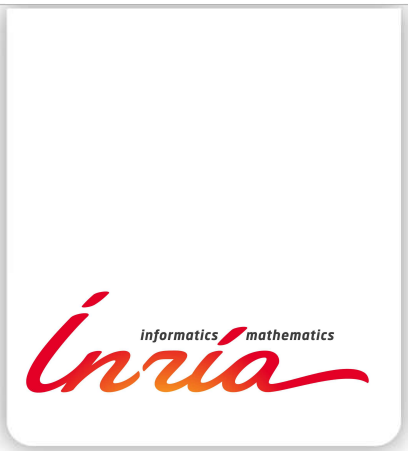
Numerical solution of the Monge-Kantorovich problem by Picard iterations

Afaf Bouharguane, Angelo Iollo , Lisl Weynans

RESEARCH REPORT
N° 8477
February 2014
Project-Team MC2

ISSN 0249-6399 ISRN INRIA/RR--8477--FR+ENG



Numerical solution of the Monge-Kantorovich problem by Picard iterations

Afaf Bouharguane*, Angelo Iollo *, Lisl Weynans *

Project-Team MC2

Research Report n° 8477 — February 2014 — 18 pages

Abstract: We present an iterative method to numerically solve the L^2 Monge-Kantorovich problem. The method is based on a Picard fixed point iteration of the linearized problem. Examples relative to the transport of two-dimensional densities show that the present method can significantly reduce the computational time over existing methods, especially when the Wasserstein distance between the densities is small.

Key-words: Optimal transport, Monge–Kantorovich problem, Numerical methods, Picard iterations

* Univ. Bordeaux, IMB, UMR 5251, F-33400 Talence, France.

RESEARCH CENTRE
BORDEAUX – SUD-OUEST

200 avenue de la Vieille Tour
33405 Talence Cedex

Résolution numérique du problème de Monge-Kantorovich par des itérations de Picard

Résumé : Nous présentons une méthode itérative pour résoudre numériquement le problème L^2 de Monge-Kantorovich. La méthode est basée sur des itérations de Picard du problème linéarisé. Des exemples relatifs aux transports de densités bidimensionnelles montrent que cette méthode réduit considérablement le temps de calcul par rapport aux méthodes existantes, en particulier lorsque la distance de Wasserstein entre les densités est petite.

Mots-clés : Transport Optimal, Problème de Monge–Kantorovich, Méthodes numériques, Itérations de Picard

1 Introduction

Optimal transportation is increasingly used to model problems in mechanics, physics, image analysis and other fields, see e.g. [10] and references therein. Because of all these applications, this old topic first introduced by Monge in 1781 [7], has attracted a lot of attention these last years especially from a numerical point of view [2], [1], [5], [6], [3], [8]. Indeed, compared to the theoretical results already obtained, the discrete solution of this problem still poses challenging problems in terms of computational burden and accuracy.

In this work we focus on the numerical solution of the L^2 Monge-Kantorovich problem (MKP) defined as follows. Let $\rho_0(\xi), \rho_1(x)$ be two non-negative scalar density functions with compact support Ω_0 and Ω_1, where $\xi, x \in \mathbb{R}^d$ and d is the space dimension. We assume that

$$\int_{\Omega_0} \rho_0(\xi) d\xi = \int_{\Omega_1} \rho_1(x) dx.$$

Let $X : \Omega_0 \to \Omega_1$ a smooth one-to-one map such that $X(\xi)$ realizes the transfer of ρ_0 onto ρ_1, i.e., a map that satisfies the following Jacobian equation:

$$\rho_0(\xi) = \det(\nabla X(\xi))\rho_1(X(\xi)).$$

This equation is underdetermined with respect to $X(\xi)$ and a solution is selected among all possible maps by introducing the following L^2 Kantorovich-Wasserstein distance:

$$\inf_X \int_{\Omega_0} \rho_0(\xi)|X(\xi) - \xi|^2 d\xi.$$

The L^2 MKP corresponds to finding a map X^* such that this infimum is achieved. It has been proved that this problem admits a unique solution [4, 9, 10], which is the gradient of a.e. convex function $\Psi : \Omega_0 \to \mathbb{R}$:

$$X^*(\xi) = \nabla\Psi(\xi).$$

In the scientific computing literature, there exist two main approaches to approximate this problem. The first one is based on a direct solution of the Monge-Ampere equation (MAE):

$$\rho_0(\xi) = \det(\nabla^2\Psi(x))\rho_1(\nabla\Psi(x)).$$

The difficulty in approaching the problem by this way is that the boundary conditions of this equation are not known a priori. Instead, the solution must verify the constraint $X(\Omega_0) = \Omega_1$. For the solution of the MAE with Dirichlet b.c. a recent numerical study in two dimensions is discussed in [5], where the solution is obtained via a least-square formulation. The full MKP solution via a MAE was considered in [3]. The numerical method employed is based on the solution of MAE with boundary conditions that are iteratively updated to converge to the MKP solution.

Another class of methods relies on ideas from continuum mechanics. Benamou and Brenier (BB) numerically solved the MKP by using an augmented Lagrangian method [2]. In their formulation a temporal dimension is introduced so that, given Π: $[0,1]\times\Omega_0 \to \mathbb{R}^d$, with $\Pi(0,\xi) = \xi$, $\Pi(1,\xi) = X(\xi)$, $x = \Pi(t,\xi)$ and $\partial_t\Pi = v(t,x)$, the MKP amounts to the solution of

$$\inf_{\rho,v} \int_{\mathbb{R}^d} \rho(t,x)|v(t,x)|^2 \, dx,$$

where the infimum is taken among all densities $\rho(t,x) \geq 0$ and velocity fields $v(t,x) \in \mathbb{R}^d$ satisfying the continuity equation

$$\partial_t\rho + \nabla\cdot(\rho v) = 0,$$

and the initial and final conditions:

$$\rho(0,\cdot) = \rho_0,\ \rho_1(1,\cdot) = \rho_1.$$

The BB method results in a robust and viable discrete minimization problem under constraints which admits a unique solution. However, since it is a gradient method in space-time, the computational cost may be relevant. Also, numerical diffusion of the transported densities is observed in the simulation of the transport for intermediate times. A recent improvement of the minimization method at the base of the BB algorithm is proposed in [8]. Using proximal splitting schemes the authors were able to solve difficult transport problems in presence of geometric constraints.

A yet different approach is devised in [1]. The idea is to first consider a mass preserving mapping, not necessary optimal, between the initial and final distributions and then to solve a PDE up to steady state in order to rearrange the non-optimal mapping into an optimal one. It is shown that also this approach leads to a gradient-based minimization problem for which many gradient steps are usually needed to converge.

