# ENTROPIC APPROXIMATION OF WASSERSTEIN GRADIENT FLOWS

GABRIEL PEYRÉ[*]

**Abstract.** This article details a novel numerical scheme to approximate gradient flows for optimal transport (i.e. Wasserstein) metrics. These flows have proved useful to tackle theoretically and numerically non-linear diffusion equations that model for instance porous media or crowd evolutions. These gradient flows define a suitable notion of weak solutions for these evolutions and they can be approximated in a stable way using discrete flows. These discrete flows are implicit Euler time stepping according to the Wasserstein metric. A bottleneck of these approaches is the high computational load induced by the resolution of each step. Indeed, this corresponds to the resolution of a convex optimization problem involving a Wasserstein distance to the previous iterate. Following several recent works on the approximation of Wasserstein distances, we consider a discrete flow induced by an entropic regularization of the transportation coupling. This entropic regularization allows one to trade the initial Wasserstein fidelity term for a Kullback-Leibler divergence, which is easier to deal with numerically. We show how Kullback-Leibler first order proximal schemes, and in particular Dykstra's algorithm, can be used to compute each step of the regularized flow. The resulting algorithm is both fast, parallelizable and versatile, because it only requires multiplications by the Gibbs kernel $e^{-c/\gamma}$ where $c$ is the ground cost and $\gamma > 0$ the regularization strength. On Euclidean domains discretized on an uniform grid, this corresponds to a linear filtering (for instance a Gaussian filtering when $c$ is the squared Euclidean distance) which can be computed in nearly linear time. On more general domains, such as (possibly non-convex) shapes or on manifolds discretized by a triangular mesh, following a recently proposed numerical scheme for optimal transport, this Gibbs kernel multiplication is approximated by a short-time heat diffusion. We show numerical illustrations of this method to approximate crowd motions with congestion on complicated domains as well as extensions to take into account interactions between multiple densities.

**Key words.** Optimal transport, gradient flow, JKO flow, Wasserstein distance, Kullback-Leibler divergence, Dykstra's algorithm, crowd motion, non-linear diffusion.

**AMS subject classifications.** 90C25, 68U10

## 1. Introduction.

### 1.1. Optimal Transport.

*Optimal transport: from theory to applications.* In the last 20 years or so, optimal transport (OT) has emerged as a foundational tool to analyze diverse problems at the interface between variational analysis, partial differential equations and probability. We refer to the book of Villani [69] for an introduction to these topics. It took more time for this notion to become progressively mainstream in various applications, which is largely due to the high computation cost of the corresponding (static) linear program of Kantorovich [47] or to the dynamical formulation of Benamou and Brenier [11]. However, one can now find many relevant uses of OT in very diverse fields such as astrophysics [43], computer vision [59], computer graphics [17], image processing [73], statistics [14] and machine learning [33], to name a few.

*Entropic regularization.* In order to obtain fast approximations of optimal transport distances (a.k.a. Wasserstein distances), there has been a recent revival of the so-called entropic regularization method. Cuturi [33] presented this scheme in the machine learning community as a fully parallelizable algorithm which can make the method scalable to large problems. He shows that this corresponds to the application of the well-known iterative diagonal scaling algorithm, which is sometimes referred to

[*] CNRS and CEREMADE, Université Paris-Dauphine, Place du Maréchal De Lattre De Tassigny, 75775 PARIS CEDEX 16, FRANCE.

as Sinkhorn's algorithm [63, 65, 64] or IPFP [36]. This method is also closely related to Schrodinger's problem [61] of projecting a Gibbs distribution on fixed marginal constraints, see [60, 50] for recent mathematical accounts on this problem. It is also related to deviation problems, see [1] for a connexion with gradient flows.

The major interest of this entropic approximation is that it allows one to re-cast various OT-related problems as optimizations over the space of probabilities endowed with the Kulback-Leibler divergence. The geometry of this space, as well as the availability of efficient first-order optimization methods, makes this novel formulation numerically more friendly than the original linear program formulation. The price to pay for such simple and efficient approaches is the presence of an extra amount of smoothing (in fact a blurring by the Gibbs kernel) on the obtained results.

*Variational problems involving OT.* These methods have been used to solve various variational optimization problem involving the Wasserstein distance. For instance the computation of Wasserstein barycenters, initially proposed in [4], has been approximated by entropic regularization in [34]. A more general class of problems, including multi-marginal transport (see [58] for recent results on this topic) as well as generalized Euler's flows (see [21] for a weak formulation of Euler's equations), partial transport (as defined in [42]) and capacity-constrained transport (as defined in [45]) have been approximated by entropic smoothing in [12].

Our work goes in the same direction of applying entropic regularization to speed up the computation of OT-related problems. Instead of considering here the minimization of functionals involving the Wasserstein distance, we consider here the minimization of convex functions according to the Wasserstein distance.

### 1.2. Previous Works.

*Wasserstein flows – theory.* It is natural to derive various partial differential equations (PDE's) as gradient flows of certain energy functionals. While it is most of time assumed that the flow follows the gradient as defined through the $L^2$ topology on some Hilbert space of functions, it is sometime desirable to consider more complicated metrics. This allows one to capture different PDE's and also sometime to give a precise meaning to weak solutions of these PDE's. One of the most striking example is the computation of gradient flows over spaces of probability distributions (i.e. positive and normalized measures) according to the topology defined by the Wasserstein metric. In this setting, the gradient descent cannot be understood directly as an infinitesimal explicit descent in the direction of some gradient, but rather as a limit of an implicit Euler step, as detailed in Section 2.3. This idea corresponds to the notion of gradient flows in metric spaces exposed in the book [5].

The pioneer paper of Jordan, Kinderlehrer and Otto [46] shows how one recovers Fokker-Planck diffusions of distributions when one minimizes entropy functionals according to the $W_2$ Wasserstein metric. The corresponding method are often referred to as'JKO flows" in reference to these authors. Since then, many non-linear PDE's have been derived as gradient flows for Wasserstein metrics, including the heat equation on manifolds [40], the porous medium equation [56], more general degenerate parabolic PDE's [2], Keller-Segel equation [15] and higher order PDE's [44] (see also [24] for applications in imaging). It is also possible to define a suitable notion of minimizing flow that cannot be written as PDE's due to the non-differentiability of the energy functional, a striking example being the model of crowd motion with congestion proposed by [53],

*Wasserstein flows – numerics.* The use of Wasserstein methods to discretize non-linear evolutions is an emerging field of research. The major difficulty lies in the high computational cost induced by the resolution of each step.

The case of 1-D densities is simpler because the optimal transport metric is a flat metric when re-parameterized using inverse cumulative functions. This idea is used in [48, 15, 16, 3, 52]. In higher dimension, a first class of approaches uses an Eulerian representation of the discretized density (i.e. on a fixed grid). The resulting problem can be solved using interior point methods for convex energies [24] or some sort of linearization in conjunction with finite elements [23] or finite volumes [25] schemes. A second class of approaches rather uses a Lagrangian representation, which is well adapted to optimal transport where the thought after solution is obtained by warping the density at the previous iterate. This idea is at the heart of several schemes, using discretized warpings [26], particules systems [72], moving meshes [22] and a consistent discretization of the gradients of convex functions (i.e. optimal transports) [13]. Lastly, let us note that gradient flows over discrete spaces (e.g. graphs) have been recently proposed [29, 51, 54] and could lead to structure preserving discretization schemes.

In this article, we use an Eulerian discretization and intend at approximating flows for energies that are already convex in the usual (Euclidean) sense. The main goal is to provide a fast and quite versatile discretization scheme through the use of an entropic smoothing method.

*First order scheme with respect to Bregman divergences.* First order proximal optimization schemes have been recently popularized in image processing and machine learning, due to their simplicity and the low computational cost of each iteration. Each step typically requires the computation of proximal operators, which are defined as strictly convex optimization sub-problems, corresponding to an implicit step according to the $L^2$ distance. We refer to the book [8] for an overview of this large class of methods and recent developments. Note that these $L^2$ proximal methods have been used to solve the dynamical formulation of OT [11, 57].

Many of these proximal algorithms have been extended when one replaces the $L^2$ metric by more general Bregman divergences. The prototypical algorithm (although rarely applicable in its original form) that has been extended to this divergence setting is the so-called proximal point algorithm [39] (see [49] for an extension to more general, possibly non-smooth, divergences) which corresponds to iteratively applying the proximal operator of the function to be minimized.

Iterative projections on convex sets is probably the simplest yet useful example of proximal methods. It has been extended to the general setting of Bregman's divergences by Bregman [19]. This scheme actually computes the projection on the intersection of convex sets if these sets are affine, which is a restrictive assumption. The natural extension of iterative projections to generic closed convex sets is the so-called Dykstra's algorithm [38, 32], which can be interpreted as a block-coordinate optimization on the dual problem. Dykstra's method has been extended to the special case of half-spaces in [27] and to generic closed convex sets in [10, 20]. Actually, as we show in Section 3.2, this result extends to arbitrary proper lower-semicontinuous convex functions (that are not necessarily indicators of closed convex sets). Note that such an extension is well-known for the case of the $L^2$ metric [7].

While in this paper we only make use of Dykstra's algorithm, it should be noted that many more proximal splitting algorithms are available in this Bregman's divergences setting, such as Douglas-Rachford and ADMM [70], primal-dual algorithms [28] and hybrid proximal point algorithms [66]

**1.3. Contributions.** In this paper, we present a novel numerical scheme to compute approximations of discrete gradient flows for Wasserstein metrics. The approximation is performed by an entropic smoothing of the original OT distance. Each step is computed as the resolution of a convex but possibly non-smooth optimization problem involving a Kulback-Leibler divergence to some Gibbs kernel. We thus propose in Section 3 to solve it using an extension of Dykstra's algorithm to this class of problems, for which we prove the convergence to the solution. Our main finding is that this scheme is both simple to implement and competitive in term of computational speed, since it only requires multiplications with the Gibbs kernel, which, for many practical scenarios, can be achieved in nearly linear time. We illustrate in Secton 4 this point by applications to a crowd motion model involving a non-smooth congestion term. Lastly, Section 5 presents a generalization of the proposed algorithm to the case were several couplings are optimized. We show the usefulness of this generalization to compute the gradient flow of a Wasserstein attraction term with congestion and to compute evolution of several coupled densities.

The code to reproduce the numerical part of this article is available online.[1]

**1.4. Notations.** In the following we consider either vectors $p \in \mathbb{R}^N$ ($N$ being the number of discretization points) that are usually in the probability simplex

$$\Sigma_N \overset{\text{def.}}{=} \left\{ p \in \mathbb{R}_+^N \; ; \; \textstyle\sum_{i=1}^N p_i = 1 \right\} \tag{1.1}$$

and couplings, that are matrices $\pi \in \mathbb{R}_+^{N \times N}$. We denote $\langle p,\, q\rangle = \sum_{i=1}^N p_i q_i$ the canonical inner product on $\mathbb{R}^N$ and similarly on $\mathbb{R}^{N\times N}$.

For some set $\mathcal{C} \subset \mathbb{R}^Q$ (typically $Q = N$ or $Q = N \times N$), we define its indicator function as

$$\forall\, a \in \mathbb{R}^Q, \quad \iota_{\mathcal{C}}(a) \overset{\text{def.}}{=} \begin{cases} 0 & \text{if} \quad a \in \mathcal{C}, \\ +\infty & \text{otherwise.} \end{cases} \tag{1.2}$$

To ease notations, we define $\odot$ and $\frac{\cdot}{\cdot}$ as being entry-wise operations, i.e. $a \odot b \overset{\text{def.}}{=} (a_i b_i)_i$ and $\frac{a}{b} \overset{\text{def.}}{=} (a_i/b_i)_i$. We denote as $\mathbb{1} \overset{\text{def.}}{=} (1, \ldots, 1)^T \in \mathbb{R}^N$ the vector filled with ones. We define

$$\forall\, \ell \in \mathbb{N}, \quad [\ell]_2 \overset{\text{def.}}{=} \begin{cases} 1 & \text{if} \quad \ell \text{ is odd,} \\ 2 & \text{if} \quad \ell \text{ is even.} \end{cases} \tag{1.3}$$

We define minus the entropy on both vectors and couplings (and we make this distinction on purpose to ease the description of the proposed methods) as

$$\forall\, p \in \mathbb{R}^N, \quad \bar{E}(p) \overset{\text{def.}}{=} \sum_{i=1}^N p_i(\log(p_i) - 1) + \iota_{\mathbb{R}^+}(p_i), \tag{1.4}$$

$$\forall\, \pi \in \mathbb{R}^{N\times N}, \quad E(\pi) \overset{\text{def.}}{=} \sum_{i,j=1}^Q \pi_{i,j}(\log(\pi_{i,j}) - 1) + \iota_{\mathbb{R}^+}(\pi_{i,j}), \tag{1.5}$$

with the convention that $0 \log(0) = 0$, and where $\iota_{\mathbb{R}^+}$ is the indicator function of $\mathbb{R}^+$, see (1.2).

[1] https://github.com/gpeyre/2015-SIIMS-wasserstein-jko/

We define the Kulback-Leibler divergence on both vectors and couplings as

$$\forall\, (p,q) \in \mathbb{R}_+^N \times \mathbb{R}_{+,*}^N, \quad \overline{\mathrm{KL}}(p|q) \overset{\text{def.}}{=} \sum_{i=1}^{Q} p_i \log\left(\frac{p_i}{q_i}\right) - p_i + q_i, \tag{1.6}$$

$$\forall\, (\pi,\xi) \in \mathbb{R}_+^{N\times N} \times \mathbb{R}_{+,*}^{N\times N}, \quad \mathrm{KL}(\pi|\xi) \overset{\text{def.}}{=} \sum_{i,j=1}^{Q} \pi_{i,j} \log\left(\frac{\pi_{i,j}}{\xi_{i,j}}\right) - \pi_{i,j} + \xi_{i,j}. \tag{1.7}$$

## 2. Entropic Discrete JKO Flows.

We describe here discrete gradient flows with entropic smoothing and briefly outline connexions with continuous flows and PDE's.

### 2.1. Discretization.

In this article, we consider discrete flows (i.e. evolutions are discretized in time) of discrete probability distributions (i.e. the space is also discretized, and we deal with finite dimensional problems). More precisely, we consider a computational grid $\{x_i\}_{i=1}^N$ of $N$ points, which can be understood for instance as an uniform grid in a sub-set of $\mathbb{R}^d$ (in the numerical illustrations of Section 4 we consider $d=2$) or as vertices of a triangulation of a surface.

We consider discrete probability distributions on this set of points which we represent in the following as vectors $p \in \Sigma_N$ in the simplex as defined in (1.1), where $p_i$ is the mass associated to the point $x_i$. One should understand these discrete vectors as a discretization of some measure, which is assumed to have a density $\rho(x)$ with respect to the usual area measure (e.g. the Lebesgue measure in $\mathbb{R}^d$ or the area measure on a surface). A typical way to map $\rho(x)$ to $p \in \Sigma_N$ is to partition the computation domain in non-overlapping cells $V_i$, located around each $x_i$, and to set $p_i = \int_{V_i} \rho(x)\mathrm{d}x$. A canonical choice is to set $(V_i)_i$ to be the Voronoi partition of the domain associated to the points $(x_i)_i$.

