VIET NAM NATIONAL UNIVERSITY HO CHI MINH CITY
HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY
FACULTY OF COMPUTER SCIENCE & ENGINEERING

GRADUATION THESIS

A STUDY OF SHAPE INTERPOLATION USING OPTIMAL TRANSPORT

Major: Computer Science

Commitee:	**Commitee 3 Computer Science**	
Supervisors:	**Dr. Nguyen An Khuong**	
	Assoc. Prof. Le Xuan Truong, UEH	
	Mr. Le Thanh Son, KTH	
	Mr. Tran Dinh Vinh Thuy, EC Lyon	
	—o0o—	
Reviewer:	**Assoc. Prof. Le Hong Trang**	
Student 1:	**Truong Hoang Nguyen Vu**	**2112673**
Student 2:	**Tran Nguyen Thai Binh**	**2110051**

Ho Chi Minh City, May 2025

ĐẠI HỌC QUỐC GIA TP.HCM
TRƯỜNG ĐẠI HỌC BÁCH KHOA

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do - Hạnh phúc

KHOA: KH & KT Máy tính
BỘ MÔN: KHMT

NHIỆM VỤ LUẬN VĂN/ ĐỒ ÁN TỐT NGHIỆP

Chú ý: Sinh viên phải dán tờ này vào trang nhất của bản thuyết trình

HỌ VÀ TÊN: Trương Hoàng Nguyên Vũ — MSSV: 2112673
Trần Nguyễn Thái Bình — MSSV: 2110051

NGÀNH: Khoa học Máy tính — LỚP: MT21KH1, MT21KH3

1. Đầu đề luận văn/ đồ án tốt nghiệp: "MỘT NGHIÊN CỨU VỀ BÀI TOÁN NỘI SUY HÌNH DẠNG SỬ DỤNG LÝ THUYẾT VẬN CHUYỂN TỐI ƯU" (AN STUDY OF SHAPE INTERPOLATION USING OPTIMAL TRANSPORT).

2. Nhiệm vụ (yêu cầu về nội dung và số liệu ban đầu):
-Study in-depth mathematical backgrounds in measure theory, probability theory, optimization theory, and optimal transport theory.
- Conduct a comprehensive survey on existing shape interpolation methods in 2D and 3D.
- Conduct a comprehensive survey on shape interpolation methods using optimal transport.
- Collect dataset of 2D and 3D shapes.
- Reproduce results of shape interpolation using known optimal transport tools for 2D and 3D shapes, as well as propose suitable approaches.
- Conduct experiments, evaluate, and discuss the obtained results.
- Write the thesis report.

3. Ngày giao nhiệm vụ: 06/01/2025
4. Ngày hoàn thành nhiệm vụ: 05/05/2025

5. Họ tên giảng viên hướng dẫn:	**Phần hướng dẫn:**
1) TS. Nguyễn An Khương	Hướng dẫn chung và giám sát thực hiện
2) PGS. TS. Lê Xuân Trường	Hướng dẫn kiến thức toán nền tảng
3) ThS. Trần Đình Vĩnh Thụy	Hướng dẫn kỹ thuật
4) ThS. Lê Thành Sơn	Hướng dẫn kỹ thuật

Nội dung và yêu cầu LVTN/ ĐATN đã được thông qua Bộ môn.

Ngày ..... tháng ..... năm .....

CHỦ NHIỆM BỘ MÔN
(Ký và ghi rõ họ tên)

Lê Hà Thụy

GIẢNG VIÊN HƯỚNG DẪN CHÍNH
(Ký và ghi rõ họ tên)

Nguyễn An Khương

PHẦN DÀNH CHO KHOA, BỘ MÔN:
Người duyệt (chấm sơ bộ):
Đơn vị:
Ngày bảo vệ:
Điểm tổng kết:
Nơi lưu trữ LVTN/ĐATN:

TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KH & KT MÁY TÍNH

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do - Hạnh phúc

Ngày 19 tháng 05 năm 2025

PHIẾU ĐÁNH GIÁ LUẬN VĂN/ ĐỒ ÁN TỐT NGHIỆP

(Dành cho người hướng dẫn/phản biện)

1. Họ và tên SV/MSSV: Trương Hoàng Nguyên Vũ, Trần Nguyễn Thái Bình
   MSSV: 2112673, 2110051 Ngành (chuyên ngành): Khoa học máy tính

2. Đề tài: "A Study of Shape Interpolation Using Optimal Transport"

3. Họ tên người hướng dẫn/phản biện: Nguyễn An Khương

4. Tổng quát về bản thuyết minh:
   Số trang: Số chương:
   Số bảng số liệu Số hình vẽ:
   Số tài liệu tham khảo: Phần mềm tính toán:
   Hiện vật (sản phẩm)

5. Những ưu điểm chính của LV/ ĐATN:

- Đồ án tốt nghiệp "Một Nghiên Cứu về Bài Toán Nội Suy Hình Dạng Sử Dụng Lý Thuyết Vận Chuyển Tối Ưu" (A Study of Shape Interpolation Using Optimal Transport) của hai sinh viên Trương Hoàng Nguyên Vũ và Trần Nguyễn Thái Bình trình bày một nghiên cứu sâu rộng và công phu về một chủ đề phức tạp và có tính thời sự cao trong lĩnh vực đồ họa máy tính và thị giác máy tính. Nhìn chung, đồ án thể hiện sự đầu tư nghiêm túc về mặt lý thuyết cũng như nỗ lực đáng kể trong việc triển khai thực nghiệm.

- Một trong những điểm mạnh nổi bật nhất của đồ án là nền tảng lý thuyết vững chắc. Sinh viên đã dành nhiều công sức để tìm hiểu và trình bày một cách chi tiết các kiến thức toán học nền tảng bao gồm lý thuyết độ đo, lý thuyết xác suất, giải tích lồi, và lý thuyết tối ưu, trước khi đi sâu vào các khía cạnh của lý thuyết vận chuyển tối ưu như bài toán Monge, công thức Kantorovich, khoảng cách Wasserstein, vận chuyển tối ưu không cân bằng và động học. Việc này không chỉ cho thấy sự hiểu biết sâu sắc của sinh viên mà còn cung cấp một nền tảng kiến thức giá trị cho người đọc.

- Về mặt phương pháp luận, đồ án không chỉ dừng lại ở việc khảo sát các phương pháp nội suy hình dạng hiện có cho dữ liệu 2D và 3D mà còn mạnh dạn triển khai và đề xuất các phương pháp mới dựa trên vận chuyển tối ưu. Cụ thể, sinh viên đã cài đặt thuật toán Sinkhorn, mô hình Dynamic Optimal Transport Variational Autoencoder (DOT VAE) cho dữ liệu 2D, và quan trọng hơn là đề xuất các biến thể DOT-MC VAE và DOT VAE cho dữ liệu 3D dạng voxel. Điều này thể hiện khả năng vận dụng kiến thức và sự sáng tạo trong nghiên cứu.

- Phần thực nghiệm và đánh giá kết quả cũng được thực hiện một cách toàn diện. Sinh viên đã so sánh các phương pháp của mình với các kỹ thuật truyền thống như nội suy tuyến tính và VAE cơ bản trên các tập dữ liệu 2D và 3D. Việc sử dụng các độ đo phù hợp (MSE cho 2D, Chamfer/Hausdorff cho 3D) và cách tiếp cận định lượng các thuộc tính "độ méo tối thiểu" và "chuyển tiếp mượt mà" thông qua phân tích thống kê trung bình và phương sai của các độ đo này cho thấy một phương pháp nghiên cứu bài bản. Phân tích độ phức tạp thời gian của các giải thuật cũng là một điểm cộng đáng giá. Báo cáo được cấu trúc logic, dễ theo dõi từ lý thuyết đến ứng dụng.

6. Những thiếu sót chính của LV/ĐATN:

- Tuy nhiên, đồ án vẫn còn một số hạn chế. Như trong ĐA đã đề cập, kích thước tập dữ liệu huấn luyện cho các mô hình VAE còn nhỏ, điều này có thể ảnh hưởng đến khả năng tổng quát hóa của mô hình.

- Việc sử dụng MSE để đánh giá nội suy hình dạng 2D, mặc dù phổ biến, nhưng có những hạn chế trong việc nắm bắt sự khác biệt về hình dạng mang tính tri giác, một điểm mà sinh viên cũng đã nhận ra.

- Một số kết quả nội suy, đặc biệt với các chuyển đổi 3D phức tạp, có thể xuất hiện các tạo tác (artifacts) hoặc chưa hoàn toàn đảm bảo tính thực tế vật lý (ví dụ: tự giao cắt), đây là một thách thức chung trong lĩnh vực này nhưng cũng chỉ ra hướng cải thiện trong tương lai.

- Tóm lại, đây là một đồ án tốt nghiệp có chất lượng rất cao, thể hiện sự nỗ lực nghiên cứu đáng kể. Sinh viên đã giải quyết một vấn đề thách thức với sự chặt chẽ về lý thuyết và thực nghiệm. Các hạn chế được xác định là điển hình cho nghiên cứu ở cấp độ này và mở ra những hướng đi rõ ràng cho công việc trong tương lai. Công trình đã hoàn thành tốt các mục tiêu đề ra và cho thấy sự nắm vững các khái niệm nâng cao trong khoa học máy tính.

7. Đề nghị: Được bảo vệ ☑ Bổ sung thêm để bảo vệ ☐ Không được bảo vệ ☐

8. Các câu hỏi SV phải trả lời trước Hội đồng:

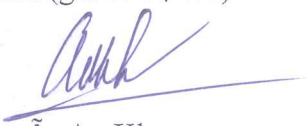
i. Hãy giải thích sự khác biệt chính giữa công thức Monge và Kantorovich của bài toán vận chuyển tối ưu. Tại sao việc nới lỏng theo Kantorovich thường được ưa chuộng trong thực tế, đặc biệt trong bối cảnh nội suy hình dạng của ĐA này?

ii. ĐA đã sử dụng thuật toán Sinkhorn vì lợi ích tính toán của nó. Hãy giải thích vai trò của số hạng chính quy hóa entropy $(1/\lambda)h(P)$ trong việc tính toán khoảng cách Sinkhorn và tham số λ ảnh hưởng đến giải pháp và tốc độ tính toán như thế nào?

iii. ĐA sử dụng các metrics đánh giá "độ méo tối thiểu" và "chuyển tiếp mượt mà" bằng cách sử dụng giá trị trung bình và phương sai của các khoảng cách theo cặp giữa các hình dạng trung gian. Hãy giải thích rõ hơn tại sao các thước đo thống kê này lại là đại diện phù hợp cho các thuộc tính định tính này? Có trường hợp nào chúng có thể gây hiểu lầm không?

9. Đánh giá chung (bằng chữ: Xuất sắc, Giỏi, Khá, TB): **Xuất sắc** Điểm: **10**/10

Ký tên (ghi rõ họ tên)



Nguyễn An Khương

TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KH & KT MÁY TÍNH

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do - Hạnh phúc

Ngày 16 tháng 5 năm 2025

PHIẾU ĐÁNH GIÁ LUẬN VĂN/ ĐỒ ÁN TỐT NGHIỆP

(Dành cho người phản biện)

1. Họ và tên SV: *Trần Nguyễn Thái Bình và Trương Hoàng Nguyên Vũ*

MSSV: 2110051 và 2112673 Ngành (chuyên ngành): Khoa học máy tính

2. Đề tài: **Một cách tiếp cận bài toán tái cấu trúc 3D bằng phương pháp vận chuyển tối ưu**

3. Họ tên người phản biện: Lê Hồng Trang

4. Tổng quát về bản thuyết minh:

Số trang: 111 Số chương: 6

Số bảng số liệu: 9 Số hình vẽ: 15

Số tài liệu tham khảo: 61 Phần mềm tính toán: 0

Hiện vật (sản phẩm): 0

5. Những ưu điểm chính của LV/ ĐATN:

- Đồ án nghiên cứu về việc sử dụng mô hình vận chuyển tối ưu (optimal transport) cho bài toán nội suy hình ảnh 3D.
- Nhóm sinh viên đã tìm hiểu kỹ về nền tảng, tính chất và một số kỹ thuật cho mô hình vận chuyển tối ưu. Đây là kiến thức nâng cao trong tối ưu toán học, nhóm sinh viên không chỉ phải tự học hoàn toàn về mô hình này, mà còn nhiều kiến thức liên quan. Qua đó cho thấy khối lượng việc học và nghiên cứu là lớn.
- Nhóm sinh viên đề xuất được một phương pháp dùng mô hình tối ưu vận chuyển kết với một kiến trúc VAE (Variational AutoEncoder) nhằm mở rộng bài toán nội suy hình ảnh trong 2D sang 3D.
- Phương pháp đề xuất được hiện thực, thực nghiệm và đánh giá.

6. Những thiếu sót chính của LV/ĐATN:

- Các nội dung tìm hiểu và nghiên cứu liên quan đến Vận chuyển tối ưu nên tách thành một chương như là nội dung chính của của đồ án.
- Trình bày về phương pháp đề xuất cần chi tiết hơn, để người đọc có thể dễ đọc và hiểu về nội dung cũng như khối lượng công việc này.

7. Đề nghị: Được bảo vệ ☑ Bổ sung thêm để bảo vệ □ Không được bảo vệ □

8. Các câu hỏi SV phải trả lời trước Hội đồng:

9. Đánh giá chung (bằng chữ: Xuất sắc, Giỏi, Khá, TB): Xuất sắc Điểm: 9.8/10

Ký tên (ghi rõ họ tên)

Lê Hồng Trang

Declaration

We certify that this capstone project "A Study of Shape Interpolation Using Optimal Transport" is our research under the supervision of Dr. Nguyen An Khuong, Assoc. Prof. Le Xuan Truong, Mr. Le Thanh Son and Mr. Tran Dinh Vinh Thuy and derived from practical needs and our desire to study. We also declare that everything written in this report, except for which from the references in bibliography section, is the result of our research and has never been published before.

Authors

Acknowledgements

This capstone project, an essential milestone of our studying path, would never be launched and reached this point without the support of many individuals. We sincerely appreciate Dr. Nguyen An Khuong, Assoc. Prof. Le Xuan Truong, Mr. Le Thanh Son and Mr. Tran Dinh Vinh Thuy, our supervisors, for the whole progress. Under their guidance, we have learned the background knowledge and be able to complete this project. Also, their invaluable assists and corrections, indicated by weekly representations and even the brothers have spent hours each week to listen and give advices to us.

In addition, we also give our utmost thanks to the lecturers at Ho Chi Minh city University of Technology, especially the Faculty of Computer Science and Engineering lecturers, for their gradual knowledge given to us over four years. Last but not least, we would like to thank our families and friends for wholeheartedly supporting and encouraging us during our learning path.

Abstract

Shape interpolation is one of the most fundamental problems in computer graphics and computer vision. It has numerous applications in real life. Several methods have been proposed to overcome this problem. However, tackling this problem require us to deal with lots of complicated challenges, e.g. feature alignment, minimal distortion or physical constraints. Optimal transport has emerged as a particularly powerful framework for this task, offering both theoretical elegance and practical advantages. In this report, we study in-depth the mathematical foundations of optimal transport and related concepts. We also show the usability of optimal transport in shape interpolation problem in term of assuring smooth transition and minimal distortion as well as its promising results compared to the classical linear interpolation method.

Keywords: Shape interpolation, optimal transport, mathematics, smooth transition, minimal distortion.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

Shape interpolation is one of the most fundamental problems in computer graphics and computer vision. It is relevant to the process of generating transitions between two or more shapes or creating intermediate forms that gradually transform one shape into another. The term shape here can be interpreted as images (2D data) and point clouds or meshes (3D data). In shape interpolation, the interpolated shapes have to satisfy key properties such as feature alignment, minimal distortion and sometimes physical constraints. In this project, we only focus on two main properties which are smooth transition and minimal distortion.

This problem has numerous applications in real life. For example, in term of 2D data or images, it can be used for data augmentation in machine learning problems [24], rigidity-preserving morphing in animation and graphics [59], image reconstruction [50], medical image analysis [54] and geometric modeling [47]. In case of 3D data such as point clouds of meshes, shape interpolation tasks contribute to animation and motion capture [60], medical imaging and reconstruction problems [17], game industry along with virtual reality [44] and object recognition as well as classification tasks [33].

Recently, several methods have been proposed to tackle this problem [4, 7, 11, 17]. The most simple and common method is linear interpolation which uses a linear function to form the interpolated shapes. While various approaches to shape interpolation have been proposed over the years, optimal transport has emerged as a particularly powerful framework for this task, offering both theoretical elegance and practical advantages.

Optimal transport theory, originally formulated by Gaspard Monge in the 18^{th} century and later refined by Leonid Kantorovich, has provided a natural way to measure distances between probability distributions and, by extension, between shapes. The theory establishes a mathematically rigorous framework for finding the most efficient way to transform one distribution into another while minimizing a specified cost function. When applied to shape interpolation, optimal transport can generate visually meaningful intermediate shapes that guarantee key properties such as minimal distortion and smooth transition [1, 5, 13].

In this work, we investigate various formulations of optimal transport for shape interpolation problem focusing on 2D and 3D shapes, analyze their properties and limitations, and conduct

experiments for some approaches to address existing challenges. Our study encompasses both theoretical developments and practical implementations, with particular attention to the compatibility of optimal transport in shape interpolation and the quality of interpolated results in term of smooth transition and minimal distortion.

1.2 Shape Interpolation and Shape Reconstruction Problem

Shape interpolation and shape reconstruction are closely related problems in the field of computer graphics, geometry processing, and machine learning, as both involve understanding and modeling the structure of shapes in a coherent way. Shape reconstruction focuses on recovering a complete shape from partial, noisy, or sparse observations, such as point clouds or 2D images. This process typically requires a strong prior about the target shape distribution and often employs methods like deep learning, implicit representations, or optimization-based techniques. Shape interpolation, on the other hand, involves generating intermediate shapes that smoothly transition between two or more known shapes. This task also relies on a learned or defined shape space that captures the underlying structure of the shapes involved.

Both problems aim to understand and model the manifold of plausible shapes, and interpolation often acts as a diagnostic tool to evaluate the quality of the reconstruction space. In fact, successful interpolation implies that the learned shape space is coherent and meaningful, which is essential for high-quality reconstruction. Moreover, many reconstruction pipelines rely on latent representations where interpolation can naturally occur, further emphasizing their interdependence. Thus, shape interpolation and reconstruction are inherently linked through their shared reliance on compact, structured shape representations.

1.3 Problem Formulation

Shape interpolation is a computational problem aiming at constructing a smooth transformation whose minimal distortion between two or more shapes. In this part, we will give the formulation for this problem for the case of two shapes, cases for multiple shapes follow as derivations.

In case of shape interpolation between two shapes, the two source (from) $\mathcal{S}_0$ and target (to) $\mathcal{S}_1$ shapes are given. These shapes can be represented using various mathematical models, each with unique advantages and challenges, particularly in interpolation tasks. Some of representing mathematical models can be listed out such as implicit representation, explicit representation, parametric representation and so on. In our study, we only consider discrete representations such as images (pixels) for 2D shapes and point clouds (points) for 3D shapes. The interpolation task introduces a blending parameter $t \in [0, 1]$, where $t = 0$ corresponds to $\mathcal{S}_0$ and $t = 1$ corresponds to $\mathcal{S}_1$.

Let f be a function to generate intermediate shapes between $\mathcal{S}_0$ and $\mathcal{S}_1$ whose three inputs which are t, $\mathcal{S}_0$ and $\mathcal{S}_1$. Using this, we have that

$$f(0, \mathcal{S}_0, \mathcal{S}_1) = \mathcal{S}_0$$

and

$$f(1, \mathcal{S}_0, \mathcal{S}_1) = \mathcal{S}_1.$$

Each value of t yields interpolated shape

$$\mathcal{S}_t = f(t, \mathcal{S}_0, \mathcal{S}_1)$$

representing the transition from $\mathcal{S}_0$ to $\mathcal{S}_1$.

Therefore, the shape interpolation problem starts with three inputs which are the source shape $\mathcal{S}_0$, the target shape $\mathcal{S}_1$ and the blending parameter $t \in [0, 1]$. The goal is to find the intermediate shapes between $\mathcal{S}_0$ and $\mathcal{S}_1$ corresponding to the blending parameter t to form a smooth transition from $\mathcal{S}_0$ to $\mathcal{S}_1$.

1.4 Motivation for Using Optimal Transport in Shape Interpolation

The significance of optimal transport-based shape interpolation lies in its ability to produce smooth transitions while respecting mass preservation constraints. Unlike simpler methods such as linear interpolation or naive geometric morphing, optimal transport-based method takes into account the global structure of the shapes and generates intermediate forms that follow the principle of least cost. This approach often results in more visually pleasing and smooth transformations, particularly when dealing with complex shapes or significant geometric differences.

Recent advances in computational methods and increased processing power have made optimal transport-based shape interpolation more practical for real-world applications. However, several challenges remain, including computational efficiency and adaptation to specific domain constraints. This study aims to explore these aspects in detail, examining both the theoretical foundations and practical implementations of shape interpolation using optimal transport.

1.5 Study Objectives

This capstone project aims to study the fundamental concepts and applications of optimal transport in shape interpolation. Against theoretical and practical challenges, our work follows these objectives:

1. Studying in-depth the principles of mathematical analysis, including measure theory, probability theory, optimization theory and optimal transport theory that are necessary for understanding and applying them to shape interpolation problem.
2. Studying predominant approaches for shape interpolation focusing on optimal transport based methods including Wasserstein distance, Sinkhorn algorithm and dynamic optimal transport. We also study their applications in both cases of 2D and 3D shapes.
3. Implementing and testing shape interpolation methods on various datasets, with experiments conducted on both images and point clouds, to evaluate and compare among different optimal transport based approaches as well as with linear interpolation approach.
4. Developing and enhancing interpolation techniques by combining optimal transport theory with modern computational methods to achieve smoother shape transitions.

These objectives reflect our goal to advance the understanding and application of optimal transport in shape interpolation problem, bridging theoretical foundations with practical implementations.

1.6 Report Structure

Based on the objectives and the scope of the problem, the main structure of our project is as following.

Chapter I. Introduction. We introduce an overview of shape interpolation problem as well as some approaches which are used to solve it. We also introduce briefly about optimal transport theory, our project problem formula and the objectives that we hope to achieve in the project.

Chapter II. Preliminaries. We discuss the essential mathematical and optimal transport preliminaries for applying optimal transport to shape interpolation problem. For mathematical preliminaries, they consist of measure theory, probability theory and optimization theory. In case of optimal transport preliminaries, they cover Monge problem, Kantorovich formulation, Wasserstein distance, unbalanced optimal transport and dynamic optimal transport.

Chapter III. Related Works. We discuss some approaches which are used in shape interpolation problem for both 2D and 3D shapes as well as applications of optimal transport for this task.

Chapter IV. Methods. In this chapter, we introduce and propose methods which are used in our experiments on both 2D and 3D shapes. Along with them, we also introduce and construct mathematical backgrounds for those methods.

Chapter V. Experiments. We implement and conduct shape interpolation methods on various datasets, with experiments conducted on both 2D and 3D shapes. We also give some discussions about the obtained results.

Chapter VI. Conclusion. This is a short chapter where we give a summary on our study and practice in this capstone project. We point out advantages, limitations, knowledge which remains unclear in this project and give a plan for future works.

Chapter 2

Preliminaries

2.1 Computer Graphics

Computer graphics involves generating visual representations of data using computers. It covers rendering, modeling, animation, and simulation of visual images and structures in two and three dimensions. Fundamental aspects include rasterization, ray tracing, shading techniques, and geometric transformations essential for image synthesis. Applications range widely from video games, films, and simulations to virtual and augmented reality environments, each requiring intricate algorithms and optimization techniques to produce realistic visualizations.

2.2 Computer Vision

Computer vision is the field that enables computers to interpret and understand visual information from the world. It encompasses image processing, feature extraction, object detection, image segmentation, and machine learning methods to analyze visual data. Techniques include convolutional neural networks, edge detection algorithms, and feature descriptors like SIFT, SURF, and ORB, which are crucial for interpreting scenes and extracting meaningful data. Advanced applications include facial recognition, autonomous vehicles, medical imaging, and surveillance, necessitating robust algorithms to manage complex real-world data.

2.3 Shape Interpolation Problem

Shape interpolation, often referred to as morphing, is the process of transitioning smoothly from one shape or geometric structure to another. Mathematically, it involves finding intermediate shapes that represent a continuous and coherent path between two given shapes. Formally, consider two shapes represented as point or vertex sets $S_1 = \{x_i\}_{i=1}^n$ and $S_2 = \{y_i\}_{i=1}^n$. The interpolation can be mathematically expressed using a parametric form as:

$$S_t = \{(1-t)x_i + ty_i \mid x_i \in S_1, y_i \in S_2\}, \quad t \in [0,1].$$

In practice, optimal shape interpolation involves solving a variational problem, aiming to minimize distortion or energy functional during morphing. This process ensures that intermediate shapes are not only visually plausible but also mathematically consistent. Such problems can be mathematically formulated as minimizing a functional defined by:

$$E(S_t) = \int_\Omega D(x, \phi_t(x))\, dx,$$

where D represents a metric or measure of the distortion or deformation between the initial shape and the intermediate shape $\phi_t(x)$, characterized by an appropriate deformation mapping. One robust mathematical tool for addressing this problem is optimal transport theory, which utilizes the Wasserstein distance. The Wasserstein distance quantifies the minimal cost required to transport a distribution of mass from one configuration to another, thus ensuring a natural and physically realistic morphing between shapes.

Specifically, optimal shape interpolation employing optimal transport theory involves solving the following minimization problem:

$$\min_{\phi \in \Phi} \int_\Omega c(x, \phi(x))\, dx,$$

where $c(x, y)$ denotes the transportation cost between points x and y, and Φ defines a class of admissible deformation mappings that satisfy certain boundary conditions, continuity constraints, and structural integrity requirements. Such approaches guarantee that interpolated shapes are smooth, intuitively consistent, and computationally tractable for practical applications.

Chapter 3

Optimal Transport

In this chapter, we establish a rigorous mathematical framework essential for understanding the foundational concepts of optimal transport theory applied throughout this thesis. We begin by reviewing fundamental topics in measure theory, which include measure spaces, Borel σ-algebras, outer measures, Radon measures and Lebesgue measures. Then, we develop the concept of measurable maps culminating in the Lebesgue integral. Subsequently, we delve into probability theory by examining probability density functions and the critical concept of Kullback-Leibler divergence, providing essential tools for quantifying statistical relationships and divergences between distributions.

Furthermore, we introduce crucial preliminaries in optimal transport theory. Firstly, We discuss Monge problem in both discrete and continuous contexts, analyze convexity characteristics and emphasize existence of solutions to this type of optimal transport problem. Next, we present the Kantorovich formulation, address the limitations inherent in Monge original formulation and outline its motivations. We also explore the discrete and continuous scenarios along with a detailed analysis of the convex properties of the Kantorovich problem. Moreover, we introduce several essential variants of optimal transport such as regularized and unbalanced optimal transport problems and discuss their key properties that lay the groundwork necessary for their applications in computational techniques. The chapter concludes by presenting the Wasserstein metric (distance) as a natural and profound consequence of optimal transport theory. Motivations for adopting Wasserstein distance are discussed and their formulation is precisely defined.

3.1 Mathematical Preliminaries

3.1.1 Measure Theory

Measure Space

Let Ω be any abstract set, analogous to a region in $\mathbb{R}^d$. In this part, we will define step-by-step what is a collection of measurable subsets of Ω and which conditions a measure must meet. Moreover, we also consider some examples to clarify our words.

Definition 3.1.1.1 (σ-algebra and measurable space) *Given a set Ω. A collection of subset $\mathcal{F}$ of Ω is called a* σ-algebra *if*

1. $\varnothing, \Omega \in \mathcal{F}$.
2. *If $A \in \mathcal{F}$, then $A^c \in \mathcal{F}$.*

3. If $A_1, A_2, \ldots \in \mathcal{F}$, then $A_1 \cup A_2 \cup \ldots \in \mathcal{F}$.

The pair $(\Omega, \mathcal{F})$ is called a measurable space*.*

In Definition 3.1.1.1, the initial condition ensures that both the empty set and the entire initial set are measurable. The second condition suggests that if a subset is measurable, its complement should also be measurable. The third condition reflects our method of dividing complex regions in $\mathbb{R}^d$. Prior to defining a measure, we explore several examples and related technical theorems.

Example 3.1.1.2. Let $\Omega = \{1, 2, 3\}$. Now, we prove that the collection $\mathcal{F} = \{\varnothing, \Omega, \{1\}, \{2, 3\}\}$ is a σ-algebra on Ω. To show that $\mathcal{F}$ is a σ-algebra, we must verify the three properties in definition 3.1.1.1. Since $\varnothing, \Omega \in \mathcal{F}$, the first property is satisfied. The pairs $(\varnothing, \Omega)$ and $(\{1\}, \{2, 3\})$ are such that each set is the complement of the other within $\mathcal{F}$. Therefore, for $A \in \mathcal{F}$, $A^c \in \mathcal{F}$. The second property is satisfied. Consider any finite or countable selection of $A_1, A_2, ...$ from $\mathcal{F}$, such as $\{1\}$, $\{2, 3\}$, $\varnothing$, and Ω. If there exists $A_i = \Omega$, then $A_1 \cup A_2 \cup \ldots = \Omega$, which is in $\mathcal{F}$. Otherwise, we consider the selection which does not contain Ω below

$$\begin{cases} \{1\} \cup \{2, 3\} & = \Omega \in \mathcal{F}, \\ \{1\} \cup \varnothing & = \{1\} \in \mathcal{F}, \\ \{2, 3\} \cup \varnothing & = \{2, 3\} \in \mathcal{F}, \\ \{1\} \cup \{2, 3\} \cup \varnothing & = \{1, 2, 3\} \in \mathcal{F}. \end{cases}$$

We see that any union finite selection of elements in $\mathcal{F}$ is in $\mathcal{F}$. Therefore, the third property is satisfied. Since all three properties are satisfied, $\mathcal{F}$ is a σ-algebra.

Proposition 3.1.1.3 *Let $\{\mathcal{F}_\alpha\}_{\alpha \in I}$ be an arbitrary family of σ-algebras on Ω. Then the intersection $\mathcal{F} = \bigcap_{\alpha \in I} \mathcal{F}_\alpha$ is a σ-algebra.*

Proof. We will check that $\mathcal{F}$ satisfies the three conditions for a σ-algebra. Since $\varnothing, \Omega \in \mathcal{F}_\alpha$, for every $\alpha \in I$, we have $\varnothing, \Omega \in \bigcap_{\alpha \in I} \mathcal{F}_\alpha$. If $A \in \bigcap_{\alpha \in I} \mathcal{F}_\alpha$, then $A \in \mathcal{F}_\alpha$ which leads to $A^c \in \mathcal{F}_\alpha$ for all $\alpha \in I$. Hence, $A^c \in \bigcap_{\alpha \in I} \mathcal{F}_\alpha$. Finally, if $A_i \in \mathcal{F}_\alpha$ for every $\forall i \in \mathbb{Z}^+$ and $\alpha \in I$, then $\bigcup_{i \in \mathbb{Z}^+} A_i \in \mathcal{F}_\alpha$ for every $\alpha \in I$. That means $\bigcup_{i \in \mathbb{Z}^+} A_i \in \bigcap_{\alpha \in I} \mathcal{F}_\alpha$. Therefore, we conclude that the intersection $\mathcal{F} = \bigcap_{\alpha \in I} \mathcal{F}_\alpha$ is a σ-algebra. □

Note that the union of σ-algebras on Ω is not necessarily a σ-algebra. Here is an example.

Example 3.1.1.4. Let $\mathcal{F}_1 = \{\varnothing, \Omega, \{1\}, \{2, 4\}\}$ and $\mathcal{F}_2 = \{\varnothing, \Omega, \{2\}, \{1, 4\}\}$ be two σ-algebras on $\Omega = \{1, 2, 4\}$. However, We have that

$$\mathcal{F} = \mathcal{F}_1 \cup \mathcal{F}_2 = \{\varnothing, \Omega, \{1\}, \{2\}, \{2, 4\}, \{1, 4\}\}$$

is not a σ-algebra since $\{4\} = \{2, 4\} \cap \{1, 4\} \notin \mathcal{F}$.

Given a set Ω, then $\{\varnothing, \Omega\}$ and the power set 2^Ω are two trivial σ-algebras. Any collection $\mathcal{C}$ of subsets of Ω, has a trivial σ-algebra 2^Ω that contains $\mathcal{C}$ i.e. $\mathcal{C} \subseteq 2^\Omega$. Therefore, there exists the smallest σ-algebra containing $\mathcal{C}$.

Theorem 3.1.1.5 *Given a set Ω and a collection of its subsets $\mathcal{C}$, then there exists the smallest σ-algebra containing $\mathcal{C}$, denoted by $\sigma(\mathcal{C})$, given by*

$$\sigma(\mathcal{C}) = \bigcap_{\alpha \in I} \mathcal{F}_\alpha$$

where $\{\mathcal{F}_\alpha\}_{\alpha \in I}$ is the set of σ-algebras containing $\mathcal{C}$. We also called $\sigma(\mathcal{C})$ to be the σ-algebra generated by $\mathcal{C}$.

Proof. Since $\sigma(\mathcal{C})$ itself is a σ-algebra containing $\mathcal{C}$ so $\sigma(\mathcal{C}) \in \{\mathcal{F}_\alpha\}_{\alpha\in I}$. That means we have

$$\bigcap_{\alpha\in I} \mathcal{F}_\alpha \subseteq \sigma(\mathcal{C}).$$

In addition, we know that $\bigcap_{\alpha\in I} \mathcal{F}_\alpha$ is also a σ-algebra containing $\mathcal{C}$ and $\sigma(\mathcal{C})$ is the smallest one. Hence, we have that

$$\sigma(\mathcal{C}) \subseteq \bigcap_{\alpha\in I} \mathcal{F}_\alpha.$$

Therefore, $\sigma(\mathcal{C}) = \bigcap_{\alpha\in I} \mathcal{F}_\alpha$. □

Here is an example of the σ-algebra generated by $\mathcal{C}$.

Example 3.1.1.6. Let $\Omega = \{1, 2, 3\}$. The collection $\mathcal{C} = \{\{1\}\}$ generates the σ-algebra $\mathcal{F} = \{\varnothing, \Omega, \{1\}, \{2, 3\}\}$.

Definition 3.1.1.7 (Measure and measure space) *Given a measurable space $(\Omega, \mathcal{F})$. A function $\mu : \mathcal{F} \to \mathbb{R}_+$ is called a* measure *if it satisfies*

1. $\mu(\varnothing) = 0$.
2. *Let $\{E_n\}_{n\in\mathbb{N}}$ be a disjoint class of pairwise disjoint subsets of Ω in $\mathcal{F}$. Then*

$$\mu\left(\bigcup_{n=1}^{\infty} E_n\right) = \sum_{i=1}^{\infty} \mu(E_n).$$

The triplet $(\Omega, \mathcal{F}, \mu)$ is called a measure space.

The criteria in Definition 3.1.1.7 align with our understanding of a measure. Specifically, the measure of the empty set is zero, any set has a non-negative measure, and the measure of a complex subset $A = \bigcup_{n=1}^{\infty} E_n$ of Ω can be determined by summing the measures of the distinct subsets E_n for every $n \in \mathbb{N}$. The depth of measure theory is thoroughly built upon this essential concept of a measure space. Here are some examples of measure spaces.

Example 3.1.1.8. Firstly, we consider the measurable space $(\mathbb{R}, \mathcal{B}(\mathbb{R}), \mu)$. The function μ of taking the length of open intervals on the real line for every $x, y \in \mathbb{R}$

$$\mu((x, y)) = |y - x|,$$

which an assumption that $\mu(\varnothing) = 0$, is a measure. Three properties of a measure are satisfied trivially.

Secondly, we consider the measurable space $(\mathbb{R}, \mathcal{F}, \mu)$ in which $\mathcal{F}$ is any σ-algebra on $\mathbb{R}$. Given $a \in \mathbb{R}$. The indicator function $\mathbf{1}_a : \mathcal{F} \to \{0, 1\}$ given by

$$\mathbf{1}_a(X) = \begin{cases} 1 & \text{if } a \in X, \\ 0 & \text{otherwise} \end{cases}$$

is a measure. We already have $\mathbf{1}_a(X) \geq 0$ for every $a \in X$. Now, we check the other two properties of a measure for this function. Since $a \notin \varnothing$, we have that $\mathbf{1}_a(\varnothing) = 0$. For disjoints $X_1, X_2, \ldots$ it cannot be the case that there are two of them contain a. If $a \notin X_k$ for all $k \in \mathbb{Z}^+$, then countable additivity holds since both sides are zero. Otherwise, if there exists $k \in \mathbb{Z}^+$ such that $a \in X_k$, then it follows that $a \notin X_\ell$ for every $\ell \neq k$ implying both sides are one.

Given measure spaces $(\Omega, \mathcal{F}, \mu)$ and $(\Gamma, \mathcal{G}, \nu)$, we also concern about which pair $(x, \gamma) \in \Omega \times \Gamma$ that can be measured and how to measure them.

Definition 3.1.1.9 (Product σ-algebra) *Let $(\Omega, \mathcal{F}, \mu)$ and $(\Gamma, \mathcal{G}, \nu)$ be measure spaces. The* product σ-algebra $\mathcal{F} \otimes \mathcal{G}$ *on $\Omega \times \Gamma$ is given by*

$$\mathcal{F} \otimes \mathcal{G} = \sigma(\{A \times B : A \in \mathcal{F}, B \in \mathcal{G}\}).$$

Definition 3.1.1.10 (Product measure) *Let $(\Omega, \mathcal{F}, \mu)$ and $(\Gamma, \mathcal{G}, \nu)$ be measure spaces. The* product measure $\lambda(A \times B)$ *on $\mathcal{F} \otimes \mathcal{G}$, where $A \in \mathcal{F}$ and $B \in \mathcal{G}$ is given by*

$$\lambda(A \times B) = \mu(A)\nu(B).$$

Borel σ-algebra

Definition 3.1.1.11 (Borel σ-algebra and Borel subset) *Let (M, d) be a metric space. We define the* Borel σ-algebra on M to be the σ-algebra generated by open subsets of M and denote by $\mathcal{B}(M)$. Each element in $\mathcal{B}(M)$ is called a Borel subset of M.

