well-known change of variables formula:

$$\rho_\alpha(x) = |\det(JT(x))|\, \rho_\beta(T(x)), \tag{5}$$

where $|\det(JT(x))|$ is the Jacobian determinant of T.

Justification for Equation(5). Recall the pushforward measure $T_\# \alpha$ is defined such that for any measurable set $B \subseteq \mathbb{R}^n$:

$$T_\# \alpha(B) = \alpha(T^{-1}(B)).$$

If the measure α has a density $\rho_\alpha(x)$ with respect to the Lebesgue measure, then under the change of variables theorem, the total mass in B is preserved:

$$\int_B \rho_\alpha(x)\, dx = \int_{T(B)} \rho_\beta(y)\, dy.$$

Switching to the original variable x, this becomes:

$$\int_{T^{-1}(B)} \rho_\alpha(x)\, dx = \int_B \rho_\beta(T(x))\, |\det(JT(x))|\ dx,$$

where $|\det(JT(x))|$ accounts for the local change in volume caused by T. Equating the integrands gives:

$$\rho_\alpha(x) = |\det(JT(x))|\, \rho_\beta(T(x)).$$

This shows that the Jacobian determinant adjusts the density ρ_β to account for the local expansion or compression induced by the transformation T, ensuring that the total mass is preserved.

Specifically, the geometric effect of the Jacobian determinant can be discussed in the following two cases:

- If $|\det(JT(x))| > 1$, T expands space at x, causing the density $\rho_\alpha(x)$ to decrease proportionally.

- If $|\det(JT(x))| < 1$, T compresses space at x, increasing the density $\rho_\alpha(x)$ to maintain the total mass.

Remark 0.1.1. *While mass is preserved during the transformation, the densities ρ_α and ρ_β are not directly inherited. Specifically, even if measures (α, β) are associated with densities $(\rho_\alpha, \rho_\beta)$ with respect to a fixed base measure, the transformed measure $T_\#\alpha$ does not simply inherit ρ_α as $\rho_\beta = \rho_\alpha \circ T^{-1}$. Instead, the Jacobian determinant $|\det(JT(x))|$ adjusts the density, reflecting the local compression or expansion caused by T.*

In applications like the Monge problem, this adjustment is crucial. For example, in image registration tasks, preserving the original density and texture of the image may be essential. The transformation T can significantly affect pixel intensities and spatial continuity, potentially leading to distortions or unnatural results if the Jacobian determinant is not properly considered. See [10] for further examples and discussion.

Example 0.1.6 (Mapping Uniform Measure via Push-Forward)**.** *Consider $X = [0, 1]$ with the uniform measure α, and let $T(x) = x^2$. The target space is also*

$Y = [0,1]$, *and the push-forward measure* $\beta = T_{\#}\alpha$ *redistributes the mass from* α *as follows:*

- *Subset* $E = [0, 0.25]$*:*

  $$T^{-1}(E) = \{x \in [0,1] : x^2 \in [0, 0.25]\} = [0, 0.5].$$

  Since α *is uniform:*

  $$\beta([0, 0.25]) = \alpha([0, 0.5]) = 0.5.$$

- *Subset* $E = [0.25, 1]$*:*

  $$T^{-1}(E) = \{x \in [0,1] : x^2 \in [0.25, 1]\} = [0.5, 1].$$

  Again, since α *is uniform:*

  $$\beta([0.25, 1]) = \alpha([0.5, 1]) = 0.5.$$

- *Total Mass: The total measure is preserved since:*

  $$\beta([0,1]) = \alpha([0,1]) = 1.$$

0.1.3 Kantorovich problem

Let $\alpha, \beta \in \mathcal{P}(\mathbb{R}^d)$. The existence of at least one map T such that (3) holds is a necessary condition for a meaningful Monge problem. Unfortunately, discrete reference measures do not always admit such maps.

Indeed, let us examine discrete one-dimensional probability distributions, α, β given by

$$\alpha = 0.2\delta_{x_1} + 0.5\delta_{x_2} + 0.3\delta_{x_3}, \quad \beta = 0.3\delta_{y_1} + 0.4\delta_{y_2} + 0.3\delta_{y_3},$$

for distinct points $\{x_i\}, \{y_j\}$. In this setup, the Monge problem does not admit a feasible push-forward function because there is no direct way to map the entire mass from x_2 to a single target in $\{y_1, y_2, y_3\}$.

Nonetheless, both distributions sum to a total mass of 1, suggesting that β can indeed accommodate all mass from α if we relax the notion of transportation and allow mass from the source to split.

Thus, despite its foundational role in the optimal transportation theory, the Monge formulation has notable limitations due to its restrictive mapping requirements. This leads us to the Kantorovich formulation, which accommodates the mass splitting at source locations $\{x_i\}$ for transportation to multiple destinations. In the following discussions, we will explore several advantages of the Kantorovich formulation over the Monge one.

Kantorovich problem between discrete measures. We start with a discussion of the Kantorovich formulation of the optimal transportation problem between discrete measures. A foundational element in this framework is the *transportation plan*, a mathematical construct that describes the possible ways in which the supports of two discrete probability distributions can be paired. For distributions

$$\alpha = \sum_{i=1}^{n} a_i \delta_{x_i}, \quad \beta = \sum_{j=1}^{m} b_j \delta_{y_j},$$

it is important to note that probability distributions are not sets and do not have elements. Instead, their supports are sets, which are the collections of points where the distributions assign positive mass.

a transportation plan is a matrix $P = (p_{ij})_{i=1,j=1}^{n,m} \in \mathbb{R}_+^{n\times m}$ such that:

- p_{ij} is the mass being transported from x_i to y_j,
- the sum of the masses in any row i of matrix P is equal to a_i, and the sum of the masses in any column j of matrix P is equal to b_j; that is,

$$a_i = \sum_{j=1}^{m} p_{ij}, \quad b_j = \sum_{i=1}^{n} p_{ij}, \quad \forall i, j.$$

Building on the concept of a transportation plan, *admissible plans* are defined as the set of all such matrices that comply with the mass distribution requirements of both source and target distributions:

$$U(\alpha, \beta) \overset{\text{def}}{=} \{P \in \mathbb{R}_+^{n\times m} : P\mathbb{1}_m = a \text{ and } P^T\mathbb{1}_n = b\} \tag{6}$$

where $1_m \in \mathbb{R}^m$ and $1_n \in \mathbb{R}^n$ are vectors with all-1 entries. This set of matrices, or couplings, forms the feasible space within which the Kantorovich problem seeks an optimal solution minimizing a given transportation cost.

Let us look back to the example discussed earlier, where we have

$$\alpha = 0.2\delta_{x_1} + 0.5\delta_{x_2} + 0.3\delta_{x_3}, \quad \beta = 0.3\delta_{y_1} + 0.4\delta_{y_2} + 0.3\delta_{y_3},$$

An example of a transportation plan P for this case is

$$P = \begin{pmatrix} 0.2 & 0 & 0 \\ 0.1 & 0.4 & 0 \\ 0 & 0 & 0.3 \end{pmatrix}$$

Indeed, this matrix satisfies the admissible coupling conditions where the row sums match α and the column sums match β:

- Row sums: $(0.2, 0.5, 0.3)$ corresponding to the weights of α
- Column sums: $(0.3, 0.4, 0.3)$ corresponding to the weights of β

The second row of P illustrates how the mass, 0.5, from x_2 in the support of α is split and transported to two different locations in the support of β:

- the entry $p_{21} = 0.1$ signifies that 0.1 units of mass are transported from x_2 to y_1,
- the entry $p_{22} = 0.4$ indicates that 0.4 units of mass are transported from the same x_2 to y_2.

This example shows how transporatation plans generalize transportation maps by allowing mass to be split across multiple destinations, thus overcoming a basic limitation inherent to the Monge formulation. Thus, Kantorovich formulation of the discrete optimal transportation problem reads as

$$\min_{P \in U(\alpha,\beta)} \sum_{i=1}^{n} \sum_{j=1}^{m} c(x_i, y_j) p_{ij}. \tag{7}$$

Permutation matrices as transportation plans. A key aspect of the Kantorovich's formulation is that it incorporates the Monge problem. Indeed, for the sake of simplicity, assume that $m = n$, and $a_i = b_j = \frac{1}{n}$ for all i, j; that is, we have n source and target points with equal weights.

In the context of the Kantorovich problem, permutation matrices provide specific examples of transportation plans. In the Monge problem, transportation maps are provided by transportation plans; indeed,

$$T(x_i) = y_j \iff \sigma(i) = j,$$

where $\sigma : \{1, 2, \cdots, n\} \to \{1, 2, \cdots, n\}$ is a permutation (bijection).

Let us now build a corresponding permutation matrix P_σ as follows:

$$\forall (i,j) \in [n]^2, (P_\sigma)_{i,j} = \begin{cases} \frac{1}{n} & \text{if } j = \sigma(i), \\ 0 & \text{elsewhere.} \end{cases}$$

Since σ is a permutation, each $1 \leq i \leq n$ is mapped to only one $1 \leq j \leq n$;

that is, each row of P_σ has only one non-zero element. Thus, mass transported according to P_σ is not split, and the Kantorovich cost reduces to a Monge cost:

$$\sum_{i=1}^{n}\sum_{j=1}^{n} c(x_i, y_j) p_{\sigma,ij} = \frac{1}{n}\sum_{i=1}^{n} c(x_i, y_{\sigma(i)}).$$

This means that the optimal Kantorovich cost can be as small as the optimal Monge cost; that is,

$$\min_{P\in U(\alpha,\beta)} \sum_{i=1}^{n}\sum_{j=1}^{n} c(x_i, y_j) p_{ij} \le \min_{\sigma} \sum_{i=1}^{n}\sum_{j=1}^{n} c(x_i, y_j) p_{\sigma,ij} = \min_{\sigma} \frac{1}{n}\sum_{i=1}^{n} c(x_i, y_{\sigma(i)}).$$

A remarkable fact about the optimal transportation theory is that the optimal values for both costs are equal. This property follows from the combination of Choquet's and Birkhoff's theorems [12]. More specifically, Kantorovich's formulation (7) of the optimal transportation problem is an instance of a linear program, which admits minimizers at the extremal points of $U(\alpha, \beta)$ (Choquet), which are precisely permutation matrices (Birkhoff):

$$\min_{\sigma} \frac{1}{n}\sum_{i=1}^{n} c(x_i, y_{\sigma(i)}) = \min_{\sigma} \sum_{i=1}^{n}\sum_{j=1}^{n} c(x_i, y_j) p_{\sigma,ij} \le \min_{P\in U(\alpha,\beta)} \sum_{i=1}^{n}\sum_{j=1}^{n} c(x_i, y_j) p_{ij}.$$

This observation underscores the reason why the Kantorovich formulation, by not limiting to permutations, can potentially yield more efficient transportation plans, providing a lower minimal cost than the assignment problem restricted to permutation matrices.

