Overrelaxed Sinkhorn–Knopp Algorithm for Regularized Optimal Transport

Alexis THIBAULT
École Normale Supérieure
Paris, France

Lénaïc CHIZAT
École Normale Supérieure
Paris Dauphine (PSL research University)
Paris, France

Charles DOSSAL
INSA Toulouse
Toulouse, France

Nicolas PAPADAKIS
CNRS, Institut de Mathématiques de Bordeaux
Talence, France

Abstract

This article describes a method for quickly computing the solution to the regularized optimal transport problem. It generalizes and improves upon the widely-used iterative Bregman projections algorithm (or Sinkhorn–Knopp algorithm). The idea is to overrelax the Bregman projection operators, allowing for faster convergence. In practice this corresponds to elevating the diagonal scaling factors to a given power, at each step of the algorithm. We propose a simple method for establishing global convergence by ensuring the decrease of a Lyapunov function at each step. An adaptive choice of overrelaxation parameter based on the Lyapunov function is constructed. We also suggest a heuristic to choose a suitable asymptotic overrelaxation parameter, based on a local convergence analysis. Our numerical experiments show a gain in convergence speed by an order of magnitude in certain regimes.

1 Introduction

Optimal Transport is an efficient and flexible tool to compare two probability distributions which has been popularized in the computer vision community in the context of discrete histograms [Rubner et al., 2000]. The introduction of entropic regularization in [Cuturi, 2013] has made possible the use of the fast Sinkhorn–Knopp algorithm [Sinkhorn, 1964] scaling with high dimensional data. Regularized optimal transport have thus been intensively used in Machine Learning with applications such as Geodesic PCA [Seguy and Cuturi, 2015], domain adaptation [Courty et al., 2015], data fitting [Frogner et al., 2015], training of Boltzmann Machine [Montavon et al., 2016] or dictionary learning [Rolet et al., 2016, Schmitz et al., 2017].

The computation of optimal transport between two data relies on the estimation of an optimal transport matrix, the entries of which represent the quantity of mass transported between data locations. Regularization of optimal transport with strictly convex regularization [Cuturi, 2013, Dessein et al., 2016] nevertheless involves a spreading of the mass. Hence, for particular purposes such as color interpolation [Rabin and Papadakis, 2014] or gradient flow [Chizat et al., 2016], it is necessary to consider very small regularization of the problem. In this setting, the regularized transport problem can be ill-conditioned and the Sinkhorn–Knopp algorithm converges slowly. This is the issue we want to tackle here. Before going into further details, we now briefly introduce the main notations and concepts used all along this article.

1.1 Discrete optimal transport

We consider two discrete probability measures $\mu^k \in \mathbb{R}^{n_k}_{+*}$. Let us define the two following linear operators

$$A_1 : \begin{cases} \mathbb{R}^{n_1 n_2} \to \mathbb{R}^{n_1} \\ (A_1 x)_i = \sum_j x_{i,j} \end{cases} \qquad A_2 : \begin{cases} \mathbb{R}^{n_1 n_2} \to \mathbb{R}^{n_2} \\ (A_2 x)_j = \sum_i x_{i,j}, \end{cases}$$

as well as the affine constraint sets

$$\mathcal{C}_k = \left\{ \gamma \in \mathbb{R}^{n_1 n_2} \mid A_k \gamma = \mu^k \right\}.$$

Given a cost matrix c, where c_{ij} represents the cost of moving mass μ^1_i to μ^2_j, the optimal transport problem corresponds to the estimation of an optimal transport matrix γ solution of:

$$\min_{\gamma \in \mathcal{C}_1 \cap \mathcal{C}_2 \cap \mathbb{R}^{n_1 n_2}_+} \langle c, \gamma \rangle := \sum_{i,j} c_{i,j} \gamma_{i,j}.$$

This is a linear programming problem whose resolution becomes intractable for large problems.

1.2 Regularized optimal transport

In [Cuturi, 2013], it has been proposed to regularize this problem by adding a strictly convex entropy regularization:

$$\min_{\gamma \in \mathcal{C}_1 \cap \mathcal{C}_2 \cap \mathbb{R}^{n_1 n_2}_+} K^\varepsilon(\gamma) := \langle c, \gamma \rangle + \varepsilon \, \mathrm{KL}(\gamma, \mathbf{1}), \tag{1}$$

with $\varepsilon > 0$, $\mathbf{1}$ is the matrix of size $n_1 \times n_2$ full of ones and the Kullback-Leibler divergence is

$$\mathrm{KL}(\gamma, \xi) = \sum_{i,j} \gamma_{i,j} \left(\log \left(\frac{\gamma_{i,j}}{\xi_{i,j}} \right) - 1 \right) + \sum_{i,j} \xi_{i,j} \tag{2}$$

with the convention $0 \log 0 = 0$. It was shown in [Benamou et al., 2015] that the regularized optimal transport matrix γ^*, which is the unique minimizer of problem (1), is the Bregman projection of $\gamma^0 = e^{-c/\varepsilon}$ (here and in the sequel, exponentiation is meant entry-wise) onto $\mathcal{C}_1 \cap \mathcal{C}_2$:

$$\gamma^* = \operatorname*{argmin}_{\mathcal{C}_1 \cap \mathcal{C}_2} K^\varepsilon(\gamma) = P_{\mathcal{C}_1 \cap \mathcal{C}_2}(e^{-c/\varepsilon}), \tag{3}$$

where $P_\mathcal{C}$ is the Bregman projection onto $\mathcal{C}$ defined as

$$P_\mathcal{C}(\xi) := \operatorname*{argmin}_{\gamma \in \mathcal{C}} \mathrm{KL}(\gamma, \xi).$$