The Borel σ-algebra is a fundamental class of σ-algebra, especially associated to $\mathbb{R}^d$. Later on, we will define random variable using $\mathcal{B}(\mathbb{R}^d)$. We can think of $\mathcal{B}(\mathbb{R}^d)$ as containing all Lebesgue measurable and well-behave subsets of $\mathbb{R}^d$.

Proposition 3.1.1.12 *The Borel σ-algebra $\mathcal{B}(\mathbb{R})$ is generated by one of the following collections:*

1. *The collection of all closed subsets of $\mathbb{R}$.*
2. *The collection of all intervals of the form $(-\infty, b]$.*
3. *The collection of all intervals of $\mathbb{R}$ of the form $(a, b]$.*

Proof. Let $\mathcal{B}_1$, $\mathcal{B}_2$ and $\mathcal{B}_3$ be three σ-algebras generated by the collections of sets in collections 1, 2 and 3 of the proposition, respectively. We need to show that

$$\mathcal{B}(\mathbb{R}) \supset \mathcal{B}_1 \supset \mathcal{B}_2 \supset \mathcal{B}_3 \supset \mathcal{B}(\mathbb{R}).$$

Since $\mathcal{B}(\mathbb{R})$ includes the family of open subsets of $\mathbb{R}$ and is closed under complement, it includes the family of closed subsets of $\mathbb{R}$. Thus, it includes $\mathcal{B}_1$. We have that the sets of the form $(-\infty, b]$ are closed. Hence, they belong to $\mathcal{B}_1$. That means $\mathcal{B}_1 \supset \mathcal{B}_2$. Since $(a, b] = (-\infty, b] \cap (-\infty, a]^c$, each set of the form $(a, b]$ belongs to $\mathcal{B}_2$. Thus, $\mathcal{B}_2 \supset \mathcal{B}_3$. Finally, we know that each open interval of $\mathbb{R}$ is the union of a sequence of sets of the form $(a, b]$ and that each open subset of $\mathbb{R}$ is the union of a sequence of open intervals. In other words, each open subset of $\mathbb{R}$ belongs to $\mathcal{B}_3$. Therefore, we have that $\mathcal{B}_3 \supset \mathcal{B}(\mathbb{R})$. □

Proposition 3.1.1.13 *The Borel σ-algebra $\mathcal{B}(\mathbb{R}^d)$ is generated by one of the following collections:*

1. *The collection of all closed subsets of $\mathbb{R}^d$.*
2. *The collection of all closed half-space of the form $\{(x_1, \ldots, x_d) : x_1 \leq b_1, \ldots, x_d \leq b_d\}$.*
3. *The collection of all boxes of the form $\{(x_1, \ldots, x_d) : a_1 \leq x_1 \leq b_1, \ldots, a_d \leq x_d \leq b_d\}$.*

Outer Measure, Lebesgue Measure and Radon Measure

In this part, for a set X we denoted by 2^X the class of subsets of X.

Definition 3.1.1.14 (Outer measure) *Let X be a set. We say that $\mu : 2^X \to [0, \infty]$ is an* outer measure *if*

1. $\mu(\varnothing) = 0$,
2. *For every $E \subset X$ and every countable family $\{E_i\}_{i\in\mathbb{N}} \subset 2^X$ (not necessarily pairwise disjoint) with $E \subset \bigcup_{i=1}^{\infty} E_i$ it holds*

$$\mu(E) \leq \sum_{i=1}^{\infty} \mu(E_i).$$

The *Lebesgue measure* $\mathcal{L}^N$ in dimension N stands as the most fundamental example of an outer measure. It provides the mathematical foundation for the concept of *length* in one dimension, *area* in two dimensions, and *volume* in three dimensions. We construct this measure using the Hausdorff construction. The underlying principle is intuitive. For basic geometric shapes, we understand their natural measurements as a segment's length, a square's area or a cube's volume. For general sets, we determine their measure by finding the optimal covering using basic geometric shapes (segments, squares, cubes, etc.). This approach ultimately connects to the Lebesgue product measure in one dimension, in the following paragraph.

Definition 3.1.1.15 (Lebesgue outer measure) *Let $Q_r(x)$ defined by*

$$Q_r(x) = \{y \in \mathbb{R}^N : |x_i - y_i| < r/2 \text{ for all } i = 1, \ldots, N\}$$

be the open cube centered at x with radius r for every $x \in \mathbb{R}^N$ and $r > 0$. For every $E \subset \mathbb{R}^N$, we define the Lebesgue outer measure *of E by*

$$\mathcal{L}^N(E) = \inf\left\{\sum_{i=1}^{\infty} r_i^N \mid E \subset \bigcup_{i=1}^{\infty} Q_{r_i}(x_i), x_i \in \mathbb{R}^N, r_i \geq 0\right\}.$$

Unfortunately, the Lebesgue outer measure is not a measure. Let us show that the Lebesgue outer measure $\mathcal{L}^1$ is not a measure on $\mathbb{R}$. We consider the Vitali construction. Let $\sim$ be an equivalence relation on $[0, 1]$ defined by

$$x \sim y \quad \text{if and only if} \quad x - y \in \mathbb{Q}.$$

Let $V \subset [0, 1]$ be a set containing exactly one representative from each equivalence class (possible by the Axiom of Choice). Now, we consider the countable collection of sets

$$V_q = V + q = \{v + q \mid v \in V\}, \quad q \in \mathbb{Q} \cap [-1, 1].$$

These sets have the following properties which are

1. The sets V_q are pairwise disjoint.
2. $\bigcup_{q\in\mathbb{Q}\cap[-1,1]} V_q \supset [0, 1]$.
3. By translation invariance of $\mathcal{L}^1$, all V_q have the same outer measure.
4. If $\mathcal{L}^1(V) = 0$, then $\mathcal{L}^1([0, 1]) = 0$.

5. If $\mathcal{L}^1(V) > 0$, then $\sum_{q\in\mathbb{Q}\cap[-1,1]} \mathcal{L}^1(V_q) = \infty$.

We know that $\mathcal{L}^1([0,1]) = 1$. That means $\mathcal{L}^1(V)$ cannot be 0. Moreover, in case $\mathcal{L}^1(V) > 0$, we have that

$$\sum_{q\in\mathbb{Q}\cap[-1,1]} \mathcal{L}^1(V_q) = \infty > 2 \geq \mathcal{L}^1\left(\bigcup_{q\in\mathbb{Q}\cap[-1,1]} V_q\right)$$

which is a contradiction. This violates the countable additivity property required for a measure which proving that $\mathcal{L}^1$ is not a measure. Given an outer measure, there are two classes of sets that are interesting: negligible sets and measurable sets.

Definition 3.1.1.16 (Negligible set) *Let $\mu : 2^X \to [0,\infty]$ be an outer measure. We say that a set $E \subset X$ is* μ-negligible *if $\mu(E) = 0$.*

We introduce a lemma to make determining whether a set is negligible easier.

Lemma 3.1.1.17 *Let $\mu : 2^X \to [0,\infty]$ be an outer measure and $E \subset X$. The following are equivalent*

1. *E is μ-negligible.*
2. *For every $\varepsilon > 0$, there exists a family $\{A_i\}_{i=1}^\infty$ with $E \subset \bigcup_{i=1}^\infty A_i$ such that*

$$\mu\left(\bigcup_{i=1}^\infty A_i\right) < \varepsilon.$$

Definition 3.1.1.18 (μ-almost) *Let $\mu : 2^X \to [0,\infty]$ be an outer measure. We say that a property holds for* μ-almost *for every $x \in X$ if it holds for all $x \in X \setminus N$, where $N \subset X$ is μ-negligible.*

We have shown that Lebesgue outer measure is not actually an measure. A question comes up that given an outer measure μ on X, is it possible to restrict it to a measure, in a particular σ-algebra, namely for which it is countably additive on pairwise disjoint families of that σ-algebra. The answer to this problem is yes and the class of sets which (countable) additivity holds can be characterized as follows.

Definition 3.1.1.19 (μ-measurable) *Let μ be an outer measure on X. A set $E \subset X$ is called* μ-measurable *if*

$$\mu(F) = \mu(E \setminus F) + \mu(E \cap F)$$

for each $F \subset X$.

Note that outer measure is defined for *all subset of X*. Those subsets who behave in a good way are called μ-measurable. The term *measurable* can sometimes be misleading.

Theorem 3.1.1.20 (Carathéodory's extension theorem) *Let X be a set and let $\mu : 2^X \to [0,\infty]$ be an outer measure. Then the family of sets*

$$\mathcal{M}_\mu = \{E \subset X \mid E \text{ is } \mu\text{-measurable}\}$$

is a σ-algebra and

$$\{E \in X \mid \mu(E) = 0\} \subset \mathcal{M}_\mu.$$

Moreover, we have that $\mu : \mathcal{M}_\mu \to [0,\infty]$ is a (countably additive) measure.

The Theorem 3.1.1.20 shows that $\mathcal{M}_\mu$ is the greatest σ-algebra where the restriction of μ is a measure. From now on, we say that (X, μ) is *measure space* which means μ is an outer measure on X and we consider its restriction to the σ-algebra of μ-measurable sets, where μ is countably additive measure. On measurable sets, we can prove the continuity properties for the outer measure μ by the following lemma.

Lemma 3.1.1.21 *Let $\mu : 2^X \to [0, \infty]$ be an outer measure. Let $\{A_i\}_{i\in\mathbb{N}} \subset X$ be a sequence of μ-measurable sets. Then*

1. *If $A_1 \subset A_2 \subset A_3 \subset \cdots$, then*

$$\lim_{i\to\infty} \mu(A_i) = \mu\left(\bigcup_{i=1}^{\infty} A_i\right).$$

2. *If $A_1 \supset A_2 \supset A_3 \supset \cdots$ and $\mu(A_1) < \infty$, then*

$$\lim_{i\to\infty} \mu(A_i) = \mu\left(\bigcap_{i=1}^{\infty} A_i\right).$$

Note that the assumption $\mu(A_1) < \infty$ in second case is crucial. We consider $X = \mathbb{R}$, $u = \mathcal{L}^1$ and the decreasing sequence of sets $A_i = (i, \infty)$. We have that $\mu(A_i) = \infty$ for each $i \in \mathbb{N}$. But, since $\bigcap_{i\in\mathbb{N}} A_i = \varnothing$, we know that $\mu(\bigcap_{i\in\mathbb{N}} A_i) = 0$.

We see that Theorem 3.1.1.20 constructs a measure from and outer measure by restricting it to the family of measurable sets. Then, is it possible to do the converse way, particularly to extend a measure on a σ-algebra to an outer measure. To answer that, we have the following proposition.

Proposition 3.1.1.22 (Hausdorff construction) *Let $(X, \mathcal{A}, \mu)$ be a measure space. We define the function $\mu^* : 2^X \to [0, \infty]$ by*

$$\mu^*(E) = \inf\{\mu(F) \mid E \subset F, F \in \mathcal{A}\}$$

for any $E \subset X$. Then μ^* is an outer measure and $\mu^*(E) = \mu(E)$ for every $E \in \mathcal{A}$.

While measures on arbitrary sets can exhibit complicated or unpredictable behavior, the presence of additional structure on the underlying space, such as a topological or a metric space, allows us to define well-behaved classes of measures. These measures, known as *regular measures*, respect and interact naturally with the space's structural properties.

Definition 3.1.1.23 (Regular measure) *Let $\mu : 2^X \to [0, \infty]$ be an outer measure. We say that μ is* regular *if for each set $E \subset X$, there exists a μ-measurable set $F \supset E$ such that $\mu(E) = \mu(F)$.*

A key feature of regular measures is that they extend continuity properties beyond measurable sets to arbitrary sets. In other words, properties like continuity from above and below, which typically hold only for measurable sets, generalize to any family of sets when the measure is regular.

Lemma 3.1.1.24 *Let $\mu : 2^X \to [0, \infty]$ be an outer measure. Let $\{A_i\}_{i\in\mathbb{N}}$ be a sequence of sets (not necessarily μ-measurable). Then*

1. *If* $A_1 \subset A_2 \subset A_3 \subset \cdots$, *then*

$$\lim_{i\to\infty} \mu(A_i) = \mu\left(\bigcup_{i=1}^{\infty} A_i\right).$$

2. *If* $A_1 \supset A_2 \supset A_3 \supset \cdots$ *and* $\mu(A_1) < \infty$, *then*

$$\lim_{i\to\infty} \mu(A_i) = \mu\left(\bigcap_{i=1}^{\infty} A_i\right).$$

When examining sets with additional structure - specifically topological spaces and metric spaces - we encounter two natural σ-algebras which are Borel σ-algebra (from the topology) and one from the collection of μ-measurable sets. Of particular interest is the case when the topologically-generated σ-algebra is contained within the σ-algebra of μ-measurable sets. This inclusion relationship ensures that topological properties naturally translate into measure-theoretic properties.

Definition 3.1.1.25 *Let* (X, τ) *be a topological space, and* $\mu : 2^X \to [0, \infty]$ *be an outer measure on* X. *We say that* μ *is*

1. Borel *if every Borel set is* μ*-measurable.*
2. Borel regular *if it is a Borel outer measure and for every* $E \subset X$, *there exists a Borel set* $B \subset X$ *with* $E \subset B$ *such that* $\mu(E) = \mu(B)$.

There is a nice characterization of Borel measures in metric spaces.

Theorem 3.1.1.26 (Carathéodory criterion) *Let* (X, d) *be a metric space. Let* $\mu : 2^X \to [0, \infty]$ *be an outer measure. Then* μ *is Borel if and only if*

$$\mu(A \cup B) = \mu(A) + \mu(B)$$

for every $A, B \subset X$ *with* $d(A, B) = \inf\{d(x, y) : x \in A, y \in B\} > 0$.

From Carathédory criterion it is possible to see that Lebesgue measure $\mathcal{L}^N$ is a Borel measure. A fundamental feature of Borel regular measures is their capacity to reduce measure calculations to the realm of Borel sets. Specifically, for any set E, we can find a Borel set B containing E such that $\mu(E) = \mu(B)$. This reduction principle is invaluable in analysis and probability theory, as Borel sets possess rich structural properties that make them more amenable to theoretical and practical manipulation. We now introduce about two type regular properties for topological space.

Definition 3.1.1.27 (Inner and outer regularity) *Let* (X, τ) *be a topological space. Let* $\mu : 2^X \to [0, \infty]$ *be an outer measure on* X. *A set* $E \subset X$ *is said to be*

1. Inner regular *if*
$$\mu(E) = \sup\{\mu(K) \mid K \subset E, K \text{ is compact}\}.$$
2. Outer regular *if*
$$\mu(E) = \inf\{\mu(A) \mid E \subset A, A \text{ is open}\}$$

A class of measures that plays a central role in measure theory is that of *Radon measures*.

Definition 3.1.1.28 (Randon outer measure) *Let (X, τ) be a topological space. Let $\mu : 2^X \to [0, \infty]$ be an outer measure on X. We say that μ is a* Radon outer measure *if*

1. *μ is a Borel regular outer measure.*
2. *$\mu(K) < \infty$ for every compact set $K \subset X$.*
3. *Every open set $A \subset X$ is inner regular.*
4. *Every set $E \subset X$ is outer regular.*

Remark 3.1.1.29. The Lebesgue measure $\mathcal{L}^N$ is a Radon measure.

The property of inner regularity extends beyond just open sets to a broader class of sets. To properly understand this extension, we must first introduce a special category of sets. These sets become particularly crucial when dealing with infinite measure spaces and play a fundamental role in establishing measure-theoretic properties for sets of infinite measure.

Definition 3.1.1.30 (σ-finity) *A set $E \subset X$ is said to be* σ-finite for the outer measure $\mu : 2^X \to [0, \infty]$ *if there exists a family $\{F_i\}_{i \in \mathbb{N}}$ with $\mu(F_i) < \infty$ for all $i \in \mathbb{N}$, such that*

$$E = \bigcup_{i=1}^{\infty} F_i.$$

We say that the outer measure μ is σ-finite *if X is σ-finite.*

Remark 3.1.1.31. The Lebesgue measure $\mathcal{L}^N$ is σ-finite.

Lemma 3.1.1.32 *Let (X, τ) be a topological space. Let $\mu : 2^X \to [0, \infty]$ be an outer Radon measure on X. Then every σ-finite μ-measurable set $E \subset X$ is inner regular.*

We now conclude this part with a proposition about the relation between Borel regularity and Radon outer measures.

Proposition 3.1.1.33 *Let (X, τ) be a topological space. Let $\mu : 2^X \to [0, \infty]$ be an outer measure on X. The followings hold*

1. *If μ is a Radon outer measure, then it is a Borel regular.*
2. *Assume that X is a locally compact Hausdorff space that can be written as a countable union of compact sets. Moreover, assume that μ is a Borel outer measure that is finite on compact sets. Then μ is a Radon outer measure.*

Measurable Map

A *measurable map* is a map between the underlying sets of two measurable spaces that preserves the structure of the spaces, i.e. the preimage of any measurable set is measurable. This is in direct analogy to the definition that a continuous function between topological spaces preserves the topological structure: the preimage of any open set is open. Measurable maps are later used in the definition of the Lebesgue integral. In probability theory, a measurable function on a probability space is known as a random variable.

Definition 3.1.1.34 (Measurability) *Given two measurable spaces $(\Omega, \mathcal{F})$ and $(\Gamma, \mathcal{G})$. A map $f : \Omega \to \Gamma$ is said to be* $(\mathcal{F}, \mathcal{G})$-measurable *if*

$$f^{-1}(G) \in \mathcal{F}, \forall G \in \mathcal{G}.$$

Proposition 3.1.1.35 *Given two measurable spaces $(\Omega, \mathcal{F})$ and $(\Gamma, \mathcal{G})$. If a map $f : \Omega \to \Gamma$ is $(\mathcal{F} \to \mathcal{G})$-measurable, then the collection $\{f^{-1}(G) : G \in \mathcal{G}\}$ is a sub-σ-algebra of $\mathcal{F}$. Moreover, it is the smallest σ-algebra with respect to which f is measurable.*

Lebesgue Integral

We now turn to the definition of integral by following the idea of *Lebesgue's integration*. In all of this part, X will be a set and $\mu : 2^X \to [0, \infty]$ will be an outer measure. Moreover, we will denote by $\overline{\mathbb{R}}$ the extended set of real number which is $\mathbb{R} \cup \{-\infty, +\infty\}$.

The construction of the *Lebesgue integral* requires a systematic approach to function approximation. Similar to the *Riemann integral*, the Lebesgue integral is established through a limiting process where general functions are approximated by simpler ones with known integral values. However, the fundamental distinction lies in the Lebesgue approach's capacity to accommodate a broader class of simple functions, thereby extending the scope of integration theory beyond the limitations of Riemann integration.

Definition 3.1.1.36 (Characteristic function) *Given a set $E \subset X$, we define the characteristic function of E as*

$$\mathbf{1}_E(x) = \begin{cases} 1 & \text{if } x \in E, \\ 0 & \text{if } x \notin E. \end{cases}$$

Note that in some books, the characteristic function of a set E is denoted by χ_E instead.

Definition 3.1.1.37 (Simplicity) *We say that a μ-measurable function $f : X \to \mathbb{R}$ is* simple *if its image is finite, namely if it is possible to write*

$$f(x) = \sum_{i=1}^{k} \mathbf{1}_{E_i}(x) y_i$$

for all $x \in X$, where $k \in \mathbb{N}$, $E_i \subset X$ and $y_1, \ldots, y_k \in Y$.

Note that it is always possible to write a simple function $f : X \to \mathbb{R}$ as

$$f(x) = \sum_{i=1}^{k} \mathbf{1}_{E_i}(x) y_i$$

in which the sets $E_1, \ldots, E_k$ are pairwise disjoint. Now, we define the notion of Lebesgue integral.

Definition 3.1.1.38 (Lebesgue intergral of a simple function) *Let $f : X \to \mathbb{R}$ be a simple μ-measurable function defined by*

$$f(x) = \sum_{i=1}^{k} \mathbf{1}_{E_i}(x) y_i.$$

We define the Lebesgue integral *of f with respect to μ by*

$$\int_X f \mathrm{d}\mu = \sum_{i=1}^{k} y_i \mu(E_i)$$

with the convention that if $y_i = 0$ and $\mu(E_i) = \infty$, then $y_i \mu(E_i) = 0$.

Remark 3.1.1.39. The μ-measurability is needed in order to have a well-defined object. Indeed, if

$$f(x) = \sum_{i=1}^{k} \mathbf{1}_{E_i}(x) y_i = \sum_{j=1}^{m} \mathbf{1}_{F_j}(x) z_j,$$

we would like

$$\sum_{i=1}^{k} y_i \mu(E_i) = \sum_{i=j}^{m} z_j \mu(F_j)$$

for any choice of the sets F_j's. This is precisely requiring that the sets E_i's are μ-measurable.

Definition 3.1.1.40 (Positive and negative part) *Given a function $f : X \to \mathbb{R} \cup \{\infty\}$, we define its* positive *and* negative *part by*

$$f^+ := \max\{f, 0\} \quad \text{and} \quad f^- := \max\{-f, 0\},$$

respectively.

Remark 3.1.1.41. We always have that $f^+, f^- \geq 0$. Moreover, we know that $f = f^+ - f^-$, $|f| = f^+ + f^-$.

Definition 3.1.1.42 (Lebesgue intergral of a positive function) *Given a μ-measurable function $f : X \to [0, \infty]$, we define the* Lebesgue integral of f *with respect to μ by*

$$\int_X f \mathrm{d}\mu = \sup \left\{ \int_X g \mathrm{d}\mu \mid g \text{ is simple, } \mu\text{-measurable and } g \leq f \right\}.$$

Definition 3.1.1.43 (Lebesgue intergral of a generic function) *Let $f : X \to \mathbb{R}$ be a μ-measurable function. Assume that f is μ-integrable, namely that*

$$\int_X f^+ \mathrm{d}\mu < \infty \quad \text{or} \quad \int_X f^- \mathrm{d}\mu < \infty. \tag{3.1.1}$$

We define the Lebesgue integral of f *with respect to μ by*

$$\int_X f \mathrm{d}\mu = \int_X f^+ \mathrm{d}\mu - \int_X f^- \mathrm{d}\mu.$$

Remark 3.1.1.44. Assumptions in (3.1.1) is in order to avoid $+\infty$ and $-\infty$ in the definition of the integral.

Proposition 3.1.1.45 *We say that a μ-integrable function $f : X \to \mathbb{R}$ belongs to the space $\mathcal{L}^1(X, \mu)$ if*

$$\int_X |f| \mathrm{d}\mu < \infty.$$

The Lebesgue integral satisfies some basic properties.

Lemma 3.1.1.46 *Let $f, g : X \to \mathbb{R}$ be μ-integrable. Then*

$$\int_X (af + bg) \mathrm{d}\mu = a \int_X f \mathrm{d}\mu + b \int_X g \mathrm{d}\mu.$$

for all $a, b \in \mathbb{R}$. Moreover, if $f \leq g$ μ-a.e., then

$$\int_X f \mathrm{d}\mu \leq \int_X g \mathrm{d}\mu.$$

Finally, if $f = g$ μ-a.e., then

$$\int_X f \mathrm{d}\mu = \int_X g \mathrm{d}\mu.$$

Now, we give some discussions about the Lebesgue and Riemann integral. The fundamental distinction between Lebesgue and Riemann integration lies in their underlying constructive approaches. While both methods serve to evaluate integrals, their theoretical foundations differ significantly. The Riemann integral, traditionally introduced in elementary analysis courses, operates by partitioning the domain of the function. In contrast, the Lebesgue approach innovates by partitioning the target space (Figure **??**).

This distinction proves one crucial thing that the class of simple functions employed in Riemann integration constitutes merely a subset of those utilized in Lebesgue integration. This broader framework of the Lebesgue approach results in a more general theory, relaxing the stringent regularity requirements inherent in Riemann integration. Consequently, the Lebesgue integral accommodates a wider class of functions, making it a more powerful tool in modern analysis.

The fundamental difference between Riemann and Lebesgue integration also lies in their approach to partitioning. Figure 3.1, adapted from [15], illustrates this contrast, showing how Riemann integration partitions the domain, while Lebesgue integration partitions the range of the function.

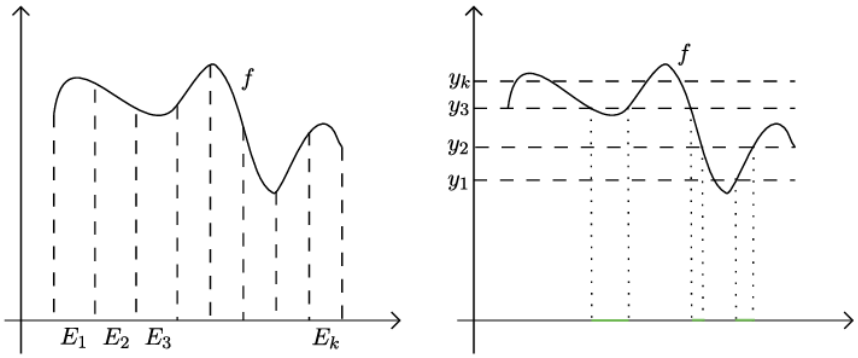


Figure 3.1: Comparison between Riemann integral (left) and Lebesgue integral (right) approaches to partitioning.

The relationship between Riemann and Lebesgue integrability is characterized by the below fundamental theorem.

Theorem 3.1.1.47 (Riemann-Lebesgue Theorem) *A function $f : \mathbb{R}^N \to \mathbb{R}^M$ is Riemann integrable if and only if the set of discontinuities of f has Lebesgue measure zero.*

In Riemann integral, the function f is required to be discontinuous at finite number of points, which is less general than Lebesgue measurability. Lebesgue integral allows to consider functions that are discontinuous everywhere. As an example, let us consider the characteristic function of the irrational numbers $\mathbf{1}_{\mathbb{R}\setminus\mathbb{Q}}$. This function is not Riemann integrable, but it is easily seen to be Lebesgue integrable and

$$\int_{(a,b)} \mathbf{1}_{\mathbb{R}\setminus\mathbb{Q}} \mathrm{d}x = b - a$$

for every $a, b \in \mathbb{R}$ with $a < b$.

Lemma 3.1.1.48 *If $f : \mathbb{R}^N \to \mathbb{R}^M$ is Riemann integrable then it is Lebesgue integrable and the two integrals coincide.*

The necessity for Lebesgue integration arises from the inherent limitations of Riemann integration. While the Riemann integral offers an intuitive geometric interpretation, its practical applicability is significantly constrained by stringent requirements on the integrand's behavior, particularly in the context of limiting processes.

Theorem 3.1.1.49 (Continuity of the Riemann integral)) *Let $f_n : [a, b] \to \mathbb{R}$ be a sequence of Riemann integrable functions. Assume that $f_n \to f$ uniformly. Then f is Riemann integrable and*

$$\lim_{n\to\infty} \int_a^b f_n(x)\,\mathrm{d}x = \int_a^b \lim_{n\to\infty} f_n(x)\,\mathrm{d}x = \int_a^b f(x)\,\mathrm{d}x.$$

The requirement of uniform continuity represents a particularly strong condition that is rarely satisfied in practical applications. In many cases, we encounter sequences of functions where only weaker forms of convergence or asymptotic behavior are known. This limitation of the Riemann integral becomes particularly evident when dealing with limiting processes. The subsequent results address three crucial scenarios where the Riemann integral proves inadequate, specifically in cases where the pointwise limit may not exist, yet meaningful conclusions about the integral sequence can still be drawn.

Lemma 3.1.1.50 (Fatou's lemma) *Let $\{f_n\}_{n\in\mathbb{N}}$ be a sequence of μ-measurable functions. If $f_n \geq g$ for all $n \in \mathbb{N}$ in which $g \in \mathcal{L}^1(X, \mu)$, then*

$$\int_X \liminf_{n\to\infty} f_n \mathrm{d}\mu \leq \liminf_{n\to\infty} \int_X f_n \mathrm{d}\mu.$$

If $f_n \leq g$ for all $n \in \mathbb{N}$ in which $g \in \mathcal{L}^1(X, \mu)$, then

$$\limsup_{n\to\infty} \int_X f_n \mathrm{d}\mu \leq \int_X \limsup_{n\to\infty} f_n \mathrm{d}\mu.$$

A special case is when the sequence $\{f_n\}_{n\in\mathbb{N}}$ is monotone, since in that case the pointwise limit $\lim_{n\to\infty} f_n(x)$ exists for all $x \in X$.

Theorem 3.1.1.51 (Lebesgue's monotone convergence theorem) *Let $\{f_n\}_{n\in\mathbb{N}}$ be an increasing sequence of μ-measurable functions such that $f_n \geq g$ with $g \in \mathcal{L}^1(X, \mu)$. Then*

$$\lim_{n\to\infty} \int_X f_n \mathrm{d}\mu = \int_X \lim_{n\to\infty} f_n \mathrm{d}\mu.$$

Finally, if the pointwise limit of a sequence of functions is known, but the sequence is not monotone, we wonder whether or not this translates into convergence of the integrals of that sequence. Next important result gives a positive answer under very mild assumptions.

Theorem 3.1.1.52 (Lebesgue's dominated convergence theorem) *Let $\{f_n\}_{n\in\mathbb{N}}$ be a sequence of μ-measurable functions such that*

$$f_n(x) \to f(x)$$

for μ-a.e. $x \in X$. We assume that

$$|f_n| \leq g$$

in which $g \in \mathcal{L}^1(X, \mu)$. Then $f \in L^1(X, \mu)$ and

$$\lim_{n\to\infty} \int_X |f_n - f| \mathrm{d}\mu = 0.$$

In particular,

$$\lim_{n\to\infty} \int_X f_n \mathrm{d}\mu = \int_X f \mathrm{d}\mu.$$

3.1.2 Probability Theory

In this part, we establish fundamental concepts from probability theory that will be essential for our later discussions.

Probability Space

Definition 3.1.2.1 (Probability measure) *Let $(\Omega, \mathcal{F})$ be a measurable space. A measure $\mathbb{P}$ on $(\Omega, \mathcal{F})$ is called a* probability measure *if $\mathbb{P}(\Omega) = 1$. The triplet $(\Omega, \mathcal{F}, \mathbb{P})$ is called a* probability space.

Here is a familiar example of probability space about rolling dices.

Example 3.1.2.2. We consider a single roll of a fair six-sided die. We define the sample space $\Omega = \{1, 2, 3, 4, 5, 6\}$ representing all possible outcomes. Using classical probability, we can determine the probabilities of specific events, for instance, the probability of rolling an odd number is

$$\mathbb{P}(\{1, 3, 5\}) = \frac{3}{6} = \frac{1}{2},$$

the probability of rolling a number smaller than 3 is

$$\mathbb{P}(\{1, 2\}) = \frac{2}{6} = \frac{1}{3}.$$

Thus, the complete probability space $(\Omega, \mathcal{F}, \mathbb{P})$ for this experiment can be expressed as following

- Sample space $\Omega = \{1, 2, 3, 4, 5, 6\}$.
- σ-algebra $\mathcal{F} = 2^{\Omega}$.
- Probability measure $\mathbb{P}$ defined by
$$\mathbb{P}(A) = \frac{|A|}{6}$$
for every $A \in \mathcal{F}$ in which $|A|$ denotes the number of outcomes in event A.

Remark 3.1.2.3. By convention, for every $A, B \in \mathcal{F}$, we usually write the *joint probability* $\mathbb{P}(A \cap B)$ by $\mathbb{P}(A, B)$.

Definition 3.1.2.4 (Almost surely holding property) *Given a probability space $(\Omega, \mathcal{F}, \mathbb{P})$. A property $\mathcal{P} : \Omega \to \{0, 1\}$ is said to hold* almost surely *with respect to $\mathbb{P}$, abbreviated by $\mathbb{P}$-a.s., if*

$$\mathbb{P}\left(\{\omega \in \Omega \mid \mathcal{P}(\omega)\}\right) = 1.$$

If $\mathbb{P}$ is clear, we can just say $\mathcal{P}$ a.s.

Random Variable

A probability calculation can sometimes be simplified by choosing a suitable probability space. Now, we consider the following examples

Example 3.1.2.5. There are four students Adam, Ben, Cam and Don seated randomly on four chairs labeled A, B, C, and D. Supposing that we want to find the probability that Adam specifically sits on chair A. Initially, there are $4! = 24$ ways to seat the four students. But, if we fix

Adam on chair A, we only need to arrange the remaining three students, giving $3! = 6$ ways. Thus, the probability is calculated as

$$\frac{6}{24} = \frac{1}{4}.$$

Alternatively, if we assume there's no bias toward any particular chair, Adam's chance of sitting on chair A is immediately clear as $\frac{1}{4}$. Therefore, we have two probability spaces here. The first space $(\Omega_1, \mathcal{F}_1, \mathbb{P}_1)$ has 24 outcomes and the second simplified space $(\Omega_2, \mathcal{F}_2, \mathbb{P}_2)$ has only 4 outcomes.

The relationship between different probability spaces highlights the existence of mappings from a larger probability space to a smaller one. Suppose we have a mapping $X : \Omega_1 \to \Omega_2$, such that for any events $E_1 \in \mathcal{F}_1$ and $E_2 \in \mathcal{F}_2$ that represent the same semantic outcomes, we must have $\mathbb{P}_1(E_1) = \mathbb{P}_2(E_2)$. We consider the next example.

Example 3.1.2.6. Referring to Example 3.1.2.5, we define the outcome as a pair (X, Y) representing student X sits on chair Y. For the first probability space, we have that

$$\Omega_1 = \{((\text{Adam}, A), (\text{Ben}, B), (\text{Cam}, C), (\text{Don}, D)), \dots, ((\text{Adam}, D), (\text{Ben}, C), (\text{Cam}, B), (\text{Don}, A))\}$$

and the probability is uniform as

$$\mathbb{P}_1(E_1) = \frac{|E_1|}{|\Omega_1|}$$

for every $E_1 \subseteq \Omega_1$. For the simpler probability space, we have that

$$\Omega_2 = \{(\text{Adam}, A), (\text{Adam}, B), (\text{Adam}, C), (\text{Adam}, D)\}$$

and the probability measure

$$\mathbb{P}_2(E_2) = \frac{|E_2|}{|\Omega_2|}$$

for all $E_2 \subseteq \Omega_2$. The mapping $X : \Omega_1 \to \Omega_2$ is explicitly given as

$$X(((\text{Adam}, Y), \dots)) = (\text{Adam}, Y)$$

for every $Y \in \{A, B, C, D\}$. It follows that this mapping preserves probability measures between these spaces.

Definition 3.1.2.7 (Probability preservation) *Let $(\Omega_1, \mathcal{F}_1, \mathbb{P}_1)$ and $(\Omega_2, \mathcal{F}_2, \mathbb{P}_2)$ be two probability spaces. A measurable map $X : \Omega_1 \to \Omega_2$ is said to be* probability preserving *if*

$$\mathbb{P}_1(X^{-1}(E_2)) = \mathbb{P}_2(E_2), \ \forall E_2 \in \mathcal{F}_2.$$

Theorem 3.1.2.8 *Let $(\Omega, \mathcal{F}_1, \mathbb{P}_1)$ be a probability space and $(\Omega_2, \mathcal{F}_2)$ be a measurable space. For every measurable map $X : \Omega_1 \to \Omega_2$, there exists a probability measure $\mathbb{P}_2$ such that X is a probability preserving.*

Proof. Since X is measurable, for all $E_2 \in \mathcal{F}_2$, $\mathbb{P}_1(X^{-1}(E_2))$ is well-defined. We can define the measure $\mathbb{P}_2$ such that

$$\mathbb{P}_2(E_2) = \mathbb{P}_1(X^{-1}(E_2))$$

for all $E_2 \in \mathcal{F}_2$. □

For computational purposes, we select Ω_2 to be a real vector space, $\mathcal{F}_2 = \mathcal{B}(\Omega)$. The map X is then called a *random variable*.

Definition 3.1.2.9 (Random variable and vector) *Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space. A function $X : \Omega \to \mathbb{R}^d$ is called a* d-dimensional random variable *if X is $(\mathcal{F}, \mathcal{B}(\mathbb{R}^d))$-measurable. If $X_1, \ldots, X_N$ in which $N \geq 2$ are random variables, then $(X_1, \ldots, X_N)$ is called a* random vector.

Typically, random variables are initially introduced within probability theory as functions mapping outcomes to numerical values, making it more intuitive to work directly with numerical results rather than abstract events. In practical computations and numerical analysis, we generally focus on real-valued outcomes of these random variables. Consequently, the underlying probability space $(\Omega, \mathcal{F}, \mathbb{P})$ and the measurability of X with respect to $\mathcal{F}$ are often implicitly assumed. For convenience and simplicity of notation, the event $\{\omega \in \Omega \mid X(\omega) \in B\}$ is abbreviated as $\mathbb{P}(X \in B)$.

Probability Density Function and Cumulative Distribution Function

Definition 3.1.2.10 (Probability density function) *A function $f : \mathbb{R}^d \to [0, \infty)$ is called a* probability density function *(PDF) if it satisfies*

1. $f(x) \geq 0$ *for all* $x \in \mathbb{R}^d$.
2. $\displaystyle\int_{\mathbb{R}^d} f(x)\mathrm{d}x = 1.$

Let X be a d-dimensional random variable. For a probability density function f of X, the probability of an event occurring in a region $A \subset \mathbb{R}^d$ is given by

$$\mathbb{P}(X \in A) = \int_A f(x)\mathrm{d}x.$$

Example 3.1.2.11. The standard normal distribution in one dimension has the probability density function f as

$$f(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}.$$

This function is symmetric about the y-axis and has its maximum value at $x = 0$.

Using the Definition 3.1.2.10 of probability density function, we can define a right-continuous monotone increasing function F called *cumulative distribution function* of X.