Linear vs. nonlinear problem. Another significant improvement of the Kantorovich problem over the Monge problem is its transformation from a nonlinear optimization problem to a linear one. Unlike nonlinear problems, linear problems are very well understood from both theoretical and computational perspectives. For instance, linear optimization problems admit polynomial-time algorithms [5, 4].

Recall the objective function for the Monge problem is

$$\min_{\sigma} \sum_{i=1}^{n} c(x_i, y_{\sigma(i)}) a_i,$$

which yields a nonlinear cost and constraints with respect to the optimization variable σ. In contrast, the Kantorovich problem (7) has a linear cost and linear equality and inequality constraints.

Kantorovich problem between arbitrary measures. Kantorovich's formulation of the optimal transportation problem admits a seamless extension to arbitrary measures. Let $\alpha, \beta \in \mathcal{P}(\mathbb{R}^d)$ be arbitrary. Then we define

$$U(\alpha, \beta) = \{\pi \in \mathcal{P}(\mathbb{R}^{2d}) \ : \ x\#\pi = \alpha, \ y\#\pi = \beta\},$$

where

$$(x, y) \mapsto x, \quad (x, y) \mapsto y,$$

are the projections maps. The Kantorovich problem is then formulated as

$$\min_{\pi \in U(\alpha,\beta)} \int_{\mathbb{R}^{2d}} c(x,y) d\pi(x,y). \tag{8}$$

0.1.4 Kantorovich duality

Linear optimization problems are often called *linear programs*, which we will use in the sequel. Kantorovich introduced an incredibly fruitful and deep idea of duality for studying linear programs in general and optimal transportation problems in particular. Before diving into the applications of duality in optimal transportation problems, let us first consider a generic linear program.

General linear program. A linear program in the standard form is an optimization problem of the following form:

$$\min_{x \in \mathbb{R}^n} \quad c^\top x \quad \text{subject to} \quad Ax \leq b, \quad x \geq 0, \tag{9}$$

where $c \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, and the inequality $x \geq 0$ is element-wise.

A formal derivation of the dual linear program proceeds as follows.

- *Step 1.* Introduce the Lagrangian $L(x, \lambda)$ with dual variables λ for the inequality constraints:

$$L(x, \lambda) = c^\top x + \lambda^\top (Ax - b),$$

and consider the following equivalent formulation of (9)

$$\min_{x\geq 0}\max_{\lambda\geq 0}\left\{c^\top x+\lambda^\top(Ax-b)\right\}.$$

- *Step 2.* Formally interchange the order of $\min$ and $\max$:

$$\max_{\lambda\geq 0}\min_{x\geq 0}\left\{(c^\top+\lambda^\top A)x-\lambda^\top b\right\}.$$

- *Step 3.* Eliminate x by solving the inner problem:

$$\min_{x\geq 0}\left\{(c^\top+\lambda^\top A)x-\lambda^\top b\right\}=\begin{cases}-\lambda^\top b, & c^\top+\lambda^\top A\geq 0,\\ -\infty, & \text{otherwise,}\end{cases}$$

and obtain

$$\max_{\lambda\in\mathbb{R}^m}-\lambda^\top b\quad\text{subject to}\quad A^\top\lambda\geq -c,\quad \lambda\geq 0.$$

We say that there is no duality gap if the (optimal) values of primal and dual problems agree. For rigorous derivations and theorems we refer, for instance, to [3, Appendix A].

Kantorovich problem. Relying on Definition 0.1.5, for $\pi \geq 0$ we have that

$$\max_{f,g\in C_0(\mathbb{R}^d)} \left\{ \int_{\mathbb{R}^d} f(x)d\alpha(x) + \int_{\mathbb{R}^d} g(y)d\beta(y) - \int_{\mathbb{R}^{2d}} (f(x)+g(y))d\pi(x,y) \right\}$$
$$= \begin{cases} 0, & \pi \in U(\alpha,\beta), \\ +\infty, & \text{otherwise.} \end{cases}$$

Hence, introducing

$$\begin{aligned} \mathcal{L}(\pi,f,g) = \int_{\mathbb{R}^{2d}} c(x,y)\, d\pi(x,y) + \int_{\mathbb{R}^d} f(x)d\alpha(x) + \int_{\mathbb{R}^d} g(y)d\beta(y) \\ - \int_{\mathbb{R}^{2d}} (f(x)+g(y))d\pi(x,y), \end{aligned}$$

we obtain an equivalent formulation of (8):

$$\min_{\pi\geq 0} \max_{f,g\in C_0(\mathbb{R}^d)} \mathcal{L}(\pi,f,g)$$

As before, by formally interchanging the order of $\min$ and $\max$ we obtain

$$\max_{f,g\in C_0(\mathbb{R}^d)} \min_{\pi\geq 0} \mathcal{L}(\pi,f,g)$$

Reorganizing the terms in $\mathcal{L}$, we find that

$$\min_{\pi\geq 0} \mathcal{L}(\pi,f,g) = \begin{cases} \int_{\mathbb{R}^d} f(x)d\alpha(x) + \int_{\mathbb{R}^d} g(y)d\beta(y), \; f(x)+g(y) \leq c(x,y), \; \forall x,y, \\ -\infty, \quad \text{otherwise.} \end{cases}$$

Hence, the dual formulation of the Kantorovich formulation is

$$\max_{(f,g)\in R(c)} \int_{\mathbb{R}^d} f(x)d\alpha(x) + \int_{\mathbb{R}^d} g(y)d\beta(y), \tag{10}$$

where

$$R(c) = \{(f,g) \in C_0(\mathbb{R}^d) \times C_0(\mathbb{R}^d) : f(x) + g(y) \leq c(x,y),\ \forall x, y \in \mathbb{R}^d\}.$$

See [12, Chapter 1] for more details and rigorous results.

Complementary Slackness. Now we have the definition of primary problem 8 and of dual problem 10. Given an optimal coupling $\pi \in \mathcal{P}(\mathbb{R}^d)$ and optimal $f, g \in C_0(\mathbb{R}^d)$, the complementary slackness conditions are:

- If $(x, y) \in \operatorname{supp} \pi$ (i.e., mass is transported between x and y), then the corresponding dual constraint must be **tight**:

  $$f(x) + g(y) = c(x,y)$$

  This means that when mass is transported between points x and y, the sum of the dual variables at these points must exactly match the transportation cost $c(x, y)$.

- If $(x, y) \notin \operatorname{supp} \pi$ (i.e., no mass is transported between x and y), then the dual constraint can be **loose**:

$$f(x) + g(y) \leq c(x, y)$$

This means that if no transport occurs between x and y, the sum of the dual variables $f(x)$ and $g(y)$ can be strictly less than $c(x, y)$, and this is still valid.

Remark 0.1.2. *In the discrete case, $(x_i, y_j) \in \operatorname{supp} \pi$ is equivalent to $\pi_{ij} > 0$.*

Theorem 0.1.3. *[10] If π is a feasible solution to the primal, and (f, g) is a feasible solution to the dual, and they are complementary, then π and (f, g) must also be optimal solutions to their respective problems.*

Proof. Recall the primal and dual objectives:

$$\begin{aligned} \text{Primal:} \quad & \int_{\mathbb{R}^{2d}} c(x, y) \, d\pi(x, y), \\ \text{Dual:} \quad & \int_{\mathbb{R}^d} f(x) \, d\alpha(x) + \int_{\mathbb{R}^d} g(y) \, d\beta(y). \end{aligned}$$

The duality gap is defined as:

$$\text{Gap} = \int_{\mathbb{R}^{2d}} c(x, y) \, d\pi(x, y) - \left(\int_{\mathbb{R}^d} f(x) \, d\alpha(x) + \int_{\mathbb{R}^d} g(y) \, d\beta(y) \right).$$

By complementary slackness, we have

$$f(x) + g(y) = c(x, y).$$

Hence, the primal objective simplifies to:

$$\int_{\mathbb{R}^{2d}} c(x,y)\, d\pi(x,y) = \int_{\mathbb{R}^d} f(x)\, d\alpha(x) + \int_{\mathbb{R}^d} g(y)\, d\beta(y).$$

The duality gap is therefore:

$$\text{Gap} = \int_{\mathbb{R}^{2d}} c(x,y)\, d\pi(x,y) - \left(\int_{\mathbb{R}^d} f(x)\, d\alpha(x) + \int_{\mathbb{R}^d} g(y)\, d\beta(y) \right) = 0.$$

This proves that the duality gap is zero. □

0.2 Review of Computational Techniques for Optimal Transportation

0.2.1 North-West Corner Method

In the context of solving transportation problems, the North-West Corner Method (NWCM) is a widely used greedy algorithm that provides an initial feasible solution.[10] As a heuristic approach, it aims to find a solution quickly but does not guarantee optimality. The method begins by allocating as much of the supply as possible to the first destination in the transportation tableau, then proceeds by moving either rightward or downward, ensuring that the supply and demand constraints are met at each step.

Example 0.2.1. *Consider source* $a = (0.2, 0.5, 0.3)$ *and target* $b = (0.5, 0.1, 0.4)$.

