Acta Appl Math
DOI 10.1007/s10440-012-9783-2

One-Dimensional Numerical Algorithms for Gradient Flows in the p-Wasserstein Spaces

Martial Agueh · Malcolm Bowles

Received: 27 January 2012 / Accepted: 24 September 2012

Abstract We numerically approximate, on the real line, solutions to a large class of parabolic partial differential equations which are "gradient flows" of some energy functionals with respect to the L^p-Wasserstein metrics for all $p > 1$. Our method relies on variational principles involving the optimal transport problem with general strictly convex cost functions.

Keywords L^p-Wasserstein metric · Numerical approximation · Gradient flow · Parabolic partial differential equation

Mathematics Subject Classification (2010) 65M22 · 49M25 · 35K20

1 Introduction

In what follows, Ω denotes an open, bounded, convex and smooth subset of $\mathbb{R}^n$, where $n \geq 1$ is an integer. Let $c : \mathbb{R}^n \to [0, \infty)$ be a strictly convex symmetric function, and $u, v : \Omega \to [0, \infty)$ be two probability densities on Ω; $u, v \in \mathcal{P}(\Omega)$. The optimal transport (or Monge-Kantorovich) problem between $u(x)$ and $v(y)$ with the cost function $c(x - y)$ consists of finding the—optimal transport—map $T : \Omega \to \Omega$ that minimizes the total cost,

$$W_c(u, v) := \inf_{T:\Omega\to\Omega} \left\{ \int_\Omega c\big(x - T(x)\big)u(x)\,\mathrm{d}x;\, T_\# u = v \right\}, \tag{1}$$

M. Agueh (✉) · M. Bowles
Department of Mathematics and Statistics, University of Victoria, PO. Box 3060 STN CSC, Victoria, BC, Canada, V8W 3R4
e-mail: agueh@math.uvic.ca

M. Bowles
e-mail: mbowles@uvic.ca

Published online: 03 October 2012

of transporting u onto v, where $c(x-y)$ is the basic cost of transporting a unit mass of matter from a position $x \in \Omega$ to a location $y \in \Omega$, and $T_{\#}u = v$ means that $v(B) = u(T^{-1}(B))$ for all Borel sets $B \subset \mathbb{R}^n$, or equivalently

$$\int_{\Omega} \varphi(y)v(y)\,\mathrm{d}y = \int_{\Omega} \varphi\big(T(x)\big)u(x)\,\mathrm{d}x, \quad \forall \varphi \in C(\Omega). \tag{2}$$

When the cost function is $c(z) = |z|^p/p$, $p > 1$, the corresponding optimal transport problem is known as the L^p optimal transport problem. Under the conditions on the cost function c, it is known that problem (1) has a unique solution (a.e. with respect to u) which is characterized by the gradient of a convex function [5] when $c(z) = |z|^2/2$, or via a c-concave function in general [6, 9]. Moreover when the cost function is $c(z) = |z|^p/p$ for any $p > 1$, then (1) defines a metric, $d_p(u, v) := (W_c(\mu, \nu))^{1/p}$, over the set of probability densities on Ω, called the L^p-Wasserstein metric, see [17].

Here, we are interested in applications of the L^p-Wasserstein metrics to partial differential equations (PDE's). It is known [1, 4, 10, 12] that solutions to parabolic PDE's of the form

$$\begin{cases} \frac{\partial \rho}{\partial t} = \nabla \cdot \{\rho \nabla c_*[\nabla(F'(\rho) + V + W \star \rho)]\} & \text{in } \Omega \times (0,\infty) \\ \rho \nabla c^*[\nabla(F'(\rho) + V + W \star \rho)] \cdot \nu = 0 & \text{on } \partial\Omega \times (0,\infty) \\ \rho(t=0) = \rho_0 & \text{on } \Omega \end{cases} \tag{3}$$

can be obtained variationally via the following time-discrete iterative scheme involving the optimal transport problem with the cost function $c_\tau(z) := c(z/\tau)$,

$$\rho_k := \mathrm{Argmin}\left\{ W_{c_\tau}(\rho, \rho_{k-1}) + \frac{1}{\tau}E(\rho) :\ \rho \in \mathcal{P}(\Omega) \right\}, \tag{4}$$

where $\rho_0 \in \mathcal{P}(\Omega)$, $\tau > 0$ denotes a time-step size, c_* is the Legendre transform of c, and the energy functional E is defined by

$$E(\rho) = \int_{\Omega} \left[F(\rho) + \rho V + \frac{1}{2}\rho(W \star \rho) \right] \mathrm{d}x, \tag{5}$$

for some given functions $F : [0,\infty) \to \mathbb{R}$, $V : \Omega \to \mathbb{R}$ and $W : \mathbb{R}^n \to \mathbb{R}$; here $W \star \rho$ denotes the convolution of W and ρ. This family of PDE's contains the heat equation, the Fokker-Planck equation and the porous medium equation, where the cost function is $c(z) = |z|^2/2$; the doubly nonlinear equation, where $c(z) = |z|^q/q$ with $q = p/(p-1)$ and $F(x) = \frac{mx^\gamma}{\gamma(\gamma-1)}$ with $\gamma = m + \frac{p-2}{p-1}$, and in particular the parabolic p-Laplacian equation where $m = 1$ in the above choice of F. These are well-known equations used to model many phenonema in mathematical physics. For other models in mathematical biology included in this class of PDE's (3), we refer to [8] and the references therein.