Definition 3.1.2.12 (Cumulative distribution function) *Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space. Let X be a d-dimensional random variable corresponding to that probability space. Then* cumulative distribution function $F : \mathbb{R}^d \to [0, 1]$ of X *(CDF) is the function given by*

$$F_X(x) = \mathbb{P}(X \leq x).$$

In fact, this function represents the probability that X has the value less than of equal to a particular value.

Remark 3.1.2.13. Let X be a random variable. From the Definition 3.1.2.12 above, we know that

$$\lim_{x \to -\infty} F_X(x) = 0 \quad \text{and} \quad \lim_{x \to \infty} F_X(x) = 1.$$

Moreover, we also have that

$$\mathbb{P}(a \leq X \leq b) = F_X(b) - F_X(a).$$

Kullback-Leibler Divergence

In probability theory and information theory, we often need to measure how different two probability distributions are from each other. The *Kullback-Leibler divergence* provides one way to quantify this difference. Before introducing Kullback-Leibler divergence, we first need to understand the concept of entropy.

Definition 3.1.2.14 (Differential entropy) *For a probability density function f on a space $\Omega = \mathbb{R}^d$, the* differential entropy *(continuous entropy) is defined as the concave energy*

$$H(f) = -\int_{\Omega} f(x) \log f(x) \mathrm{d}x.$$

Example 3.1.2.15. To understand the behavior of entropy $H(f)$, we consider two extreme cases. The first case is that the distribution is highly concentrated (approaching a Dirac delta function). Now, we have that $\log f(x)$ approaches ∞ in regions where $f(x)$ is large. That means $H(f)$ approaches $-\infty$. This reflects the high certainty or low randomness in the distribution. The second case is a uniform distribution where $f(x) \equiv 1$ over its domain. We have that

$$H(f) = -\int_{\Omega} 1 \log 1 \mathrm{d}x = 0.$$

This maximum value of entropy reflects the maximum uncertainty or randomness in the distribution.

Definition 3.1.2.16 (Kullback-Leibler divergence) *Given two probability density functions p and q on a space Ω, the Kullback-Leibler divergence from q to p is defined as*

$$KL(p \parallel q) = \int_{\Omega} p(x) \log \frac{p(x)}{q(x)} \mathrm{d}x + \int_{\Omega} p(x) - q(x) \mathrm{d}x. \tag{3.1.2}$$

Proposition 3.1.2.17 (Properties of Kullback-Leibler divergence) *The Kullback-Leibler divergence satisfies the following properties*

1. *$KL(p \parallel q) \geq 0$ for all probability density functions p and q.*
2. *$KL(p \parallel q) = 0$ if and only if $p = q$ almost everywhere.*
3. *In general, $KL(p \parallel q) \neq KL(q \parallel p)$.*

Proof. The non-negativity property follows from Jensen's inequality applied to the convex function $x \log x$. For any probability densities p and q, we have that

$$\mathrm{KL}(p \parallel q) = \int_{\Omega} p(x) \log \frac{p(x)}{q(x)} \mathrm{d}x - \int_{\Omega} p(x) - q(x) \mathrm{d}x.$$

The second integral is zero since both p and q are probability density function and Jensen's inequality ensures the first integral is non-negative. □

The Kullback-Leibler divergence and entropy are closely related through the equation

$$\mathrm{KL}(p \parallel q) = -H(p) - \int_{\Omega} p(x) \log q(x) \mathrm{d}x$$

in which $H(p)$ is the differential entropy defined earlier. This relationship highlights how Kullback-Leibler divergence measures the relative entropy between two distributions. While not a true metric due to its asymmetry, the Kullback-Leibler divergence is a fundamental concept in information theory and statistics. In the context of optimal transport theory, it plays a crucial role in the development of efficient computational methods, particularly in the Regularized Wasserstein distance which we will discuss in Section 5.1 of Chapter 5.

3.1.3 Convex Analysis Theory

Convex analysis is a specialized field within applied mathematics that focuses on understanding the characteristics of convex sets and functions. It plays a crucial role in optimization, providing the theoretical framework for various mathematical techniques used to solve optimization problems efficiently.

In this part, we will cover topics about basic convexity concepts in $\mathbb{R}^n$, topological properties of convex sets in $\mathbb{R}^n$, separation of convex sets and convex function.

Subspaces, Affine Manifolds and Convex Sets

Definition 3.1.3.1 *Let $x_1, x_2, \ldots, x_k \in \mathbb{R}^n$ be given vectors and let $\lambda_1, \lambda_2, \ldots, \lambda_k \in \mathbb{R}$ be given numbers. Then a point of the form*

$$x = \lambda_1 x_1 + \lambda_2 x_2 + \cdots + \lambda_k x_k = \sum_{i=1}^{k} \lambda_i x_i \tag{1.1}$$

is called

(i) the linear combination of $x_1, \ldots, x_k$;

(ii) the affine combination of $x_1, \ldots, x_k$, if $\sum_{i=1}^{k} \lambda_i = 1$;

(iii) the conic combination of $x_1, \ldots, x_k$, if $\lambda_i \geq 0$ for each $i \in \{1, 2, \ldots, k\}$;

(iv) the convex combination of $x_1, \ldots, x_k$, if $\lambda_i \geq 0$ for each $i \in \{1, 2, \ldots, k\}$ and $\sum_{i=1}^{k} \lambda_i = 1$.

Definition 3.1.3.2 *A non-empty subset V of $\mathbb{R}^n$ is called*

(i) a linear subspace if for any $x, y \in V$ and any $\lambda \in \mathbb{R}$, both the points $x + y$ and λx lie in V;

(ii) an affine manifold if for any two distinct points $x, y \in V$ and any $\lambda \in \mathbb{R}$, we have

$$\lambda x + (1 - \lambda) y \in V.$$

Example 3.1.3.3. Given non-zero vector $p \in \mathbb{R}^n$ and given number $\alpha \in \mathbb{R}$, a hyperplane is a set of the form

$$H_{p,\alpha} = \{x \in \mathbb{R}^n : \langle p, x \rangle = \alpha\}.$$

Let $x_1, x_2, ..., x_k \in H_{p,\alpha}$ and $\lambda_1, \lambda_2, ..., \lambda_k \in \mathbb{R}$, we have for $x = \sum_{i=1}^{k} \lambda_i x_i$

$$\begin{aligned} \langle p, x \rangle &= \left\langle p, \sum_{i=1}^{k} \lambda_i x_i \right\rangle = \sum_{i=1}^{k} \langle p, \lambda_i x_i \rangle \\ &= \sum_{i=1}^{k} \lambda_i \langle p, x_i \rangle = \sum_{i=1}^{k} \lambda_i = 1. \end{aligned}$$

Therefore, $x \in H_{p,\alpha}$, which means $H_{p,\alpha}$ is a affine manifold.

We have the following characterization of affine manifolds.

Proposition 3.1.3.4 *A subset V of $\mathbb{R}^n$ is an affine manifold if and only if there exists $a \in \mathbb{R}^n$ and a linear subspace $V_0 \subset \mathbb{R}^n$ such that*

$$V = a + V_0,$$

i.e. V is a translation of the linear subspace V_0 of $\mathbb{R}^n$. The linear subspace V_0 is called the direction of V.

Proof. Suppose V is an affine manifold and let $a \in V$, then the set

$$V_0 := V - a = \{v - a : v \in V\},$$

is a linear subspace. Indeed, it is easy to see that $0 = a - a \in V_0$. Now, let $v_1, v_2 \in V_0$ and $\alpha, \beta \in \mathbb{R}$ be two scalars, we have

$$\alpha v_1 + \beta v_2 + a = \alpha(v_1 + a) + \beta(v_2 + a) + (1 - \alpha - \beta)a.$$

Since V is an affine manifold, then it is stable under affine combination. Hence, we have

$$\alpha(v_1 + a) + \beta(v_2 + a) + (1 - \alpha - \beta)a \in V.$$

Consequently,

$$\alpha v_1 + \beta v_2 \in V - a := V_0,$$

which means that V_0 is a linear subspace under linear combination, and therefore V_0 is a linear subspace of $\mathbb{R}^n$. Reciprocally, it is clear that if V_0 is a linear space and $a \in \mathbb{R}^n$, then $V := a + V_0$ is stable under affine combination and therefore V is an affine manifold. □

Definition 3.1.3.5 *Let A be a set in $\mathbb{R}^n$. The affine hull, denoted by aff(A), is the smallest affine manifold containing the set A, i.e. if V is an affine manifold such that $A \subset V$, then aff$(A) \subset V$*

Example 3.1.3.6. Let $p \in \mathbb{R}^n$ be a fixed point and let

$$S \;=\; \big\{\, p,\; p + v_1,\; p + v_2,\; \ldots,\; p + v_k \big\},$$

where the direction vectors $v_1, \ldots, v_k \in \mathbb{R}^n$ are linearly independent $(1 \le k \le n)$. The *affine hull* of S is

$$\mathrm{aff}(S) \;=\; p + \mathrm{span}\{v_1, \ldots, v_k\} \;=\; \Big\{ p + \sum_{i=1}^{k} \lambda_i v_i \;\Big|\; \lambda_1, \ldots, \lambda_k \in \mathbb{R} \Big\}.$$

Hence $\mathrm{aff}(S)$ is an affine subspace of $\mathbb{R}^n$ of dimension k. In particular, if $k = n$ and the v_i form a basis of $\mathbb{R}^n$, then $\mathrm{aff}(S) = \mathbb{R}^n$.

Proposition 3.1.3.7 *A non-empty subset V of $\mathbb{R}^n$ is an affine manifold if and only if for some $m \in \mathbb{N}^*$, there exists a matrix $A \in \mathbb{R}^{m\times n}$ and a vector $b \in \mathbb{R}^m$ such that*

$$V = \{x \in \mathbb{R}^n : Ax = b\}.$$

Proof. Assume V is a non-empty affine manifold. Pick a reference point $x_0 \in V$ and define the translation of V to the origin,

$$L \;=\; V - x_0 \;=\; \{\, x - x_0 : x \in V\}.$$

Because V is affine, L is a linear subspace of $\mathbb{R}^n$. Let $\{u_1, \dots, u_k\}$ be a basis of L; then its orthogonal complement $L^\perp$ has dimension $m := n - k$. Choose an orthonormal basis $\{w_1, \dots, w_m\}$ of $L^\perp$ and assemble the row vectors into a matrix

$$A \;=\; \begin{pmatrix} w_1^{\mathsf{T}} \\ \vdots \\ w_m^{\mathsf{T}} \end{pmatrix} \in \mathbb{R}^{m\times n}, \qquad b \;=\; Ax_0 \in \mathbb{R}^m.$$

For any $x \in V$ we have $x - x_0 \in L$, hence $w_i^{\mathsf{T}}(x - x_0) = 0$ for every i. Consequently $Ax = Ax_0 = b$, showing $V \subseteq \{x : Ax = b\}$. Conversely, if $Ax = b$, then $A(x - x_0) = 0$, so $x - x_0 \in L$ and therefore $x \in x_0 + L = V$. Thus equality holds.

Conversely, suppose $V = \{x \in \mathbb{R}^n : Ax = b\}$ for some $A \in \mathbb{R}^{m\times n}$ and $b \in \mathbb{R}^m$. Let $x_1, x_2 \in V$ and $t \in \mathbb{R}$. Because A is linear,

$$A\big(tx_1 + (1-t)x_2\big) = tAx_1 + (1-t)Ax_2 = tb + (1-t)b = b,$$

so every affine combination of points in V remains in V. Hence V is closed under affine combinations and is therefore an affine manifold. □

Definition 3.1.3.8 (Convex set) *A subset C of $\mathbb{R}^d$ is said to be* convex *if for any $x, y \in C$, the line segment $[x, y]$ is contained within C. In other words, we have that for every $(x, y) \in C^2$ and $\lambda \in [0, 1]$,*

$$\lambda x + (1 - \lambda)y \in C.$$

Example 3.1.3.9. We can easily find some of the convex sets in $\mathbb{R}^d$. Two of the most common examples are disk in $\mathbb{R}^2$ and sphere in $\mathbb{R}^3$. Let $D(c, r)$ be a disk in $\mathbb{R}^2$ whose radius $r > 0$ and center $c \in \mathbb{R}^2$. Let x and y be two points which belong to $D(c, r)$. Let $\lambda \in [0, 1]$ be a number. Note that $\|x - c\|_2 \le r$ and $\|y - c\|_2 \le r$ in which $\|x\|_2$ denotes the ℓ^2-norm. We have that

$$\begin{aligned} \|\lambda x + (1-\lambda)y - c\|_2 &= \|\lambda(x-c) + (1-\lambda)(y-c)\|_2 \\ &\le \lambda\|x - c\|_2 + (1-\lambda)\|y - c\|_2 \\ &\le \lambda r + (1-\lambda)r = r. \end{aligned}$$

That means $\lambda x + (1 - \lambda)y$ belongs to $D(c, r)$. Thus, $D(c, r)$ is convex. The same procedure can be used to prove that a sphere is also convex in $\mathbb{R}^3$.

Remark 3.1.3.10. By Definition 3.1.3.8, we know that $\varnothing$ and $\mathbb{R}^d$ are convex as well.

Topological Properties of Convex Sets in $\mathbb{R}^n$

Let (X, d) be a metric space with distance $d : X \times X \to [0, \infty)$. An open ball of radius $r > 0$ centered at $x_0 \in X$ is defined as

$$B_r(x_0) = \{x \in X : d(x, x_0) < r\}.$$

Definition 3.1.3.11 *Let $A \subset \mathbb{R}^n$ be non-empty, a point $x \in A$ is* interior point *of A if*

$$\exists r > 0, B_r(x) \subset A.$$

The interior of A, denoted by int(A), is the set of all interior points of A.

Example 3.1.3.12. Consider the set $A = [0, 1]$, the interior set of A is $\text{int}(A) = (0, 1)$.

Definition 3.1.3.13 *Let $C \subset \mathbb{R}^n$, the relative interior of C, denoted as ri(C) is the interior of C in the relative topology induced by aff(C), i.e.*

$$ri(C) = \{x \in C : B_r(x) \cap aff(C) \subset C \text{ for some } r > 0\}.$$

Example 3.1.3.14. Consider the set

$$C \;=\; \{(x, y) \in \mathbb{R}^2 \mid y = 0,\; x \geq 0\}.$$

Its *affine hull* is the x-axis,

$$\text{aff}(C) \;=\; \{(x, 0) \mid x \in \mathbb{R}\}.$$

The *relative interior* of C is therefore taken with respect to this one-dimensional affine space:

$$\text{ri}(C) \;=\; \Big\{(x, 0) \in C \,\Big|\, x > 0\Big\}.$$

In words, every point on the x-axis with strictly positive x-coordinate is in the relative interior, while the "boundary points" $(0, 0)$ and all points with $y \neq 0$ are excluded.

Projection on a Convex Set

Suppose that C is a convex set in $\mathbb{R}^n$. For each $x \in \mathbb{R}^n$, we want to find a point $\bar{x} \in C$ which minimizes the distance from x to C, i.e.

$$\|x - \bar{x}\| \leq \|x - c\| \quad \text{for all} \quad c \in C.$$

Theorem 3.1.3.15 (Projection onto a closed convex set) *Let $C \subset \mathbb{R}^n$ be non-empty, closed, and convex, and fix any $x \in \mathbb{R}^n$. Then there exists a unique point $\bar{x} \in C$ such that*

$$\|x - \bar{x}\| \;=\; \min_{y \in C} \|x - y\|.$$

Moreover, this optimal point is characterised by the following equivalent conditions:

(i) $\bar{x} \in C$;

(ii) $\langle x - \bar{x},\, c - \bar{x} \rangle \;\leq 0$ *for all* $c \in C$.

Proof. If $x \in C$ we simply take $\bar{x} = x$. Assume $x \notin C$ and pick an arbitrary $x_0 \in C$. Set $r := \|x - x_0\| > 0$ and define the closed, bounded set

$$K \;:=\; \big(x + r\overline{B}\,\big) \cap C,$$

where $\overline{B}$ is the closed unit ball. Because C is closed, K is closed and because we intersect with a ball of finite radius, K is also bounded, hence compact. The continuous map $f(y) := \|x - y\|$ attains its minimum on K, say at $\bar{x} \in K$. For any $y \in K$ we have $\|x - \bar{x}\| \leq \|x - y\|$. Moreover, for any $y \in C \setminus K$ we even have $\|x - y\| > r \geq \|x - \bar{x}\|$. Therefore

$$\|x - \bar{x}\| \;=\; \min_{y \in C} \|x - y\|,$$

so a minimizer exists. Suppose, for contradiction, that there is another point $x_1 \in C$ with $x_1 \neq \bar{x}$ and the same distance $d := \|x - \bar{x}\| = \|x - x_1\|$. Convexity of C gives $\dfrac{x_1 + \bar{x}}{2} \in C$. We then obtain

$$\begin{aligned} d^2 &\leq \left\| \frac{x - x_1}{2} + \frac{x - \bar{x}}{2} \right\|^2 \\ &= \left\langle \frac{x - x_1}{2} + \frac{x - \bar{x}}{2}, \frac{x - x_1}{2} + \frac{x - \bar{x}}{2} \right\rangle \\ &= \left\langle \frac{x - x_1}{2}, \frac{x - x_1}{2} \right\rangle + \left\langle \frac{x - \bar{x}}{2}, \frac{x - \bar{x}}{2} \right\rangle + 2 \left\langle \frac{x - \bar{x}}{2}, \frac{x - x_1}{2} \right\rangle \\ &= \frac{1}{4} d^2 + \frac{1}{4} d^2 + \frac{1}{2} \langle x - \bar{x}, x - x_1 \rangle . \end{aligned}$$

Therefore,

$$d^2 \leq \langle x - \bar{x}, x - x_1 \rangle \leq \|x - \bar{x}\| \|x - x_1\| = d^2.$$

Hence $\langle x - \bar{x}, x - x_1 \rangle = d^2 = \|x - \bar{x}\| \|x - x_1\|$. By Cauchy-Schwarz inequality, $x - \bar{x} = \lambda(x - x_1)$ for some $\lambda \in \mathbb{R}$. Then $d = \|x - \bar{x}\| = |\lambda| \|x - x_1\| = |\lambda| d$. Hence $|\lambda| = 1$. If $\lambda = 1$, then $\bar{x} = x_1$. If $\lambda = -1$, then we have $x = (x_1 + \bar{x})/2 \in C$, since C is convex. In both cases we have obtained a contradiction since $\bar{x} \neq x_1$ and $x \notin C$.

First observe that $\bar{x} \in C$ by construction. Let $c \in C$. Writing

$$\|x - c\|^2 = \|x - \bar{x} + \bar{x} - c\|^2 = \|x - \bar{x}\|^2 + \|\bar{x} - c\|^2 + 2\langle x - \bar{x},\, \bar{x} - c \rangle \geq \|x - \bar{x}\|^2,$$

and cancelling $\|x - \bar{x}\|^2$, we find $\langle x - \bar{x},\, c - \bar{x} \rangle \leq 0$. Conversely, suppose a point $\tilde{x} \in C$ satisfies (i) and (ii). Choose $c_0 \in C$ minimising the distance to x. For any $t \in (0, 1)$ and $c \in C$,

$$\|x - (tc_0 + (1 - t)c)\|^2 \;\geq\; \|x - c_0\|^2.$$

Simplifying, we have

$$(1 - t)\|x - c\|^2 + 2t\langle x - c, x - c_0 \rangle \geq (1 + t)\|x - c_0\|^2.$$

So, letting $t \to 1$, we obtain

$$\begin{aligned} 2\langle x - c, x - c_0 \rangle &\geq 2\|x - c_0\|^2 \\ \langle x - c, x - c_0 \rangle &\geq \langle x - c_0, x - c_0 \rangle \\ \langle x - c - (x - c_0), x - c_0 \rangle &\geq 0 \\ \langle c_0 - c, x - c_0 \rangle &\geq 0 \\ \langle c - c_0, x - c_0 \rangle &\leq 0. \end{aligned}$$

This completes the proof. □

Separation of Convex Sets

This part deals with some important properties of convex sets that are useful in convex optimization and duality theory.

Definition 3.1.3.16 *Let C_1 and C_2 be convex subsets of $\mathbb{R}^n$. Given $r \in \mathbb{R}$ and a non-zero $p \in \mathbb{R}^n$, the set $H_{p,r}$ is called a* separating hyperplane *if*

$$\langle p, x \rangle \leq r \leq \langle p, y \rangle \quad \textit{whenever} \quad x \in C_1 \quad \textit{and} \quad y \in C_2.$$

We say that $H_{p,r}$ strictly separates C_1 *and* C_2 *if*

$$\langle p, x\rangle < r < \langle p, y\rangle \quad \textit{whenever} \quad x \in C_1 \quad \textit{and} \quad y \in C_2.$$

Example 3.1.3.17. Consider the two convex subsets of $\mathbb{R}^2$

$$C_1 \;=\; \big\{(x_1, x_2) \in \mathbb{R}^2 \;:\; x_1 \le 0\big\}, \qquad C_2 \;=\; \big\{(x_1, x_2) \in \mathbb{R}^2 \;:\; x_1 \ge 1\big\}.$$

Geometrically, C_1 is the closed left half-plane and C_2 is the closed right half-plane lying at least one unit to the right of the y–axis. Choose $p \;=\; (1, 0)^\top \neq 0$ and $r \;=\; \dfrac{1}{2}$. The affine set

$$H_{p,r} \;=\; \big\{z \in \mathbb{R}^2 \;:\; \langle p, z\rangle = r\big\} \;=\; \big\{(x_1, x_2) \in \mathbb{R}^2 \;:\; x_1 = \tfrac{1}{2}\big\}$$

is the vertical line through $x_1 = \dfrac{1}{2}$. For any $x \in C_1$ we have $\langle p, x\rangle = x_1 < \dfrac{1}{2} = r$, while for any $y \in C_2$ we have $\langle p, y\rangle = y_1 > \dfrac{1}{2} = r$. Hence the inequalities $\langle p, x\rangle < r < \langle p, y\rangle$ hold simultaneously, so $H_{p,r}$ *strictly separates* C_1 and C_2.

If instead we take the same normal vector $p = (1, 0)^\top$ but set $r = 0$, then

$$H_{p,0} \;=\; \big\{(x_1, x_2) \in \mathbb{R}^2 \;:\; x_1 = 0\big\}$$

is the y–axis. Now $\langle p, x\rangle = x_1 \le 0 = r \le y_1 = \langle p, y\rangle$ for every $x \in C_1$ and $y \in C_2$, so $H_{p,0}$ *separates* the two sets, but not strictly, since points of C_1 may satisfy $\langle p, x\rangle = r$.

Definition 3.1.3.18 *Let $C \subset \mathbb{R}^n$ be a non-empty convex set and H be a hyperplane. We say that H is a* supporting hyperplane *if C is contained in a half-space delimited by H, i.e., there exist $r \in \mathbb{R}$ and a non-zero $p \in \mathbb{R}^n$ such that*

$$\langle p, c\rangle \le r, \quad \textit{for all } c \in C,$$

and there exists $x_0 \in C$ such that

$$\langle p, x_0\rangle = r.$$

Remark 3.1.3.19. We will also say that H *supports* C at x_0, and a half-space determined by H which contains C will be called the *supporting half-space* of C at x_0.

The following is one of the most fundamental theorems about convex sets

Theorem 3.1.3.20 *Let C and D be two convex sets in $\mathbb{R}^n$ that do not intersect (i.e., $C \cap D = \varnothing$). Then, there exists $a \in \mathbb{R}^n$, $a \neq 0$, $b \in \mathbb{R}$, such that*

$$a^T x \le b \quad \textit{for all } x \in C, \quad \textit{and} \quad a^T x \ge b \quad \textit{for all } x \in D.$$

We now introduce a version of supporting hyperplane theorem.

Theorem 3.1.3.21 *Let $C \subset \mathbb{R}^n$ be convex and let $\bar{x} \notin \operatorname{int} C$. Then there exists a non–zero vector $a \in \mathbb{R}^n$ such that*

$$\langle a, x\rangle \;\ge\; \langle a, \bar{x}\rangle \quad \textit{for all } x \in C.$$

Proof. Because $\bar{x} \notin \operatorname{int} C$ there is a radius $\varepsilon > 0$ for which the open ball

$$B_\varepsilon := \{y \in \mathbb{R}^n \mid \|y - \bar{x}\| < \varepsilon\}$$

is disjoint from C. Hence the two sets C and B_ε are non-empty, convex, and disjoint, while B_ε is open. By using theorem 3.1.3.20, there exist a vector $a \neq 0$ and real numbers α such that

$$\langle a, x\rangle \ \leq \ \alpha \quad \text{for all } x \in C, \qquad \langle a, y\rangle \ \geq \ \alpha \quad \text{for all } y \in B_\varepsilon.$$

Choose a sequence $(y_k)_k \subset B_\varepsilon$ with $y_k \to \bar{x}$. Passing to the limit in the right-hand inequality gives $\langle a, \bar{x}\rangle \ \geq \ \alpha$. So, we have

$$\langle a, x\rangle \ \leq \ \alpha \ < \ \beta \ \leq \ \langle a, \bar{x}\rangle \quad \text{for all } x \in C.$$

Finally, multiply the entire inequality by -1 and define $\tilde{a} := -a$. Because $a \neq 0$, we also have $\tilde{a} \neq 0$, and

$$\langle \tilde{a}, x\rangle \ = \ -\langle a, x\rangle \ \geq \ -\langle a, \bar{x}\rangle \ = \ \langle \tilde{a}, \bar{x}\rangle \quad \text{for all } x \in C.$$

This is precisely the required supporting-hyperplane inequality with the normal vector $\tilde{a}$. □

Convex Functions

Definition 3.1.3.22 *Let C be a convex subset of $\mathbb{R}^d$ and $f : C \to \mathbb{R}$ be a function. We say that f is* convex *if for every $(x, y) \in C^2$ and $\lambda \in [0, 1]$,*

$$f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y).$$

Moreover, we also have a more strict definition of convex function.

Definition 3.1.3.23 *Let C be a convex subset of $\mathbb{R}^d$ and $f : C \to \mathbb{R}$ be a function. We say that f is* strictly convex *if for every $(x, y) \in C^2,\ x \neq y$ and $\lambda \in (0, 1)$,*

$$f(\lambda x + (1 - \lambda)y) < \lambda f(x) + (1 - \lambda)f(y).$$

These definitions of convex functions can be used to define concave functions.

Definition 3.1.3.24 (Concave and strictly concave function) *Let C be a convex subset of $\mathbb{R}^d$ and $f : C \to \mathbb{R}$ be a function. We say that f is* concave *if $-f$ is convex. The definition of* strictly concave *function follows as derivation.*

Example 3.1.3.25. We consider the two functions $f_1, f_2 : \mathbb{R} \to \mathbb{R}$ defined by

$$f_1(x) = x^2$$

and

$$f_2(x) = x.$$

Let x and y be two numbers. Let $\lambda_1 \in [0, 1]$ be a number. In case of f_1, we have that

$$\begin{aligned} f_1(\lambda x + (1 - \lambda)y) - \lambda f_1(x) - (1 - \lambda)f_1(y) &= \lambda^2 x^2 + 2\lambda(1 - \lambda)xy + (1 - \lambda)^2 y^2 - \lambda x^2 - (1 - \lambda)y^2 \\ &= -\lambda(1 - \lambda)(x^2 - 2xy + y^2) \\ &= -\lambda(1 - \lambda)(x - y)^2 \\ &\leq 0. \end{aligned}$$

That means f_1 is a convex function. In fact, for every two numbers $x \neq y$ and $\lambda \in (0, 1)$, we know that

$$-\lambda(1 - \lambda)(x - y)^2 < 0$$

which implies that f_1 is strictly convex. In case of f_2, we have that

$$\begin{aligned} f_2(\lambda x + (1-\lambda)y) &= \lambda x + (1-\lambda)y \\ &= \lambda f_2(x) + (1-\lambda) f_2(y). \end{aligned}$$

Therefore, we conclude that f_2 is convex but not strictly convex. There are also lots of other convex functions such as $f(x) = e^x, f(x) = |x|, f(x) = 1/x$ and so on. For examples of concave functions, using those two above functions and proofs, we obtain that $-f_1$ is a strictly concave function and $-f_2$ is a concave function.

Beside the definitions, we have more tools to determine the convexity of a function. Dealing with the the Hessian matrix of a function, one of the most important criteria for convexity is as following.

Theorem 3.1.3.26 *Let C be an open and convex subset of $\mathbb{R}^d$ and $f : C \to \mathbb{R}$ be a twice continuously differentiable function. The function f is convex if and only if for every $x \in C$,*

$$\nabla^2 f(x) \succeq 0 \quad \textit{(in the sense of symmetric matrices)}.$$

Moreover, if we also have

$$\nabla^2 f(x) \succ 0 \quad \textit{(in the sense of symmetric matrices)},$$

for every $x \in C$, then the function f is strictly convex on C.

In the Theorem 3.1.3.26, the term $\nabla^2 f(x) \succeq 0$ means that $\nabla^2 f(x)$ is semi-positive definite. Also, the term $\nabla^2 f(x) \succ 0$ means that $\nabla^2 f(x)$ is positive-definite.

Example 3.1.3.27. Referring to the Example 3.1.3.25, we consider the function f_1 defined by $f_1(x) = x^2$. Let $x \in \mathbb{R}$ be a number. Note that $\mathbb{R}$ is a open set. We have that

$$f_1''(x) = 2 > 0.$$

Using Theorem 3.1.3.26, we conclude that f_1 is strictly convex.

This part of convexity will play crucial roles in the following part in which we introduce and discuss about optimization.

3.1.4 Optimization Theory

Optimization theory serves as a foundational framework in numerous fields, including economics, finance, engineering, and the physical sciences. Its primary role is to mathematically characterize and solve problems where a quantitative measure - known as an *objective function* - must be optimized, such as maximizing profit or minimizing cost. This objective function typically depends on various factors termed *decision variables*. The core aim is to determine the values of these variables that yield an optimal value (either maximum or minimum) for the specified objective. Usually, solutions must also satisfy predefined *constraints* or limitations that reflect real-world or theoretical restrictions. After formulating a practical or theoretical optimization problem mathematically, the resulting abstract representation is referred to as a *mathematical model*. Since exact analytical solutions are often unattainable for complex problems, optimization algorithms are designed to approximate solutions as accurately as possible. Finally, a set of *optimality conditions* - mathematical criteria for optimality - are typically employed to validate that the algorithm-generated solutions effectively solve the posed optimization problem. In this part, we establish fundamental concepts from convexity and optimization theory that will be essential for our latter works.

Basic Concepts in Optimization

In this part, we focus on optimization theory as well as KKT theorem used in optimization. To begin, we discuss about the infimum of a function. Let $f : \mathbb{R}^d \to \mathbb{R}$ be a function. Let K be a non-empty subset of $\mathbb{R}^d$. We have the following definition.

Definition 3.1.4.1 (Infimum of a function) *The* infimum *of f on K is the value $\ell \in [-\infty, +\infty)$ such that*

1. *$f(x) \geq \ell$ for every $x \in K$.*
2. *There exists a sequence $(x_n)_{n\in\mathbb{N}}$ of elements from K ($x_n \in K$ for all $n \in \mathbb{N}$) such that*

$$\lim_{n\to+\infty} f(x_n) = \ell.$$

This value ℓ is denoted as $\inf_{x\in K} f(x)$ which is

$$\ell = \inf_{x\in K} f(x).$$

Remark 3.1.4.2. Unlike minimum, the infimum always exists. Recalling about the minimum, it exists if and only if there exists a x_0 such that

$$f(x_0) = \inf_{x\in K} f(x) = \min_{x\in K} f(x).$$

Back to infimum, it is finite (i.e., $\ell \neq -\infty$) if and only if the function f is bounded on K, meaning that there exists a constant $M \in \mathbb{R}$ such that

$$\forall x \in K,\ f(x) \geq M.$$

Otherwise, if f is not bounded, the infimum of f is $-\infty$.

Definition 3.1.4.3 *The* minimum *of a function f on a set K is defined as the value $l \in]-\infty, +\infty[$, provided it exists, for which there is an element $\bar{x} \in K$ satisfying:*

1. *$\forall x \in K, \quad f(x) \geq l$.*
2. *$f(\bar{x}) = l$.*

This value is denoted by

$$\min_{x\in K} f(x).$$

In this case, we say that the function f achieves its minimum at the point $\bar{x}$, or equivalently, the minimization problem $\min_{x\in K} f(x)$ admits a solution $\bar{x}$.

Remark 3.1.4.4. Due to common language usage, an element $\bar{x} \in K$ satisfying the above properties is often also called a minimum (though strictly speaking, $\bar{x}$ should be termed the "argument of the minimum").

Remark 3.1.4.5. Unlike the infimum, a minimum may not always exist. Sufficient conditions guaranteeing its existence will be discussed subsequently.

Example 3.1.4.6. Determine the infimum- and potentially the minimum if it exists- of the following function

$$f(x, y) = x^2 + y^2 \text{ for } K = \mathbb{R}^2 \setminus \{(0, 0)\}.$$

We see that $f(x,y) > 0\ \forall (x,y) \in \mathbb{R}^2 \setminus \{(0,0)\}$. Choose $x_n = \dfrac{1}{n}, y_n = \dfrac{1}{n}$, we have

$$\lim_{n\to+\infty} f(x_n, y_n) = \lim_{n\to+\infty} \left(\frac{1}{n^2} + \frac{1}{n^2}\right) = 0.$$

Hence, by definition $\inf\limits_{\mathbb{R}^2\setminus\{(0,0)\}} f(x,y) = 0$. Now assume $f(x,y)$ has the minimum at x_0, y_0. By choosing $x = x_0, y = \dfrac{y_0}{2}$, we will have

$$f\left(x_0, \frac{y_0}{2}\right) = x_0^2 + \frac{y_0^2}{4} < x_0^2 + y_0^2 = f(x_0, y_0),$$

which is a contradiction. Hence f does not have minimal value.

Necessary Optimality Condition in an Open Set

Definition 3.1.4.7 *A function $f(x)$ is called a $\mathcal{C}^k$* function *if f has k continuous derivatives. The most common $\mathcal{C}^k$ space is $\mathcal{C}^0$, the space of* continuous functions*, whereas $\mathcal{C}^1$ is the space of* continuously differentiable functions*.*

Example 3.1.4.8. We can see that $|x|^{k+1}$ (for k even) and $x^{k+1}\sin\left(\dfrac{1}{x}\right)$ are $\mathcal{C}^k$ functions, and they do not have $(k+1)$-th derivative at 0.

Theorem 3.1.4.9 *If K is an open subset of $\mathbb{R}^n$ and f is a C^1 function from $\mathbb{R}^n$ to $\mathbb{R}$, and if x^* is a minimum of the problem $\mathcal{P}$, then*

$$\nabla f(x^*) = 0.$$

Remark 3.1.4.10. (i) Recall that there is a second-order condition for C^2 functions: the symmetric matrix $\nabla^2 f(x^*) := \mathrm{Hess}_f(x^*)$ is positive definite, meaning its eigenvalues are all positive or zero.

(ii) Since the proof of this result serves as a prototype for proofs in optimization, it is essential to understand it thoroughly.

Proof. Let v be an arbitrary vector in $\mathbb{R}^n$. As x^* belongs to K, which is open, there exists $h_0 > 0$ such that, for all $h \in [0, h_0]$, the point $x^* + hv$ belongs to K. Since x^* is a minimum of the problem, we have

$$f(x^* + hv) - f(x^*) \geq 0.$$

Since

$$f(x^* + hv) - f(x^*) = h\langle \nabla f(x^*), v\rangle + h\varepsilon(hv),$$

dividing the above inequality by $h > 0$ and taking the limit as h approaches 0, we obtain

$$\langle \nabla f(x^*), v\rangle \geq 0.$$

Since this inequality holds for all $v \in \mathbb{R}^n$, it also holds for $-v$. Thus,

$$\langle \nabla f(x^*), -v\rangle \geq 0,$$

which implies

$$\langle \nabla f(x^*), v\rangle = 0.$$

Therefore, finally,

$$\forall v \in \mathbb{R}^n, \quad \langle \nabla f(x^*), v\rangle = 0,$$

i.e., $\nabla f(x^*) = 0$. □

Example 3.1.4.11. Let $K = \mathbb{R}^n$ (an open set) and define

$$f(x) = \|x\|^2 = \sum_{i=1}^{n} x_i^2.$$

The function f is C^∞ on K. The unique minimizer of f is $x^\star = 0$, and

$$\nabla f(x) = 2x,$$

which means $\nabla f(x^\star) = 0$, so Euler's first-order necessary condition is satisfied.

Example 3.1.4.12. Take $K = [0, \infty) \subset \mathbb{R}$ *(not open)* and let $f(x) = x$. The minimizer is $x^\star = 0$, but

$$\nabla f(0) = 1 \neq 0,$$

because the minimizer lies on the boundary of a non-open set.

Example 3.1.4.13. Let $K = \mathbb{R}$ and $f(x) = |x|$. Although K is open and the minimizer is $x^\star = 0$, the gradient $\nabla f(x)$ does not exist at $x = 0$, so Euler's condition is inapplicable.

Example 3.1.4.14. Consider the constrained problem

$$\min\{-\|x\|^2 : \|x\| \leq 1\}, \qquad x \in \mathbb{R}^2.$$

The feasible set $\{x \in \mathbb{R}^2 : \|x\| < 1\}$ is open, but every maximiser (the minima of $-\|x\|^2$) occurs on the boundary $\|x\| = 1$. For any such $x^\star$ we have

$$\nabla(-\|x\|^2)\big|_{x=x^\star} = -2x^\star \neq 0.$$

Hence, the interior Euler conditions fail.

Intuitively, the boundary prevents us from moving in all directions; only directions tangent to the feasible set are admissible. Therefore we need optimality conditions that incorporate both the *objective and the active constraints*. This leads us to the *Karush–Kuhn–Tucker (KKT) conditions* in the next part, which generalize Euler's rule to constrained problems.