Discretized optimal transport between such discrete measures is defined according to some ground cost $c \in \mathbb{R}^{N\times N}$. A typical scenario is when $x_i \in \mathbb{R}^d$ are points in the Euclidean space and one considers $c_{i,j} = \|x_i - x_j\|^\alpha$, corresponding to the definition of Wasserstein distances. The case $\alpha = 1$ corresponds to Monge's original problem, and $\alpha = 2$ to the so-called $W_2$ metric, which is by far the most studied case because of its geometrical properties [69]. A natural extension is to consider points $x_i$ on a smooth manifold $\mathcal{M}$ and to define $c_{i,j} = d_{\mathcal{M}}(x_i,x_j)^\alpha$ where $d_{\mathcal{M}}$ is the geodesic distance on the manifold.

### 2.2. Entropic Regularization of Wasserstein Distance.

Following several recent works (see Sections 1.2), the entropic-regularized transportation distance between $(p,q) \in \Sigma_N^2$ for a ground cost $c \in \mathbb{R}^{N\times N}$ is

$$W_\gamma(p,q) \overset{\text{def.}}{=} \min_{\pi \in \Pi(p,q)} \langle c,\, \pi\rangle + \gamma E(\pi)$$

for some regularization parameter $\gamma \geqslant 0$, where the set of couplings with prescribed marginals $(p,q)$ is

$$\Pi(p,q) \overset{\text{def.}}{=} \left\{\pi \in \mathbb{R}_+^{N\times N} \;;\; \pi\mathbb{1} = p, \pi^T\mathbb{1} = q\right\}.$$

The case $\gamma = 0$ corresponds to the classical, un-regularized, optimal transport, and is a linear program. The case $\gamma > 0$ corresponds to a strictly convex minimization problem, where $E$ plays the role of a barrier function of the positive octant making the optimization problem better conditioned numerically. But there is more than merely a strict-convexification of the original functional, otherwise one could have settled for more classical log-barrier routinely used in interior-point methods [55]. Algebraic properties of the entropy, and its close relationship with the Kulback-Leibler (KL) divergence (1.6) (see Section 3.2 for a precise statement) indeed enables closed-form solutions for the various marginal projections problems encountered in OT problems (see for instance Proposition 3.1).

It is important to remind that $W_\gamma$ is not a distance for $\gamma > 0$, and we refer to Section 5.3 for a discussion on the impact of this deficiency.

**2.3. JKO Stepping.** Following the initial work of [46] (which gives the name of the method, "JKO flows"), it is possible to discretize various non-linear PDE's as a gradient flow of a functional $f$ using implicit gradient step with respect to a Wasserstein distance. Our method relies on the idea of replacing the initial Wasserstein metric by its entropic regularized approximation.

A entropically regularized JKO iteration is an implicit descent descent step with respect to the $W_\gamma$ "metric". To be consistent with notations introduced in the remaining parts of this article, we thus refer to it as a proximal operator according to $W_\gamma$, and its definition reads, for $\tau > 0$,

$$\forall\, q \in \mathbb{R}^N, \quad \operatorname{Prox}_{\tau f}^{W_\gamma}(q) \overset{\text{def.}}{=} \underset{p \in \Sigma_N}{\operatorname{argmin}}\, W_\gamma(p,q) + \tau f(p). \tag{2.1}$$

Note that since $p \mapsto W_\gamma(p,q)$ is a strictly convex and coercive function, this operator is uniquely defined.

Starting from some fixed discrete density $p_{t=0} \in \Sigma_N$, one defines the discrete JKO follow as

$$\forall\, t > 0, \quad p_{t+1} \overset{\text{def.}}{=} \operatorname{Prox}_{\tau_t f}^{W_{\gamma_t}}(p_t), \tag{2.2}$$

where $\tau_t > 0$ is the step size, $\gamma_t > 0$ the entropic regularization parameter. Note that we allow here these parameters to vary during the iterations, although we use fixed parameters in the numerical sections 4 and 5.

When $c_{i,j} = d_{\mathcal{M}}(x_i, x_j)^2$ (the geodesic distance squared on some smooth manifold $\mathcal{M}$), $f$ is smooth and $\gamma = 0$ (no entropic regularization), a formal computation shows that this scheme discretizes, as $(\tau, 1/N) \to 0$, the PDE

$$\frac{\partial \rho_t}{\partial t} = \operatorname{div}_{\mathcal{M}}\left(\rho_t \nabla_{\mathcal{M}}(F'(\rho_t))\right).$$

Here $p_t \in \Sigma_N$ is a discretization of some density $\rho_t$ as detailed in Section 2.1 and $f(p)$ is a discretization of some functional $F(\rho)$ defined on densities. The operators $\operatorname{div}_{\mathcal{M}}$ and $\nabla_{\mathcal{M}}$ are the gradient and divergence operators on the manifold $\mathcal{M}$, and $F'$ is the differential of $F$ (the gradient for the $L^2$ metric on the probability distributions on $\mathcal{M}$). We refer to [40] for a proof of this relationship when $f$ is an entropy on a manifold.

For instance, in the case where $f(p) = \bar{E}(p) + \langle p,\, w\rangle$, this discrete flow thus discretizes a linear diffusion-advection on the manifold

$$\frac{\partial \rho_t}{\partial t} = \Delta_{\mathcal{M}}(\rho_t) + \operatorname{div}_{\mathcal{M}}(\rho_t z) \quad \text{where} \quad z = \nabla_{\mathcal{M}}(w).$$

so that the mass get advected by the vector field $z$.

**2.4. KL Proximal Operators.** In order for the method that we propose to be applicable, the function $f$ must be convex and should be "simple" in the sense that one should be able to compute easily its proximal operator for the KL divergence. Similarly to (2.1), this proximal operator is defined as

$$\forall\, q \in \mathbb{R}^N, \quad \mathrm{Prox}_{\tau f}^{\overline{\mathrm{KL}}}(q) \overset{\text{def.}}{=} \underset{p \in \mathbb{R}_+^N}{\mathrm{argmin}}\ \overline{\mathrm{KL}}(p|q) + \tau f(p). \tag{2.3}$$

Note that since $p \mapsto \overline{\mathrm{KL}}(p|q)$ is a strictly convex and coercive function, this operator is uniquely defined. Section (4) shows two examples of such "simple" functions: an indicator of a box constraint (for crowd movement) and generalized entropies (for non-linear diffusions).

The underlying rationale behind the framework we propose in this article is that, while it is in general impossible to directly compute the operator $\mathrm{Prox}_{\tau f}^{W_\gamma}$, there are many functionals for which $\mathrm{Prox}_{\tau f}^{\overline{\mathrm{KL}}}$ is accessible either in closed form, or through a fast and precise algorithm. We will thus trade the application of a single implicit $W_\gamma$ proximal step by the iterative application of several KL implicit proximal steps. Note that, in particular, $f$ does not need to be smooth, which is crucial to model non-PDE evolutions such as crowd movements with congestion [53].

The main property of the KL proximal operator are recalled in Appendix A.

## 3. A Bregman Proximal Splitting Approach.

In this section, we show how to re-formulate a single entropic regularized JKO step in order to introduce a KL divergence penalty. This is useful to allow for the application of generalized first order proximal methods.

**3.1. Reformulation as a KL Minimization.** We consider a single time step $t$, and denoting $q \overset{\text{def.}}{=} p_t$ the previous iterate of the flow, one can re-write the JKO stepping operator (2.1) as

$$\mathrm{Prox}_{\tau f}^{W_\gamma}(q) = \pi \mathbb{1}$$

where $\pi \in \mathbb{R}^{N \times N}$ solves the following strictly convex optimization problem

$$\min_{\pi}\ \langle c,\, \pi\rangle + \gamma E(\pi) + \tau f(\pi \mathbb{1}) + \iota_{\mathcal{C}_q}(\pi) \tag{3.1}$$

where we introduced the constraint set

$$\mathcal{C}_q \overset{\text{def.}}{=} \left\{\pi \in \mathbb{R}^{N\times N}\ ;\ \pi^T \mathbb{1} = q\right\}.$$

The initial formulation (3.1) can be re-cast as

$$\min_{\pi}\ \mathrm{KL}(\pi|\xi) + \varphi_1(\pi) + \varphi_2(\pi) \quad \text{where} \quad \begin{cases} \varphi_1(\pi) \overset{\text{def.}}{=} \iota_{\mathcal{C}_q}(\pi), \\ \varphi_2(\pi) \overset{\text{def.}}{=} \frac{\tau}{\gamma} f(\pi \mathbb{1}), \end{cases} \tag{3.2}$$

where we defined the Gibbs kernel $\xi$ as

$$\xi \overset{\text{def.}}{=} e^{-c/\gamma} \in \mathbb{R}_{+,*}^{N\times N}.$$

It should be noted that functions $(\varphi_1, \varphi_2)$ only depends on the marginals $\pi\mathbb{1}$ and $\pi^T\mathbb{1}$, which is a crucial point, and is extensively used to compute proximal operators (see in particular formula (A.6)).

### 3.2. Dykstra Algorithm with Bregman Divergences.

*Bregman divergence and proximal map.* In order to give a more general treatment of optimization problems of the form (3.2), that can be useful beyond the particular context of this article, we consider a generic Bregman divergence $\mathrm{D}_\Gamma$, defined on some convex set $\mathcal{D}$.

We follow [10] and define a Bregman divergence (see for instance) as

$$\forall\,(\pi,\xi)\in\mathcal{D}\times\mathrm{int}(\mathcal{D}),\quad \mathrm{D}_\Gamma(\pi|\xi)=\Gamma(\pi)-\Gamma(\xi)-\langle\nabla\Gamma(\xi),\,\pi-\xi\rangle.$$

where $\Gamma$ is a strictly convex function, smooth on $\mathrm{int}(\mathcal{D})$ where $\mathcal{D}=\mathrm{dom}(\Gamma)$ such that its Legendre transform

$$\Gamma^*(\rho)=\max_{\pi\in\mathcal{D}}\langle\pi,\,\rho\rangle-\Gamma(\pi),$$

is also smooth and strictly convex. In particular, one has that $\nabla\Gamma$ and $\nabla\Gamma^*$ are bijective maps between $\mathrm{int}(\mathcal{D})$ and $\mathrm{int}(\mathrm{dom}(\Gamma^*))$ such that $\nabla\Gamma^*=\nabla\Gamma^{-1}$.

For $\Gamma=\|\cdot\|^2$, one recovers the squared Euclidean norm $\mathrm{D}_\Gamma=\|\cdot\|^2$. One has $\mathrm{KL}=\mathrm{D}_\Gamma$ for $\Gamma(\pi)=E(\pi)$, which is cased we used to tackle (3.2). Note that in general, $\mathrm{D}_\Gamma$ is not symmetric and does not satisfy the triangular inequality, so that it is not a distance. We refer to [10] for a table detailing many examples of Bregman's divergences.

Let us write the general form of problem (3.2) as

$$\min_{\pi\in\mathcal{D}}\ \mathrm{D}_\Gamma(\pi|\xi)+\varphi_1(\pi)+\varphi_2(\pi)\tag{3.3}$$

where $\varphi_1,\varphi_2$ are two proper, lower-semicontinuous convex functions defined on $\mathcal{D}$. We also assume that the following qualification constraint holds

$$\mathrm{ri}(\mathrm{dom}(\varphi_1))\cap\mathrm{ri}(\mathrm{dom}(\varphi_2))\cap\mathrm{ri}(\mathrm{dom}(\Gamma))\neq\emptyset.\tag{3.4}$$

where ri is the relative interior and $\mathrm{dom}(\varphi)=\{\pi\;;\;\varphi(\pi)\neq+\infty\}$.

We define the proximal map of a convex function $\varphi$ according to this divergence as

$$\mathrm{Prox}_\varphi^{\mathrm{D}_\Gamma}(\pi)\overset{\text{def.}}{=}\underset{\tilde\pi\in\mathcal{D}}{\mathrm{argmin}}\ \mathrm{D}_\Gamma(\tilde\pi|\pi)+\varphi(\tilde\pi).\tag{3.5}$$

We assume that $\varphi$ is coercive, so that $\mathrm{Prox}_\varphi^{\mathrm{D}_\Gamma}(\pi)$ is always uniquely defined by strict convexity. Furthermore, one has $\mathrm{Prox}_\varphi^{\mathrm{D}_\Gamma}(\pi)\in\mathrm{int}(\mathcal{D})$, see [10].

*Dykstra's iterations.* Dykstra's algorithm starts by initializing

$$\pi^{(0)}\overset{\text{def.}}{=}\xi\quad\text{and}\quad v^{(0)}=v^{(-1)}\overset{\text{def.}}{=}0.$$

One then iteratively defines, for $\ell>0$

$$\pi^{(\ell)}\overset{\text{def.}}{=}\mathrm{Prox}_{\varphi_{[\ell]_2}}^{\mathrm{D}_\Gamma}(\nabla\Gamma^*(\nabla\Gamma(\pi^{(\ell-1)})+v^{(\ell-2)})),\tag{3.6}$$

$$v^{(\ell)}\overset{\text{def.}}{=}v^{(\ell-2)}+\nabla\Gamma(\pi^{(\ell-1)})-\nabla\Gamma(\pi^{(\ell)}),\tag{3.7}$$

where $[\ell]_2$ is defined in (1.3). Note that the iterates satisfies $\pi^{(\ell)}\in\mathrm{int}(\mathcal{D})$, so that the algorithm is well defined.

The iterates $\pi^{(\ell)}$ of this algorithm are known to converge to the solution of (3.3) in the case where $\varphi_1$ and $\varphi_2$ are indicators of convex sets, see [10]. This corresponds to the case where $\mathrm{Prox}_{\varphi_i}^{\mathrm{D}_\Gamma}$ for $i=1,2$ are projectors according to the Bregman divergence.

*Convergence proof.* This convergence result in fact caries over to the more general setting where $(\varphi_1, \varphi_2)$ are arbitrary proper and lower-semicontinuous convex functions. The proof follows from the fact that Dykstra's iterations correspond to an alternate block minimization algorithm on the dual problem. This idea was suggested to us by Antonin Chambolle and Jalal Fadili.

Proposition 1. *If condition* (3.4) *holds, then* $\pi^{(\ell)}$ *converges to the solution of* (3.3).

*Proof.* The dual problem to (3.3) reads

$$\max_{u_1,u_2} \; -\varphi_1^*(u_1) - \varphi_2^*(u_2) - \Gamma^*(\alpha - u_1 - u_2) - C(\xi) \tag{3.8}$$

where the constant is $C(\xi) \overset{\text{def.}}{=} \langle \alpha,\, \xi \rangle - \Gamma(\xi)$ and where we defined $\alpha \overset{\text{def.}}{=} \nabla\Gamma(\xi)$.