In [3, 4, 7, 13], it is shown that PDE (3) can be interpreted as a gradient flow of the energy functional (5) w.r.t. the optimal transport cost W_c. Here, we are concerned with the numerical approximation of the solution to PDE (3), using the scheme (4). For that, we are inspired by the work of Kinderlehrer and Walkington [11] where some numerical algorithms were given in one dimension, to solve (3) in the particular case when the cost function is quadratic, $c(z) = |z|^2/2$. We propose to extend their work to all strictly convex cost functions, then solving numerically all PDE's of the form (3) in a one dimensional space. More specifically, we will focus on the following, more general class of parabolic PDE's with forcing terms, which includes the non-homogeneous analogue of (3), in one dimension:

$$\begin{cases} \frac{\partial u}{\partial t} - \frac{\partial}{\partial x}(uc_*'[\frac{\partial}{\partial x}(F'(u) + V + W \star u)]) = f(x,t) & \text{in } \Omega \times (0,\infty) \\ uc_*'[\frac{\partial}{\partial x}(F'(u) + V + W \star u)] = 0 & \text{on } \partial\Omega \times (0,\infty) \\ u(t=0) = u^0 & \text{in } \Omega \end{cases} \tag{6}$$

where $u^0 : \Omega \to [0, \infty)$ belongs to $L^1(\Omega)$, $c : \mathbb{R} \to [0, \infty)$ is a strictly convex symmetric cost function, c_* denotes the Legendre transform of c, $c_*(x) := \sup_{y \in \mathbb{R}} \{xy - c(y)\}$, and $\star$ stands for the convolution operation, $W \star u(x) := \int_\Omega W(x - y)u(y)\,\mathrm{d}y$. We will assume throughout the paper that $c \in C^1(\mathbb{R})$, $\lim_{|x| \to \infty} \frac{c(x)}{|x|} = \infty$, and we fix $\Omega := (0, 1)$. A typical example of c is $c(x) = |x|^q/q$ for any $q > 1$, in which case $c_*(x) = |x|^p/p$ where $\frac{1}{p} + \frac{1}{q} = 1$. The functions $f : \Omega \times [0, \infty) \to [0, \infty)$, $F : [0, \infty) \to \mathbb{R}$, $V : \Omega \to \mathbb{R}$ and $W : \mathbb{R} \to \mathbb{R}$ are sufficiently smooth and satisfy the assumptions in [1, 4, 14] which guarantee the existence of a unique solution to PDE (6) via the variational scheme

$$u^n := \mathrm{Argmin}\left\{ W_{c_\tau}\big(u, v^{n-1}\big) + \frac{1}{\tau} E(u) : \int_\Omega u(x)\,\mathrm{d}x = \int_\Omega v^{n-1}(x)\,\mathrm{d}x \right\}, \tag{7}$$

where $u : \Omega \to [0, \infty)$, $v^{n-1}(x) = u^{n-1}(x) + \int_{(n-1)\tau}^{n\tau} f(x, t)\,\mathrm{d}t$, and $u^0 : \Omega \to [0, \infty)$ is given in $L^1(\Omega)$. The existence of solutions to (6) can also formally be seen from the Euler-Lagrange equation to the variational problem (7) that we recall in Sect. 2, using the explicit formula of the optimal transport cost (1) in one dimension, as we show below. For simplicity, assume that $u, v : \Omega \to (0, \infty)$ are continuous with $\int_\Omega u(x)\,\mathrm{d}x = \int_\Omega v(x)\,\mathrm{d}x$. If $T : \Omega \to \Omega$ denotes the optimal transport map in (1), then T is C^1, and by the change of variable $y = T(x)$ in (2), we have the identity

$$v\big(T(x)\big)T'(x) = u(x), \tag{8}$$

which by integration, gives $V(T(x)) = U(x)$, where $U(x) = \int_0^x u(y)\,\mathrm{d}y$ and $V(x) = \int_0^x v(y)\,\mathrm{d}y$ are respectively the cumulative distribution functions of u and v. Therefore, the optimal transport map in (1) is explicitly given by

$$T(x) = V^{-1} \circ U(x). \tag{9}$$

Inserting formula (9) into (1) and using the substitution $\eta = U(x)$, we obtain the formula of the optimal transport cost (1) in one dimension,

$$W_c(u, v) = \int_0^1 c\big(x - V^{-1} \circ U(x)\big)u(x)\,\mathrm{d}x = \int_0^{U(1)} c\big(U^{-1}(\eta) - V^{-1}(\eta)\big)\,\mathrm{d}\eta. \tag{10}$$

The rest of the paper is organized as follows. In Sect. 3, we present the numerical algorithms that will be used to approximate the solutions to PDE (6). These algorithms extend to all strictly convex symmetric cost functions, c, the ones given in [11] for the quadratic cost $c(z) = |z|^2/2$. In Sect. 4, we test the algorithms on some numerical examples, and we conclude with some remarks in Sect. 5.

2 Gradient Flows

Here we derive the Euler-Lagrange equation to the discrete scheme (7) of PDE (6) in $\Omega = (0, 1)$. This derivation follows the lines of the proof in [11] for the quadratic cost $c(z) = |z|^2/2$, but mainly relies on ideas in [1]. Indeed, assume that $u^{n-1} : \Omega \to (0, \infty)$ is a given integrable function, and denote by u^n the minimizer of the variational problem (7), that we also assume to be positive; (the positivity of the u^n are guaranteed provided that $u^0 > M$ for some $M > 0$, see e.g., [1, 16]). Our goal is to show that u^n satisfies, in a weak sense, the following equation which can be viewed as a discrete formulation of PDE (6) for a time step $\tau > 0$ small enough,

$$\frac{u^n - u^{n-1}}{\tau} - \frac{\mathrm{d}}{\mathrm{d}x}\left[u^n c_*'\left(\frac{\mathrm{d}}{\mathrm{d}x}\left[\left(F'(u^n) + V + W \star u^n\right]\right)\right)\right]$$
$$= \frac{1}{\tau}\int_{(n-1)\tau}^{n\tau} f(x,t)\,\mathrm{d}t + \text{Correction terms}. \tag{11}$$

In fact, (11) justifies that (7) is a correct discrete scheme for PDE (6) or in other words, that PDE (6) can be interpreted as a gradient flow of the energy functional (5) w.r.t. the optimal transport cost (1).