Here, we investigate two computationally efficient algorithms, easily implemented, to solve the optimal mass transfer problem. The paper is organised as follows. In the next section we explain how to get a relevant initialisation of the mapping and in section 3, we present the main idea of the numerical methods. Sections 4 and 5 are respectively devoted to the implementation of the Lagrangian and Eulerian methods and to numerical tests. Comparisons with the BB algorithm are presented in section 6, where a final discussion is provided.

2 The initial mapping

The initial mapping is found by a modification of the approach proposed in [1]. As a first step, we compute a mapping that transports ρ_0 onto ρ_1. This mapping will not necessarily be the gradient of a convex potential. Without loss of generality we assume that $\Omega = [-\Lambda, \Lambda] \times [-\Lambda, \Lambda]$, $\Lambda \in \mathbb{R}$. We assume also that the initial and final densities have compact support included in Ω. Let $a : [-\Lambda, \Lambda] \to \mathbb{R}$ be the solution of

$$a'(\xi_1) \int_{-\Lambda}^{\Lambda} \rho_1(a(\xi_1), x_2)\, dx_2 = \int_{-\Lambda}^{\Lambda} \rho_0(\xi_1, \xi_2)\, d\xi_2 \tag{1}$$

with $x = X(\xi)$, $\xi = (\xi_1, \xi_2)$, $x = (x_1, x_2)$ and $a(-\Lambda) = -\Lambda$. Also let $b : \Omega \to \mathbb{R}$ be solution of the parametric ordinary differential equation

$$a'(\xi_1) \frac{\partial b(\xi_1, \xi_2)}{\partial \xi_2} \rho_1(a(\xi_1), b(\xi_1, \xi_2)) = \rho_0(\xi_1, \xi_2) \tag{2}$$

with $b(\xi_1, -\Lambda) = -\Lambda$, $\forall \xi_1 \in [-\Lambda, \Lambda]$. The mapping $X_\omega = (a(\xi_1), b(\xi_1, \xi_2))$ takes ρ_0 onto ρ_1 by construction. Other choices of $a(\xi)$ respecting monotonicity, regularity and appropriate initial conditions are possible.

The mapping $X_\omega = a(\xi_1)\, b(\xi_1, \xi_2)$ is not in general the gradient of a complex potential. This requirement is a necessary condition for optimality. However, this mapping can always be decomposed as the sum of an irrotational field and a solenoidal field:

$$X_\omega = \nabla_\xi \Psi_i + \nabla \times A. \tag{3}$$

Hence, the actual initial mapping $\nabla_\xi \Psi_i$ is found by solving $\nabla \times \nabla \times A = \nabla \times X_\omega$ with homogeneous Neumann boundary conditions, and computing $\nabla_\xi \Psi_i = X_\omega - \nabla \times A$. In two space dimensions, this amounts to the solution of one Laplace equation.

3 Picard iteration

The main idea of solution relies on the assumption that we dispose of an initial mapping that is a perturbation of the optimal mapping. Then, we derive a linear equation for the perturbation that is used to iteratively improve the initial guess.

3.1 Linearization

Let us assume that the optimal mapping is

$$X_o(\xi) = \nabla_\xi \Psi_o$$

and that the mapping obtained at iteration n is

$$X^n(\xi) = \nabla_\xi \Psi_i = \nabla_\xi \Psi_o + \nabla_\xi \Psi_\epsilon^n,$$

where the error Ψ_ϵ^n satisfies $\|\Psi_\epsilon^n\| \approx \epsilon$. We define $\rho_0^n(\xi)$ as the initial density that mapped by $X^n(\xi)$ gives the exact final density $\rho_1(x)$. Then, taking a first-order Taylor expansion, we have

$$\begin{aligned}\rho_0^n(\xi) &:= \rho_1(X^n(\xi)) \det\left(\nabla_\xi X^n(\xi)\right) \\ &= \underbrace{\rho_1(X_o) \det\left(\nabla_\xi X_o\right)}_{=\rho_0(\xi)} + \det\left(\nabla_\xi X_o\right) \rho_1(X_o) \mathrm{Tr}\left((\nabla_\xi X_o)^{-1} \nabla_\xi^2 \Psi_\epsilon^n\right) \\ &+ \det\left(\nabla_\xi X_o\right) \nabla_x \rho_1(X_o) \cdot \nabla_\xi \Psi_\epsilon^n + o(\epsilon),\end{aligned}$$

where Tr designs the matrix trace operator.
At first order in ϵ, we have

$$\frac{\rho_0^n(\xi) - \rho_0(\xi)}{\det\left(\nabla_\xi X_o\right)} = \rho_1(X_o) \mathrm{Tr}\big((\nabla_\xi X_o)^{-1} \nabla_\xi^2 \Psi_\epsilon^n\big) + \nabla_x \rho_1(X_o) \cdot \nabla_\xi \Psi_\epsilon^n$$

and finally, using the assumption that $\|\Psi_\epsilon^n\| \approx \epsilon$, we replace X_o by X^n so that

$$\frac{\rho_0^n(\xi) - \rho_0(\xi)}{\det\left(\nabla_\xi X^n\right)} = \rho_1(X^n) \mathrm{Tr}\big((\nabla_\xi X^n)^{-1} \nabla_\xi^2 \Psi_\epsilon^n\big) + \nabla_x \rho_1(X^n) \cdot \nabla_\xi \Psi_\epsilon^n. \tag{4}$$

The mapping update is then found by the iteration

$$X^{n+1} = X^n - \alpha \nabla \Psi_\epsilon^n,$$

that converges to X_o as a geometric series, for $\alpha \in [0, 1]$, in the linearized approximation.

In particular, if the optimal mapping is a perturbation of identity, i.e.,

$$X_o(\xi) = \xi + \nabla_\xi \Psi_o$$

with $\|\nabla_\xi \Psi_o\| = o(\epsilon)$, then at first order in ϵ equation (4) reduces to

$$\rho_0^n(\xi) - \rho_0(\xi) = \nabla_\xi \cdot (\rho_1(X^n(\xi)) \nabla_\xi \Psi_\epsilon^n) \tag{5}$$

that is equivalent to a semidiscretization in time of the continuity equation written in Eulerian form.

3.2 Regularization and continuation

The above equations are well defined only on the support of the initial and final densities. Therefore, in actual computations the initial and final densities are regularized by adding a small constant ζ so that the leading order differential operator is well defined on the entire domain:

$$\left(\rho_1\left(X^n\right)+\zeta\right)\mathrm{Tr}\left((\nabla_\xi X^n)^{-1}\nabla_\xi^2\Psi_\epsilon^n\right)$$

with $\zeta \approx 10^{-3}$ in practice.