Duality means that under the domain qualification hypothesis (3.4), the minimum value of (3.3) and the maximum value of (3.8) are the same, and that the primal solution $\pi$ can be recovered from the dual one $(u_1, u_2)$ as

$$\pi = \nabla\Gamma^*(-u_1 - u_2). \tag{3.9}$$

Starting from $(u_1^{(0)}, u_2^{(0)}) = (0, 0)$, the alternate block optimization on (3.8) defines a sequence $(u_1^{(\ell)}, u_2^{(\ell)})$, where, denoting $i = [\ell]_2$ (as defined in (1.3)) and $j = 3 - i \in \{1, 2\}$, the update at iteration $\ell$ reads

$$u_j^{(\ell)} \overset{\text{def.}}{=} u_j^{(\ell-1)} \quad \text{and} \quad u_i^{(\ell)} \overset{\text{def.}}{=} \underset{u_i}{\operatorname{argmax}} \; -\varphi_i^*(u_i) - \Gamma^*(q - u_i) \tag{3.10}$$

where we defined $q \overset{\text{def.}}{=} \alpha - u_j^{(\ell-1)}$.

Since in (3.8) the coupling term $\Gamma^*(\alpha - u_1 - u_2)$ between $(u_1, u_2)$ is smooth, a classical result ensures that $(u_1^{(\ell)}, u_2^{(\ell)})$ converges to the solution $(u_1^\star, u_2^\star)$ of (3.8), see for instance [30].

The primal problem associated to the dual maximization (3.10) is

$$\min_{\pi_i} \Gamma(\pi_i) - \langle q,\, \pi_i \rangle + \varphi_i(\pi_i) = \mathrm{D}_\Gamma(\pi_i | \nabla\Gamma^*(q)) + \varphi_i(\pi_i) + C \tag{3.11}$$

where $C \in \mathbb{R}$ is a constant. The primal-dual relationship between the solutions of (3.10) and (3.11) reads

$$\pi_i = \nabla\Gamma^*(q - u_i). \tag{3.12}$$

Equations (3.10) and (3.12) show that one has

$$u_i^{(\ell)} = \alpha - u_j^{(\ell-1)} - \nabla\Gamma \circ \operatorname{Prox}_{\varphi_i}^{\mathrm{D}_\Gamma}\left(\nabla\Gamma^*(\alpha - u_j^{(\ell-1)})\right). \tag{3.13}$$

We now perform the following change of variables $(u_1^{(\ell)}, u_2^{(\ell)}) \to (\pi^{(\ell)}, v^{(\ell)})$

$$\pi^{(\ell)} = \left\{ \begin{array}{ll} \nabla\Gamma^*(\alpha - u_1^{(\ell)} - u_2^{(\ell-1)}) & \text{if} \quad [\ell]_2 = 1, \\ \nabla\Gamma^*(\alpha - u_1^{(\ell-1)} - u_2^{(\ell)}) & \text{if} \quad [\ell]_2 = 2, \end{array} \right. \quad \text{and} \quad v^{(\ell)} = -u_{[\ell]_2}^{(\ell)}.$$

One then verifies that these variables satisfy the relationship (3.7) and that (3.13) is equivalent to (3.6). This shows by recursion that $(\pi^{(\ell)}, v^{(\ell)})$ corresponds to Dykstra's variables. The convergence of $(u_1^{(\ell)}, u_2^{(\ell)})$ toward $(u_1^\star, u_2^\star)$ implies that $\pi^{(\ell)}$ converges to $\pi^\star \overset{\text{def.}}{=} \nabla\Gamma^*(\alpha - u_1^\star - u_2^\star)$ which is the solution of (3.3) thanks to the primal-dual relationship (3.9). □

**3.3. Dykstra's Algorithm for** KL **divergence.** We now consider the case where $\Gamma = E$, $\mathrm{D}_\Gamma = \mathrm{KL}$. To ease the notations, we make the change of variables $z^{(\ell)} \overset{\text{def.}}{=} \nabla\Gamma^*(v^{(\ell)})$. One has that $\nabla\Gamma = \log$ and $\nabla\Gamma^* = \exp$ and thus one has the iterates

$$\pi^{(0)} \overset{\text{def.}}{=} \xi \quad \text{and} \quad z^{(0)} = z^{(-1)} \overset{\text{def.}}{=} \mathbb{1} \tag{3.14}$$

$$\forall\, \ell > 0, \quad \pi^{(\ell)} \overset{\text{def.}}{=} \mathrm{Prox}^{\mathrm{KL}}_{\varphi_{[\ell]_2}}(\pi^{(\ell-1)} \odot z^{(\ell-2)}), \tag{3.15}$$

$$z^{(\ell)} \overset{\text{def.}}{=} z^{(\ell-2)} \odot \frac{\pi^{(\ell-1)}}{\pi^{(\ell)}}. \tag{3.16}$$

Recall here that $\odot$ and $\dot{\div}$ denotes entry-wise operations.

**3.4. KL Proximal Operator for JKO Stepping.** In order to be able to apply iterations (3.15) and (3.16), one needs to be able to compute the proximal operator for the KL divergence of $\varphi_1$ and $\varphi_2$ for the functionals defined in (3.2).

The following proposition shows that these proximal operators for the KL divergence can be indeed computed in closed form as long as one can compute in closed form the proximal operator of $f$ for the $\overline{\mathrm{KL}}$ divergence.

Proposition 3.1. *For any $\pi \in \mathbb{R}_+^{N\times N}$, for $(\varphi_1, \varphi_2)$ defined in (3.2), one has*

$$\mathrm{Prox}^{\mathrm{KL}}_{\varphi_1}(\pi) = \pi\,\mathrm{diag}\left(\frac{q}{\pi^T \mathbb{1}}\right) \quad \textit{and} \quad \mathrm{Prox}^{\mathrm{KL}}_{\varphi_2}(\pi) = \mathrm{diag}\left(\frac{\mathrm{Prox}^{\overline{\mathrm{KL}}}_{\frac{\tau}{\gamma}f}(\pi\mathbb{1})}{\pi\mathbb{1}}\right)\pi \tag{3.17}$$

*Proof.* The computation of $\mathrm{Prox}^{\mathrm{KL}}_{\varphi_1}$ is obtained by combining (A.7) and (A.2) in the special case $M = 1$. The computation of $\mathrm{Prox}^{\mathrm{KL}}_{\varphi_2}$ is obtained by applying (A.6) in the special case of $M = 1$ coupling. □

**3.5. Dykstra Algorithm for JKO Stepping.** Writing down the first order optimality conditions with respect to $\pi$ for problem (3.1) shows that there exists $(a, b) \in (\mathbb{R}_+^N)^2$ such that the optimal $\pi$ satisfies $\pi = \mathrm{diag}(a)\xi\,\mathrm{diag}(b)$. It means that, just as for the classical entropic regularization of optimal transport [33], the optimal coupling $\pi$ is a diagonal scaling of the initial Gibbs kernel $\xi$. This remark actually not only holds for the optimal $\pi$, but it also holds for each iterate $\pi^{(\ell)}$ constructed by iterations (3.15) and (3.16) that defines $(\pi^{(\ell)}, z^{(\ell)}) \in (\mathbb{R}_+^N)^2$.

The following proposition makes use of this remark and shows how to actually implement numerically iterations (3.15) and (3.16) of the method in a fast and parallel way using only matrix-vector multiplications against the kernel $\xi$.

Proposition 3.2. *The iterates of Dykstra's algorithm can be written as*

$$\pi^{(\ell)} = \mathrm{diag}(a^{(\ell)})\xi\,\mathrm{diag}(b^{(\ell)}) \quad \textit{and} \quad z^{(\ell)} = u^{(\ell)}v^{(\ell),T} \tag{3.18}$$

*(i.e. $z^{(\ell)}$ is a rank-1 matrix) where $(a^{(\ell)}, b^{(\ell)}, u^{(\ell)}, v^{(\ell)}) \in (\mathbb{R}_{+,*}^N)^4$, with the initialization*

$$a^{(0)} = b^{(0)} = u^{(0)} = v^{(0)} = \mathbb{1}. \tag{3.19}$$

*For odd $\ell$, the update of $(a^{(\ell)}, b^{(\ell)})$ reads*

$$a^{(\ell)} = a^{(\ell-1)} \odot u^{(\ell-2)} \quad \textit{and} \quad b^{(\ell)} = \frac{q}{\xi^T(a^{(\ell)})}, \tag{3.20}$$

*while for even $\ell$ it reads*

$$b^{(\ell)} = b^{(\ell-1)} \odot v^{(\ell-2)} \quad \textit{and} \quad a^{(\ell)} = \frac{p^{(\ell)}}{\xi(b^{(\ell)})}, \tag{3.21}$$

*where we defined*

$$p^{(\ell)} \overset{\text{def.}}{=} \operatorname{Prox}^{\mathrm{KL}}_{\frac{\tau}{\gamma} f}(a^{(\ell-1)} \odot u^{(\ell-2)} \odot \xi(b^{(\ell)})). \tag{3.22}$$

*The update of $(u^{(\ell)}, v^{(\ell)})$ is always*

$$u^{(\ell)} = u^{(\ell-2)} \odot \frac{a^{(\ell-1)}}{a^{(\ell)}} \quad \textit{and} \quad v^{(\ell)} = v^{(\ell-2)} \odot \frac{b^{(\ell-1)}}{b^{(\ell)}}. \tag{3.23}$$

*Proof.* One verifies that the format (3.18) holds for the initialization (3.19) and that it is maintained by the update formulas (3.17). Formulas (3.20), (3.21) and (3.23) are obtained by identifying the different terms when plugging the format (3.18) into the update formulas (3.17). □

The Pseudo-code 1 recaps all the successive steps needed to compute the full JKO flow (2.2) with entropic smoothing. This resolution thus only requires to iteratively apply, until a suitable convergence criterion is met, the update rules (3.20), (3.21) and (3.23). In practice, we found that monitoring the violation of the constraint $\mathcal{C}_q$ to be both a simple and efficient way to enforce a stopping criterion This criterion allows furthermore to precisely enforce mass conservation and positivity, i.e. $p_t \in \Sigma_N$ stays normalized to unit mass, which is important in many practical cases.

The crux of the method, that is extensively used in the numerical section (see in particular Section 4.1) is that one only needs to know how to apply the kernel $\xi$ and its adjoint $\xi^T$ (which are in most practical situations equals), which can be achieved either exactly or approximately in fast and highly parallelizable manner.

```
1. Initialize $t = 0$ and $p_{t=0}$.
2. Initialize $\ell = 1$ and set
       $$a^{(0)} = b^{(0)} = u^{(0)} = v^{(0)} = \mathbb{1}.$$
3. Setting $q \overset{\text{def.}}{=} p_t$, update $(a^{(\ell)}, b^{(\ell)})$ using (3.20) is $\ell$ is odd, and using (3.21) if
   $\ell$ is even.
4. Update $(u^{(\ell)}, v^{(\ell)})$ using (3.23).
5. If $\|b^{(\ell)} \odot \xi^T(a^{(\ell)}) - q\| > \varepsilon$ or if $\ell$ is odd, set $\ell \leftarrow \ell + 1$ and go back to step 3.
6. Set $p_{t+1} = p^{(\ell)}$ as defined by (3.22), $t \leftarrow t + 1$ and go back to step 2.
```

**Pseudo-code 1:** Iterations computing the full JKO flow. The inputs are the initial density $p_{t=0}$, the parameters $(\gamma, \tau)$ and the tolerance $\varepsilon$. The outputs are the iterates $(p_t)_{t>0}$.

**4. Numerical Results.** We now illustrate the usefulness and versatility of our approach to compute approximate solutions to various non-linear diffusion processes. The videos showing the time evolutions displayed in the figures below are available online[2].

[2] https://github.com/gpeyre/2015-SIIMS-wasserstein-jko/tree/master/videos

**4.1. Exact and Approximate Kernel Computation.** As already highlighted in Section 3.5, our method is efficient if one can compute in a fast way the multiplication $\xi p$ between the Gibbs kernel $\xi = e^{-c/\gamma}$ and a vector $p \in \mathbb{R}^N$. In the general case, this is intractable because this is a full matrix-vector multiplication. Even if $\xi$ usually has an exponential decay away from the diagonal, precisely capturing this decay is important to achieve transportation of mass effects. In particular, truncating the kernel to obtain a sparse matrix is prohibited.

*Translation invariant metrics.* The simplest setting is when the sampling points $(x_i)_{i=1}^N$ (as defined in Section 2) correspond to an uniform grid discretization of a translation invariant distance, i.e. $c_{i,j} = D(x_i - x_j)^\alpha$ for some function $D : \mathbb{R}^d \to \mathbb{R}$. In this case, $\xi$ is simply a discrete convolution against the kernel $D(\cdot)^\alpha$ sampled on an uniform grid. For instance, when $D(\cdot) = \|\cdot\|$ and $\alpha = 2$, $\xi$ is simply a convolution with a Gaussian kernel of width $\gamma$. When using periodic or Neumann boundary conditions, it is thus possible to implement this convolution in $O(N\log(N))$ operations using Fast Fourier Transforms (FFT's). There also exists a flurry of linear time approximate convolutions, the most popular one being Deriche's factorization with IIR filters [37]. We used this method to generate the top rows of Figure 4.2 and 5.2. The other figures require a more advanced treatment because the kernel is not translation invariant. We now detail this approach.

*Riemannian metrics.* Unfortunately, many case of practical interest correspond to diffusions inside non-convex domains, or even on non-Euclidean domains, typically a Riemannian manifold $\mathcal{M}$ (possibly with a boundary). In this setting, the natural choice for the ground cost $c$ is to set $c_{i,j} = d_{\mathcal{M}}(x_i, x_j)^\alpha$, where $d_{\mathcal{M}}$ is the geodesic distance on the manifold. A major issue is that computing this matrix $c$ is costly, for instance it would take $O(N^2\log(N))$ using Fast-Marching technics [62] on a grid or a triangulated mesh of $N$ points. Even storing this non-sparse matrix can be prohibitive, and applying it at each step of the Dykstra algorithm would require $O(N^2)$ operations. Inspired by the "geodesic in heat" method of [31], it has recently been proposed by [67] to speed up these computations by approximating the kernel $\xi$ by a Laplace or a heat kernel $\tilde{\xi}$ (depending on wether $\alpha = 1$ or $\alpha = 2$). This means that $c$ does not need to be pre-computed and stored, and that, as explained below, one can apply it on the fly at each iteration using a fast sparse linear solver. In the numerical tests, we have used this approximation.

To this end, one only needs to have at its disposal a discrete approximation $\Delta_{\mathcal{M}}$ of the Laplacian operator on the manifold $\mathcal{M}$. This is easily achieved using axis-aligned finite differences for manifold discretized on a rectangular grid, and this is the case for Figures 4.2. When considering a discretized manifold $\mathcal{M}$ which is a triangulated surface (as this is the case for Figure 4.3), one can use piecewise linear $P_1$ finite elements, and the operator $\Delta_{\mathcal{M}}$ is then the popular Laplacian with cotangent weights, see [18].

Following [67], the kernel $\xi$ is then approximated using $L \in \mathbb{N}^*$ steps of implicit Euler time discretization of the resolution of the heat equation on $\mathcal{M}$ until time $\gamma$, i.e.

$$\xi \approx \tilde{\xi} \overset{\text{def.}}{=} \left(\mathrm{Id} - \frac{\gamma}{L}\Delta_{\mathcal{M}}\right)^{-L} \tag{4.1}$$

where $(\cdot)^{-L}$ means that one iterates $L$ times matrix inversion.