The algorithm explores as following:

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \to \begin{pmatrix} 0.2 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \to \begin{pmatrix} 0.2 & 0 & 0 \\ 0.3 & 0.1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$\to \begin{pmatrix} 0.2 & 0 & 0 \\ 0.3 & 0.1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \to \begin{pmatrix} 0.2 & 0 & 0 \\ 0.3 & 0.1 & 0.1 \\ 0 & 0 & 0 \end{pmatrix} \to \begin{pmatrix} 0.2 & 0 & 0 \\ 0.3 & 0.1 & 0.1 \\ 0 & 0 & 0.3 \end{pmatrix}$$

Here is a detailed look at the algorithm:

Algorithm 1 North-West Corner Method (NWCM)

```
Initialize $i \leftarrow 1$, $j \leftarrow 1$, $r \leftarrow a_1$, $c \leftarrow b_1$
while $i \leq n$ and $j \leq m$ do
    Set $t \leftarrow \min(r, c, P_{ij})$
    Update $r \leftarrow a_i$ if $i \leq n$
    if $c = 0$ then
        Increment $j$
        Update $c \leftarrow b_j$
    end if
    Update $r \leftarrow r - t$
    Update $c \leftarrow s - t$
end while
```

Total number of floating-point operations per second (FLOPs) for NWCM is $O(n \cdot m)$, where n and m represent the number of rows and columns in the transportation problem, respectively. In particular, n represents the number of supply points, and m represents the number of target points. Hence, this algorithm is relatively computationally efficient, as it indicates a linear growth rate with respect to the product of the number of rows and columns. However, it is also apparent that the algorithm does not guarantee an optimal solution but instead explores all feasible solutions.

0.2.2 Network Simplex Algorithm

Building upon the greedy search is the Network Simplex Algorithm, which uses NWCM to find a solution first and next verifies its optimality. The discussion starts with definitions of foundational concepts.

What are networks in the context of optimal transport problems? In optimal transport problems, a network is represented by a directed graph $G = (V, E)$, where V consists of supply nodes (sources) and demand nodes (sinks), and E represents the possible paths (edges) for transporting mass between them. Each edge $(i, j) \in E$ is associated with:

- **Transportation cost $C_{i,j}$:** The expense of transporting one unit of mass from node i to node j.

- **Capacity $u_{i,j}$:** The maximum amount of mass that can be transported along edge (i, j). Capacity is often used in practical applications to model constraints like limited resources or physical restrictions, but in pure optimal transport, it is typically assumed to be unconstrained (infinite).

A feasible flow $P = \{P_{i,j}\}$ assigns a non-negative amount $P_{i,j}$ of mass to each edge (i, j) while satisfying the following constraints:

1. **Capacity constraint** (if defined): $P_{i,j} \leq u_{i,j}$, ensuring that the flow on each edge does not exceed its capacity.

2. **Flow conservation constraint:** The total outgoing flow from each source equals its supply, and the total incoming flow to each sink equals its demand:

   $$\sum_j P_{i,j} = a_i, \quad \sum_i P_{i,j} = b_j, \quad P_{i,j} \geq 0,$$

   where a_i is the supply at source i and b_j is the demand at sink j.

Convex Combination: A convex combination of two points P_1 and P_2 is defined as a linear combination of these two points where the coefficients are non-negative and sum to one. Mathematically, it is expressed as:

$$P = \lambda P_1 + (1 - \lambda) P_2, \quad 0 \leq \lambda \leq 1.$$

Extremal Point: An extremal point (or extremal solution) of a feasible region is a point that cannot be expressed as a convex combination of two distinct feasible points within the region.

Theorem 0.2.1. *[10] Let P be an extremal point of the polytope $U(a, b)$ and suppose that its corresponding set $S(P)$ of edges, denoted F, forms a graph $G(P) = (V \cup V', S(P))$. Then the graph $G(P)$ has no cycles. In particular, P cannot have more than $n + m - 1$ nonzero entries.*

Proof. (See *Computational Optimal Transport*, by Gabriel Peyré and Marco Cuturi [10], for more complete proof.) If the graph contains a cycle, it implies that one can remove one edge and still have a valid solution, meaning the solution P is not unique. In particular, P can be 'reached' by combining two distinct solutions or be written as a convex combination of two distinct solutions corresponding to the two trees formed by removing edges from the cycle. That contradicts that P is extremal. □

This theorem provides essential insight into the structure of extremal points in network flow problems, which is essential in the process of deleting not optimal solutions.

Network Simplex Algorithm begins with an extremal solution P, which can be obtained through a simple rule such as the North-West Corner Rule. Then, relying on Theorem 0.1.3, to prove coupling P is feasible, it's sufficient to obtain a solution (f, g) to the dual problem that is feasible and complimentary to P. This outlines two primary steps of Network Simplex Algorithm: first, find a pair of complementary dual variables to a extremal solution P; second, prove the pair is feasible.

After obtaining extremal solution P, the next step is to assign dual variables f and g, which satisfies the dual constraints in the network:

1. **Arbitrary Selection of Root Node:** Begin by selecting one node arbitrarily and setting its corresponding dual variable to 0. This node will serve as the "root" of the tree. Let the dual variable for the root node be $f_{\text{root}} = 0$.

2. **Traversal Using Search Algorithm and Assigning Dual Variables:** Using a breadth-first search (BFS) or depth-first search (DFS), traverse the

graph starting from the root node. When traverse the tree, assign values to the dual variables f_{i_k} (for source nodes) and g_{j_k} (for target nodes) for each edge (i_k, j_k) in the tree. The values are determined by the following systems of equations:

$$f_{i_1} + g_{j_1} = C_{i_1,j_1}, \quad f_{i_2} + g_{j_1} = C_{i_2,j_1}, \quad \dots, \quad f_{i_s} + g_{j_s} = C_{i_s,j_s}$$

that's derived based on the complementary slackness Theorem 0.1.3.

3. **Special Case $s < n + m - 1$:** Since P is the extremal solution, we have $s \leq n+m-1$. However, if the number of dual variables s is strictly smaller than $n+m-1$, the system is underdetermined. It means that there are more dual variables than independent constraints, resulting in multiple possible solutions for the dual variables. This can be resolved by arbitrarily fixing one dual variable, similar to the assignment to the root, and propagating the others along the tree structure.

After finding the complementary pair, the next step is to check if the dual solution is feasible and update the structure kick-out non-feasible solutions. If this condition, $f_i + g_j \leq C_{i,j} \quad$ for all edges (i, j), is violated for any edge, the violating edge (i, j) is added to the current graph and this addition may result in two possible scenarios:

1. **The Graph Remains a Tree:** If adding the violating edge (i, j) does not form a cycle, the graph remains a valid spanning tree. In this case, we can

proceed with the optimization process as usual, and no further adjustments are necessary to the primal solution P.

2. **A Cycle is Formed**: If adding the edge (i, j) creates a cycle, to maintain the acyclicity of the graph which is the requirement for extremal solution, we need to remove one edge from the cycle. The edge to remove is typically chosen based on criteria that ensure primal feasibility.After removing an edge from the cycle, we update the primal solution P as follows:

   $$P^{i_k,j_k}_{\text{new}} := P^{i_k,j_k} + \theta, \quad P^{i_{k+1},j_k}_{\text{new}} := P^{i_{k+1},j_k} - \theta \quad \forall k \leq l$$

   Here, θ represents the maximum possible increase to the flow along the positive edges in the cycle, while maintaining primal feasibility.

Following is the complete algorithm for the network simplex method :

Algorithm 2 Network Simplex Algorithm

```
Input: Network flow problem with capacities $C_{i,j}$, initial flow $P$, and initial dual variables $f, g$
Output: Optimal flow $P^*$ and dual variables $f^*, g^*$
Initialize the primal solution $P$ and dual solution $f, g$
while dual feasibility is violated for some edge $(i, j)$ do
    Add the violating edge $(i, j)$ to the graph
    if adding the edge creates a cycle then
        Remove one edge from the cycle to break the cycle
        Update the primal flow by adjusting the flow along the cycle
        Compute $\theta$ as the maximum feasible flow change along the cycle
        Update the primal flow $P$
    else
        Lift the indeterminacy by choosing values for undetermined dual variables
    end if
    Update the dual solution $f, g$ based on the new primal solution
end while
Return the optimal primal solution $P^*$ and dual solution $f^*, g^*$
```

The number of FLOPs for this algorithm is $O((n \cdot m)^2)$. Notably, this provides the upper-bound of the time complexity for this algorithm, and the number of pivots required often is much smaller than this worst-case bound. However, when dealing with very large-scale networks, this algorithm can still be very complex and time-consuming.

0.2.3 Auction Algorithm

The Auction Algorithm offers an efficient approach to solving the assignment problem in optimal transport by iteratively refining a solution that approximates the primal and dual problems' optimal solutions. This method is particularly effective due to its convergence properties and computational efficiency under certain conditions. We begin with necessary definitions and then outline the process of the algorithm.

Complementary Slackness: Recall the complementary slackness condition for the optimal transport problem is 0.1.3:

$$f_i^* + g_j^* = C_{i,j}, \quad \text{if } \pi_{i,j} > 0,$$

where:

- $\pi_{i,j}$ is the optimal transport plan, indicating the amount of mass transported between source i and target j,
- $C_{i,j}$ is the cost of transporting mass between i and j,
- f_i^* and g_j^* are optimal dual variables corresponding to the source and target, respectively.

ϵ-Complementary Slackness: To allow for approximate solutions, the complementary slackness condition can be relaxed. The ϵ-complementary slackness condition is defined as:

$$C_{i,j} - g_j \leq \min_{j'}(C_{i,j'} - g_{j'}) + \epsilon, \quad \text{if } \pi_{i,j} > 0,$$

where:

- $\pi_{i,j}$ is the transport plan, specifying the mass transported between source i

and target j,

- $C_{i,j}$ is the transportation cost between source i and target j,
- g_j and $g_{j'}$ are dual variables corresponding to the target indices j and j',
- $\epsilon > 0$ is the allowed margin of error in the dual feasibility condition.

