OPTIMAL TRANSPORT WITH PROXIMAL SPLITTING

NICOLAS PAPADAKIS*, GABRIEL PEYRÉ†, AND EDOUARD OUDET‡

Abstract. This article reviews the use of first order convex optimization schemes to solve the discretized dynamic optimal transport problem, initially proposed by Benamou and Brenier. We develop a staggered grid discretization that is well adapted to the computation of the L^2 optimal transport geodesic between distributions defined on a uniform spatial grid. We show how proximal splitting schemes can be used to solve the resulting large scale convex optimization problem. A specific instantiation of this method on a centered grid corresponds to the initial algorithm developed by Benamou and Brenier. We also show how more general cost functions can be taken into account and how to extend the method to perform optimal transport on a Riemannian manifold.

Key words. Optimal transport, proximal splitting, Douglas-Rachford, ADMM, primal-dual.

AMS subject classifications. 90C25, 68U10

1. Introduction.

Optimal transport is a well developed mathematical theory that defines a family of metrics between probability distributions [76]. These metrics measure the amplitude of an optimal displacement according to a so-called ground cost defined on the space supporting the distributions. The resulting distance is sometimes referred to as the Wasserstein distance in the case of L^p ground costs. The geometric nature of optimal transportation, as well as the ability to compute optimal displacements between densities, make this theory progressively mainstream in several applicative fields (see bellow). However, the numerical resolution of the optimal transportation problem raises several challenges. This article is focused on the computation of geodesics for the optimal transport metric associated to the L^2 cost. It reviews and extends the approach pioneered by Benamou and Brenier [8] from the perspective of proximal operator splitting in convex optimization. This shows the simplicity and efficiency of this method, which can easily be extended beyond the setting of optimal transport by considering various convex cost functions.

1.1. Previous Works.

Applications of optimal transport. Early successes of applications of optimal transport were mostly theoretical, such as for instance the derivation of functional inequalities [27] or the construction of solutions of some non-linear partial differential equations [52]. But recently, computational optimal transport as gained much interest and is progressively becoming mainstream in several applicative fields. In computer vision, the Wasserstein distance has been shown to outperform other metrics between distributions for machine learning tasks [69, 65]. In image processing, the warping provided by the optimal transport has been used for video restoration [31], color transfer [64], texture synthesis [39] and medical imaging registration [49]. It has also been applied to interpolation in computer graphics [13] and surface reconstruction in computational geometry [34]. Optimal transport is also used to model various physical phenomena, such as for instance in astrophysics [42] and oceanography [5].

Discrete optimal transport. The easiest way to discretize and compute numerically optimal transports is to consider finite sums of weighted Diracs. In this specific case, the optimal transport is a multi-valued map between the Diracs locations. Specific linear solvers can be used in this context and in particular network and transportation simplices [28] can scale up to a few thousands of Dirac masses. Dedicated combinatorial optimization methods have been proposed, such as the auction algorithm [11], which can handle integer costs between the Diracs. In the even more restricted case of two distributions with the same number of Diracs with equal weights, the transportation is a bijection between the points and thus corresponds to the optimal assignment problem [19]. Combinatorial optimization methods such as the Hungarian algorithm [47] have roughly cubic complexity in the number of Diracs. Faster schemes exist for specific cost functions, such as for instance convex cost of the distance on the line (where it boils down to a sorting of the positions) and the circle [32], concave costs on the line [33], the ℓ^1 distance [55]. The computation can

*IMB, Université Bordeaux 1, 351, cours de la Libération, F-33405 TALENCE, FRANCE.

†CEREMADE, Université Paris-Dauphine, Place du Marechal De Lattre De Tassigny, 75775 PARIS CEDEX 16, FRANCE.

‡LJK, Université de Grenoble, 51 rue des Mathématiques, Campus de Saint Martin d'Hères, BP 53, 38041 GRENOBLE CEDEX 09

be accelerated using multi-scale clustering [59]. Note also that various approximations of the transportation distance have been proposed, see for instance [71].

Despite being numerically intensive for finely discretized distributions, this discrete transport framework has found many applications, such as for instance color transfer between images [67], shape retrieval [69], surface reconstruction [29] and interpolation for computer graphics [13]

Optimal transport and PDE's. The optimal transport for the L^2 ground cost has a special structure. It can be shown to be uniquely defined and to be the gradient of a convex function [16]. This implies that it is also the solution of the fully non-linear Monge-Ampère partial differential equation. Several methods have been proposed to discretize and solve this PDE, such as for instance the method of [63] which converges to the Aleksandrov solution and the one of [62] which converges to the viscosity solution of the equation. Alternative methods such as [30] and [38] are efficient for regular densities. A major difficulty in these approaches is to deal with compactly supported densities, which requires a careful handing of the boundary conditions. [43] proposes to enforce these conditions by iteratively solving a Monge-Ampere equation with Neumann boundary conditions. [10] introduces a method requiring the solution of a well-posed Hamilton-Jacobi equation. Another line of methods iteratively constructs mass preserving mappings converging to the optimal transport [3]. This explicitly constructs the so-called polar factorization of the initial map, see also [6] for a different approach. This method is enhanced in [48] to avoid drifting from the preservation constraint during the iterations. These PDE's based approaches to the resolution of the optimal transport have found several applications, such as image registration [49], density regularization [18], optical flow [23] and grid generation [74].

Another line of research consists in using gradient flows where the gradient direction is computed according to the Wasserstein distance. This was initially proposed in [52] to build solutions to certain non-linear PDE's. This technique is now being used to design numerical approximation schemes for the solution of these equations, see for instance [21, 36, 40].

Dynamical optimal transport. Instead of computing directly the transport, it is possible to consider the geodesic path between the two densities according to the Wasserstein metric (the so-called displacement interpolation [57]). For the L^2 ground cost, this geodesic is obtained by linear interpolation between the identity and the transport. The geodesic can thus be computed by first obtaining the transport and then evolving the densities. If one considers discrete sums of Diracs, this corresponds to solving a convex linear program and can also be understood as a Lagrangian approximation of the transport between (possibly continuous) densities that have been discretized. This approach is refined in [51], which considers discretization with mixture of Gaussians.

It is also possible to consider an Eulerian formulation of the geodesic problem, for which densities along the path are discretized on a fixed spatial grid. Conservation of mass is achieved by introducing an incompressible velocity field transporting the densities. The breakthrough paper [8] shows that it is possible to perform a change of variable to obtain a convex problem. They propose to solve numerically the discretized problem with a first order iterative method. We give further details bellow on this method in the paragraph on proximal methods.

Geodesics between pairs of distributions can be extended to barycenters between an arbitrary finite collection of distributions. Existence and uniqueness of this barycenter is studied in [1]. Computing the barycenter between discrete distributions requires the resolution of a convex linear program that corresponds to a multi-marginal optimal transportation, as proved in [1]. However, in sharp contrast with the case of two distributions, the special case of un-weighted sums of Diracs is not anymore equivalent to an assignment problem, which is known to be NP-hard [19]. Computing numerically this barycenter for large scale problems can however be tackled using a non-convex formulation to solve for a Lagrangian discretization, which finds applications in image processing [68].

Generalized transport problems. The formulation of the geodesic computation as a convex optimization problem initiated by [8] enables the definition of various metrics obtained by changing the objective function. A penalization of the matching constraint [7] allows one to compute an unbalanced transport where densities are not normalized to have the same mass. An interpolation between the L^2-Wasserstein and L^2 distances is proposed in [9]. Lastly, an interpolation between L^2-Wasserstein and H^{-1} distances is described in [35]. This extension relies in a crucial manner on

the convexity of the extended objective function, which enables a theoretical analysis to characterize minimizing geodesics [20]. Convexity also allows one to use the numerical scheme we propose with only slight modifications with respect to the L^2-Wasserstein case, as we detail in Section 5.

Optimal transport on Riemannian manifolds. Many properties of the L^2-Wasserstein distance extend to the setting where the ground cost is the square of the geodesic distance on a Riemannian manifold. This includes in particular the existence and uniqueness of the transport map, which is the manifold exponential of the gradient of a semi-convex map [58]. Displacement interpolation for transport on manifolds has the same variational characterization as the one introduced in [9] for Euclidean transport, see [77] for a detailed review of optimal transport on manifolds. Interpolation between pairs of measures generalizes to barycenters of a family of measures, see [53].

Displacement interpolation between two measures composed, each one composed of a single Dirac, amounts to computing a single geodesic curve on the manifold. Discretization and numerical solutions to this problem are numerous. A popular method is the Fast Marching algorithm introduced jointly by [70, 75] for isotropic Riemannian metrics (i.e. when the metric at each point is a scalar multiple of the identity) discretized on a rectangular grid. The complexity of the method is $O(N \log(N))$ operations, where N is the number of grid points. This algorithm has been extended to compute geodesics on 2-D triangular meshes with only acute angles [54]. More general discretizations and the extension to Finsler metrics require the use of slower iterative schemes, see for instance [14].

Computing numerically optimal transport on manifolds has been less studied. For weighted sums of Diracs, displacement interpolation is achieved by solving the linear program to compute the coupling between the Diracs and then advancing the Diracs with the corresponding weights and constant velocity along the geodesics. In this article, we propose to extend the Eulerian discretization method [9] to solve for the displacement interpolation on a Riemannian manifold.

First order and proximal methods. The convex problem considered by Benamou and Brenier [8] can be re-casted as the optimization of a linear functional under second order conic constraints (see Section 3.6 for more details). This class of programs can be solved in time polynomial with the desired accuracy using interior points methods, see for instance [61].

However, the special structure of the problem, especially when discretized on an uniform grid, makes its suitable for first order schemes and in particular proximal splitting methods. While they do not reach the same convergence speed for arbitrary conic programs, they work well in practice for large scale problems, in particular when high accuracy is not mandatory, which is a common setup for problems in image processing. Proximal splitting schemes are first order optimization methods that allows one to minimize a sum of so-called "simple" functionals, possibly (for some methods) pre-composed by linear operators. A functional is called "simple" when it is possible to compute its proximal operator (see expression 4.1 for its precise definition) either in closed form, or with high accuracy using a few iterations of some sub-routine. In this article, we focus our attention to the Douglas-Rachford algorithm, introduced by [56] and on primal-dual methods. We make use of the recently proposed method [22], but other schemes could be used as well, see for instance [17]. We refer the reader to [26] and the references therein for more information about the properties of proximal maps and the associated proximal splitting schemes.

Note that the ALG2 algorithm proposed by [8] corresponds to applying the Alternating Direction Method of Multiplier (ADDM) [41] to the Fenchel-Rockafeller dual of the (primal) dynamical transport problem. As shown by [44, 37], this corresponds exactly to applying directly (a specific instanciation of) the Douglas-Rachford method to the primal problem, see Section 4.6 for more details.

Fluid mechanics discretization. While Lagrangian methods utilize a mesh-free discretization (see for instance [51]), thats typically tracks the movement of centers of masses during the transportation, Eulerian methods require a fixed discretization of the spatial domain. The most straightforward strategy is to use an uniform centered discretization of an axis-aligned domain, which is used in most previously cited works, see for instance [8, 3]. Because of the close connection between dynamical optimal transport and fluid dynamics, we advocate in this article the use of staggered grids [2], which better cope with the incompressibility condition.

1.2. Contributions. Our first contribution is to show how the method initially proposed in [8] is a specific instance of the Douglas-Rachford algorithm. This allows one to use several variations on the initial method, by changing the values of the two relaxation parameters and using different proximal splittings of the functional (possibly introducing auxiliary variables). Our second contribution is the introduction of a staggered grid discretization which is an efficient and convenient way to enforce incompressibility constraints. We show how this discretization fits into our proximal splitting methodology by introducing an interpolation operator and either making use of auxiliary variables or primal-dual methods. Our last contribution includes an exploration of several variations on the original convex transportation objective, the one proposed in [35] and a spatially varying penalization which can be interpreted as replacing the L^2 ground cost by a geodesic distance on a Riemannian manifold. Note that the Matlab source code to reproduce the figures of this article is available online[1].