When $L = 1$, and ignoring discretization errors, the result of Varadhan [68] shows that in the limit of small $\gamma$, $\tilde{\xi}$ converges to the kernel $\xi$ obtained when using $\alpha = 1$ (i.e. "$W_1$" optimal transport). Indeed, this formula state that $-\frac{1}{\gamma}\log(\tilde{\xi})$ converge uniformly when $\gamma \to 0$ toward the geodesic distance on $\mathcal{M}$. As $L$ increases, $\tilde{\xi}$ converges to the heat kernel, which can be shown, also using [68] to be consistent with the case $\alpha = 2$ (i.e. "$W_2$" optimal transport). In the numerical tests, we have used $L = 10$.

Numerically, the computation of matrix/vector multiplications $\tilde{\xi}p$ appearing the Dykstra updates thus requires the resolution of $L$ linear systems. Since these matrix/vector multiplications are computed many times during the iterations, a considerable speed-up is achieved by first pre-computing a sparse Cholesky factorization of $\mathrm{Id} - \frac{\gamma}{L}\Delta_{\mathcal{M}}$ and then solving the $L$ linear systems by back-substitution [35]. Although there is no theoretical guarantee, we observed numerically that each step typically has a linear time complexity because the factorization is indeed highly sparse. We refer to [31] for an experimental analysis of this class of Laplacian solvers.

**4.2. Numerical Settings.** In the following sections, we show simulations on 2-D domains, either on a flat square (possibly equipped with an anisotropic Riemannian metric) or on a surface embedded in $\mathbb{R}^3$ (thus equipped with the induced Riemannian metric). The square domain is set to $[0,1]^2$ and we use an uniform grid of $N = 100 \times 100$ points. The surface used for the numerical tests is a triangulated mesh of $N = 20000$ vertices, fitting in the bounding box $[0,1]^3$.

We consider only cost functions corresponding to squared geodesic distances, i.e. we use $\alpha = 2$. When the cost function is $c_{i,j} = \|x_i - x_j\|^2$, we implemented multiplications by $\xi$ with a fast Gaussian filtering as explained in Section 4.1. For cost functions corresponding to geodesic distances $c_{i,j} = d_{\mathcal{M}}(x_i, x_j)^2$, we implemented the heat kernel approximation (4.1) with $L = 10$. We always consider Neumann boundary conditions, which correspond, for the case of translation invariant kernels, to convolutions with symmetric extensions across boundaries.

We set the entropic smoothing parameter to $\gamma = 1/N$, and the time step to $\tau = 1/\sqrt{N}$. The variable $t$ indicates the iteration number, as defined in (2.2).

**4.3. Crowd Motion Model.** To model crowd motion, we follow [53], where a congestion effect (not too many people can be at the same position) is enforced by imposing that the density $p$ of peoples follows a JKO flow with the functional $f$ defined as

$$\forall\, p \in \mathbb{R}^N, \quad f(p) \overset{\text{def.}}{=} \iota_{[0,\kappa]^N}(p) + \langle w,\, p \rangle \tag{4.2}$$

where $\kappa \geqslant \|p_{t=0}\|_\infty$ is the congestion parameter (the smaller, the more congestion) and $w \in \mathbb{R}^N$ is a potential (the mass should move along the gradient of $w$).

For such a function, the KL proximal operator is easy to compute, as detailed in the following proposition.

Proposition 2. *One has*

$$\forall\, p \in \mathbb{R}^N, \quad \mathrm{Prox}^{\mathrm{KL}}_{\sigma f}(p) = \min(p \odot e^{-\sigma w}, \kappa) \tag{4.3}$$

*where the min should be understood components-wise.*

*Proof.* By separability of the functional, one only needs to prove the 1-D case. One first proves the formula when $w = 0$. One then applies (A.3) in the case $M = 1$. □

Note that it is of course possible to consider a $\kappa$ that is spatially varying to model a non-homogeneous congestion effect.

Figure 4.1 shows the influence of the congestion parameter $\kappa$. This figure is obtained for the quadratic Euclidean cost $c_{i,j} = \|x_i - x_j\|^2$. Figure 4.2 shows various evolutions for different potentials $w$ (guiding the crowd to the exit) on a manifold $\mathcal{M}$

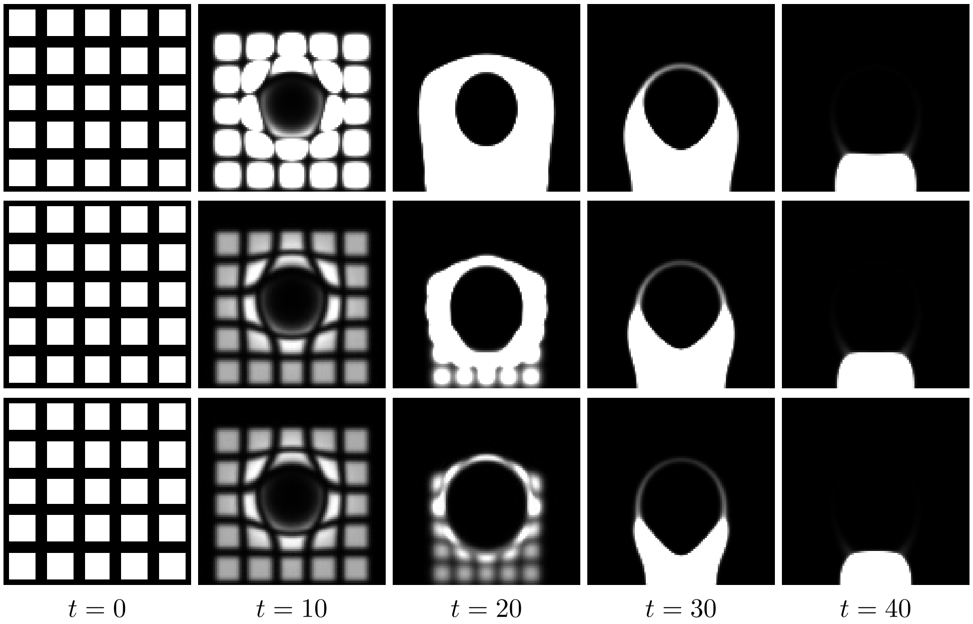


FIG. 4.1. *Display of the influence of the congestion parameter $\kappa$ on the evolution (the driving potential $w$ is displayed on the upper-right corner of Figure 4.2). From top to bottom, the parameters are set to $\kappa/\|p_{t=0}\|_\infty \in \{1, 2, 4\}\}$.*

which is a sub-set of a square in $\mathbb{R}^2$. This means that locally the Riemannian metric is Euclidean, but since the domain is non-convex, the transport is defined according to a geodesic distance $d_{\mathcal{M}}$ which is not the Euclidean distance. The discretization is achieved using the approximate heat kernel (4.1).

Lastly Figure 4.3 shows the evolution on a triangulated mesh, which is also implemented using the same heat kernel, but this time on a 3-D triangulation using piecewise linear finite elements (hence a discrete Laplacian with cotangent weights [18]).

**4.4. Anisotropic Diffusion Kernels.** We consider the crowd motion functional (4.2) over measures defined on $\mathcal{M} = \mathbb{R}^2$ now equipped with a Riemannian manifold structure defined by some tensor field $T(x) \in \mathbb{R}^{d\times d}$ of symmetric positive matrices. We use the heat kernel approximation detailed in Section 4.1. The kernel (4.1) thus corresponds to a discretization of an anisotropic diffusion, which are routinely used to perform image restoration [71]. As the anisotropy (i.e. the maximum ratio between the maximum and minium eigenvalues) of the tensors increases, the corresponding linear PDE becomes ill-posed, and traditional discretizations using finite differences are inconsistent, leading to unacceptable artifacts. We thus use the adaptive anisotropic stencils recently proposed in [41] to define the sparse Laplacian matrix discretizing the manifold Laplacian $\Delta_{\mathcal{M}} u(x) = \mathrm{div}(T(x)\nabla u(x))$. This discrete Laplacian is able to cope with highly anisotropic tensor fields. This is illustrated in Figure 4.4, which shows the impact of the anisotropy on the trajectory of the mass. The potential $w$ creates an horizontal movement of the mass, but the circular shape of the tensor orientations forces the mass to rather follow a curved trajectory. Ultimately, mass accumulates on the left side and the congestion effect appears.

**4.5. Non-linear Diffusions.**

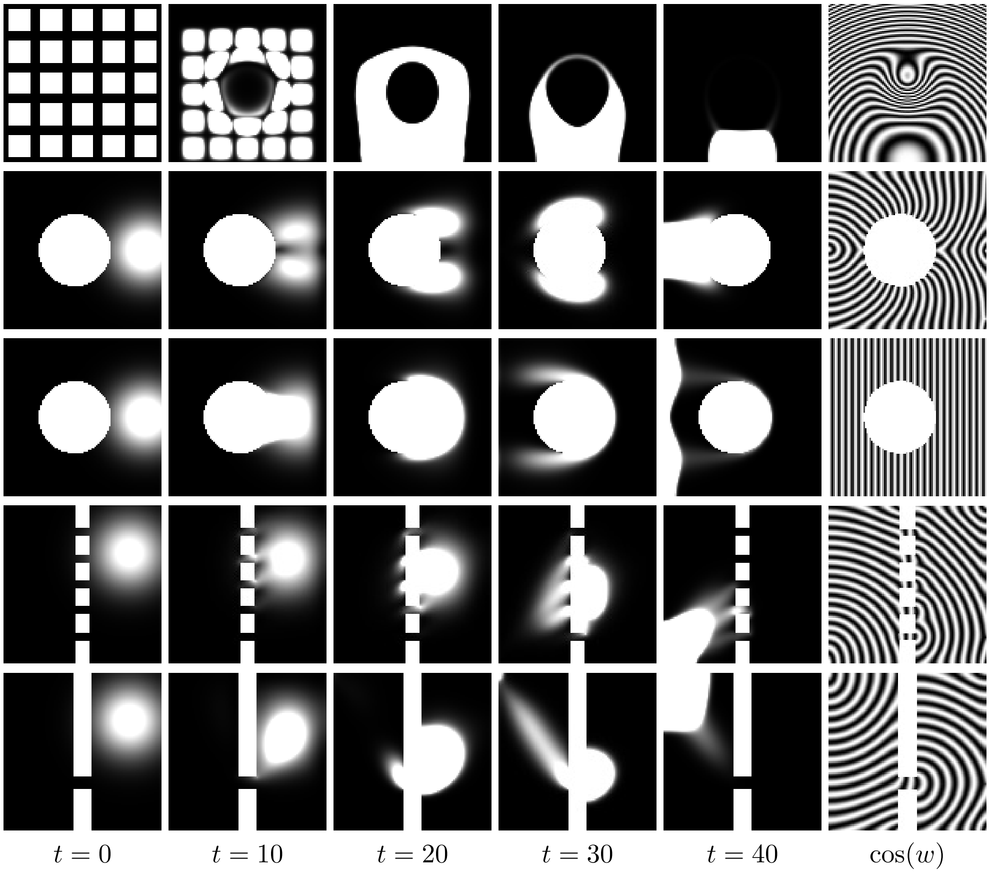


FIG. 4.2. *Display of crowd evolution for $\kappa = \|p_{t=0}\|_\infty$. The rightmost column display equispaced level-sets of the driving potential $w$.*

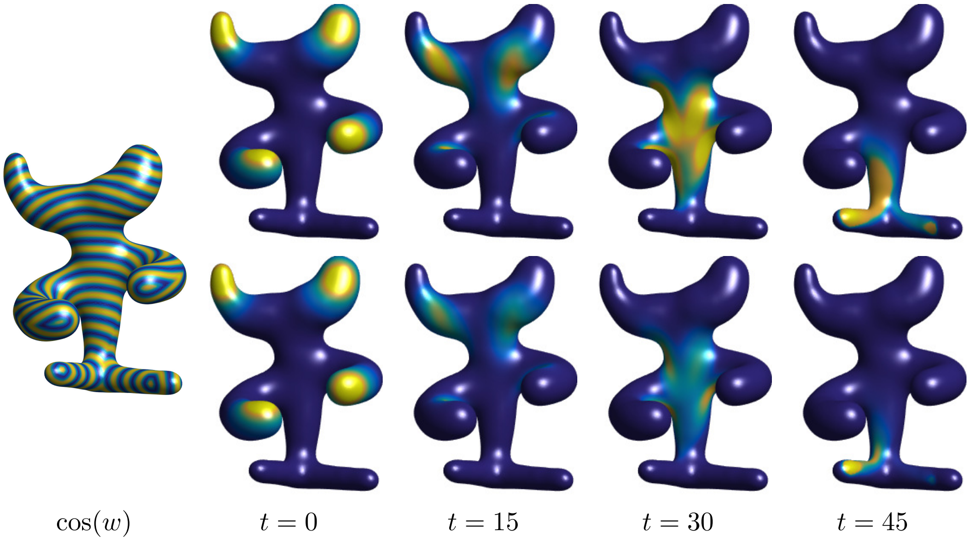


FIG. 4.3. *Display of the evolution $p_t$ on a triangulated surface. From top to bottom, the congestion parameter is set to $\kappa/\|p_{t=0}\|_\infty \in \{1, 6\}$.*

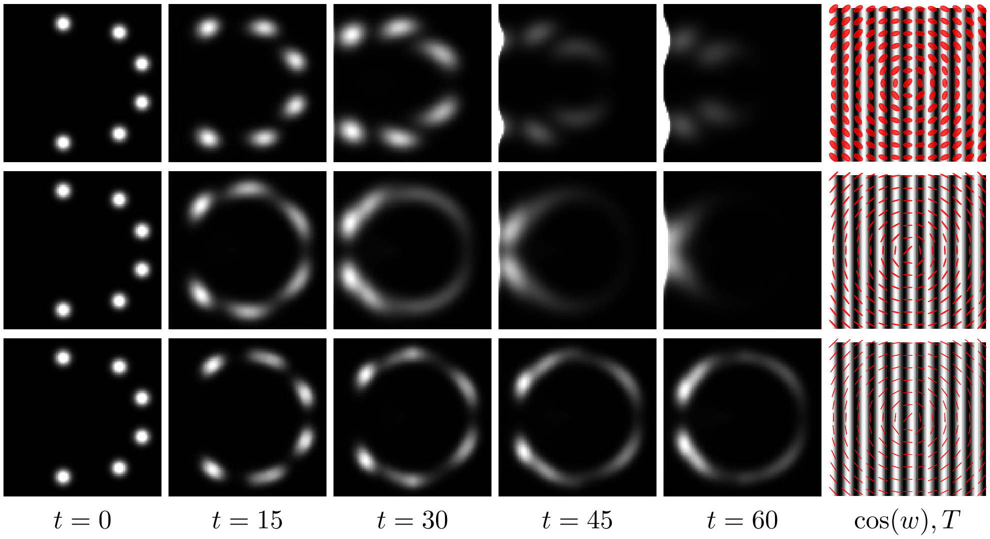


Fig. 4.4. *Display of the evolution $p_t$ using anisotropic diffusion kernels. The right column displays in the background the level-sets of $w$ and the tensor field $T(x)$ displayed as red ellipsoids. An ellipsoid at point $x$ is oriented along the principal axis of $T(x)$, and the length/width ratio is proportional to the anisotropy of $T(x)$. The metric thus favors mass movements along the direction of the ellipsoids. The anisotropy (ratio between maximum and minimum eigenvalues of $T(x)$) is set respectively from top to bottom to $\{2, 10, 30\}$ in each of the successive rows.*

**4.6. Non-linear Diffusions.** To model non-linear diffusion equations, we consider (possibly space-varying) generalized entropies

$$f(p) \overset{\text{def.}}{=} \sum_i b_i e_{m_i}(p_i) \quad \text{where} \quad \forall\, m \geqslant 1, e_m(s) \overset{\text{def.}}{=} \begin{cases} s(\log(s) - 1) & \text{if} \quad m = 1, \\ s\frac{s^{m-1} - m}{m-1} & \text{if} \quad m > 1. \end{cases} \tag{4.4}$$

Here $(b_i)_{i=1}^N$ is a set of weights $b_i \geqslant 0$ that enable a specially varying diffusion strength, while $(m_i)_{i=1}^N$ is a set of exponents that enable to make the evolution more non-linear at certain locations. Note that the case $m = 1$ corresponds to minus the entropy defined in (1.4).