This condition ensures that the reduced cost for a transported pair (i, j), where $\pi_{i,j} > 0$, is within an ϵ-margin of the minimum reduced cost across all potential targets for i.

Algorithm Process: The auction algorithm adjusts the assignments and dual variables by following these key steps to maintain ϵ-complementary slackness throughout iterations:

1. Initialize with an empty set S and vectors $g = 0$ and ξ as empty.
2. Update g_i by setting it to the difference between the lowest and second-lowest adjusted costs:

$$g_i \leftarrow g_i - (\min_{j \neq i}(C_{i,j} - g_j) - (C_{i,\xi_i} - g_i)) - \epsilon.$$

3. Update S and ξ by removing or adding indices based on the fulfillment of ϵ-complementary slackness.
4. Repeat the process until all elements satisfy the ϵ-complementary slackness condition.

The algorithm's efficiency is highlighted by the proof that it converges in at most $N = \frac{n\|C\|_\infty}{\epsilon}$ iterations, where n is the number of elements and $\|C\|_\infty$ is the maximum cost.

Proof. The dual variable g_i starts at 0 and decreases at most by $-\|C\|_\infty$, the maximum range of the cost matrix C. Therefore, the total adjustment across all g_i is bounded by $n \cdot \|C\|_\infty$. as there are n elements. Meanwhile, each iteration decreases the dual variable g_i by at least ϵ.

Since the total adjustment required is bounded by $n \cdot \|C\|_\infty$, and each iteration makes progress of at least ϵ, the total number of iterations required is bounded by $\frac{n \cdot \|C\|_\infty}{\epsilon}$ (see [10, Chapter 3] for more details and rigorous proof.)

The complete algorithm for this method is as follows:

Algorithm 3 Auction Algorithm

```
Input: Cost matrix $C \in \mathbb{R}^{n \times n}$, slack parameter $\epsilon > 0$
Output: Assignment $\xi$ and dual variables $g$ satisfying $\epsilon$-complementary slackness
Initialize $S \leftarrow \emptyset$, $\xi \leftarrow \emptyset$, and $g_i \leftarrow 0$, $\forall i \in \{1, \ldots, n\}$
while $S \neq \{1, \ldots, n\}$ do
    for each $i \notin S$ do
        Compute: $j_1 \leftarrow \arg\min_j (C_{i,j} - g_j)$,    $j_2 \leftarrow \arg\min_{j \neq j_1} (C_{i,j} - g_j)$
        Update dual variable $g_i$ as: $g_i \leftarrow g_i - ((C_{i,j_2} - g_{j_2}) - (C_{i,j_1} - g_{j_1})) - \epsilon$
        Assign: $\xi_i \leftarrow j_1$
        Update the set $S \leftarrow S \cup \{i\}$
    end for
end while
Return $\xi$ and $g$
```

0.3 Limitations of Current Works

While the Network Simplex Algorithm and Auction Algorithm are widely used and effective for solving network flow and assignment problems, they exhibit several limitations when applied to large-scale or complex networks.

Network Simplex Algorithm The Network Simplex Algorithm is a specialized approach for solving network flow problems, which share foundational similarities with optimal transportation problems. In both cases, the objective is to minimize a cost function while satisfying flow conservation constraints: in network flow problems, these constraints represent the balance of flow across nodes; in optimal transportation, they correspond to satisfying supply at source nodes and demand at target nodes. The network in the context of the optimal transportation problem represents a graph where the nodes correspond to supply points (sources) and demand points (sinks), and the edges correspond to potential paths for transporting mass, with associated costs.

While the Network Simplex Algorithm is effective for many network flow problems, it has certain limitations when applied to large-scale or complex instances of optimal transportation. Its worst-case time complexity of $O((nm)^2)$, where n is the number of nodes and m is the number of edges, can become computationally expensive for large transportation networks with many sources and sinks. Although the number of pivot steps is typically much smaller in practice, the algorithm's performance still depends heavily on the structure of the network and the initial solution provided. Poor choices of pivoting rules can lead to a large number of iterations, reducing overall efficiency. These limitations highlight the challenges of directly applying network flow techniques like the Network Simplex Algorithm to very large or high-dimensional optimal transportation problems, where computational resources may be constrained.

Auction Algorithm The Auction Algorithm also has notable limitations:

- **Approximation by ϵ:** The algorithm produces an ϵ-optimal solution, meaning the result is not guaranteed to be exact. While ϵ can be reduced to increase precision, this comes at the cost of increased computational time due to more iterations being required.

- **Scalability for Large Problems:** The time complexity of $O(n^2 \cdot \frac{\|C\|_\infty}{\epsilon})$ makes the auction algorithm computationally expensive for large-scale problems, particularly when the cost matrix C is dense or has high variance.

- **Dependency on Cost Matrix Structure:** Unlike the network simplex algorithm, the auction algorithm does not leverage sparsity or specific structures of the cost matrix, which can lead to inefficiencies in cases where such structures could otherwise be exploited.

- **Iterative Nature:** The iterative dual updates can make the convergence slower in cases where the cost matrix C has widely varying or high-cost entries, resulting in a higher number of iterations to reduce the slack to ϵ.

Comparison of Limitations While both algorithms share the goal of solving assignment and network flow problems, they exhibit contrasting limitations:

- The Network Simplex Algorithm achieves exact solutions but suffers from high worst-case complexity and sensitivity to the pivoting rule.
- The Auction Algorithm, by contrast, offers a more flexible trade-off between computational effort and solution precision via the ϵ-parameter but can be inefficient for dense or large-scale networks due to its dependency on the cost matrix and iterative dual updates.

Broader Challenges in Current Works Beyond the specific limitations of these algorithms, broader challenges exist in the field:

- **Scalability to Very Large Networks:** Neither algorithm scales efficiently to massive networks with millions of nodes and edges, which are common in modern applications like logistics, social networks, and supply chains.
- **Numerical Stability:** Both algorithms can face numerical issues in cases of highly skewed cost matrices or degenerate solutions, leading to challenges in maintaining accuracy and convergence.
- **Dynamic and Stochastic Settings:** Most traditional algorithms, including the Network Simplex and Auction Algorithms, assume static input data and cannot efficiently handle dynamic or stochastic changes in the network, which are increasingly relevant in real-world applications.

These limitations highlight the need for more scalable, robust, and adaptive approaches to solving network flow and assignment problems in the context of modern large-scale and dynamic systems.

Alternative Transportation Distances

In exploring the computational challenges and applications of optimal transport (OT), it becomes evident that alternative approaches are necessary to address the inherent complexity and scalability issues, particularly in high-dimensional settings. This section examines alternative transportation distances and methodologies that aim to approximate or simplify OT computations while retaining key properties of the original framework. Techniques such as Sliced Wasserstein Distances, Linearized Optimal Transportation Distances, and no-collision maps have emerged as promising tools to reduce computational cost and adapt OT principles to specific problem contexts. These alternatives pave the way for more efficient solutions, bridging the gap between theoretical elegance and practical feasibility, and serve as a precursor to the no-collision methods discussed in the subsequent section.

For Wassertein and Sliced Wasserstein distances we follow closely the exposition in [10, 6]; for Linearized Optimal Transportation we follow [6] and references therein; for No-collision Transportation distances we follow [9, 8].

0.4 Wasserstein Metric

0.4.1 Intuition Behind Wasserstein Metric

The pseudo-inverses map cumulative probabilities z to the corresponding quantiles of the distributions. The p-norm of the differences between these quantiles captures the transportation cost between distributions. This makes the one-dimensional p-Wasserstein metric computationally practical and conceptually clear, as it avoids the geometric complexities of higher dimensions.

0.4.2 Concept Definitions

Definition 0.4.1. *The p-Wasserstein metric W_p is defined as:*

$$W_p(I_0, I_1) = \left(\inf_{\pi \in \mathcal{M}} \int_{\Omega \times \Omega} |x - y|^p \, d\pi(x, y) \right)^{1/p}.$$

where:

- *π is a transport plan that matches I_0 (source distribution) to I_1 (target distribution).*
- *$|x - y|^p$ is the cost of moving mass from x to y.*

The Wasserstein metric measures the “minimum effort” required to transform one probability distribution into another. The cost function $|x - y|^p$ quantifies the "distance" of transport between locations x and y, while the transport plan π determines how the probability mass is reallocated. The infimum ensures that the total transport cost is minimized, resulting in an optimal transport plan.

Definition 0.4.2. *The CDF $F_i(x)$ of a probability distribution $I_i(x)$ is defined as:*

$$F_i(x) = \int_{-\infty}^{x} I_i(t)dt, \quad i = 0, 1.$$

Here, $F_i(x)$ measures the total probability up to point x, and it is a non-decreasing function that goes from 0 to 1 as $x \to \infty$. The CDF encodes the cumulative distribution of mass, providing a complete description of the probability distribution in one dimension.

Definition 0.4.3. *The pseudo-inverse $F_i^{-1}(z)$ of the CDF is defined as:*

$$F_i^{-1}(z) = \inf\{x \in \Omega : F_i(x) \geq z\}, \quad z \in [0, 1].$$

where z represents a cumulative probability, and $z \in [0, 1]$.

The pseudo-inverse $F_i^{-1}(z)$ maps a cumulative probability z (ranging from 0 to 1) to the corresponding quantile in the distribution. It provides a direct way to work with the "quantile" of a probability distribution and is particularly useful for defining the Wasserstein metric in one dimension.

One-Dimensional Case: In one dimension, the ordering of points along the real line is natural and unique. The z-quantile of one distribution can be directly matched to the z-quantile of another without ambiguity. This property ensures that the p-Wasserstein metric has the following closed-form solution:

$$W_p(I_0, I_1) = \left(\int_0^1 |F_0^{-1}(z) - F_1^{-1}(z)|^p dz \right)^{1/p}.$$

The closed-form solution avoids the need to explicitly solve the optimization problem, making the computation more efficient. By leveraging the pseudo-inverses of the cumulative distribution functions (CDFs) $F_0^{-1}(z)$ and $F_1^{-1}(z)$, we compute the difference between quantiles and integrate over $z \in [0, 1]$.