1.3 Sinkhorn–Knopp algorithm

Iterative Bregman projections onto $\mathcal{C}_1$ and $\mathcal{C}_2$ converge to a point in the intersection $\mathcal{C}_1 \cap \mathcal{C}_2$ [Bregman, 1967]. Hence, the so-called Sinkhorn–Knopp algorithm (SK) [Sinkhorn, 1964] that performs alternate Bregman projections, can be considered to compute the regularized transport matrix:

$$\gamma^0 = e^{-c/\varepsilon} \qquad \gamma^{\ell+1} = P_{\mathcal{C}_2}(P_{\mathcal{C}_1}(\gamma^\ell)),$$

and we have $\lim_{l \to +\infty} \gamma^\ell = P_{\mathcal{C}_1 \cap \mathcal{C}_2}(\gamma^0) = \gamma^*$. In the discrete setting, these projections correspond to diagonal scalings of the input:

$$\begin{aligned} P_{\mathcal{C}_1}(\gamma) &= \mathrm{diag}(a)\gamma & \text{with} \quad a &= \mu^1 \oslash A_1 \gamma \\ P_{\mathcal{C}_2}(\gamma) &= \gamma \, \mathrm{diag}(b) & \text{with} \quad b &= \mu^2 \oslash A_2 \gamma \end{aligned} \tag{4}$$

where $\oslash$ is the pointwise division. To compute numerically the solution one simply has to store $(a^\ell, b^\ell) \in \mathbb{R}^{n_1} \times \mathbb{R}^{n_2}$ and to iterate

$$a^{\ell+1} = \mu^1 \oslash \gamma^0 b^\ell \qquad b^{\ell+1} = \mu^2 \oslash {}^t\gamma^0 a^{\ell+1}.$$

We then have $\gamma^\ell = \operatorname{diag}(a^\ell)\gamma^0 \operatorname{diag}(b^\ell)$.

Another way to interpret the SK algorithm is as an alternate maximization algorithm on the dual of the regularized optimal transport problem. The dual problem of (1) is

$$\max_{\substack{\alpha\in\mathbb{R}^n\\ \beta\in\mathbb{R}^m}} E(\alpha,\beta) := \langle\alpha,\mu^1\rangle + \langle\beta,\mu^2\rangle - \varepsilon\sum_{i,j} e^{(\alpha_i+\beta_j-c_{i,j})/\varepsilon}. \tag{5}$$

The function E is concave, continuously differentiable and admits a maximizer, so alternate maximization converges and we recover SK algorithm by posing for $a_i = e^{\alpha_i/\varepsilon}$, $b_j = e^{\beta_j/\varepsilon}$ and $\gamma^0_{i,j} = e^{-c_{i,j}/\varepsilon}$.

Efficient parallel computations can be considered [Cuturi, 2013] and one can almost reach real-time computation for large scale problem for certain class of cost matrices c allowing the use of seprable convolutions [Solomon et al., 2015]. For small values of the parameter ε, numerical issues can arise and a stabilization of the algorithm is necessary [Chizat et al., 2016]. The convergence of the process can nevertheless be very slow when ε is small.

1.4 Overview and contributions

In this paper, we consider an overrelaxation scheme designed to accelerate the Sinkhorn–Knopp algorithm. We first present and show the convergence of our algorithm in Section 2. In Section 3, we analyze the local convergence rate of the algorithm to justify the acceleration. We finally demonstrate numerically in Section 4 the good behavior of our method, where larger accelerations are observed for decreasing values of ε.

1.5 Related works

The introduction of relaxation variables through heavy ball approaches [Polyak, 1964] has recently gained in popularity to speed up the convergence of algorithms optimizing convex [Ghadimi et al., 2014] or non convex [Zavriev and Kostyuk, 1993, Ochs, 2016] problems. Our specific approach is very much related to the SOR algorithm [Young, 2014], which is a classical way to solve linear systems. Similar schemes have been empirically considered to accelerate the SK algorithm in [Peyré et al., 2016, Schmitz et al., 2017]. The convergence of these algorithms has nevertheless not been studied yet in the context of regularized optimal transport.

2 Overrelaxed Sinkhorn–Knopp algorithm

As illustrated in Figure 1 (a-b), SK algorithm, that performs alternate Bregman projections onto the affine sets $\mathcal{C}_1$ and $\mathcal{C}_2$, can be very slow when $\varepsilon \to 0$. The idea developed in this paper is to perform overrelaxed projections in order to accelerate the process, as displayed in Figure 1 (c).