The Karush-Kuhn-Tucker (KKT) Theorem with Generalized Lagrangian

In the general framework of the Karush-Kuhn-Tucker theorem, the constraint set K has the form

$$K = \{x \in \mathbb{R}^n \mid g_i(x) \leq 0,\ i \in I,\ h_j(x) = 0,\ j \in J\}$$

where $I = \{1, \ldots, l\}$ indexes the inequality constraints and $J = \{1, \ldots, m\}$ indexes the equality constraints. The functions g_i and h_j are all assumed to be $\mathcal{C}^1$ functions from $\mathbb{R}^n$ to $\mathbb{R}$.

For any $x \in K$, the indices $i \in \{1, \ldots, l\}$ such that $g_i(x) = 0$ are called saturated constraints

$$I(x) = \{i \in \{1, \ldots, l\} \mid g_i(x) = 0\}$$

Theorem 3.1.4.15 *If a point x^* is a minimum of the problem $\mathcal{P}$, then there exist $p_0 \in \mathbb{R}_+$, $p \in \mathbb{R}_+^l$, and $q \in \mathbb{R}^m$ with*

$$\begin{cases} (i) & \displaystyle\sum_i p_i g_i(x^*) = 0 \quad \textit{(exclusion condition)} \\ (ii) & (p_0, p, q) \neq 0 \\ (iii) & \displaystyle p_0 \nabla f(x^*) + \sum_i p_i \nabla g_i(x^*) + \sum_j q_j \nabla h_j(x^*) = 0 \quad \textit{(necessary condition)} \end{cases}$$

Remark 3.1.4.16. (i) The vector (p_0, p, q) is called the *generalized multiplier* associated with the solution x^* ("generalized" because, as we will see later, p_0 can often be taken as 1).

(ii) The exclusion condition means that if $i \notin I(x^*)$, then $p_i = 0$.

(iii) It is possible that $p_0 = 0$ in the above expression. However, this case is somewhat "pathological," in the sense that it corresponds to a less "regular" constraint. The true Karush-Kuhn-Tucker theorem asserts that if the constraint is "qualified," then $p_0 = 1$ can be taken (see theorem 2.2 below).

(iv) The important part of the theorem is, of course, the necessary condition. We will later see a geometric interpretation of this condition.

(v) The function

$$L(x, p_0, p, q) = p_0 f(x) + \sum_i p_i g_i(x) + \sum_j q_j h_j(x)$$

is called the *generalized Lagrangian*. The necessary optimality condition can also be written as

$$\frac{\partial L}{\partial x}(x^*, p_0, p, q) = 0.$$

We will prove the theorem by penalization. Consider the following penalization

$$f_N(x) = f(x) + \|x - x^*\|^2 + \frac{N}{2}\left[\sum_{i\in I} \max(0, g_i(x))^2 + \sum_{j\in J} h_j(x)^2\right]$$

Note that f_N is a $\mathcal{C}^1$ function, and $f_N(x^*) = f(x^*)$. Now we show that there exists $\varepsilon_0 > 0$ such that for any $\varepsilon \in (0, \varepsilon_0]$, there exists $N_\varepsilon > 0$ such that for any x with $\|x - x^*\| = \varepsilon$, we have $f_N(x) > f_N(x^*)$. Since x^* is a local minimum of f over the constraint K, there exists $\varepsilon_0 > 0$ such that for all $x \in K$ and $\|x - x^*\| \leq \varepsilon_0$, we have $f(x) \geq f(x^*)$.

Fix an arbitrary $\varepsilon \in (0, \varepsilon_0]$. We now reason by contradiction. Suppose that for all N, there exists x_N such that $\|x_N - x^*\| = \varepsilon$ and $f_N(x_N) \leq f(x^*)$. Since x_N is a bounded sequence, x_N converges, up to a subsequence (still denoted (x_n)), to a point $\bar{x}$. Let's show that $\bar{x}$ belongs to K, satisfies $\|\bar{x} - x^*\| = \varepsilon$, and $f(\bar{x}) \leq f(x^*)$. This will lead to a contradiction.

The fact that $\|\bar{x} - x^*\| = \varepsilon$ is clear because $\|x_N - x^*\| = \varepsilon$. Moreover, we have

$$f(x_N) + \|x_N - x^*\|^2 + \frac{N}{2}\left(\sum_{i\in I} \left(\max(0, g_i(x_N))\right)^2 + \sum_{j\in J} \left(h_j(x_N)\right)^2\right) \leq f(x^*) \qquad (3.1.3)$$

Thus,

$$0 \leq \left(\sum_{i\in I} \left(\max(0, g_i(x_N))\right)^2 + \sum_{j\in J} \left(h_j(x_N)\right)^2\right) \leq \frac{2}{N}\left(f(x^*) - f(x_N) - \varepsilon^2\right)$$

As $N \to +\infty$, we obtain

$$0 \leq \left(\sum_{i\in I} \left(\max(0, g_i(\bar{x}))\right)^2 + \sum_{j\in J} \left(h_j(\bar{x})\right)^2\right) \leq 0$$

meaning that $\bar{x}$ belongs to K. Finally, from inequality (2.1) and the fact that $\|x_N - x^*\|^2 = \varepsilon^2$, we have

$$f(x_N) + \varepsilon^2 \leq f(x^*)$$

Thus, by taking the limit, we obtain $f(\bar{x}) \leq f(x^*) - \varepsilon^2 < f(x^*)$. However, by the definition of x^*, it's impossible to have $\bar{x} \in K$, $\|\bar{x} - x^*\| \leq \varepsilon_0$, and $f(\bar{x}) \leq f(x^*)$ simultaneously. This leads to a contradiction with the hypothesis. Thus, we conclude that for any $\varepsilon \in]0, \varepsilon_0]$, there exists $N_\varepsilon > 0$ such that for any x with $\|x - x^*\| = \varepsilon$, we have $f_N(x) > f_N(x^*)$.

Now fix $\varepsilon \in (0, \varepsilon_0)$. Since for any x with $\|x - x^*\| = \varepsilon$, we have $f_N(x) > f_N(x^*)$, the function f_N has a local minimum in the open set $\{x,\ \|x - x^*\| < \varepsilon\}$ at a point denoted x_ε (cf. Proposition ??). At this point, the necessary optimality condition for an open set applies (cf. Theorem 1.1), and we obtain

$$\nabla f_N(x_\varepsilon) = 0,$$

meaning

$$\nabla f(x_\varepsilon) + 2(x_\varepsilon - x^*) + N \left[\sum_{i \in I} \max(0, g_i(x_\varepsilon)) \nabla g_i(x_\varepsilon) + \sum_{j \in J} h_j(x_\varepsilon) \nabla h_j(x_\varepsilon) \right] = 0$$

Let's define

$$\rho^\varepsilon = \left(1 + N^2 \sum_{i \in I} \max(0, g_i(x_\varepsilon))^2 + N^2 \sum_{j \in J} h_j(x_\varepsilon)^2 \right)^{\frac{1}{2}}$$

and

$$p_0^\varepsilon = \frac{1}{\rho^\varepsilon}, \quad p_i^\varepsilon = N p_0^\varepsilon \max(0, g_i(x_\varepsilon)), \quad q_j^\varepsilon = N p_0^\varepsilon h_j(x_\varepsilon)$$

As ε tends to 0, x_ε tends to x^*, and the vector $(p_0^\varepsilon, p^\varepsilon, q^\varepsilon)$ that has a norm of 1 in $\mathbb{R}^{1+l+m}$ converges, up to a subsequence, to a vector of norm 1, denoted (p_0, p, q). By dividing the necessary condition obtained above by ρ^ε and taking the limit as ε tends to 0, we eventually get the equality:

$$p_0 \nabla f(x^*) + \left[\sum_{i \in I} p_i \nabla g_i(x^*) + \sum_{j \in J} q_j \nabla h_j(x^*) \right] = 0$$

Finally, note that the exclusion condition is satisfied. If $g_i(x) < 0$ for some i, then we have $g_i(x) < 0$ for sufficiently small .Thus,$p_i = 0$ by definition. Taking the limit gives us $p_i = 0$.

Example 3.1.4.17. Let's now consider the problem

$$\min_{\substack{x^2+y^2=1 \\ y^2+z^2=4}} x + z$$

The constraint $K = \{(x, y) \mid x^2 + y^2 = 1,\ y^2 + z^2 = 4\}$ is compact since it is closed and contained within

$$[-1, 1] \times [-1, 1] \times [-2, 2].$$

The function $f(x, y, z) = x + z$ is continuous, so the problem has a solution (x^*, y^*, z^*). Let $h_1(x, y) = x^2 + y^2 - 1$ and $h_2(x, y, z) = y^2 + z^2 - 4$.

Let's find the points where the constraint is qualified. These are the points $(x, y, z) \in K$ such that the family $\{\nabla h_1(x, y, z), \nabla h_2(x, y, z)\}$ is linearly independent. Now,

$$\nabla h_1(x, y, z) = \begin{pmatrix} 2x \\ 2y \\ 0 \end{pmatrix}, \quad \nabla h_2(x, y, z) = \begin{pmatrix} 0 \\ 2y \\ 2z \end{pmatrix}.$$

These two vectors are linearly dependent if and only if $x = z = 0$. However, this case is not possible since it would imply $y^2 = 1$ and $y^2 = 4$. Hence, the system of vectors is always linearly independent, and the constraint is qualified.

Now let's consider $(x^*, y^*, z^*) \in K$ as a minimum of the problem. The necessary optimality conditions can be written as follows: there exist μ_1 and μ_2 such that

$$\begin{cases} 1 + 2\mu_1 x^* = 0, \\ 2\mu_1 y^* + 2\mu_2 y^* = 0, \\ 1 + 2\mu_2 z^* = 0, \\ (x^*)^2 + (y^*)^2 = 1, \\ (y^*)^2 + (z^*)^2 = 4. \end{cases}$$

Note that μ_1 and μ_2 are non-zero, and

$$x^* = -\frac{1}{2\mu_1}, \quad z^* = -\frac{1}{2\mu_2}.$$

Furthermore, either $\mu_1 + \mu_2 = 0$ or $y^* = 0$. The first case is impossible since it would lead to $x^* = -z^*$ and

$$(y^*)^2 = 1 - (x^*)^2 = 1 - (z^*)^2 = 4 - (z^*)^2,$$

which results in a contradiction.

Thus, $y^* = 0$, implying $x^* = \pm 1$ and $z^* = \pm 2$. It's easy to see that the minimum of the problem is $(-1, 0, -2)$.

General Optimization Problem and Its Primal-Dual Form

In this part, we explore optimization theory by examining functions $f : \mathbb{R}^n \to \mathbb{R}$, $g : \mathbb{R}^n \to \mathbb{R}^m$, $h : \mathbb{R}^n \to \mathbb{R}^p$, and a subset $X \subseteq \mathbb{R}^n$. The focus is on a broad class of optimization problems, often referred to as the *primal problem*.

Optimization plays a fundamental role across numerous scientific and engineering disciplines. It entails identifying the most optimal solution from a range of possibilities based on specified criteria. Mathematically, this involves minimizing or maximizing an objective function while satisfying a set of constraints.

The *primal problem* lies at the heart of constrained optimization and can be written as

$$\begin{aligned} &\text{minimize} && f(x) && (3.1.4)\\ &\text{subject to} && g(x) = 0, && (3.1.5)\\ &&& h(x) \leq 0, && (3.1.6)\\ &&& x \in X, && (3.1.7) \end{aligned}$$

where $f\colon X \to \mathbb{R}$ is the function to be minimized, $g\colon X \to \mathbb{R}^m$ enforces equality constraints, $h\colon X \to \mathbb{R}^p$ imposes inequality constraints, and the set X specifies the permissible values of the decision variable x.

A fundamental tool for handling constraints in optimization is the *Lagrangian*, which merges the objective and the constraints into a single scalar function

$$L(x, y, z) = f(x) + y^\top g(x) + z^\top h(x),$$

where $y \in \mathbb{R}^m$ and $z \in \mathbb{R}^p$ are the Lagrange multipliers (often called "dual variables"). By introducing these multipliers, one can transform the original constrained problem into an unconstrained saddle-point problem, thus linking its primal and dual formulations.

We study the fundamental properties of the Lagrangian. Comprehending the behavior of the *Lagrangian* is key to analyzing both the primal and dual formulations of an optimization problem. A particularly useful feature is that the Lagrangian always supplies a *lower bound* on the objective value at feasible points.

Lemma 3.1.4.18 *For every feasible point x and for all multipliers $(y, z) \in \mathbb{R}^m \times \mathbb{R}^p_+$ we have*

$$L(x, y, z) \;\le\; f(x).$$

Proof. Feasibility gives $g(x) = 0$ and $h(x) \le 0$. Because the multipliers attached to the inequality constraints are restricted to be non-negative ($z \ge 0$), we obtain

$$L(x, y, z) = f(x) + y^\top g(x) + z^\top h(x) \;\le\; f(x),$$

since $y^\top g(x) = 0$ and $z^\top h(x) \le 0$. Hence, inside the feasible set the Lagrangian never exceeds the objective value.

Let $f^\star$ denote the optimal value of the primal problem. Taking the infimum of $L(x, y, z)$ over feasible x and the supremum over multipliers yields

$$\inf_{x \in X} L(x, y, z) \;\le\; f^\star,$$

and consequently,

$$\sup_{(y,z) \in \mathbb{R}^m \times \mathbb{R}^p_+} \; \inf_{x \in X} L(x, y, z) \;\le\; f^\star.$$

This inequality is the cornerstone of Lagrange duality. □

One insightful perspective on the primal optimization problem is to reinterpret it as a min–max formulation, a viewpoint that helps illuminate the problem's inherent dual structure. Consider the following expressions involving the supremum

$$\sup_{y \in \mathbb{R}^m} y^\top g(x) = \begin{cases} 0, & \text{if } g(x) = 0, \\ \infty, & \text{otherwise.} \end{cases}$$

and

$$\sup_{z \in \mathbb{R}^p_+} z^\top h(x) = \begin{cases} 0, & \text{if } h(x) \le 0, \\ \infty, & \text{otherwise.} \end{cases}$$

These formulas capture the core role of the constraint functions g and h, they restate the equality and inequality conditions in a form that fits seamlessly into the Lagrangian framework.

Building on the preceding results, we can recast the primal program in a *min–max* setting

$$\sup_{(y,z) \in \mathbb{R}^m \times \mathbb{R}^p_+} L(x, y, z) = \begin{cases} f(x), & \text{when } g(x) = 0 \text{ and } h(x) \le 0, \\ \infty, & \text{otherwise.} \end{cases}$$

Consequently, the primal problem (3.1.5) can be rewritten as

$$\inf_{x\in X} \sup_{(y,z)\in\mathbb{R}^m\times\mathbb{R}^p_+} L(x,y,z).$$

This re-expression acts as the crucial link between the primal and dual viewpoints in optimization. Duality theory shows that the dual problem is simply the primal problem examined from a different angle. Formally, the dual problem of (3.1.5) is given by

$$\sup_{(y,z)\in\mathbb{R}^m\times\mathbb{R}^p_+} \inf_{x\in X} L(x,y,z).$$

From Lemma 3.1.4.18 and the formulation of the primal problem as a min-max problem, we derive an important result known as a weak duality theorem.

Theorem 3.1.4.19

$$\sup_{(y,z)\in\mathbb{R}^m\times\mathbb{R}^p_+} \inf_{x\in X} L(x,y,z) \leq \inf_{x\in X} \sup_{(y,z)\in\mathbb{R}^m\times\mathbb{R}^p_+} L(x,y,z).$$

We know ignore the proof at this part and return to this later. This theorem lays the foundation for the relationship between the primal and dual problems,highlighting the inherent connection and interdependence between them.

Abstract Duality in Optimization

In the field of optimization, duality is not confined to concrete mathematical formulations but can be generalized to more abstract scenarios. Suppose we have two arbitrary sets X and Y, and a function $L : X \times Y \to \mathbb{R}$. Within this general framework, we can formulate what are referred to as the primal and dual problems. We give again the formulation of primal and dual problems as follows

$$(P) \quad : \quad \inf_{x\in X} \sup_{y\in Y} L(x,y), \tag{3.1.8}$$

$$(D) \quad : \quad \sup_{y\in Y} \inf_{x\in X} L(x,y), \tag{3.1.9}$$

Here, (P) represents the primal problem and (D) the dual problem. These problems encapsulate the min-max and max-min relationships to many optimization scenarios. A fundamental result in the theory of duality is weak duality, which can be expressed as

Theorem 3.1.4.20

$$\sup_{y\in Y} \inf_{x\in X} L(x,y) \leq \inf_{x\in X} \sup_{y\in Y} L(x,y).$$

Proof. The proof of this theorem is straight forward yet insightful. For all $(x',y') \in X \times Y$, it holds that

$$\inf_{x\in X} L(x,y') \leq L(x',y') \leq \sup_{y\in Y} L(x',y).$$

Then, for every $x' \in X$

$$\sup_{y'\in Y} \inf_{x\in X} L(x,y') \leq \sup_{y\in Y} L(x',y).$$

Taking the infimum over X on the right completes the proof. □

Saddle Point

A saddle point is a pivotal concept in optimization, particularly in context of Lagrangian function.

Definition 3.1.4.21 *A pair $(x^*, y^*) \in X \times Y$ is defined as a saddle point of L if, for all $(x, y) \in X \times Y$*

$$L(x^*, y) \leq L(x^*, y^*) \leq L(x, y^*).$$

The characterization of a saddle point is given by a theorem.

Theorem 3.1.4.22 *The characterization of a saddle point is given by*

$$\sup_{y \in Y} \inf_{x \in X} L(x, y) = L(x^*, y^*) = \inf_{x \in X} \sup_{y \in Y} L(x, y).$$

Furthermore, x^* is optimal solution of (P), y^* is an optimal solution of (D), and their optimal values are equal.

Proof. Suppose (x^*, y^*) is a saddle point. By definition

$$\sup_{y \in Y} L(x^*, y) = L(x^*, y^*) = \inf_{x \in X} L(x, y^*).$$

This implies

$$\inf_{x \in X} \sup_{y \in Y} L(x, y) \leq L(x^*, y^*) \leq \sup_{y \in Y} \inf_{x \in X} L(x, y).$$

By weak duality, the reverse is also true, thus establishing equality. Conversely, if the equation holds, then

$$\inf_{x \in X} L(x, y^*) = L(x^*, y^*) = \sup_{y \in Y} L(x^*, y).$$

Thus, for all $(x, y) \in X \times Y$

$$L(x, y^*) \leq L(x^*, y^*) \quad \text{and} \quad L(x^*, y) \geq L(x^*, y^*).$$

Hence, the conditions for a saddle point are met. □

The concept of a saddle point plays a pivotal role in optimization theory, particularly in establishing the link between primal and dual formulations. In the context of optimal transport, the existence of a saddle point ensures not only the optimality of solutions for both the primal and dual problems but also the equality of their objective values—a fundamental aspect of duality theory.

Recognizing and analyzing saddle points allows for a deeper understanding of optimality conditions and facilitates the formulation of efficient solution methods. This concept is instrumental across various domains, such as economics, game theory, and mathematical programming, where optimal transport problems often arise in modeling and solving complex resource allocation challenges.

Lagrangian Dual Function and Duality in Optimization

In this section, our discussion revolves around the concept of the Lagrangian dual function, a fundamental construct in optimization theory. The Lagrangian dual function is particularly valuable as it illuminates the intricate relationship between primal and dual optimization problems, providing deeper insights into their duality structure. By utilizing this dual function, we can

systematically examine and interpret the interplay of primal constraints and their corresponding dual variables, facilitating a robust framework for analyzing and solving complex optimization scenarios.

We restate again and denote the formulation

$$\min(P) = \min_{x \in X} \sup_{(y,z) \in \mathbb{R}^m \times \mathbb{R}^p_+} L(x, y, z) \tag{3.1.10}$$

and

$$\max(D) = \max_{(y,z) \in \mathbb{R}^m \times \mathbb{R}^p_+} \inf_{x \in X} L(x, y, z). \tag{3.1.11}$$

Theorem 3.1.4.23 *An important property of the dual function is its concavity over $\mathbb{R}^m \times \mathbb{R}^p$, which leads us to a fundamental result*

$$\max(D) \leq \min(P).$$

Proof. Recall the convexity principle, if $\{f_j\}_{j \in J}$ is a family of convex functions indexed by J and defined on a convex set C, the function

$$x \in C \mapsto \sup_{j \in J} f_j(x)$$

is convex on C. For $x \in \mathbb{R}^n$, the function

$$(y, z) \mapsto L(x, y, z) = f(x) + y^\top g(x) + z^\top h(x)$$

is affine. Therefore, $-d(y, z) = \sup_{x \in X} -L(x, y, z)$ is convex as the supremum of convex functions. This leads to the conclusion that $\max(D) \leq \min(P)$. □

The *duality gap*, $\min(P) - \max(D)$, is a measure of the difference between the primal and dual solution.

We then add two more properties for saddle point characterization in theorem 3.1.4.22.

Theorem 3.1.4.24 *A triplet $(x^*, y^*, z^*) \in X \times \mathbb{R}^m \times \mathbb{R}^p_+$ is the saddle point of Lagrangian for problem* (3.1.10) *if and only if*

1. *x^* is the global minimum of* (3.1.10).
2. *(y^*, z^*) is the global maximum of* (3.1.11).
3. *The duality gap is zero,* $\min(P) - \max(D) = 0$.

Furthermore, if (x^*, y^*, z^*) is a saddle point, then

$$\max(D) = L(x^*, y^*, z^*) = \min(P),$$

and the complementarity conditions are satisfied

$$0 \leq z^* \perp h(x^*) \leq 0.$$

To prove complementarity conditions, we note that $L(x^*, y^*, z^*) = \min(P)$, we have

$$f(x^*) = f(x^*) + y^{*\top} g(x^*) + z^{*\top} h(x^*).$$

With $g(x^*) = 0$, it follows that $z^{*\top} h(x^*) = 0$.

Remark 3.1.4.25. This theorem serves as an essential bridge connecting the primal and dual formulations in optimization, thus representing one of the foundational pillars within optimization theory. It is important to emphasize that these theoretical constructs have been developed without imposing any particular structural constraints on the set $X \subset \mathbb{R}^n$. Furthermore, when we specifically consider the scenario where the feasible set is the entirety of the Euclidean space, $X = \mathbb{R}^n$, and the involved functions f, g, and h are assumed to be differentiable across the domain $\mathbb{R}^n$, the concept of a saddle point emerges clearly.

In this context, the detailed characterization of a saddle point not only enhances our understanding of optimality conditions but also provides deeper insights into the inherent interplay between the primal-dual structures of optimization problems.

Corollary 3.1.4.26 *If (x^*, y^*, z^*) is a saddle point, then the gradient of Lagrangian at this point vanishes*

$$\nabla_x L(x^*, y^*, z^*) = \nabla f(x^*) + \nabla g(x^*) y^* + \nabla h(x^*) z^* = 0.$$

Proof. Given that (x^*, y^*, z^*) is a saddle point, x^* represent a minimum of the function $x \mapsto L(x^*, y^*, z^*)$ over $\mathbb{R}^n$. Consequently, the necessary condition for the minimum, the vanishing gradient, leads to

$$\nabla_x L(x^*, y^*, z^*) = \nabla f(x^*) + \nabla g(x^*) y^* + \nabla h(x^*) z^* = 0.$$

□

Remark 3.1.4.27. This corollary, in conjunction with theorem 3.1.4.24, reveals that for differentiable functions defining the problem, a saddle point of the Lagrangian necessitates the satisfaction of the KKT conditions.

Remark 3.1.4.28. It is crucial to recognize that while the KKT conditions are necessary for optimality under certain conditions, they are not always sufficient to ensure that the corresponding primal-dual solution forms a saddle point.

Example 3.1.4.29. Consider the following optimization problem

$$\text{minimize } -x^2, \text{ subject to } 0 \leq x \leq 1.$$

For this problem, the Lagrangian is

$$L(x, z) = -x^2 - z_1 x + z_2(x - 1)$$

and the gradient of the Lagrangian with respect to x is

$$\nabla_x L(x, z) = -2x - z_1 + z_2.$$

The solution to this problem are found to be

$$x^* = 1 \text{ and } z^* = (0, 2).$$

However, for the dual function

$$d(z) = \inf_{x \in \mathbb{R}} L(x, z) = -\infty.$$

This implies that the maximum value of the dual problem, $\max(D)$, is $-\infty$, while the minimum value of the primal problem, $\min(P)$, is -1.

KKT Conditions for Convex Problem

In this part, we examine the Karush-Kuhn-Tucker (KKT) conditions in the context of a convex optimization problem. Consider the convex problem

$$\begin{aligned} &\text{Minimize} && f(x) \\ &\text{Subject to} && Ax = b, \\ & && h(x) \leq 0, \\ & && x \in C, \end{aligned}$$

where $C \subset \mathbb{R}^n$ is a convex set, $f, h_1, \ldots, h_p$ are convex functions on C, $A \in \mathbb{R}^{m\times n}$, and $b \in \mathbb{R}^m$.

If $C = \mathbb{R}^n$ and the KKT conditions are satisfied at a minimum x^* with dual variables (y^*, z^*), we have

$$\begin{aligned} \nabla_x L(x^*, y^*, z^*) = \nabla f(x^*) + A^\top y^* + \nabla h(x^*) z^* = 0, \\ Ax^* = b, \\ 0 \leq z^* \perp h(x^*) \leq 0. \end{aligned}$$

The feasibility and complementarity conditions imply that

$$\min(P) = f(x^*) = L(x^*, y^*, z^*).$$

The first condition, along with the convexity of the problem and weak duality, leads to

$$\max(D) = f(x^*) = d(y^*, z^*) = \inf_{x\in\mathbb{R}^n} L(x, y^*, z^*).$$

Therefore, if the KKT conditions are satisfied at (x^*, y^*, z^*), this point is a saddle point of the Lagrangian.

Next, we study about one of the common constraint qualifications, which is Slater's conditions. In convex optimization, Slater's condition serves as a fundamental criterion for guaranteeing strong duality and the existence of optimal Lagrange multipliers. This condition is particularly applicable to a specific class of optimization problems, typically denoted as problem (3.1.4) in the given context.

Definition 3.1.4.30 *Slater's conditions for problem* (3.1.4) *include the following criteria*

1. *The set C must be a convex subset of $\mathbb{R}^n$, and the function $f, h_1, ..., h_p$ should be convex over C.*
2. *The optimal value $\min(P)$ should be finite.*
3. *There should exist an $\overline{x} \in ri(C)$ (relative interior of C) such that*

$$A\overline{x} = b \quad \textit{and} \quad h(\overline{x}) < 0.$$

Note that for any inequality function $h(x)$ is affine, strict inequality is not required.

Example 3.1.4.31. Consider the problem

$$\begin{aligned} \min_{x\in\mathbb{R}^2} f(x) = \frac{1}{2}\|x\|_2^2 \\ \text{such that } a^\top x = 1, \\ h_1(x) := -x_1 \leq 0, \\ h_2(x) := -x_2 \leq 0, \end{aligned}$$

where $a = (1, 1)^{\mathsf{T}}$.

We have for $\lambda \in (0, 1)$

$$\nabla^2 f(x) = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}.$$

So, $x^\top \nabla^2 f(x) x = 2x_1^2 + 2x_2^2 > 0$ for all $x \neq 0$, which means $f(x)$ is strictly convex. Those inequalities $h_i(x)$ are affine, hence convex. The equality $a^\top x = 1$ is affine, which is a convex set. Choose the point

$$\bar{x} = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix},$$

then

$$a^\top \bar{x} = \frac{1}{2} + \frac{1}{2} = 1, \quad h_1(\bar{x}) = -\frac{1}{2} < 0, \quad h_2(\bar{x}) = -\frac{1}{2} < 0.$$

Because $\bar{x}$ lies in the *relative interior* of the feasible set and satisfies all inequalities strictly, Slater's condition holds.

Now, we introduce the following theorem for the existence of optimal multipliers.

Theorem 3.1.4.32 *If Slaters conditions are satisfied for problem 3.1.4, then it is guaranteed that there exist Lagrange multipliers* $(y^*, z^*) \in \mathbb{R}^m \times \mathbb{R}^p_+$ *such that*

$$\inf(P) = \inf_{x \in C} L(x^*, y^*, z^*) = \max(D).$$

Proof. We begin by noting that C is assumed to have nonempty interior that has $\text{rank}(A) = m$. If C has an empty interior, we can reframe the problem in the subspace parallel to C's affine hull. Likewise, if A is not full rank, redundant equations can be eliminated until a full row rank is achieved.

Let $f^* = \inf(P)$ and define the set

$$A = \{(t, u, v) : \exists x \in C, f(x) \leq t, Ax - b = u, h(x) \leq v\}.$$

Given the convexity of f and h_i, it is evident that A forms a convex subset $\mathbb{R} \times \mathbb{R}^m \times \mathbb{R}^p$. We see that the point $(f^*, 0, 0)$ is not an interior point of A, as otherwise, $(f^*, -\varepsilon, 0)$ would belong to A for some $\varepsilon > 0$, contradicting the fact that $f^* = \inf(P)$. Using the Supporting Hyperplane Theorem in theorem 3.1.3.21, there exists a non-zero (α, y, z) such that

$$\alpha f^* \leq \alpha t + y^\top u + z^\top v \quad \forall\, (t, u, v) \in A.$$

Suppose that $\alpha = 0$, for $\overline{x} \in \text{int}(C)$ satisfying the Slater's conditions, we have $0 \leq z^\top h(\overline{x})$, implying $z = 0$. This leads to $0 \leq y^\top A(x - \overline{x})$, and since $\overline{x} \in \text{int}(C)$, it follows that $A^\top y = 0$, which contradicts $\text{rank}(A) = m$. Hence, $\alpha > 0$. With $\alpha > 0$, we define $y^* = \dfrac{y}{\alpha}, z^* = \dfrac{z}{\alpha}$, leading to the inequality

$$f^* \leq d(y^*, z^*) = \inf L(x, y^*, z^*) \leq f^* \quad \forall x \in C.$$

Here, the second inequality follows from weak duality, completing the proof. □

Example 3.1.4.33. Solve the problem using duality

$$\min_{\frac{1}{2}x^\top Ax \leq 1} c^\top x$$

where A is a positive definite symmetric matrix of size $n \times n$, and c is a vector in $\mathbb{R}^n$.

Forming the Lagrangian with multiplier $\lambda \geq 0$, we have

$$L(x, \lambda) = c^\top x + \lambda(\frac{1}{2}x^\top Ax - 1).$$

Taking the gradient with respect to x and setting it to zero, we have

$$\nabla_x L = c + \lambda Ax = 0.$$

Since A is positive definite, it is invertible, and we can solve for x

$$x = -\frac{1}{\lambda}A^{-1}c.$$

Substituting this expression back into the Lagrangian gives us the dual function

$$d(\lambda) = -\frac{1}{\lambda}c^\top A^{-1}c + \frac{\lambda}{2}(\frac{1}{\lambda^2}c^\top A^{-1}AA^{-1}c - 1).$$

Simplifying the objective function of dual problem, we have

$$d(\lambda) = -\frac{1}{2\lambda}c^\top A^{-1}c - \lambda.$$

The dual problem becomes

$$\max_{\lambda \geq 0} -\frac{1}{2\lambda}c^\top A^{-1}c - \lambda.$$

Taking the derivative with respect to λ and setting it to zero

$$\frac{d}{d\lambda}d(\lambda) = \frac{1}{2\lambda^2}c^\top A^{-1}c - 1 = 0.$$

Solving for λ, we have

$$\lambda^* = \sqrt{\frac{1}{2}c^\top A^{-1}c}.$$

The optimal primal solution is therefore

$$x^* = -\frac{1}{\lambda^*}A^{-1}c = -\sqrt{\frac{2}{c^\top A^{-1}c}}A^{-1}c.$$

Substituting x^* into the constraint, we have

$$\frac{1}{2}(x^*)^\top Ax^* = \frac{1}{2}(\sqrt{\frac{2}{c^\top A^{-1}c}})^2(c^\top A^{-1})A(A^{-1}c) = \frac{1}{c^\top A^{-1}c}(c^\top A^{-1}c) = 1.$$

Since A is positive definite, A^{-1} is also positive definite. Therefore, for any nonzero vector c, we have $c^\top A^{-1}c > 0$. This implies $\lambda^* = \sqrt{\frac{1}{2}c^\top A^{-1}c} > 0$.

Finally, strong duality holds in this problem because it is convex (the objective is linear and the constraint is quadratic with positive definite A) and Slater's condition is satisfied. To see this, note that we can find a strictly feasible point $\tilde{x}$ such that $\frac{1}{2}\tilde{x}^\top A\tilde{x} < 1$ by taking any nonzero vector and scaling it appropriately.

Therefore, we can conclude the optimal value of the objective function is

$$c^\top x^* = -\sqrt{2c^\top A^{-1}c}.$$

Remark 3.1.4.34. Despite the problem being convex, there can be scenarios where the duality gap is nonzero if constraint qualification is not satisfied. The following example is in this sense.

Example 3.1.4.35. Consider the convex problem

$$\min e^{-\sqrt{x_1 x_2}}, \quad \text{subject to} \quad x_2 = 0, \quad x \geq 0.$$

By defining $C = \{x \in \mathbb{R}^2 : x \geq 0\}$ and considering $y \in \mathbb{R}$, the dual function becomes

$$d(y) = \begin{cases} \inf e^{-\sqrt{x_1 x_2}} + yx_2 & \text{if } y \geq 0, \\ -\infty & \text{if } y < 0. \end{cases}$$

The dual problem is then $\max_{y\geq 0} d(y)$, leading to $\max(D) = 0$, while $\inf(P) = 1$. The resulting duality gap is

$$\inf(P) - \max(D) = 1.$$

In this example, Slaters conditions are not satisfied, demonstrating that even in convex problems, the absence of these conditions can lead to a nonzero duality gap.

3.2 Optimal Transport

The term *transport* appears everyday at any moment of our life. It exists not only in daily situations such as transporting and arranging personal objects to personal storing space, delivering (transporting) dishes to customers' tables in restaurant, ... but also in much more large-scale ones like distributing goods which are produced in manufactories to warehouses and from those warehouses to shops where buyers can purchase them. In 1781, Gaspard Monge asked himself about the problem of transporting soil tiles from source location to digged holes at desination one to strengthen embankments serving military purposes. Like other mathematicians, Gaspard Monge wanted to find the most efficient strategy to complete that work. In other words, he tried to create an optimal plan for transporting soil tiles between two places. The problem we have just stated are now known as Monge problem and in that, the concept of "optimal transport" is used for the first time. In Monge problem, we assert the efficiency of one solution based on how much effort we have made for it. To estimate the effort, a *cost function* is used. This function gives us the cost we must spend for transporting some units of a thing from its source position to target one. For example, in problem of distributing goods from warehouses to shops, the cost function expresses the amount of resources which have been used to deliver a unit of good from any warehouse to any shop such as fuel, time, human resource, ... Coming back to the Monge problem, this problem's cost function demonstrates how far a soil tiles is shipped to its proposed position at target place. Now, using the cost function defined previously, we need to find the most optimal strategy to complete the work. This strategy is called a *transport map* which shows the position at target location where a soil tile should be transported to.

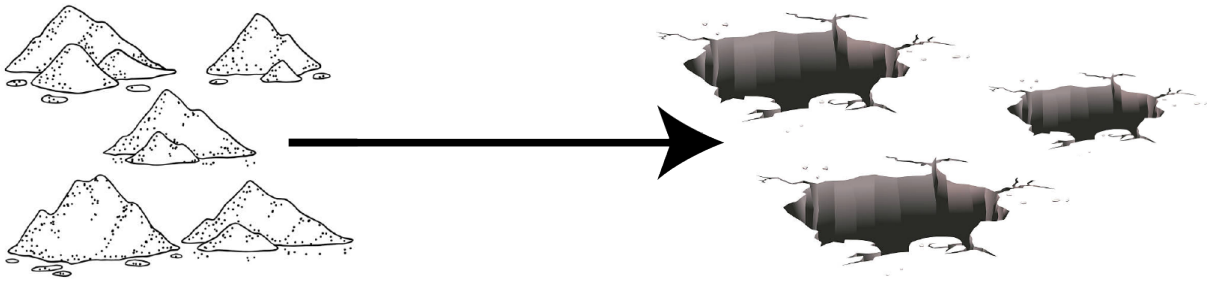


Figure 3.2: Monge problem finds the most efficient way to transport soil tiles from their source locations to soil holes which need to be filled at target location.

For centuries, Monge problem and all of its varied versions have attracted lots of concern and attention from researchers. Many progresses have been made, many approaches to solve the problem have been developed and optimal transport is now still an active topic and gradually prove its remarkable advantages in practice. This chapter will cover important concepts and notes about optimal transport as well as important tools which are created from its theory.

3.2.1 Monge Problem

Discrete Case

In practice, we usually deal with optimal transport in term of objects which can be thought as discrete-like problem. Therefore, it is reasonable to start from discrete case for further exploration. Now, we will try to model the original Monge problem in such a way. At first, we have n soil tiles which need transporting. Let

$$\mathcal{X} = \{x_1, x_2, \ldots, x_n\} \subset \mathbb{R}^2$$

be a set of source positions (longitude and latitude pairs) of n soil tiles with set of masses

$$\mathcal{A} = \{a_1, a_2, \ldots, a_n\} \subset \mathbb{R}_+.$$

Let

$$\mathcal{Y} = \{y_1, y_2, \ldots, y_m\} \subset \mathbb{R}^2$$

be a set of m proposed target positions where n soil tiles will be transported to whose capacities

$$\mathcal{B} = \{b_1, b_2, \ldots, b_m\} \subset \mathbb{R}_+.$$

We also have a cost function $c : \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}_+$ defined by

$$c(x, y) = \|x - y\|_2$$

which shows us the Euclidean distance between two positions x and y for every $x, y \in \mathbb{R}^2$. What we need to find is a strategy, or in other words, a transport map from X to Y illustrating which target position a soil tile will be moved to. Let $T : \mathbb{R}^2 \to \mathbb{R}^2$ be such that map. The function T is defined by

$$T(x_i) = y_j$$

showing that a soil tile at x_i will be transported to y_j in which $1 \leq i \leq n$ and $1 \leq j \leq m$.