2. Dynamical Optimal Transport Formulation.

2.1. Optimal Transport. In the following, we restrict our exposition to smooth maps $T : [0,1]^d \mapsto [0,1]^d$ where $d > 0$ is the dimension of the problem. A valid transport map T is a map that pushes forward the measure $f^0(x)\mathrm{d}x$ onto $f^1(x)\mathrm{d}x$. In term of densities, this corresponds to the constraint

$$f^0(x) = f^1(T(x))|\det(\partial T(x))|$$

where $\partial T(x) \in \mathbb{R}^{d\times d}$ is the differential of T at x. This is known as the gradient equation. We call $\mathcal{T}(f^0, f^1)$ the set of transport that satisfies this constraint. An optimal transport T solves

$$\min_{T\in\mathcal{T}(f^0,f^1)} \int C(x, T(x))\mathrm{d}x \tag{2.1}$$

where $C(x, y) \geqslant 0$ is the cost of assigning $x \in [0,1]^d$ to $y \in [0,1]^d$. In the case $C(x, y) = \|x-y\|^2$, the optimal value of (2.1), the so-called optimal transport distance, is often called the L^2-Wasserstein distance between the densities f^0 and f^1.

2.2. Fluid Mechanics Formulation. The geodesic path between the measures with densities $f^0(x)$ and $f^1(x)$ can be shown to have density $t \mapsto f(x,t)$ where $t \in [0,1]$ parameterizes the path, where

$$f(x,t) = f^0(T_t(x))|\det(\partial T_t(x))| \quad \text{where} \quad T_t = (1-t)\mathrm{Id}_d + tT.$$

Benamou and Brenier showed in [8] that this geodesic solves the following non-convex problem over the densities $f(x,t) \in \mathbb{R}$ and a velocity field $v(x,t) \in \mathbb{R}^2$

$$\min_{(v,f)\in\mathcal{C}^0} \frac{1}{2}\int_{[0,1]^d}\int_0^1 f(x,t)\|v(x,t)\|^2\mathrm{d}t\mathrm{d}x, \tag{2.2}$$

under the set of non-linear constraints

$$\mathcal{C}^0 = \left\{(v,f)\ ;\ \partial_t f + \mathrm{div}_x(fv) = 0,\ v(0,\cdot) = v(1,\cdot) = 0,\ f(\cdot,0) = f^0,\ f(\cdot,1) = f^1\right\}. \tag{2.3}$$

where the first relation in $\mathcal{C}^0$ is the continuity equation. We impose homogeneous Neumann conditions on the velocity field v which are the more natural boundary condition in the case of the square. Notice that both Neumann and Dirichlet boundary conditions can easily be implemented in our framework. The difference relies in the projection step on the divergence constraint. This step which is carried out using the Fast Fourier Transform algorithm, has to be adapted depending on the chosen boundary conditions. We refer to [43, 10] for relevant boundary conditions for other convex geometries. The temporal boundary constraints on f impose a match with the input density data.

From a theoretical point of view, the natural setting to prove existence of minimizers of (2.2) is to relax the variational problem and perform the optimization over the Banach space of Radon

[1] `https://github.com/gpeyre/2013-SIIMS-ot-splitting/`

measures (i.e. finite Borel measures). It is a sub-space of the space of distributions and the incompressibility constraint (2.3) should be understood in the sense of distributions. We refer the interested reader to [20] for more details regarding the theoretical analysis of a class of variational problems generalizing (2.2).

Note that once an optimal vector field v solving (2.2) has been computed, it is possible to recover an optimal transport T by integrating the flow in time. From a given $x \in [0,1]^d$, we define the solution $t \mapsto T_t(x)$ solving

$$T_0(x) = x \quad \text{and} \quad \forall\, t > 0, \quad \frac{\partial T_t(x)}{\partial t} = v(T_t(x), t).$$

The optimal transport is then obtained at $t = 1$, i.e. $T = T_1$, see [8] for more details.

Following [8], introducing the change of variable $(v, f) \mapsto (m, f)$, where m is the momentum $m = fv$, one obtains a convex optimization problem over the couple (f, m)

$$\min_{(m,f)\in\mathcal{C}} \mathcal{J}(m, f) = \int_{[0,1]^d} \int_0^1 J(m(x,t), f(x,t)) \mathrm{d}t\mathrm{d}x, \tag{2.4}$$

$$\text{where} \quad \forall\, (m, f) \in \mathbb{R}^d \times \mathbb{R}, \quad J(m, f) = \begin{cases} \frac{\|m\|^2}{2f} & \text{if } f > 0, \\ 0 & \text{if } (m, f) = (0, 0), \\ +\infty & \text{otherwise.} \end{cases} \tag{2.5}$$

and the set of linear constraints reads

$$\mathcal{C} = \left\{ (m, f) \,;\, \partial_t f + \mathrm{div}_x(m) = 0,\ m(0,\cdot) = m(1,\cdot) = 0,\ f(\cdot, 0) = f^0,\ f(\cdot, 1) = f^1 \right\}.$$

3. Discretized Dynamic Optimal Transport. For simplicity of exposure, we describe the discretization for the 1-D case. It extends verbatim to higher dimensional discretization $d > 1$.

3.1. Centered Grid. We denote $N + 1$ the number of discretization points in space, and $P + 1$ the number of discretization points in time. We introduce the centered grid discretizing the space-time square $[0, 1]^2$ in $(N + 1) \times (P + 1)$ points as

$$\mathcal{G}_{\mathrm{c}} = \left\{ (x_i = i/N,\, t_j = j/P) \in [0,1]^2 \,;\, 0 \leqslant i \leqslant N, 0 \leqslant j \leqslant P \right\}.$$

We denote

$$V = (m, f) \in \mathcal{E}_{\mathrm{c}} = (m_{i,j}, f_{i,j})_{0\leqslant i\leqslant N}^{0\leqslant j\leqslant P}$$

the variables discretized on the centered grid, where $\mathcal{E}_{\mathrm{c}} = (\mathbb{R}^{d+1})^{\mathcal{G}_{\mathrm{c}}} = (\mathbb{R}^2)^{\mathcal{G}_{\mathrm{c}}}$ is the finite dimensional space of centered variables.

3.2. Staggered Grid. The use of a staggered grid is very natural in the context of the discretization of a divergence operator associated to a vector field $u = (u_i)_{i=1}^d$ on $\mathbb{R}^d$ (we focus on the case $d = 2$ here). The basic idea is to allow an accurate evaluation of every partial derivative $\partial_{x_i} u_i$ at prescribed nodes of a cartesian grid using standard centered finite differences. One way to perform this computation is to impose to the grid on which the u_i scalar field is defined to be centered on P points along the x_i direction. This simple requirement forces the u_i scalar field to be defined on different grids. The resulting discrete vector field gives us the possibility to evaluate the divergence operator by a uniform standard centered scheme which is not possible using a single grid of discretization for every component of $(u_i)_i$. As a consequence, similarly to the discretization of PDE's in incompressible fluid dynamics (see for instance [50]), we consider a staggered grid discretization which is more relevant to deal with the continuity equation, and is defined as

$$\mathcal{G}_{\mathrm{s}}^x = \left\{ (x_i = (i + 1/2)/N,\, t_j = j/P) \in \frac{[-1, 2N + 1]}{2N} \times [0,1] \,;\, -1 \leqslant i \leqslant N, 0 \leqslant j \leqslant P \right\},$$

$$\mathcal{G}_{\mathrm{s}}^t = \left\{ (x_i = i/N,\, t_j = (j + 1/2)/P) \in [0,1] \times \frac{[-1, 2P + 1]}{2P} \,;\, 0 \leqslant i \leqslant N, -1 \leqslant j \leqslant P \right\}.$$

From these definitions, we see that $\mathcal{G}_{\mathrm{s}}^x$ contains $(N+2)\times(P+1)$ points and $\mathcal{G}_{\mathrm{s}}^t$ corresponds to a $(N+1)\times(P+2)$ discretization. We finally denote

$$U=(\bar{m},\bar{f})\in\mathcal{E}_{\mathrm{s}}=((\bar{m}_{i,j})_{-1\leqslant i\leqslant N}^{0\leqslant j\leqslant P},(\bar{f}_{i,j})_{0\leqslant i\leqslant N}^{-1\leqslant j\leqslant P})$$

the variables discretized on the staggered grid, where $\mathcal{E}_{\mathrm{s}}=\mathbb{R}^{\mathcal{G}_{\mathrm{s}}^x}\times\mathbb{R}^{\mathcal{G}_{\mathrm{s}}^t}$ is the finite dimensional space of staggered variables.

3.3. Interpolation and Divergence Operators. We introduce a midpoint interpolation operator $\mathcal{I}:\mathcal{E}_{\mathrm{s}}\to\mathcal{E}_{\mathrm{c}}$, where, for $U=(\bar{m},\bar{f})\in\mathcal{E}_{\mathrm{s}}$, we define $\mathcal{I}(U)=(m,f)\in\mathcal{E}_{\mathrm{c}}$ as

$$\forall\, 0\leqslant i\leqslant N,\quad \forall\, 0\leqslant j\leqslant P,\quad \left\{\begin{array}{ll} m_{i,j} & =(\bar{m}_{i-1,j}+\bar{m}_{i,j})/2,\\ f_{i,j} & =(\bar{f}_{i,j-1}+\bar{f}_{i,j})/2.\end{array}\right. \tag{3.1}$$

The space-time divergence operator $\mathrm{div}:\mathcal{E}_{\mathrm{s}}\to\mathbb{R}^{\mathcal{G}_{\mathrm{c}}}$ is defined, for $U=(\bar{m},\bar{f})\in\mathcal{E}_{\mathrm{s}}$ as

$$\forall\, 0\leqslant i\leqslant N,\ \forall\, 0\leqslant j\leqslant P,\quad \mathrm{div}(U)_{i,j}=N(\bar{m}_{i,j}-\bar{m}_{i-1,j})+P(\bar{f}_{i,j}-\bar{f}_{i,j-1}).$$

3.4. Boundary Constraints. We extract the boundary values on the staggered grid using the linear operator b, defined, for $U=(\bar{m},\bar{f})\in\mathcal{E}_{\mathrm{s}}$ as

$$b(U)=\big((\bar{m}_{-1,j},\bar{m}_{N,j})_{j=0}^{P},(\bar{f}_{i,-1},\bar{f}_{i,P})_{i=0}^{N}\big)\in\mathbb{R}^{P+1}\times\mathbb{R}^{P+1}\times\mathbb{R}^{N+1}\times\mathbb{R}^{N+1}.$$

We impose the following boundary values

$$b(U)=b_0\quad\text{where}\quad b_0=(0,0,f^0,f^1)\in\mathbb{R}^{P+1}\times\mathbb{R}^{P+1}\times\mathbb{R}^{N+1}\times\mathbb{R}^{N+1},$$

where $f^0,f^1\in\mathbb{R}^{N+1}$ are the discretized initial (time $t=0$) and final (time $t=1$) densities and the spatial boundary constraint $0\in\mathbb{R}^P$ on the momentum $m=fv$ comes from the discretized Neumann boundary conditions on the velocity field v. Notice that in the 2-D or 3-D cases, Neumann and Dirichlet Boundary conditions do not coincide. For instance in 2-D, Neumann conditions are equivalent to force the first component of $\bar{m}$ to be zero only on the vertical segments of the boundary whereas the second component vanishes only on the horizontal ones.

3.5. Discrete Convex Problem. The initial continuous problem (2.4) is approximated on the discretization grid by solving the finite dimensional convex problem

$$\min_{U\in\mathcal{E}_{\mathrm{s}}}\mathcal{J}(\mathcal{I}(U))+\iota_{\mathcal{C}}(U). \tag{3.2}$$

Here, for a closed convex set $\mathcal{C}$, we have denoted its associated indicator function

$$\iota_{\mathcal{C}}(U)=\left\{\begin{array}{ll} 0 & \text{if } U\in\mathcal{C},\\ +\infty & \text{otherwise.}\end{array}\right.$$

The discrete objective functional $\mathcal{J}$ reads for $V=(m,f)\in\mathcal{E}_{\mathrm{c}}$:

$$\mathcal{J}(V)=\sum_{k\in\mathcal{G}_{\mathrm{c}}}J(m_k,f_k), \tag{3.3}$$

where we denote $k=(i,j)\in\mathcal{G}_{\mathrm{c}}$ the indexes on the centered grid, and the functional J is defined in (2.5).