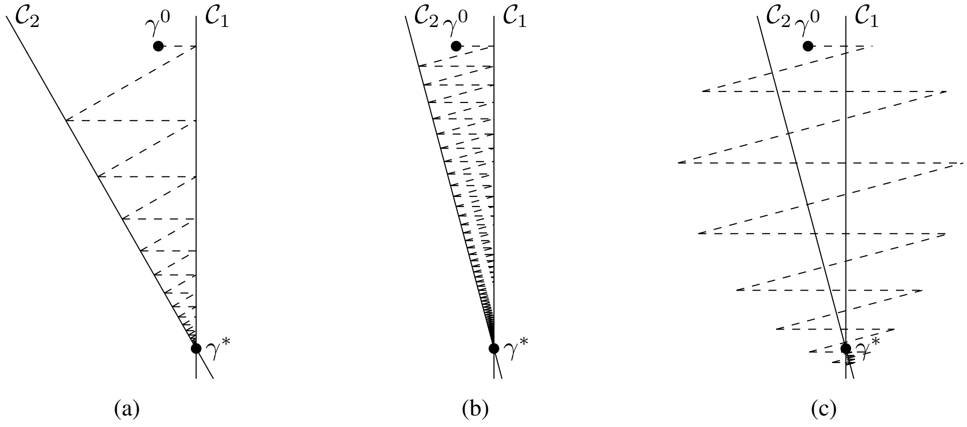


Figure 1: The trajectory of γ^ℓ given by the SK algorithm is illustrated for decreasing values of ε in (a) and (b). Overrelaxed projections (c) typically accelerate the convergence rate.

2.1 Overrelaxed projections

We define the ω-overrelaxed projection operator $P^{\omega}_{\mathcal{C}_k}$ as

$$\log P^{\omega}_{\mathcal{C}_k}(\gamma) = (1-\omega)\log\gamma + \omega \log P_{\mathcal{C}_k}(\gamma), \tag{6}$$

where the logarithm is taken coordinate-wise. Note that $P^0_{\mathcal{C}_k}$ is the identity, $P^1_{\mathcal{C}_k} = P_{\mathcal{C}_k}$ is the standard Bregman projection and $P^2_{\mathcal{C}_k}$ is an involution (in particular because $\mathcal{C}_k$ is an affine subspace). A naive algorithm would then consist in iteratively applying $P^{\omega}_{\mathcal{C}_2} \circ P^{\omega}_{\mathcal{C}_1}$ for some choice of ω. While it often behaves well in practice, this algorithm may sometimes not converge even for reasonable values of ω. Our goal in this section is to modify this algorithm to make it robust and to guarantee convergence.

Duality gives another point of view on the iterative overrelaxed Bregman projections: they indeed correspond to a successive overrelaxation (SOR) algorithm on the dual objective E. This is a procedure which, starting from $(\alpha^0, \beta^0) = (\mathbf{0}, \mathbf{0})$, defines for $\ell \in \mathbb{N}^*$,

$$\alpha^{\ell+1} = (1-\omega)\alpha^{\ell} + \omega \arg\max_{\alpha} E(\alpha, \beta^{\ell}) \tag{7}$$

$$\beta^{\ell+1} = (1-\omega)\beta^{\ell} + \omega \arg\max_{\beta} E(\alpha^{\ell+1}, \beta). \tag{8}$$

This can be seen by using the relationships given after equation (5).

2.2 Lyapunov function

Convergence of the successive overrelaxed projections is not guaranteed in general. In order to derive a robust algorithm with provable convergence, we introduce the Lyapunov function

$$F(\gamma) = \mathrm{KL}(\gamma^*, \gamma), \tag{9}$$

where γ^* denotes the solution of the regularized OT problem. We will use this function to enforce the strict descent criterion $F(\gamma^{\ell+1}) < F(\gamma^{\ell})$ as long as the process has not converged.

The choice of (9) as a Lyapunov function is of course related to the fact that Bregman projections are used throughout the algorithm. Further, we will show (Lemma 1) that its decrease is very easy to compute and this descent criterion still allows enough freedom in the choice of the overrelaxation parameter.

Crucial properties of this Lyapunov function are gathered in the next lemma.

Lemma 1. *For any $M \in \mathbb{R}^*_+$, the sublevel set $\{\gamma \mid F(\gamma) \le M\}$ is compact. Moreover, for any γ in $\mathbb{R}^{mn}_{+*}$, the decrease of the Lyapunov function after an overrelaxed projection can be computed as*

$$F(\gamma) - F(P^{\omega}_{\mathcal{C}_k}(\gamma)) = \left\langle \mu^k, \varphi_\omega\left((A_k\gamma) \oslash \mu^k\right)\right\rangle, \tag{10}$$

where

$$\varphi_\omega(x) = x(1 - x^{-\omega}) - \omega \log x \tag{11}$$

is a real function, applied coordinate-wise.

Proof. The fact that the Kullback-Leibler divergence is jointly lower semicontinuous implies in particular that K is closed. Moreover, $K \subset \mathbb{R}^{n_1 \times n_2}_+$ is bounded because F is the sum of nonnegative, coercive functions of each component of its argument γ.

