GridOT – a discrete optimal transport solver on grids

Johannes Rauch ✉ ⓘD
Ulm University, Institute of Optimization and Operations Research, Germany

Leo Zanotti ✉
Ulm University, Institute of Optimization and Operations Research, Germany

──── **Abstract** ────

TODO: Abstract; Leo's ORCID; Check if references are actually OT on grids

1 Introduction

Informally, in the *optimal transport* problem we are given two probability distributions and a cost function, and we seek a minimum cost transport function that transforms one probability function into the other. In *discrete* optimal transport, the given probability distributions are discrete. Discrete optimal transport *on grids* means that the underlying cost function has geometric properties; it is for example a metric. Discrete optimal transport on grids is an important special case of the general optimal transport problem, because it has numerous applications in image processing and computer vision. For instance, Werman et al. [6] use a discrete optimal transport problem to define a distance metric for multidimensional histograms. Building on this, Peleg et al. [3] present a unified treatment of spatial and gray-level resolution in image digitization. As image pixels are conventionally arranged in two-dimensional grids, their approaches rely on solving discrete optimal transport on 2-dimensional grids. Thus, the need for a practically fast solver is evident.

1.1 Preliminaries

We introduce discrete optimal transport on grids formerly before proceeding. For a finite set S let $\mathcal{P}(S)$ denote the set of probability measures over S with the power set as the σ-algebra. Given two finite sets X and Y together with two probability measures $\mu \in \mathcal{P}(X)$ and $\nu \in \mathcal{P}(Y)$, the set of couplings between μ and ν is given by

$$\Pi(\mu, \nu) = \{\pi \in \mathcal{P}(X \times Y) : \pi(\{x\} \times Y) = \mu(x), \mu(X \times \{y\}) = \nu(y) \text{ for all } x \in X, \, y \in Y\}.$$

For a cost function $c : X \times Y \to \mathbb{R} \cup \{\infty\}$ the discrete optimal transport problem is

$$\min_{\pi \in \Pi(\mu,\nu)} C(\pi), \quad \text{where} \quad C(\pi) = \sum_{(x,y) \in X \times Y} c(x,y)\pi(x,y). \qquad (P)$$

We say that (P) is on a grid if $X = [x_1] \times \cdots \times [x_d]$ and $Y = [y_1] \times \cdots \times [y_d]$ for some positive integers $d, x_1, \ldots, x_d, y_1, \ldots, y_d$ and c denotes the squared Euclidean distance. The discrete optimal transport problem (P) is "dense" in the sense that an algorithm solving (P) has

to consider couplings of all elements in $X \times Y$, of which there are quadratically many. If $N \subset X \times Y$, then the restriction

$$\min_{\pi \in \Pi(\mu,\nu),\text{supp }\pi \subseteq N} C(\pi) \tag{P'}$$

of (P) is "sparse" in the sense that an algorithm solving (P') only has to consider couplings of all elements in N. We say that N is the *neighborhood* of (P'). Of course, $\text{OPT}(P) \leq \text{OPT}(P')$ holds for any neighborhood $N \subseteq X \times Y$, and ideally one would want to have a "small" neighborhood N such that $\text{OPT}(P) = \text{OPT}(P')$.

1.2   Related work

Schmitzer [4] devised a sparse multi-scale algorithm for dense optimal transport, which works very well in practice [5]. Ignoring the multi-scale scheme, we outline his algorithm in Algorithm 1. It repeatedly solves a restricted discrete optimal transport problem (P') in a neighborhood N and updates N to solve the discrete optimal transport problem (P) at hand. For grids he shows how to construct the neighborhoods N in a sparse manner explicitly. In particular, $\text{OPT}(P) = \text{OPT}(P')$ holds for the neighborhood N after the termination of the algorithm. We refer to his article for more details [4].

Input: An instance of discrete optimal transport (P) and a neighborhood N.
Output: An optimal coupling.
repeat
> solve the restricted discrete optimal transport problem (P') in neighborhood N
> update N

until *the cost of the coupling does not improve anymore*
output the last coupling

◾ **Algorithm 1** An outline of Schmitzer's algorithm for dense optimal transport [4] without the multi-scale scheme.

Schmitzer solves the restricted discrete optimal transport problem (P') in Algorithm 1 with the network simplex algorithm [1]. He uses the CPLEX[1] and LEMON [2] network simplex implementations for his numerical experiments. Since the problems (P') are very similar in every iteration of Algorithm 1, it is natural to preserve information in the network simplex algorithm and use warmstarts. While the CPLEX provides an interface to set a basis for a warmstart, LEMON does not. To make up for this, Schmitzer uses a trick that modifies the cost throughout the execution of Algorithm 1 with the LEMON library. Remarkably, he finds that LEMON outperforms CPLEX [4].

1.3   Our contribution

We provide a new C++-implementation of Schmitzer's [4] sparse multi-scale algorithm for dense optimal transport on grids and a network simplex implementation that is adapted from LEMON. Our program distinguishes itself from Schmitzer's in the following points. We adhere to LEMON's design choice and use compile-time polymorphisms (templates) instead of run-time polymorphisms for effiency. Moreover, we use many of the standard library utilities to avoid dynamically allocating memory and handling raw pointers manually to

---

[1] `https://www.ibm.com/products/ilog-cplex-optimization-studio`

avoid memory leaks and ensure memory safety. This results in a 3–4 times faster run-time compared to Schmitzer's solver on the DOTmark benchmark [5]. Our code is open source and publicly available on GitHub[2].

2 Results

──── **References** ────

1 Ravindra K. Ahuja, Thomas L. Magnanti, and James B. Orlin. *Network flows - theory, algorithms and applications.* Prentice Hall, 1993.

2 Balázs Dezso, Alpár Jüttner, and Péter Kovács. LEMON - an open source C++ graph template library. In Zoltán Porkoláb and Norbert Pataki, editors, *Proceedings of the Second Workshop on Generative Technologies, WGT@ETAPS 2010, Paphos, Cyprus, March 27, 2010*, volume 264 of *Electronic Notes in Theoretical Computer Science*, pages 23–45. Elsevier, 2010.

3 Shmuel Peleg, Michael Werman, and Hillel Rom. A unified approach to the change of resolution: Space and gray-level. *IEEE Trans. Pattern Anal. Mach. Intell.*, 11(7):739–742, 1989.

4 Bernhard Schmitzer. A sparse multiscale algorithm for dense optimal transport. *Journal of Mathematical Imaging and Vision*, 56:238–259, 2016.

5 Jörn Schrieber, Dominic Schuhmacher, and Carsten Gottschlich. DOTmark - A benchmark for discrete optimal transport. *IEEE Access*, 5:271–282, 2017.

6 Michael Werman, Shmuel Peleg, and Azriel Rosenfeld. A distance metric for multidimensional histograms. *Comput. Vis. Graph. Image Process.*, 32(3):328–336, 1985.

---

[2] `https://github.com/johannesrauch/GridOT/`