The initial mapping determined as explained in section 2 may in general be far from the optimal mapping. This implies that the perturbation equation (4) is not a good approximation of the original problem. In those cases it possible to solve the problem by continuation adding to the initial and final densities a constant. This constant is then iteratively brought to $O(10^{-3})$. Taking $\zeta^k \in \mathbb{R}^+$ large enough, we define

$$\rho_0^k = \rho_0 + \zeta^k \text{ and } \rho_1^k = \rho_1 + \zeta^k$$

so that the condition $\|\Psi_\epsilon\| \approx \epsilon$ is verified. Once the optimal mapping Ψ_o^k for such a ζ^k is determined, we define

$$\zeta^{k+1} = (1-\beta)\zeta^k$$

with $\beta \in [0,1]$, and Ψ_o^k as the initial mapping for the next optimal transport problem. In practice, we choose $\beta = 0.1$.

In order to give a first estimate of ζ, it is natural to assume that it is proportional to the Wasserstein distance between the two densities. However, this distance in unknown a priori. Therefore, we approximate the Wasserstein distance using the initial mapping computed as explained in 2. We take

$$\zeta \propto \int \rho_0 |\nabla_\xi \Psi_i - \xi|^2 \, dx.$$

For example, for test case TC3 (see section 5) we took $\zeta = 1$ as the Wasserstein distance approximation above is of the same.

4 Numerical solution

We detail two solution approaches. A Lagrangian approach, that can readily be coded as it implies the solution of a rather simple elliptic PDE and an Eulerian approach, that can more easily be applied in more general geometries.

4.1 Lagrangian approach

According to previous sections, the following iterative method (Algorithm 1) is considered. In the limits of the first order perturbation analysis of this section, Algorithm 1 converges as a geometric series, i.e.,

$$\Psi_\epsilon^n \approx \Psi_o\left(1-(1-\alpha)^n\right). \tag{6}$$

Therefore, defining the normalized residual as $r = \|\Psi_\epsilon^n - \Psi_o\|/\|\Psi_o\|$, we have:

$$\log r = n \log(1-\alpha), \tag{7}$$

where $\log(1-\alpha)$ is the rate of convergence of the iterative scheme.

To approximate the problem, we discretize equation (4) by a standard second-order finite difference scheme on a Cartesian grid. At the discrete level, we impose homogeneous Dirichlet

Algorithm 1 Lagrangian iterative algorithm

1. $n = 0$;
2. compute $X^0 = \nabla\Psi_i$ as explained in section 2;
3. $\rho_0^n(\xi) = \rho_1(X^n(\xi)) \det \nabla_\xi X^n(\xi)$;
4. compute Ψ_ϵ^n by solving equation (4);
5. $X^{n+1} = X^n - \alpha \nabla\Psi_\epsilon^n$;
6. $n = n + 1$;
7. goto 3 if convergence is not attained;

boundary conditions to equation (4). In practice, since the initial and final densities have compact support, the boundary conditions imposed at the borders of the computational domain do not significantly affect the solution as $\zeta \to 0$. Mass transportation is performed by the continuity equation in Lagrangian coordinates.

In the next sub-section we will provide an alternative approach where the continuity equation is integrated in Eulerian coordinates. For numerical examples, see section 5, we will assume that the convergence is attained when a threshold criterion on $\|\rho_0^n - \rho_0\|_\infty$ is satisfied.

4.2 Eulerian approach

In the Eulerian framework, the solution of the L^2 MKP consists in determining the initial velocity $u(0,.) = u_0$ satisfying:

$$\begin{cases} \partial_t(\rho u) + \nabla \cdot (\rho u) = 0 & (8) \\ \partial_t(\rho u) + \nabla \cdot (\rho u \otimes u) = 0 & (9) \\ \rho(0,\cdot) = \rho_0, \quad \rho(1,\cdot) = \rho_1 & (10) \end{cases}$$

The optimal initial velocity u_0 is written under the form $u_0 = \nabla\Psi$, where the potential Ψ is the Lagrange multiplier of constraints (8) and (10). We propose an alternative solution method for u_0 by adapting Algorithm 1 to the Eulerian framework.

Starting from an initial guess for u_0, we numerically solve equations (8)-(9) using $\rho(0,.) = \rho_0$ as the initial condition for density. Solving system (8)-(9) as detailed hereafter, we get a final density $\tilde{\rho}_1$ and a final velocity field that we denote $\tilde{u}_1$. Since in general the initial velocity guess does not correspond to the optimal solution, the density $\tilde{\rho}_1$ is different from ρ_1. The next step is to find a potential velocity field transporting the data $\tilde{\rho}_1$ on ρ_1. This velocity field is considered as a correction to the initial mapping.

To find the additional velocity field, we approach the continuity equation by the following elliptic problem:

$$\nabla \cdot \left(\frac{\rho_1 + \tilde{\rho}_1}{2} \nabla\tilde{\Psi} \right) = \tilde{\rho}_1 - \rho_1.$$

This problem is solved using classical second-order centered finite-differences.
From the potential $\tilde{\Psi}$ we compute a velocity field $\tilde{u} = \nabla\tilde{\Psi}$, which can be seen as the velocity field necessary to advect $\tilde{\rho}_1$ to ρ_1. As we want to use it as a correction to u_0, we have to advect

it backward in time to make it match with the initial density ρ_0. We therefore perform an integration backward in time of the system (8)-(9) with final conditions $\tilde{\rho}_1$ and $\tilde{u}_1$, obtained by the forward numerical computation. The result of the backward advection of $\tilde{u}$, that we denote $\tilde{u_0^n}$, is added to u_0 as a corrective term. The full method is detailed in Algorithm 2.

The numerical resolution of equations (8)-(9) is performed using a remeshed particle method, similar to the one developed in [11] for compressible Euler equations. In this class of methods, the fluid is discretized on small masses concentrated on points. These points, the particles, are displaced in a Lagrangian way. New particles uniformly distributed are created at regular time intervals by an interpolation of the values of the existing particles, what is usually called remeshing the particles. The remeshing step creates new particles in a conservative way, by distributing the quantities carried by the particles at the nodes of an underlying grid.

Because the particles themselves are moved in a Lagrangian way, remeshed particle methods are submitted to less restrictive stability conditions, in the context of advection problems, than more classical grid-based methods such as finite-differences or finite-volume methods. Indeed their stability condition is typically proportional to the inverse of the velocity gradient (meaning that the particles trajectories do not cross), instead of a classical CFL condition. In the context of the method that we present here for optimal mass transport, we need to compare the numerical result computed at the final time to the final exact density ρ_1. Therefore, the use of larger time steps is an advantage because it means that the final state is computed with less numerical dissipation due to the temporal integration than in the case of more classical grid-based methods.