Using all things defined above, we have that the total transport cost we need to minimize is

$$\sum_{i=1}^{n} c\left(x_i, T(x_i)\right) a_i$$

in which the term $c\left(x_i, T(x_i)\right) a_i$ is the effort to move a_i of soil tile at x_i to its corresponding target position $T(x_i)$ for every $1 \leq i \leq n$. Next, we define the constraints of the problem. Due to mass preservation principle, we have that total mass at source location must be equal to total capacity at target location. That means

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

Additionally, mass preservation principle states that for all soil tiles which are transported to y_j for every $1 \leq j \leq m$, their total mass must be equal to the capacity in y_j. In other words, we have

$$b_j = \sum_{i:\ T(x_i)=y_j} a_i.$$

Therefore, discrete Monge problem can be considered as an optimization problem defined by

$$\inf_{T} \sum_{i=1}^{n} c\left(x_i, T(x_i)\right) a_i \tag{3.2.1}$$

subjecting to the constraints

$$\begin{cases} \sum_{i=1}^{n} a_i = \sum_{j=1}^{m} b_j, \\ b_j = \sum_{i:\ T(x_i)=y_j} a_i \text{ for every } 1 \leq j \leq m. \end{cases}$$

In general cases, we can use any two sets of source and target points $\mathcal{X}$ and $\mathcal{Y}$, *e.g.* sets of cities' names. Moreover, we need to define more general cost function $c : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}_+$ as well as a transport map $T : \mathcal{X} \to \mathcal{Y}$ and form the problem as defined above. Following is an example.

Example 3.2.1.1. There are two nations X, Y and some cities of Y want to import computers from some cities of X. The set of cities providing computers in X is $C_X = \{x_1, x_2, x_3\}$ with numbers of computers $N_X = \{2, 4, 2\}$. Also, the set of cities importing computers in Y is $C_Y = \{y_1, y_2\}$ with needed numbers of computers $N_Y = \{4, 4\}$. We have the cost function $c : C_X \times C_Y \to \mathbb{R}_+$ defined by

$$\begin{cases} c(x_1, y_1) &= \$11, \\ c(x_1, y_2) &= \$100, \\ c(x_2, y_1) &= \$3, \\ c(x_2, y_2) &= \$5, \\ c(x_3, y_1) &= \$6, \\ c(x_3, y_2) &= \$20 \end{cases}$$

which gives us the cost of transporting one computer among cities between two nations. Therefore, the Monge problem of this example is

$$\inf_{T} \sum_{i=1}^{3} c\left(x_i, T(x_i)\right) (N_X)_i$$

subjecting to the constraints

$$\begin{cases} \sum_{i=1}^{3} (N_X)_i = \sum_{j=1}^{2} (N_Y)_j), \\ (N_Y)_j = \sum_{i:\ T(x_i)=y_j} (N_X)_i \text{ for every } 1 \leq j \leq 2 \end{cases}$$

in which $(N_X)_i$ is the number of computers city x_i has and $(N_Y)_j$ is the number of computers city y_j needs for every $i \in \{1, 2, 3\}$ and $j \in \{1, 2\}$.

In most of cases, the equality constraint of total mass is attached to the problem statement itself. That means in Problem (3.2.1), the constraint

$$\sum_{i=1}^{n} a_i = \sum_{j=1}^{m} b_j$$

is always true. By normalizing two sets of masses $\mathcal{A}$ and $\mathcal{B}$ in Problem (3.2.1) such that the total mass is equal to 1, we can create two probability vectors which are

$$\mathbf{a} = \frac{1}{\sum_{i=1}^{n} a_i} \left[a_1, a_2, \ldots, a_n\right]^\top$$

and

$$\mathbf{b} = \frac{1}{\sum_{j=1}^{m} b_j} \left[b_1, b_2, \ldots, b_m\right]^\top$$

in which $a_i \in \mathcal{A}$ and $b_j \in \mathcal{B}$ for every $1 \leq i \leq n$ and $1 \leq j \leq m$. Using these two vectors, we construct two discrete probability measures on $\mathcal{X}$ and $\mathcal{Y}$ as

$$\alpha = \sum_{i=1}^{n} \mathbf{a}_i \delta_{x_i} \text{ and } \beta = \sum_{j=1}^{m} \mathbf{b}_j \delta_{y_j} \tag{3.2.2}$$

in which δ_{x_i} is the indicator function of $x_i \in \mathcal{X}$ which has value 1 if its input includes x_i and value 0 otherwise. Now, we need the definition of push-forward operator and measure to demonstrate the constraint between two measures α and β.

Definition 3.2.1.2 (Push-forward operator and measure) *Let $T : \mathcal{X} \to \mathcal{Y}$ be a measurable mapping and both $\mathcal{X}$ and $\mathcal{Y}$ are measurable sets. Let α and β be two measures on $\mathcal{X}$ and $\mathcal{Y}$, respectively. $T_\sharp$ is called the push-forward operator and the push-forward measure*

$$T_\sharp \alpha = \beta$$

if and only if for any measurable subset K of $\mathcal{Y}$,

$$\beta(K) = \alpha\left(T^{-1}(K)\right).$$

Using Definition 3.2.1.2, we can rewrite our first Problem (3.2.1) in a more general way. Let $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ and $\mathcal{Y} = \{y_1, y_2, \ldots, y_m\}$ be two measurable sets. Let α and β be two discrete probability measures on $\mathcal{X}$ and $\mathcal{Y}$, respectively. Let $c : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}_+$ be a cost function. Now the problem turns into finding a measurable mapping $T : \mathcal{X} \to \mathcal{Y}$

$$\inf_T \sum_{i=1}^{n} c\left(x_i, T(x_i)\right) \alpha(x_i) \tag{3.2.3}$$

subjecting to the constraint

$$T_\sharp \alpha = \beta.$$

The one above is known as *Monge discrete formulation* for the Monge problem in discrete case.

Existence of Monge Problem for Discrete Case

From the previous part, a central question in the Monge problem is whether minimizers concentrate on the graphs of function over a variable x_1. Solution for this structure are called *Monge solution*.

In general, the behavior of Monge solutions in the discrete setting exhibits considerable complexity. When dealing with discrete measures, optimal couplings generally lack uniqueness, and moreover, solutions of Monge type (deterministic mappings) may fail to exist entirely within the space of admissible couplings $\Pi(\mu_1, \mu_2, ..., \mu_N)$ (probability measures on $X_1 \times X_2 \times \cdot \times X_N$ whose probability measure are the μ_i).

However, the situation becomes more tractable when we consider a specific class of measures: the m-empirical measures, where each marginal is uniformly distributed on exactly m points. This class is particularly significant as it frequently appears in practical applications. For these measures, we can guarantee the existence of transport maps, though they may not be unique.

The case of two marginals ($N = 2$) is especially well-behaved. First we introduce a theorem

Theorem 3.2.1.3 (Birkhoff-von Neumann) *Every doubly stochastic matrix can be written as a convex combination of permutation matrices. Specifically, if P is an $n \times n$ doubly stochastic*

matrix (non-negative entries with all rows and columns summing to 1), then there exist permutation matrices $\Pi_1, \ldots, \Pi_K$ *and coefficients* $\alpha_1, \ldots, \alpha_K \geq 0$ *with* $\sum_{k=1}^{K} \alpha_k = 1$ *such that*

$$P = \sum_{k=1}^{K} \alpha_k \Pi_k.$$

Armed with this fundamental result, we can now prove the existence of Monge solutions in the two-marginal case

Theorem 3.2.1.4 (Existence of Monge Solutions for Two Marginals) *Let* μ_1 *and* μ_2 *be* m*-empirical measures on* $\mathbb{R}^d$*, i.e.,*

$$\mu_1 = \frac{1}{m} \sum_{i=1}^{m} \delta_{x_i} \quad \textit{and} \quad \mu_2 = \frac{1}{m} \sum_{j=1}^{m} \delta_{y_j}.$$

Then for any cost function $c : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$*, there exists at least one optimal transport plan that is of Monge type.*

The situation changes dramatically when we consider three or more marginals. Even in the seemingly well-behaved case of m-empirical measures, Monge solutions may fail to exist. This phenomenon can be understood through the geometric structure of the transport problem:

For $N \geq 3$, Gero Friesecke has shown that the set of admissible transport plans $\Pi(\mu_1, \ldots, \mu_N)$ forms a convex polytope whose extreme points exhibit a crucial difference from the two-marginal case. While all extreme points correspond to Monge maps when $N = 2$, this property fails for $N \geq 3$. Some extreme points of the polytope are inherently non-deterministic, meaning they cannot be represented as Monge maps.

This structural property has immediate consequences for optimal transport: since linear functionals on polytopes achieve their minima at extreme points, we can construct cost functions whose optimal solutions must occur at these non-Monge extreme points. Such cost functions arise naturally in physical applications, and Gero Friesecke also provides concrete examples where Monge solutions cannot exist.

Continuous Case

In the continuous case, we consider Monge problem over continuous measures rather than discrete ones. This generalization is crucial in understanding a wider class of problems where the distributions of sources and targets are described by probability measures on continuous spaces.

Let $\mathcal{X}$ and $\mathcal{Y}$ be two measurable subsets of $\mathbb{R}^d$, representing the source and target domains, respectively. Let μ and ν be two probability measures defined on $\mathcal{X}$ and $\mathcal{Y}$, respectively. The problem is to find a measurable mapping $T : \mathcal{X} \to \mathcal{Y}$ such that

1. The transport map T pushes forward the measure μ to ν, denoted as $T_\sharp \mu = \nu$. This means for any measurable subset $B \subset \mathcal{Y}$,

$$\nu(B) = \mu\left(T^{-1}(B)\right).$$

2. The cost of transportation, given by the cost function $c : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}_+$, is minimized. The total transport cost can then be expressed as

$$\int_{\mathcal{X}} c(x, T(x))\, d\mu(x),$$

where the goal is to minimize this integral subject to the constraint $T_\sharp \mu = \nu$.

Thus, Monge problem in the continuous case can be written as

$$\inf_T \int_{\mathcal{X}} c(x, T(x))\, d\mu(x), \tag{3.2.4}$$

subject to the constraint

$$T_\sharp \mu = \nu.$$

Monge problem in the continuous case can be written as

$$\inf_T \int_{\mathcal{X}} c(x, T(x))\, d\mu(x),$$

subject to the constraint

$$\beta(B) = \alpha\left(T^{-1}(B)\right)\ \forall B \subset \mathcal{Y}.$$

Example 3.2.1.5. Let us consider transporting a continuous resource from a source region $\mathcal{X}$ to a target region $\mathcal{Y}$. The source region is $\mathcal{X} = [0, 1]$ and target region is $\mathcal{Y} = [0, 1]$. The source density is

$$\rho_{\text{source}}(x) = 3x^2, \quad x \in [0, 1],$$

and the target density is

$$\rho_{\text{target}}(y) = 2 - 2y, \quad y \in [0, 1].$$

The cost of transporting a unit of resource from $x \in \mathcal{X}$ to $y \in \mathcal{Y}$ is given by the squared Euclidean distance

$$c(x, y) = (x - y)^2.$$

The transport map $T : \mathcal{X} \to \mathcal{Y}$ must satisfy the mass preservation condition

$$\rho_{\text{target}}(T(x)) \cdot T'(x) = \rho_{\text{source}}(x),$$

or equivalently,

$$(2 - 2T(x)) \cdot T'(x) = 3x^2.$$

This is a first-order differential equation that relates the transport map $T(x)$ to the source and target densities. The objective is to minimize the total transport cost

$$\int_{\mathcal{X}} (x - T(x))^2 \cdot \rho_{\text{source}}(x)\, dx.$$

Substituting the source density

$$\int_0^1 (x - T(x))^2 \cdot 3x^2\, dx.$$

The continuous Monge problem can now be expressed as

$$\inf_T \int_0^1 (x - T(x))^2 \cdot 3x^2\, dx,$$

subject to the constraint

$$(2 - 2T(x)) \cdot T'(x) = 3x^2.$$

Analysis about The Convexity of Monge Problem

The Monge optimal transport problem, formulated as

$$\inf_T \int_{\mathcal{X}} c(x, T(x))\, d\mu(x),$$

where $T : \mathcal{X} \to \mathcal{Y}$ is the transport map, $c : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ is the cost function, and μ is the source measure, inherently involves challenges related to convexity. While the functional's convexity plays a vital role in optimization, the Monge problem's structure does not guarantee a convex formulation due to constraints and non-linearities.

In the Monge formulation, the requirement that T is a map (i.e., deterministic) introduces significant non-convexity. Specifically, the set of all admissible transport maps T such that $T_\sharp \mu = \nu$ is generally not convex. This is because a convex combination of two transport maps is not, in general, a valid transport map. This non-convexity makes the Monge problem challenging to solve using standard convex optimization techniques.

The cost function $c(x, T(x))$ also affects the convexity of the problem. If $c(x, y)$ is strictly convex in y for each fixed x, the functional $\int_{\mathcal{X}} c(x, T(x))\, d\mu(x)$ can exhibit convex-like behavior, but the non-convexity of the feasible set remains dominant. Conversely, if $c(x, y)$ is linear or piecewise convex, the problem is further complicated by potential non-uniqueness in the solutions.

Example 3.2.1.6. *Non-convexity in Transport Maps* Consider the following setup, the source measure μ is uniformly distributed on the unit interval $[0, 1]$, with density

$$\rho_{\text{source}}(x) = 1, \quad x \in [0, 1].$$

The target measure ν is also uniform on $[0, 1]$, with density

$$\rho_{\text{target}}(y) = 1, \quad y \in [0, 1].$$

The cost function is $c(x, y) = |x - y|^2$. The goal is to find a transport map $T(x)$ such that

1. The mass preservation condition $T_\sharp \mu = \nu$ holds.
2. The total transport cost is minimized

$$\inf_T \int_0^1 (x - T(x))^2\, dx.$$

In this setup, the cost function $c(x, y) = |x - y|^2$ is strictly convex in y for fixed x, encouraging a unique optimal map under certain conditions. However, the set of admissible transport maps $T(x)$, which satisfy $T_\sharp \mu = \nu$, is not convex. For example, if $T_1(x) = x$ and $T_2(x) = 1 - x$ are both valid transport maps, then any convex combination $(1 - \lambda)T_1(x) + \lambda T_2(x)$, for $\lambda \in (0, 1)$, is generally not a valid transport map.

This non-convexity prevents direct application of standard convex optimization methods and necessitates specialized approaches to solve the Monge problem. While the admissible set of transport maps is non-convex, the cost function $c(x, y)$ impacts the tractability of the problem

- For $c(x, y) = |x - y|^2$, the problem has favorable properties due to the convexity of $y \mapsto c(x, y)$, which enables uniqueness under suitable conditions (e.g., absolute continuity of the measures).

- If $c(x, y)$ is not strictly convex (e.g., $c(x, y) = |x - y|$), the Monge problem may admit multiple solutions, even if transport maps exist.

The Monge problem's inherent non-convexity arises primarily from the deterministic nature of transport maps. While convexity in the cost function helps ensure well-behaved solutions, the non-convex feasible set limits the direct application of convex optimization techniques. To address these challenges, methods such as the Kantorovich relaxation (discussed later) or specialized algorithms are often employed.

3.2.2 Kantorovich Formulation

Limitations of Monge Formulation

Recall the general Problem (3.2.3), our goal is to find a mapping $T : \mathcal{X} \to \mathcal{Y}$ which minimizes the total transport cost. Because T is indeed a mapping, we know that any $x \in \mathcal{X}$ can not be mapped to more than one target position in $\mathcal{Y}$. We now consider the following example.

Example 3.2.2.1. There are two nations X, Y and some cities of Y want to import computers from some cities of X. The set of cities providing computers in X is $C_X = \{x_1, x_2, x_3\}$ with numbers of computers $N_X = \{4, 4, 4\}$. Also, the set of cities importing computers in Y is $C_Y = \{y_1, y_2, y_3, y_4, y_5, y_6\}$ with needed numbers of computers $N_Y = \{2, 2, 2, 2, 2, 2\}$. According to Monge problem idea, we need to find a strategy such that each city of X only exports all its computers to one city of Y. Because every number of computers in each city of X is greater than the needed one in every city of Y, we know that such a strategy is impossible.

In above example, the key disadvantage is that a city in X can not split its computer supply to distribute them to more than two cities in Y. It is the property of a mapping. That means Monge problem does not always have solution. Thinking about it, in 1939, mathematician and economist Kantorovich proposed a method to overcome this limitation of Monge problem. Kantorovich allowed a mass to be splitted and its proportions can be transported to multiple different target points. It is called *Kantorovich relaxation* in which the constraint of a mapping is relaxed.

Discrete Case

Now, we consider the Problem (3.2.1) again with the same settings. As stated before, we know that the property of mapping does not allow splitting mass when transporting. Therefore, applying *Kantorovich relaxation* to Problem (3.2.1), instead of find an optimal transport map T, we need to find a transport plan P which can represent proportions of mass to be transported rather than the whole mass itself.

In fact, the transport plan P is a matrix in $\mathbb{R}^{n\times m}$ in which n and m are the number of points in source location and target location, respectively. Let $P_{i,j}$ be the entry at row i and column j of matrix P. We define $P_{i,j}$ as the amount of mass a_i at source position x_i which is transported to the target position y_j. For example we have a transport plan $P' \in \mathbb{R}^{2\times 2}$ which is defined by

$$P' = \begin{pmatrix} 1 & 3 \\ 0.2 & 0 \end{pmatrix}.$$

We have from P' that 1 unit of mass a_1 at source position x_1 will be transported to target position y_1, 3 unit of mass a_1 at source position x_1 will be transported to target position y_2, 0.2 unit of mass a_2 at source position x_2 will be transported to target position y_1 and no unit of

mass a_2 at source position x_2 will be transported to target position y_2.

Also, rather than using the cost function c, we introduce a cost matrix $C \in \mathbb{R}^{n \times m}$ which has the same role as c. Let $C_{i,j}$ be the transport cost to transport a unit of mass a_i at source position x_i to the target position y_j. In other words, $C_{i,j} = c(x_i, y_j)$.

Using all things defined above, we have that the total transport cost we need to mminimize is

$$\langle P, C \rangle_F$$

in which $\langle P, C \rangle_F$ denotes Frobenius inner product between P and C. Next, we need to determine the constraints of the problem. Again, due to mass preservation principle, we have that total mass at source location must be equal to total capacity at target location. That means

$$\sum_{i=1}^{n} a_i = \sum_{j=1}^{m} b_j$$

Let $1_{\mathbb{R}^n}$ be a vector whose all entries are one in $\mathbb{R}^n$. We will use this denotation to define next two constraints. Because of mass preservation principle, we have that the total transported mass from x_i must be equal to a_i for every $i \in \{1, \ldots, n\}$. Let $P_i \in \mathbb{R}^m$ be the transposed row vector i of P. That means we have

$$P_i 1_{\mathbb{R}^m} = a_i \tag{3.2.5}$$

for every $i \in \{1, \ldots, n\}$. With the same intuition, we have that the total mass which is transported to target position y_j must be equal to b_j for every $j \in \{1, \ldots, m\}$. Let $P_j^\top \in \mathbb{R}^n$ be the transposed column vector j of P. We also have that

$$P_j^\top 1_{\mathbb{R}^n} = b_j \tag{3.2.6}$$

for every $j \in \{1, \ldots, m\}$. From Constraint (3.2.5) and Constraint (3.2.6), our next two constraints are

$$\begin{cases} P 1_{\mathbb{R}^m} &= (a_1, \ldots, a_n)^\top, \\ P^\top 1_{\mathbb{R}^n} &= (b_1, \ldots, b_m)^\top. \end{cases}$$

Finally, because we can not transport a negative mass, all the entries of P must be non-negative. In other words, we have

$$P_{i,j} \geq 0$$

for every $i \in \{1, \ldots, n\}$ and $j \in \{1, \ldots, m\}$.

Therefore, we redefine the Problem (3.2.1) with Kantorovich relaxation by

$$\inf_P \langle P, C \rangle_F \tag{3.2.7}$$

subjecting to the constraints

$$\begin{cases} \sum_{i=1}^{n} a_i &= \sum_{j=1}^{m} b_j \\ P 1_{\mathbb{R}^m} &= (a_1, \ldots, a_n)^\top, \\ P^\top 1_{\mathbb{R}^n} &= (b_1, \ldots, b_m)^\top, \\ P_{i,j} &\geq 0 \text{ for every } i \in \{1, \ldots, n\} \text{ and } j \in \{1, \ldots, m\}. \end{cases}$$

Like original discrete Monge problem, in general case, we can use any two sets of source and target points $\mathcal{X}$ and $\mathcal{Y}$, *e.g.* sets of cities' names to form the problem as defined above.

Example 3.2.2.2. Using the same provided information from Example 3.2.2.1, we now also define the cost matrix C by

$$C = \begin{pmatrix} 32 & 13 & 54 & 1 & 100 & 120 \\ 6 & 22 & 1000 & 19 & 73 & 82 \\ 200 & 44 & 97 & 11 & 3 & 55 \end{pmatrix}.$$

After solving it, we obtain the optimal transport plan P In intuition of discrete Monge problem, such a transport map T can not be found. However, with Kantorovich relaxation, we can find an optimal transport plan P instead. Here, the Monge problem with Kantorovich of this example is

$$\inf_P \langle P, C \rangle_F$$

subjecting to the constraints

$$\begin{cases} \sum_{i=1}^{2} (N_X)_i &= \sum_{j=1}^{6} (N_Y)_j \\ P1_{\mathbb{R}^6} &= (4,4,4)^\top, \\ P^\top 1_{\mathbb{R}^2} &= (2,2,2,2,2,2)^\top, \\ P_{i,j} &\in \mathbb{N} \text{ for every } i \in \{1,2,3\} \text{ and } j \in \{1,\ldots,6\} \end{cases}$$

in which $(N_X)_i$ is the number of computers city x_i has and $(N_Y)_j$ is the number of computers city y_j needs for every $i \in \{1,2,3\}$ and $j \in \{1,\ldots,6\}$.

Like original Monge problem, in most of cases, the equality constraint of total mass is also attached to the problem statement itself. If we use the discrete probability measures α and β which are defined at (3.2.2), we can define the Problem (3.2.7) in a more general way. Again, let $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ and $\mathcal{Y} = \{y_1, y_2, \ldots, y_m\}$ be two measurable sets. Let α and β be two discrete probability measures on $\mathcal{X}$ and $\mathcal{Y}$, respectively. Let $C \in \mathbb{R}^{n \times m}$ be a cost matrix. Now the problem turns into finding a transport plan $P \in \mathbb{R}^{n \times m}$

$$\inf_P \langle P, C \rangle_F$$

subjecting to the constraints

$$\begin{cases} P1_{\mathbb{R}^m} &= (\alpha(x_1), \ldots, \alpha(x_n))^\top, \\ P^\top 1_{\mathbb{R}^n} &= (\beta(y_1), \ldots, \beta(y_m))^\top, \\ P_{i,j} &\geq 0 \text{ for every } i \in \{1,\ldots,n\} \text{ and } j \in \{1,\ldots,m\}. \end{cases}$$

Let $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ and $\mathcal{Y} = \{y_1, y_2, \ldots, y_m\}$ be two measurable sets. Let α and β be two discrete probability measures on $\mathcal{X}$ and $\mathcal{Y}$, respectively. Let $C \in \mathbb{R}^{n \times m}$ be a cost matrix. The problem is to find a transport plan $P \in \mathbb{R}^{n \times m}$

$$\inf_P \langle P, C \rangle_F$$

subjecting to the constraints

$$\begin{cases} P1_{\mathbb{R}^m} &= (\alpha(x_1), \ldots, \alpha(x_n))^\top, \\ P^\top 1_{\mathbb{R}^n} &= (\beta(y_1), \ldots, \beta(y_m))^\top, \\ P_{i,j} &\geq 0 \text{ for every } i \in \{1,\ldots,n\} \text{ and } j \in \{1,\ldots,m\}, \end{cases}$$

where $1_{\mathbb{R}^n}$ be a vector whose all entries are one in $\mathbb{R}^n$ and $\langle P, C \rangle_F$ denotes Frobenius inner product between P and C

The one above is known as *Kantorovich's discrete formulation* for the Monge problem with Kantorovich relaxation in discrete case.

Continuous Case

In the previous part, we have known the idea of finding transport plan instead of transport map in Kantarovich relaxation. In continuous case, splitting mass can be made rigorous by using the notion of measure. Moreover, it allows to consider more general initial and final measures.

Definition 3.2.2.3 *Let X be a topological space. We define $\mathcal{P}(X)$ as the space of probability measures on X, where each element is a Radon measure $\mu : X \to [0,1]$ satisfying the probability constraint $\mu(X) = 1$.*

Definition 3.2.2.4 *Consider two measure spaces (X, μ) and (Y, ν), where $\mu \in \mathcal{P}(X)$ and $\nu \in \mathcal{P}(Y)$. A measure $\gamma \in \mathcal{P}(X \times Y)$ is called a transport plan (or coupling) between μ and ν if it satisfies the following marginal constraints:*

(i) The first marginal of γ coincides with μ, that is, $(\pi_1)_{\#}\gamma = \mu$

(ii) The second marginal of γ coincides with ν, that is, $(\pi_2)_{\#}\gamma = \nu$

The collection of all such transport plans between μ and ν is denoted by $\Pi(\mu, \nu)$.

The concept of measures provides a mathematical foundation for rigorously defining mass transport. As illustrated in Figure 3.3, adapted from [15], we can precisely quantify mass transfer between source and destination sets using measure theory.

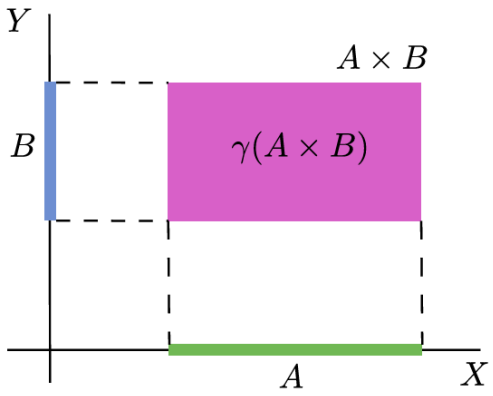


Figure 3.3: For any measurable subsets $A \subset X$ and $B \subset Y$, the measure $\gamma(A \times B)$ quantifies the amount of mass transferred from source set A to destination set B.

Remark 3.2.2.5 (Probabilistic Interpretation)**.** In probabilistic terminology, the measure γ can be understood as having μ and ν as its first and second marginal distributions, respectively. The set of transport plans $\Pi(\mu, \nu)$ is non-empty, as evidenced by the product measure $\mu \otimes \nu \in \Pi(\mu, \nu)$.

Remark 3.2.2.6 (Transport Plan Interpretation)**.** A transport plan γ can be interpreted as a prescription for mass redistribution: for each subset $A \subset X$, it specifies how the mass initially in A should be distributed across the target space Y. The marginal conditions (i) and (ii) in the previous definition arise naturally from two fundamental constraints:

- All initial mass μ must be transported from its source
- All final mass ν must be accounted for at its destination

Lemma 3.2.2.7 (Transport Plan Induced by a Map) *Let (X, μ) and (Y, ν) be measure spaces, where $\mu \in \mathcal{P}(X)$ and $\nu \in \mathcal{P}(Y)$. Consider a Borel measurable map $T : X \to Y$ that pushes forward μ to ν (i.e., $T_{\#}\mu = \nu$). Then the transport plan induced by T is given by*

$$\gamma_T := (\mathrm{Id}, T)_{\#}\mu,$$

where $(\mathrm{Id}, T) : X \to X \times Y$ is defined by $(\mathrm{Id}, T)(x) := (x, T(x))$.

Proof. Firstly, we verify the mass transport property. For Borel sets $A \subset X$ and $B \subset Y$, the mass transported by T from A to B is

$$\begin{aligned}\mu(A \cap T^{-1}(B)) &= \mu(\mathrm{Id}^{-1}(A) \cap T^{-1}(B)) \\ &= \mu((\mathrm{Id}, T)^{-1}(A, B)) \\ &= (\mathrm{Id}, T)_{\#}\mu(A, B).\end{aligned}$$

This establishes that γ_T indeed represents the desired transport plan.

We now verify that $\gamma_T \in \Pi(\mu, \nu)$. By construction, $\gamma_T \in \mathcal{P}(X \times Y)$. We need to verify conditions (i) and (ii) from definition 3.2.2.4.

For any μ-measurable set $A \subset X$

$$\begin{aligned}(\pi_1)_{\#}\gamma_T(A) &= \gamma_T((\pi_1)^{-1}(A)) \\ &= \gamma_T(A \times Y) \\ &= (\mathrm{Id}, T)_{\#}\mu(A \times Y) \\ &= \mu((\mathrm{Id}, T)^{-1}(A \times Y)) \\ &= \mu(A).\end{aligned}$$

For any ν-measurable set $B \subset Y$

$$\begin{aligned}(\pi_2)_{\#}\gamma_T(B) &= \gamma_T((\pi_2)^{-1}(B)) \\ &= \gamma_T(X \times B) \\ &= (\mathrm{Id}, T)_{\#}\mu(X \times B) \\ &= \mu((\mathrm{Id}, T)^{-1}(X \times B)) \\ &= \mu(T^{-1}(B)) \\ &= \nu(B),\end{aligned}$$

where the last equality follows from the assumption that $T_{\#}\mu = \nu$. □

Definition 3.2.2.8 (Kantorovich Problem) *Let (X, μ) and (Y, ν) be measure spaces, with $\mu \in \mathcal{P}(X)$ and $\nu \in \mathcal{P}(Y)$. Given a continuous cost function $c : X \times Y \to [0, \infty)$, the Kantorovich optimal transport problem is formulated as:*

$$\min \left\{ \int_{X \times Y} c(x, y)\, d\gamma : \gamma \in \Pi(\mu, \nu) \right\}. \tag{3.2.8}$$

Remark 3.2.2.9 (Connection to Monge Problem)**.** The Kantorovich problem provides a generalization of the classical Monge problem. This connection can be understood as follows:

(i) For a transport map T, the cost integral can be written as:

$$\int_{X \times Y} c(x, y)\, d\gamma_T = \int_{\mathbb{R}^N} c(x, T(x))\, d\mu.$$

(ii) In the specific case where $\mu = f\mathcal{L}^N$ (that is, μ is absolutely continuous with respect to the Lebesgue measure with density f), we have:

$$\int_{X\times Y} c(x,y)\, d\gamma_T = \int_{\mathbb{R}^N} c(x,T(x))f(x)\, dx.$$

Furthermore, by Lemma 3.2.2.7, since $\Pi(\mu,\nu)$ contains all transport maps, we obtain the inequality:

$$\inf\left\{\int_{X\times Y} c(x,y)\, d\gamma : \gamma \in \Pi(\mu,\nu)\right\} \leq \inf\left\{\int_{\mathbb{R}^N} c(x,T(x))f(x)\, dx : g = T_{\#}f\right\}.$$

This inequality demonstrates that the Kantorovich formulation provides a lower bound for the corresponding Monge problem.

Analysis about The Convexity of Kantorovich Problem

The Kantorovich problem possesses important structural properties, particularly concerning convexity, which make it more tractable than the original Monge problem. In this part, we will analyze these properties in detail.

Proposition 3.2.2.10 (Convexity of Transport Plans) *The set of transport plans $\Pi(\mu,\nu)$ is convex. That is, for any $\gamma_1,\gamma_2 \in \Pi(\mu,\nu)$ and $t \in [0,1]$, we have:*

$$t\gamma_1 + (1-t)\gamma_2 \in \Pi(\mu,\nu)$$

Proof. Let $\gamma_1,\gamma_2 \in \Pi(\mu,\nu)$ and $t \in [0,1]$. Set $\gamma_t = t\gamma_1 + (1-t)\gamma_2$. We need to verify that γ_t satisfies the marginal constraints.

For any Borel set $A \subset X$

$$\begin{aligned}(\pi_1)_{\#}\gamma_t(A) &= \gamma_t(\pi_1^{-1}(A)) \\ &= t\gamma_1(\pi_1^{-1}(A)) + (1-t)\gamma_2(\pi_1^{-1}(A)) \\ &= t\mu(A) + (1-t)\mu(A) \\ &= \mu(A).\end{aligned}$$

Similarly for any Borel set $B \subset Y$

$$(\pi_2)_{\#}\gamma_t(B) = \nu(B).$$

Therefore, $\gamma_t \in \Pi(\mu,\nu)$. □

Theorem 3.2.2.11 (Convexity of Kantorovich Functional) *Let $c : X \times Y \to [0,\infty)$ be a continuous cost function. The functional*

$$J(\gamma) = \int_{X\times Y} c(x,y)\, d\gamma$$

is linear, and hence convex, over $\Pi(\mu,\nu)$.

Proof. To prove linearity, we need to verify two properties

(i) *Homogeneity*: $J(\alpha\gamma) = \alpha J(\gamma)$ for any $\alpha \in \mathbb{R}$.

(ii) *Additivity*: $J(\gamma_1 + \gamma_2) = J(\gamma_1) + J(\gamma_2)$.

For any $\alpha \in \mathbb{R}$ and $\gamma \in \Pi(\mu, \nu)$

$$\begin{aligned} J(\alpha\gamma) &= \int_{X\times Y} c(x,y)\, d(\alpha\gamma) \\ &= \alpha \int_{X\times Y} c(x,y)\, d\gamma \quad \text{(by properties of integration)} \\ &= \alpha J(\gamma). \end{aligned}$$

For any $\gamma_1, \gamma_2 \in \Pi(\mu, \nu)$

$$\begin{aligned} J(\gamma_1 + \gamma_2) &= \int_{X\times Y} c(x,y)\, d(\gamma_1 + \gamma_2) \\ &= \int_{X\times Y} c(x,y)\, d\gamma_1 + \int_{X\times Y} c(x,y)\, d\gamma_2 \\ &= J(\gamma_1) + J(\gamma_2), \end{aligned}$$

where the second equality follows from the linearity of integration with respect to measures.

By combining homogeneity and additivity, we have for any $\alpha, \beta \in \mathbb{R}$

$$J(\alpha\gamma_1 + \beta\gamma_2) = \alpha J(\gamma_1) + \beta J(\gamma_2),$$

which proves that J is linear. Note that since J is linear, it is automatically both convex and concave, as

$$J(t\gamma_1 + (1-t)\gamma_2) = tJ(\gamma_1) + (1-t)J(\gamma_2),$$

for all $t \in [0,1]$ and $\gamma_1, \gamma_2 \in \Pi(\mu, \nu)$. □

Corollary 3.2.2.12 *The Kantorovich problem*

$$\min\left\{ \int_{X\times Y} c(x,y)\, d\gamma : \gamma \in \Pi(\mu,\nu) \right\}$$

is a convex optimization problem over a convex set.

3.2.3 Wasserstein Metric - A Property of Optimal Transport

Motivations

Recalling the metric (distance) definition, we know that one of the most naive intuitions of it is representing how far a "point" is from another one. The term point here can be any type of thing. For example, when we consider a travel from Ho Chi Minh city to the outskirt, we have that the distance from Ho Chi Minh city to Dong Nai province is 40km. Here, in this example, our two points are Ho Chi Minh city and Dong Nai province. Distance gives people a clear view along with good intuition about the farness or difference between things such as two places on map, two marks on a boards, two sewing lines on clothes, ... Not only in those cases, people also want to compute the distance between two arbitrary things, may be between two measures.

Coming back to the optimal transport problem we have developed so far, we know that its main goal is to find an optimal transport map or transport plan such that the transport cost is minimized. In fact, the transport cost is the total effort we need to "shift" a measure to another measure. In other words, if we think two measures in the optimal problem as two "points", then the minimized transport cost is indeed the distance between them. In this section, we will consider optimal transport when it takes a role as metric.

Wasserstein Metric

Now, we redefine our optimal transport problem which is Monge problem with Kantorovich's relaxation in discrete case. Let $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ and $\mathcal{Y} = \{y_1, y_2, \ldots, y_m\}$ which are two measurable sets. Let α and β are two discrete probability measures on $\mathcal{X}, \mathcal{Y}$, respectively. Let $C \in \mathbb{R}_+^{n\times m}$ be a cost matrix. Our goal is to find a optimal transport plan $P \in \mathbb{R}^{n\times m}$

$$\inf_P \langle P, C\rangle_F$$

subjecting to the constraints

$$\begin{cases} P1_{\mathbb{R}^m} &= (\alpha(x_1), \ldots, \alpha(x_n))^\top, \\ P^\top 1_{\mathbb{R}^n} &= (\beta(y_1), \ldots, \beta(y_m))^\top, \\ P_{i,j} &\geq 0 \text{ for every } i \in \{1, \ldots, n\} \text{ and } j \in \{1, \ldots, m\}. \end{cases}$$

Here, we introduce a new notation $L_C(\alpha, \beta)$ which represents the minimized transport cost obtained by solving the above problem with respect to cost matrix C and two discrete probability measures α, β. Note that the cost matrix C is constructed by taking the value of the cost function $c : \mathcal{X} \times \mathcal{Y} \mapsto \mathbb{R}_+$ at corresponding position, i.e. we have

$$C_{i,j} = c(x_i, y_j)$$

in which $x_i \in \mathcal{X}$ and $y_j \in \mathcal{Y}$.

When we delve into metric property of optimal transport, we assume that $\mathcal{X} = \mathcal{Y}$. With this assumption, for some $p \geq 1$, let d be a metric on $\mathcal{X}$ such that $c(x_i, x_j) = d(x_i, x_j)^p$ for every $i, j = 1, 2, \ldots, n$. Let $D \in \mathbb{R}_+^{n\times n}$ be a matrix defined by

$$D_{i,j} = d(x_i, x_j).$$

Thus, we have our new cost matrix $C \in \mathbb{R}_+^{n\times n}$ defined by

$$C = D^p$$

in which D^p denotes that all its entries are taken the power of p. By this, we can define a metric using optimal transport.