In the case of constant weights $b$ and exponents $m$, the gradient flows of these functionals lead to non-linear diffusions of the form $\partial_t p = b \Delta p^m$. The case $m = 1$ is the usual linear heat diffusion, as considered in the initial work of [46]. The case $m = 2$ is the so-called porous medium equation [56], where diffusion is slower in areas where the density $p$ is small. In particular, solutions might have a compact support that evolves in time, on contrary to the linear heat diffusion where mass can travel with infinite speed.

The following proposition, details how to compute the proximal operator of $h$.

Proposition 3. *The proximal operator of $f$ satisfies*

$$\text{Prox}^{\overline{\text{KL}}}_{\sigma f}(r) = (\text{Prox}^{\overline{\text{KL}}}_{\sigma b_i e_{m_i}}(r_i))_{i=1}^N.$$

*For $m = 1$, the proximal operator of $e_1$ reads*

$$\forall\, s > 0, \quad \text{Prox}^{\overline{\text{KL}}}_{\sigma e_1}(s) = s^{\frac{1}{1+\sigma}}. \tag{4.5}$$

*If $m \neq 1$, then for any $s > 0$, $\text{Prox}^{\text{KL}}_{\sigma e_m}(s) = \psi$ is the unique positive root of the* *equation*

$$\log(\psi) + m\sigma\frac{\psi^{m-1}-1}{m-1} = \log(s) \tag{4.6}$$

*Proof.* The proof follows from writing the first order optimality condition of (3.5), which are exactly (4.6). For $m = 1$, this equation can be solved in closed form. □

In the numerical applications, we compute $\text{Prox}^{\text{KL}}_{\sigma e_m}$ by using a few steps of Newton iterations to solve (4.6), which can be parallelized over all the grid's locations. Figure 4.5 shows examples of the energy $e_m$ and the corresponding proximal maps $\text{Prox}^{\text{KL}}_{\sigma e_m}$. They act as pointwise non-linear thresholding operators that are applied iteratively on the probability distribution being computed. In some sense, the congestion term (4.2) and the corresponding proximal operator (4.3) can be understood as a limit of this model as $m \to +\infty$.

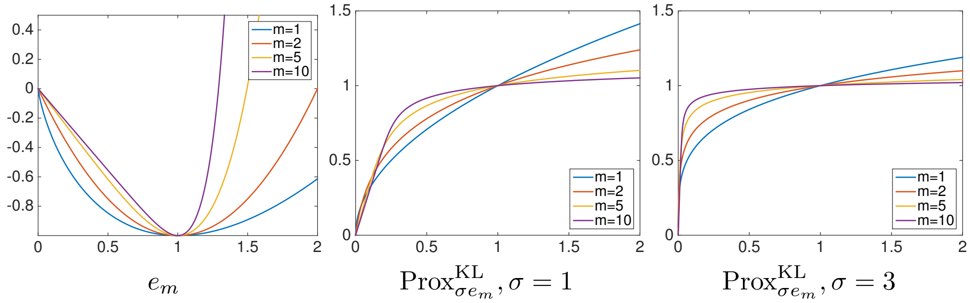


FIG. 4.5. *Display of the graphs of functions $e_m$ and $\text{Prox}^{\text{KL}}_{\sigma e_m}$ for some values of $(\sigma, m)$.*

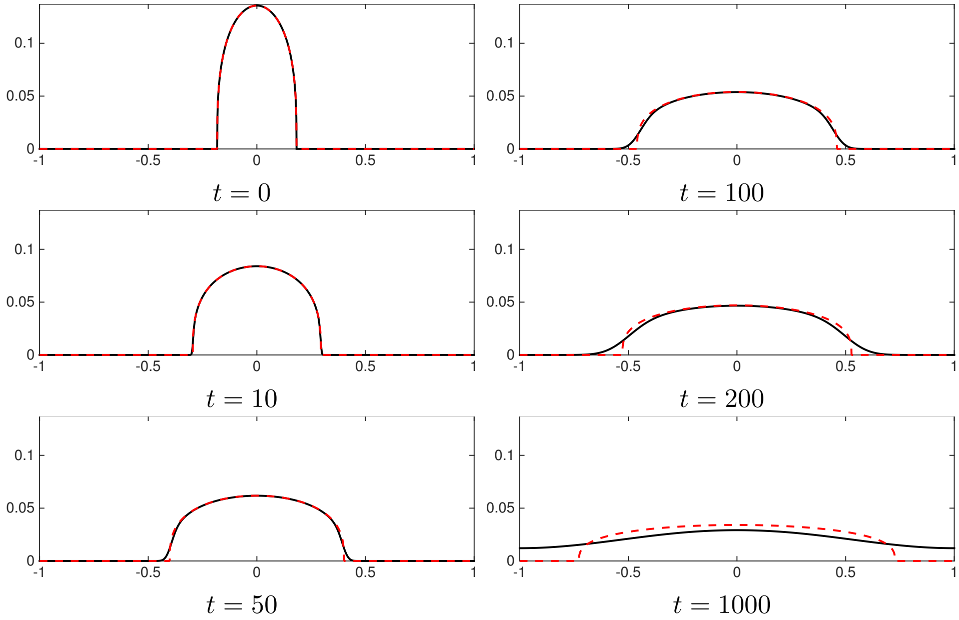
We first consider an homogeneous 1-D setting with $b_i = b$ and $m_i = m$. This corresponds to the discretization of the porous medium equation $\partial_t p = b\Delta p^m$. Following [72] we set $b = \frac{(m-1)^2}{4m}$. There exists a family of explicit solutions, the so-called Barenblatt profiles, see [72] for instance, given by the expressions, for $t > -t_0$,

$$(t+t_0)^{-\alpha}\left(C^2 - k(t+t_0)^{-2\alpha}x^2\right)_+^{\frac{1}{m-1}} \quad \text{where} \quad \begin{cases} \alpha = \frac{1}{m+1}, \\ k = \frac{m-1}{2m(m+1)}. \end{cases} \tag{4.7}$$

where $t_0$ is a time shift and $C > 0$ is a constant that controls the total mass of the solution.

Figure (4.6) shows a comparison between the ground trust solution (4.7) and the approximation computed by the entropic gradient flow for $m = 4$, $t_0 = 1$, $C = 1/20$, on a grid of $N = 2048$ points. The extra-smoothing added by the entropic scheme is clearly visible and it increases roughly linearly with time, so that the support of the solution is less and less compact. This is the main weakness of this numerical scheme, so that more conservative scheme such as [72] should be considered, at least for this homogeneous 1-D setting.

Figure (4.7) shows illustration of the models in the case where either $b$ or $m$ is varying, thus producing a spatially varying flow. The initial distribution $p_{t=0}$ is computed as a white noise realization, where the pixels are independently and identically drawn according to a uniform distribution on $[0, 1]$ (and then $p$ is normalized to unit mass).

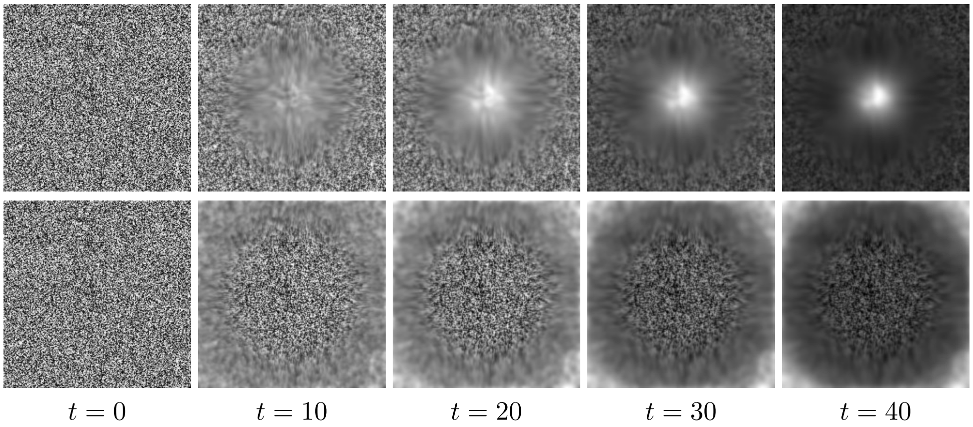


$t = 0$ $t = 100$

$t = 10$ $t = 200$

$t = 50$ $t = 1000$

FIG. 4.6. *Comparison of the Barenblatt profile (dashed curves) and the approximated solution (plain curves) for $m = 4$.*



$t = 0$ $t = 10$ $t = 20$ $t = 30$ $t = 40$

FIG. 4.7. *Non-linear diffusion by gradient flow on the generalized entropies* (4.4)*. Top row: fixed $m_i = 1.4$ and varying weights $b_i \in [1, 20]$ (1 in the boundary, 20 in the center). Bottom row: fixed $b_i = 1$ and varying exponents $m_i \in [1, 2]$ (1 in the boundary, 2 in the center).*

**4.7. Non-convex Functionals.** It is formally possible to apply Dykstra's algorithm detailed in Section 3.4 to a non-convex function $f$, if one is able to compute in closed form the proximal operator (3.5) (which then might be a multi-valued map). Of course there is no hope for the resulting non-convex Dykstra's algorithm to converge in general to the global minimizer of the non-convex optimization (3.2). Even worse, to the best of our knowledge, there is currently no proof that the non-convex Dykstra's algorithm converges to a stationary point of the energy, even in the case of an Euclidean divergence. However, we found that applying Dykstra's algorithm to non-convex functions works remarkably well in practice. Note that the closely related Douglas-Rachford (DR) algorithm is known to converge in some particular non-convex cases [6]. DR is known to perform very well on several non-convex optimization problems such as phase retrieval [9].

To test this non-convex setting, we replace the congestion box constraint (4.2) by the non-convex function

$$f(p) \overset{\text{def.}}{=} \iota_{\{0,\kappa\}^N} + \langle w,\, p\rangle. \tag{4.8}$$

This function enforces that the thought after solution is binary, so that each value $p_i$ is in $\{0,\kappa\}$. The proximal operator of this non-convex function can be computed explicitly using

$$\forall\, i \in \{1,\ldots,N\}, \quad \mathrm{Prox}^{\overline{\mathrm{KL}}}_{\sigma\iota_{\{0,\kappa\}^N}}(p)_i = \begin{cases} 0 \quad \text{if} \quad p_i < \kappa/e, \\ \{0,\kappa\} \quad \text{if} \quad p_i = \kappa/e, \\ \kappa \quad \text{if} \quad p_i > \kappa/e, \end{cases}$$

where $e = \exp(1)$. Note that $\mathrm{Prox}^{\overline{\mathrm{KL}}}_{\sigma\iota_{\{0,\kappa\}^N}}(p)_i$ is multi-valued at $p_i = \kappa/e$, and numerically one needs to chose one of the two values. Figure 4.8 shows a comparison of the evolutions obtained with the convex and non-convex functionals. The non-convex one suffers from binary noise artefacts, which could be partly due to the non-convexity, but also to the amplification of discretization errors by the proximal mapping which is not Lipschitz continuous.

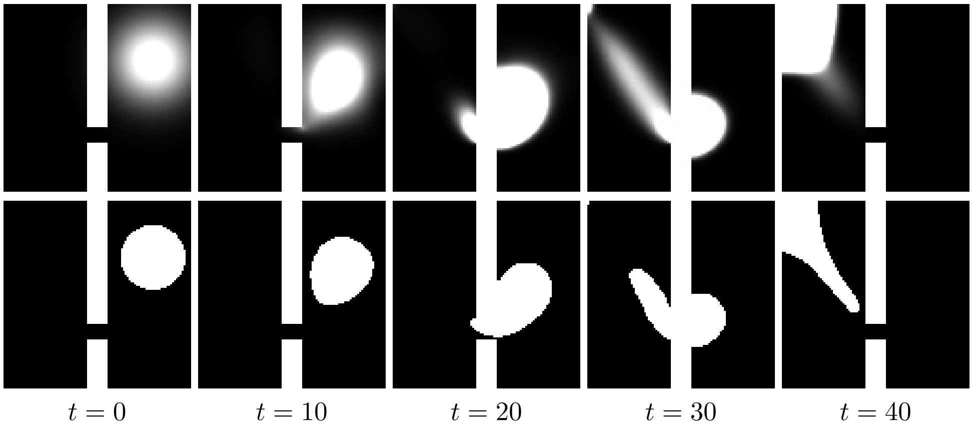


FIG. 4.8. *Display of crowd evolution for* $\kappa = \|p_{t=0}\|_\infty$. *Top row: convex function* (4.2). *Bottom row: non-convex function* (4.8).

## 5. More General Functionals.

In order to highlight the power of the proposed entropic regularization approach, we show here how to adapt the algorithm detailed in Section 3 in order to deal with more involved functionals. These functionals require the introduction of several couplings, which in turn necessitates to develop a generic iterative scaling procedure derived from Dykstra's algorithm. This new method has its own interest, beyond the computation of Wasserstein gradient flows.

### 5.1. A Generic Diagonal Scaling Algorithm.

In order to tackle a more general class of functions $f$, we consider here a generalization of problem (3.2) where one wants to optimize over a family $\pi = (\pi_1,\ldots,\pi_M)$ of $M$ couplings $\pi_m \in \mathbb{R}^{N\times N}$ a functional of the form

$$\min_{\pi\in(\mathbb{R}^{N\times N})^M} \mathrm{KL}_\lambda(\pi|\xi) + \varphi_1(\pi) + \varphi_2(\pi) \quad \text{where} \quad \begin{cases} \varphi_1(\pi) \overset{\text{def.}}{=} \psi_1(\pi\mathbb{1}), \\ \varphi_2(\pi) \overset{\text{def.}}{=} \psi_2(\pi^T\mathbb{1}), \end{cases} \tag{5.1}$$

where, for $\lambda \in \mathbb{R}_+^M$, $\mathrm{KL}_\lambda$ is the weighted KL divergence (see also (A.1))

$$\forall\, (\pi, \xi) \in (\mathbb{R}^{N\times N})^M \times (\mathbb{R}^{N\times N})^M, \quad \mathrm{KL}_\lambda(\pi|\xi) = \sum_{m=1}^{M} \lambda_m \,\mathrm{KL}(\pi_m|\xi_m)$$

and where we denoted, with a slight abuse of notations, the collection of left and right marginals as

$$\pi\mathbb{1} = (\pi_1\mathbb{1}, \ldots, \pi_M\mathbb{1}) \in (\mathbb{R}^N)^M \quad \text{and} \quad \pi^T\mathbb{1} = (\pi_1^T\mathbb{1}, \ldots, \pi_M^T\mathbb{1}) \in (\mathbb{R}^N)^M$$

and $\psi_i : (\mathbb{R}^N)^M \to \mathbb{R}$ are convex functions for which one can compute the proximal operator $\mathrm{Prox}_{\psi_i}^{\overline{\mathrm{KL}}_\lambda}$ according to the $\overline{\mathrm{KL}}_\lambda$ divergence.