Algorithm 2 Eulerian iterative algorithm

1. $n = 0$;
2. initialize u_0;
3. for initial data ρ_0 and $u_0 = X^n$, compute ρ_1^n and u_1^n by numerically solving system (8)-(9);
4. compute $\tilde{\Psi}$ by solving equation:
$$\nabla \cdot \left(\frac{\rho_1 + \tilde{\rho}_1}{2} \nabla \tilde{\Psi} \right) = \tilde{\rho}_1 - \rho_1$$
5. integrate backward in time system (8)-(9) with final conditions $\tilde{\rho}_1 = \nabla\tilde{\Psi}$ and $\tilde{u}_1$ to get at initial time the velocity correction $\tilde{u}_0^n$;
6. $X^{n+1} = X^n + \alpha\, \tilde{u}_0^n$;
7. $n = n + 1$;
8. goto 3 if convergence is not attained;

5 Numerical illustrations

In this section, we provide several numerical validations of our methods. We perform the following test cases (TC):

- TC1. We consider two density distributions having a non-negligible support intersection.

We take two Gaussians of unit mass, same variance and displaced of 0.2 in the vertical direction. The final density has a cross correlation of 0.5, the initial is isotropic. We have:

$$\rho_0(\xi) = 3.97887 e^{-12.5\left(\xi_1^2+(\xi_2+0.1)^2\right)}$$

$$\rho_1(x) = 4.59441\, e^{-16.66666\left(x_1^2+x_1(x_2-0.1)+(x_2-0.1)^2\right)}.$$

- TC2. This is similar to TC1, but the centroid of the initial density is further displaced. We take the initial and final density distributions:

$$\rho_0(\xi) = 1.98944 \left(e^{-12.5\left((\xi_1-0.2)^2+(\xi_2+0.1)^2\right)} + e^{-12.5\left((\xi_1+0.2)^2+(\xi_2+0.1)^2\right)}\right)$$

$$\rho_1(x) = 3.97887\, e^{-12.5\left(x^2+y^2\right)}.$$

- TC3. In this case there is mass separation and virtually no intersection between the initial and final density support. The initial and final density are

$$\rho_0(\xi) = 1.98944 \left(e^{-12.5\left((\xi_1-0.5)^2+(\xi_2+0.3)^2\right)} + e^{-12.5\left((\xi_1+0.5)^2+(\xi_2+0.3)^2\right)}\right)$$

$$\rho_1(x) = 3.97887\, e^{-12.5\left(x^2+y^2\right)}.$$

- TC4. A final case showing the robustness of the Picard iteration is presented. The mapping between two scans of 168^2 pixels relative to the abdomen of a breathing patient, see figure 9, is determined.

Test case TC1, TC2, TC3 are performed using both the Lagrangian and the Eulerian schemes. We consider a square domain with $\Lambda = 2$ discretized using a uniform 200×200 grid. Compared to the Lagangian case, the corresponding Eulerian TCs are rotated of $\pi/4$ in order to show the accuracy of the particle remeshing in a transverse direction with respect to the grid.

For TC1, with $\alpha = 0.2$, the expected rate of convergence is of 0.09691, see (7). The fitting of the convergence curve with a straight line gives a convergence rate of 0.0958315 for the algorithm 1, see figure 1. The relative error in the max norm is of the order of 10^{-3} after 23 iterations. The Eulerian solution is given in figure 2.

In the next test case (TC2), the expected rate of convergence is of .0457575 ($\alpha = 0.1$), the fitting of the convergence curve with a straight line gives a convergence rate of 0.444000, see figure 3. The Eulerian solution is given in figure 4.

Finally, in TC3 the Lagrangian scheme was initialized with a regularization $\zeta = 1$ and in 40 steps it was reduced to 10^{-3}. The initial condition along with the optimal mapping stream curves are shown in figure 5. The intermediate and final solution and error are shown in figures 6 and 7. The Eulerian solution is given in figure 8.

The last case presented, TC4, is relative to front abdomen sections of a breathing patient, figure 9 (courtesy of B.D. de Senneville). The figure shows the initial velocity field relative to the optimal transport between the two subsequent scans. In certain focalized beam therapies, like focalized ultrasounds, it is crucial to accurately predict the movement of a patient in order to calibrate the displacement of the targeted region. In applications, real-time optical flow techniques based on heuristic arguments are employed. The methods presented here are not real time, although the solution of the MKP took a few seconds in this case. However, the solution of the MKP to determine the displacement between two images offers the advantage over optical flow techniques of being objective in the sense that it is based on a clearly identified model.