Formula (10) comes from the expression $F(\gamma^1) - F(\gamma^2) = \sum_{i,j}\left(\gamma^*_{i,j}\log(\gamma^2_{i,j}/\gamma^1_{i,j}) + \gamma^1_{i,j} - \gamma^2_{i,j}\right)$ and the relations (6) and (4). □

It follows from Lemma 1 that the decrease of F for an overrelaxed projection is very cheap to estimate, since its computational cost is linear with respect to the dimension of data μ^k. In Figure 2, we display the function $\varphi_\omega(x)$. Notice that for the Sinkhorn–Knopp algorithm, which corresponds to $\omega = 1$, the function φ_ω is always nonnegative. For other values $1 \le \omega < 2$, it is nonnegative for x close to 1.

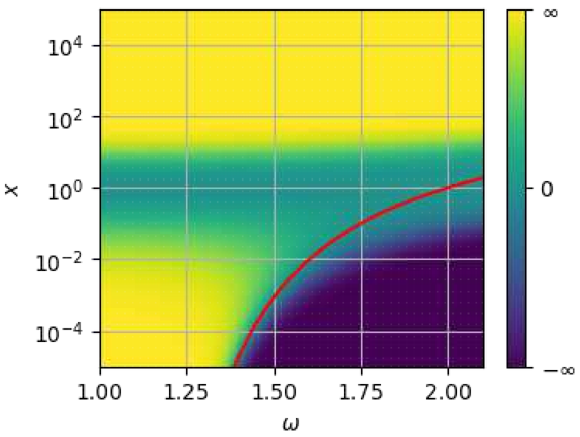


Figure 2: Value of $\varphi_\omega(x)$. The function is positive above the red line, negative below. For any relaxation parameter ω smaller than 2, there exists a neighborhood of 1 on which $\varphi_\omega(\cdot)$ is positive.

2.3 Proposed algorithm

We first give a general convergence result that later serves as a basis to design an explicit algorithm.

Theorem 1. *Let θ_1 and θ_2 be two continuous functions of γ such that*

$$\forall \gamma \in \mathbb{R}_{+*}^{n_1 n_2}, \quad F(P_{\mathcal{C}_k}^{\theta_k(\gamma)}(\gamma)) \leq F(\gamma), \tag{12}$$

where the inequality is strict whenever $\gamma \notin \mathcal{C}_k$. Consider the sequence defined by $\gamma^0 = e^{-c/\varepsilon}$ and

$$\begin{aligned}\tilde{\gamma}^{\ell+1} &= P_{\mathcal{C}_1}^{\theta_1(\gamma^\ell)}(\gamma^\ell)\\ \gamma^{\ell+1} &= P_{\mathcal{C}_2}^{\theta_2(\tilde{\gamma}^{\ell+1})}(\tilde{\gamma}^{\ell+1}).\end{aligned}$$

Then the sequence (γ^ℓ) converges to γ^*.

Lemma 2. *Let us take γ^0 in $\mathbb{R}_{+*}^{n_1 n_2}$, and denote*

$$S = \left\{\operatorname{diag}(a)\gamma^0 \operatorname{diag}(b), \quad (a,b) \in \mathbb{R}_{+*}^{n_1+n_2}\right\}$$

the set of matrices that are diagonally similar to γ^0. Then the set $S \cap \mathcal{C}_1 \cap \mathcal{C}_2$ contains exactly one element $\gamma^* = P_{\mathcal{C}_1 \cap \mathcal{C}_2}(\gamma^0)$.

Proof. We refer to [Cuturi, 2013] for a proof of this lemma. □

Proof of the theorem. First of all, notice that the operators $P_{\mathcal{C}_k}^\theta$ apply a scaling to lines or columns of matrices. All (γ^ℓ) are thus diagonally similar to γ^0:

$$\forall \ell \geq 0, \quad \gamma^\ell \in S$$

By construction of the functions θ_k, the sequence of values of the Lyapunov function $(F(\gamma^\ell))$ is non-increasing. Hence (γ^ℓ) is precompact. If ξ is a cluster point of (γ^ℓ), let us define

$$\begin{aligned}\tilde{\xi} &= P_{\mathcal{C}_1}^{\theta_1(\xi)}(\xi)\\ \xi' &= P_{\mathcal{C}_2}^{\theta_2(\tilde{\xi})}(\tilde{\xi}).\end{aligned}$$

Then by continuity of the applications, $F(\xi) = F(\tilde{\xi}) = F(\xi')$. From the hypothesis made on θ_1 and θ_2, it can be deduced that ξ is in $\mathcal{C}_1$ and that $\tilde{\xi}$ is in $\mathcal{C}_2$. Therefore $\xi' = \tilde{\xi} = \xi$ is in the intersection $\mathcal{C}_1 \cap \mathcal{C}_2$. By Lemma 2, $\xi = \gamma^*$, and the whole sequence (γ^ℓ) converges to the solution. □

We can construct explicitly functions θ_k as needed in Theorem 1 using the following lemma.