0.4.3 Advantages of Wasserstein Metric

- The solution reduces the problem from solving an optimization over all possible transport plans to evaluating a single integral in one dimension.
- The monotonic structure of one-dimensional distributions ensures that aligning quantiles directly minimizes the transport cost.

0.5 Sliced Wasserstein Matrix

The Sliced-Wasserstein Metric (SW) is an approach to approximate the Wasserstein distance efficiently, especially in high-dimensional spaces. In particular, Any high-dimensional probability distribution $I(x)$ in $\mathbb{R}^d$ can be "sliced" into one-dimensional distributions by projecting along a direction θ on the unit sphere S^{d-1}. The projection is performed using the Radon Transform.

Definition 0.5.1. *The Radon Transform is defined as:*

$$\mathcal{R}I(t,\theta) = \int_{\mathbb{R}^d} I(x)\delta(t - x\cdot\theta)dx,$$

where t represents the position along the 1D projected space and δ is the Dirac delta function.

The Sliced-Wasserstein Metric works as follows:

1. **Projection:** Project the high-dimensional distributions I_0 and I_1 onto a series of one-dimensional subspaces parameterized by $\theta \in S^{d-1}$ using the Radon Transform.

2. **1D Wasserstein Distance:** Compute the Wasserstein metric W_p for each 1D projection:

$$W_p(\mathcal{R}I_0(\cdot,\theta), \mathcal{R}I_1(\cdot,\theta)).$$

3. **Aggregate:** Integrate the p-th powers of these distances across all directions θ:

$$SW_p(I_0, I_1) = \left(\int_{S^{d-1}} W_p^p(\mathcal{R}I_0(\cdot,\theta), \mathcal{R}I_1(\cdot,\theta))d\theta\right)^{1/p}.$$

In practice, this integral is approximated by sampling a finite number of directions θ from S^{d-1}.

0.6 Linearized Optimal Transportation (LOT)

Linearized Optimal Transportation (LOT) is a method designed to approximate the Wasserstein metric by embedding probability distributions into a linear space, allowing for computationally efficient operations such as addition, subtraction, and projection. This method is particularly useful when working with high-dimensional distributions, where direct computation of the Wasserstein distance becomes computationally expensive.

0.6.1 Intuition Behind LOT

Linearized Optimal Transportation approximates the Wasserstein geometry by working in the *tangent space* of a reference distribution I_0:

- The tangent space T_{I_0} is a linear space that locally approximates the curved Wasserstein space at I_0.
- Instead of solving the full optimal transport problem in the nonlinear Wasserstein space, the distributions are mapped to this linear tangent space, where computations such as distances or projections are simpler.

0.6.2 Concepts Definitions

Definition 0.6.1. *Given a base distribution I_0, the Linearized Optimal Transportation framework defines a linear embedding of a probability distribution I*

into a tangent space at I_0, denoted as T_{I_0}. This embedding is expressed as:

$$\Phi_{I_0}(I) = \nabla\phi,$$

where ϕ is the optimal transport potential that satisfies:

$$I = \nabla\phi \# I_0.$$

LOT Distance: The LOT distance between two distributions I_1 and I_2 with respect to a reference distribution I_0 is computed as:

$$\text{LOT}(I_1, I_2; I_0) = \|\Phi_{I_0}(I_1) - \Phi_{I_0}(I_2)\|_{L^2(I_0)},$$

where:

- $\Phi_{I_0}(I)$ represents the embedding of the distribution I into the tangent space T_{I_0}.
- The norm $\|\cdot\|_{L^2(I_0)}$ is computed with respect to the reference measure I_0.

0.6.3 Procedures of Linearized Optimal Transportation

The LOT framework involves the following steps:

1. **Reference Distribution:** Select a reference distribution I_0 around which the tangent space is constructed.

2. **Optimal Transport Map:** Compute the optimal transport map between I_0 and the target distributions I_1 and I_2.

3. **Embedding:** Map the distributions I_1 and I_2 into the tangent space T_{I_0} using the transport map's gradient.

4. **LOT Distance:** Compute the distance between the embedded distributions using the L^2 norm with respect to I_0.

0.6.4 Analysis of LOT

Advantages of LOT:

- **Linearization:** LOT maps nonlinear Wasserstein geometry into a linear tangent space, enabling fast and efficient computations.
- **Scalability:** By avoiding the need to solve multiple nonlinear optimization problems, LOT is computationally efficient for high-dimensional or large-scale datasets.
- **Applications:** LOT is particularly useful in applications such as shape analysis, generative modeling, and time-series analysis, where one needs to efficiently compare distributions.

Limitations of LOT: While LOT is computationally efficient, it is an approximation of the true Wasserstein metric and has the following limitations:

- The quality of the approximation depends on the choice of the reference distribution I_0. If I_0 is far from the distributions being compared, the results may lose accuracy.

- LOT assumes that the distributions lie close to the tangent space of I_0. For highly nonlinear Wasserstein spaces, this assumption may not hold.

In summary, Linearized Optimal Transportation is a powerful tool for approximating the Wasserstein metric, offering computational efficiency while maintaining the core geometric properties of optimal transport. It is particularly effective in scenarios where high-dimensional data needs to be compared quickly and accurately.

0.7 No-Collision Transportation Maps

Here, we discuss so-called no-collision transportation maps and follow closely the exposition of the subject in [9, 8].

0.7.1 Intuition Behind No-Collision

The no-collision property is of paramount importance because it guarantees that the transportation map preserves the structure of the original measure without overlap. Optimal transportation maps inherently satisfy the no-collision property, as they aim to map one distribution onto another while minimizing transportation cost.

However, computing optimal transport maps is often computationally intensive. Thus, focusing on the no-collision property allows us to relax the strict optimality condition in favor of simpler, computationally feasible maps, while still ensuring the preservation of separation between points in the transformed space.

0.7.2 Concepts Definition

A map $T : \Omega \subset \mathbb{R}^d \to \mathbb{R}^d$ is called a *no-collision map* if it satisfies the condition that for any pair of distinct points $x_1, x_2 \in \Omega$, the images $T(x_1)$ and $T(x_2)$ do not collide, i.e., they are distinct in the transformed space. More formally, there exists a separation between $T(x_1)$ and $T(x_2)$, ensuring that they do not overlap after the transformation.

In the context of transportation, a *no-collision transportation map* is a map T that pushes one measure μ onto another measure ν, while preserving the no-collision property. In other words, for any pair of distinct points x_1 and x_2 in the support of μ, their images $T(x_1)$ and $T(x_2)$ under the map T should remain distinct in the support of ν, thus ensuring no collision between transported points. This is crucial for ensuring that the transformation preserves the structure of the original measure without any overlap.

A map $T : \Omega \subset \mathbb{R}^d \to \mathbb{R}^d$ is *half-space-preserving* if, for any $x_1 \neq x_2 \in \Omega$, there exists a unit vector $v \in S^{d-1}$ such that:

$$x_2 \cdot v - x_1 \cdot v \geq 0, \quad T(x_2) \cdot v - T(x_1) \cdot v \geq 0,$$

and at least one of these inequalities holds strictly.

Theorem 0.7.1. *A map $T : \Omega \subset \mathbb{R}^d \to \mathbb{R}^d$ is half-space-preserving if and only if it satisfies the no-collision property.*

Intuitively, a half-space-preserving map ensures that for any two distinct points x_1 and x_2, there exists a pair of parallel hyperplanes separating x_1 and x_2 as well as their images $T(x_1)$ and $T(x_2)$. Furthermore, the separation respects the original ordering: x_i and $T(x_i)$ lie on the same side of their respective hyperplanes.

Theorem 0.7.1 provides an essential result: it gives us a useful framework for constructing transportation maps that are simpler but still preserve the crucial separation properties, as this theorem guarantees that maps satisfying this simpler criterion will still preserve the no-collision property.

0.7.3 Construction of No-Collision Maps

To construct no-collision maps, we begin by selecting hyperplanes that divide the points in the original space. These hyperplanes ensure that the transformed points do not overlap. The goal is to choose partitioning directions that create the necessary separation between points while maintaining the structure of the space.

The construction procedure involves partitioning the space based on these selected directions, ensuring that the images $T(x_1)$ and $T(x_2)$ of any distinct points x_1 and x_2 remain distinct after the transformation. By applying this partitioning strategy, we can ensure the no-collision property while keeping the map computationally tractable.

Previous Work on No-Collision Maps. Previous approaches to no-collision transportation maps often rely on simplifying the problem by partitioning the probability measure into distinct subregions. Specifically, these methods perform either *strictly vertical* or *strictly horizontal* splits of the domain Ω, dividing it into disjoint sections. Each subregion is then mapped to a corresponding portion of the target measure, ensuring that the map T avoids any overlap or intersection of the transported mass.

In this framework:

- A vertical split divides Ω along hyperplanes parallel to the y-axis, resulting in a partition of the domain into vertical strips.
- A horizontal split partitions Ω along hyperplanes parallel to the x-axis, dividing the domain into horizontal layers.

These partitioning strategies provide a simple yet effective way to ensure the no-collision property in the transformation.