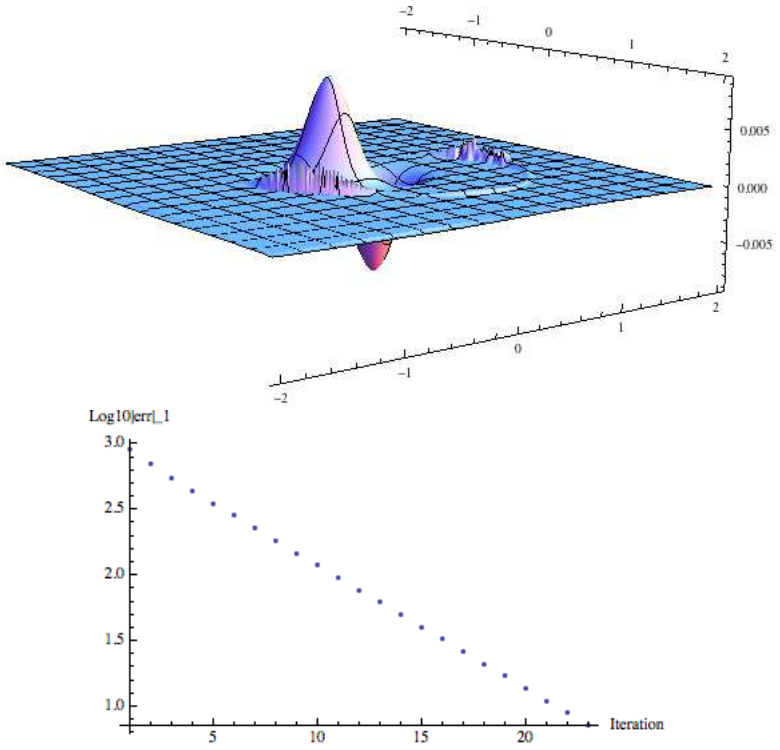


Figure 1: TC1. Error after 23 iterations and convergence for algorithm 1

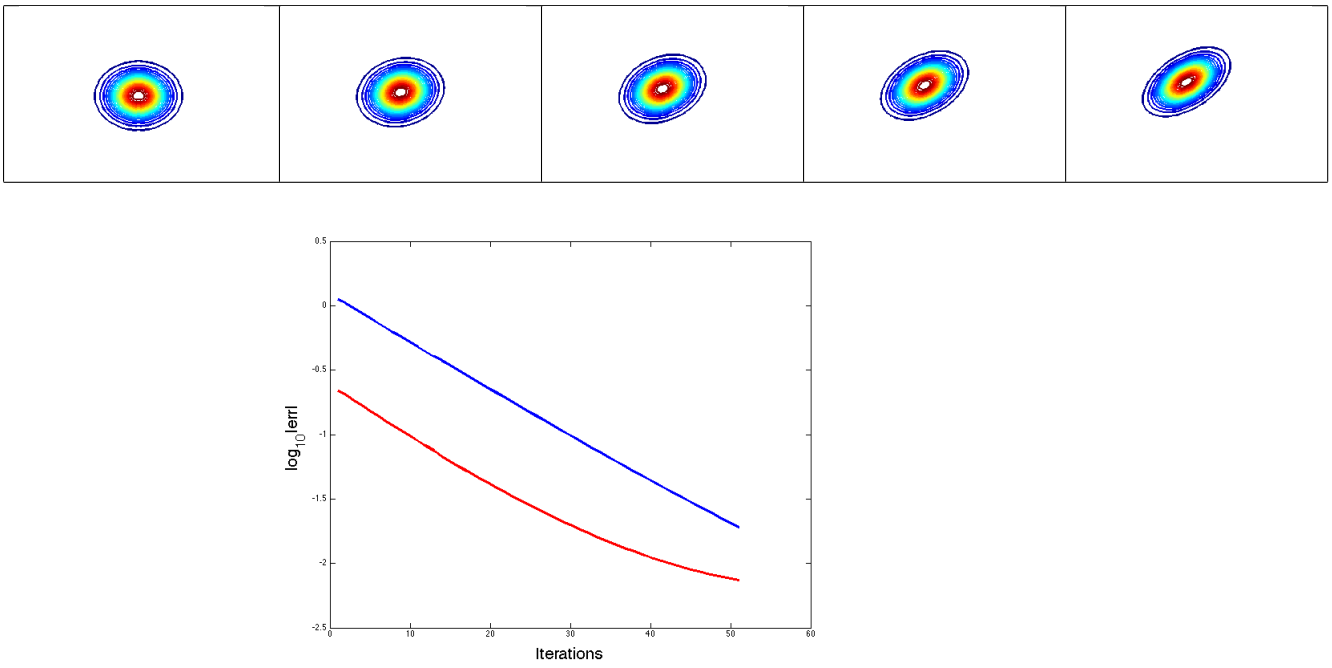


Figure 2: TC1. Top picture row: plot of density isolines $\rho(t,x)$ for $t = 0, 0.25, 0.5, 0.75, 1$ along with the optimal path computed by algorithm 2 . Bottom: red (resp. blue) line represents the error in L^1 norm (resp. L^∞ norm).

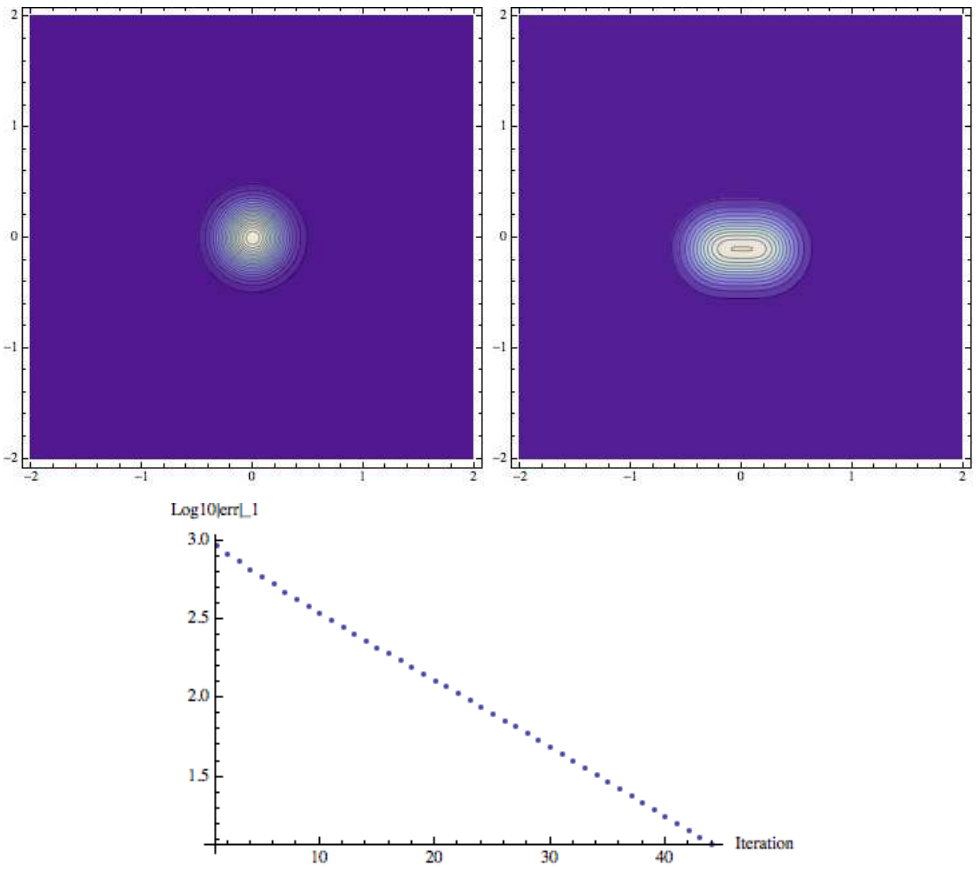


Figure 3: TC2: initial distribution, final distribution and convergence. Isolines from 0 to the max spaced of 0.2.