The constraint set $\mathcal{C}$ corresponds to the divergence-free constraint together with the boundary constraints

$$\mathcal{C}=\{U\in\mathcal{G}_{\mathrm{s}}\ ;\ \mathrm{div}(U)=0\quad\text{and}\quad b(U)=b_0\}\,.$$

3.6. Second Order Cone Programming Formulation. The discretized problem (3.2) can be equivalently solved as the following minimization over variables $(U, V, W) \in \mathcal{E}_{\mathrm{s}} \times \mathcal{E}_{\mathrm{c}} \times \mathbb{R}^{\mathcal{G}_{\mathrm{c}}}$

$$\min_{(V,W)\in\mathcal{K}, V=\mathcal{I}(U)} \sum_{k\in\mathcal{G}_{\mathrm{c}}} W_k \tag{3.4}$$

where $\mathcal{K}$ is the product of (rotated) Lorentz cones

$$\mathcal{K} = \left\{ (V = (\bar{m}, \bar{f})), W) \in \mathcal{E}_{\mathrm{c}} \times \mathbb{R}^{\mathcal{G}_{\mathrm{c}}} \;;\; \forall\, k \in \mathcal{G}_{\mathrm{c}},\; \|\bar{m}_k\|^2 - W_k \bar{f}_k \leqslant 0 \right\}.$$

Problem (3.4) is a specific instance of so-called second-order cone program (SOCP), that can be solved in time polynomial with the accuracy using interior point methods (see Section 1.1 for more details). As already mentioned in Section 1.1, we focus in this article on proximal splitting methods, that are more adapted to large scale imaging problems.

4. Proximal Splitting Algorithms. In this section, we review some splitting schemes and detail how they can be applied to solve (3.2). This requires to compute the proximal operators of the cost function J and of indicator of the constraint set $\mathcal{C}$.

4.1. Proximal Operators and Splitting Schemes. The minimization of a convex functional F over some Hilbert space $\mathcal{H}$ requires the use of algorithms that are tailored to the specific properties of the functional. Smooth functions can be minimized through the use of gradient descent steps, which amounts to apply repeatedly the mapping $\mathrm{Id}_{\mathcal{H}} - \gamma\nabla F$ for a small enough step size $\gamma > 0$. Such schemes can not be used for non-smooth functions such as the one considered in (3.2).

A popular class of first order methods, so-called proximal splitting schemes, replaces the gradient descent step by the application of the proximal operator. The proximal operator $\mathrm{Prox}_{\gamma F} : \mathcal{H} \to \mathcal{H}$ of a convex, lower semi-continuous and proper functional $F : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$ is defined as

$$\mathrm{Prox}_{\gamma F}(z) = \operatorname*{argmin}_{\tilde{z}\in\mathcal{H}} \frac{1}{2}\|z - \tilde{z}\|^2 + \gamma F(\tilde{z}). \tag{4.1}$$

This proximal operator is a single-valued map, which is related to the set-valued map of the subdifferential ∂F by the relationship $\mathrm{Prox}_{\gamma F} = (\mathrm{Id}_{\mathcal{H}} + \gamma\partial F)^{-1}$. This is why the application of $\mathrm{Prox}_{\gamma F}$ is often referred to as an implicit descent step, which should be compared with the explicit gradient descent step, $\mathrm{Id}_{\mathcal{H}} - \gamma\nabla F$.

Proximal operators enjoy several interesting algebraic properties which help the practitioner to compute it for complicated functionals. A typical example is the computation of the proximal operator of the Legendre-Fenchel transform of a function. The Legendre-Fenchel transform of F is defined as

$$F^*(w) = \max_{z\in\mathcal{H}} \langle z,\, w\rangle - F(z). \tag{4.2}$$

Note that thanks to Moreau's identity [60]

$$\forall\, w \in \mathcal{H}, \quad \mathrm{Prox}_{\gamma F^\star}(w) = w - \gamma\,\mathrm{Prox}_{F/\gamma}(w/\gamma), \tag{4.3}$$

computing the proximal operator of F^* has the same complexity as computing the proximal operator of F.

To enable the use of these proximal operators within an optimization scheme, it is necessary to be able to compute them efficiently. We call a function F such that $\mathrm{Prox}_{\gamma F}$ can be computed in closed form a *simple function*. It is often the case that the function to be minimized is not simple. The main idea underlying proximal splitting methods is to decompose this function into a sum of simple functions (possibly composed by linear operators). We detail bellow three popular splitting schemes, the Douglas-Rachford (DR) algorithm, the Alternating Direction Method of Multipliers (ADMM) and a primal-dual algorithm. We refer the reader to [26] for a detailed review of proximal operators and proximal splitting schemes.

4.2. Computing $\mathrm{Prox}_{\gamma\mathcal{J}}$. The following proposition shows that the functional $\mathcal{J}$ defined in (3.3) is simple, in the sense that its proximal operator can be computed in closed form.

PROPOSITION 1. *One has*

$$\forall\, V \in \mathcal{E}_c, \quad \mathrm{Prox}_{\gamma\mathcal{J}}(V) = (\mathrm{Prox}_{\gamma J}(V_k))_{k\in\mathcal{G}_c}$$

where, for all $(\tilde{m}, \tilde{f}) \in \mathbb{R}^d \times \mathbb{R}$,

$$\mathrm{Prox}_{\gamma J}(\tilde{m}, \tilde{f}) = \begin{cases} (\mu(f^\star), f^\star) & \text{if } f^\star > 0, \\ (0,0) & \text{otherwise.} \end{cases}$$

$$\text{where} \quad \forall\, f \geqslant 0, \quad \mu(f) = \frac{f\tilde{m}}{f+\gamma} \tag{4.4}$$

and $f^\star$ *is the largest real root of the third order polynomial equation in* X

$$P(X) = (X - \tilde{f})(X+\gamma)^2 - \frac{\gamma}{2}\|\tilde{m}\|^2 = 0. \tag{4.5}$$

Proof. We denote $(m, f) = \mathrm{Prox}_{\gamma J}(\tilde{m}, \tilde{f})$. If $f > 0$, since J is C^1 and is strongly convex on $\mathbb{R}^d \times \mathbb{R}^{+,*}$, necessarily (m, f) is the unique solution of $\nabla J(m, f) = 0$, which reads

$$\begin{cases} \gamma\frac{m}{f} + m - \tilde{m} = 0, \\ -\gamma\frac{\|m\|^2}{f^2} + f - \tilde{f} = 0. \end{cases}$$

Reformulating these equations leads to the following equivalent conditions

$$P(f) = 0 \quad \text{and} \quad m = \mu(f).$$

This shows that if P as at least a strictly positive real root $f^\star$, it is necessarily unique and that $(f = f^\star, m = \mu(f^\star))$. On the contrary, one necessarily has $f = 0$ and, by definition of J, $m = 0$ as well. □

4.3. Computing $\mathrm{Proj}_{\mathcal{C}}$. The proximal mapping of $\iota_{\mathcal{C}}$ is $\mathrm{Proj}_{\mathcal{C}}$ the orthogonal projector on the convex set $\mathcal{C}$. This is an affine set that can be written as

$$\mathcal{C} = \{U = (m, f) \in \mathcal{E}_{\mathrm{s}}\ ;\ AU = y\} \quad \text{where} \quad AU = (\mathrm{div}(U), b(U)) \quad \text{and} \quad y = (0, b_0).$$

This projection can be computed by solving a linear system as

$$\mathrm{Proj}_{\mathcal{C}} = (\mathrm{Id} - A^*\Delta^{-1}A) + A^*\Delta^{-1}y$$

where applying $\Delta^{-1} = (AA^*)^{-1}$ requires solving a Poisson equation on the centered grid $\mathcal{G}_{\mathrm{c}}$ with the prescribed boundary conditions. This can be achieved with Fast Fourier Transform in $O(NP\log(NP))$ operations where N and P are number of spatial and temporal points, see [73].

4.4. Douglas-Rachford Solver.

DR algorithm. The Douglas-Rachford (DR) algorithm [56] is a proximal splitting method that allows one to solve

$$\min_{z\in\mathcal{H}} G_1(z) + G_2(z) \tag{4.6}$$

where G_1 and G_2 are two simple functions defined on some Hilbert space $\mathcal{H}$.

The iterations of the DR algorithm define a sequence $(z^{(\ell)}, w^{(\ell)}) \in \mathcal{H}^2$ using a initial $(z^{(0)}, w^{(0)}) \in \mathcal{H}^2$ and

$$\begin{aligned} w^{(\ell+1)} &= w^{(\ell)} + \alpha(\mathrm{Prox}_{\gamma G_1}(2z^{(\ell)} - w^{(\ell)}) - z^{(\ell)}), \\ z^{(\ell+1)} &= \mathrm{Prox}_{\gamma G_2}(w^{(\ell+1)}). \end{aligned} \tag{4.7}$$

If $0 < \alpha < 2$ and $\gamma > 0$, one can show that $z^{(\ell)} \to z^\star$ a solution of (4.6), see [25] for more details.

In the following, we describe several possible ways to map the optimal transport optimization (3.2) into the splitting formulation (4.6), which in turn gives rise to four different algorithms.

Asymmetric-DR (A-DR) splitting scheme. We recast the initial optimal transport problem (3.2) as an optimization of the form (4.6) by using the variables

$$z = (U, V) \in \mathcal{H} = \mathcal{E}_{\mathrm{s}} \times \mathcal{E}_{\mathrm{c}}$$

and setting the functionals minimized as

$$\forall\, z = (U, V) \in \mathcal{E}_{\mathrm{s}} \times \mathcal{E}_{\mathrm{c}}, \quad G_1(z) = \mathcal{J}(V) + \iota_{\mathcal{C}}(U) \quad \text{and} \quad G_2(z) = \iota_{\mathcal{C}_{s,c}}(z).$$

In this expression, $\mathcal{C}_{s,c}$ is the constraint set that couples staggered and centered variables

$$\mathcal{C}_{s,c} = \{z = (U, V) \in \mathcal{E}_{\mathrm{s}} \times \mathcal{E}_{\mathrm{c}} \;;\; V = \mathcal{I}(U)\}$$

and $\mathcal{I}$ is the interpolation operator defined in (3.1).

Both G_1 and G_2 are simple functions since

$$\mathrm{Prox}_{\gamma G_1}(U, V) = (\mathrm{Proj}_{\mathcal{C}}(U), \mathrm{Prox}_{\gamma \mathcal{J}}(V)), \tag{4.8}$$

$$\mathrm{Prox}_{\gamma G_2}(U, V) = \mathrm{Proj}_{\mathcal{C}_{s,c}}(U, V) = (\tilde{U}, \mathcal{I}(\tilde{U})) \quad \text{where} \quad \tilde{U} = (\mathrm{Id} + \mathcal{I}^*\mathcal{I})^{-1}(U + \mathcal{I}^*(V)) \tag{4.9}$$

where $\mathcal{I}^*$ is the adjoint of the linear interpolation operator. Note that computing $\mathrm{Proj}_{\mathcal{C}_{s,c}}$ requires solving a linear system, but this system is separable along each dimension of the discretization grid, so it only requires solving a series of small linear systems. Furthermore, since the corresponding inverse matrix is the same along each dimension, we pre-compute explicitly the inverse of these $d + 1$ matrices.

In our case, the iterates variables appearing in (4.7) read $z^{(\ell)} = (U^{(\ell)}, V^{(\ell)})$, which allows one to retrieve an approximation $f^{(\ell)}$ of the transport geodesic as $U^{(\ell)} = (m^{(\ell)}, f^{(\ell)})$.

A nice feature of this scheme A-DR is that the iterates always satisfy $U^{(\ell)} \in \mathcal{C}$, but in general one does not have $V^{(\ell)} = \mathcal{I}(U^{(\ell)})$.

Asymmetric-DR' (A-DR') splitting scheme. In the DR algorithm (4.7), the functionals G_1 and G_2 do not play a symmetric role. One thus obtains a different algorithm (that we denote as A-DR'), by simply exchanging the definitions of G_1 and G_2 in (4.8). Using this scheme A-DR', one has $V^{(\ell)} = \mathcal{I}U^{(\ell)}$, but in general $U^{(\ell)}$ is not in the constraint set $\mathcal{C}$.

Symmetric-DR (S-DR) splitting scheme. In order to restore symmetry between the functionals $\mathcal{J}$ and $\iota_{\mathcal{C}}$ involved in the minimization algorithm, one can consider the following formulation

$$z = (U, V, \tilde{U}, \tilde{V}) \in \mathcal{H} = (\mathcal{E}_{\mathrm{s}} \times \mathcal{E}_{\mathrm{c}})^2$$

using the following functionals

$$G_1(z) = \mathcal{J}(V) + \iota_{\mathcal{C}}(U) + \iota_{\mathcal{C}_{s,c}}(\tilde{U}, \tilde{V}) \quad \text{and} \quad G_2(z) = \iota_{\mathcal{D}}(z),$$

where $\mathcal{D}$ is the diagonal constraint

$$\mathcal{D} = \left\{ z = (U, V, \tilde{U}, \tilde{V}) \in \mathcal{H} \;;\; U = \tilde{U} \quad \text{and} \quad V = \tilde{V} \right\}. \tag{4.10}$$

These functionals are simple since, for $z = (U, V, \tilde{U}, \tilde{V}) \in \mathcal{H}$, one has

$$\mathrm{Prox}_{\gamma G_1}(z) = \left(\mathrm{Proj}_{\mathcal{C}}(U), \mathrm{Prox}_{\gamma \mathcal{J}}(V), \mathrm{Proj}_{\mathcal{C}_{s,c}}(\tilde{U}, \tilde{V}) \right)$$

$$\text{and} \quad \mathrm{Prox}_{\gamma G_2}(z) = \frac{1}{2} \left(U + \tilde{U}, V + \tilde{V}, U + \tilde{U}, V + \tilde{V} \right).$$

The splitting reformulation of the form (4.10) was introduced in [72], and the corresponding DR scheme, extended to a sum of an arbitrary number of functionals, is detailed in [66, 24].