Once partitioned, the subregions are paired using a prescribed matching strategy, typically aligning mass in one region to its corresponding counterpart in a manner that respects the no-collision property. Figure 1 is an illustration from [8] showing how the partition and transporting work for probability measures given by grid functions of their probability distributions:

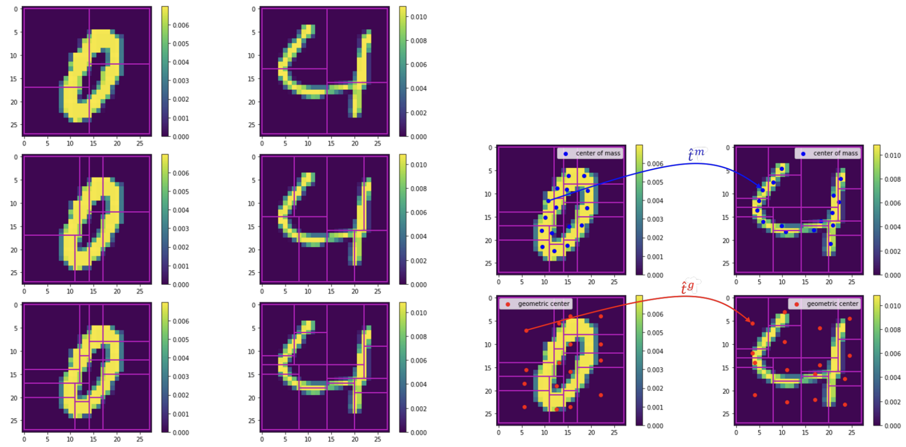

Figure 1: Illustration of No-Collision Transport Map

While effective for certain settings, these strictly axis-aligned splits are limited in flexibility and may not minimize transportation cost or fully leverage the geometry of the distributions. The limitations of axis-aligned methods motivate the exploration of more general partitioning strategies, as discussed in the next section. This approach seeks to improve the flexibility and efficiency of no-collision maps while maintaining their core property of avoiding transported mass overlap.

Theoretical Framework

This section establishes the theoretical underpinnings of two novel partitioning techniques—Random Split and PCA-based Split—that extend traditional horizontal and vertical partitioning methods. These advanced techniques are designed to enhance the flexibility and robustness of the partitioning process while maintaining the no-collision property. The discussion focuses on the conceptual and mathematical principles of each method, comparing their theoretical robustness against the conventional splits. Computational implementation details are deferred to the subsequent section.

0.8 Traditional Horizontal and Vertical Splits

0.8.1 Rationale

Traditional partitioning techniques involve splitting the dataset along fixed axes—either horizontally (along the x-axis) or vertically (along the y-axis). These methods are straightforward and computationally efficient, making them suitable for balanced and uniformly distributed data.

0.8.2 Theoretical Robustness

While horizontal and vertical splits are simple to implement, their rigidity can lead to suboptimal partitioning in datasets with complex or anisotropic distributions. These methods are susceptible to biases introduced by the alignment of data patterns with the fixed splitting axes, potentially resulting in unbalanced partitions and reduced generalizability across diverse datasets. The lack of adaptability limits their robustness in handling varied data structures, especially in high-dimensional or irregularly distributed datasets.

0.9 Random Split

0.9.1 Rationale

The Random Split method enhances traditional horizontal and vertical splits by introducing a stochastic rotation matrix to determine the splitting direction. Specifically, each time before performing a horizontal or vertical split, the dataset is rotated by a randomly generated rotation matrix. This approach maintains the no-collision property by applying the same rotation matrix to both source and target datasets at each recursive step. The incorporation of randomness ensures that the partitioning process does not favor any fixed direction, thereby avoiding biases introduced by inherent data patterns or anomalies.

0.9.2 Visualization

The normal split method shown in the plot below involves first applying a rotation along a randomly chosen direction. This transformation ensures that the structure of the source and target points is adjusted by the same direction and hence the no-collision feature. In the second (right) plot, both source and target points are projected onto the chosen direction, with their respective medians (blue and red dashed lines) marked to indicate potential partition boundaries. This approach introduces randomness to explore different splitting scenarios. By comparing the source and target medians in the rotated space, this method enables an effective division of data while preserving the geometric relationships in the original points.

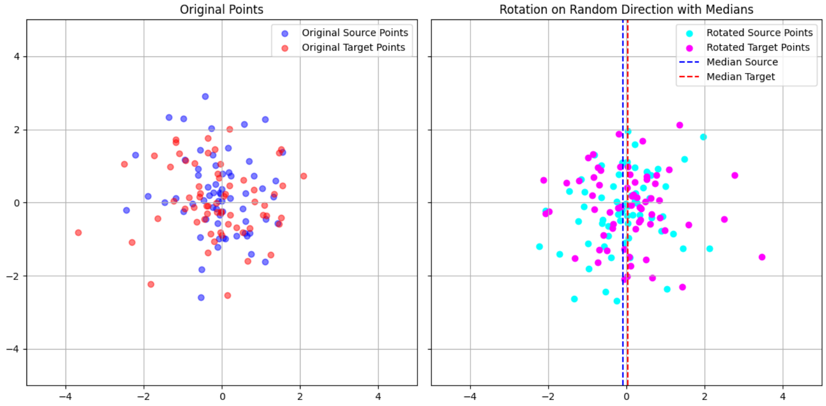

Figure 2: Normal Distribution before/after random rotation

0.9.3 Theoretical Robustness

The theoretical robustness of the Random Split method stems from its ability to uniformly explore a wide range of splitting directions through random rotations. By avoiding reliance on predefined axes, the method mitigates the risk of biased partitions that can occur when data align with specific directions. This stochastic approach ensures that the partitioning process remains flexible and adaptable to diverse and complex data distributions. Additionally, the uniform sampling of splitting directions enhances the method's generalizability, making it resilient against structural anomalies and heterogeneous data patterns. The preservation of the no-collision property across varied rotation scenarios further solidifies its theoretical soundness, ensuring consistent performance regardless of the underlying data structure.

0.10 PCA-based Split

0.10.1 Rationale

The PCA-based Split method advances beyond both traditional and Random Split techniques by aligning the partitioning direction with the principal component of the combined source and target datasets. This is achieved by recentering and combining the datasets before computing the principal eigenvector of the covariance matrix, which serves as the splitting axis. Unlike the Random Split, which uses stochastic rotations, the PCA-based approach dynamically adapts the splitting direction based on the data's intrinsic variance structure. Regularization is incorporated into the covariance matrix to ensure numerical stability, particularly in cases of low variance or near-singular configurations.

0.10.2 Visualization

Below are an simple illustration of how PCA-based split work, in the first step. Here, normally distributed data is applied.

The PCA-based partitioning involves identifying the principal component of the combined source and target datasets, as illustrated by the dashed line in the first plot. This principal component represents the direction of maximum variance across the points. By rotating both source and target points by this vector, the dimensionality is reduced and the data is aligned along a single axis, as shown in the second plot.

After rotation, the median of the new combine data set is found, illustrated as the blue dotted line in the second plot, which is then used to split the data into lower and upper halves.

0.10.3 Theoretical Robustness

The PCA-based Split method achieves enhanced theoretical robustness through its data-driven approach to determining splitting directions. By aligning partitions with the principal components, the method ensures that splits capture the most significant variance in the data, leading to more meaningful and balanced

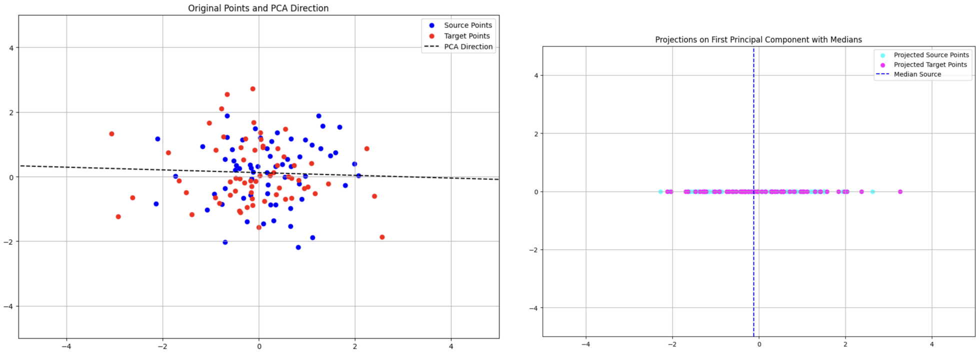

Figure 3: Normal Distribution before/after PCA rotation

partitions. This adaptability allows the method to respond dynamically to evolving data subsets, preventing the exhaustion of informative splitting directions and maintaining relevance throughout the recursive partitioning process. The combination of source and target datasets before PCA ensures that the splitting direction respects the relational structure between them, thereby preserving the no-collision property. Furthermore, the introduction of a regularization term in the covariance matrix enhances numerical stability, making the method resilient to irregularities and ensuring consistent performance across various data conditions. Compared to traditional and Random Splits, the PCA-based method offers superior adaptability and resilience, particularly in handling anisotropic and high-dimensional datasets.

0.11 Comparative Analysis of Robustness

When comparing the robustness of the Random Split and PCA-based Split methods to previously developed horizontal and vertical splits, several key distinctions emerge:

- **Adaptability to Data Structure:** Traditional splits are limited by their fixed axes, making them less adaptable to complex data distributions. In contrast, both Random Split and PCA-based Split dynamically adjust their splitting directions—Random Split through stochastic rotations and PCA-based Split through data-driven principal components—thereby accommodating a wider variety of data structures.

- **Bias Mitigation:** Traditional splits can inadvertently introduce biases if data patterns align with the splitting axes. Random Split mitigates this by uniformly sampling splitting directions, while PCA-based Split reduces bias by aligning splits with the direction of maximum variance, ensuring that partitions are informed by the data's inherent structure.

- **Generalizability:** The flexibility inherent in Random Split and PCA-based Split enhances their generalizability across different datasets. Traditional splits may perform well on uniformly distributed data but struggle with heterogeneous or anisotropic datasets. The Random and PCA-based methods maintain robust performance across diverse scenarios, making them more versatile in practical applications.

- **Preservation of No-Collision Property:** All three methods uphold the no-collision property, ensuring consistent and collision-free mappings. However, the Random and PCA-based methods achieve this while also enhancing partitioning robustness, whereas traditional splits may require additional considerations to maintain this property under complex data distributions.

Computational Results

This section presents the computational validation of the proposed Random Split and PCA-based Split methods, compared against traditional horizontal and vertical splits and an optimal transport solution obtained through linear programming. The experiments evaluate the methods across varying point distributions, assessing partition quality, computational efficiency, and relative performance in pairing costs.