We wish to apply Dykstra's iterations (3.15) and (3.16) to (5.1). This requires to compute the proximal operator of the functions $\varphi_i$. The following proposition details how to achieve this using the proximal operator of the functions $\psi_i$ alone.

Proposition 5.1. *We denote, for* $i = 1, 2$, $\pi^{[i]} \overset{\text{def.}}{=} \mathrm{Prox}_{\varphi_i}^{\mathrm{KL}_\lambda}(\pi)$. *We denote, for* $m = 1, \ldots, M$, $\tilde{p}_m^{[1]} \overset{\text{def.}}{=} \pi_m\mathbb{1}$ *and* $\tilde{p}_m^{[2]} \overset{\text{def.}}{=} \pi_m^T\mathbb{1}$. *One has*

$$\forall\, m \in \{1, \ldots, M\}, \quad \pi_m^{[1]} = \mathrm{diag}\left(\frac{p_m^{[1]}}{\tilde{p}_m^{[1]}}\right)\pi_m \quad \textit{and} \quad \pi_m^{[2]} = \pi_m \,\mathrm{diag}\left(\frac{p_m^{[2]}}{\tilde{p}_m^{[2]}}\right)$$

$$\textit{where} \quad \forall\, i \in \{1, 2\}, \quad (p^{[i]})_m = \mathrm{Prox}_{\psi_i}^{\overline{\mathrm{KL}}_\lambda}\left((\tilde{p}^{[i]})_m\right).$$

*Proof.* This corresponds to an application of formulas (A.6) and (A.7). □

The following proposition, which is similar to Proposition 3.2, explains how to implement the iterations of Dykstra's algorithm using only multiplications with the kernels $(\xi_m)_m$.

Proposition 5.2. *The iterates* $\pi^{(\ell)} = (\pi_1^{(\ell)}, \ldots, \pi_M^{(\ell)})$ *of Dykstra's algorithm can be written as*

$$\forall\, m \in \{1, \ldots, M\}, \quad \pi_m^{(\ell)} = \mathrm{diag}(a_m^{(\ell)})\xi_m \,\mathrm{diag}(b_m^{(\ell)}) \quad \textit{and} \quad z_m^{(\ell)} = u_m^{(\ell)} v_m^{(\ell),T}.$$

*with the initialization*

$$\forall\, m \in \{1, \ldots, M\}, \quad a_m^{(0)} = b_m^{(0)} = u_m^{(0)} = v_m^{(0)} \overset{\text{def.}}{=} \mathbb{1}.$$

*We define,* $\forall\, m \in \{1, \ldots, M\}$,

$$\tilde{a}_m^{(\ell-1)} \overset{\text{def.}}{=} a_m^{(\ell-1)} \odot u_m^{(\ell-2)} \quad \textit{and} \quad \tilde{b}_m^{(\ell-1)} \overset{\text{def.}}{=} b_m^{(\ell-1)} \odot v_m^{(\ell-2)},$$

$$(p_m)_m \overset{\text{def.}}{=} \mathrm{Prox}_{\psi_{[\ell]_2}}^{\overline{\mathrm{KL}}_\lambda}((\tilde{p}_m)_m) \quad \textit{where} \quad \tilde{p}_m \overset{\text{def.}}{=} \begin{cases} \tilde{a}_m^{(\ell-1)} \odot \xi_m(\tilde{b}_m^{(\ell-1)}) & \textit{if} \quad [\ell]_2 = 1, \\ \tilde{b}_m^{(\ell-1)} \odot \xi_m^T(\tilde{a}_m^{(\ell-1)}) & \textit{if} \quad [\ell]_2 = 2. \end{cases}$$

*The update reads,* $\forall\, m \in \{1, \ldots, M\}$,

$$a_m^{(\ell)} \overset{\text{def.}}{=} \begin{cases} p_m \odot [\xi_m(\tilde{b}_m^{(\ell-1)})]^{-1} & \textit{if} \quad [\ell]_2 = 1 \\ \tilde{a}_m^{(\ell-1)} & \textit{if} \quad [\ell]_2 = 2 \end{cases}$$

$$b_m^{(\ell)} \overset{\text{def.}}{=} \begin{cases} \tilde{b}_m^{(\ell-1)} & \textit{if} \quad [\ell]_2 = 1 \\ p_m \odot [\xi_m^T(\tilde{a}_m^{(\ell-1)})]^{-1} & \textit{if} \quad [\ell]_2 = 2 \end{cases}$$

$$u_m^{(\ell)} \overset{\text{def.}}{=} u_m^{(\ell-2)} \odot \frac{a_m^{(\ell-1)}}{a_m^{(\ell)}} \quad \textit{and} \quad v_m^{(\ell)} \overset{\text{def.}}{=} v_m^{(\ell-2)} \odot \frac{b_m^{(\ell-1)}}{b_m^{(\ell)}}.$$

**5.2. Wasserstein Attraction with Congestion.** We now give a first concrete example of functional $f$ for which the formulation (5.1) should be used in place of (3.2).

Instead of advecting the mass of $p_t$ according to a fixed potential $w$ as it is considered in the functional (4.2), it is possible to make it evolve toward a "target" distribution $r \in \Sigma_N$ by minimizing the Wasserstein distance between $p_t$ and $r$. It thus consists in considering the gradient flow of the function

$$\forall\, p \in \Sigma_N, \quad f(p) = W_\gamma(r,p) + h(p) + \iota_{\Sigma_N}(p), \tag{5.2}$$

where $h(p)$ is a function for which one can compute its $\overline{\mathrm{KL}}$ proximal operator as defined in (2.3).

We now denote $q \overset{\text{def.}}{=} p_t$ the previous iterate, and aim at solving a single JKO step (3.1). It is not possible to compute in closed form the $\overline{\mathrm{KL}}$ proximal operator of the function $f$ defined in (5.2), so that the algorithm detailed in Section 3 is not directly applicable.

Instead, we re-formulate (3.1) as a KL minimization of the form (5.1) involving $M = 2$ couplings $\pi = (\pi_1, \pi_2) \in (\mathbb{R}^{N\times N})^2$ and kernels $\xi = (\xi_1, \xi_2) \overset{\text{def.}}{=} (e^{-c/\gamma}, e^{-c/\gamma})$. This encodes implicitly the solution $p = \pi_1 \mathbb{1} = \pi_2 \mathbb{1}$ of (3.1) using the solution $\pi = (\pi_1, \pi_2)$ of (5.1) when introducing the functions

$$\begin{aligned}
&\psi_1(p_1,p_2) = \iota_{\mathcal{D}}(p_1,p_2) + h(p_1) \quad \text{where} \quad \mathcal{D} = \{(p_1,p_2)\,;\, p_1 = p_2\}\,,\\
&\psi_2(p_1,p_2) = \iota_{\{q,r\}}(p_1,p_2),
\end{aligned}$$

and the weights $\lambda = (1,\tau) \in \mathbb{R}_+^2$. The following proposition details how to compute the proximal operator of these functionals. It is important to remind that that these functionals as well as their respective proximal operators operate on vectors of $\mathbb{R}^N$, not on couplings.

Proposition 5.3. *One has*

$$\mathrm{Prox}_{\psi_1}^{\overline{\mathrm{KL}}_\lambda}(p_1,p_2) = (p,p) \quad \textit{where} \quad p = \mathrm{Prox}_{\frac{1}{1+\tau}h}^{\overline{\mathrm{KL}}}\left(p_1^{\frac{1}{1+\tau}} \odot p_2^{\frac{\tau}{1+\tau}}\right) \tag{5.3}$$

$$\mathrm{Prox}_{\psi_2}^{\overline{\mathrm{KL}}_\lambda}(p_1,p_2) = (q,r) \tag{5.4}$$

*Proof.* The computation of $\mathrm{Prox}_{\psi_1}^{\overline{\mathrm{KL}}_\lambda}$ follows from (A.4). The computation of $\mathrm{Prox}_{\psi_2}^{\overline{\mathrm{KL}}_\lambda}$ follows from (A.2). □

With these proximal maps at hands, and with the formula for the iterations detailed in Proposition 5.2, one can solve for each JKO step by computing the optimal $(\pi_1, \pi_2)$ using $q \overset{\text{def.}}{=} p_t$ and then updating $p_{t+1} \overset{\text{def.}}{=} \pi_1 \mathbb{1} = \pi_2 \mathbb{1}$.

*Numerical Illustrations.* In order to introduce some congestion, we consider here the function $h(p) = \iota_{[0,\kappa]^N}$, as in (4.2), and its KL proximal operator is computed as detailed in (4.3).

Figure 5.1 shows some examples of such a JKO flows computed on a rectangular grid. The right hand side column shows the target distribution $r$. Note that the flow $p_t$ typically does not converge toward $r$ as $t \to +\infty$, because of the congestion effect.

**5.3. Multiple Densities Evolutions.** A natural extension of the JKO flow (2.2) is to describe the evolution of a finite family of densities $p_t = (p_{i,t})_i$ by minimizing a function $f((p_i)_i)$, where one defines the transport distance as the sum of independent Wasserstein distances

$$W_\gamma((p_i)_i, (q_i)_i) \overset{\text{def.}}{=} \sum_i W_\gamma(p_i, q_i).$$

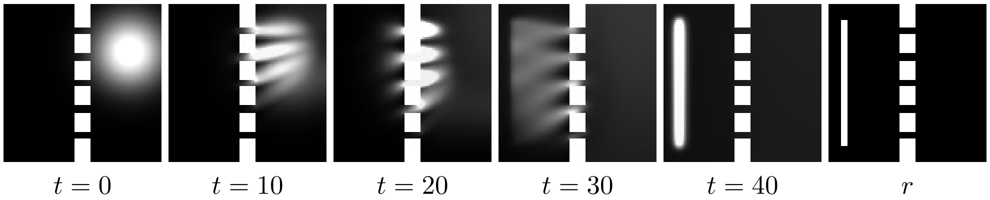


FIG. 5.1. *Examples of gradient flows for the Wasserstein attraction toward the density $r$ displayed on the right column. The congestion parameter is set to $\kappa = \|p_{t=0}\|_\infty$.*

The function $f$ thus introduce a coupling between densities during the evolution. For simplicity we consider in the following the case of 2 densities.

*Wasserstein pairwise attraction.* We first consider the case where the coupling is a Wasserstein attraction between the two densities

$$f(p_1, p_2) = \alpha W(p_1, p_2) + h_1(p_1) + h_1(p_2)$$

where the functions $h_i$ are "simple" so that one can compute easily $\mathrm{Prox}_{h_i}^{\overline{\mathrm{KL}}}$.

Denoting $q = (q_1, q_2) \overset{\text{def.}}{=} (p_{1,t}, p_{2,t})$ the previous iterate at time $t$, the solution $p = p_{t+1}$ to the JKO step (3.1) can be written as

$$p = (p_1, p_2) = (\pi_1 \mathbb{1}, \pi_2 \mathbb{1}) = (\pi_3^T \mathbb{1}, \pi_3^T \mathbb{1}),$$

where one needs to solve for $M = 3$ couplings $(\pi_1, \pi_2, \pi_3)$ the problem (5.1) with the functionals

$$\psi_1(a_1, a_2, a_3) = \iota_{\mathcal{D}}(a_1, a_3) + \iota_{\{q_2\}}(a_2) + \frac{\tau}{\gamma} h_1(a_1),$$
$$\psi_2(a_1, a_2, a_3) = \iota_{\mathcal{D}}(a_2, a_3) + \iota_{\{q_1\}}(a_1) + \frac{\tau}{\gamma} h_2(a_2)$$

with KL weights $\lambda = (1, 1, \tau\alpha)$, and where $\mathcal{D} \overset{\text{def.}}{=} \{(a, b)\ ;\ a = b\}$. The proximal operators of these functions are easy to compute as detailed in the following proposition.

PROPOSITION 4. *For $i \in \{1, 2\}$, denoting $j = 3 - i \in \{1, 2\}$, one has*

$$(b_m)_m = \mathrm{Prox}_{\psi_i}^{\overline{\mathrm{KL}}_\lambda}(a_m)_m \quad \textit{where} \quad b_i = b_3 = \mathrm{Prox}_{h_i}(a_i^{\frac{1}{1+\tau\alpha}} \odot a_3^{\frac{\tau}{1+\tau\alpha}}), \quad b_j = q_j.$$

*Proof.* The expression for $(b_i, b_3)$ is obtained by using (A.4). The expression for $b_j$ is obtained by using (A.2). □

In the numerical example, we used $h_i(p_i) \overset{\text{def.}}{=} \iota_{[0,\kappa_i]^N}(p_i) + \langle w_i,\, p_i \rangle$ for potentials $(w_1, w_2) \in \mathbb{R}^N \times \mathbb{R}^N$ and thresholds $(\kappa_1, \kappa_2) \in \mathbb{R}_+^2$. The KL proximal operator of these functions can be computed as detailed in Proposition 2. Figure 5.2 displays the results obtained on a rectangular grid.

*Summation couplings.* Another way to introduce some interaction between $p_1$ and $p_2$ is to consider a coupling on the sum

$$f(p_1, p_2) \overset{\text{def.}}{=} h(p_1 + p_2) + \langle p_1,\, w_1 \rangle + \langle p_2,\, w_2 \rangle.$$

for some function $h$ for which one can compute easily $\mathrm{Prox}_h^{\mathrm{KL}}$.

In this case, the solution $p = p_{t+1}$ to the JKO step (3.1) can be written as $p = (p_1, p_2) = (\pi_1 \mathbb{1}, \pi_2 \mathbb{1})$ where the $M = 2$ couplings $(\pi_1, \pi_2)$ solves problem (5.1)

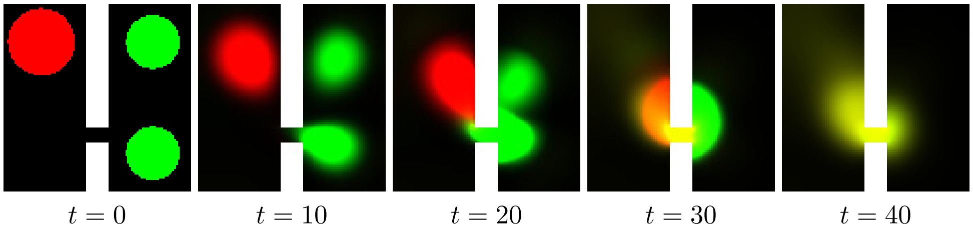


FIG. 5.2. *Evolution with a pairwise attraction between two densities, with congestion parameter $\kappa_i = \|p_{i,t=0}\|_\infty$ and $\alpha = 1$. Display of both $p_{1,t}$ (red) and $p_{2,t}$ (green), yellow indicates a mixing. Top row: $w_1 = w_2$ are potentials with constant gradients $\nabla w_i = (-1, 0)$, and the domain is a square. Middle and bottom row: $w_1 = w_2 = 0$, and the domain is a non-convex subset of the square.*

with the functionals

$$\psi_1(a_1, a_2) = \frac{\tau}{\gamma} f(a_1, a_2),$$
$$\psi_2(a_1, a_2) = \iota_{\{q_1, q_2\}}(a_1, a_2)$$

and weights $\lambda = (1, 1)$. The proximal operators of the functions are easy to compute as detailed in the following proposition.