Symmetric-DR' (S-DR') splitting scheme. Similarely to the relationship between A-DR and A-DR' algorithm, it is possible to define a S-DR' algorithm by reversing the roles of G_1 and G_2 in the algorithm S-DR.

4.5. Primal-Dual Solver. Primal dual (PD) algorithms such as the relaxed Arrow-Hurwitz method introduced in [22] allows one to minimize functionals of the form $G_1 + G_2 \circ A$ where A is a linear operator and G_1, G_2 are simple functions. One can thus directly apply this method to problem (3.2) with $G_2 = \mathcal{J}$, $A = \mathcal{I}$ and $G_1 = \iota_{\mathcal{C}}$.

The iterations compute a sequence $(U^{(\ell)}, \Upsilon^{(\ell)}, V^{(\ell)}) \in \mathcal{E}_{\mathrm{s}} \times \mathcal{E}_{\mathrm{s}} \times \mathcal{E}_{\mathrm{c}}$ of variables from an initial $(\Upsilon^{(0)}, V^{(0)})$ according to

$$\begin{aligned} V^{(\ell+1)} &= \mathrm{Prox}_{\sigma G_2^*}(V^{(\ell)} + \sigma \mathcal{I} \Upsilon^{(\ell)}), \\ U^{(\ell+1)} &= \mathrm{Prox}_{\tau G_1}(U^{(\ell)} - \tau \mathcal{I}^* V^{(\ell+1)}), \\ \Upsilon^{(\ell+1)} &= U^{(\ell+1)} + \theta(U^{(\ell+1)} - U^{(\ell)}). \end{aligned} \tag{4.11}$$

Note that $\mathrm{Prox}_{\sigma G_2^*}$ can be computed using Prox_{G_2} following equation (4.3). If $0 \leqslant \theta \leqslant 1$ and $\sigma\tau\|\mathcal{I}\|^2 < 1$ then one can prove that $U^{(\ell)} \to U^\star$ which is a solution of (3.2), see [22].

The case $\theta = 0$ corresponds to the Arrow-Hurwicz algorithm [4], and the general case can be interpreted as a preconditioned version of the ADDM algorithm, as detailed in [22].

4.6. ADMM Solver on Centered Grid. In this Section, we give some details about the relationship between the algorithm ALG2 developed by Benamou and Brenier in [8] and our DR algorithms. The original paper [8] considers a finite difference implementation on a centered grid, which leads to solve the following optimization problem

$$\min_{V \in \mathcal{E}_{\mathrm{c}}} \mathcal{J}(V) + \iota_{\tilde{\mathcal{C}}}(V) \quad \text{where} \quad \tilde{\mathcal{C}} = \{V \in \mathcal{E}_{\mathrm{s}} \;;\; AV = y\} \tag{4.12}$$

where A is this time defined on a centered grid.

Primal problem. The minimization (4.12) has the form

$$\min_{z \in \mathcal{H}} F \circ A(z) + G(z), \tag{4.13}$$

where $A : \mathcal{H} \to \tilde{\mathcal{H}}$ is a linear operator, and $F \circ A$ and G are supposed to be simple functions.

For the OT problem on a centered grid (4.12), the identification is obtained by setting

$$F = \iota_{\{y\}} \quad \text{and} \quad G = \mathcal{J},$$

which are indeed simple functions. One can use the DR algorithm (4.7) with

$$G_1 = F \circ A \quad \text{and} \quad G_2 = G \tag{4.14}$$

to solve problems of the form (4.13). Of course, the variants A-DR', S-DR and S-DR' considered in Section 4.4 could be used as well on a centered grid. Since we focus on the relationship with the work [8], we only consider the splitting (4.14).

Dual problem. Following [8], one can consider the Fenchel-Rockafeller dual to the primal program (4.13), which reads

$$\max_{s \in \tilde{\mathcal{H}}} \; -\left(G^* \circ (-A^*)(s) + F^*(s)\right). \tag{4.15}$$

In the specific setting of the OT problem (4.12), $F^*(p) = \langle y, p \rangle$ and the following proposition, proved in [8], shows that $G^* = \mathcal{J}^*$ is a projector on a convex set (which is a consequence of the fact that $\mathcal{J}$ is a 1-homogenous functional).

Proposition 2. *One has*

$$\mathcal{J}^* = \iota_{\mathcal{C}_{\mathcal{J}}} \quad \textit{where} \quad \begin{cases} \mathcal{C}_{\mathcal{J}} = \{V \in \mathcal{E}_c \;;\; \forall\, k \in \mathcal{G}_c,\; V_k \in \mathcal{C}_J\}, \\ \mathcal{C}_J = \left\{(a,b) \in \mathbb{R}^d \times \mathbb{R} \;;\; \|a\|^2 + 2b \leqslant 0\right\}. \end{cases}$$

Proof. The Lengendre-Fenchel transform (4.2) of the functional J defined in (2.5) at point $(a,b) \in \mathbb{R}^d \times \mathbb{R}$ reads

$$J^*(a,b) = \max_{(m,f) \in \mathbb{R}^d \times \mathbb{R}} \langle a,\, m \rangle + \langle b,\, f \rangle - \frac{m^2}{2f},$$

where we just focus on the case $f > 0$, since $f = 0$ (resp. $f < 0$) will always give $J^* = 0$ (resp. $J^* = -\infty$). The optimality conditions are given by

$$a = \frac{m}{f} \quad \text{and} \quad b = -\frac{m^2}{2f^2}.$$

Hence, we have that

$$\begin{aligned} J^*(a,b) &= \max_{(m,f)\in\mathcal{E}_c} f(\langle a, \frac{m}{f}\rangle + b) - \frac{m^2}{2f^2} \\ &= \max_{(m,f)\in\mathcal{E}_c} f\left(\langle a, a\rangle + 2b\right), \end{aligned}$$

which is 0 if $\|a\|^2 + 2b \leqslant 0$ and $+\infty$ otherwise. □

Note that one can use the Moreau's identity (4.3) to compute the proximal operator of J using the orthogonal projection on $\mathcal{C}_J$ and vice-versa

$$\mathrm{Prox}_{\gamma J}(v) = v - \gamma\,\mathrm{Proj}_{\mathcal{C}_J}(v/\gamma).$$

ADMM method. The Alternating Direction Method of Multipliers (ADMM) is an algorithm to solve a minimization of the form

$$\min_{s\in\tilde{\mathcal{H}}} H \circ B(s) + K(s), \tag{4.16}$$

where we assume that the function $K^* \circ B^*$ is simple and that $B : \tilde{\mathcal{H}} \to \mathcal{H}$ is injective. It was initially proposed in [45, 46].

We introduce the Lagrangian associated to (4.16) to account for an auxiliary variable $q \in \mathcal{H}$ satisfying $q = Bs$ with a multiplier variable $v \in \mathcal{H}$

$$\forall\,(s,q,v) \in \tilde{\mathcal{H}} \times \mathcal{H} \times \mathcal{H}, \quad L(s,q,v) = K(s) + H(q) + \langle v,\, Bs - q\rangle,$$

and the augmented Lagrangian for $\gamma > 0$

$$\forall\,(s,q,v) \in \tilde{\mathcal{H}} \times \mathcal{H} \times \mathcal{H}, \quad L_\gamma(s,q,v) = L(s,q,v) + \frac{\gamma}{2}\|Bs - q\|^2$$

The ADMM algorithm generates iterates $(s^{(\ell)}, q^{(\ell)}, v^{(\ell)}) \in \tilde{\mathcal{H}} \times \mathcal{H} \times \mathcal{H}$ as follow

$$\begin{aligned} s^{(\ell+1)} &= \operatorname*{argmin}_s L_\gamma(s, q^{(\ell)}, v^{(\ell)}), \\ q^{(\ell+1)} &= \operatorname*{argmin}_q L_\gamma(s^{(\ell+1)}, q, v^{(\ell)}), \\ v^{(\ell+1)} &= v^{(\ell)} + \gamma(Bs^{(\ell+1)} - q^{(\ell+1)}). \end{aligned} \tag{4.17}$$

This scheme can be shown to converge for any $\gamma > 0$, see [45, 46]. A recent review of the ADMM algorithm and its applications in machine learning can be found in [15].

We introduce the proximal operator $\mathrm{Prox}^B_{\gamma F} : \mathcal{H} \to \tilde{\mathcal{H}}$ with a metric induced by an injective linear map B

$$\forall\, z \in \mathcal{H}, \quad \mathrm{Prox}^B_{K/\gamma}(z) = \operatorname*{argmin}_{s\in\tilde{\mathcal{H}}} \frac{1}{2}\|Bs - z\|^2 + \frac{1}{\gamma}K(s). \tag{4.18}$$

One can then re-write the ADMM iterations (4.17) using proximal maps

$$\begin{aligned} s^{(\ell+1)} &= \mathrm{Prox}^B_{K/\gamma}(q^{(\ell)} - u^{(\ell)}), \\ q^{(\ell+1)} &= \mathrm{Prox}_{H/\gamma}(Bs^{(\ell+1)} + u^{(\ell)}), \\ u^{(\ell+1)} &= u^{(\ell)} + Bs^{(\ell+1)} - q^{(\ell+1)}. \end{aligned} \tag{4.19}$$

where we have performed the change of variable $u^{(\ell)} = v^{(\ell)}/\gamma$ to simplify the notations.

The following proposition shows that if $K^* \circ B^*$ is simple, one can indeed perform the ADMM algorithm. Note that in the case $\mathcal{H} = \tilde{\mathcal{H}}$ and $B = \mathrm{Id}_{\mathcal{H}}$, one recovers Moreau's identity (4.3).

PROPOSITION 3. *One has*

$$\forall\, z \in \mathcal{H}, \quad \mathrm{Prox}^B_{K/\gamma}(z) = B^+ \left(z - \frac{1}{\gamma} \mathrm{Prox}_{\gamma K^* \circ B^*}(\gamma z) \right).$$

Proof. We denote $\mathcal{U} = \partial K$ the set-valued maximal monotone operator. One has $\partial K^* = \mathcal{U}^{-1}$, where $\mathcal{U}^{-1}$ is the set-valued inverse operator, and $\partial(K^* \circ B^*) = B \circ \mathcal{V} \circ B^*$. One has $\mathrm{Prox}_{\gamma K^* \circ B^*} = (\mathrm{Id} + \gamma \mathcal{V})^{-1}$, which a single-valued operator. Denoting $s = \mathrm{Prox}^B_{K/\gamma}(z)$, the optimality conditions of (4.18) lead to

$$0 \in B^*(Bs - z) + \frac{1}{\gamma}\mathcal{U}(s) \Longleftrightarrow s \in \mathcal{U}^{-1}\left(\gamma B^* B s - \gamma B^* z\right) \Longleftrightarrow \gamma B s \in \gamma \mathcal{V}\left(\gamma z - \gamma B s\right) \tag{4.20}$$

$$\Longleftrightarrow \gamma B s \in (\mathrm{Id} + \gamma \mathcal{V})\left(\gamma z - \gamma B s\right) + \gamma B s - \gamma z \Longleftrightarrow \gamma z \in (\mathrm{Id} + \gamma \mathcal{V})\left(\gamma z - \gamma B s\right) \tag{4.21}$$

$$\Longleftrightarrow \gamma z - \gamma B s = (\mathrm{Id} + \gamma \mathcal{V})^{-1}(\gamma z) \Longleftrightarrow s = B^+ \left(z - \frac{1}{\gamma}(\mathrm{Id} + \gamma \mathcal{V})^{-1}(\gamma z) \right) \tag{4.22}$$

where we used the fact that B is injective. □

Equivalence between ADMM and DR. The ALG2 algorithm of [8] corresponds to applying the ADMM algorithm to the dual problem (4.15) that can be formulated as

$$\min_{s \in \tilde{\mathcal{H}}} \left(G^* \circ (-A^*)(s) + F^*(s) \right), \tag{4.23}$$

while retrieving at each iteration the primal iterates in order to solve the primal problem (4.13). The following proposition, which was initially proved in [44], shows that applying ADMM algorithm to the dual (4.15) is equivalent to solving the primal (4.13) using DR. More precisely, the dual variable (the Lagrange multiplier) $v^{(\ell)}$ of the ADMM iterations is equal to the primal variable $z^{(\ell)}$ of the DR iterations. This result was further extended by [37] which shows the equivalence of ADMM with the proximal point algorithm and propose several generalizations. For the sake of completeness, we detail the proof of this result using our own notations.