Lemma 3. *Let $1 \leq \theta < \omega$. Then, for any $\gamma \in \mathbb{R}_{+*}^{n_1 n_2}$, one has*

$$F(P_{\mathcal{C}_k}^{\theta}(\gamma)) \leq F(P_{\mathcal{C}_k}^{\omega}(\gamma)). \tag{13}$$

Moreover, equality occurs if and only if $\gamma \in \mathcal{C}_k$.

Proof. Thanks to Lemma 1, one knows that

$$F(P_{\mathcal{C}_k}^{\theta}(\gamma)) - F(P_{\mathcal{C}_k}^{\omega}(\gamma)) = \left\langle \mu^k, (\varphi_\omega - \varphi_\theta)\left(\frac{A_k\gamma}{\mu^k}\right)\right\rangle.$$

The function that maps $t \in [1, \infty)$ to $\varphi_t(x)$ is non-increasing since $\partial_t \varphi_t(x) = (x^{1-t} - 1)\log x$. For $x \neq 1$, it is even strictly decreasing. Thus inequality (13) is valid, with equality *iff* $A_k\gamma = \mu^k$. □

We now argue that a good choice for the functions θ_k may be constructed as follows. Pick a target parameter $\theta_0 \in [1; 2)$ and a small security distance $\delta > 0$. Define the functions Θ^* and Θ as

$$\Theta^*(u) = \sup\left\{\omega \in [1;2] \mid \varphi_\omega(\min u) \geq 0\right\}, \tag{14}$$
$$\Theta(u) = \min(\max(1, \Theta^*(u) - \delta), \theta_0), \tag{15}$$

where $\min u$ denotes the smallest coordinate of the vector u.

Proposition 1. *The function*

$$\theta_k(\gamma) = \Theta\left((A_k\gamma) \oslash \mu^k\right) \tag{16}$$

is continuous and verifies the descent condition (12).

Proof. Looking at Figure 2 can help understand this proof. Since the partial derivative of $\partial_\omega \varphi_\omega(x)$ is nonzero for any $x < 1$, the implicit function theorem proves the continuity of Θ^*. The function $\Theta^*\left((A_k\gamma) \oslash \mu^k)\right)$ is such that every term in relation (10) is non-negative. Therefore, by Lemma 3, using this parameter minus δ ensures the strong decrease (12) of the Lyapunov function. Constraining the parameter to $[1, \theta_0]$ preserves this property. □

This construction, which is often an excellent choice in practice, has several advantages:

- it allows to choose arbitrarily the parameter θ_0 that will be used eventually when the algorithm is close to convergence (we motivate what are good choices for θ_0 in Section 3);
- it is also an easy approach to having an adaptive method, as the approximation of Θ^* has a negligible cost (it only requires to solve a one dimensional problem that depends on the smallest value of $(A_k\gamma) \oslash \mu^k$, which can be done in a few iterations of Newton's method).

The resulting algorithm, which is proved to be convergent by Theorem 1, is written in pseudocode in Algorithm 1. It uses the function Θ defined implicitly in (15), which in practice is approximated with a few iterations of Newton's method on the function $\omega \mapsto \varphi_\omega(\min u)$ which is decreasing is as can be seen on Figure 2. With the choice $\theta_0 = 1$, one recovers exactly the original SK algorithm.is

3 Acceleration of local convergence rate

In order to justify the acceleration of convergence that is observed in practice, we now study the local convergence rate of the overrelaxed algorithm, which follows from the classical convergence analysis of the linear SOR method. Our result involves the second largest eigenvalue of the matrix

$$M_1 = \operatorname{diag}(1 \oslash \mu^1)\,\gamma^*\ \operatorname{diag}(1 \oslash \mu^2)\,{}^t\gamma^* \tag{17}$$

where γ^* is the solution to the regularized OT problem (the largest eigenvalue is 1, associated to the eigenvector $\mathbf{1}$). We denote the second largest eigenvalue by $1 - \eta$, it satisfies $\eta > 0$ [Knight, 2008].

Algorithm 1 Overrelaxed SK algorithm

```
Require: $\mu^1 \in \mathbb{R}^{n_1}$, $\mu^2 \in \mathbb{R}^{n_2}$, $c \in \mathbb{R}_+^{n_1 \times n_2}$
  Set $a = \mathbf{1}_{n_1}$, $b = \mathbf{1}_{n_2}$, $\gamma^0 = e^{-c/\varepsilon}$, $\theta_0 \in [1; 2)$ and $\eta > 0$
  while $||a \otimes \gamma^0 b - \mu_1|| > \eta$ do
    $\tilde{a} = \mu_1 \oslash (\gamma^0 b)$,
    $\omega = \Theta(a \oslash \tilde{a})$
    $a = a^{1-\omega} \otimes \tilde{a}^{\omega}$
    $\tilde{b} = \mu_2 \oslash ({}^t\gamma^0 a)$
    $\omega = \Theta(b \oslash \tilde{b})$
    $b = b^{1-\omega} \otimes \tilde{b}^{\omega}$
  end while
  return $\gamma = \operatorname{diag}(a)\gamma^0 \operatorname{diag}(b)$
```