PROPOSITION 5. *One has*

$$\mathrm{Prox}_{\psi_1}^{\mathrm{KL}_\lambda}(a_1, a_2) = \frac{\mathrm{Prox}_{\frac{\tau}{\gamma}h}^{\overline{\mathrm{KL}}}(\tilde{a}_1 + \tilde{a}_2)}{\tilde{a}_1 + \tilde{a}_2} \odot (\tilde{a}_1, \tilde{a}_2)$$
$$\mathrm{Prox}_{\psi_2}^{\mathrm{KL}_\lambda}(a_1, a_2) = (q_1, q_2).$$

*where* $\tilde{a}_i = a_i \odot e^{-\frac{\tau}{\gamma} w_i}$

*Proof.* The expression for $\mathrm{Prox}_{\psi_1}^{\mathrm{KL}_\lambda}$ is obtained by combining (A.3) and (A.6). The expression for $\mathrm{Prox}_{\psi_2}^{\mathrm{KL}_\lambda}$ is obtained by using (A.2). □

As a first example, we consider an entropic coupling $h = E$, with $w_1 = w_2 = 0$. Its proximal operator is

$$\forall\, p \in \Sigma_N, \quad \mathrm{Prox}_{\sigma E}^{\overline{\mathrm{KL}}}(p) = (p_i^{\frac{1}{1+\sigma}})_i.$$

A formal computation shows that, for the Euclidean $W_2$ transport on $\mathbb{R}^d$, the corresponding discrete JKO steps (2.2) is intended at approximating the non-linear PDE over $p_t = (p_{1,t}, p_{2,t})$

$$\forall\, i \in \{1, 2\}, \quad \forall\, t > 0, \quad \partial_t p_{i,t} = \mathrm{div}\left( \frac{p_{i,t}}{p_{1,t} + p_{2,t}} \nabla p_{i,t} \right).$$

This shows that while $p_t$ follows a non-linear coupled diffusion, $p_{1,t} + p_{2,t}$ follows a linear heat diffusion. Figure 5.3 shows a numerical illustration on a regular grid.

As a second example, we consider a congestion coupling $h = \iota_{[0,\kappa]^N}$. Its proximal operator is computed in (4.3). Figure 5.4 shows a numerical illustration on a regular grid. It shows two densities, initially supported on non-overlapping squares, moving in opposite directions under potentials $(w_1, w_2)$ such that $\nabla w_1 = (1, 0)^T$ and $\nabla w_2 = (-1, 0)^T$ (constant horizontal gradients). A congestion shock is created by the overlap of the densities, which in turn forces the support of the densities to be deformed and vertically enlarged.

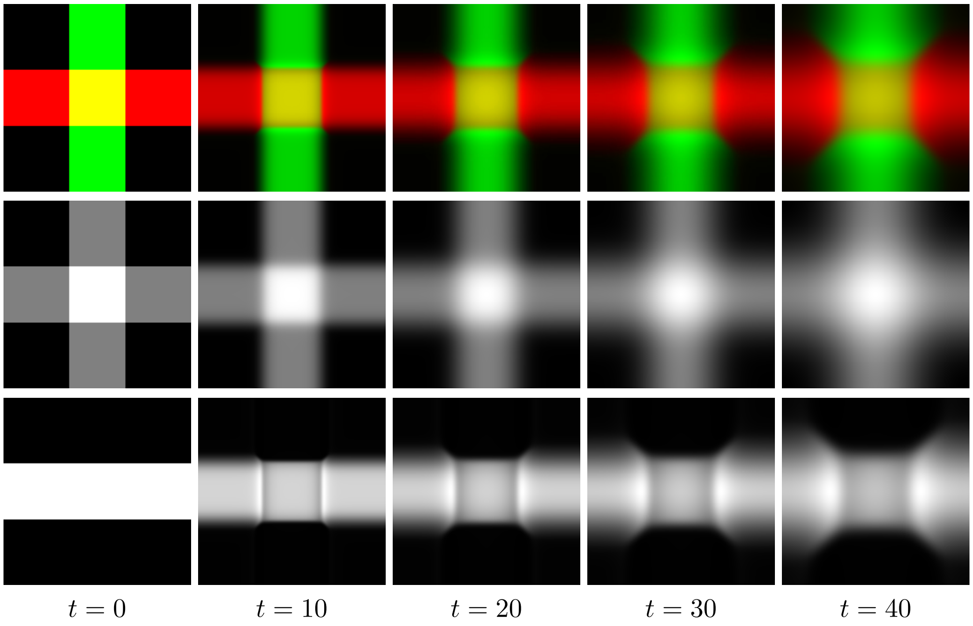


FIG. 5.3. *Evolution with a summation coupling $E(p_1 + p_2)$. Top row: display of both $p_1$ (red) and $p_2$ (green), yellow indicates a mixing. Middle row: display of $p_1 + p_2$, which evolves according to a linear heat diffusion. Bottom row: display of $p_1$.*

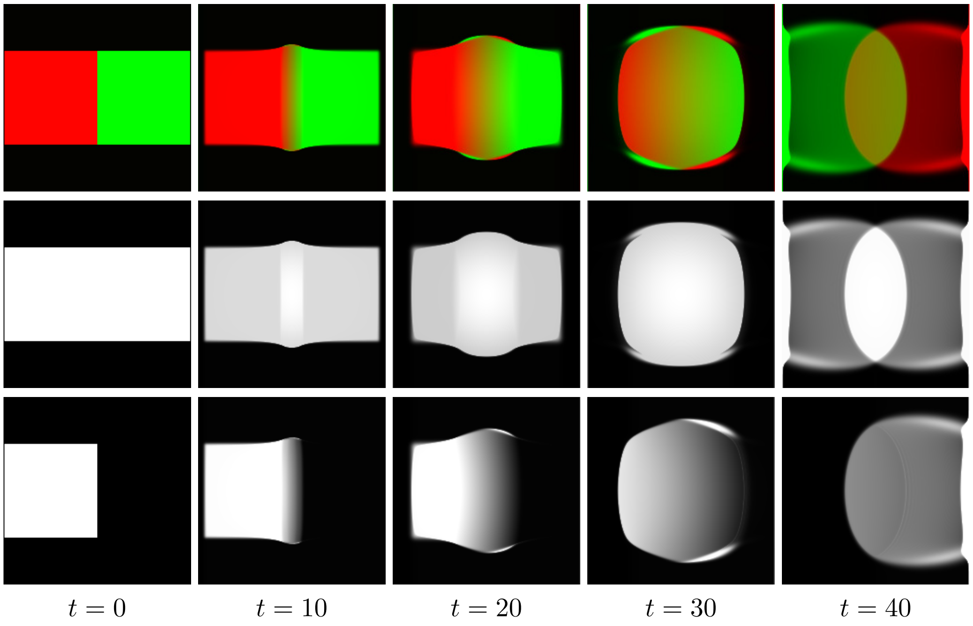


FIG. 5.4. *Evolution with a summation coupling $\iota_{[0,\kappa]^N}(p_1 + p_2)$ where $\kappa = \|p_{1,t=0}\|_\infty = \|p_{2,t=0}\|_\infty$. Top row: display of both $p_{1,t}$ (red) and $p_{2,t}$ (green), yellow indicates a mixing. Middle row: display of $p_{1,t} + p_{2,t}$. Bottom row: display of $p_{1,t}$.*

**Discussion and Conclusion.** In this paper, we have presented a novel algorithm to compute approximate discrete gradient flows according to an entropic smoothing of the Wasserstein distance. The main interest of the method is its speed, simplicity and versatility. This is achieved because the iterations only require (beside pointwise multiplications, divisions and exponentiations) to compute the successive applications of a "convolution-like" operator corresponding to the Gibbs kernel associated to the metric.

A natural question is to explore whether the discrete flow defined by (2.2) has a continuous limit when $\tau_t = \tau \to 0$. If one uses a fixed $\gamma_t = \gamma > 0$, this is not the case, because $W_\gamma$ does not satisfies $W_\gamma(p,p) = 0$. More precisely, one has that

$$\underset{q}{\operatorname{argmin}}\; W_\gamma(p,q) = \xi\left(\frac{p}{\xi^T(p)}\right),$$

so that the limit for small $\tau$ of $p_{t+1}$ defined by (2.2) is a blurred (i.e. multiplied by $\xi$) version of $p_t$. An interesting area of future work is to study the setting where $\gamma_t$ is chosen as a function of $\tau_t$.

**Acknowledgements.** This work has been supported by the European Research Council (ERC project SIGMA-Vision). I would like to acknowledge stimulating discussions with Marco Cuturi, Justin Solomon, Jean-David Benamou, Guillaume Carlier and Quentin Merigot. I would like to thank Guillaume Carlier for suggesting to apply the method to multiple densities (Section 5.3). I would like to thank Jean-Marie Mirebeau for giving me access to his code for anisotropic diffusion. I would like to thank Antonin Chambolle and Jalal Fadili for suggesting me the proof strategy of Proposition 1.

## Appendix A. KL Proximal Calculus.

The following proposition details some useful property of the $\overline{\mathrm{KL}}$ proximal operator (2.3). This enables a powerful "proximal calculus" by combining these rules, which eases and simplifies the implementation of the algorithms. Note that we also consider generalized KL divergence over sets $(p_1, \ldots, p_M)$ of $M$ densities according to some weight $\lambda \in \mathbb{R}_+^M$

$$\forall\, (p_m)_m, (q_m)_m, \quad \overline{\mathrm{KL}}_\lambda((p_m)_m|(q_m)_m) \overset{\text{def.}}{=} \sum_{m=1}^M \lambda_m \overline{\mathrm{KL}}(p_m|q_m). \tag{A.1}$$

Proposition A.1. *For* $f(p_1, \ldots, p_M) \overset{\text{def.}}{=} \iota_{\{(q_1,\ldots,q_M)\}}(p_1, \ldots, p_M)$*, one has*

$$\mathrm{Prox}_f^{\overline{\mathrm{KL}}_\lambda}(p_1, \ldots, p_M) = (q_1, \ldots, q_M). \tag{A.2}$$

*For* $f(p_1, \ldots, p_M) \overset{\text{def.}}{=} h(p_1, \ldots, p_M) + \sum_{i=1}^M \langle w_i,\, p_i\rangle$*, one has*

$$\mathrm{Prox}_f^{\overline{\mathrm{KL}}_\lambda}(p_1, \ldots, p_M) = \mathrm{Prox}_h^{\overline{\mathrm{KL}}_\lambda}(p_1 \odot e^{-w_1/\lambda_1}, \ldots, p_M \odot e^{-w_M/\lambda_M}). \tag{A.3}$$

*For* $f(p_1, \ldots, p_M) \overset{\text{def.}}{=} \iota_{\mathcal{D}}(p_1, \ldots, p_M) + h(p_1, \ldots, p_M)$ *where*

$$\mathcal{D} \overset{\text{def.}}{=} \{(p_1, \ldots, p_M)\,;\; p_1 = \ldots = p_M\},$$

*one has*

$$\mathrm{Prox}_f^{\overline{\mathrm{KL}}_\lambda}(p_1, \ldots, p_M) = (p, \ldots, p) \quad \textit{where} \quad p = \mathrm{Prox}^{\mathrm{KL}}_{\frac{1}{\sum_i \lambda_i}\tilde{h}}\left(p_1^{\tilde\lambda_1} \odot \ldots \odot p_M^{\tilde\lambda_M}\right), \tag{A.4}$$

*where we denoted* $\tilde{\lambda}_i \overset{\text{def.}}{=} \lambda_i / \sum_j \lambda_j$ *and* $\tilde{h}(p) = h(p, \dots, p)$.
*For* $f(p_1, \dots, p_M) \overset{\text{def.}}{=} h(p_1 + \dots + p_M)$ *and* $\lambda \overset{\text{def.}}{=} (1, \dots, 1)$, *one has*

$$\operatorname{Prox}_f^{\overline{\mathrm{KL}}_\lambda}(p_1, \dots, p_M) = \frac{\operatorname{Prox}_h^{\overline{\mathrm{KL}}}(p_1 + \dots + p_M)}{p_1 + \dots + p_M}(p_1, \dots, p_M) \tag{A.5}$$

*We define* $f(\pi_1, \dots, \pi_M) \overset{\text{def.}}{=} h(\pi_1 \mathbb{1}, \dots, \pi_M \mathbb{1})$. *We denote*

$$\forall\, m \in \{1, \dots, M\}, \quad p_m \overset{\text{def.}}{=} \pi_m \mathbb{1} \quad \textit{and} \quad (\tilde{p}_1, \dots, \tilde{p}_M) \overset{\text{def.}}{=} \operatorname{Prox}_h^{\overline{\mathrm{KL}}_\lambda}(p_1, \dots, p_M).$$

*One has*

$$\operatorname{Prox}_f^{\mathrm{KL}_\lambda}(\pi_1, \dots, \pi_M) = \left( \operatorname{diag}\left( \frac{\tilde{p}_m}{p_m} \right) \; \pi \right)_m \tag{A.6}$$

*We define* $f(\pi_1, \dots, \pi_M) \overset{\text{def.}}{=} h(\pi_1^T \mathbb{1}, \dots, \pi_M^T \mathbb{1})$. *We denote*

$$\forall\, m \in \{1, \dots, M\}, \quad p_m \overset{\text{def.}}{=} \pi_m^T \mathbb{1} \quad \textit{and} \quad (\tilde{p}_1, \dots, \tilde{p}_M) \overset{\text{def.}}{=} \operatorname{Prox}_h^{\overline{\mathrm{KL}}_\lambda}(p_1, \dots, p_M).$$

*One has*

$$\operatorname{Prox}_f^{\mathrm{KL}_\lambda}(\pi_1, \dots, \pi_M) = \left( \pi \operatorname{diag}\left( \frac{\tilde{p}_m}{p_m} \right) \right)_m \tag{A.7}$$

*Proof.* **Proof of (A.2).** This is straightforward.
**Proof of (A.3).** If follows from the relation

$$\begin{aligned} &\overline{\mathrm{KL}}_\lambda((q_1, \dots, q_M) | (p_1, \dots, p_M)) + \sum_{i=1}^{M} \langle w_i,\, q_i \rangle \\ =&\overline{\mathrm{KL}}_\lambda((q_1, \dots, q_M) | (p_1 \odot e^{-w_1/\lambda_1}, \dots, p_M \odot e^{-w_M/\lambda_M})). \end{aligned}$$

**Proof of (A.4).** We denote $(q_m)_m \overset{\text{def.}}{=} \operatorname{Prox}_{\psi_1}^{\overline{\mathrm{KL}}_\lambda}((p_m)_m)$, so that $q = q_1 = \dots = q_M$ solves

$$\min_q \sum_m \lambda_m \overline{\mathrm{KL}}(q | p_m) + \tilde{h}(q).$$

The result follow from the relation

$$\sum_m \lambda_m \overline{\mathrm{KL}}(q | p_m) = \left( \textstyle\sum_m \lambda_m \right) \overline{\mathrm{KL}}\left( q | p_1^{\tilde{\lambda}_1} \odot \dots \odot p_M^{\tilde{\lambda}_M} \right).$$

**Proof of (A.5).** Denoting $(q_m)_m = \operatorname{Prox}_f^{\overline{\mathrm{KL}}_\lambda}((p_m)_m)$, the first order optimality condition for $\operatorname{Prox}_f^{\overline{\mathrm{KL}}_\lambda}$ reads

$$\forall\, m \in \{1, \dots, M\}, \quad \log\left( \frac{q_m}{p_m} \right) + u = 0$$

where $u \in \partial h(p_1 + \dots + p_M)$. Respectively summing and subtracting these equations lead to

$$q_1 + \dots + q_M = \operatorname{Prox}_h(p_1 + \dots + p_M) \quad \text{and} \quad \frac{q_1}{p_1} = \dots = \frac{q_m}{p_m}.$$

Solving for $(q_1, \dots, q_m)$ in these equations leads to the desired solution.