PROPOSITION 4. *Setting*

$$H = G^*, \quad K = F^*, \quad \textit{and} \quad B = -A^*, \tag{4.24}$$

the DR iterations (4.7) *with functionals* (4.14) *and* $\alpha = 1$ *are recovered from the ADMM iterations* (4.17) *using*

$$\begin{aligned} v^{(\ell)} &= z^{(\ell)}, \\ \gamma q^{(\ell)} &= w^{(\ell)} - z^{(\ell)}, \\ \gamma A^* s^{(\ell+1)} &= z^{(\ell)} - w^{(\ell+1)}. \end{aligned}$$

Proof. Denoting $\bar{s}^{(\ell)} = A^* s^{(\ell)} \in \mathrm{Im}(A^*)$ (recall that $B = -A^*$ is injective), using the change of notations (4.24) and using the result of Proposition 3, the iterates (4.19) read

$$\begin{aligned} -\bar{s}^{(\ell+1)} &= q^{(\ell)} - u^{(\ell)} + \frac{1}{\gamma} \mathrm{Prox}_{\gamma F \circ A}\left(\gamma(u^{(\ell)} - q^{(\ell)}) \right), \\ q^{(\ell+1)} &= u^{(\ell)} - \bar{s}^{(\ell+1)} - \frac{1}{\gamma} \mathrm{Prox}_{\gamma G}\left(\gamma(u^{(\ell)} - \bar{s}^{(\ell+1)}) \right), \\ u^{(\ell+1)} &= u^{(\ell)} - \bar{s}^{(\ell+1)} - q^{(\ell+1)} \end{aligned} \tag{4.25}$$

where we have used the fact that $\mathrm{Prox}_{\gamma F \circ (-A)}(z) = -\mathrm{Prox}_{\gamma F \circ A}(-z)$. Recall that the DR iterations (4.7) read in the setting $\alpha = 1$

$$\begin{aligned} w^{(\ell+1)} &= w^{(\ell)} + \mathrm{Prox}_{\gamma F \circ A}(2 z^{(\ell)} - w^{(\ell)}) - z^{(\ell)}, \\ z^{(\ell+1)} &= \mathrm{Prox}_{\gamma G}(w^{(\ell+1)}). \end{aligned} \tag{4.26}$$

Iterations (4.25) and (4.26) are using the following identification between $(q^{(\ell)}, q^{(\ell+1)}, u^{(\ell)}, \bar{s}^{(\ell+1)})$ and $(w^{(\ell)}, z^{(\ell)}, w^{(\ell+1)}, z^{(\ell+1)})$

$$\gamma(u^{(\ell)} - q^{(\ell)}) = 2z^{(\ell)} - w^{(\ell)}, \tag{4.27}$$
$$\gamma(u^{(\ell)} - \bar{s}^{(\ell+1)}) = w^{(\ell+1)}, \tag{4.28}$$
$$\gamma(u^{(\ell)} - q^{(\ell)} - \bar{s}^{(\ell+1)}) = w^{(\ell+1)} + z^{(\ell)} - w^{(\ell)}, \tag{4.29}$$
$$\gamma(u^{(\ell)} - q^{(\ell+1)} - \bar{s}^{(\ell+1)}) = z^{(\ell+1)}, \tag{4.30}$$

Solving the linear system (4.27)-(4.30), one gets

$$\gamma u^{(\ell)} = z^{(\ell)}, \tag{4.31}$$
$$\gamma q^{(\ell)} = w^{(\ell)} - z^{(\ell)}, \tag{4.32}$$
$$\gamma \bar{s}^{(\ell+1)} = z^{(\ell)} - w^{(\ell+1)}, \tag{4.33}$$
$$\gamma q^{(\ell+1)} = w^{(\ell+1)} - z^{(\ell+1)}. \tag{4.34}$$

First notice that relations (4.32) and (4.34) are compatible at iterations ℓ and $\ell+1$. Then, identifying the update of the variable u presented in (4.25) with relation (4.30), we recover $z^{(\ell+1)} = \gamma u^{(\ell+1)} = v^{(\ell+1)}$, which corresponds to the relation (4.31) at iteration $\ell+1$. □

This proposition thus shows that, on a centered grid, ALG2 of [8] gives the same iterates as DR on the primal problem. Since we consider a staggered grid, the use of the interpolation operator makes our optimization problem (4.6) different from the original ALG2 and requires the introduction of an auxiliary variable V. Furthermore, the introduction of an extra relaxation parameter α is useful to speed-up the convergence of the method, as will be established in the experimentations. Lastly, let us recall that it is possible to use the variants A-DR', S-DR and S-DR' of the initial A-DR formulation, as detailed in Section 4.4.

5. Generalized Cost Functions. Following [35, 20], we propose to use a generalized cost function that allows one to compute geodesics that interpolate between the L^2-Wasserstein and the H^{-1} geodesics. To introduce further flexibility, we introduce spatially varying weights, which corresponds to approximating a transportation problem on a Riemannian manifold.

5.1. Spatially Varying Interpolation between L^2-Wasserstein and H^{-1} . We define our discretized generalized transportation problem as

$$\min_{U \in \mathcal{E}_{\mathrm{s}}} \mathcal{J}_\beta^w(\mathcal{I}(U)) + \iota_{\mathcal{C}}(U), \tag{5.1}$$

where the vector of weights $w = (w_k)_{k \in \mathcal{G}_{\mathrm{c}}}$ satisfies $c < w_k \leqslant +\infty$ and $c > 0$ is a small constant. Note that we allow for infinite weights $w_k = +\infty$, which corresponds to forbidding the transport to put mass in a given cell indexed by k. The generalized functional reads, for $\beta \geqslant 0$

$$\mathcal{J}_\beta^w(V) = \sum_{k \in \mathcal{G}_{\mathrm{c}}} w_k J_\beta(m_k, f_k), \tag{5.2}$$

$$\forall\, (m,f) \in \mathbb{R}^d \times \mathbb{R}, \quad J_\beta(m,f) = \begin{cases} \frac{\|m\|^2}{2f^\beta} & \text{if } f > 0, \\ 0 & \text{if } (m,f) = (0,0), \\ +\infty & \text{otherwise.} \end{cases} \tag{5.3}$$

Note that $\mathcal{J}_1^1 = \mathcal{J}$ and that for $\beta \in [0,1]$, $\mathcal{J}_\beta^w$ is convex. Furthermore, one has, for $f > 0$

$$\det(\partial^2 J_\beta(m,f)) = \frac{\beta(1-\beta)\|m\|^2}{f^{3\beta+2}}. \tag{5.4}$$

This shows that J_β is strictly convex on $\mathbb{R}^{*,d} \times \mathbb{R}^{+,*}$ for $\beta \in]0,1[$.

The case of constant weights $w_k = 1$ is studied in [35, 20]. The case $\beta = 1$ corresponds to the Wasserstein L^2 distance. In a continuous (not discretized) domain, the value of the problem (5.1) is equal to the H^{-1} Sobolev norm over densities $\|f_0 - f_1\|_{H^{-1}}$, as detailed in [35]. Note that since in this case the induced distance is actually an Hilbertian norm, the corresponding geodesic is a linear interpolation of the input measures, and thus for measures having densities, one obtains $f_t = (1-t)f_0 + tf_1$.

When $\beta = 1$ and the weights w_k are constant in time, the solution of (5.2) discretizes the displacement interpolation between the densities (f^0, f^1) for a ground cost being the squared geodesic distance on a Riemannian manifold (see Section 1.1 for more details). Note that we restrict our attention to isotropic Riemannian metrics (being proportional to the identity at each point), but this extends to arbitrary Riemannian or even Finsler metrics. Studying the properties of this transportation distance is however not the purpose of this work, and we show in Section 6.3 numerical illustrations of the influence of w.

5.2. Computing $\mathrm{Prox}_{\gamma J_\beta}$. The following proposition, which generalizes Proposition 1, shows how to compute $\mathrm{Prox}_{\gamma \mathcal{J}_\beta^w}$.

PROPOSITION 5. *One has*

$$\forall\, V \in \mathcal{E}_c, \quad \mathrm{Prox}_{\gamma \mathcal{J}_\beta^w}(V) = (\mathrm{Prox}_{\gamma w_k J_\beta}(V_k))_{k \in \mathcal{G}_c}$$

where, for all $(\tilde{m}, \tilde{f}) \in \mathbb{R}^d \times \mathbb{R}$,

$$\mathrm{Prox}_{\gamma J_\beta}(\tilde{m}, \tilde{f}) = \begin{cases} (\mu(f^\star), f^\star) & \text{if } f^\star > 0 \text{ and } \gamma < \infty, \\ (0,0) & \text{otherwise.} \end{cases}$$

$$\text{where} \quad \forall\, f \geqslant 0, \quad \mu(f) = \frac{f^\beta \tilde{m}}{f^\beta + \gamma} \in \mathbb{R}^d \tag{5.5}$$

and $f^\star$ *is the largest real root of the following equation in* X

$$P_\beta(X) = X^{1-\beta}(X - \tilde{f})(X^\beta + \gamma)^2 - \frac{\gamma}{2}\beta\|\tilde{m}\|^2 = 0 \tag{5.6}$$

Proof. We denote $(\bar{m}, \bar{f}) = \mathrm{Prox}_{\gamma J_\beta}(\tilde{m}, \tilde{f})$, and

$$\forall\, (m, f) \in \mathbb{R}^d \times \mathbb{R}, \quad \Phi_\beta(m, f) = \frac{1}{2}\|(m, f) - (\tilde{m}, \tilde{f})\|^2 + \gamma J_\beta(m, f).$$

If $\bar{f} > 0$, since Φ_β is C^1 and is strongly convex on $\mathbb{R}^d \times \mathbb{R}^{+,*}$, necessarily $(\bar{m}, \bar{f})$ is the unique solution of $\nabla \Phi_\beta(\bar{m}, \bar{f}) = 0$, which reads

$$\begin{cases} \gamma \frac{\bar{m}}{\bar{f}^\beta} + \bar{m} - \tilde{m} = 0 \\ -\gamma\beta \frac{\|\bar{m}\|^2}{2\bar{f}^{\beta+1}} + \bar{f} - \tilde{f} = 0. \end{cases}$$

Reformulating these equations leads to the following equivalent conditions

$$P_\beta(\bar{f}) = 0 \quad \text{and} \quad \bar{m} = \mu_\beta(\bar{f}).$$

This shows that if P_β as at least a strictly positive real root $f^\star$, it is necessarily unique and that $(\bar{f} = f^\star, \bar{m} = \mu_\beta(f^\star))$. Otherwise, necessarily $(\bar{f} = 0, \bar{m} = 0)$. □

Note that when $\beta = p/q \in \mathbb{Q}$ is a rational number, equation $P_\beta(X) = 0$ corresponds to finding the root of a polynomial. It can be solved efficiently using a few Newton descent iterations starting from a large enough value of f.

6. Numerical Simulations.

6.1. Comparison of Proximal Schemes. This section compares the following algorithms introduced in Section 4:

- Douglas-Rachford (DR in the following) as exposed in Section 4.4, parameterized with α and γ ;
- ADDM on the dual (ADDM in the following, which is DR with $\alpha = 1$) parameterized with γ ;
- Primal-dual (PD in the following) as exposed in Section 4.5, parameterized with σ and τ.

Note that this ADMM formulation is related to the ALG2 method introduced in [8], but is computed over a staggered grid. For the DR algorithm, we first compared the 4 possible implementations previously described. It appears in our experiments that A-DR and A-DR' (resp. S-DR and S-DR') have almost the same behavior.