Proposition 2. *The SK algorithm converges locally at a linear rate $1-\eta$. For the optimal choice of extrapolation parameter $\theta^* = 2/(1+\sqrt{\eta})$, the overrelaxed projection algorithm converges locally linearly at a rate $(1-\sqrt{\eta})/(1+\sqrt{\eta})$. The local convergence of the overrelaxed algorithm is guaranteed for $\theta \in]0,2[$ and the linear rate is given on Figure 3 as a function of $1-\eta$ and θ.*

Proof. In this proof, we focus on the dual problem and we recall the relationship $\gamma^\ell = e^{\alpha^\ell/\varepsilon}\gamma^0 e^{\beta^\ell/\varepsilon}$ between the iterates of the overrelaxed projection algorithm γ^ℓ and the iterates $(\alpha^\ell, \beta^\ell)$ of the SOR algorithm on the dual problem (7), initialized with $(\alpha^0, \beta^0) = (0,0)$. The dual problem (5) is invariant by translations of the form $(\alpha, \beta) \mapsto (\alpha - k, \beta + k)$, $k \in \mathbb{R}$, but is otherwise strictly convex on any subspace which does not contain the line $\mathbb{R}(\mathbf{1}, -\mathbf{1})$. In order to deal with this invariance (which cancels in the corresponding primal iterates), consider the subspace S of pairs of dual variables (α, β) that satisfy $\alpha_1 = 0$, let π_S be the (non orthogonal) projection on S of kernel $(\mathbf{1}, -\mathbf{1})$ and let $(\alpha^*, \beta^*) \in S$ be a dual maximizer.

Since one SOR iteration is a smooth map, the local convergence properties of the SOR algorithm are characterized by the local convergence of its linearization, which here corresponds to the SOR method applied to the maximization of the quadratic Taylor expansion of the dual objective E at (α^*, β^*). This defines an affine map $M_\theta : (\alpha^\ell, \beta^\ell) \mapsto (\alpha^{\ell+1}, \beta^{\ell+1})$ whose spectral properties are well known [Ciarlet, 1982, Young, 2014] (see also [Chizat, 2017, chapter 4] for the specific case of convex minimization). For the case $\theta = 1$, this corresponds to the matrix M_1 defined in (17). Specifically, in the non strictly concave case [Hadjidimos, 1985], we have that the operator $\pi_S \circ M_1^\ell$ converges at the linear rate $1-\eta$ towards the projector on (α^*, β^*) and that the convergence of $\pi_S \circ M_\theta^\ell$ is guaranteed for $\theta \in]0,2[$, with the rate

$$f(\theta, \eta) = \begin{cases} \theta - 1 & \text{if } \theta > \theta^* \\ \frac{1}{2}\theta^2(1-\eta) - (\theta - 1) + \frac{1}{2}\sqrt{(1-\eta)\theta^2(\theta^2(1-\eta) - 4(\theta - 1))} & \text{otherwise,} \end{cases}$$

where $\theta^* := 2/(1+\sqrt{\eta})$ is the optimal parameter, for which $f(\theta^*, \eta) = (1-\sqrt{\eta})/(1+\sqrt{\eta})$. The function f is plotted in Figure 3.

To switch from these dual convergence results to primal convergence results, remark that $\gamma^\ell \to \gamma^*$ implies $\mathrm{KL}(\gamma^\ell, \gamma^0) \to \mathrm{KL}(\gamma^*, \gamma^0)$ which in turn implies $E(\alpha^\ell, \beta^\ell) \to \max E$ so invoking the partial strict concavity of E, $\pi_S(\alpha^\ell, \beta^\ell) \to (\alpha^*, \beta^*)$. The converse implication is direct so it holds $[\pi_S(\alpha^\ell, \beta^\ell) \to (\alpha^*, \beta^*)] \Leftrightarrow [\gamma^\ell \to \gamma^*]$. We conclude by noting the fact that $\pi_S(\alpha^\ell, \beta^\ell)$ converges at a linear rate implies the same rate on γ^ℓ, thanks to the relationship between the iterates. □

Corollary 1. *The previous local convergence analysis applies to Algorithm 1 and the local convergence rate is governed by the choice of the target extrapolation parameter θ_0.*

Proof. What we need to show is that eventually one always has $\Theta(\gamma^\ell) = \theta_0$. This can be seen from the quadratic Taylor expansion $\varphi_{\theta_0}(1+z) = z^2(\theta_0 - \theta_0^2/2) + o(z^2)$, which shows that for any choice of $\theta_0 \in]1,2[$, there is a neighborhood of 1 on which $\varphi_{\theta_0}(\cdot)$ is nonnegative. □