Definition 3.2.3.1 (Wasserstein metric) *Let $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ be a measurable set. Let α and β be two discrete probability measures on $\mathcal{X}$. Let d be a metric on $\mathcal{X}$. Let $D \in \mathbb{R}_+^{n\times n}$ be a matrix in which each its entry $D_{i,j}$ at i^{th} row and j^{th} column is defined by $D_{i,j} = d(x_i, x_j)$. For $p \geq 1$, let $C \in \mathbb{R}_+^{n\times n}$ be a cost matrix defined by $C = D^p$ in which each its entry $C_{i,j}$ at i^{th} row and j^{th} column is defined by $C_{i,j} = D_{i,j}^p$. Then*

$$\mathcal{W}_p(\alpha, \beta) = L_C(\alpha, \beta)^{1/p} = L_{D^p}(\alpha, \beta)^{1/p}$$

defines the Wasserstein metric on the set of discrete probability measures on $\mathcal{X}$. In particular, the equation above give us the definition of p-Wasserstein *metric.*

The Wasserstein metric was first defined by Leonid Kantorovich and coined by R. L. Dobrushin in 1970. It has a several other names such as Kantorovich–Rubinstein metric or earth mover's metric (involving in the earth moving problem which is the original form of Monge problem). The Wasserstein metric plays an important role in many tasks which is relevant to optimal transport. One of the most popular and powerful tools it provides for our problem is the Wasserstein space.

Remark 3.2.3.2. If we have the value of p as $0 < p < 1$, then D^p is itself metric. This implies that while for $p \geq 1$, $\mathcal{W}_p(\alpha, \beta)$ is a metric, in the case $0 < p < 1$, it is actually $\mathcal{W}_p(\alpha, \beta)^p$ which defined a metric.

Let us now consider an example.

Example 3.2.3.3. Let $\mathcal{X} = \{(1,0), (0,1), (1,2), (2,1)\}$ be a set of points in $\mathbb{R}^2$. Let α and β be two discrete probability measures on $\mathcal{X}$ defined by

$$\begin{cases} \alpha((1,0)) &= 0.5, \\ \alpha((0,1)) &= 0, \\ \alpha((1,2)) &= 0.5, \\ \alpha((2,1)) &= 0. \end{cases} \quad \text{and} \quad \begin{cases} \beta((1,0)) &= 0, \\ \beta((0,1)) &= 0.5, \\ \beta((1,2)) &= 0, \\ \beta((2,1)) &= 0.5. \end{cases}$$

With metric d, we will use the Euclidean norm which is a metric on $\mathbb{R}^2$. For $p = 2$, from that, we have our cost matrix C is

$$C = \begin{pmatrix} 0 & 2 & 2 & 2 \\ 2 & 0 & 2 & 2 \\ 2 & 2 & 0 & 2 \\ 2 & 2 & 2 & 0 \end{pmatrix}.$$

Solving the problem defined by Kantorovich formula, we obtain the optimal transport plan P^* is

$$P^* = \begin{pmatrix} 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.5 \\ 0 & 0 & 0 & 0 \end{pmatrix}.$$

Moreover, the distance between α and β as calculated by 2-Wasserstein metric is

$$\mathcal{W}_2 = L_C(\alpha, \beta) = \sqrt{\langle P^*, C \rangle_F} = \sqrt{2}.$$

For continuous case, we can generalize the Wasserstein distance as below.

Definition 3.2.3.4 *Let $p \geq 1$. For $\mu, \nu \in \mathcal{P}(\Omega)$ define the p-Wasserstein distance between μ and ν by*

$$W_p(\mu, \nu) := \left[\min \left\{ \int_{\Omega \times \Omega} |x - y|^p d\gamma : \gamma \in \pi(\mu, \nu) \right\} \right]^{\frac{1}{p}}.$$

This is also known as the p-Monge-Kantorovich metric, or the p-Earth-Mover metric.

When $p = 2$, we have the the 2-Wasserstein metric between μ and nu is defined as

$$W_2(\mu, \nu) := \left[\min \left\{ \int_{\Omega \times \Omega} |x - y|^2 d\gamma : \gamma \in \pi(\mu, \nu) \right\} \right]^{\frac{1}{2}}. \tag{3.2.9}$$

Entropy-Regularized Wasserstein Metric

In this subsection, we introduce an adaptation of Wasserstein metric that is designed for computational efficiency on geometric domains. Initially, we assume that the pairwise distance function $d(\cdot, \cdot)$ is given. Subsequently, we utilize heat kernels to relax this assumption.

The objective function for 2-Wasserstein distances, as shown in (3.2.9), is adjusted by incorporating an entropy term $H(\pi)$. This addition encourages more dispersed transportation plans π,

promoting a balanced distribution of mass across the space. By spreading out the transportation, the solution avoids concentrating mass in specific areas, resulting in a smoother and more stable transport plan. We then introduce *Entropy-Regularized Wasserstein Distance*.

$$\mathcal{W}_{2,\gamma}^2(\mu,\nu) = \inf_{\pi\in\Pi}\left[\iint_{M\times M} d(x,y)^2\,\pi(x,y)\,dx\,dy - \gamma H(\pi)\right] \tag{3.2.10}$$

3.2.4 Unbalanced Optimal Transport

Standard optimal transport (OT) strictly enforces mass conservation, which limits its use to true probability measures. The classical OT formulation then becomes sensitive to noise and small mass discrepancies. By relaxing the conservation requirement, which is represented in unbalanced optimal transport below, we can discard such outliers.

In what follows we consider a compact metric space X with distance d_X. We denote $\mathcal{C}(X)$ as the space of continuous functions with the supremum norm $\|f\|_\infty = \max_{x\in X}|f(x)|$ for $f \in \mathcal{C}(X)$. For $f \in C(X)$, we denote as

$$\int_X f\,d\alpha = \int_X f(x)\,d\alpha(x)$$

the pairing between functions $f \in C(X)$ and measures in the dual space $\alpha \in \mathcal{M}(X)$ (so that α can be identified as a linear form). A positive measure $\alpha \in \mathcal{M}_1^+(X)$ is such that

$$\int_X f\,d\alpha \geq 0$$

for all positive functions $f \geq 0$.

Csiszár $(\varphi-)$ divergences

Before introducing unbalanced optimal transport we need a notion of similarity that does not require strict mass conservation yet stays strictly positive when the two inputs differ.

Definition 3.2.4.1 *A* divergence $\mathcal{L}(\alpha,\beta)$ *between two positive Radon measures* $\alpha,\beta \in \mathcal{M}_+(\Omega)$ *is any functional such that*

1. $\mathcal{L}(\alpha|\beta) \geq 0$.
2. $\mathcal{L}(\alpha|\beta) = 0$ *if and only if* $\alpha = \beta$.

Unlike a metric, a divergence is not required to be symmetric nor to satisfy the triangle inequality—properties that are irrelevant once it is used as a penalty inside a larger optimization problem.

Definition 3.2.4.2 *Define an entropy function as a function* $\varphi : (0,\infty) \to [0,\infty]$, *which is convex, lower-semi-continuous function and such that* $\varphi(1) = 0$. *Denote its recession constant*

$$\varphi'_\infty = \lim_{z\to\infty}\frac{\varphi(z)}{z}$$

For $\alpha,\beta \in \mathcal{M}_+(\Omega)$, *consider the Lesbegue-Radon-Nikodym decomposition*

$$\alpha = \frac{d\alpha}{d\beta}\beta + \alpha^\perp, \quad \textit{with} \quad \alpha^\perp \perp \beta.$$

The Csiszár divergence (also called $\varphi-$divergence) is defined by

$$D_\varphi(\alpha\|\beta) \triangleq \int_X \varphi\left(\frac{d\alpha}{d\beta}(x)\right) d\beta(x) + \varphi'_\infty \int_X d\alpha^\perp(x).$$

One of the popular instances of $\varphi-$divergences are Kullback-Leibler divergence, which has been introduced in definition 3.1.2.14. In this case, $\varphi(p) = p\log p - p + 1$ and $\varphi' = +\infty$. We can rewrite the formula again as below

$$\mathrm{KL}(\alpha\|\beta) \triangleq \int_X \log\left(\frac{\mathrm{d}\alpha}{\mathrm{d}\beta}(x)\right) \mathrm{d}\alpha(x) - \int_X \mathrm{d}\beta + \int_X \mathrm{d}\alpha,$$

when $\alpha^\perp = 0$ (which corresponds to α as being absolutely continuous with respect to β, and is noted as $\alpha \ll \beta$), and $\mathrm{KL}(\alpha\|\beta) = +\infty$ otherwise.

Formulation

As mentioned above, balanced optimal transport only has a finite value when the two inputs have the same mass, specifically $\int_X d\alpha = \int_X d\beta$. In Kantorovich problem, the transport plan γ, and two marginal constraints α, β are imposed by $\int_{X\times Y} d\gamma = \int_X d\alpha = \int_Y d\beta$.

Over recent years, various extensions of optimal transport (OT) have been introduced to facilitate comparisons between arbitrary positive measures, drawing upon the diverse equivalent formulations of OT. One method was proposed by [35] which replace the hard constraint $(\pi_1)_\#\gamma = \alpha$ and $(\pi_2)_\#\gamma = \beta$ by the $\varphi-$divergences $D_\varphi((\pi_1)_\#\gamma\|\alpha)$ and $D_\varphi((\pi_2)_\#\gamma\|\beta)$. The approach was introduced in [23] specifically for the case where $D = \mathrm{KL}$. Our primary focus is on this setting due to its computational efficiency, which we elaborate on later section.

Definition 3.2.4.3 *The unbalanced optimal transport problem using Kullback-Leipler divergence penalty aims to optimize the objective function*

$$\min_{\pi\geq 0} \int_{X^2} c(x,y)\, d\pi(x,y) + \mathit{KL}(\pi_1\|\alpha) + \mathit{KL}(\pi_2\|\beta)$$

over transport plan $\pi \in \mathcal{M}^+(X\times Y)$ where $(\pi_1,\pi_2) = ((\pi_1)_\#\gamma, (\pi_2)_\#\gamma)$ are the plan's marginals.

In the discrete case, assume that the two measures are supported on finite point clouds

$$\alpha = \sum_{i=1}^n h_i\,\delta_{x_i}, \qquad \beta = \sum_{j=1}^m g_j\,\delta_{y_j}, \qquad h\in\mathbb{R}_+^n,\ g\in\mathbb{R}_+^m.$$

Let $C\in\mathbb{R}^{n\times m}$ be the *ground–cost matrix* with entries $C_{ij} = c(x_i,y_j) = \|x_i - y_j\|^p$. A discrete transport plan is an *unnormalised coupling matrix*

$$\Gamma = (\gamma_{ij})_{i,j} \in \mathbb{R}_+^{n\times m},$$

where γ_{ij} is the amount of mass sent from x_i to y_j. Its row and column sums are

$$\widehat{h} = \Gamma\mathbf{1}_m \in \mathbb{R}_+^n, \qquad \widehat{g} = \Gamma^\top\mathbf{1}_n \in \mathbb{R}_+^m,$$

written $\widehat{h}$, $\widehat{g}$ for later brevity. With Kullback-Leipler divergence, the *discrete unbalanced OT* objective value is

$$\mathcal{U}_{\lambda,\varphi}(h,g) = \min_{\Gamma\geq 0}\left[\langle C,\Gamma\rangle + \lambda\Big(\mathrm{KL}(\widehat{h}\,|\,h) + \mathrm{KL}(\widehat{g}\,|\,g)\Big)\right].$$

3.2.5 Dynamic Optimal Transport

Given two density functions $\rho_0(x) \geq 0$ and $\rho_T(x) \geq 0$ of $x \in \mathbb{R}^d$ and assume that

$$\int_{\mathbb{R}^d} \rho_0(x)dx = \int_{\mathbb{R}^d} \rho_T(x)dx = 1,$$

recall that L^2 Wasserstein distance between ρ_0 and ρ_T is defined by

$$d_2(\rho_0, \rho_T)^2 = \inf_M \int |M(x) - x|^2 \rho_0(x)dx,$$

where M is a map from $\mathbb{R}^d$ to $\mathbb{R}^d$. A numerical method was proposed in [8] to calculate L^2 Wasserstein distance, based on resetting of mass transfer problem into a continuum mechanics framework. We consider a time interval $[0, T]$ and analyze all sufficiently smooth time-dependent density and velocity fields, $\rho(t, x) \geq 0$, $v(t, x) \in \mathbb{R}^d$, subject to the continuity equation

$$\partial_t \rho + \nabla \cdot (\rho v) = 0 \tag{3.2.11}$$

for $0 < t < T$ and $x \in \mathbb{R}^d$, along with the initial and final conditions

$$\rho(0, \cdot) = \rho_0, \quad \rho(T, \cdot) = \rho_T. \tag{3.2.12}$$

We have the following proposition

Proposition 3.2.5.1 *The square of L^2 Wasserstein distance corresponds to the minimum value of the following quantity*

$$T \int_{\mathbb{R}^d} \int_0^T \rho(t, x)|v(x, t)|^2 dtdx,$$

taken over all pairs (ρ, v) that satisfy the continuity equation and boundary conditions described in equations (3.2.11) *and* (3.2.12)*.*

Proof. We prove the result using Lagrangian coordinates. Assume that the initial and final densities ρ_0 and ρ_T are compactly supported and bounded in $\mathbb{R}^d$. Let $\rho(t, x)$ and $v(t, x)$ be sufficiently smooth functions satisfying the continuity equation (3.2.11) and the boundary conditions (3.2.12). Define the Lagrangian map $X(t, x)$ by

$$X(0, x) = x, \quad \partial_t X(t, x) = v(t, X(t, x)). \tag{3.2.13}$$

Then, for any test function f, we have

$$\int_{\mathbb{R}^d} \int_0^T f(t, x)\rho(t, x)\, dx\, dt = \int_{\mathbb{R}^d} \int_0^T f(t, X(t, x))\rho_0(x)\, dt\, dx, \tag{3.2.14}$$

$$\int_{\mathbb{R}^d} \int_0^T f(t, x)\rho(t, x)v(t, x)\, dx\, dt = \int_{\mathbb{R}^d} \int_0^T \partial_t X(t, x) f(t, X(t, x))\rho_0(x)\, dt\, dx. \tag{3.2.15}$$

Using these, the kinetic energy becomes

$$\begin{aligned}
T \int_{\mathbb{R}^d} \int_0^T \rho(t, x)|v(t, x)|^2\, dx\, dt &= T \int_{\mathbb{R}^d} \int_0^T \rho_0(x)|v(t, X(t, x))|^2\, dx\, dt \quad \text{(by (3.2.14))} \\
&= T \int_{\mathbb{R}^d} \int_0^T \rho_0(x)|\partial_t X(t, x)|^2\, dt\, dx \quad \text{(by (3.2.13))} \\
&\geq \int_{\mathbb{R}^d} \rho_0(x)|X(T, x) - X(0, x)|^2\, dx \quad \text{(by Jensen's inequality)} \\
&= \int_{\mathbb{R}^d} \rho_0(x)|X(T, x) - x|^2\, dx \quad \text{(by (3.2.13) again)}
\end{aligned}$$

Since $X(T,x)$ must coincide with the optimal transport map $\nabla\Psi(x)$, we obtain

$$\int_{\mathbb{R}^d} \rho_0(x)|X(T,x)-x|^2\,dx \geq \int_{\mathbb{R}^d} \rho_0(x)|\nabla\Psi(x)-x|^2\,dx. \tag{3.2.16}$$

Equality holds when we choose

$$X(t,x) = x + \frac{t}{T}(\nabla\Psi(x)-x), \tag{3.2.17}$$

which corresponds to the velocity field

$$v(t,x) = \frac{\nabla\Psi(x)-x}{T}.$$

Using this, we define

$$\rho(t,x) = \rho_0\left(x + \frac{t}{T}(\nabla\Psi(x)-x)\right), \tag{3.2.18}$$

$$v(t,x) = \frac{\nabla\Psi(x)-x}{T}, \tag{3.2.19}$$

which corresponds to the pair (ρ,ν) defined by

$$\int f(t,x)\rho(t,x)\,dt\,dx = \int f\left(t, x + t\frac{\nabla\Psi(x)-x}{T}\right)\rho_0(x)\,dt\,dx, \tag{3.2.20}$$

$$\int f(t,x)\rho(t,x)\nu(t,x)\,dt\,dx = \int \frac{\nabla\Psi(x)-x}{T} f\left(t, x + t\frac{\nabla\Psi(x)-x}{T}\right)\rho_0(x)\,dt\,dx, \tag{3.2.21}$$

for all test function f. This pair (ρ,v) satisfies the continuity equation and boundary conditions, thus attaining the minimum in the proposition. □

Remark 3.2.5.2. After introducing a new variable, momentum $\mathbf{m} = \rho\mathbf{v}$, the problem can be reformulated as a convex optimization problem

$$\min_{\rho,\mathbf{m}} \frac{1}{2}\int_0^T \int_{[0,1]^2} \frac{|\mathbf{m}(t,s)|^2}{\rho(t,s)}\,ds\,dt \tag{3.2.22}$$

subject to

$$\partial_t\rho + \nabla\cdot(\mathbf{m}) = 0,$$
$$\rho(0,\cdot) = \rho_0, \quad \rho(T,\cdot) = \rho_T.$$

Chapter 4

Related Works

So far, the problem of shape interpolation has gained a lot of attention and is still an active researching field in modern computer graphics and computer vision today. Along with that, there are numerous approaches developed to address the problem efficiently. Researchers have explored various methodologies to try achieve the smoothest transitions between shapes, each with its own advantages and limitations. Techniques range from simple linear interpolation to more sophisticated methods such as optimal transport. Moreover, types of data to which those approaches tackle also varies from 2D (images) to 3D such (point clouds, meshgrids, ...). In this chapter, we will delve into these diverse techniques, providing a comprehensive overview of the current landscape in shape interpolation. By introducing many interpolating methodologies for specific types of data, examining the strengths as well as weaknesses of each approach, we aim to highlight the progress made in this field up to now and identify potential of optimal transport in shape interpolation task.

4.1 Shape Interpolation in 2D

Shape interpolation problem for 2D data often deals with images. There are many types of them which can range from common color RGB, RGBA or gray images to scanning, heating or UV ones. In term of shape interpolation, various methods have been developed to achieve effectiveness in such 2D data. In this section, we will explore those methodologies and discuss their advantages along with their major disadvantages.

The very first and most naive method for 2D data shape interpolation problem is linear interpolation. In mathematics, linear interpolation is a method of curve fitting using linear polynomials to construct new data points within the range of a discrete set of known data points. To be clearer, let us consider two points in $\mathbb{R}^2$ which is $x = (x_0, x_1)$ and $y = (y_0, y_1)$. Then, the intermediate points $z = (z_0, z_1)$ between those two points can be calculated using the following formula

$$\begin{cases} z_0 &= x_0 + t(y_0 - x_0), \\ z_1 &= x_1 + t(y_1 - x_1) \end{cases}$$

When coming to interpolation tasks with images, we can consider images as a collection of pixels which are actually points having values. Each type of image data such as RGB, RGBA or gray gives different dimensions of those values. However, the formula used for linear interpolation still works. An example of linear interpolation method on 2D data is given at Figure 4.1. Another variance of linear interpolation is bilinear interpolation method in which repeated linear interpolation process is performed, first in one direction and then again in

another direction.

Figure 4.1: An example of linear interpolation between two gray images.

Some use Fourier transform methods for shape interpolation. The Fourier transform will be applied to the parameterized contour function to decompose the shape into frequency components, resulting in a set of Fourier coefficients. Burget et al. [10] uses the Fourier descriptors of two shapes are blended linearly in the frequency domain to generate intermediate shapes. In [21], Feichtinger and Hans G mentions using Harmonic functions, which are solutions to Laplace's equation to interpolate between the Fourier coefficients of two shapes. After interpolation in Fourier space, the inverse Fourier transform is applied to reconstruct the intermediate shapes in the spatial domain.

Other approaches for 2D shape interpolation use level set representation. When shapes are implicitly represented as level set functions, these functions can be used for facilitating shape evolution, ensure smooth transitions and adherence to predefined shape constraints during interpolation by leveraging the same level set framework [41, 38, 48]. However, one of the biggest limitation of this approach is about computational cost. To overcome this, Malladi et al. [37] introduces a specialized algorithm named fast marching method for solving level set equations under the constraint of monotonically advancing fronts, it helps computation time for applications like shape interpolation, where monotonicity is often inherent.

Geodesic-based methods are also used in this problem. In geometry, a geodesic is a curve representing in some sense the shortest path or arc between two points in a manifold. In term of shape interpolation, geodesic can be considered as the shorted path connecting the source and target shape. That means if we take intermediate shapes in a geodesic between two shapes, we obtain a remarkably good transition between them. This intuition makes these approaches very compatible with the sense of shape interpolation. In geodesic-based methods, the biggest challenge for them is how to find the most effective geodesic between two given shapes. Avant et al. [4] introduces a geodesic estimation method within the framework of large deformation diffeomorphic metric mapping. It ensures that interpolations are both smooth and anatomically meaningful even in large deformation scenarios. There are also many different researches working on this tasks given efficient ways to compute the geodesic between two shapes [3, 51, 36]. Others use the observation that heat diffusion over time encodes information about the underlying geodesic distances. Crane et al. [14] takes advantage of that observation. By solving the heat equation and analyzing the resulting heat distribution, geodesic distances can be inferred.

Recently, with the rapid development of deep learning in which large-scale and effective neural networks take major roles, neural network-based shape interpolation cannot be out of the race. In intuition of generating models, some proposes utilizing encoding and decoding process and perform interpolation task on latent space. This interpolated representation will go through decoder to generate the intermediate shape. Those methods are applied in [55, 40, 22]. With the same intuition, generative adversarial networks (GAN) are suitable and in fact, widely used in the field of shape interpolation problem by leveraging them to learn a rich latent representation

of shapes. Instead of being represented in original forms, shapes can be represented as implicit functions, such as signed distance functions (SDFs) or occupancy fields, parameterized by a neural network. These neural field representations can be then mapped into some latent spaces to perform interpolating process. Yang et al. [57] proposes using neural field representations on multiple geometry processing tasks including shape interpolation. Their results show that this approach is extremely potential. Besides, other works also give optimistic views about neural field representation in shape interpolation tasks [7, 12]. Unsupervised learning is also used to automatically learn the structure and features of shapes from data, those features will be then mapped in some compact latent spaces for interpolation. By applying this, the latent space generalizes to unseen shapes, enabling interpolation beyond the training data. Some works on this approaches can be listed out as [26, 6, 49].

4.2 Shape Interpolation in 3D

In this section, we will next explore methodologies for interpolating 3D shape data.

Like 2D data shape interpolation problem, similar approaches can be applied to the cases of 3D data as well. Level sets methods are used in [39, 20, 32]. Some use Fourier transform for 3D data shape interpolation [28]. Takeda et al. [53] and Duan et al. [18] propose methods to measure 3D shape using Fourier transform, they achieve high sensitivity in capturing 3D shapes and facilitate effective interpolation.

There are multiple methods for 3D shape data in point clouds, which are called point cloud based methods. Zheng et al. [61] introduces NeuralPCI, an end-to-end 4D spatio-temporal neural field designed for 3D point cloud interpolation. Rakotosaona et al. [45] present a learning-based method for interpolating 3D shapes represented as point clouds by constructing a dual encoding space. Cat et al. [11] addresses point cloud completion by leveraging both extrapolation and interpolation techniques which focuses on generating complete 3D shapes from partial inputs. With 3D shape data in mesh, mesh-based interpolation methods are suitable. Some works can be listed out as [56, 46].

Coming to geodesic-based methods, the works of Avant et al. [4] which introduces a geodesic estimation method within the framework of large deformation diffeomorphic metric mapping can also be used for 3D shape data. Moreover, other methods give great results in term of 3D shape data as well [3, 51, 36]. Also, heat flow based approach such as Crane et al. [14] can be used to infer geodesic for two 3D shape data in shape interpolation problem.

Neural network based approaches are well adapted to this problem. Some use neural networks for mapping 3D shape data into some latent spaces for interpolating process, then inversely transform the interpolated representation into original space. Using this intuition, Yang et al. [58] proposes DSG-Net for learning separate latent spaces for structure and geometry which allows controlled shape interpolation. Also, Rakotosaona et al. [45] introduces a learning-based method for interpolating 3D point clouds by navigating a dual latent space. Besides, neural field representation can be used for 3D shape interpolation problem. One remarkable work on this idea is of Park et al. [42]. They introduce DeepSDF, a deep learning framework that represents 3D shapes as continuous signed distance functions. By using this, smooth transitions between different 3D shape can be facilitated.

4.3 Optimal Transport in Shape Interpolation

Optimal transport problem deal with finding an optimal transporting strategy so that the total transporting effort is minimized. Given two measures, optimal transport problem will find an optimal transport plan P to minimize the total transport cost. In practice, shapes can be thought as the form of some measures. For example, an gray image with each pixel whose a value stands for its color can be represented as a measure γ. The measure $\gamma(i, j)$ will return the value of the i^{th} row and j^{th} column pixel of give image. Or in case of 3D shape represented by a point cloud, a measure $\gamma(i)$ will return the mass of i^{th} point if masses are assigned to point cloud to express the density. Therefore, optimal transport based approaches can be naturally used for the shape interpolation problem.

Hug et al. [31] uses optimal transportation theory for image interpolation, presenting a method called dynamic optimal transport that minimize kinetic energy under mass conservation constraints to achieve smooth transitions between images. Using the same intuition, Feng et al. [22] introduces the path energy term which represents the least kinetic energy through current geodesic. By adding that path energy term to the loss value which is used to trained the autoencoder, smooth transitions between images can be constructed.

Not directly dealing with the shape interpolation problem, the following works try to explore efficient ways to compute the best geodesic between two measures in optimal transport problem. Using this geodesic and applying geodesic based methods in shape interpolation problem, intermediate shapes can be constructed. In [9] and [22], authors use dynamic optimal transport theory to estimate the best geodesic between two shapes by minimizing the kinetic energy through it. This estimated geodesic then will be used to generate transitions between two shapes. Another approach is to compute the transport plan to generate such geodesic. In optimal transport problem, calculating an optimal transport plan is extremely consuming. To overcome this, many works have proposed efficient methods for fast computing that value. Cuturi et al. [16] introduces Sinkhorn distance and Sinkhorn iteration algorithm for light speed computing transport plan for the optimal transport problem. Solomon et al. [52] proposes a method using iterative kernel convolutions to construct optimal transport plan which is based on Sinkhorn iteration. In case of semi-discrete optimal transport, Herrou et al. [29] presents an alternating algorithm utilizing semi-discrete optimal transport that computes a semi-discrete optimal transport map between two shapes.

Chapter 5

Methods

5.1 Sinkhorn Distance and Sinkhorn Algorithm

As introduced above, the optimal transport based geodesic method is a compatible approach for shape interpolation problem. By solving the optimal transport optimization problem and finding an optimal transport plan P, we can use that P to compute the intermediate shapes between two input shapes which lies on a geodesic in Wasserstein space. However, calculating such an optimal transport plan P is very resource and time-consuming. For example, let our two input shapes be two point clouds with each shape having about 100000 points in it. Then we know that our transport plan will have the dimension as 100000×100000. In worst case, working directly on this optimization problem using the best algorithm still has the complexity of $O(n^3 \log n)$ in which n is the dimension of the transport plan P in general [43]. To overcome this difficulty, Cuturi and Marco [16] proposes an algorithm for fast computing optimal transport plan P called Sinkhorn's algorithm. In this section, we will introduce and explore how this algorithm is constructed.

Now, we recall about Kullback-Leibler divergence in Section 3.1.2. Let $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ be a measurable set. Let $P \in \mathbb{R}_+^{n \times n}$ and $Q \in \mathbb{R}_+^{n \times n}$ be two transport plans between any two pairs of discrete probability measures on $\mathcal{X}$. Indeed, they can be identified with a joint probability for two discrete probability measures such that $P(x_i, x_j) = P_{i,j}$ in which $i, j = 1, \ldots, n$. We have the Kullback-Leibler divergence between P and Q is

$$KL(P \parallel Q) = \sum_{i,j} P_{i,j} \log \frac{P_{i,j}}{Q_{i,j}}.$$

Also, we define the entropy h of P as

$$h(P) = -\sum_{i,j} P_{i,j} \log P_{i,j}.$$

Let α and β be two discrete probability measures on $\mathcal{X}$. Let a and b be two vectors defined by

$$a = (\alpha(x_1), \ldots, \alpha(x_n))^\top$$

and

$$b = (\beta(y_1), \ldots, \beta(y_m))^\top.$$

Now, we introduce a new notation $U(\alpha, \beta)$ defined by

$$U(\alpha, \beta) = \{P \in \mathbb{R}_+^{n \times m} : P 1_{\mathbb{R}^m} = a, P^\top 1_{\mathbb{R}^n} = b\}.$$

In fact, $U_{\alpha,\beta}$ is the convex feasible set of the Kantorovich formula for optimal transport problem between α and β. Note that $ab^\top$ belongs to $U_{\alpha,\beta}$. Using Kullback-Leibler divergence, we can define another convex set $U_\gamma(\alpha,\beta)$ which is

$$U_\gamma(\alpha,\beta) = \{P \in U(\alpha,\beta) : KL(P \parallel ab^\top) \leq \gamma\}. \tag{5.1.1}$$

We can define a metric using this set.

Definition 5.1.0.1 (Sinkhorn distance) *Let $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ be a measurable set. Let α and β be two discrete probability measures on $\mathcal{X}$. Let $C \in \mathbb{R}_+^{n\times n}$ be a cost matrix. Then*

$$\mathcal{S}_{C,\gamma}(\alpha,\beta) = \inf_{P\in U_\gamma(\alpha,\beta)} \langle P, C\rangle_F$$

defines the Sinkhorn distance *on the set of discrete probability measures on $\mathcal{X}$.*

We can rewrite Equation (5.1.1) as

$$U_\gamma(\alpha,\beta) = \{P \in U(\alpha,\beta) : h(P) \geq h(a) + h(b) - \gamma\}$$

in which $h(a)$ and $h(b)$ define entropies on vector a and b as

$$h(a) = -\sum_i a_i \log a_i \quad \text{and} \quad h(b) = -\sum_j b_j \log b_j.$$

Now, we introduce dual Sinkhorn divergence.

Definition 5.1.0.2 (Dual Sinkhorn divergence) *Let $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ be a measurable set. Let α and β be two discrete probability measures on $\mathcal{X}$. Let $C \in \mathbb{R}_+^{n\times n}$ be a cost matrix. Then, for $\lambda > 0$,*

$$\mathcal{S}_C^\lambda(\alpha,\beta) = \langle P^\lambda, C\rangle_F$$

defines the dual Sinkhorn divergence *on the set of discrete probability measures on $\mathcal{X}$ in which*

$$P^\lambda = \operatorname*{arg\,inf}_{P\in U(\alpha,\beta)} \langle P, C\rangle_F - \frac{1}{\lambda} h(P). \tag{5.1.2}$$

By duality theory, we have that to each γ corresponds a $\lambda \in [0,\infty)$ such that $\mathcal{S}_C^\lambda(\alpha,\beta) = \mathcal{S}_{C,\gamma}(\alpha,\beta)$ holds for the pair (α,β). Moreover, computing $\mathcal{S}_{C,\gamma}(\alpha,\beta)$ can be carried out by computing $\mathcal{S}_C^\lambda(\alpha,\beta)$ with increasing values of λ until $h(P^\lambda)$ reaches $h(a) + h(b) - \gamma$ [16]. We have from the objective function of (5.1.2) that

$$\begin{aligned}
\langle P, C\rangle_F - \frac{1}{\lambda} h(P) &= \textstyle\sum_{i,j} P_{i,j} C_{i,j} + \frac{1}{\lambda} \sum_{i,j} P_{i,j} \log P_{i,j} \\
&= \textstyle\sum_{i,j} P_{i,j} C_{i,j} + \frac{1}{\lambda} P_{i,j} \log P_{i,j} \\
&= \textstyle\sum_{i,j} P_{i,j} \left(C_{i,j} + \frac{1}{\lambda} \log P_{i,j} \right) \\
&= \textstyle\frac{1}{\lambda} \sum_{i,j} P_{i,j} \left(\lambda C_{i,j} + \log P_{i,j} \right) \\
&= \textstyle\frac{1}{\lambda} \sum_{i,j} P_{i,j} \left(\log \left(e^{\lambda C_{i,j}} P_{i,j} \right) \right).
\end{aligned}$$

Let $K \in \mathbb{R}_+^{n\times n}$ be a matrix defined by

$$K_{i,j} = e^{-\lambda C_{i,j}}. \tag{5.1.3}$$

Then, the objective function of (5.1.2) can be rewritten as

$$\langle P, C\rangle_F - \frac{1}{\lambda}h(P) = \frac{1}{\lambda}\sum_{i,j} P_{i,j}\log\frac{P_{i,j}}{K_{i,j}} = \frac{1}{\lambda}KL(P \parallel K).$$

In fact, we have the following lemma.

Lemma 5.1.0.3 *Let K be the matrix defined by (5.1.3). For $\lambda > 0$, the solution P^λ of the Problem (5.1.2) is unique and has the form $P^\lambda = \mathbf{diag}(u)\ K\ \mathbf{diag}(v)$ in which u and v are two non-negative vectors of $\mathbb{R}^n$. Note that $\mathbf{diag}(u)$ is the diagonal matrix whose entries in main diagonal is vector u.*

By this lemma, now we can use *Sinkhorn fixed point iteration* to compute two vector u and v using following formulas

$$u \leftarrow \frac{a}{Kv} \quad \text{and} \quad v =\leftarrow \frac{b}{K^\top u}.$$

Finally, P^λ which is a approximate value of optimal transport plan P is calculated by

$$P^\lambda = \text{diag}(u)\ K\ \text{diag}(v).$$

The relationship between P^λ and optimal transport plan P^* by Kantorovich formulation is illustrated in Figure 5.1. As we can see, the path of P^λ starts at $rc^\top$ (which is actually $ab^\top$ in our formulation of Sinkhorn algorithm) and going toward P^* when we modify the regularization param λ.

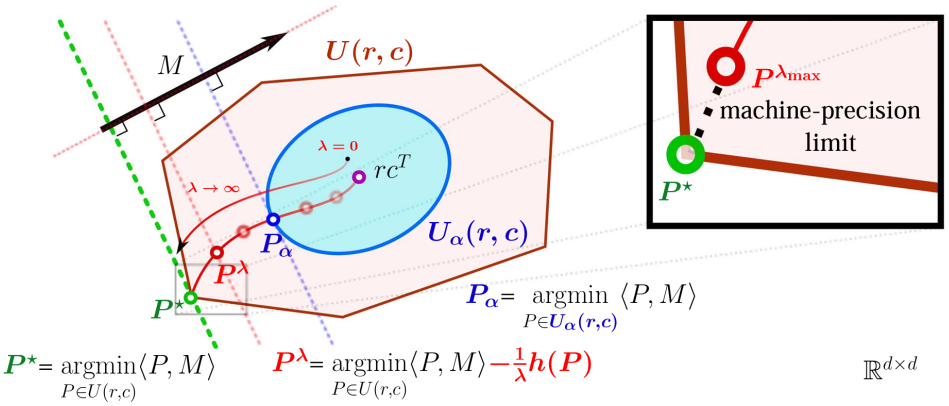


Figure 5.1: Illustration of relationship between P^λ and P^*. Larger value of λ makes P^λ closer to P^*. In contrast, smaller value of λ makes P^λ closer to $rc^\top$ which is $ab^\top$ in our formulation (Source: [16]).

The detailed steps of Sinkhorn algorithm are described in Algorithm 1 below.

Algorithm 1 Sinkhorn algorithm

Input: a, b, C, λ
$K \leftarrow \exp(-\lambda C)$
$u \leftarrow \text{init}(n)$
$v \leftarrow \text{init}(m)$
while not `stopping_criteria` **do**
    $u \leftarrow a/Kv$
    $v \leftarrow b/K^\top u$
end while
$P^\lambda \leftarrow \text{diag}(u)\ K\ \text{diag}(v)$

Some stopping criteria can be listed out as the pre-defined maximum number of iterations, or the difference between two consecutive iterations of u and/or v are below some thresholds.

5.2 Shape Interpolation Using Kantorovich Formulation

As we know, given the two shapes α and β, by solving the optimal transport Kantorovich problem (see Section 3.2.2) associated to these two shapes, we can find an optimal transport plan P. Using this optimal transport plan P, a geodesic between α and β can be constructed, and by this we can generate the transitions between them.

In this method, we will try two approaches to compute such an optimal transport plan P. The first one is using convex optimization solvers which are available in multiple libraries of many programming languages. The second one uses Sinkhorn's algorithm, which is introduced in Section 5.1 of this chapter. The experiments using this method will be conducted on both 2D and 3D shape data.

5.3 Dynamic Optimal Transport Variational Autoencoder

In this part, we discuss how to apply dynamic optimal transport which is introduced in Section 3.2.5 to variational autoencoder (VAE).

For VAE, it is a type of generative model that combines principles from deep learning and probabilistic inference to learn efficient latent representations of data. This probabilistic framework allows VAE not only to compress data but also to generate new samples by sampling from the learned latent space. For dynamic optimal transport, it tends to minimize the total cost (which is actually kinetic energy) when traveling from the source shape to the target shape. In scope of this project, when combining them together, we create a *dynamic optimal transport variational autoencoder (DOT VAE)* to solve the shape interpolation problem in the sense of 2D shapes (images) and 3D shapes (point clouds).

The original DOT VAE for 2D shapes which are images are proposed in [22]. We will enhance it to create two more versions of DOT VAE for 3D shapes which will be introduced in the following.

5.3.1 Voxelization

In the scope of our project, *voxelization* is the process of converting a 3D shape which are point clouds, into sets of volumetric elements called voxels - the 3D equivalent of pixels. Each voxel represents a small *cube* of space in a regular 3D grid and contains information about the presence of the points in the point clouds within that region. Voxelization is commonly used in computer graphics. It allows complex shapes to be represented in a structured, grid-based format that makes it easier to be processed. An example of voxelization is given in Figure 5.2.