The first comparison is done on a simple example with two 2-D isotropic Gaussian distributions (f^0, f^1) with the same variance. In the continuous case, the solution is known to be a translation between the mean of the Gaussians. The spatial domain is here of dimension $d = 2$ and it is discretized on a grid with $N = M = 32$ points for both each dimension. The temporal discretization has also been fixed to $P = 32$. We first compute an (almost) exact reference solution $(m^\star, f^\star)$ of the discrete problem with 10^6 iterations of the DR. The obtained transported mass $f^\star(\cdot, t)$ is illustrated in Figure 6.1. Regarding the computation time, with a bi-processor system Intel Core i7 with 2.4 GHz, 1000 iterations are done in 45 seconds for a 32^3 domain with our Matlab implementation.

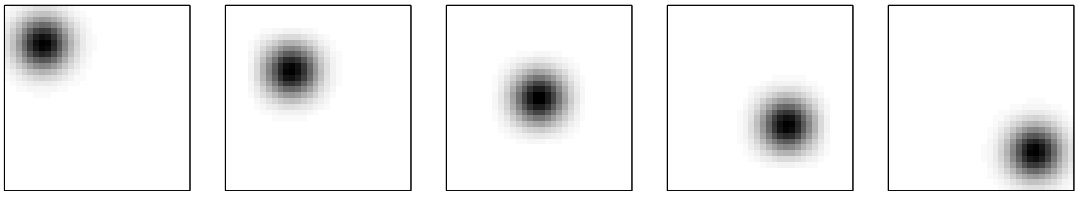

FIG. 6.1. *Display of $f^\star(\cdot, t)$ for several value of t (note that for $t = 0$ and $t = 1$, this corresponds to f^0 and f^1). The grayscale values are linearly interpolating from black to white between 0 and the maximum value of $f^\star$.*

For each algorithm, we perform an exhaustive search of the best possible set of parameters. These optimal parameters are those minimizing $\|(m^\star, f^\star) - (m^{(\ell)}, f^{(\ell)})\|$, the ℓ^2 distance between $f^\star$ and the output of the algorithm after $\ell = 500$ iterations. The optimal parameters for this data set are: $\gamma = 1/80$ for ADMM on the dual, $(\gamma = 1/75, \alpha = 1.998)$ for DR and $\sigma = 85$ for PD. For PD, we found that simply setting $\tau = \frac{0.99}{\sigma \|\mathcal{I}\|^2}$ leads to almost optimal convergence rate in our tests, so we use this rule to only introduce a single parameter σ. Notice that this parameter choice is within the range of parameters $\sigma\tau\|\mathcal{I}\|^2 < 1$ that guaranties convergence of the PD method. Figure 6.2 displays, for this optimal choice of parameters, the evolution of the cost function value as well as the convergence on the staggered grid toward $(m^\star, f^\star)$ as a function of the iteration number ℓ.

One can observe that the quality of the approximation can not easily be deduced from the cost function evolution alone since the functional is very flat. Indeed, an almost minimal value of the function is reached by all the algorithms after roughly 10^3 iterations, whereas the ℓ^2 distance to the reference solution continues to decrease almost linearly in log-log scale. The last plot of the Figure 6.2 shows that asymptotically, all methods tend to satisfy the positivity constraint on the staggered grid at the same rate.

When comparing the three approaches, A-DR and A-PD shows the fastest convergence rate to the reference solution and then, the S-DR algorithm performs equally good as ADMM. This shows the advantage of introducing the α parameter, while symmetrizing the DR does not speed-up the convergence. Notice also that the computational cost is smaller for PD, as it takes 0.13s for one PD iteration and $0.2s$ for one DR or ADMM iteration for this example. Note that these convergence results are related to this specific example, but they illustrate the general behaviour of the different algorithms.

Finally, Figure 6.3 shows an experiment in the context of vanishing and irregular densities. This figure shows the geodesic, computed with the PD algorithm, between two characteristic functions of two connected sets, one being convex. Note that the geodesic is not composed of characteristic

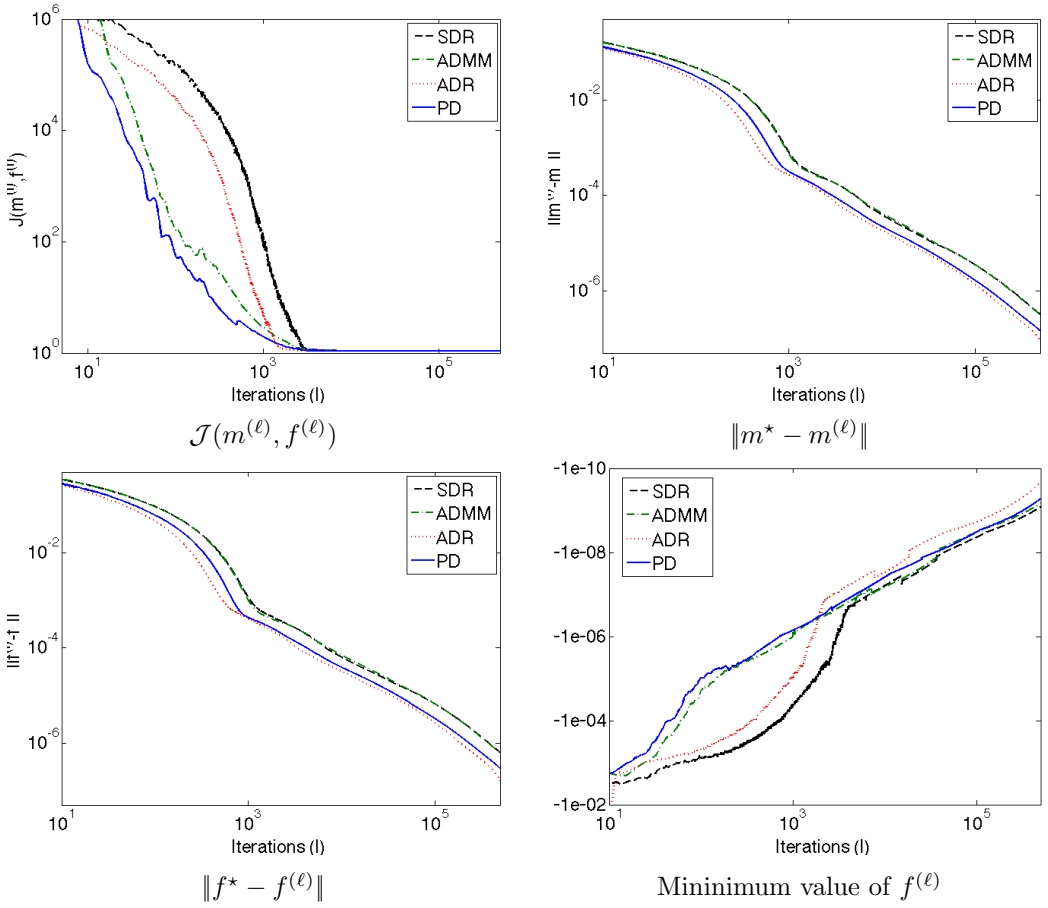

FIG. 6.2. *At each iteration ℓ on the staggered grid, we plot in the log-log scale the value of the cost function $\mathcal{J}$, the distance between the reference solution $(m^\star, f^\star)$ and the estimation $(m^{(\ell)}, f^{(\ell)})$ and the current minimum value of $f^{(\ell)}$ for the different proximal splitting algorithms with the best found parameters.*

functions of sets, which is to be expected. This shows the ability of our methods to cope with vanishing densities.

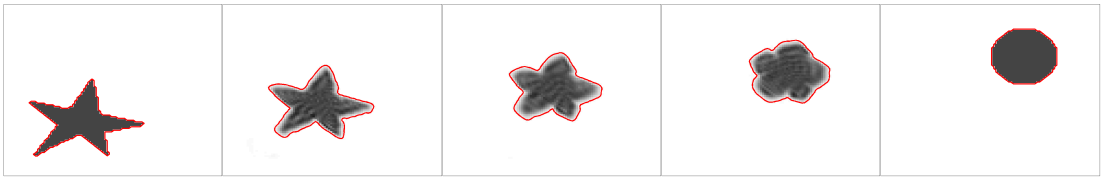

FIG. 6.3. *Transport between characteristic functions. Evolution of $f^\star(\cdot, t)$ for several values of t. The red curve denotes the boundary the area with positive density.*

6.2. Interpolation Between L^2-Wasserstein and H^{-1}. We first apply the PD algorithm for different values of β on the bump example introduced in the previous section. The results are presented in the Figure 6.4, which shows the level-lines of the estimated densities $f^{(\ell)}(\cdot, t)$ for $\ell = 1000$ iterations. It shows the evolution of the solution between a linear interpolation of the densities ($\beta = 0$) and a displacement interpolation with transport ($\beta = 1$).

Figure 6.5 shows for $\beta = 1/2$ and $\beta = 3/4$ the evolution of the cost function with the iterations index ℓ, together with the convergence of the estimate to the reference solution $(m^\star, f^\star)$ (obtained after 10^5 iterations of the PD algorithm). We can observe that the behavior of the process with $\beta \in]0, 1[$ is different than the one observed for $\beta = 1$ in Figure 6.2. Indeed, we observe a faster convergence of the functional value, which is consistent with the fact that $\mathcal{J}_\beta$ becomes more and more strongly convex as β approaches $1/2$ (see (5.4)). The oscillations come from the Newton's

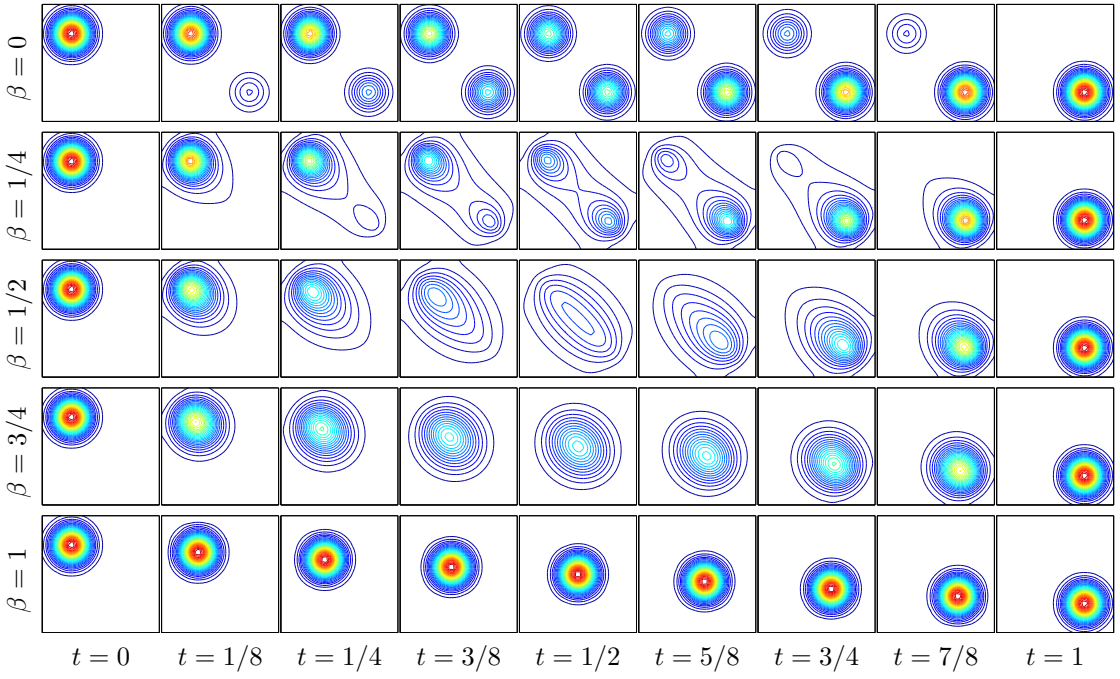

FIG. 6.4. *Display of the level sets of $f^{(\ell)}(\cdot, t)$ for several value of t and β (note that for $t = 0$ and $t = 1$, this corresponds to f^0 and f^1).*

descent that only approximate the computation of $\mathrm{Prox}_{\gamma J_\beta}$.