We build on the methods developed in [9], including and the code developed by A. Iannantuono available at the github repository https://github.com/armeehn/no-collision-transportation-maps.

0.12 Experimental Settings

Dataset Characteristics Synthetic datasets of $2^6 = 64$ points were generated, with coordinates drawn from two different distributions:

- **Normal Distribution:** Points sampled independently from a standard normal distribution.

- **Exponential Distribution:** Points sampled independently from an exponential distribution with a scale parameter of 1.

Each dataset contains equal numbers of source and target points.

Evaluation Metrics To compare the methods, the following metrics were computed:

- **Total Cost:** The sum of pairing costs (e.g., squared Euclidean distances) for all paired points. This is the primary metric reported in computational results.
- **Relative Error (%):** The percentage difference in total cost between each method and the optimal transport linear programming (OTLP) solution. It highlights how much worse each method performs relative to OTLP.
- **Cost Ratio:** The ratio of each method's total cost to the OTLP cost, serving as a normalized measure of performance.

Optimal Transport Plan An optimal transport plan was computed using the Earth Mover's Distance (EMD) via linear programming, ensuring a global optimal pairing cost. This served as the benchmark for relative error and cost ratio calculations.

0.13 Methodology Implementation

The implementation of all methods—Random Split, PCA-based Split, and traditional horizontal and vertical splits—follows a unified framework that combines recursive splitting and pairing. This general structure ensures a consistent approach while leveraging each method's unique splitting strategy, as detailed in the theoretical section.

Splitting Process At the core of the implementation is the recursive splitting process, which partitions the source and target datasets into lower and upper subsets at each step. To ensure consistency and preserve the no-collision property:

- **Partitioning Direction:** The direction of the split (e.g., random rotation, principal component, or fixed axis) determines how points are projected for partitioning.

- **Median-Based Splits:** The projection values of combined source and target points are computed along the chosen direction, and a median-based threshold is used to split the points into lower and upper subsets.

- **Handling Edge Cases:** When multiple points share the median value or subsets are empty, alternative thresholds (e.g., mean) are used to ensure balanced partitions.

Recursive Pairing The pairing process relies on a registry system to encode the hierarchical paths of points in the partitioning tree. The steps are as follows:

- **Registry Updates:** During each split, points in the lower subset are assigned a binary value of 0, and those in the upper subset are assigned 1. These values are recursively appended to the registry, creating a unique binary identifier for each point.

- **Path Matching:** The final registry values represent the paths taken by points during recursive splitting. Points in the source and target datasets are matched by aligning their binary paths, ensuring a consistent pairing that preserves the no-collision property.

- **Registry Sorting:** After all splits, the registries are sorted based on their binary paths to facilitate efficient pairing of source and target points.

Example 0.13.1. *PCA-based Split For illustration, the PCA-based Split implementation uses the following steps:*

1. *Combine source and target points to compute the covariance matrix and determine the principal eigenvector, which serves as the splitting axis.*

2. *Project points onto this axis, compute the median projection value, and partition the points into lower and upper subsets.*

3. *Recursively repeat the process for each subset, updating the registries at each step.*

4. *After all splits, sort the registries and generate pairings based on aligned paths.*

0.13.1 Pseudo-code

Below is the pseudo-code for this method.

Algorithm 4 Rotation and Split for No-Collision Partitioning

```
Require: Source points $\mathcal{S}$, Target points $\mathcal{T}$, Rotation matrix $R$, Median splitting direction $d$
Ensure: Partitioned sets $(\mathcal{S}_1, \mathcal{S}_2)$ and $(\mathcal{T}_1, \mathcal{T}_2)$
Apply rotation $R$ to both $\mathcal{S}$ and $\mathcal{T}$:
        $\mathcal{S}' \leftarrow R \cdot \mathcal{S}$
        $\mathcal{T}' \leftarrow R \cdot \mathcal{T}$
Compute median along direction $d$ for $\mathcal{S}'$:
        $m_S \leftarrow \text{Median}(\mathcal{S}'_d)$
Compute median along direction $d$ for $\mathcal{T}'$:
        $m_T \leftarrow \text{Median}(\mathcal{T}'_d)$
Partition $\mathcal{S}'$ based on $m_S$:
        $\mathcal{S}_1 \leftarrow \{s \in \mathcal{S}' : s_d \leq m_S\}$
        $\mathcal{S}_2 \leftarrow \{s \in \mathcal{S}' : s_d > m_S\}$
Partition $\mathcal{T}'$ based on $m_T$:
        $\mathcal{T}_1 \leftarrow \{t \in \mathcal{T}' : t_d \leq m_T\}$
        $\mathcal{T}_2 \leftarrow \{t \in \mathcal{T}' : t_d > m_T\}$
Return partitioned sets $(\mathcal{S}_1, \mathcal{S}_2)$ and $(\mathcal{T}_1, \mathcal{T}_2)$
```

0.14 Results

0.14.1 Exponential Distribution

The experiments with datasets generated from an exponential distribution (scale parameter = 1) reveal significant differences in pairing costs among the evaluated methods, illustrate in Table 0.14.1. As expected, the optimal transport linear programming (OTLP) solution achieves the lowest total cost, serving as the benchmark with a minimal cost of 0.3624. The traditional horizontal and vertical (HV) split method incurs a total cost of 0.7936, resulting in a relative error of 118.99% and a cost ratio of 2.19, providing the best performance among the heuristic approaches. In contrast, the Random Split method shows a much higher total cost of 3.1275, with a relative error of 762.97% and a cost ratio of 8.63. The PCA-based Split method, while better than the Random Split, achieves a total cost of 1.3785, with a relative error of 280.37% and a cost ratio of 3.80, reflecting a moderate increase in cost compared to the OTLP solution.

Method	**Total Cost**	**Rel. Error (%)**	**Cost Ratio**
OTLP	0.3624	0.00	1.00
HV	**0.7936**	118.99	**2.19**
Random	**3.1275**	762.97	**8.63**
PCA	**1.3785**	280.37	**3.80**

Table 1: Pairing Costs and Performance Metrics for Exponential Distribution

Visualization of Pairings To further illustrate the differences in pairing strategies, the pairings generated by each method for this exponential distribution are visualized in Figure 4. This visualization highlights how each method approaches the problem of matching points within the distribution, with the OTLP method showing the most efficient pairings, leading to the lowest total cost. In contrast, the HV method exhibit more dispersed pairings but still successfully maintain most pairings. The PCA-based Split method, while an improvement over the Random Split, still shows some inefficiencies in comparison to OTLP, as seen in the distribution of pairings.

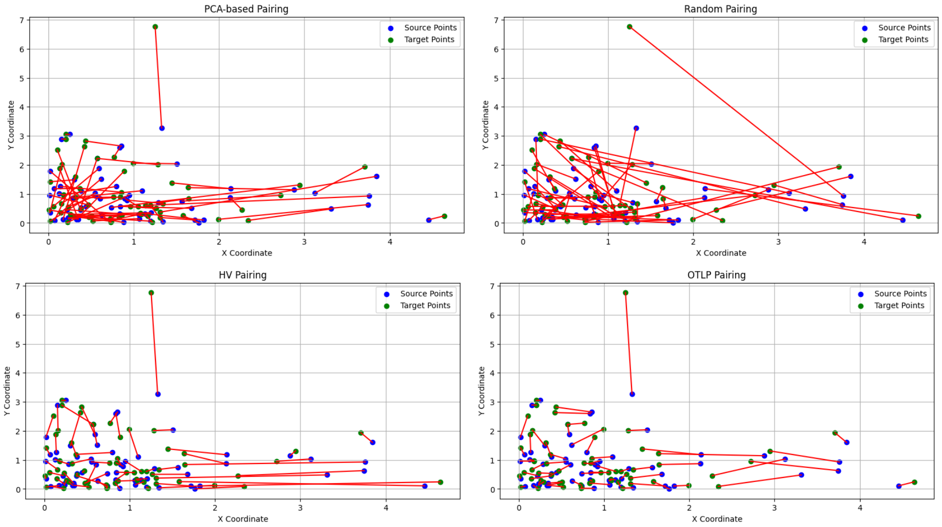

Figure 4: Pairings Generated by PCA-based, Random, HV, and OTLP Methods for Exponential Distribution

Discussion of Exponential Distribution Results The OTLP method consistently outperforms all heuristic approaches, establishing a clear benchmark for optimal pairing costs with a total cost of 0.3624. The HV Pairing method, while simpler, performs significantly better than both the PCA-based and Random Split methods, achieving a total cost of 0.7936 and a relative error of 118.99%, as shown in Table 0.14.1. In particular, the Random Pairing method fares the worst, with a total cost of 3.1275 and a relative error of 762.97%, underscoring the challenges of stochastic splitting in highly variable distributions.

The PCA Pairing method performs better than the Random Split, with a total cost of 1.3785 and a relative error of 280.37%. However, it still falls short of the HV Split, which maintains a lower total cost and relative error. From the visualization plot, it can be seen that this performance gap is particularly pronounced in densely distributed regions of the dataset, where PCA generates much messier pairings. The PCA-based Split captures the direction of maximum variance across the entire dataset, which may not effectively reflect the local variability within densely populated clusters. In areas where data points are densely packed, the local variance is relatively low, causing PCA to be less effective in distinguishing meaningful partition directions. Consequently, while PCA offers improved adaptability over the Random Split by aligning with the data's global variance structure, it struggles to accommodate the nuanced variability present in densely distributed regions, leading to higher pairing costs compared to the HV Split method.

0.14.2 Normal Distribution

In contrast, datasets generated from a normal distribution (mean 0, standard deviation 1) exhibit different pairing cost dynamics. The OTLP method remains the most cost-effective approach, with a total cost of 0.2477, setting the benchmark. The HV Pairing method shows improved performance compared to the exponential distribution, achieving a total cost of 0.4529, with a relative error of 82.85% and a cost ratio of 1.83. The Random Split method incurs a total cost of 2.7777, with a relative error of 1021.50% and a cost ratio of 11.22. Similarly, the PCA-based Split method results in a total cost of 2.5446, a relative error of 927.40%, and a cost ratio of 10.27, indicating a significant increase in cost relative to the OTLP benchmark.