Figure 5.2: An example of conducting voxelization on bunny 3D shape (red one). As we can see, if we choose the size of grids in result voxels well enough, the voxelized 3D shapes still keep all the important features of the input 3D shapes (rightmost one) in a more well-structured format.

The voxelizing process used in our proposed methods starts by normalizing the input 3D shape (point cloud) so that it is contained within a cube space whose size is $[0, 1] \times [0, 1] \times [0, 1]$. Let n be the dimension of the output voxel. By this, we know that our output voxel will have in total n^3 elements. We initialize all the elements of our output voxel as 0. Now, we iterate through all points in the normalized input point cloud to determine which elements of the output voxel contain them. For every element that contains point of the normalized input point cloud, we assign the value 1 to it. Finally, we obtain the output voxel that is a structured, grid-based and more formal representation of the input 3D shape.

Voxelization will be used to transform the input 3D shape into a formal input format for our DOT VAE in 3D shape interpolation problem.

5.3.2 Dynamic Optimal Transport Variational Autoencoder for 2D Shape Interpolation

In this method, we will combine VAE with dynamic optimal transport to construct a dynamic optimal transport variational autoencoder (DOT VAE) by adding a *path energy* term to the loss value during training process. This path energy term is the minimal kinetic energy value associated with the current geodesic constructed by the VAE. This value is achieved by optimizing the Equation (3.2.22) through the momentum term $\mathbf{m}$.

When we obtain the VAE for the problem, interpolation process can be conducted by applying linear interpolation on latent vectors of two inputs shapes. In particular, let α and β be our two input shapes and $\rho_{\alpha,\beta}$ be the geodesic between α, β which is constructed by the VAE. Firstly, we compute the latent vectors $\widetilde{\alpha}$ and $\widetilde{\beta}$ of α and β through the encoding process of the VAE. Secondly, we compute the interpolated latent vector x_t at time slice t by using linear

interpolation on the two latent vectors of α and β, that means

$$x_t = t\widetilde{\beta} + (1-t)\widetilde{\alpha}$$

in which $t \in [0, 1]$. The interpolated shape $\rho_{\alpha,\beta}(t)$ at time slice t of two input shapes α and β is then computed by decoding x_t, that means

$$\rho_{\alpha,\beta}(t) = \text{decode}(x_t)$$

in which $t \in [0, 1]$.

Providing the interpolation process, we can describe the training process of this method. In training process, in each epoch, we will firstly choose many pairs of shapes from the training set. With each pair of shapes, let α and β be these two shapes, we will generate n intermediate shapes on the current geodesic $\rho_{\alpha,\beta}$ in which n is the number of time steps on the interval $[0, 1]$. In other words, if we have n time steps which are $\{t_1 = 0, t_2, t_3, \ldots, t_n\}$ and $t_{n+1} = 1$, we will try to generate n intermediate shapes which are $\rho_{\alpha,\beta}(t_i)$ for every $i = 1, \ldots, n$. Those intermediate shapes are computed using the interpolation process we introduce previously. Having all the intermediate shapes, we now calculate the path energy term which is indeed the least total kinetic energy on $\rho_{\alpha,\beta}$ by optimizing the Equation (3.2.22) through the momentum term $\mathbf{m}$. This value will be added to the loss value which is primarily used for evaluating the reconstructing process of the VAE. By this, both reconstructing and generalizing the best geodesic of shapes are optimized. Therefore, our final VAE will have both abilities of good reconstructing and interpolation. The Figure 5.3 gives a good overview of the architecture and mechanism of the DOT VAE when dealing with 2D shapes (images).

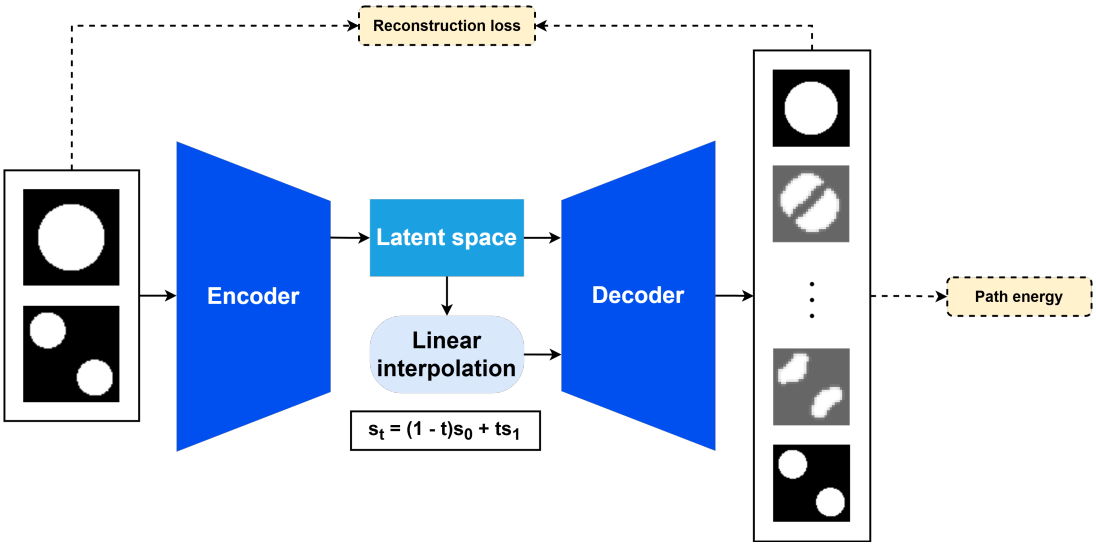


Figure 5.3: Illustration of interpolation and training process in DOT VAE. In DOT VAE, source and target 2D shapes are mapped into a latent space which is learnt through training process. Then, those mapped latent vectors (which represent the input 2D shapes in latent space) will be linearly interpolated to construct intermediate latent vectors. Those vectors are again decoded into intermediate images to form transitions from source to target 2D shapes. They are also used to compute path energy term which is added to loss function.

The detailed steps of VAE interpolation and training process are described in Algorithm 2 and Algorithm 3 below. Note that in the algorithms, VAE is the autoencoder, L is the loss function, DOT is the function to compute the least total kinetic energy on the current geodesic, λ is the energy weight used to regularize the path energy term, ϵ is the learning rate.

Algorithm 2 Interpolation process of DOT VAE

Input: $\mathrm{VAE}, \alpha, \beta, t$
$\widetilde{\alpha} \leftarrow \mathrm{VAE.encode}(\alpha)$
$\widetilde{\beta} \leftarrow \mathrm{VAE.encode}(\beta)$
$x_t \leftarrow t\widetilde{\beta} + (1-t)\widetilde{\alpha}$
$\rho_{\alpha,\beta}(t) \leftarrow \mathrm{VAE.decode}(x_t)$

Algorithm 3 Training process of DOT VAE

Input: $\mathrm{VAE}, n, (X_{\text{train}}, y_{\text{train}}), \mathrm{L}, \mathrm{DOT}, \lambda, \epsilon$
$T \leftarrow \{t_1 = 0, t_2, \ldots, t_n\}$ in which $t_{n+1} = 1$
for each epoch **do**
  for some (i, j) **in** $\mathrm{id}(X_{\text{train}})$ **do**
    for t **in** T **do**
      $\rho_{(X_{\text{train}})_i,(X_{\text{train}})_j}(t) \leftarrow \mathrm{interpolation\_process}(\mathrm{VAE}, (X_{\text{train}})_i, (X_{\text{train}})_j, t)$
    end for
  end for
  $\hat{y}_{\text{train}} \leftarrow \mathrm{VAE}(X_{\text{train}})$
  $\mathrm{loss} \leftarrow \mathrm{L}(\hat{y}_{\text{train}}, y_{\text{train}}) + \lambda \sum_{i,j} \mathrm{DOT}(\rho_{(X_{\text{train}})_i,(X_{\text{train}})_j})$
  $\mathrm{optimize}(\mathrm{VAE}, \mathrm{loss}, \epsilon)$
end for

5.3.3 Dynamic Optimal Transport Multiple Channels Variational Autoencoder for 3D Shape Interpolation

We have introduced how we adapt DOT VAE for the 2D shape interpolation problem on images. In this part, we introduce a new version of DOT VAE that can be applied for 3D shape interpolation problem called *dynamic optimal transport multiple channels variational autoencoder (DOT-MC VAE)*.

Let α and β be our two inputs 3D shapes. In this method, we need to voxelize those shapes using method introduced in Section 5.3.1. Let n be the dimension of both two output voxels obtained when conduct voxelization on two input 3D shapes. Let $\hat{\alpha}$ be the output voxel of α and $\hat{\beta}$ be the output voxel of β. We know that, indeed, $\hat{\alpha}$ and $\hat{\beta}$ can be considers as two images whose size $n \times n$ with n channels. This intuition gives us an idea to enhance the original DOT VAE to be able to work on 3D input shapes. The term "multiple channels" in DOT-MC VAE is derived from this.

Let n be the dimension of all generated voxels. Given the training set which consists of input voxels obtained by voxelizing the original 3D shape training set, we start the training process by applying the DOT VAE individually on corresponding $n \times n$ images of voxels in training set on each channel. Also, in the interpolation process, we can do the same procedure as in original DOT VAE on corresponding $n \times n$ images of voxels in test set on each channel. After combining those result images together to form a voxel, the intermediate shape of the transition between source and target shape is obtained. The Figure 5.4 gives us an overview of DOT-MC VAE method.

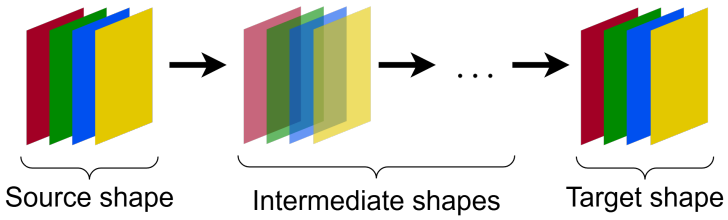


Figure 5.4: An overview of DOT-MC VAE method. The voxelized input 3D shapes are considered as multiple channels images. Now, we conduct original DOT VAE of 2D shapes on corresponding channels which are indeed 2D images of those voxelized input 3D shapes. The interpolation process is conducted by combining all the intermediate images of all channels to form complete voxelized intermediate 3D shapes. However, this method only considers mass flows among corresponding channels not across different channels.

However, this method only considers the mass flow among elements of each channel in the input voxels but not the mass flow among the elements throughout the channels. Thus, it seems not to satisfy the minimum kinetic energy in the shape interpolation problem.

5.3.4 Dynamic Optimal Transport Variational Autoencoder for 3D Shape Interpolation

We also develop a new version of DOT VAE to deal with the limitation of DOT-MC VAE introduced in Section 5.3.3. In this method, voxelization is also applied for the input 3D shape data. Also, this method will work on the whole input voxels themselves not just on individual channel of the input voxels. The Figure 5.5 gives us a good illustration of this method.

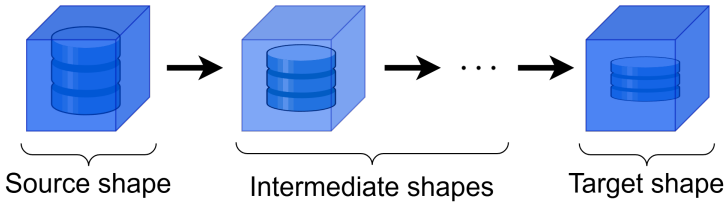


Figure 5.5: Core idea of our proposed method DOT VAE for 3D shapes. The input 3D shapes are first voxelized before conducting training or interpolation process. The big challenge is how to discretize and compute gradient values for those voxelized input 3D shapes.

The following parts show how we discretize the problem and compute the gradients for optimizing process. Note that the whole procedures of training and interpolation process given in Algorithm 3 and Algorithm 2 are remained.

Discretization

In this part, we will discuss about how we discretize the path energy term with staggered grid scheme in 3D.

Consider the time to be discreted into $\{0, 1, \cdots, T\}$ and space $\Omega = [0, 1]^3$ into a uniform $n \times n \times n$ grid. At each discrete time $t = 0, 1, \cdots, T$, we have the density $\rho_{\alpha,\beta}(t)$ (denoted as $\rho^t_{\alpha,\beta}$). We denote by $\rho_t \in \mathbb{R}^{n\times n\times n}$ the density at time step $t \in \{0, 1, \ldots, T\}$. The momentum (or flux) field $\mathbf{m}(t) = (m^1_t, m^2_t, m^3_t)$ (denoted as $\mathbf{m}_t$) is discretized on a *staggered grid*, where each component m^α_t is defined on the corresponding set of cell faces aligned with axis α. Specifically,

- $m^1_t \in \mathbb{R}^{(n+1)\times n\times n}$ stores the x-directed momentum on faces normal to the x-axis,

- $m_t^2 \in \mathbb{R}^{n\times(n+1)\times n}$ stores the y-directed momentum on faces normal to the y-axis,
- $m_t^3 \in \mathbb{R}^{n\times n\times(n+1)}$ stores the z-directed momentum on faces normal to the z-axis.

Using the staggered grid discretization scheme, the divergence operator associated with a vector field m_t in linear constraint of problem is defined as

$$(\nabla\cdot\mathbf{m})_{i,j,k} = \left(m^1_{t,i+1,j,k} - m^1_{t,i,j,k}\right) + \left(m^2_{t,i,j+1,k} - m^2_{t,i,j,k}\right) + \left(m^3_{t,i,j,k+1} - m^3_{t,i,j,k}\right).$$

We also let $b_t = -\dfrac{\partial(\rho_{\alpha,\beta})}{\partial t} = \rho_{\alpha,\beta}(t) - \rho_{\alpha,\beta}(t+1)$, the linear constraint of problem 4.2 in discretization becomes

$$\nabla\cdot(\mathbf{m}_t) = b_t, t = 0, 1, 2, \cdots, T-1.$$

The weight vector $\mathbf{w}(t) = (w_t^1, w_t^2, w_t^3)$ is then determined via midpoint averaging

$$\begin{aligned}
w_t^1(i,j,k) &= \frac{2}{\rho^t_{\alpha,\beta}(i,j,k) + \rho^t_{\alpha,\beta}(i+1,j,k)},\\
w_t^2(i,j,k) &= \frac{2}{\rho^t_{\alpha,\beta}(i,j,k) + \rho^t_{\alpha,\beta}(i,j+1,k)},\\
w_t^3(i,j,k) &= \frac{2}{\rho^t_{\alpha,\beta}(i,j,k) + \rho^t_{\alpha,\beta}(i,j,k+1)}.
\end{aligned}$$

After flattening the vectors $\mathbf{m}$ and $\mathbf{w}$, problem (4.2) becomes

$$J(\rho) = \min_{\mathbf{m}} \sum_{t=0,1,2,\ldots,T-1} \mathbf{m}_t^\top \mathrm{Diag}(\mathbf{w}_t)\mathbf{m}_t$$

such that

$$\nabla\cdot(\mathbf{m}_t) = b_t, t = 0, 1, 2, \cdots, T-1$$

where $b_t = p_t - p_{t+1}$. This is a quadratic problem with linear constraint, and its optimality condition (KKT condition) is given by

$$\begin{bmatrix} \mathrm{Diag}(\mathbf{w}_t) & \nabla\cdot^\top \\ \nabla\cdot & 0 \end{bmatrix} \begin{bmatrix} \mathbf{m}_t \\ \lambda_t \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{b}_t \end{bmatrix}, t = 0, 1, 2, ..., T-1,$$

where λ_t is the Lagrange multiplier. From the first equation, we have

$$\mathbf{m}_t = \mathrm{Diag}(\mathbf{w}_t)^{-1}\nabla(\lambda).$$

Substitute to the second equation, we have

$$\nabla\cdot\left(\mathrm{Diag}(\mathbf{w}_t)^{-1}\nabla(\lambda)\right) = b_t,$$

which is equivalent to

$$\nabla(\lambda) = \mathrm{Diag}(\mathbf{w}_t)(\nabla\cdot)^{-1}b_t.$$

Rename the density $\rho_{\alpha,\beta}(t)$ into $\rho(t)$, the objective function $J(\rho)$ becomes

$$\begin{aligned}
J(\rho) &= \min_{\mathbf{m}} \sum_{t=0,1,2,\ldots,T-1} \left[\mathrm{Diag}(\mathbf{w}_t)^{-1}\nabla(\lambda)\right]^\top \mathrm{Diag}(\mathbf{w}_t)\left[\mathrm{Diag}(\mathbf{w}_t)^{-1}\nabla(\lambda)\right]\\
&= \min_{\mathbf{m}} \sum_{t=0,1,2,\ldots,T-1} \nabla(\lambda)^\top \mathrm{Diag}(\mathbf{w}_t)^{-1}\mathrm{Diag}(\mathbf{w}_t)\left[\mathrm{Diag}(\mathbf{w}_t)^{-1}\nabla(\lambda)\right]\\
&= \min_{\mathbf{m}} \sum_{t=0,1,2,\ldots,T-1} \nabla(\lambda)^\top \left[\mathrm{Diag}(\mathbf{w}_t)^{-1}\nabla(\lambda)\right]\\
&= \min_{\mathbf{m}} \sum_{t=0,1,2,\ldots,T-1} \left[\mathrm{Diag}(\mathbf{w}_t)(\nabla\cdot)^{-1}b_t\right]^\top \mathrm{Diag}(\mathbf{w}_t)^{-1}\mathrm{Diag}(\mathbf{w}_t)(\nabla\cdot)^{-1}b_t\\
&= \min_{\mathbf{m}} \sum_{t=0,1,2,\ldots,T-1} b_t^\top \left(\nabla\cdot^\top\right)^{-1}\mathrm{Diag}(\mathbf{w}_t)\mathrm{Diag}(\mathbf{w}_t)^{-1}\mathrm{Diag}(\mathbf{w}_t)(\nabla\cdot)^{-1}b_t\\
&= \min_{\mathbf{m}} \sum_{t=0,1,2,\ldots,T-1} b_t^\top \left(\nabla\cdot^\top\right)^{-1}\mathrm{Diag}(\mathbf{w}_t)(\nabla\cdot)^{-1}b_t.
\end{aligned}$$

Therefore, the path energy can be written as

$$J(p) = \min_{\mathbf{m}} \sum_{t=0,1,2,...,T-1} b_t^\top \left(\nabla \cdot \text{Diag}(\mathbf{w}_t)^{-1} \nabla \cdot^\top\right)^{-1} b_t \tag{5.3.1}$$

by assuming $\mathbf{w}_t$ are positive.

Gradient Computation

In this part, we will discuss how to calculate the first-order gradient of the path energy, which is crucial in the backpropagation of neural network training.

First, we denote $\nabla \cdot \text{Diag}\left(\mathbf{w}_t\right)^{-1} \nabla \cdot^\top$ as A_t, which is a reweighted poison operator. There are T sparse linear system to solve $J(p)$

$$A_t y = b_t, \quad t = 0, 1, ..., T-1. \tag{5.3.2}$$

We now introduce the formula of first-order gradient in 3D.

Theorem 5.3.4.1 *The first order gradient of the path energy defined in (5.3.1) at time $t = 0, 1, 2, ..., T-1$ is given by*

$$\left(\frac{\partial J}{\partial \rho_t}\right)_{i,j} = -\frac{1}{6} \sum_{(l,m,n) \in \mathcal{O}_{i,j,k}} \left(y_{t,l,m,n} - y_{t,i,j,k}\right)^2 + 2 y_{t,i,j,k},$$

where $\mathcal{O}_{i,j,k}$ is the connected neighbor of (i,j,k), i.e., $\{(i-1,j,k),(i+1,j,k),(i,j-1,k),(i,j+1,k),(i,j,k-1),(i,j,k+1)\}$ and $y_t \in \mathbb{R}^{n \times n \times n}$ is the solution of (5.3.2).

Proof. At time t, let $\mathbf{u}$ represent the elementwise inverse of $\mathbf{w}$, that is, each element of $\mathbf{u}$ is reciprocal of the corresponding element of $\mathbf{w}$. Applying the chain rule, we have

$$\begin{aligned} \frac{\partial J}{\partial \rho_t} &= \frac{\partial J}{\partial \mathbf{u}} \cdot \frac{\partial \mathbf{u}}{\partial \rho_t} + \frac{\partial J}{\partial b_t} \cdot \frac{\partial b_t}{\partial \rho_t} \\ &= \frac{\partial J}{\partial \mathbf{u}} + \frac{\partial J}{\partial b_t}. \end{aligned}$$

Now, we have

$$\begin{aligned} \frac{\partial J}{\partial \mathbf{u}} &= \frac{\partial}{\partial \mathbf{u}} b_t^\top \left(\nabla \cdot \text{Diag}(\mathbf{w}_t)^{-1} \nabla \cdot^\top\right)^{-1} b_t \\ &= \frac{\partial}{\partial \mathbf{u}} b_t^\top A_t^{-1} b_t = b_t^\top \left(\frac{\partial}{\partial \mathbf{u}} A_t^{-1}\right) b_t \\ &= -b_t^\top A_t^{-1} \left(\frac{\partial A_t}{\partial \mathbf{u}}\right) A_t^{-1} b_t = -\left(A_t^{-1} b_t\right)^\top \left(\frac{\partial A_t}{\partial \mathbf{u}}\right) \left(A_t^{-1} b_t\right) \\ &= -y_t^\top \frac{\partial}{\partial \mathbf{u}} \left(\nabla \cdot \text{Diag}\left(\mathbf{w}_t\right)^{-1} \nabla \cdot^\top\right) y_t \\ &= -y_t^\top \nabla \cdot \frac{\partial}{\partial \mathbf{u}} \left(\text{Diag}(\mathbf{u})\right) \nabla \cdot^\top y_t \\ &= -\left(\nabla y_t\right) \odot \left(\nabla y_t\right), \end{aligned}$$

and

$$\frac{\partial J}{\partial b_t} = \frac{\partial}{\partial b_t} \left(b_t^\top A_t^{-1} b_t\right) = 2 A_t^{-1} b_t = 2 y_t.$$

Combine all these, we have

$$\frac{\partial J}{\partial \rho_t} = -(\nabla y_t) \odot (\nabla y_t) + 2y_t.$$

In discretization, we use 6-connected neighborhoods, the gradient becomes

$$\left(\frac{\partial J}{\partial \rho_t}\right)_{i,j} = -\frac{1}{6} \sum_{(l,m,n)\in\mathcal{O}_{i,j,k}} (y_{t,l,m,n} - y_{t,i,j,k})^2 + 2y_{t,i,j,k},$$

where $\mathcal{O}_{i,j,k}$ is the connected neighbor of (i, j, k).

□

Chapter 6

Experiments and Results

6.1 Environments

The following experiments are conducted on two platforms which are our local personal computers and Google Colab. We use Python as the main programming language for all the works of this chapter. In this section, we will introduce the devices' configurations which are used to complete the experiments. We also present information of Python environment as well as all Python's materials on which all the experiments are working.

Our personal computer (PC) configurations are given in the Table 6.1 below.

Table 6.1: Configurations of our local personal computer.

Configurations	
Operating system	Windows 11 24H2 64bit
CPU	AMD Ryzen 7 5800U
GPU	Intergrated Radeon Graphics
RAM	16 GB

The following Table 6.2 shows the configuration of Google Colab session we use to conduct the experiments.

Table 6.2: Configurations of Google Colab session which is used to conduct our experiments.

Configurations	
Operating system	Ubuntu 22.04.3 LTS
CPU	Intel(R) Xeon(R)
GPU	T4 GPU
RAM	12.7 GB

Now, we present the Python environment information of both platforms which is given in the Table 6.3 below.

Table 6.3: Python environment information of both our local personal computer and Google Colab session.

		Version
Local PC	Python	3.10.6
	NumPy	1.24.4
	PyTorch	2.4.1
Google Colab	Python	3.10.12
	NumPy	1.26.4
	PyTorch	2.5.1

Finally, we also use another programming language that focuses on solving the optimization problem, which is *AMPL*. We will utilize a specific software called *AMPL IDE* that supports that programming language with student license which gives us access to most of the currently best solvers. This tool will be used to solve the Kantorovich problem using convex optimizaiton solvers.

6.2 Datasets

6.2.1 Shape Dataset in 2D

For the 2D shape dataset, we choose a *image based* dataset. Our dataset is collected from multiple sources on the internet which is a set of binary images representing common shapes. It contains in total over 50 images of duck, star, disk, flower, ... whose the sizes as 70×70 pixels. In the scope of our project, we will use it to perform the transitions between two shapes which is the result of the shape interpolation problem. Several samples from our dataset are given in the Figure 6.1.

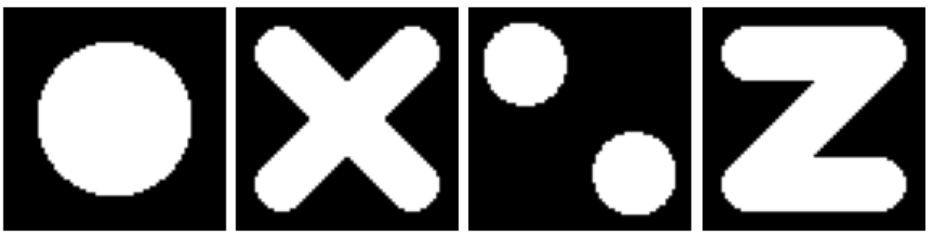


Figure 6.1: Several samples taken from our dataset with from left to right are disk, letter X, two disks and letter Z shapes.

6.2.2 Shape Dataset in 3D

For 3D shape dataset, we choose a *point cloud based* dataset. Like 2D shape dataset above, our dataset is collected from multiple sources on the internet which is a set of point clouds representing common shapes such as sphere, torus, ... Moreover, it also contains more complicated shapes such as duck, airplane, robot, animal, ... It contains in total over 100 samples whose the numbers of points ranging from 10000 to 100000. Those 3D shapes are illustrated in the Figure 6.2 below.

Figure 6.2: Several samples taken from our dataset which from left to right are duck, sphere, torus and human hand.

6.3 Metrics

In the context of shape interpolation, evaluating the quality of intermediate shapes as well as the whole transition between two shapes are critical. Several metrics are commonly used to assess the performance of interpolation methods those focus mainly on how far the two adjacent shapes in the transition is. In case of 2D shapes which are images, popular metric is *mean squared error* (MSE) which is often used to compute the difference (can be thought as distance) between two shapes. In case of 3D shapes which are point clouds, *chamfer metric* and *Hausdorff metric* are usually used to determine the distance between two shapes. In this part, we will introduce those metrics and how to apply them into our experiments especially in term of *minimal distortion* and *smooth transition* properties.

6.3.1 Mean Squared Error

Mean squared error (MSE) is one of the most widely used metrics for measuring the difference between two images. In general, it calculates the average of the squared differences between corresponding elements of the two inputs, typically a predicted output and a ground truth reference. In images processing, mean squared error serves as a basic yet effective way to quantify the pixel-wise discrepancy between two images. Now, we introduce the definition of mean squared error used in term of images.

Definition 6.3.1.1 (Mean squared error) *Let $\mathcal{X}$ and $\mathcal{Y}$ be two images whose the same size which is $n \times m$. We define the* ***mean squared error*** *between $\mathcal{X}$ and $\mathcal{Y}$ as*

$$\text{MSE}(\mathcal{X}, \mathcal{Y}) = \frac{1}{nm} \sum_{i=1}^{n} \sum_{j=1}^{m} (\mathcal{X}_{ij} - \mathcal{Y}_{ij})^2$$

in which $\mathcal{X}_{ij}$ is the corresponding pixel at position (i, j) of the image $\mathcal{X}$.

By using this metric, we can determine how far an image from each other. Moreover, it is suitable and can also be used to evaluate the transition in shape interpolation problem.

6.3.2 Chamfer Distance and Hausdorff Distance

When dealing with 3D shapes especially point clouds, common metrics such as mean squared error mentioned above seem not to be so effective. Thus, we must use other metrics which are created specially for point-cloud-based data. In this part, we introduce the two most common metrics for point clouds which are *Chamfer distance* and *Hausdorff distance*. Both metrics use the relation (in fact the distance) among points between two point clouds. The difference is, in case of Chamfer distance, that it concerns about the mean value of distances from each point in one point cloud to its closest point in the other point cloud and vice versa. In case of Hausdorff distance, it measures the maximum distance of a point in one point cloud to the nearest point in

another point cloud and vice versa. Formally, it captures the worst-case mismatch between two point clouds by identifying the point that is farthest from any point in the opposite one. Now, we consider the Chamfer distance.

Definition 6.3.2.1 (Chamfer distance) *Let $\mathcal{X}$ and $\mathcal{Y}$ be two sets of points (point clouds). Let $n > 0$ and $m > 0$ be the numbers of elements of $\mathcal{X}$ and $\mathcal{Y}$, respectively. Let $\mathrm{NN}(x, \mathcal{X})$ be a function returning a point of $\mathcal{X}$ which is the nearest neighbor of the point x. Then*

$$\mathrm{chamfer}(\mathcal{X}, \mathcal{Y}) = \frac{1}{2n} \sum_{x \in \mathcal{X}} \|x - \mathrm{NN}(x, \mathcal{Y})\|_2 + \frac{1}{2m} \sum_{y \in \mathcal{Y}} \|y - \mathrm{NN}(y, \mathcal{X})\|_2$$

*defines the **Chamfer distance** between $\mathcal{X}$ and $\mathcal{Y}$.*

Here is the definition of Hausdorff distance.

Definition 6.3.2.2 (Hausdorff distance) *Let $\mathcal{X}$ and $\mathcal{Y}$ be two sets of points (point clouds). Let $n > 0$ and $m > 0$ be the numbers of elements of $\mathcal{X}$ and $\mathcal{Y}$, respectively. Let $\mathrm{NN}(x, \mathcal{X})$ be a function returning a point of $\mathcal{X}$ which is the nearest neighbor of the point x. Then*

$$\mathrm{hausdorff}(\mathcal{X}, \mathcal{Y}) = \frac{1}{2} \max_{x \in \mathcal{X}} \|x - \mathrm{NN}(x, \mathcal{Y})\|_2 + \frac{1}{2} \max_{y \in \mathcal{Y}} \|y - \mathrm{NN}(y, \mathcal{X})\|_2$$

*defines the **Hausdorff distance** between $\mathcal{X}$ and $\mathcal{Y}$.*

Using these metrics, we can evaluate the quality of the interpolation between source and target shape by considering the distance between two consecutive intermediate shapes of the transition.

6.3.3 Minimal Distortion and Smooth Transition

When dealing with the shape interpolation problem, there are many key properties we need to guarantee [1, 5, 13]. In the scope of our capstone project, we only consider two key properties which are *minimal distortion* and *smooth transition* on the whole transition from the source to the target shape itself. Using the metrics defined previously, in this part, we will discover the relationship between them and those two key properties.

For minimal distortion property, it is defined as how straight-forward the transition is. In other words, minimal distortion property assures that our obtained transition will have no redundant intermediate shapes when going from source shape to target shape. Straight-forward transition without redundant intermediate shapes means that the chosen distances between two consecutive shapes in the transition must be minimum or the path from source to target shape must be the shortest path in the chosen metric space. For example, given a shape interpolation problem which is going from a disk to itself, what we expect is that nothing else should appear in the "path" from a disk to a disk. That means we need all the intermediate shapes of the transition must be the disk. The minimal distortion property is violated if there is other shapes rather than a disk appear in the transition, or, the disk is split when going from it to itself. That circumstance is illustrated in Figure 6.3 below.

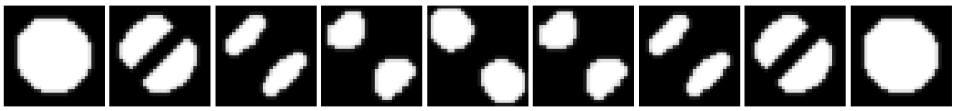


Figure 6.3: This is an example of the case when minimal distortion property is violated (visual results obtained by using Sinkhorn algorithm). We can see in the figure that when going through a "path" from a disk to itself, in the middle, the disk is split into two smaller disks.

Assume that we have all the distance values, for instance MSE for images, between every pair of consecutive shapes in the transition. We consider the shape interpolation problem above from a disk to itself. Minimal distortion property states that all the distance values must be zero. If minimal distortion property is violated, as in Figure 6.3, those distance values must be greater than zero which causes the *mean* of those value to be greater than it should be. This statement is also true for general case when solving the shape interpolation problem. Any redundant intermediate shapes in the transition will make the mean of all distance values between two consecutive shapes greater than usual. In fact, this mean value is widely used to evaluate the quality of the transition obtained by the shape interpolation problem [19]. However, small mean value can guarantee minimal distortion property but not smooth transition property.

For smooth transition property, it is defined as how smooth and good visualizing the transition is. In other words, the differences between two consecutive shapes in the transition must be small. The Figure 6.4 gives us an example of a transition that has minimal distortion but not smooth transition. In this case, the transition is straight-forward with a big jump in the middle of the "path" from one disk shape to two disks shape. Considering the same shape interpolation problem, Figure 6.5 gives us a smoother transition. Assuming that the transition in Figure 6.5 is well defined, we obtain that the mean value of distances between two consecutive shapes is the same in both cases. The difference is that transition in Figure 6.4 has larger change in distance values between consecutive shapes than one in Figure 6.5. That is due to the big jump in the middle and consistency in first and last phase of the transition. For more formal representation, we can consider it as the *variance* of distance values between every pair of consecutive shapes in the transition. In particular, the greater variance value is, the less smooth transition is.

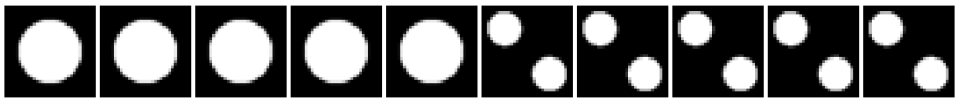


Figure 6.4: This is an example of the transition from a disk shape to two disks shape. The obtained transition has minimal distortion (straight-forward transition) but not smooth transition.

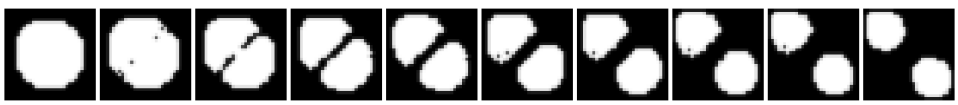


Figure 6.5: This is an example of the shape interpolation problem from a disk shape to two disks shape (visual results obtained by using convex optimization solver). The obtained transition is smooth.

From all above, we can consider the two properties which are minimal distortion and smooth transition by computing the distance values between consecutive intermediate shapes in the obtained transition of the shape interpolation problem. Then we calculate the mean and variance value from these distance values to evaluate those two properties of the transition.

6.4 Shape Interpolation in 2D

In this part, we conduct experiments for shape interpolation problem on 2D shapes. In our original dataset, all the images have the size as 70×70 pixels. In this experiment, we will resize them so that they have the size as 32×32 pixels. All the resized images are gray images whose pixel values ranging from 0 to 1. We will conduct the experiments on two typical transitions which are *from a disk shape to two disks shape* and *from a letter Z to letter X*. Those two cases stand for two transitions which have different topologies (from a disk to two disks) and complex geometry features (from letter Z to letter X). For each case, we will solve the shape interpolation problem using five distinct methods which are linear interpolation, baseline variational autoencoder (baseline VAE) proposed in [34], dynamic optimal transport variational autoencoder (DOT VAE) introduced in Section 5.3.2, Sinkhorn algorithm introduced in Section 5.1 and convex optimization solver. In the method using convex optimization solver, we choose solver *Gurobi* to solve the problem.

For each method, we generate *11 shapes* including the source and target shape to form a transition between them. We also record *10 mean squared error (MSE) values* between pair of consecutive intermediate shapes in the transition. Finally, for each transition of each method, we compute the mean value (μ) and variance value (σ^2) of those MSE values to evaluate the minimal distortion and smooth transition property. All the results are given in Table 6.4 and Table 6.5 below.

Table 6.4: The MSE values between each pair of consecutive shapes in two transitions from a disk to two disks and from letter Z to letter X using five different methods.

Source/Target	Methods	MSE between two consecutive shapes									
Disk/two disks	Linear interpolation	0	0.0046	0.0046	0.0046	0.0046	0.0046	0.0046	0.0046	0.0046	0.0046
	Baseline VAE	0.00062	0.00041	0.00016	0.22	0.024	0.16	0.3	0.0035	0.0015	8.5×10^{-5}
	DOT VAE	2.2×10^{-5}	0.01	0.025	0.05	0.04	0.032	0.019	0.014	0.013	0.019
	Sinkhorn alg.	0.14	0.015	0.011	0.0092	0.0083	0.0088	0.0077	0.0056	0.011	0.063
	Gurobi	0.044	0.075	0.022	0.027	0.026	0.031	0.036	0.039	0.034	0.02
Z/X	Linear interpolation	0	0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.002
	Baseline VAE	0.00033	0.0015	0.00093	0.0028	0.15	0.0098	0.01	8.9×10^{-5}	9.3×10^{-5}	0.00036
	DOT VAE	2.9×10^{-5}	0.0014	0.01	0.0098	0.015	0.018	0.011	0.0049	0.004	0.002
	Sinkhorn alg.	0.024	0.054	0.027	0.025	0.036	0.0088	0.0077	0.0056	0.011	0.063
	Gurobi	0.13	5.9×10^{-5}	0.0027	0.0096	0.011	0.001	0.012	0.003	3.7×10^{-5}	0.11

In both transitions of this experiment, from Table 6.4, we see that the best results are obtained when using linear interpolation method. These results do not match with what we expect because linear interpolation method is the most simple interpolation approach, we do not look forward to such so good results from it. However, those results make sense. Recalling about MSE value, MSE value in term of distance between two images is computed using pixel values of the images and does not concern about the position information. Note that shape information not only consists of density values (pixel values in this case) but also position values (pixels contained in shape). Moreover, MSE values are also strongly related to linear interpolation method by working only on the pixel values in the images. Thus, when using MSE value as the distance between two images, linear interpolation will give us the best results. From now on, we will leave the results by linear interpolation method out and discuss other methods' results to compare among them. In fact, in spite of having the best results, linear interpolation methods give us not a well visualizing transition.