Let $g : \Omega \to \Omega$ be a continuous function, and set $G(x) = \int_0^x g(y)\,\mathrm{d}y$. We assume that $G(1) = \int_\Omega g(y)\,\mathrm{d}y = 0$. Consider now the variation $u_\epsilon = u^n + \epsilon g$ of u^n for some $\epsilon \in \mathbb{R}$ small enough, and denote by $U_\epsilon(x) = \int_0^x u_\epsilon(y)\,\mathrm{d}y$ the cumulative distribution function of u_ϵ. Since $G(0) = G(1) = 0$, then u_ϵ is admissible in (7) and we have

$$0 = \frac{\mathrm{d}I(u_\epsilon)}{\mathrm{d}\epsilon}\bigg|_{\epsilon=0} = \frac{\mathrm{d}}{\mathrm{d}\epsilon}W_{c_\tau}\left(u_\epsilon, v^{n-1}\right)\bigg|_{\epsilon=0} + \frac{1}{\tau}\frac{\mathrm{d}E(u_\epsilon)}{\mathrm{d}\epsilon}\bigg|_{\epsilon=0}, \tag{12}$$

where

$$I(u) := W_{c_\tau}\left(u, v^{n-1}\right) + \frac{1}{\tau}E(u), \quad E(u) := \int_\Omega \left[F(u) + uV + \frac{1}{2}u(W \star u)\right]\mathrm{d}x. \tag{13}$$

By a direct computation where we use that W is symmetric, the second term in (12) gives

$$\frac{\mathrm{d}E(u_\epsilon)}{\mathrm{d}\epsilon}\bigg|_{\epsilon=0} = \int_\Omega \left[F'\left(u^n(x)\right) + V(x) + W \star u^n(x)\right]g(x)\,\mathrm{d}x. \tag{14}$$

To compute the first term in (12), we first note using (9), that $T_\epsilon(x) = V_{n-1}^{-1} \circ U_\epsilon(x)$ (resp. $T(x) = V_{n-1}^{-1} \circ U_n(x)$) is the optimal map in $W_{c_\tau}(u_\epsilon, v^{n-1})$ (resp. $W_{c_\tau}(u^n, v^{n-1})$), where

$$V_{n-1}(x) = \int_0^x v^{n-1}(y)\,\mathrm{d}y, \qquad U_n(x) = \int_0^x u^n(y)\,\mathrm{d}y.$$

So using (10), we have that

$$\frac{\mathrm{d}}{\mathrm{d}\epsilon}W_{c_\tau}\left(u_\epsilon, v^{n-1}\right)\bigg|_{\epsilon=0}$$
$$= \frac{\mathrm{d}}{\mathrm{d}\epsilon}\int_\Omega c_\tau\left(x - T_\epsilon(x)\right)u_\epsilon(x)\,\mathrm{d}x\bigg|_{\epsilon=0}$$
$$= \int_\Omega \left[-c_\tau'\left(x - T(x)\right)T'(x)G(x) + c_\tau\left(x - T(x)\right)g(x)\right]\mathrm{d}x.$$

Inserting the identity

$$-c_\tau'\left(x - T(x)\right)T'(x)G(x) + c_\tau\left(x - T(x)\right)g(x)$$
$$= \frac{\mathrm{d}}{\mathrm{d}x}\left[c_\tau\left(x - T(x)\right)G(x)\right] - c_\tau'\left(x - T(x)\right)G(x)$$

in the subsequent equality, we have

$$\frac{\mathrm{d}}{\mathrm{d}\epsilon}\bigg|_{\epsilon=0} W_{c_\tau}\left(u_\epsilon, v^{n-1}\right) = c_\tau\left(x - T(x)\right)G(x)\big|_0^1 - \int_\Omega c_\tau'\left(x - T(x)\right)G(x)\,\mathrm{d}x.$$

Since $G(0) = G(1) = 0$, the first term on the r.h.s. of the above equality vanishes. Using an integration by parts in the second term, and the fact that $G(0) = G(1) = 0$, we obtain that

$$\frac{\mathrm{d}}{\mathrm{d}\epsilon}\bigg|_{\epsilon=0} W_{c_\tau}\left(u_\epsilon, v^{n-1}\right) = \int_\Omega \left[\int_0^x c_\tau'\left(s - T(s)\right)\mathrm{d}s\right]g(x)\,\mathrm{d}x. \tag{15}$$

We combine (12)–(15) to have that

$$\begin{aligned}&\left.\frac{\mathrm{d}I(u_\epsilon)}{\mathrm{d}\epsilon}\right|_{\epsilon=0}\\&=\int_\Omega\left[\int_0^x c_\tau'\big(s-T(s)\big)\,\mathrm{d}s+\frac{1}{\tau}\big(F'\big(u^n(x)\big)+V(x)+W\star u^n(x)\big)\right]g(x)\,\mathrm{d}x\\&=0\end{aligned}\tag{16}$$

for all continuous functions $g:\Omega\to\mathbb{R}$ such that $\int_\Omega g(x)\,\mathrm{d}x=0$. Therefore

$$\tau\int_0^x\big[c_\tau'\big(s-T(s)\big)\big]\,\mathrm{d}s+F'\big(u^n(x)\big)+V(x)+W\star u^n(x)=\text{constant},$$

which gives after differentiation w.r.t. x,

$$\tau c_\tau'\big(x-T(x)\big)=-\frac{\mathrm{d}}{\mathrm{d}x}\big[F'\big(u^n(x)\big)+V(x)+W\star u^n(x)\big],$$

i.e.,

$$\frac{x-T(x)}{\tau}=c_*'\left(-\frac{\mathrm{d}}{\mathrm{d}x}\big[F'\big(u^n(x)\big)+V(x)+W\star u^n(x)\big]\right),\tag{17}$$

where we have used that $\tau c_\tau'(z)=c'(x/\tau)$ and $(c')^{-1}=c_*'$. Next, we use that c_*' is an odd function (since c is even) to rewrite (17) as

$$\frac{x-T(x)}{\tau}=-c_*'\left(\frac{\mathrm{d}}{\mathrm{d}x}\big[F'\big(u^n(x)\big)+V(x)+W\star u^n(x)\big]\right).\tag{18}$$

To obtain (11) from (18), we proceed as in [1]. First, we multiply (18) by $u_n(x)g(x)$ and integrate over $\Omega=(0,1)$. We have after using an integration by parts and the fact that $G(0)=G(1)=0$,

$$\begin{aligned}&\int_\Omega\frac{x-T(x)}{\tau}u^n(x)g(x)\,\mathrm{d}x\\&\quad=\int_\Omega\frac{\mathrm{d}}{\mathrm{d}x}\left[u^nc_*'\left(\frac{\mathrm{d}}{\mathrm{d}x}\big[F'\big(u^n\big)+V+W\star u^n\big]\right)\right]G(x)\,\mathrm{d}x.\end{aligned}\tag{19}$$