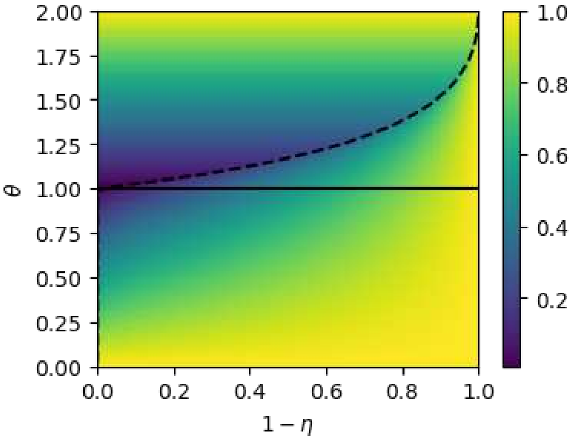
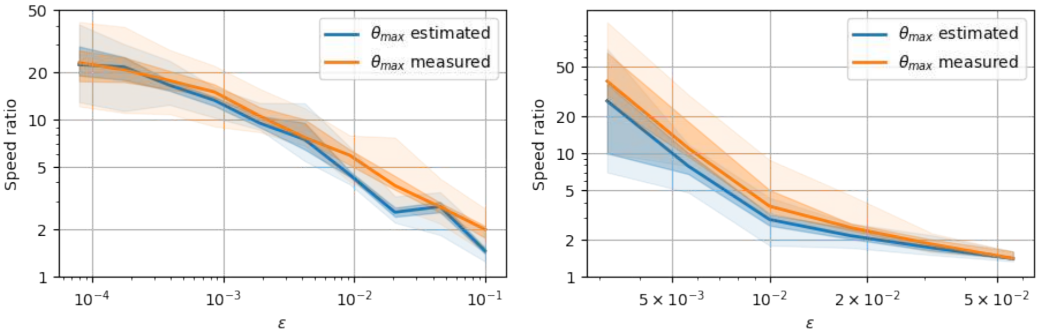


Figure 3: Local linear rate of convergence of the overrelaxed algorithm as a function of $1 - \eta$, the local convergence rate of SK algorithm and θ the overrelaxation parameter. (plain curve) the original rate is recovered for $\theta = 1$. (dashed curve) optimal overrelaxation parameter θ^*.



(a) Quadratic cost, random marginals

(b) Random cost, uniform marginals

Figure 4: Speed ratio of SK algorithm and its accelerated version Algorithm 1 w.r.t parameter ε.

4 Experimental results

We compare Algorithm 1 to SK algorithm on two very different optimal transport settings. In setting (a) we consider the domain $[0, 1]$ discretized into 100 samples and the squared Euclidean transport cost on this domain. The marginals are densities made of the sum of a base plateau of height 0.1 and another plateau of height and boundaries chosen uniformly in $[0, 1]$, subsequently normalized. In setting (b) the cost is a 100×100 random matrix with entries uniform in $[0, 1]$ and the marginals are uniform.

Given an estimation of $1 - \eta$, the local convergence rate of SK algorithm, we define θ_0 as the optimal parameter as given in Proposition 2. For estimating η, we follow two strategies. For strategy "estimated" (in blue on Figure 4), η is measured by looking at the local convergence rate of SK run on another random problem of the same setting and for the same value of ε. For strategy "measured" (in orange on Figure 4) the parameter is set using the local convergence rate of SK run on the same problem. Of course, the latter is an unrealistic strategy but it is interesting to see in our experiments that the "estimated" strategy performs almost as well as the "measured" one, as shown on 4.

Figure 4 displays the ratio of the number of iterations required to reach a precision of 10^{-6} on the dual variable α for SK algorithm and Algorithm 1. It is is worth noting that the complexity per iteration of these algorithms is the same modulo negligible terms, so this ratio is also the runtime ratio (our algorithm can also be parallelized on GPUs just as SK algorithm). In both experimental settings, for low values of the regularization parameter ε, the acceleration ratio is above 20 with Algorithm 1.

5 Conclusion and perspectives

The SK algorithm is widely used to solve entropy regularized OT. The use of overrelaxed projections turns out to be a natural and simple idea to accelerate convergence while keeping many nice properties of this algorithm (first order, parallelizable, simple). We have proposed an algorithm that adaptively chooses the overrelaxation parameter so as to guarantee global convergence. The acceleration of the convergence speed is numerically impressive, in particular in low regularization regimes. It is theoretically supported in the local regime by the standard analysis of SOR iterations.

This idea of overrelaxation can be generalized to solve more general problems such as multi-marginal OT, barycenters, gradient flows, unbalanced OT [Chizat, 2017, chap. 4] but there is no systematic way to derive globally convergent algorithms. Our work is a step in the direction of building and understanding the properties of robust first order algorithms for solving OT. More understanding is needed regarding SOR itself (global convergence speed, choice of θ_0), but also its relation to other acceleration methods [Scieur et al., 2016, Altschuler et al., 2017].

Acknowledgments

This study has been carried out with financial support from the French State, managed by the French National Research Agency (ANR) in the frame of the GOTMI project (ANR-16-CE33-0010-01).

References

[Altschuler et al., 2017] Altschuler, J., Weed, J., and Rigollet, P. (2017). Near-linear time approximation algorithms for optimal transport via Sinkhorn iteration. *ArXiv e-prints*, arXiv:1705.09634.

[Benamou et al., 2015] Benamou, J.-D., Carlier, G., Cuturi, M., Nenna, L., and Peyré, G. (2015). Iterative Bregman projections for regularized transportation problems. *SIAM Journal on Scientific Computing*, 37(2):A1111–A1138.

[Bregman, 1967] Bregman, L. M. (1967). 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.