Table 6.5: The mean value (μ) and variance value (σ^2) of the MSE values between each pair of consecutive shapes in two transitions from a disk to two disks and from letter Z to letter X using five different methods.

Transitions	**Methods**	μ	σ^2
Disk/two disks	Baseline VAE	0.072	0.012
	DOT VAE	**0.022**	**0.0002**
	Sinkhorn alg.	0.027	0.0016
	Gurobi	0.035	0.00023
Z/X	Baseline VAE	0.018	0.002
	DOT VAE	**0.0078**	$\mathbf{3.5\times10^{-5}}$
	Sinkhorn alg.	0.0081	0.00013
	Gurobi	0.0095	0.000091

Now, we look at the Table 6.5. For shape interpolation method using baseline VAE, we obtain the worst results overall. That means it does not guarantee so well both minimal distortion and smooth transition property. In fact, this method almost does no interpolation between two shapes. This circumstance can be explained by that for a baseline VAE, reconstructing shapes is its main goal but not interpolating. That means, in the sense of baseline VAE, optimizing the interpolation process is not concerned. This contrasts to the objective of the DOT VAE which we will discuss now.

In comparison with other three other interpolation methods (after leaving linear interpolation out), we see that DOT VAE generates better results when coming to both minimal distortion and smooth transition property. Unlike baseline VAE which focuses only on reconstructing task, DOT VAE, by using the intuition of optimal transport, tries to do both works which are shape reconstructing and shape interpolating. There is one remarkable observation here is that method performs well on the shape interpolation problem between two shape whose different topologies (from a disk to two disks) or different complex geometry features (from letter Z to letter X).

For shape interpolation problem using Sinkhorn algorithm, the results are good. This method gives transitions which guarantee better minimal distortion property but worse smooth transition property than solutions by using convex optimization solver. Note that Sinkhorn algorithm is originated by the problem of unbalanced optimal transport. Because the two shape interpolation problems in this experiment are both unbalanced. Thus, the results by Sinkhorn seem to be better than ones by convex optimization solver in the sense of minimal distortion

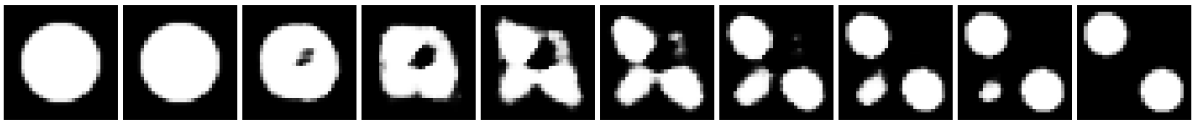
property. When considering the smooth transition property, because Sinkhorn algorithm is just an algorithm to solve the regularized optimal transport problem, its results are not good as the absolute results in term of smooth transition given by convex optimization solver.

For shape interpolation problem using convex optimization solver, the results are still good. Moreover, we can also see the tradeoff of minimal distortion and smooth transition property between two methods which are Sinkhorn algorithm and convex optimization solver based on its characteristics when dealing with the shape interpolation problem. In fact, those methods generate very good visualizing transitions.

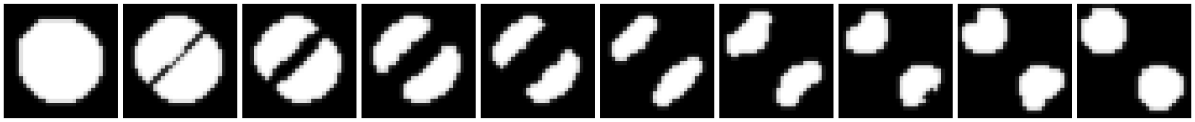
We know that the results given by Sinkhorn algorithm and convex optimization solver are not so good as ones given by DOT VAE. One more explanation is that in case of using DOT VAE, the interpolation process is conducted directly on the images. In case of Sinkhorn algorithm or convex optimizaiton solver, however, the interpolation process firstly converts the images into suitable form to do the interpolation using optimal transport and then converts it back to image. In image processing, there is a problem called *hole problem* or *resampling artifact* in which the some pixels do not have any value because of discretizing process. They make our results not good as we expect in the sense of Sinkhorn algorithm and convex optimization solver method.

(a) Linear interpolation

(b) Baseline VAE



(c) DOT VAE



(d) Sinkhorn algorithm

(e) Gurobi

Figure 6.6: The results of 2D shape interpolation from a disk shape image to two disks shape image from our dataset using five methods which are linear interpolation, baseline VAE, DOT VAE, Sinkhorn algorithm and convex optimization solver Gurobi. There are 5 subfigures which represent 5 transitions given by those 5 methods which are (a) Linear interpolation, (b) Baseline VAE, (c) DOT VAE, (d) Sinkhorn algorithm and (e) Gurobi. At each subfigure, there are 10 shapes standing for a transition from source shape to target shape. The first shape to the left of each subfigure is the source shape, the first shape to the right of each subfigure is the target shape.

We also provide a visual results for five methods in this experiment on the shape interpolation problem from a disk shape image to two disks shape image in Figure 6.6. Each given transition has in total ten intermediate shapes including source shape and target shape in which the source shape is the first shape to the left and the target shape is the first shape to the right of the transition.

Now, we look at the Figure 6.6. For the linear interpolation method, although it has the best results based on Table 6.4, its visual results are not well visualizing. In the transition by using this method, all the intermediate shapes seem to be blurred when traveling from source shape (disk shape) to target shape (two disks shape). In case of baseline VAE, we see that except for a noisy changing state image in the middle, the whole transition generated by this method remains unchanged. That makes the obtained results by this method is good. Coming to DOT VAE, this method generates a transition which is smooth, clear enough and definitely better than

baseline VAE. Moreover, the achieved transition seems to preserve some physical constraints when masses from source shape travels to reach the target shape. Both Sinkhorn algorithm and convex solver methods give us very well visualizing result transitions. Moreover, we can see the differences between transitions of the shape interpolation problem when applying dynamic optimal transport and optimal transport. About the artifact resampling problem mentioned above, those artifacts, in fact, exist in the obtained transition using Sinkhorn algorithm and convex optimization solver in Figure 6.6 if we look more carefully.

6.5 Shape Interpolation in 3D

In this part, we conduct experiments for shape interpolation problem on 3D shapes. In our original dataset, the format of point cloud data is not suitable for some methods. Moreover, all of the original point clouds in the dataset have too large number of points which is not suitable to conduct the shape interpolation experiments. Thus, in this experiment, we will voxelize all the point clouds into cubes whose size as $32 \times 32 \times 32$ using the voxelization method introduced in Section 5.3.1. All the voxelized data has element values 0 or 1 in which 0 value stands for non-existence of the point in the point cloud and 1 value stands for existence of the point in the point cloud. We will conduct the experiments on two typical transitions which are *from duck to torus* and *from sphere to human hand*. Those two cases stand for two transitions which have different topologies (from a duck to a torus) and complex geometry features (from a sphere to a human hand). For each case, we will solve the shape interpolation problem using five distinct methods which are linear interpolation, baseline VAE, dynamic optimal transport multiple channels variational autoencoder (DOT-MC VAE) introduced in Section 5.3.3, DOT VAE introduced in Section 5.3.4 and Sinkhorn algorithm.

For each method, we generate *11 shapes* including the source and target shape to form a transition between them. We also record *10 Chamfer distance values* and *10 Hausdorff distance values* between every pair of consecutive intermediate shapes in the transition. Finally, for each transition of each method, we compute the mean value (μ) and variance value (σ^2) of those Chamfer and Hausdorff distance values to evaluate the minimal distortion and smooth transition property. All the results are given in Table 6.6, Table 6.7, Table 6.8 and Table 6.9 below.

Table 6.6: The Chamfer distance values between each pair of consecutive shapes in two transitions from duck to torus and from sphere to human hand using five different methods.

Source/Target	Methods	Chamfer distances between two consecutive shapes									
Duck/torus	Linear interpolation	0.0083	0.011	0.012	0.019	0.18	1.69	0.14	0.028	0.013	0.011
	Baseline VAE	0	0	0.0022	1.42	1.11	0.38	0.92	0.5	0	0
	DOT-MC VAE	0.0085	0.019	0.032	0.067	0.082	0.21	0.18	0.083	0.041	0.017
	DOT VAE	0.03	0.072	0.056	0.066	0.083	0.11	0.15	0.066	0.029	0.014
	Sinkhorn alg.	0.36	0.18	0.094	0.11	0.11	0.1	0.11	0.091	0.094	0.16
Sphere/hand	Linear interpolation	0.0084	0.0077	0.0077	0.013	0.9	0.79	0.018	0.015	0.0085	0.0056
	Baseline VAE	0	0	0.011	0.052	2.06	0.039	0.024	0.018	0.021	0.02
	DOT-MC VAE	0.79	0.036	0.012	0.005	0.0021	0.0011	0.0017	0.0012	0.00039	0.00062
	DOT VAE	0.05	0.062	0.064	0.058	0.074	0.11	0.15	0.066	0.029	0.014
	Sinkhorn alg.	0.033	0.032	0.041	0.042	0.063	0.046	0.078	0.097	0.16	0.35

Table 6.7: The mean value (μ) and variance value (σ^2) of the Chamfer distance values between each pair of consecutive shapes in two transitions from duck to torus and from sphere to human hand using five different methods.

Transitions	Methods	μ	σ^2
Duck/torus	Linear interpolation	0.21	0.25
	Baseline VAE	0.43	0.26
	DOT-MC VAE	0.074	0.0044
	DOT VAE	**0.067**	**0.0015**
	Sinkhorn alg.	0.14	0.0061
Sphere/hand	Linear interpolation	0.18	0.11
	Baseline VAE	0.22	0.38
	DOT-MC VAE	**0.085**	0.055
	DOT VAE	0.12	0.0092
	Sinkhorn alg.	0.094	**0.0086**

In the sense of Chamfer distance, the two methods which are linear interpolation and baseline VAE give the worst results in both transition from duck to torus and from sphere to hand. Their results are much worse than other three methods in this experiment. In case of linear interpolation, because of its simplicity, smooth and clear transitions are not often generated. In case of baseline VAE, as explained in the experiments on 2D shapes, its main goal is not optimizing the interpolation process but reconstructing process. Thus, the transitions generated by baseline VAE seem to do no interpolation between two shapes.

Now, we consider the transition from a duck to a torus. Coming to shape interpolation using DOT-MC VAE, it gives us good results which are actually not so far from the best results given by using DOT VAE. In the sense of DOT VAE, the best results are obtained. From those results of both DOT-MC VAE and DOT VAE, we see that they adapt well with the shape interpolation problem between two shapes which have different topologies (a duck and a torus) when guaranteeing minimal distortion and smooth transition property. For Sinkhorn algorithm, the obtained results are not good as ones given by DOT-MC VAE and DOT VAE. This can be explained by that Sinkhorn algorithm sometimes generates slightly "redundant" intermediate shapes in the transition. This can cause that the solutions of Sinkhorn algorithm are not good as we expect.

We consider the transition from a sphere to a human hand. Coming to shape interpolation using DOT-MC VAE, it obtains the best results in term of minimal distortion. For Sinkhorn algorithm method, it obtains the best results in term of smooth transition. Note that the results given by Sinkhorn algorithm are just slightly worse than ones by DOT-MC VAE. In case of DOT VAE method, in spite of worse results than DOT-MC VAE and Sinkhorn algorithm, it still gives us good results. In the sense of smooth transition property, DOT VAE obtain much better results than DOT-MC VAE and only slightly worse than ones by Sinkhorn algorithm.

Table 6.8: The Hausdorff distance values between each pair of consecutive shapes in two transitions from duck to torus and from sphere to human hand using five different methods.

Transitions	**Methods**	**Hausdorff distances between two consecutive shapes**										
Duck/torus	Linear interpolation	1	1	1	1	1.5	10.5	1.21	1	1	1	
	Baseline VAE	0	0	0.5	9.95	9.09	3.72	8.33	6.4	0	0	
	DOT-MC VAE	1	1	1	2.71	2.66	5.07	4.24	2.5	2.83	1	
	DOT VAE	4.53	3.19	1.93	1.71	1.93	2.34	3.56	2.28	1.57	1	
	Sinkhorn alg.	2.37	1.93	2.28	2.28	2.09	1.83	2.09	1.62	1.37	1.37	
Sphere/hand	Linear interpolation	1	1	1.62	1	4.74	5.7	1	1	1	0.5	
	Baseline VAE	0	0	1	2	7.43	1.71	1.5	1.21	1.21	0.71	
	DOT-MC VAE	4.34	4.22	2	1	1	1	1	1	1	1	
	DOT VAE	1.91	2.5	2	1.62	2.08	1.91	2.29	2.71	3.56	4.24	
	Sinkhorn alg.	1.21	1.21	1.21	1.21	1.5	1.5	1.62	1.62	2	1.66	

Table 6.9: The mean value (μ) and variance value (σ^2) of the Hausdorff distance values between each pair of consecutive shapes in two transitions from duck to torus and from sphere to human hand using five different methods.

Transitions	Methods	μ	σ^2
Duck/torus	Linear interpolation	2.02	8.01
	Baseline VAE	3.8	16.17
	DOT-MC VAE	2.4	1.87
	DOT VAE	2.41	1.01
	Sinkhorn alg.	**1.92**	**0.12**
Sphere/hand	Linear interpolation	1.86	2.94
	Baseline VAE	1.68	4.07
	DOT-MC VAE	1.76	1.68
	DOT VAE	2.48	0.61
	Sinkhorn alg.	**1.47**	**0.064**

In the sense of Hausdorff distance, however, the two methods which are linear interpolation and baseline VAE give the worst results in both transition from duck to torus and from sphere to hand on smooth transition property. Coming to the minimal distortion property, the worst result on transition from duck to torus is given by the baseline VAE method and on transition from sphere to hand is given by the DOT VAE. Those not good results of linear interpolation and baseline VAE are due to their simplicity and reconstructing priority, respectively.

Now, we consider the transition from a duck to a torus. The best results are obtained by Sinkhorn algorithm. Despite bad results on minimal distortion property of DOT-MC VAE and DOT VAE, the smooth transition property is guaranteed by them. We can explain the reason why these circumstances happen. Note that because Hausdorff distance concerns only about the longest distances, it is sensitive to outlier points in intermediate shapes. That means if the transitions generated by DOT-MC VAE and DOT VAE exist artifacts in intermediate shapes, then, in term of minimal distortion, the results given by them will be not so good even worse than given by linear interpolation and baseline VAE. For Sinkhorn algorithm, because this algorithm generates a transport plan to construct the transition, that means the interpolation given by it is straight-forward and rarely contains artifacts.

We consider the transition from a sphere to a human hand. The best results are still obtained by Sinkhorn algorithm. Despite bad results on minimal distortion property of DOT-MC VAE and DOT VAE, the smooth transition property is guaranteed by them. However, in case of DOT-MC VAE, the smooth transition property is not guaranteed so well. Those results can be explained similarly to the case of transition from a duck to a torus based on the sensitivity of Hausdorff distance to artifacts.

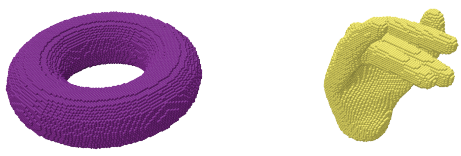
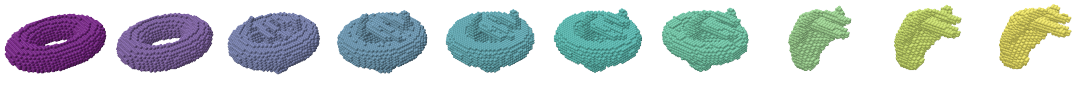
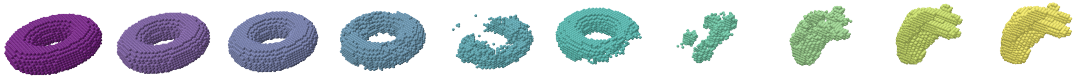


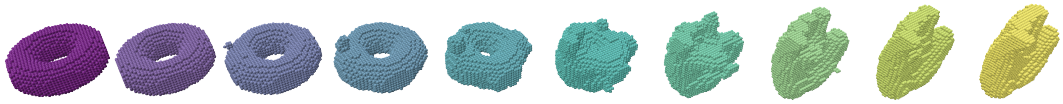
Figure 6.7: The source and target shape which are torus and human hand used to perform the 3D shape interpolation for visual results.



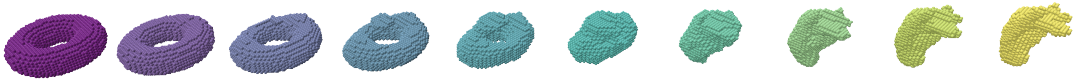
(a) Linear interpolation



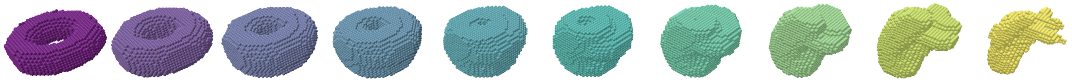
(b) Baseline VAE



(c) DOT-MC VAE



(d) DOT VAE



(e) Sinkhorn algorithm

Figure 6.8: The results of 3D shape interpolation from torus shape to human hand shape given in Figure 6.7 using five methods which are linear interpolation, baseline VAE, DOT-MC VAE, DOT VAE and Sinkhorn algorithm. There are 5 subfigures which represent 5 transitions given by those 5 methods which are (a) Linear interpolation, (b) Baseline VAE, (c) DOT-MC VAE, (d) DOT VAE and (e) Sinkhorn algorithm. At each subfigure, there are 10 shapes standing for a transition from source shape to target shape. The first shape to the left of each subfigure is the source shape, the first shape to the right of each subfigure is the target shape.

We also provide a visual results for five methods in this experiment on the shape interpolation problem from torus shape to human hand shape in Figure 6.8. Each given transition has in total ten intermediate shapes including source shape and target shape in which the source shape is the first shape to the left and the target shape is the first shape to the right of the transition.

Now, we look at Figure 6.8. For linear interpolation method, the obtained transition looks like a fusion process of two source and target shapes. Thus, it gives us poorly visualizing results. In case of baseline VAE, all the intermediate shapes of the transition given by it, except for noisy shapes in the middle, remain unchanged. Hence, it also gives us poorly visualizing results. The transitions given by two methods which are DOT-MC VAE and DOT VAE, we can see, are clear, smooth and straight-forward. However, those transitions generated by DOT-MC VAE and DOT VAE sometimes exist artifacts in intermediate shapes which are indeed outlier points. Therefore, in term of minimal distortion on Hausdorff distance in Table 6.9, the results given by them are not so good even worse than ones given by linear interpolation and baseline VAE. Sinkhorn algorithm generates straight-forward and smooth transitions. But, the transition generated by Sinkhorn algorithm contains slightly "redundant" intermediate shapes in the transition. This can cause that the solutions of Sinkhorn algorithm are not good as we expect in

the sense of Chamfer distance.

We also give some more results of the shape interpolation problem using optimal transport theory.

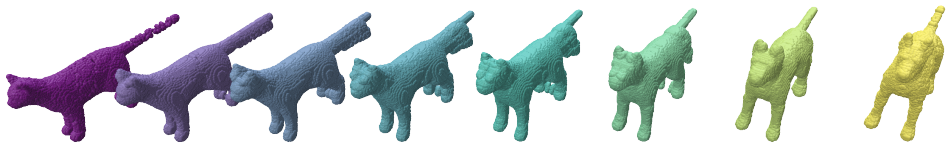


Figure 6.9: The result of 3D shape interpolation from cat to dog using Sinkhorn algorithm.

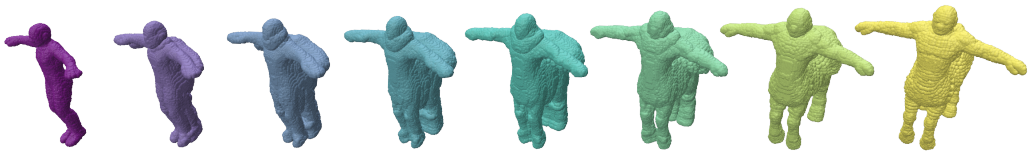


Figure 6.10: The result of 3D shape interpolation from human to centaurus using Sinkhorn algorithm.

6.6 Time Complexity

As mentioned above, Kantorovich problem can be written as a minimizing problem in which the decision variable is the transport plan P. In fact, this problem is a convex linear optimization problem and can be easily solved by using convex solvers. However, the complexity of solving Kantorovich problem by those solvers increases rapidly when the size of the decision varible increases. Previously, we also introduce Sinkhorn algorithm which tends to decrease remarkably the complexity of solving such problem [16]. In this part, we survey the time complexity when using convex optimization solvers as well as Sinkhorn algorithm to solve the Kantorovich problem dealing with images.

For data, we use two binary images to be the source and target shape of the problem. Those images are resized to six sizes which are 10×10, 20×20, 30×30, 40×40, 50×50, 60×60. Note that if an image whose size $n \times n$, the transport plan P will have the size as $n^2 \times n^2$. The For the solvers, we choose four currently best solvers for optimization problem which are Gurobi [27], MOSEK [2], HiGHS [30] and SNOPT [25]. We also conduct the same experiments using Sinkhorn algorithm. We will record all values of the times solvers and the Sinkhorn algorithm used to solve the problem. To conduct experiments on solvers, we use AMPL which is an algebraic modeling language to describe and solve high-complexity problems for large-scale mathematical computing. We also use Python programming language to conduct the experiment using Sinkhorn algorithm. All the experiments are conducted on local PC. Here is the plot of the time complexity when solving the Kantorovich problem on images whose different sizes by convex solvers and Sinkhorn algorithm.

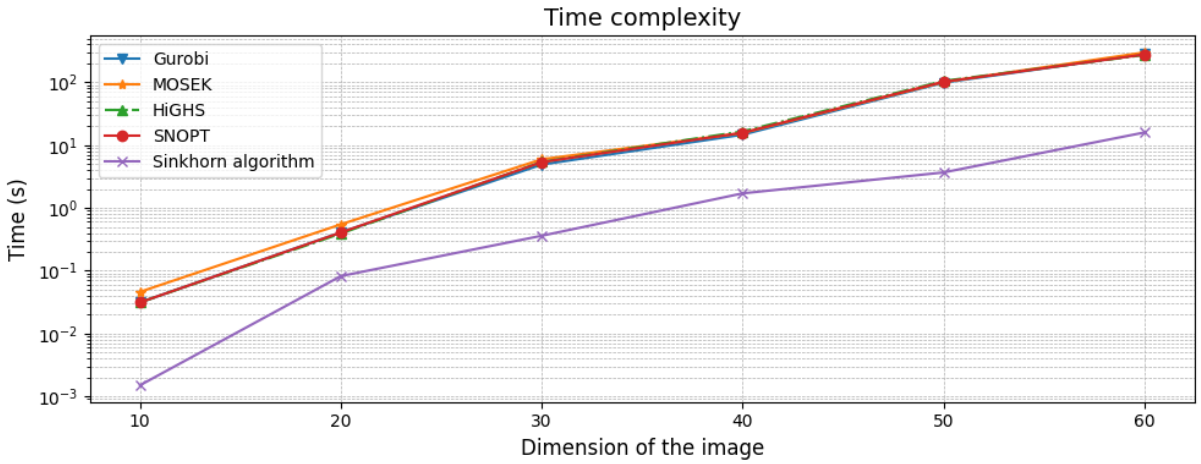


Figure 6.11: Time complexities when solving the Kantorovich problem on images whose different sizes by convex solvers and Sinkhorn algorithm.

By the plot given in Figure 6.11, we can see that the time complexity of all methods increases fast for larger images. However, Sinkhorn algorithm seems to be remarkably better in term of consumed time when solving the Kantorovich problem. One interesting result is that all the solvers tends to have the same time complexity when dealing with this problem although they use different general algorithms to solve the optimization problems.

Chapter 7

Conclusion

7.1 Summary

In this capstone project, we have studied about optimal transport theory and how to apply it to the shape interpolation problem. For shape interpolation problem, we have investigated existing methods for 2D and 3D shape data such as using neural network, geodesic based and so on. In order to gain a deep understanding on this problem using optimal transport theory, we have explored fundamental areas of mathematics, highlighting the elegance of logical reasoning and its relevance to the shape interpolation problem. Our study systematically covers key topics including convex analysis, measure theory, probability theory and optimization theory culminating in the study of optimal transport theory. We have introduced many methods to solve the shape interpolation problem using optimal transport theory in both 2D and 3D shape data. Those approaches include using convex solvers, Sinkhorn algorithm and combining dynamic optimal transport with variational autoencoder. We have also proposed two new approaches for the 3D shape data which are dynamic optimal transport mutiple channels variational autoencoder in Section 5.3.3 and dynamic optimal transport variational autoencoder for 3D in Section 5.3.4. In the sense of metrics, we have used popular metrics which are MSE, Chamfer distance and Hausdorff distance to form new approaches for evaluating the two key properties of the shape interpolation problem which are minimal distortion and smooth transition.

On the practical front, we have conducted many experiments for shape interpolation problem using optimal transport theory on both 2D and 3D data. All the obtained results are visualized for comparison in Figure 6.6 and Figure 6.8. In case of 2D shape data, method using convex solvers gives us the best qualitive results. For other methods, Sinkhorn algorithm gives us a good results which is much better visualizing than ones by DOT VAE. Additionally, in case of 3D shape data, our new two methods give remarkably good results on 3D shape data in the sense of both minimal distortion and smooth transition property in comparison with intuitive methods using optimal transport theory such as Sinkhorn algorithm.

7.2 Limitations and Future Works

7.2.1 Limitations

For method using convex solvers, the time complexities rises rapidly when the size of the input shape data is larger. In case of Sinkhorn algorithm, although it achieves lower running time, it is still time consuming when dealing with large scale data. Thus, these two methods are not suitable for practical applications. However, in case of methods using variational autoencoder, the fast running time is an advantage although it is resource and time consuming when training the autoencoders. Unfortunately, in our experiment, the size of the dataset is still small which can cause the variational autoencoder not generalizing well enough. This makes our variational autoencoder still not suitable for practical applications.

Other limitations are about the obtained results when solving the shape interpolation problem using optimal transport. The first one is that there still exists some artifacts in the result transition which makes it not so well visualizing, for example in Figure 6.9. The second one is that the transition between two shapes sometimes looks not realistic, for example, there exists self-intersection. Another limitation is that using MSE for estimating distance between two 2D shapes is not good for evaluating the quality of interpolation results.

7.2.2 Future Works

To overcome the limitation about time complexity, we will investigate more effective methods to compute the transport plan for the shape interpolation problem. One potential candidate is convolutional Wasserstein distance. We will also try to collect more data for our shape dataset, improve the methods using variational autoencoder with dynamic optimal transport in the sense of quality of results such as less artifacts, smoother transition, ...

Moreover, we will research effective ways to reduce the appearances of artifact in obtained result transition. Also, we will investigate other approaches to enforce the interpolation results looking more realistic, preventing circumstances such as components self-intersection or components split out.

Finally, we will find more compatible metrics for computing distances between two 2D shapes in the shape interpolation problem. One potential candidate is 2D Chamfer distance.

Bibliography

[1] Marc Alexa, Daniel Cohen-Or, and David Levin. "As-rigid-as-possible shape interpolation". In: *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*. 2023, pp. 165–172.

[2] MOSEK ApS. *The MOSEK optimization toolbox for MATLAB manual. Version 9.0.* 2019. URL: `http://docs.mosek.com/9.0/toolbox/index.html`.

[3] Brian Avants and James Gee. "Symmetric geodesic shape averaging and shape interpolation". In: *International Workshop on Mathematical Methods in Medical and Biomedical Image Analysis*. Springer. 2004, pp. 99–110.

[4] Brian Avants and James C Gee. "Geodesic estimation for large deformation anatomical shape averaging and interpolation". In: *Neuroimage* 23 (2004), S139–S150.

[5] Seung-Yeob Baek, Jeonghun Lim, and Kunwoo Lee. "Isometric shape interpolation". In: *Computers & Graphics* 46 (2015), pp. 257–263.

[6] Guha Balakrishnan et al. "An unsupervised learning model for deformable medical image registration". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 9252–9260.

[7] Mojtaba Bemana et al. "X-fields: Implicit neural view-, light-and time-image interpolation". In: *ACM Transactions on Graphics (TOG)* 39.6 (2020), pp. 1–15.

[8] Jean-David Benamou and Yann Brenier. "A computational fluid mechanics solution to the Monge-Kantorovich mass transfer problem". In: *Numerische Mathematik* 84.3 (2000), pp. 375–393.

[9] Martin Burger et al. "Covariance-modulated optimal transport and gradient flows". In: *arXiv preprint arXiv:2302.07773* (2023).

[10] Wilhelm Burger et al. "Fourier shape descriptors". In: *Principles of Digital Image Processing: Advanced Methods* (2013), pp. 169–227.

[11] Pingping Cai et al. "EINet: Point Cloud Completion via Extrapolation and Interpolation". In: *European Conference on Computer Vision*. Springer. 2025, pp. 377–393.

[12] Zhiqin Chen and Hao Zhang. "Learning implicit fields for generative shape modeling". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 5939–5948.

[13] Edward Chien, Renjie Chen, and Ofir Weber. "Bounded distortion harmonic shape interpolation". In: *ACM Transactions on Graphics (TOG)* 35.4 (2016), pp. 1–15.

[14] Keenan Crane, Clarisse Weischedel, and Max Wardetzky. "Geodesics in heat: A new approach to computing distance based on heat flow". In: *ACM Transactions on Graphics (TOG)* 32.5 (2013), pp. 1–11.

[15] RICCARDO CRISTOFERI. "NWI-WM246-OPTIMAL TRANSPORT LECTURE NOTES". In: ().

[16] Marco Cuturi. "Sinkhorn distances: Lightspeed computation of optimal transport". In: *Advances in neural information processing systems* 26 (2013).

[17] Stefano De Marchi et al. “Shape-driven interpolation with discontinuous kernels: Error analysis, edge extraction, and applications in magnetic particle imaging”. In: *SIAM Journal on Scientific Computing* 42.2 (2020), B472–B491.

[18] Chengpu Duan et al. “Improving the performance of 3D shape measurement of moving objects by fringe projection and data fusion”. In: *IEEE Access* 9 (2021), pp. 34682–34691.

[19] Marvin Eisenberger and Daniel Cremers. “Hamiltonian dynamics for real-world shape interpolation”. In: *European conference on computer vision*. Springer. 2020, pp. 179–196.

[20] Moshe Eliasof, Andrei Sharf, and Eran Treister. “Multimodal 3D shape reconstruction under calibration uncertainty using parametric level set methods”. In: *SIAM Journal on Imaging Sciences* 13.1 (2020), pp. 265–290.

[21] Hans G Feichtinger. “Choosing function spaces in harmonic analysis”. In: *Excursions in Harmonic Analysis, Volume 4: The February Fourier Talks at the Norbert Wiener Center* (2015), pp. 65–101.

[22] Xue Feng and Thomas Strohmer. “Improving Autoencoder Image Interpolation via Dynamic Optimal Transport”. In: *arXiv preprint arXiv:2404.08900* (2024).

[23] Charlie Frogner et al. “Learning with a Wasserstein loss”. In: *Advances in neural information processing systems* 28 (2015).

[24] Emna Ghorbel and Faouzi Ghorbel. “Data augmentation based on shape space exploration for low-size datasets: application to 2D shape classification”. In: *Neural Computing and Applications* (2024), pp. 1–24.

[25] Philip E Gill, Walter Murray, and Michael A Saunders. “SNOPT: An SQP algorithm for large-scale constrained optimization”. In: *SIAM review* 47.1 (2005), pp. 99–131.

[26] Aaron Gokaslan et al. “Improving shape deformation in unsupervised image-to-image translation”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018, pp. 649–665.

[27] Gurobi Optimization, LLC. *Gurobi Optimizer Reference Manual*. 2024. URL: `https://www.gurobi.com`.

[28] Stephan Haefner, Robert Mueller, and Reiner S Thomae. “Full 3D antenna pattern interpolation using Fourier transform based wavefield modelling”. In: *WSA 2016; 20th International ITG Workshop on Smart Antennas*. VDE. 2016, pp. 1–8.

[29] Agathe Herrou et al. “Symmetrized semi-discrete optimal transport”. In: *arXiv preprint arXiv:2206.04529* (2022).

[30] Qi Huangfu and JA Julian Hall. “Parallelizing the dual revised simplex method”. In: *Mathematical Programming Computation* 10.1 (2018), pp. 119–142.

[31] Romain Hug, Emmanuel Maitre, and Nicolas Papadakis. “Multi-physics optimal transportation and image interpolation”. In: *ESAIM: Mathematical Modelling and Numerical Analysis* 49.6 (2015), pp. 1671–1692.

[32] Haifeng Ji. “An immersed Crouzeix–Raviart finite element method in 2D and 3D based on discrete level set functions”. In: *Numerische Mathematik* 153.2 (2023), pp. 279–325.

[33] Chiyu Jiang et al. “Shapeflow: Learnable deformation flows among 3d shapes”. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 9745–9757.

[34] Diederik P Kingma. “Auto-encoding variational bayes”. In: *arXiv preprint arXiv:1312.6114* (2013).

[35] Matthias Liero, Alexander Mielke, and Giuseppe Savaré. “Optimal entropy-transport problems and a new Hellinger–Kantorovich distance between positive measures”. In: *Inventiones mathematicae* 211.3 (2018), pp. 969–1117.

[36] Xiuwen Liu et al. “A computational model of multidimensional shape”. In: *International journal of computer vision* 89.1 (2010), pp. 69–83.

[37] Ravi Malladi and James A Sethian. "Level set and fast marching methods in image processing and computer vision". In: *Proceedings of 3rd IEEE international conference on image processing*. Vol. 1. IEEE. 1996, pp. 489–492.

[38] Ravi Malladi, James A Sethian, and Baba C Vemuri. "Shape modeling with front propagation: A level set approach". In: *IEEE transactions on pattern analysis and machine intelligence* 17.2 (1995), pp. 158–175.

[39] Mateusz Michalkiewicz et al. "Deep level sets: Implicit surface representations for 3d shape inference". In: *arXiv preprint arXiv:1901.06802* (2019).

[40] Maciej Mikulski and Jaroslaw Duda. "Toroidal autoencoder". In: *arXiv preprint arXiv:1903.12286* (2019).

[41] Nikos Paragios. "A level set approach for shape-driven segmentation and tracking of the left ventricle". In: *IEEE transactions on medical imaging* 22.6 (2003), pp. 773–776.

[42] Jeong Joon Park et al. "Deepsdf: Learning continuous signed distance functions for shape representation". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 165–174.

[43] Ofir Pele and Michael Werman. "Fast and robust earth mover's distances". In: *2009 IEEE 12th international conference on computer vision*. IEEE. 2009, pp. 460–467.

[44] Shichong Peng, Yanshu Zhang, and Ke Li. "PAPR in Motion: Seamless Point-level 3D Scene Interpolation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024, pp. 21007–21016.

[45] Marie-Julie Rakotosaona and Maks Ovsjanikov. "Intrinsic point cloud interpolation via dual latent space navigation". In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*. Springer. 2020, pp. 655–672.

[46] Carlos Rojas et al. "Edge length interpolation". In: *ACM Symposium on Solid and Physical Modeling*. 2014.

[47] Daniel Schmitter et al. "Compactly-supported smooth interpolators for shape modeling with varying resolution". In: *Graphical Models* 94 (2017), pp. 52–64.

[48] James A Sethian and Peter Smereka. "Level set methods for fluid interfaces". In: *Annual review of fluid mechanics* 35.1 (2003), pp. 341–372.

[49] Zhixin Shu et al. "Deforming autoencoders: Unsupervised disentangling of shape and appearance". In: *Proceedings of the European conference on computer vision (ECCV)*. 2018, pp. 650–665.

[50] Ivan Skorokhodov. "Interpolating points on a non-uniform grid using a mixture of Gaussians". In: *arXiv preprint arXiv:2012.13257* (2020).

[51] Pierre Soille. "Generalized geodesic distances applied to interpolation and shape description". In: *Mathematical morphology and its applications to image processing* (1994), pp. 193–200.

[52] Justin Solomon et al. "Convolutional wasserstein distances: Efficient optimal transportation on geometric domains". In: *ACM Transactions on Graphics (ToG)* 34.4 (2015), pp. 1–11.

[53] Mitsuo Takeda and Kazuhiro Mutoh. "Fourier transform profilometry for the automatic measurement of 3-D object shapes". In: *Applied optics* 22.24 (1983), pp. 3977–3982.

[54] Dilip Kumar Verma and Ahmadreza Baghaie. "Convolutional Neural Networks vs. Deformable Image Registration For Medical Slice Interpolation". In: *arXiv preprint arXiv:2004.13784* (2020).

[55] Shantanu Vyas et al. "Latent embedded graphs for image and shape interpolation". In: *Computer-Aided Design* 140 (2021), p. 103091.

[56] Tim Winkler et al. "Multi-scale geometry interpolation". In: *Computer graphics forum*. Vol. 29. 2. Wiley Online Library. 2010, pp. 309–318.

[57] Guandao Yang et al. “Geometry processing with neural fields”. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 22483–22497.

[58] Jie Yang et al. “DSG-Net: Learning disentangled structure and geometry for 3D shape generation”. In: *ACM Transactions on Graphics (TOG)* 42.1 (2022), pp. 1–17.

[59] Wen-Wu Yang, Jing Hua, and Kun-Yang Yao. “Cr-morph: Controllable rigid morphing for 2d animation”. In: *Journal of Computer Science and Technology* 34 (2019), pp. 1109–1122.

[60] Xiao Zhan, Rao Fu, and Daniel Ritchie. “CharacterMixer: Rig-Aware Interpolation of 3D Characters”. In: *Computer Graphics Forum*. Wiley Online Library. 2024, e15047.

[61] Zehan Zheng et al. “Neuralpci: Spatio-temporal neural field for 3d point cloud multi-frame non-linear interpolation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 909–918.