Next, we express $\int_\Omega\frac{x-T(x)}{\tau}u^n(x)g(x)\,\mathrm{d}x$ in terms of $\int_\Omega[\frac{u^n(x)-u^{n-1}(x)}{\tau}]G(x)\,\mathrm{d}x$. We first use $T_\#u^n=v^{n-1}$ to have that

$$\int_\Omega\frac{u^n(x)-v^{n-1}(x)}{\tau}G(x)\,\mathrm{d}x=-\int_\Omega\frac{G(T(x))-G(x)}{\tau}u^n(x)\,\mathrm{d}x.\tag{20}$$

Now we insert the Taylor expansion of G,

$$G\big(T(x)\big)-G(x)=g(x)\big(T(x)-x\big)+\text{Correction terms},$$

into (20) to get

$$\int_\Omega\frac{x-T(x)}{\tau}u^n(x)g(x)\,\mathrm{d}x=\int_\Omega\frac{u^n(x)-v^{n-1}(x)}{\tau}G(x)\,\mathrm{d}x+\text{Correction terms}.\tag{21}$$

Combining (19) and (21), we obtain in a weak sense,

$$\frac{u^n-v^{n-1}}{\tau}=\frac{\mathrm{d}}{\mathrm{d}x}\left[u^nc_*'\left(\frac{\mathrm{d}}{\mathrm{d}x}\big[F'\big(u^n\big)+V+W\star u^n\big]\right)\right.+\text{Correction terms}.\tag{22}$$

Finally we substitute $v^{n-1}(x)=u^{n-1}(x)+\int_{(n-1)\tau}^{n\tau}f(x,t)\,\mathrm{d}t$ into (22) to conclude (11). □

3 Numerical Algorithms

Following [11], we approximate solutions to PDE (6) using piecewise constant functions. The data u_0, V and W in (6) are taken to be piecewise constant functions whose values coincide with the actual values of these functions at the midpoint of each interval $[x_i, x_{i+1}]$, where $x_i = ih$, $h = 1/N$ denotes the mesh size, and N is a given positive integer. For example, a function $v : \Omega \to \mathbb{R}$ will be numerically represented by the sequence $(v_i)_i$, where $v_i := v(\frac{x_i+x_{i+1}}{2}) = v((i+1/2)h)$, $i = 0, \dots, N-1$, denotes the value of the approximation of v on the interval $[x_i, x_{i+1}]$. Similarly, the time interval $[0, \infty)$ will be discretized as $[t_n, t_{n+1}]$ where $t_n = n\tau$, $n = 0, 1, \dots$, and $\tau = 1/M > 0$ denotes the time-step size, and M is a given integer. So a function $g : [0, \infty) \to \mathbb{R}$ will be numerically represented by the sequence $(g^{n+1/2})_n$ where $g^{n+1/2} := g(\frac{t_n+t_{n+1}}{2}) = g((n+1/2)\tau)$ represents the value of the approximation of g on the interval $[t_n, t_{n+1}]$. Therefore, a function $h(x,t)$, $(x,t) \in \Omega \times [0,\infty)$, (and in particular the datum $f(x,t)$ in (6)), is approximated by the sequence $(h_i^{n+1/2})$ where $h_i^{n+1/2} := h((i+1/2)h, (n+1/2)\tau)$ represents the value of the approximation of h on the rectangle $[x_i, x_{i+1}] \times [t_n, t_{n+1}]$. Under these approximations, the integral constraint in the scheme (7) becomes:

$$\int_\Omega u(x)\,\mathrm{d}x \simeq h\sum_{i=1}^{N-1} u_i, \tag{23}$$

$$\int_\Omega v^n(x)\,\mathrm{d}x = \int_\Omega \left[u^n(x) + \int_{n\tau}^{(n+1)\tau} f(x,t)\,\mathrm{d}t\right]\mathrm{d}x \simeq h\sum_{i=0}^{N-1}\left(u_i^n + \tau f_i^{n+1/2}\right).$$

Therefore, given $u^n \simeq (u_i^n)_i$, we will look for the minimizer $u^{n+1} \simeq (u_i^{n+1})_i$ of (7) in the space

$$S = \left\{u \in L^1(\Omega) \mid u \geq 0, u|_{(x_i,x_{i+1})} = u_i, \sum_{i=0}^{N-1} u_i = \sum_{i=0}^{N-1} u_i^n + \tau f_i^{n+1/2}\right\}, \tag{24}$$

where a function $u \in S$ can be identified with the sequence $(u_i)_i \in \mathbb{R}^N$ defined as above. Similarly, the energy functional (13) is approximated as:

$$E(u) \simeq h\sum_{i=0}^{N-1}\big(F(u_i) + u_i V_i\big) + \frac{1}{2}h^2\sum_{i,j=1}^{N-1} u_i u_j W_{i,j} \tag{25}$$

where $\int_{x_i}^{x_{i+1}}\int_{y_j}^{y_{j+1}} W(x-y)\,\mathrm{d}x\,\mathrm{d}y \simeq h^2 W_{i,j} := h^2 W(\frac{x_i+x_{i+1}}{2} - \frac{y_j+y_{j+1}}{2})$ and $V_i := V(\frac{x_i+x_{i+1}}{2})$. Moreover, using formula (10), the transport cost,

$$W_{c_\tau}\big(u, v^n\big) = \int_0^{U(1)} c_\tau\big(U^{-1}(\eta) - V_n^{-1}(\eta)\big)\,\mathrm{d}\eta,$$

can also be computed in the same way, by approximating $U(x) = \int_0^x u(y)\,\mathrm{d}y$ and $V_n(x) = \int_0^x V_n(y)\,\mathrm{d}y$ as in (23). This allows to find a numerical approximation of the minimizing functional $I(u) = W_{c_\tau}(u, v^n) + \frac{1}{\tau}E(u)$ in (7) for any $u \in S$. As a consequence, given $u^n \simeq (u_i^n)_i$, we can approximate the minimizer $u^{n+1} \simeq (u^{n+1})_i$ of (7),

$$u^{n+1} := \mathrm{Argmin}\left\{I(u) = W_{c_\tau}\big(u, v^n\big) + \frac{1}{\tau}E(u) : u \in S\right\},$$

$$v^n(x) := u^n(x) + \int_{n\tau}^{(n+1)\tau} f(x,t)\,\mathrm{d}t,$$

by using either of the two algorithms proposed in [11], that we next recall here for completeness. Below, $\Pi : \mathbb{R}^N \to S$ denotes the orthogonal projection onto the subspace S of $\mathbb{R}^N$; more details on this projection will be given later.