Method	**Total Cost**	**Rel. Error (%)**	**Cost Ratio**
OTLP Pairing	0.2477	0.00	1.00
HV Pairing	**0.4529**	82.85	**1.83**
Random Pairing	**2.7777**	1021.50	**11.22**
PCA Pairing	**2.5446**	927.40	**10.27**

Table 2: Pairing Costs and Performance Metrics for Normal Distribution

Visualization of Pairings To further illustrate the differences in pairing strategies, the pairings generated by each method for the normal distribution are visualized in Figure 5, similar to that of exponential distribution.

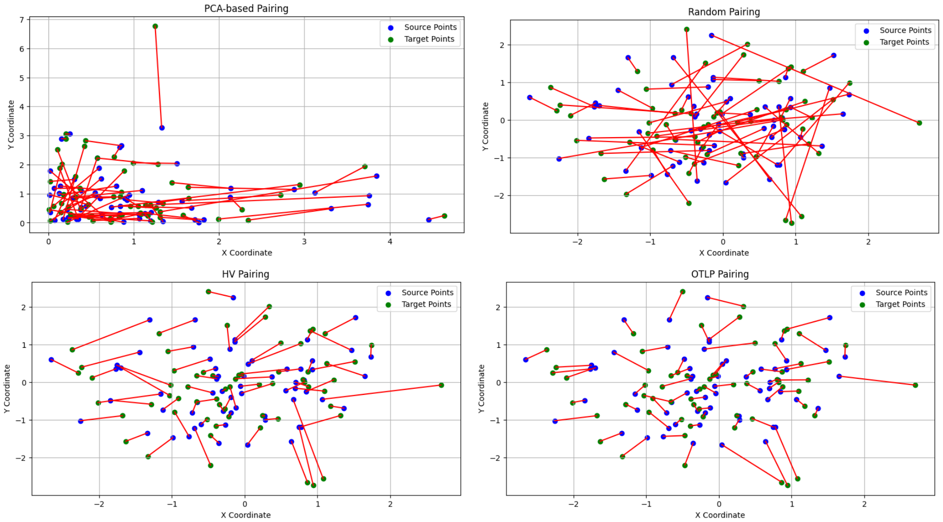

Figure 5: Pairings Generated by PCA-based, Random, HV, and OTLP Methods for Normal Distribution

Discussion of Normal Distribution Results The OTLP method consistently outperforms all heuristic approaches, establishing a clear benchmark for optimal pairing costs with a total cost of 0.2477. The HV Pairing method, while simpler, incurs a significantly higher total cost of 0.4529, but still performs better than the Random Split and PCA-based Split methods.

Interestingly, the PCA Pairing method shows minimal improvement over the Random Split, achieving a total cost of 2.5446 and a relative error of 927.40%. The Random Split method records a total cost of 2.7777 with a relative error of 1021.50%. In repeated experiments, the Random Split sometimes performs comparably to or even better than the PCA-based Split. Several factors may contribute to this outcome:

- **Global vs. Local Variance Capture:** PCA-based Splits rely on capturing the direction of maximum global variance in the dataset. In a normal distribution, especially one that is isotropic (having similar variance in all directions), the principal component may not provide a significant advantage, as the variance is uniformly distributed. Consequently, the splitting direction chosen by PCA may not align optimally with the underlying data structure for pairing purposes.

- **Overfitting to Global Structure:** PCA aims to maximize variance along a single direction, which may lead to overemphasis on certain data characteristics while neglecting others even if re-centering is applied. In densely populated regions with low local variance, PCA may fail to capture subtle but important variations, resulting in less effective splits. The Random Split, by exploring diverse directions, can sometimes better accommodate these local variations, enhancing pairing performance in specific instances.

- **Flexibility and Diversity in Splitting Directions:** The inherent randomness in the Random Split method allows it to explore a broader range of splitting directions across different recursive steps. This flexibility can lead to more adaptable partitions that better handle the uniform spread of normally distributed data. In contrast, the PCA-based Split is constrained to principal directions, limiting its ability to adapt to varying local data distributions.

These factors collectively suggest that while the PCA-based Split generally offers improved adaptability over the Random Split by leveraging global variance structure, the Random Split's inherent flexibility can occasionally result in more effective pairings, especially in datasets where local variance plays a crucial role. This underscores the complex interplay between global and local data characteristics in determining the efficacy of partitioning methods. Further refinement of the PCA-based approach, potentially incorporating mechanisms to account for local variance, could bridge the performance gap observed in these experiments.

Conclusion and Future Work

0.15 Conclusion

This study evaluated the efficiency of PCA-based and Random Split partitioning methods in minimizing transportation costs compared to previously developed horizontal and vertical (HV) cuts and the Optimal Transport Linear Programming (OTLP) approach. The results consistently demonstrated that both PCA-based and Random Split methods incur significantly higher costs than the OTLP benchmark, with the Random Split method generally showing the least efficiency. Specifically, while the HV Pairing method, though less adaptable, outperformed both PCA and Random Splits in most scenarios, PCA-based methods occasionally approached the performance of Random Splits in certain experiments.

The comparative analysis revealed several key insights:

- **Optimal Transport Linear Programming (OTLP):** As expected, OTLP consistently achieved the lowest pairing costs, establishing a clear benchmark for optimal performance in transportation models.
- **Horizontal and Vertical (HV) Splits:** The HV Pairing method, despite its

simplicity and rigidity, performed better than both PCA and Random Splits. This indicates that fixed partitioning directions, while less adaptable, can still offer reasonable efficiency in certain data distributions.

- **PCA-based Split:** The PCA-based Split method generally outperformed the Random Split by leveraging global variance structures within the data. However, its performance was hindered in densely distributed regions where local variance is low, limiting its effectiveness in capturing nuanced data structures.

- **Random Split:** Although Random Split exhibited the highest relative errors and cost ratios overall, it occasionally outperformed PCA-based Splits in repeated experiments. This suggests that the inherent flexibility and diversity in splitting directions of the Random Split method can sometimes better accommodate local data variations that PCA-based methods may overlook.

These findings highlight the complex interplay between global and local data characteristics in determining the efficacy of partitioning methods. While PCA-based Splits offer improved adaptability over Random Splits by aligning with the data's global variance structure, the Random Split's stochastic nature provides a level of flexibility that can occasionally lead to more effective pairings, especially in datasets with significant local variability. Nonetheless, both heuristic methods fall short of the OTLP benchmark, underscoring the challenges inherent in developing partitioning strategies that can consistently approach optimal transport costs across diverse data distributions.

0.16 Implications for Future Research

The findings from this study suggest several avenues for future research to enhance the efficiency of partitioning methods in transportation models:

- **Different Evaluation Metrics:** Future experiments could incorporate a broader range of evaluation metrics beyond transportation cost alone, such as computational complexity, scalability, or adaptability to different data distributions.
- **Enhanced Split Considerations:** Incorporating additional considerations into the split decision, such as direct measurements of Euclidean distance or other relevant features from optimal transport theory, could potentially yield more cost-effective solutions.
- **Hybrid Methods:** As PCA-based splitting loses significance during iterative partitioning due to diminishing data variance, combining PCA's structured approach with randomized methods could enhance performance. For instance, after several iterations of PCA, transitions to hybrid splitting strategies, such as horizontal/vertical (HV) cuts or randomized cuts, may maintain efficiency while preserving flexibility. This hybrid approach balances initial structure with adaptability as the data becomes less variant.

In conclusion, while the tested methods did not outperform traditional or optimal approaches under the conditions set forth in this study, they offer a foundation upon which more refined and situationally adaptive methods can be developed.

The insights gained underscore the complexity of partitioning tasks and the necessity of aligning methodological choices with specific operational objectives and data characteristics.

Bibliography

[1] Tomer J. Czaczkes, Christoph Grüter, Sam M. Jones, and Francis L. W. Ratnieks. Uncovering the complexity of ant foraging trails. *Communications in Integrative Biology*, 5(1):78–80, 2012.

[2] Lawrence C. Evans and Ronald F. Gariepy. *Measure Theory and Fine Properties of Functions, Revised Edition*. Chapman and Hall/CRC, 1st edition, 2015.

[3] Anna R. Karlin and Y. Peres. *Game Theory, Alive*. American Mathematical Society, Providence, 2016. 1 online resource (400 pages).

[4] N. Karmarkar. A new polynomial-time algorithm for linear programming. *Combinatorica*, 4(4):373–395, 1984.

[5] L.G. Khachiyan. Polynomial algorithms in linear programming. *USSR Computational Mathematics and Mathematical Physics*, 20(1):53–72, 1980.

[6] Soheil Kolouri, Se Rim Park, Matthew Thorpe, Dejan Slepcev, and Gustavo K. Rohde. Optimal mass transport: Signal processing and machine-

learning applications. *IEEE Signal Processing Magazine*, 34(4):43–59, 2017.

[7] Gaspard Monge. *Mémoire sur la théorie des déblais et des remblais*. [De l'Imprimerie Royale], [Paris], 1781.

[8] Elisa Negrini and Levon Nurbekyan. Applications of no-collision transportation maps in manifold learning. *SIAM Journal on Mathematics of Data Science*, 6(1):97–126, 2024.

[9] Levon Nurbekyan, Alexander Iannantuono, and Adam M. Oberman. No-collision transportation maps. *Journal of Scientific Computing*, 82(2):45, Feb 2020.

[10] Gabriel Peyré and Marco Cuturi. Computational optimal transport, 2020.

[11] Walter Rudin. *Real and complex analysis, 3rd ed.* McGraw-Hill, Inc., USA, 1987.

[12] Cédric Villani. *Topics in optimal transportation*, volume 58 of *Graduate Studies in Mathematics*. American Mathematical Society, Providence, RI, 2003.