[Chizat, 2017] Chizat, L. (2017). *Unbalanced optimal transport: models, numerical methods, applications*. PhD thesis, Université Paris Dauphine.

[Chizat et al., 2016] Chizat, L., Peyré, G., Schmitzer, B., and Vialard, F.-X. (2016). Scaling Algorithms for Unbalanced Transport Problems. *ArXiv e-prints*, arXiv:1607.05816.

[Ciarlet, 1982] Ciarlet, P. (1982). Introduction à l'analyse numérique matricielle et à l'optimisation.

[Courty et al., 2015] Courty, N., Flamary, R., Tuia, D., and Rakotomamonjy, A. (2015). Optimal Transport for Domain Adaptation. *ArXiv e-prints*, arXiv:1507.00504.

[Cuturi, 2013] Cuturi, M. (2013). Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems (NIPS'13)*, pages 2292–2300.

[Dessein et al., 2016] Dessein, A., Papadakis, N., and Rouas, J.-L. (2016). Regularized Optimal Transport and the Rot Mover's Distance. *ArXiv e-prints*, arXiv:1610.06447.

[Frogner et al., 2015] Frogner, C., Zhang, C., Mobahi, H., Araya-Polo, M., and Poggio, T. (2015). Learning with a Wasserstein Loss. *ArXiv e-prints*, arXiv:1506.05439.

[Ghadimi et al., 2014] Ghadimi, E., Feyzmahdavian, H. R., and Johansson, M. (2014). Global convergence of the Heavy-ball method for convex optimization. *ArXiv e-prints*, arXiv:1412.7457.

[Hadjidimos, 1985] Hadjidimos, A. (1985). On the optimization of the classical iterative schemes for the solution of complex singular linear systems. *SIAM Journal on Algebraic Discrete Methods*, 6(4):555–566.

[Knight, 2008] Knight, P. A. (2008). The Sinkhorn–Knopp algorithm: convergence and applications. *SIAM Journal on Matrix Analysis and Applications*, 30(1):261–275.

[Montavon et al., 2016] Montavon, G., Müller, K.-R., and Cuturi, M. (2016). Wasserstein training of restricted boltzmann machines. In Lee, D. D., Sugiyama, M., Luxburg, U. V., Guyon, I., and Garnett, R., editors, *Advances in Neural Information Processing Systems 29*, pages 3718–3726.

[Ochs, 2016] Ochs, P. (2016). Local Convergence of the Heavy-ball Method and iPiano for Non-convex Optimization. *ArXiv e-prints*, arXiv:1606.09070.

[Peyré et al., 2016] Peyré, G., Chizat, L., Vialard, F.-X., and Solomon, J. (2016). Quantum optimal transport for tensor field processing. *ArXiv e-prints*, arXiv:1612.08731.

[Polyak, 1964] Polyak, B. (1964). Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 4(5):1 – 17.

[Rabin and Papadakis, 2014] Rabin, J. and Papadakis, N. (2014). Non-convex relaxation of optimal transport for color transfer between images. In *NIPS Workshop on Optimal Transport for Machine Learning (OTML'14)*.

[Rolet et al., 2016] Rolet, A., Cuturi, M., and Peyré, G. (2016). Fast dictionary learning with a smoothed Wasserstein loss. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 51 of *JMLR Workshop and Conference Proceedings*, pages 630–638.

[Rubner et al., 2000] Rubner, Y., Tomasi, C., and Guibas, L. J. (2000). The Earth Mover's Distance as a metric for image retrieval. *International Journal of Computer Vision*, 40(2):99–121.

[Schmitz et al., 2017] Schmitz, M. A., Heitz, M., Bonneel, N., Ngolè Mboula, F. M., Coeurjolly, D., Cuturi, M., Peyré, G., and Starck, J.-L. (2017). Wasserstein Dictionary Learning: Optimal Transport-based unsupervised non-linear dictionary learning. *ArXiv e-prints*, arXiv:1708.01955.

[Scieur et al., 2016] Scieur, D., d'Aspremont, A., and Bach, F. (2016). Regularized Nonlinear Acceleration. *ArXiv e-prints*, arXiv:1606.04133.

[Seguy and Cuturi, 2015] Seguy, V. and Cuturi, M. (2015). Principal geodesic analysis for probability measures under the optimal transport metric. In *Advances in Neural Information Processing Systems*, pages 3312–3320.

[Sinkhorn, 1964] Sinkhorn, R. (1964). A relationship between arbitrary positive matrices and doubly stochastic matrices. *The annals of mathematical statistics*, 35(2):876–879.

[Solomon et al., 2015] Solomon, J., de Goes, F., Peyré, G., Cuturi, M., Butscher, A., Nguyen, A., Du, T., and Guibas, L. (2015). Convolutional Wasserstein distances: Efficient optimal transportation on geometric domains. *ACM Transactions on Graphics*, 34(4):66:1–66:11.

[Young, 2014] Young, D. M. (2014). *Iterative solution of large linear systems*. Elsevier.

[Zavriev and Kostyuk, 1993] Zavriev, S. K. and Kostyuk, F. V. (1993). Heavy-ball method in nonconvex optimization problems. *Computational Mathematics and Modeling*, 4(4):336–341.