The Relaxation Algorithm We search the minimizer $u^{n+1} \simeq (u_j^{n+1})_j$ among the functions $\Pi(u^\epsilon) \in S$, where $u^\epsilon \equiv (u_j^\epsilon)_j$ is of the form $u^\epsilon = v^n + \sum_{i=0}^{N-1} \epsilon_i e_i$, ϵ_i are some small real numbers, and e_i are the vectors of the standard orthonormal basis $\{e_0, e_1, \dots e_{N-1}\}$ of $\mathbb{R}^N$, i.e., $e_i = (z_0, z_1, \dots z_{N-1})$ with $z_i = 1$ and $z_j = 0$ for $j \neq i$; here $v^n \simeq (v_j^n)_j$. More precisely, the relaxation algorithm is as follows:

1. Let $\epsilon = 1$ and set $u = (v_j^n)_j$ where $v_j^n = u_j^n + \tau f_j^{n+1/2}$.
2. While $\epsilon > 10^{-6}$, do:
   (a) For $i = 0$ to $N-1$, let $z_i = 1$ and $z_j = 0$ for $j \neq i$, and set $e = (z_0, z_1, \dots z_{N-1})$.
   (b) Compute $v = \Pi(u + \epsilon e)$. If $I(v) < I(u)$ assign $u = v$.
   (c) Compute $v = \Pi(u - \epsilon e)$. If $I(v) < I(u)$ assign $u = v$.
   (d) Repeat steps (a)–(c) until u does not change, and then assign $\epsilon = \epsilon/2$.
3. Assign $u^{n+1} = u$.

The Projected Gradient Algorithm Contrarily to the relaxation algorithm where the search of the minimizer is performed among the functions $\Pi(v^n + \sum_{i=0}^{N-1} \epsilon_i e_i)$, here we directly follow the direction of $-\nabla I(u)$ where the functional $I(u)$ decreases most rapidly. Therefore, we look for the minimizer in the smaller class of functions of the form $u - \epsilon \nabla I(u)$, where $\epsilon > 0$ is small. While this method is in general faster than the previous one, it could sometimes take many iterates to converge (or may not converge at all!) when a component of $u \simeq (u_i)_i$ is very close to zero, in which case the direction of $-\nabla I(u)$ might not be well defined numerically (this is the case in the example of Sect. 4.3). More precisely, the gradient algorithm is as follows:

1. Let $\epsilon = 1$, and $u = (v_j^n)_j$ where $v_j^n = u_j^n + \tau f_j^{n+1/2}$.
2. While $\epsilon < 10^{-16}$, do:
   (a) Compute $v = \Pi(u - \epsilon \nabla I(u))$.
   (b) If $I(v) < I(u)$, assign $u = v$, otherwise assign $\epsilon = \epsilon/2$.
   (c) Repeat steps (a)–(b) until convergence, or until $\epsilon < 10^{-16}$.
3. Assign $u^{n+1} = u$.

Now we comment on the projection $\Pi : \mathbb{R}^N \to S$. If we denote $m := \sum_{i=0}^{N-1} u_i^n + \tau f_i^{n+1/2}$, then the set S can be identified with the affine subset $S = A + \mathcal{S}$ of $\mathbb{R}^N$, where $A = (0, \dots, 0, m)$, and $\mathcal{S}$ is the subspace of $\text{span}\{w_0, w_1, \dots, w_{N-2}\}$ whose vectors have non-negative components along the w_i, and $w_i = (z_0, \dots, z_{N-1})$ with $z_i = 1, z_{N-1} = -1$ and $z_j = 0$ for $j \notin \{1, N-1\}$. So if $M' = (z'_0, \dots, z'_{N-1}) \in S$ is the orthogonal projection of a point $M = (z_0, \dots, z_{N-1}) \in \mathbb{R}^N$, $M' = \Pi(M)$, then the coordinates of M' can be computed by solving the system of N equations with N unknowns

$$\sum_{i=0}^{N-1} z'_i = m, \qquad \big(M - M'\big) \cdot w_i = 0 \quad \text{for all } i = 0, \dots N-2,$$

by using matrix algebra arguments, and also by enforcing that they are all non-negative.

To evaluate $\nabla I(u)$ in the projected gradient algorithm, we proceed as follows. From (16), we have that the "directional derivative" of the functional I at u in the "direction" of g is

$$\delta I(u)g(x) = \int_\Omega \left[\int_0^x c'_\tau\big(s - T(s)\big)\,\mathrm{d}s + \frac{1}{\tau}\big(F'\big(u(x)\big) + V(x) + W \star u(x)\big) \right] g(x)\,\mathrm{d}x.$$

Approximating $u(x)$, $g(x)$, $V(x)$ and $W(x-y)$ by piecewise constant functions as above, we have that

$$\delta I(u)g(x) \simeq \sum_{i=0}^{N-1} \left\{ \int_{x_i}^{x_{i+1}} \int_0^x c'_\tau\big(s - T(s)\big)\,\mathrm{d}s\,\mathrm{d}x + \frac{h}{\tau}\left[F'(u_i) + V_i + h\sum_{j=0}^{N-1} u_j W_{i,j} \right] \right\} g_i. \quad (26)$$

But by viewing I as a function on $\mathbb{R}^N$ and then $u \simeq (u_i)_i \in \mathbb{R}^N$ and $g = (g_i)_i \in \mathbb{R}^N$, we also have

$$\delta I(u)g(x) := \lim_{\epsilon\to 0} \frac{I(u+\epsilon g) - I(u)}{\epsilon} \simeq \big\langle \nabla I(u), g\big\rangle_{\mathbb{R}^N} = \sum_{i=0}^{N-1} \big(\nabla I(u)\big)_i g_i. \quad (27)$$

Combining (26) and (27), we obtain

$$\big(\nabla I(u)\big)_i = \int_{x_i}^{x_{i+1}} \int_0^x c'_\tau\big(s - T(s)\big)\,\mathrm{d}s\,\mathrm{d}x + \frac{h}{\tau}\left[F'(u_i) + V_i + h\sum_{j=0}^{N-1} u_j W_{i,j} \right]. \quad (28)$$