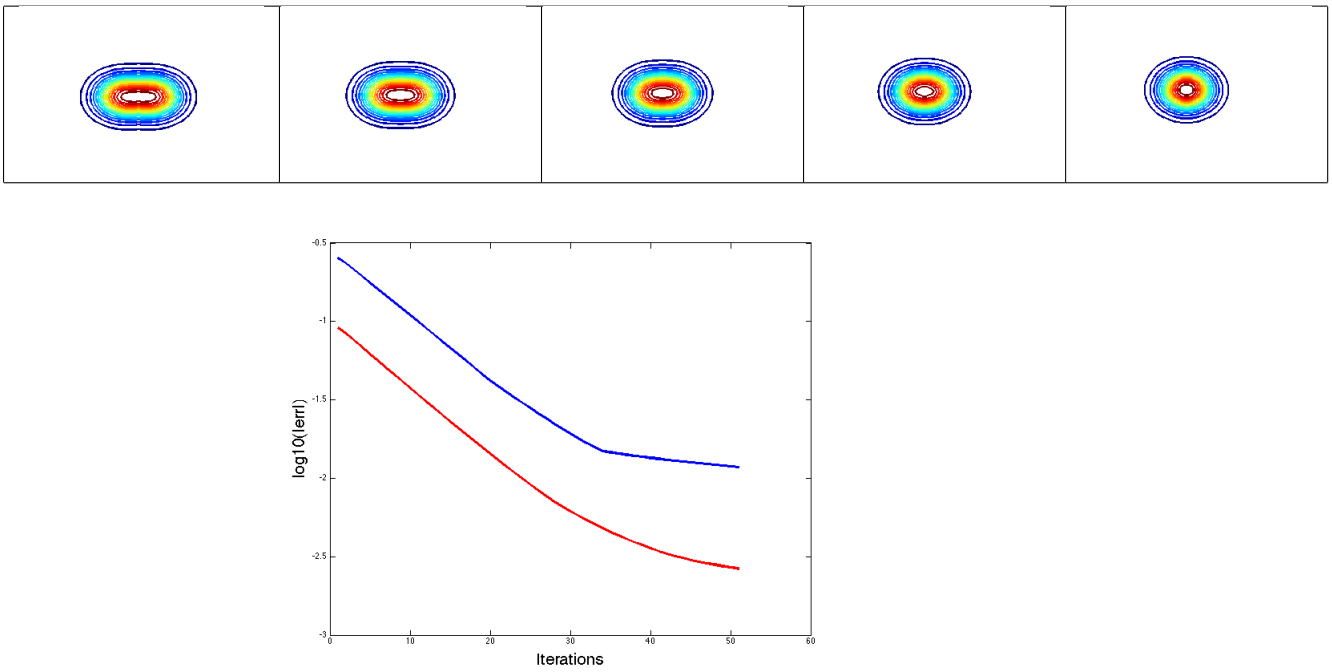


Figure 4: TC2. Top picture row: plot of the isolines of the density $\rho(t,x)$ for $t = 0, 0.25, 0.5, 0.75, 1$. Bottom: convergence of algorithm 2. Red (resp. blue) data represent the $\log_{10}(||\rho_1^n - \rho_1||_1)$ (resp. $\log_{10}(||\rho_1^n - \rho_1||_\infty)$) as a function of the number of iterations

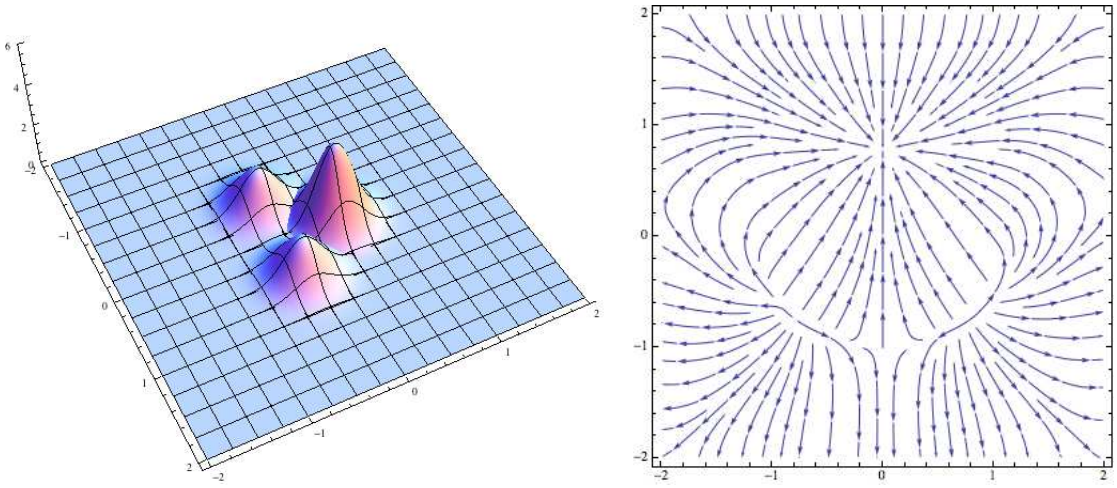


Figure 5: TC3. Initial and final densities and stream curves corresponding to $\nabla_\xi \Psi_o$, the optimal mapping.

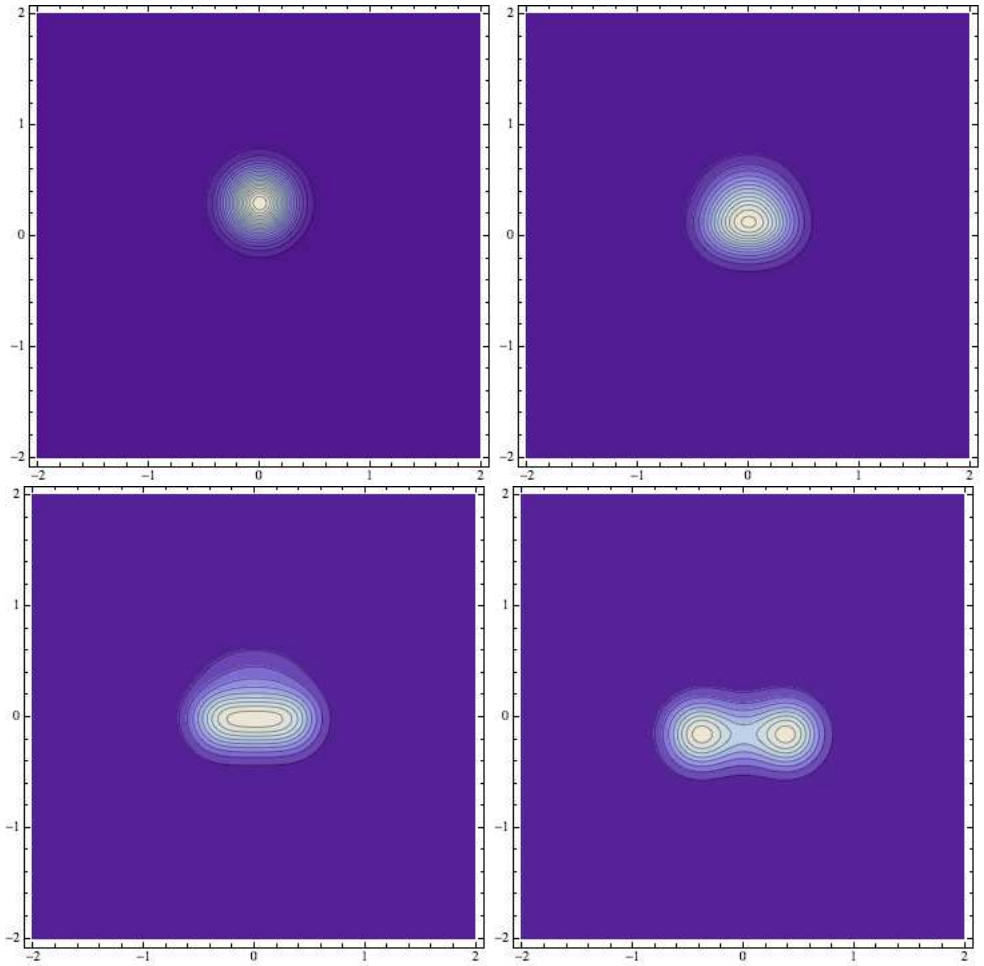


Figure 6: TC3. Intermediate densities for $X_\alpha = \xi + \chi\left(\nabla_\xi \Psi_o - \xi\right)$ with $\chi = \left\{0, \frac{1}{4}, \frac{1}{2}, \frac{3}{4}\right\}$ from left to right, top to bottom. Isolines from 0 to the max spaced of 0.2.