**Proof of** (A.6)**.** The first order condition for $\tilde{\pi}$ being a solution of (3.5) states the existence of $(z_m)_m \in \partial f(\tilde{p}_1, \dots, \tilde{p}_M)$ where $\tilde{p}_m = \tilde{\pi}_m \mathbb{1}$ such that

$$\lambda_m \log\left(\frac{\tilde{\pi}_m}{\pi_m}\right) + z_m \mathbb{1}^T = 0 \;\Rightarrow\; \tilde{\pi}_m = \operatorname{diag}(e^{-z_m/\lambda_m})\pi_m \;\Rightarrow\; \tilde{p}_m = \operatorname{diag}(e^{-z_m/\lambda_m})p_m,$$

which corresponds to the first order condition for $(\tilde{p}_m)_m$ being a solution of (2.3) for the function $h$, i.e.

$$(\tilde{p}_m)_m = \operatorname{Prox}_h^{\overline{\mathrm{KL}}}((p_m)_m).$$

Finally, one obtains

$$\tilde{\pi}_m = \operatorname{diag}(e^{-z_m/\lambda_m})\pi_m = \operatorname{diag}\left(\frac{\tilde{p}_m}{p_m}\right)\pi_m$$

and hence the desired result.

**Proof of** (A.7)**.** It is obtained by transposing formula (A.6). $\square$

## REFERENCES

[1] S. Adams, N. Dirr, M. Peletier, and J. Zimmer. From a large-deviations principle to the wasserstein gradient flow: A new micro-macro passage. *Communications in Mathematical Physics*, 307(3):791–815, 2011.

[2] M. Agueh. *Existence of solutions to degenerate parabolic equations via the Monge-Kantorovich theory.* PhD thesis, Georgia Institute of Technology, USA, 2002.

[3] M. Agueh and M. Bowles. One-dimensional numerical algorithms for gradient flows in the $p$-Wasserstein spaces. *Acta Applicandae Mathematicae*, 125(1):121–134, 2013.

[4] M. Agueh and G. Carlier. Barycenters in the Wasserstein space. *SIAM J. on Mathematical Analysis*, 43(2):904–924, 2011.

[5] L. Ambrosio, N. Gigli, and G. Savaré. *Gradient flows: in metric spaces and in the space of probability measures.* Springer, 2006.

[6] A. Artacho, Borwein F. J., and J. M. Global convergence of a non-convex douglas-rachford iteration. *J. Global Optimization*, 57(3):1–17, 2012.

[7] H. H. Bauschke and P. L. Combettes. A Dykstra-like algorithm for two monotone operators. *Pacific Journal of Optimization*, 4(3):383–391, 2008.

[8] H. H. Bauschke and P. L. Combettes. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces.* Springer-Verlag, New York, 2011.

[9] H. H. Bauschke, P. L. Combettes, and D. R. Luke. Phase retrieval, error reduction algorithm, and Fienup variants: A view from convex optimization. *J. Opt. Soc. Am. A*, 19(7):1334–1345, 2002.

[10] H. H. Bauschke and A. S. Lewis. Dykstra's algorithm with Bregman projections: a convergence proof. *Optimization*, 48(4):409–427, 2000.

[11] 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.

[12] J-D. Benamou, G. Carlier, M. Cuturi, L. Nenna, and G. Peyré. Iterative Bregman projections for regularized transportation problems. *to appear in SIAM J. Sci. Comp.*, 2015.

[13] J-D. Benamou, G. Carlier, Q. Mérigot, and E. Oudet. Discretization of functionals involving the Monge-Ampère operator. *Preprint arXiv:1408.4536*, 2014.

[14] J. Bigot and T. Klein. Consistent estimation of a population barycenter in the Wasserstein space. *Preprint arXiv:1212.2562*, 2012.

[15] A. Blanchet, V. Calvez, and J. A Carrillo. Convergence of the mass-transport steepest descent scheme for the subcritical patlak-keller-segel model. *SIAM Journal on Numerical Analysis*, 46(2):691–721, 2008.

[16] A. Blanchet and G. Carlier. Optimal transport and Cournot-Nash equilibria. *arXiv preprint arXiv:1206.6571*, 2012.

[17] N. Bonneel, M. van de Panne, S. Paris, and W. Heidrich. Displacement interpolation using Lagrangian mass transport. *ACM Transactions on Graphics (SIGGRAPH ASIA'11)*, 30(6), 2011.

[18] M. Botsch, L. Kobbelt, M. Pauly, P. Alliez, and B. Levy. *Polygon Mesh Processing*. Taylor & Francis, 2010.

[19] L. M. Bregman. The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. *USSR computational mathematics and mathematical physics*, 7(3):200–217, 1967.

[20] L.M. Bregman, Y. Censor, and S. Reich. Dykstra's algorithm as the nonlinear extension of Bregman's optimization method. *Journal of Convex Analysis*, 6:319–333, 1999.

[21] Y. Brenier. The least action principle and the related concept of generalized flows for incompressible perfect fluids. *J. of the AMS*, 2:225–255, 1990.

[22] C.J. Budd, M.J.P. Cullen, and E.J. Walsh. Monge-Ampère based moving mesh methods for numerical weather prediction, with applications to the eady problem. *Journal of Computational Physics*, 236:247–270, 2013.

[23] M. Burger, J. A. Carrillo, and M-T. Wolfram. A mixed finite element method for nonlinear diffusion equations. *Kinetic and Related Models*, 3(1):59–83, 2010.

[24] M. Burger, M. Franeka, and C-B. Schonlieb. Regularised regression and density estimation based on optimal transport. *Appl. Math. Res. Express*, 2:209–253, 2012.

[25] J. A. Carrillo, A. Chertock, and Y. Huang. A finite-volume method for nonlinear nonlocal equations with a gradient flow structure. *Communications in Computational Physics*, 17:233–258, 1 2015.

[26] J. A Carrillo and J. S. Moll. Numerical simulation of diffusive and aggregation phenomena in nonlinear continuity equations by evolving diffeomorphisms. *SIAM Journal on Scientific Computing*, 31(6):4305–4329, 2009.

[27] Y. Censor and S. Reich. The Dykstra algorithm with Bregman projections. *Communications in Applied Analysis*, 2:407–419, 1998.

[28] A. Chambolle and T. Pock. On the ergodic convergence rates of a first-order primal-dual algorithm. *preprint*, 2014.

[29] S-N. Chow, W. Huang, Y. Li, and H. Zhou. Fokker-planck equations for a free energy functional or markov process on a graph. *Archive for Rational Mechanics and Analysis*, 203(3):969–1008, 2012.

[30] P. G. Ciarlet. *Introduction to Numerical Linear Algebra and Optimisation*. Cambridge University Press, Cambridge, 1989. Originally published in French under the title, *Introduction à l'analyse numérique matricielle et à l'optimisation* in 1982.

[31] K. Crane, C. Weischedel, and M. Wardetzky. Geodesics in heat: A new approach to computing distance based on heat flow. *ACM Trans. Graph.*, 32(5):152:1–152:11, October 2013.

[32] I. Csiszár. *I*-divergence geometry of probability distributions and minimization problems. *Ann. Probability*, 3:146–158, 1975.

[33] M. Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems (NIPS) 26*, pages 2292–2300, 2013.

[34] M. Cuturi and A. Doucet. Fast computation of Wasserstein barycenters. In *Proceedings of the 31st International Conference on Machine Learning (ICML), JMLR W&CP*, volume 32, 2014.

[35] T. A. Davis. *Direct Methods for Sparse Linear Systems*. SIAM, 2006.

[36] W. E. Deming and F. F. Stephan. On a least squares adjustment of a sampled frequency table when the expected marginal totals are known. *Annals Mathematical Statistics*, 11(4):427–444, 1940.

[37] R. Deriche. Recursively implementing the Gaussian and its derivatives. Technical Report RR-1893, INRIA, 1993.

[38] R. L. Dykstra. An iterative procedure for obtaining *I*-projections onto the intersection of convex sets. *Ann. Probab.*, 13(3):975–984, 1985.

[39] J. Eckstein. Nonlinear proximal point algorithms using Bregman functions, with applications to convex programming. *Mathematics of Operations Research*, 18(1):202–226, 1993.

[40] M. Erbar. The heat equation on manifolds as a gradient flow in the Wasserstein space. *Annales de l'Institut Henri Poincaré, Probabilités et Statistiques*, 46(1):1–23, 2010.

[41] J. Fehrenbach and J-M. Mirebeau. Sparse non-negative stencils for anisotropic diffusion. *Journal of Mathematical Imaging and Vision*, 49(1):123–147, 2014.

[42] A. Figalli. The optimal partial transport problem. *Arch. Ration. Mech. Anal.*, 195(2):533–560, 2010.

[43] U. Frisch, S. Matarrese, R. Mohayaee, and A. Sobolevski. Monge-Ampere-Kantorovitch (MAK) reconstruction of the early universe. *Nature*, 417(260), 2002.

[44] U. Gianazza, G. Savaré, and G. Toscani. The wasserstein gradient flow of the Fisher information and the quantum drift-diffusion equation. *Archive for Rational Mechanics and Analysis*, 194(1):133–220, 2009.

[45] K. Jonathan and R. J. McCann. Insights into capacity constrained optimal transport. *Proc. Natl. Acad. Sci. USA*, 110:10064–10067, 2013.

[46] R. Jordan, D. Kinderlehrer, and O. Otto. The variational formulation of the Fokker-Planck equation. *SIAM journal on mathematical analysis*, 29(1):1–17, 1998.

[47] L. Kantorovich. On the transfer of masses (in russian). *Doklady Akademii Nauk*, 37(2):227–229, 1942.

[48] D. Kinderlehrer and N. J Walkington. Approximation of parabolic equations using the Wasserstein metric. *ESAIM: Mathematical Modelling and Numerical Analysis*, 33(04):837–852, 1999.

[49] K.C. Kiwiel. Proximal minimization methods with generalized Bregman functions. *SIAM J. Control Optim.*, 35(4):1142–1168, 1997.

[50] C. Leonard. A survey of the Schrodinger problem and some of its connections with optimal transport. *Discrete Contin. Dyn. Syst. A*, 34(4):1533–1574, 2014.

[51] J. Maas. Gradient flows of the entropy for finite markov chains. *Journal of Functional Analysis*, 261(8):2250–2292, 2011.

[52] D. Matthes and H. Osberger. Convergence of a variational lagrangian scheme for a nonlinear drift diffusion equation. *Preprint arXiv:1301.0747*, 2014.

[53] B. Maury, A. Roudneff-Chupin, and F. Santambrogio. A macroscopic crowd motion model of gradient flow type. *Mathematical Models and Methods in Applied Sciences*, 20(10):1787–1821, 2010.

[54] A. Mielke. Geodesic convexity of the relative entropy in reversible markov chains. *Calculus of Variations and Partial Differential Equations*, 48(1-2):1–31, 2013.

[55] Y. Nesterov, A. Nemirovskii, and Y. Ye. *Interior-point polynomial algorithms in convex programming*, volume 13. SIAM, 1994.

[56] F. Otto. The geometry of dissipative evolution equations: the porous medium equation. *Communications in partial differential equations*, 26(1-2):101–174, 2001.

[57] N. Papadakis, G. Peyré, and E. Oudet. Optimal transport with proximal splitting. *SIAM Journal on Imaging Sciences*, 7(1):212–238, 2014.

[58] B. Pass. On the local structure of optimal measures in the multi-marginal optimal transportation problem. *Calc. Var. Partial Differential Equations*, 43(3-4):529–536, 2012.

[59] Y. Rubner, C. Tomasi, and L.J. Guibas. The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision*, 40(2), 2000.

[60] L. Ruschendorf and W. Thomsen. Closedness of sum spaces and the generalized Schrodinger problem. *Theory of Probability and its Applications*, 42(3):483–494, 1998.

[61] E. Schrodinger. Uber die umkehrung der naturgesetze. *Sitzungsberichte Preuss. Akad. Wiss. Berlin. Phys. Math.*, 144:144–153, 1931.

[62] J.A. Sethian. *Level Sets Methods and Fast Marching Methods.* Cambridge University Press, 2nd edition, 1999.

[63] R. Sinkhorn. A relationship between arbitrary positive matrices and doubly stochastic matrices. *Ann. Math. Statist.*, 35:876–879, 1964.

[64] R. Sinkhorn. Diagonal equivalence to matrices with prescribed row and column sums. *Amer. Math. Monthly*, 74:402–405, 1967.

[65] R. Sinkhorn and P . Knopp. Concerning nonnegative matrices and doubly stochastic matrices. *Pacific J. Math.*, 21:343–348, 1967.

[66] M. V. Solodov and B. F. Svaiter. An inexact hybrid generalized proximal point algorithm and some new results on the theory of Bregman functions. *Mathematics of Operations Research*, 25(2):214–230, 2000.

[67] J. Solomon, F. de Goes, G. Peyré, M. Cuturi, A. Butscher, A. Nguyen, T. Du, and L. Guibas. Convolutional Wasserstein distances: Efficient optimal transportation on geometric domains. *Preprint*, 2015.

[68] S. R. S. Varadhan. On the behavior of the fundamental solution of the heat equation with variable coefficients. *Communications on Pure and Applied Mathematics*, 20(2):431–455, 1967.

[69] C. Villani. *Topics in Optimal Transportation.* Graduate Studies in Mathematics Series. American Mathematical Society, 2003.

[70] H. Wang and A. Banerjee. Bregman alternating direction method of multipliers. *preprint arXiv:1306.3203*, 2014.

[71] J. Weickert. *Anisotropic Diffusion in Image Processing.* Teubner, Stuttgart, 1998.

[72] M. Westdickenberg and J. Wilkening. Variational particle schemes for the porous medium equation and for the system of isentropic Euler equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 44(1):133–166, 2010.

[73] G-S. Xia, S. Ferradans, G. Peyré, and J-F. Aujol. Synthesizing and mixing stationary Gaussian

texture models. *SIAM Journal on Imaging Sciences*, 7(1):476–508, 2014.