But note that T can be evaluated in every interval $[x_i, x_{i+1}]$ according to our approximation. Therefore, $c'_\tau(s - T(s))$ is defined piecewise, and we have

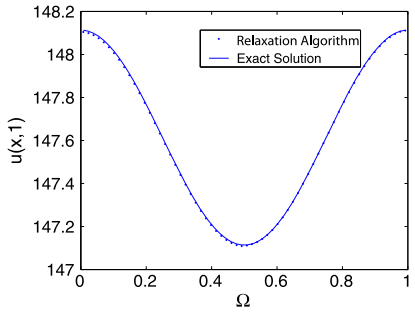
$$\begin{aligned}
&\int_{x_i}^{x_{i+1}} \int_0^x c'_\tau\big(s - T(s)\big)\,\mathrm{d}s\,\mathrm{d}x \\
&= \int_{x_i}^{x_{i+1}} \int_0^{x_i} c'_\tau\big(s - T(s)\big)\,\mathrm{d}s\,\mathrm{d}x + \int_{x_i}^{x_{i+1}} \int_{x_i}^{x} c'_\tau\big(s - T(s)\big)\,\mathrm{d}s\,\mathrm{d}x \\
&= h\sum_{j=0}^{i-1} \int_{x_j}^{x_{j+1}} c'_\tau\big(s - T(s)\big)\,\mathrm{d}s + \int_{x_i}^{x_{i+1}} \int_{x_i}^{x} c'_\tau\big(s - T(s)\big)\,\mathrm{d}s\,\mathrm{d}x.
\end{aligned}$$

We then deduce the formula:

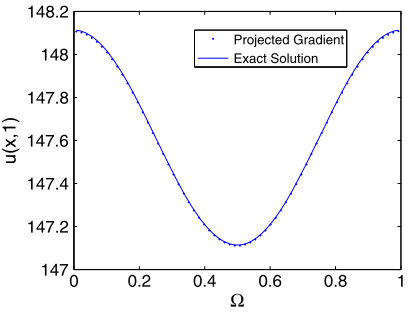
$$\begin{aligned}
\big(\nabla I(u)\big)_i = {} & h\sum_{j=0}^{i-1} \int_{x_j}^{x_{j+1}} c'_\tau\big(s - T(s)\big)\,\mathrm{d}s + \int_{x_i}^{x_{i+1}} \int_{x_i}^{x} c'_\tau\big(s - T(s)\big)\,\mathrm{d}s\,\mathrm{d}x \\
& + \frac{h}{\tau}\left[F'(u_i) + V_i + h\sum_{j=0}^{N-1} u_j W_{i,j} \right]. \quad (29)
\end{aligned}$$

4 Numerical Tests

We implement the two numerical algorithms using MATLAB. We use the projected gradient algorithm whenever possible, since it is the faster of the two algorithms.



(a) The solid line is the explicit solution, and the dotted line is our approximation using the relaxation method.



(b) The solid line is the explicit solution, and the dotted line is our approximation using the projected gradient method.

Fig. 1 Comparison between the exact solution of the non-homogeneous parabolic p-Laplacian equation (30) and its numerical approximations for $\tau = 1/128$ and $h = 1/64$ at time $t = 1$

Table 1 Errors for the solution to the non-homogeneous parabolic p-Laplacian equation (30) using the projected gradient method (PG)

τ	$h = 1/64$		
	Algorithm	$\|u(1) - \tilde{u}(1)\|_{\ell^1}$	$\|u - \tilde{u}\|_{\ell^1(\ell^1)}$
1/128	PG	0.004459	0.001997
1/256	PG	0.001118	0.000644
1/512	PG	0.000362	0.000457
h	$\tau = 1/128$		
	Algorithm	$\|u(1) - \tilde{u}(1)\|_{\ell^1}$	$\|u - \tilde{u}\|_{\ell^1(\ell^1)}$
1/16	PG	0.004459	0.004961
1/64	PG	0.004459	0.001997
1/128	PG	0.004459	0.001751

4.1 Non-homogeneous Parabolic p-Laplacian Equation

We test the numerical scheme with an explicit solution of the non-homogeneous parabolic p-Laplacian equation

$$\begin{cases} u_t - [(u_x)^3]_x = f(x,t) & x \in (0,1), t > 0 \\ u_x = 0 & x \in \{0,1\}, t > 0 \\ u(x,0) = 1 & x \in (0,1) \end{cases} \tag{30}$$

where $p = 4$ and $f(x,t) := \cos^2(\pi x) + 6t^3\pi^4(1 + \cos(2\pi x)\sin^2(2\pi x))$. Clearly, $u(x,t) = 1 + t\cos^2(\pi x) + \frac{3\pi^4}{2}t^4$ solves (30). Moreover, if we choose $c(x) = \frac{3}{4}|x|^{4/3}$ (i.e. $c_*(x) = x^4/4$), $F(x) = \frac{9}{10}x^{5/3}$ and $V = W = 0$ in (6), it easy to check that (30) is on the form (6). We first compare the actual solution of (30) at time $t = 1$ with the numerical approximations obtained by both the relaxation algorithm and the projected gradient algorithm. The results are presented in Fig. 1.

Next, we compare the errors between the actual solution u of (30) and the numerical solution $\tilde{u}$ computed by using the projected gradient method. The results are reported in

Table 2 Comparison between the equilibrium solution u_∞ and the numerical solution u_i at $t = 3$ for the rescaled porous medium equation (34), using $h = 1/8$ and $\tau = 1/64$

Solution	i							
	0	1	2	3	4	5	6	7
$u_i(t=3)$	1.082031	1.074219	1.058594	1.035156	1.003906	0.964844	0.917969	0.863281
$u_\infty((i+1/2)h)$	1.082357	1.074544	1.058919	1.035482	1.004232	0.965169	0.918294	0.863607

Table 1. From this table, it appears that the ℓ^1 and $\ell^1(\ell^1)$ errors due to time discretization are larger than that of spatial discretization, which suggests that the algorithm has greater sensitivity to changes in τ than changes in h.