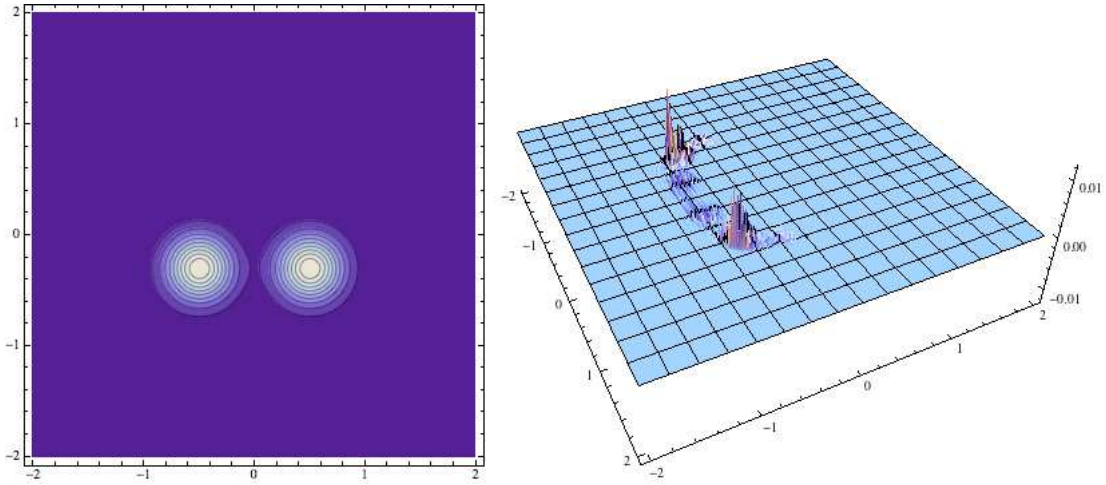


Figure 7: TC3. Final density ($\alpha = 1$) and error.

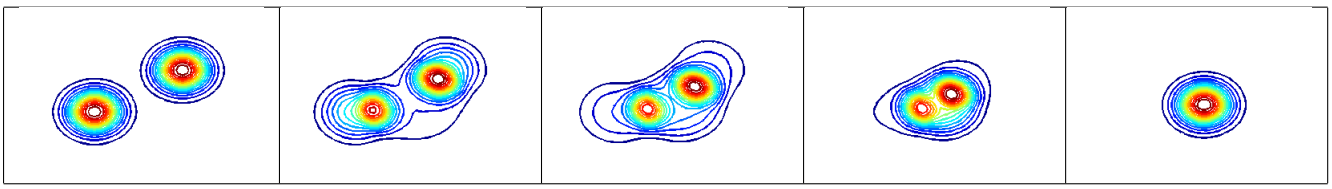


Figure 8: TC3. Plot of the isolines of the density $\rho(t, x)$ along the optimal path computed with algorithm 2

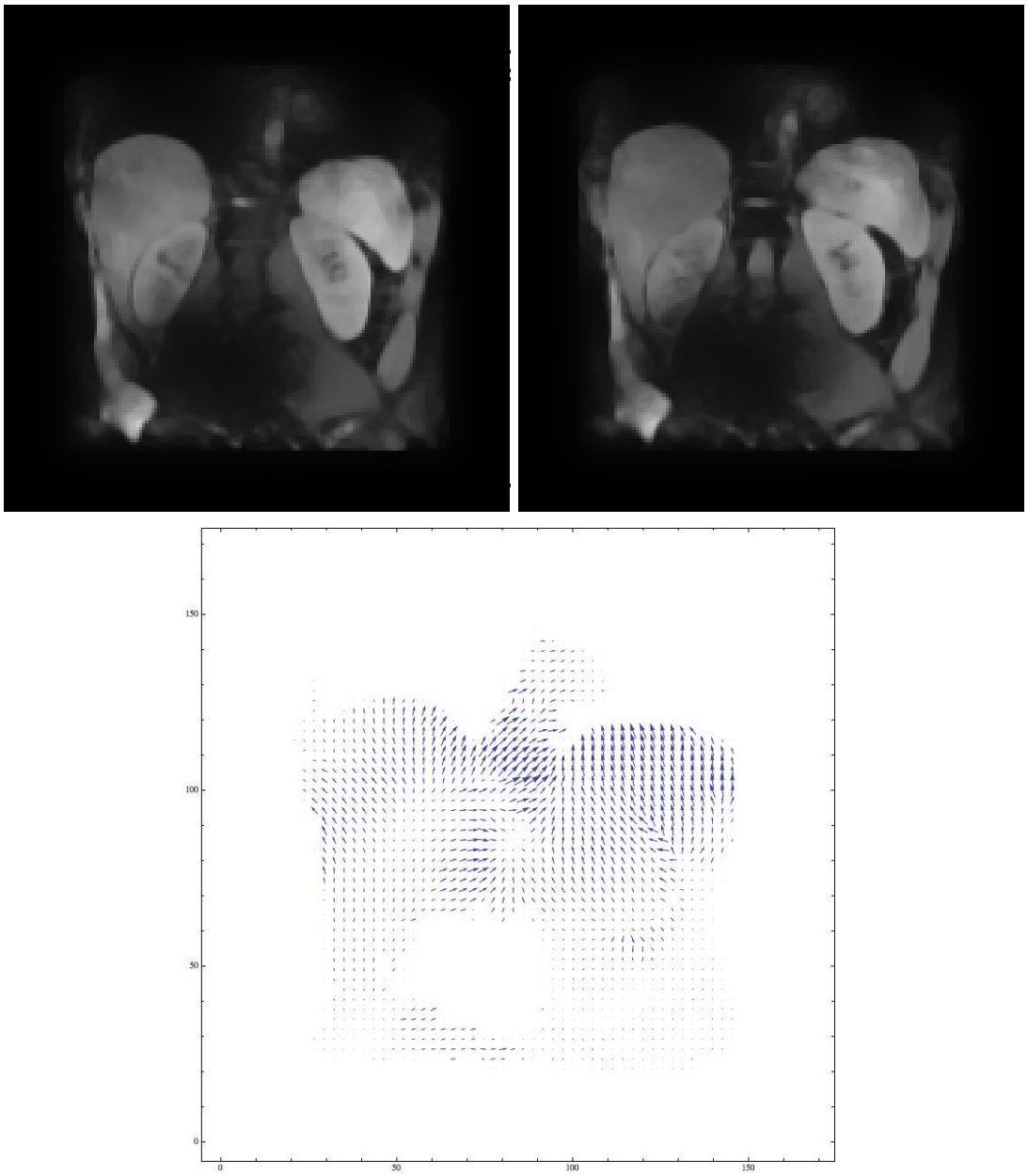


Figure 9: TC4. Top row: initial and final grey-scale densities. Bottom: optimal mapping.