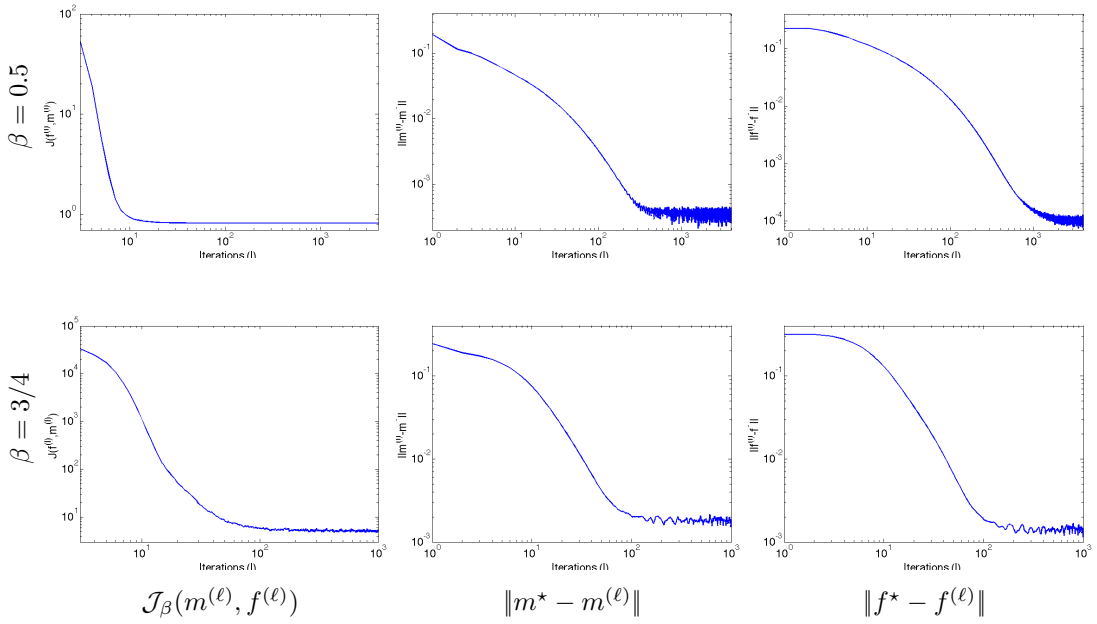

FIG. 6.5. *At each iteration ℓ, we plot the value of the cost function $\mathcal{J}_\beta(m^{(\ell)}, f^{(\ell)})$ and the distance between the reference solution $(m^\star, f^\star)$ and the estimation $(m^{(\ell)}, f^{(\ell)})$. The first (resp. second) row presents the result with $\beta = 1/2$ (resp. $\beta = 3/4$).*

As a second example, we show in Figure 6.6 the different morphings obtained between pictures of Gaspard Monge and Leonid Kantorovich. The grayscale representation scales linearly between black (value of 0) and white (value of 1) and the dimensions are $N + 1 = 75$, $M + 1 = 100$, $P + 1 = 60$, M being the number of discrete points in the second spatial dimension.

FIG. 6.6. *Evolution of $f^\star(\cdot, t)$ for several value of t and β. The first and last columns represent the data f^0 and f^1. The intermediate ones present the reference solution $f^\star(t)$ for successive times $t = i/6$, $i = 1 \cdots 5$. Each line illustrates $f^\star$ for different values $\beta = j/4$, $j = 0 \cdots 4$ of the generalized cost function.*

6.3. Riemannian Transportation. We investigate in this section the approximation of a displacement interpolation for a ground cost being the squared geodesic distance on a Riemannian manifold. This is achieved by solving (5.1) with $\beta = 1$ but a non-constant weight map w.

We exemplify this setting by considering optimal transport with obstacles, which corresponds to choosing weights w that are infinity on the obstacle $\mathcal{O} \subset \mathbb{R}^d \times \mathbb{R}$, i.e.

$$\forall\, k \in \mathcal{G}_{\mathrm{c}}, \quad w_k = 1 + \iota_{\mathcal{O}}(x_k, t_k) \in \{1, +\infty\}.$$

Note that the obstacles can be dynamic, i.e. the weight w needs not to be constant in time.

Figure 6.7 shows a first example where $\mathcal{O}$ is a 2-D ($d = 2$) static labyrinth map (the walls of the labyrinth being the obstacles and are displayed in black). We use a $50 \times 50 \times 100$ discretization grid of the space-time domain $[0, 1]^3$ and the input measures (f^0, f^1) are Gaussians with standard deviations equal to 0.04. For Gaussians with such a small variance, this example shows that the displacement interpolation is located closely to the geodesic path between the centers of the gaussians.

Figure 6.8 shows a more complicated setting that includes a labyrinth with moving walls: a wall appears at time $t = 1/4$ and another one disappears at time $1/2$. The difference with respect to the previous example is the fact that w is now time dependent. This simple modification has a strong impact on the displacement interpolation. Indeed, the speed of propagation of the mean of the density is not constant anymore since the density measure is confined in a small area surrounded

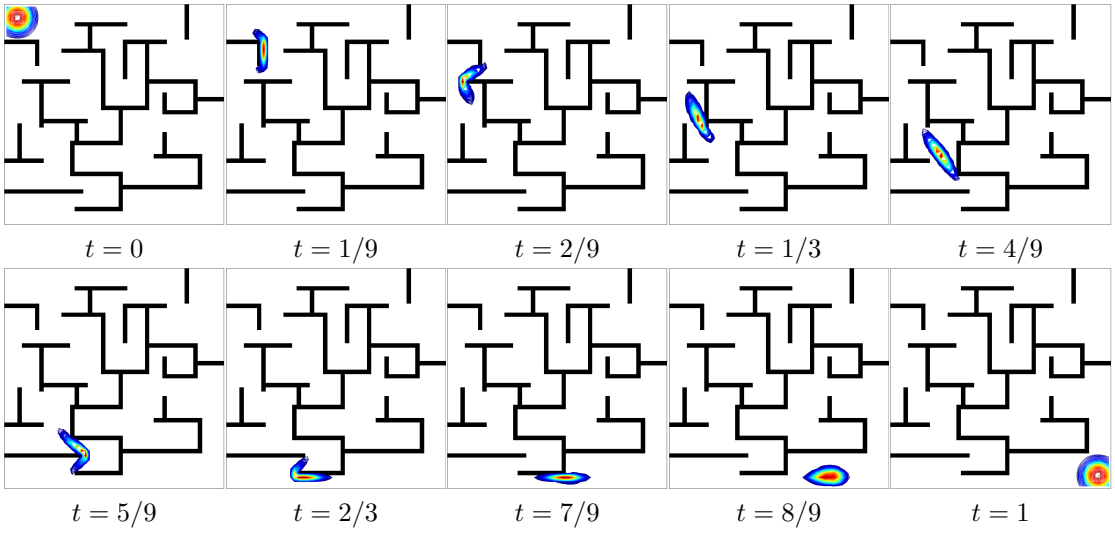

FIG. 6.7. *Evolution of $f^\star(\cdot, t)$ for several values of t, using a Riemannian manifold with weights w_k (constant in time) restricting the densities to lie within a 2-D static labyrinth map.*

by walls for $t \in [1/4, 1/2]$.

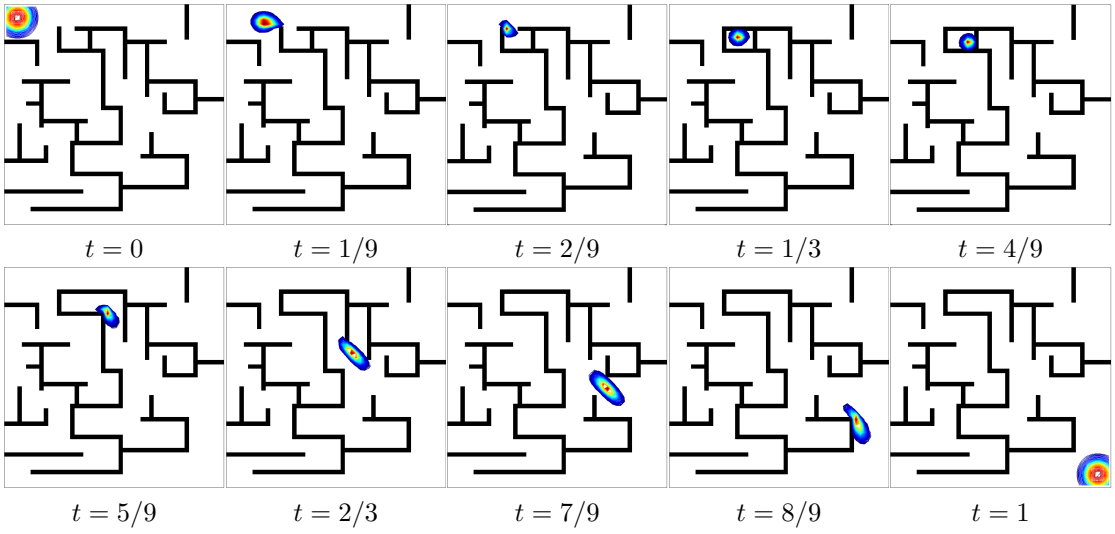

FIG. 6.8. *Evolution of $f^\star(\cdot, t)$ for several values of t, using a Riemannian manifold with weights w_k (evolving in time) restricting the densities to lie within a 2-D dynamic labyrinth map (i.e. with moving walls).*

As a last example, we present in Figure 6.9 an interpolation result in the context of oceanography in the presence of coast. We here consider Gaussian mixture data in order to simulate the Sea Surface Temperature that can be observed from satellite. In order to model the influence of the sea ground height, we here considered weights w varying w.r.t the distance to the coast. Denoting as $\mathcal{O}$ the area representing the complementary of the sea, we define

$$\forall\, k \in \mathcal{G}_{\mathrm{c}}, \quad w_k = 1 + d(x_k, \partial\mathcal{O}) + \iota_{\mathcal{O}} \in \{1, +\infty\},$$

where $d(x, \partial\mathcal{O})$ is the Euclidean distance between a pixel location x and the boundary of $\mathcal{O}$. The estimation of such interpolations are of main interest in geophysic forecasting applications where the variables of numerical models are calibrated using external image observations (such as the Sea Surface Temperature). Data assimilation methods used in geophysics look for the best compromise between a model and the observations (see for instance [12]) and making use of optimal transportation methods in this context is an open research problem.

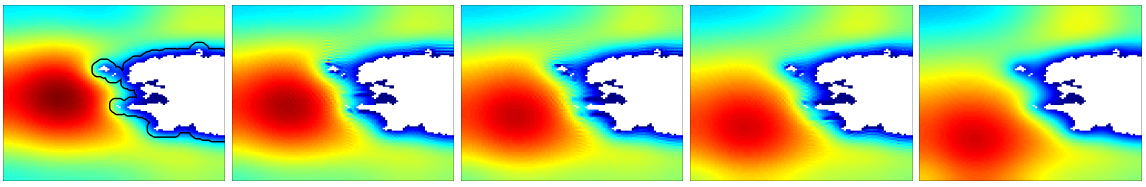

FIG. 6.9. *Evolution of $f^\star(\cdot, t)$ for several values of t, using a Riemannian manifold with weights w_k defined from the distance to the boundary of the sea domain (its frontier is displayed in black in the first figure).*

Conclusion. In this article, we have shown how proximal splitting schemes offer an elegant and unifying framework to describe computational methods to solve the dynamical optimal transport with an Eulerian discretization. This allowed use to extend the original method of Benamou and Brenier in several directions, most notably the use of staggered grid discretization and the introduction of generalized, spatially variant, cost functions.

Acknowledgment. We would like to thank to Jalal Fadili for his detailed explanations of the connexion between the ADMM and DR algorithms. Nicolas Papadakis would like to acknowledge partial support from the LEFE program of INSU (CNRS). Gabriel Peyré acknowledges support from the European Research Council (ERC project SIGMA-Vision). This work was also partially funded by the French Agence Nationale de la Recherche (ANR, Project TOMMI) under reference ANR-11-BS01-014-01.