The discrete $L^1(\Omega)$ errors are computed using the following formulae given in [11]:

$$\big\|u(t) - \tilde{u}(t)\big\|_{\ell^1} = h \sum_{i=0}^{N-1} \big|u\big((i+1/2)h, t\big) - \tilde{u}_i(t)\big| \tag{31}$$

where $\tilde{u}_i(t)$ is the ith component of the numerical solution at time t. Also,

$$\|u - \tilde{u}\|_{\ell^1(\ell^1)} = \tau \sum_{m=1}^{M} \big\|u(m\tau) - \tilde{u}(m\tau)\big\|_{\ell^1}, \tag{32}$$

where M denotes the number of time discretization points.

Remark 1 Comparing the projected gradient algorithm with the finite-difference method of Crank-Nicolson time stepping, using the homogeneous heat equation

$$u_t = u_{xx}, \qquad u(x,0) = 1 + 0.5\cos(\pi x), \qquad u_x|_{\partial\Omega} = 0, \tag{33}$$

we notice that the projected gradient method is more computationally costly and yields errors two to ten times larger than the Crank-Nicolson algorithm. However in some cases the finite difference scheme performs poorly (see the example in Sect. 4.3).

4.2 Rescaled Porous Medium and Doubly Nonlinear Equations

Example with quadratic cost function Consider the PDE,

$$u_t = \big(u^2\big)_{xx} + (xu)_x \tag{34}$$

which is the rescaled *porous medium equation* $w_T = (w^2)_{yy}$, where the relations between (x,t) and (y,T) are given as in [2], and $x, y \in \mathbb{R}$ and $t, T \in [0,\infty)$. If $x \in \Omega$, then (34) is of the form (6) with $F(x) = x^2$, $V(x) = c(x) = x^2/2$ and $f = W = 0$. The equilibrium solution $u_\infty(x)$, is the Barenblatt profile which solves

$$2\frac{\mathrm{d}u_\infty(x)}{\mathrm{d}x} + x = 0, \qquad \int_\Omega u_\infty(x)\,\mathrm{d}x = 1,$$

that is, $u_\infty(x) = 13/12 - x^2/4$. We evaluate this solution at the points $(x_i + x_{i+1})/2 = (i+1/2)h$ and compare the values with the numerical approximations u_i at time $t = 3$ obtained by the projected gradient method. The results are presented in Table 2. As shown in this table, our approximations are already close to the actual equilibrium solution u_∞ (where $u_\infty(x) = \lim_{t\to\infty} u(t,x)$) at time $t = 3$.

Table 3 Comparison between the equilibrium solution and the numerical solution of the rescaled doubly nonlinear equation at $t = 3$, for $h = 1/8$ and $\tau = 1/64$

Solution	i							
	0	1	2	3	4	5	6	7
$u_i(t = 3)$	1.02753	1.02682	1.02416	1.01825	1.00778	0.99145	0.96796	0.93600
$u_\infty((i + 1/2)h)$	1.02775	1.02704	1.02438	1.01847	1.00800	0.99167	0.96818	0.93623

Example with non-quadratic cost function Consider now the *doubly nonlinear equation* $w_T = [|(w^m)_y|^{p-2}(w^m)_y]_y$, where $y \in \mathbb{R}$ and $T \in [0, \infty)$. By rescaling (y, T) into (x, t) using the relations in [2], we may rewrite this PDE in the form

$$u_t = \left[uc'_*\left(F'(u)_x\right)\right]_x + (xu)_x \tag{35}$$

where $c(x) = |x|^q/q$ with $q = p/(p-1)$, and $F(x) = \frac{mx^\gamma}{\gamma(\gamma-1)}$ with $\gamma := m + \frac{p-2}{p-1}$. Note that (35) is of the form (6) where $V(x) = c(x)$ and $f = W = 0$. As shown in [2], the equilibrium solution of (35) solves

$$\left(F'(u) + c(x)\right)_x = 0, \qquad \int_\Omega u(x)\,\mathrm{d}x = 1. \tag{36}$$

If we set $c(x) = |x|^3/3$ (i.e., $q = 3$ or $p = 3/2$) and $F(x) = 3x^2/2$ (i.e., $m = 3$ and $\gamma = 2$), then the equilibrium solution is $u_\infty(x) = 37/36 - x^3/9$. As in the previous example, we compare this solution at positions $(x_i + x_{i+1})/2 = (i + 1/2)h$ with the numerical approximation at time $t = 3$ obtained by the projected gradient method. The results are recorded in Table 3. These results confirm that the conclusions drawn in the previous example extend to all cost functions of the form $c(x) = |x|^q/q$ for any $q > 1$.

4.3 Convection-Diffusion

Consider the convection-diffusion equation

$$\begin{cases} \frac{\partial u}{\partial t} = \frac{\partial}{\partial x}(uc'_*[(\nu \ln u + x)_x]) & x \in (0, 1); t > 0 \\ (\nu \ln u + V'u)_x = 0 & x = 0, 1, t > 0 \\ u(x, 0) = 1 & x \in (0, 1) \end{cases} \tag{37}$$

where $c(x) = |x|^3/3$ and $\nu = 1/100$. This equation is of the form (6) with $F(x) = \nu x \ln(x)$, $V(x) = x$ and $f = W = 0$. It is easy to check that the equilibrium solution of (37) is the probability density function solution of $(\nu \ln u + x)_x = 0$, i.e.,

$$u_\infty(x) = K \exp(-x/\nu), \quad 1/K = \nu\left(1 - \exp(-1/\nu)\right). \tag{38}$$

To test our numerical algorithm on this example, we use the relaxation method since the projected gradient method failed. However, the relaxation method is very slow, requiring almost nine hours of computation time (on a Windows PC with an Intel Core i5 CPU M 520 @2.40 GHz processor and 4 gb RAM) for the solution where $\tau = 1/128$ and $h = 1/16$. The results are recorded in Table 4. They show that as the time step size gets smaller, then our numerical results u_i (computed at time $t = 2$), get closer to the actual equilibrium solution u_∞ for small values of x where $u_\infty(x)$ is not too small (e.g., $x \in (0, 3/16)$); but when the values of $u_\infty(x)$ get very small (i.e., close to zero), then our numerical results become less accurate, but they still follow the same trend. This explains why the projected gradient algorithm failed here, as we already mentioned in Sect. 3.