6 Discussion and conclusions

The Picard iteration approach presented in this paper is suitable for transports where the distance travelled by the elementary masses is small compared to the characteristic length of the density distributions. The reason is that the unknowns of this approach are the initial transport velocities. If the distance travelled becomes large, small deviations of the initial condition lead to large errors. The BB approach has the advantage of solving for a space-time saddle point. Hence the trajectories are not identically straight line, as in the case of the present method, and the problem remains well conditioned also for large mass displacements as shown in [8]. This is of course at the price of a costly computational problem in $d+1$ dimensions.

In order to show the computational advantage of the Picard iteration for suitable cases, we compare the CPU time to solve the MKP by the BB method, to the CPU time of algorithms presented here. Since the convergence criteria are different for the two methods, we will determine the CPU time so that the initial criteria will be divided by 30, 60 and 100.

We recall that the convergence criterion of the BB method is based on the following residual of the Hamilton-Jacobi equation (see [2]):

$$res^n = \partial_t \phi^n + \frac{|\nabla \phi^n|}{2},$$

where ϕ is the Lagrange multiplier of constraints (8) and (10). The convergence criterion is given by:

$$crit_b^n = \sqrt{\frac{\int_0^1 \int_\Omega \rho^n |res^n|}{\int_0^1 \int_\Omega \rho^n |\nabla \phi^n|^2}}.$$

For the Picard iteration method presented here, we consider the error between the given final distribution ρ_1 and the final distribution ρ_1^n at iteration n:

$$crit^n = ||\rho_1 - \rho_1^n||_\infty$$

in the Eulerian case. For the Lagrangian case we consider the equivalent error based on the initial density. We consider TC1. For the BB algorithm, we discretize the time domain using 32 nodes and space with 200×200 grid points. We take optimal numerical parameters in the Uzawa iteration in order to converge as quickly as possible.

	BB method		**Eulerian method**		**Lagrangian method**	
	$crit_b^n$	CPU time	$crit^n$	CPU time	$crit^n$	CPU time
Initial Criteria (IC)	0.707107	0m0s	2.063231	0m0s	2.063231	0m0s
$\approx IC/30$	0.023547	2m36s	0.06004	0m4.13s	0.060026	0m2.6s
$\approx IC/60$	0.011784	5m12s	0.030851	0m5.32s	0.034312	0m3.2s
$IC/60 < IC \leq IC/100$	0.009841	19m31s	0.018421	0m6.22s	0.027887	0m3.74s

Table 1: CPU time: BB method vs. Our methods

Table 1 shows the evolution of the CPU time for different convergence thresholds. We can see that CPU time is very small for the present methods compared to the BB method. Note also that it is impossible to reduce the initial criteria by 100 in a reasonable time for BB. This is the reason we consider a convergence $IC/60 < IC \leq IC/100$ in the table above. Usually, in order

to get to convergence of the BB algorithm ($IC/100$), we have about 3000 Uzawa iterations with a computing time larger than three hours.

Cases where the Wasserstein distance between the densities is small are particularly favorable for the Picard fixed point solution because there is no need to regularize the solution by adding an initial constant ζ to both the initial and final distributions. In cases where the Wasserstein distance is larger, like TC3, the computational advantage of the Picard iteration with respect to the BB scheme is reduced (initially, $\zeta = 1$). For example, in TC3 the solution is obtained with the Lagrangian scheme in about 5 min CPU time, where the BB scheme takes 20 min to get to $IC/60 < IC \leq IC/100$. Still , the Picard iteration is more advantageous.

In conclusion, the Picard iteration that we presented can lead to a significant improvement in terms of convergence rate over the BB scheme, in particular when the Wasserstein distance is small. This can be a significant advantage for large three-dimensional problems in imagery and computational mechanics.

References

[1] S. Angenent; S. Haker; A. Tennenbaum, *Minimizing flows for the Monge-Kantorovich problem*, SIAM J. Math. Anal. (35), pp. 61–97 (2003).

[2] J.-D. Benamou; Y. Brennier, *A computational fluid mechanics solution to the Monge–Kantorovich mass transfer problem* , Numerische Matematik (84), pp. 375–393 (2000).

[3] J-.D. Benamou ; A. Oberman ; F. Britanny, *Numerical solution of the second boundary value problem for the Elliptic Monge-Ampère equation*, Rapport de recherche: `http://hal.inria.fr/hal-00703677` (2012).

[4] Y. Brenier, *Polar factorization and monotone rearrangement of vector-valued functions*, Communication in Pure and Applied Mathematics (64), pp. 375–417 (1991).

[5] E.J. Dean; R. Glowinski, *Numerical methods for fully nonlinear elliptic equations of the Monge Ampere type* , Computer Methods in Applied Mechanics and Engineering (195), pp. 1344–1386 (2006).

[6] A. Iollo; D. Lombardi, *A lagrangian scheme for the solution of the optimal mass transfer problem*, Journal of Computational Physics, pp. 3430–3442 (2011).

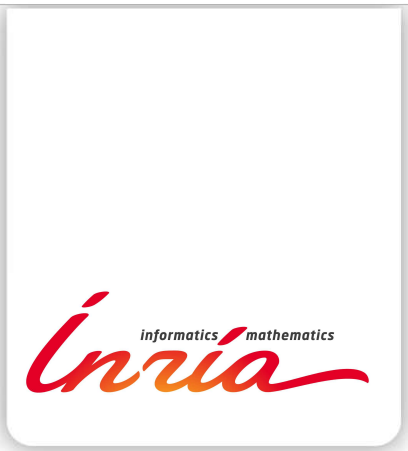
[7] G. Monge, *Memoire sur la Theorie des Déblais et des Remblais*, Histoire de l'Academie des Sciences de Paris, (1781).

[8] N. Papadakis ; G. Peyré ; E. Oude, *Optimal Transport with Proximal Splitting*, SIAM on Journal on Imaging Sciences (7), pp. 212–238 (2014).

[9] C. Villani, *Topics in optimal transportation*, American Mathematical Society (2003).

[10] C. Villani, *Optimal Transport, old and new*, Springer-Verlag (2009).

[11] L. Weynans; A. Magni; *Consistency, accuracy and entropy behaviour of remeshed particle methods*; ESAIM: Mathematical Modelling and Numerical Analysis (47), pp. 57–81 (2013).



RESEARCH CENTRE
BORDEAUX – SUD-OUEST

200 avenue de la Vieille Tour
33405 Talence Cedex

Publisher
Inria
Domaine de Voluceau - Rocquencourt
BP 105 - 78153 Le Chesnay Cedex
inria.fr

ISSN 0249-6399