REFERENCES

[1] M. Agueh and G. Carlier. Barycenters in the Wasserstein space. SIAM Journal on Mathematical Analysis, 43(2):904–924, 2011.
[2] J. Anderson. Computational Fluid Dynamics. McGraw-Hill Science/Engineering/ Math, 1995.
[3] S. Angenent, S. Haker, and A. Tannenbaum. Minimizing flows for the Monge-Kantorovich problem. SIAM Journal on Mathematical Analysis, 35(1):61–97, 2003.
[4] K.J. Arrow, L. Hurwicz, and H. Uzawa. Studies in linear and non-linear programming. Stanford University Press, 1958.
[5] J. Benamou and Y. Brenier. Weak existence for the semigeostrophic equations formulated as a coupled Monge–Ampere/transport problem. SIAM Journal on Applied Mathematics, 58(5):1450–1461, 1998.
[6] J.-D. Benamou. A domain decomposition method for the polar factorization of vector-valued mappings. SIAM Journal on Numerical Analysis, 32(6):1808–1838, 1995.
[7] J.-D. Benamou. Numerical resolution of an "unbalanced" mass transport problem. ESAIM: Mathematical Modelling and Numerical Analysis, 37(5):851–868, 2010.
[8] J.-D. Benamou and Y. Brenier. A computational fluid mechanics solution of the Monge-Kantorovich mass transfer problem. Numerische Mathematik, 84(3):375–393, 2000.
[9] J.-D. Benamou and Y. Brenier. Wasserstein optimal mapping between prescribed densities functions. Journal of Optimization Theory and Applications, 111(2):255–271, 2001.
[10] J.-D. Benamou, B. D. Froese, and Oberman A. M. Numerical solution of the optimal transportation problem via viscosity solutions for the Monge-Ampere equation. CoRR, abs/1208.4873, 2012.
[11] D.P. Bertsekas. The auction algorithm: A distributed relaxation method for the assignment problem. Annals of Operations Research, 14:105–123, 1988.
[12] J. Blum, F-X. Le Dimet, and I. M. Navon. Data assimilation for geophysical fluids. In Computational Methods for the Atmosphere and then Oceans, volume 14, pages 385–442. Elsevier, 2009.
[13] N. Bonneel, M. van de Panne, S. Paris, and W. Heidrich. Displacement interpolation using Lagrangian mass transport. ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia 2011), 30(6), 2011.
[14] F. Bornemann and C. Rasch. Finite-element discretization of static hamilton-jacobi equations based on a local variational principle. Comput. Visual Sci., 9(2):57–69, 2006.
[15] S. Boyd, B. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. Found. Trends Mach. Learn., 3(1):1–122, January 2011.
[16] Y. Brenier. Polar factorization and monotone rearrangement of vector-valued functions. Communications on Pure and Applied Mathematics, 44(4):375–417, 1991.
[17] L. M. Briceño-Arias and P. L. Combettes. A monotone+skew splitting model for composite monotone inclusions in duality. SIAM Journal on Optimization, 21(4):1230–1250, 2011.
[18] M. Burger, M. Franek, and C.-B. Schönlieb. Regularized regression and density estimation based on optimal transport. Applied Mathematics Research eXpress, 2011.
[19] R. Burkard, M. Dell'Amico, and S. Martello. Assignment Problems. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2009.
[20] P. Cardaliaguet, G. Carlier, and B. Nazaret. Geodesics for a class of distances in the space of probability measures. Calculus of Variations and Partial Differential Equations, pages 1–26, 2012.

[21] J. Carrillo and J. Moll. Numerical simulation of diffusive and aggregation phenomena in nonlinear continuity equations by evolving diffeomorphisms. SIAM Journal on Scientific Computing, 31(6):4305–4329, 2010.
[22] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. Journal of Mathematical Imaging and Vision, 40(1):120–145, 2011.
[23] P. Clarysse, B. Delhay, M. Picq, and J. Pousin. Optimal extended optical flow subject to a statistical constraint. Journal of Computational and Applied Mathematics, 234(4):1291–1302, 2010.
[24] P. L. Combettes and J.-C. Pesquet. A proximal decomposition method for solving convex variational inverse problems. Inverse Problems, 24(6):065014, 2011.
[25] P.L. Combettes and J.-C. Pesquet. A Douglas-Rachford splitting approach to nonsmooth convex variational signal recovery. IEEE Journal of Selected Topics in Signal Processing, 1(4):564 –574, 2007.
[26] P.L. Combettes and J-C. Pesquet. Proximal splitting methods in signal processing. In Fixed-Point Algorithms for Inverse Problems in Science and Engineering, pages 185–212. Springer, 2011.
[27] D. Cordero-Erausquin, B. Nazaret, and C. Villani. A mass-transportation approach to sharp Sobolev and Gagliardo-Nirenberg inequalities. Advances in Mathematics, 182(2):307–332, 2004.
[28] G. B. Dantzig. Linear Programming and Extensions. Princeton University Press, Princeton, NJ, 1963.
[29] F. de Goes, D. Cohen-Steiner, P. Alliez, and M. Desbrun. An optimal transport approach to robust reconstruction and simplification of 2D shapes. Computer Graphics Forum, 30(5):1593–1602, 2011.
[30] E. J. Dean and R. Glowinski. An augmented Lagrangian approach to the numerical solution of the dirichlet problem for the elliptic Monge-Ampère equation in two dimensions. Electron. Trans. Numer. Anal., 22:71–96, 2006.
[31] J. Delon. Movie and video scale-time equalization application to flicker reduction. IEEE Transactions on Image Processing, 15(1):241–248, 2006.
[32] J. Delon, J. Salomon, and A. Sobolevski. Fast transport optimization for Monge costs on the circle. SIAM Journal on Applied Mathematics, 70(7):2239–2258, 2010.
[33] J. Delon, J. Salomon, and A. Sobolevskii. Local matching indicators for transport problems with concave costs. SIAM Journal on Discrete Mathematics, 26(2):801–827, 2012.
[34] J. Digne, D. Cohen-Steiner, P. Alliez, F. Goes, and M. Desbrun. Feature-preserving surface reconstruction and simplification from defect-laden point sets. Journal of Mathematical Imaging and Vision, pages 1–14, 2013.
[35] J. Dolbeault, B. Nazaret, and G. Savaré. A new class of transport distances between measures. Calculus of Variations and Partial Differential Equations, 34(2):193–231, 2009.
[36] B. During, D. Matthes, and J.P. Milisic. A gradient flow scheme for nonlinear fourth order equations. Discrete and Continuous Dynamical Systems - Series B, 14(3):935–959, 2010.
[37] J. Eckstein and D. P. Bertsekas. On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators. Mathematical Programming, 55:293–318, 1992.
[38] X. Feng and M. Neilan. Mixed finite element methods for the fully non- linear Monge-Ampère equation based on the vanishing moment method. SIAM Journal on Numerical Analysis, 2(47):1226–1250, 2009.
[39] S. Ferradans, G-S. Xia, G. Peyré, and J-F. Aujol. Static and dynamic texture mixing using optimal transport. In Proc. SSVM'13, volume 7893 of Lecture Notes in Computer Science, pages 137–148. Springer Berlin Heidelberg, 2013.
[40] L. Ferragut and I. Asensio. Mixed finite element methods for a class of nonlinear reaction diffusion problems. Neural, Parallel Sci. Comput., 10(1):91–112, 2002.
[41] M. Fortin and R. Glowinski. Augmented Lagrangian Methods: Applications to the Numerical Solution of Boundary-Value Problems. Elsevier Science, 1983.
[42] U. Frisch, s. Matarrese, R. Mohayaee, and A. Sobolevski. A reconstruction of the initial conditions of the universe by optimal mass transportation. Nature, 417:260?–262, 2002.
[43] B. D. Froese. A numerical method for the elliptic Monge-Ampère equation with transport boundary conditions. SIAM Journal on Scientific Computing, 34(3):A1432–A1459, 2012.
[44] D. Gabay. Applications of the method of multipliers to variational inequalities. In M. Fortin and R. Glowinski, editors, Augmented Lagrangian Methods: Applications to the Solution of Boundary Value Problems, chapter IX, pages 299–340. North-Holland, Amsterdam, 1983.
[45] D. Gabay and B. Mercier. A dual algorithm for the solution of nonlinear variational problems via finite element approximation. Computers & Mathematics with Applications, 2(1):17–40, 1976.
[46] R. Glowinski and A. Marroco. Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problèmes de dirichlet non linéaires. ESAIM: Mathematical Modelling and Numerical Analysis - Modélisation Mathématique et Analyse Numérique, 9(R2):41–76, 1975.
[47] H. W. Kuhn. The Hungarian method of solving the assignment problem. Naval Res. Logistics Quart., 2:83–97, 1955.
[48] E. Haber, T. Rehman, and A. Tannenbaum. An efficient numerical method for the solution of the l_2 optimal mass transfer problem. SIAM Journal on Scientific Computing, 32(1):197–211, 2010.
[49] S. Haker, L. Zhu, A. Tannenbaum, and S. Angenent. Optimal mass transport for registration and warping. International Journal of Computer Vision, 60(3):225–240, December 2004.
[50] F. H. Harlow and J. E. Welch. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. Physics of Fluids, 8(12):2182–2189, 1965.
[51] A. Iollo and D. Lombardi. A Lagrangian scheme for the solution of the optimal mass transfer problem. Journal of Computational Physics, 230(9):3430–3442, 2011.
[52] R. Jordan, D. Kinderlehrer, and F. Otto. The variational formulation of the Fokker–Planck equation. SIAM Journal on Mathematical Analysis, 29(1):1–17, 1998.
[53] Y-H. Kim and B. Pass. Multi-marginal optimal transport on Riemannian manifolds. Technical report, Preprint arXiv:1303.6251, 2013.

[54] R. Kimmel and J.A. Sethian. Computing geodesic paths on manifolds. Proc. of the National Academy of Sciences, 95(15):8431–8435, 1998.
[55] H. Ling and K. Okada. An efficient earth mover's distance algorithm for robust histogram comparison. IEEE Transactions on Pattern Analysis and Machine Intelligence, 29(5):840–853, 2007.
[56] P. L. Lions and B. Mercier. Splitting algorithms for the sum of two nonlinear operators. Siam J. Numer. Anal., 16:964–979, 1979.
[57] R. J. McCann. A convexity principle for interacting gases. advances in mathematics, 128(1):153–179, 1997.
[58] R.J. McCann. Polar factorization of maps on Riemannian manifolds. Geometric & Functional Analysis, 11(3):589–608, 2001.
[59] Q. Mérigot. A multiscale approach to optimal transport. Computer Graphics Forum, 30(5):1583–1592, 2011.
[60] J.J. Moreau. Proximité et dualité dans un espace hilbertien. Bulletin de la Société Mathématique de France, 93:273–299, 1965.
[61] Y. E. Nesterov and A. S. Nemirovsky. Interior Point Polynomial Methods in Convex Programming : Theory and Algorithms. SIAM Publishing, 1993.
[62] A. M. Oberman. Wide stencil finite difference schemes for the elliptic Monge-Ampere equation and functions of the eigenvalues of the hessian. Discrete and Continuous Dynamical Systems - Series B, 1(10):221–238, 2008.
[63] V. I. Oliker and L. D. Prussner. On the numerical solution of the equation $\frac{\partial^2 z}{\partial x^2}\frac{\partial^2 z}{\partial y^2} - \left(\frac{\partial^2 z}{\partial x \partial y}\right)^2 = f$ and its discretizations. Numerische Mathematik, 3(54):271–293, 1988.
[64] N. Papadakis, E. Provenzi, and V. Caselles. A variational model for histogram transfer of color images. Image Processing, IEEE Transactions on, 20(6):1682–1695, 2011.
[65] O. Pele and M. Werman. Fast and robust earth mover's distances. In IEEE International Conference on Computer Vision, ICCV'09, pages 460–467, 2009.
[66] N. Pustelnik, C. Chaux, and J. Pesquet. Parallel proximal algorithm for image restoration using hybrid regularization. IEEE Transactions on Image Processing, 20(9):2450–2462, 2011.
[67] J. Rabin and G. Peyré. Wasserstein regularization of imaging problem. In IEEE International Conference on Image Processing (ICIP'11), pages 1541–1544, 2011.
[68] J. Rabin, G. Peyré, J. Delon, and M. Bernot. Wasserstein barycenter and its application to texture mixing. In Scale Space and Variational Methods in Computer Vision (SSVM'11), volume 6667, pages 435–446, 2011.
[69] Y. Rubner, C. Tomasi, and L.J. Guibas. A metric for distributions with applications to image databases. In IEEE International Conference on Computer Vision (ICCV'98), pages 59–66, 1998.
[70] J. A. Sethian. A fast marching level set method for monotonically advancing fronts. Proc. of the National Academy of Sciences, 93(4):1591–1595, 1996.
[71] S. Shirdhonkar and D.W. Jacobs. Approximate earth mover's distance in linear time. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR'08), pages 1–8, 2008.
[72] J.E. Spingarn. Applications of the method of partial inverses to convex programming: Decomposition. Mathematical Programming, 32(2):199–223, 1985.
[73] G. Strang. Linear Algebra and Its Applications. Brooks Cole, 1988.
[74] M. Sulman, J. F. Williams, and R. D. Russell. Optimal mass transport for higher dimensional adaptive grid generation. Journal of Computational Physics, 230(9):3302–3330, 2011.
[75] J. Tsitsiklis. Efficient algorithms for globally optimal trajectories. IEEE Trans. on Automatic Control, 40(9):1528–1538, Step. 1995.
[76] C. Villani. Topics in Optimal Transportation. Graduate Studies in Mathematics Series. American Mathematical Society, 2003.
[77] C. Villani. Optimal Transport: Old and New. Grundlehren der mathematischen Wissenschaften. Springer, November 2008.