Table 4 Comparison between average steady state solution over each interval with numerical solution at $t = 2$, using $h = 1/16$, and varying values of τ

τ	i	0	1	2	3	4	5	6	7	8	9–15
1/8	u_i	15.82105	0.102865	0.034603	0.018440	0.010578	0.005993	0.003285	0.001715	0.000853	$< 10^{-4}$
1/32	u_i	15.96494	0.032877	0.001688	0.000307	0.000130	3.29E-05	8.11E-06	2.38E-06	2.38E-06	$< 10^{-6}$
1/128	u_i	15.96833	0.031269	0.000200	8.11E-06	0.000114	3.10E-05	8.11E-06	8.11E-06	8.11E-06	$< 10^{-6}$
$\frac{1}{h}\int_{x_i}^{x_{i+1}} u_\infty(x)\,\mathrm{d}x$		15.96911	0.030827	0.000059	1.15E-07	2.21E-10	$< 10^{-10}$	$< 10^{-10}$	$< 10^{-10}$	$< 10^{-10}$	$< 10^{-10}$

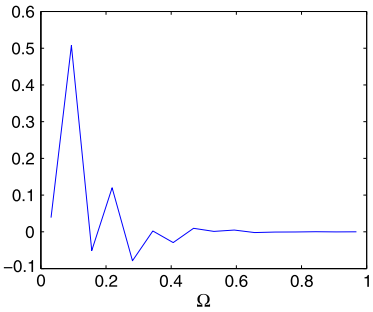


Fig. 2 Approximate solution at time $t = 2$ to the equilibrium solution of the convection diffusion equation (37), using Crank-Nicolson finite difference scheme

As pointed out in [11], when we use a Crank-Nicolson finite difference scheme to solve this convection-diffusion problem, oscillations occur (as shown on Fig. 2). Therefore, the relaxation algorithm, while slow, yields a better approximation to the solution of this problem.

5 Conclusion

We extend the numerical methods of [11] to all strictly convex symmetric cost functions, and in particular to cost functions of the form $|x|^q/q$ for $q > 1$. This allows us to numerically approximate, on the real line, solutions to a large class of parabolic partial differential equations which are gradient flows in the L^p-Wasserstein spaces for all $p > 1$. It was found that the two algorithms of this paper require significantly more computation time than traditional finite difference solvers for simple diffusion PDE's, however in certain cases they give a better approximation than finite differences. In order for these variational methods to be numerically practical, new numerical algorithms need to be developed to improve their speed.

Another interesting issue that we plan to investigate in the near future is to extend these numerical results to higher dimensions. Related to this problem, the only available literature so far seems to be the work by Carrillo and Moll [8] where an alternative numerical algorithm is proposed for solving problem (3) in higher dimensions. In [8] the authors reformulated this equation using Lagrangian coordinates, which permits to avoid computing numerically the Wasserstein distance in higher dimensions. Our goal instead is to combine the present work (performed in one dimension) with a numerical resolution of the optimal transport problem in higher dimensions obtained by the first author et al. in [15] (perhaps at the discrete level), to produce a numerical algorithm for solving problem (3) in higher dimensions via the scheme (4).

Acknowledgements The authors are supported by grants from the Natural Science and Engineering Research Council of Canada (NSERC).

References

1. Agueh, M.: Existence of solutions to degenerate parabolic equations via the Monge-Kantorovich theory. Adv. Differ. Equ. **10**(3), 309–360 (2005)
2. Agueh, M.: Rates of decay to equilibria for p-Laplacian type equations. Nonlinear Anal. **68**, 1909–1927 (2008)

3. Agueh, M.: Finsler structure in the p-Wasserstein space and applications to PDEs. C. R. Math. Acad. Sci. Paris, Ser. I **350**, 35–40 (2012)
4. Ambrosio, L., Gigli, N., Savaré, G.: Gradient Flows in Metric Spaces and in the Space of Probability Measures. Lectures in Mathematics. Birkhäuser, Basel (2005)
5. Brenier, Y.: Polar factorization and monotone rearrangements of vector valued functions. Commun. Pure Appl. Math. **44**, 375–417 (1991)
6. Caffarelli, L.: Allocation Maps with General Cost Functions. Partial Differential Equations and Applications, pp. 29–35. Dekker, New York (1996)
7. Carrillo, J.A., McCann, R.J., Villani, C.: Contractions in the 2-Wasserstein length space and thermalization of granular media. Arch. Ration. Mech. Anal. **179**, 217–263 (2006)
8. Carrillo, J.A., Moll, J.S.: Numerical simulation of diffusive and aggregation phenomena in nonlinear continuity equations by evolving diffeomorphisms. SIAM J. Sci. Comput. **31**(6), 4305–4329 (2009)
9. Gangbo, W., McCann, R.J.: Optimal maps in Monge's mass transport problem. C. R. Acad. Sci. Paris Sér. I. Math. **321**, 1653–1658 (1995)
10. Jordan, R., Kinderlehrer, D., Otto, F.: The variational formulation of the Fokker-Planck equation. SIAM J. Math. Anal. **29**(1), 1–17 (1998)
11. Kinderlehrer, D., Walkington, N.J.: Approximation of parabolic equations using the Wasserstein metric. Math. Model. Numer. Anal. **33**(4), 837–852 (1999)
12. Otto, F.: Doubly degenerate diffusion equations as steepest descent. Manuscript (1996)
13. Otto, F.: The geometry of dissipative evolution equations: the porous medium equation. Commun. Partial Differ. Equ. **26**, 101–174 (2001)
14. Petrelli, L., Tudorascu, A.: Variational principle for general diffusion problems. Appl. Math. Optim. **50**, 229–257 (2004)
15. Saumier, L.P., Agueh, M., Khouider, B.: An efficient numerical algorithm for the L^2 optimal transport problem with applications to image processing. Preprint (2011)
16. Tudorascu, A.: On the Jordan-Kinderlehrer-Otto variational scheme and constrained optimization in the Wasserstein metric. Calc. Var. Partial Differ. Equ. **32**(2), 155–173 (2008)
17. Villani, C.: Topics in Optimal Transportation. Graduate Studies in Math., vol. 58. AMS, Providence (2003)