Geometric Data Analysis project: convolutional Wasserstein distances.

Hippolyte Pilchen*, Victor Barberteguy†– Presented on December, 18^{th} 2023

Abstract

Wasserstein distance is a core topic in Optimal Transport tasks such as segmentation or computing barycenters. We analyze and run additional experiments to the article of Solomon et al. *Convolutional Wasserstein Distances: Efficient Optimal Transportation on Geometric Domains*, that proposes a computationally efficient method based on Wasserstein distances and diffusion to tackle tasks on 3D meshes or images. In addition to try the method on connected graphs, we quantify the error brought by approximating the true kernel with the heat kernel and highlight the effect of several parameters such as the diffusion time and the regularization value. This led us to eventually identify issues to be addressed in future work to improve on the results obtained by Solomon et al.

Keywords: Optimal Transport, Wasserstein distances, Shape barycenter

1 Introduction

To work on shapes and other kind of structures, one can think of it as probability distributions. Then, it becomes possible to map these shapes one to the other and modify them enabling new unknown distributions. To compute this mapping between distributions, one need a metric which takes into account the shape of the manifold on which the distribution is supported. Since the optimal mapping is desired, the problem becomes a minimizing problem over this particular metric. This appropriate metric is known as the Wasserstein distance. It was first introduced in the optimal transport problem: the Earth Mover problem as described by Gaspard Monge. On an intuitive level, considering each distribution as a unit of earth (soil) deposited on a manifold M, the metric represents the minimal "cost" associated with transforming one heap into another. This cost is presumed to be the product of the amount of earth that must be relocated and the mean distance it needs to be moved. This metric was subsequently used in several more complex problems derived from Monge's problem and is now at the heart of various optimal transport tasks. Even though these problems can be formulated orally in simple ways, their computation remains a real challenge. The complexity of this optimization problem makes it difficult to use this distance in cases with relatively large dimensions. Indeed, computing a single Wasserstein distance is quadratic on the distribution dimension. Reducing the computation cost of some optimal transport tasks remains a complex problem and is not completely solved. The paper we studied, *Convolutional Wasserstein Distances: Efficient Optimal Transportation on Geometric Domains* [Solomon et al. 2015], is part of this line of research. The authors have succeeded in reducing the computational costs of several tasks on 3D meshes: calculating the barycenter of distributions on meshes, manipulating the color histogram of an image, mapping points between different shapes, etc. Theses achievements were possible thanks to entropic regularization of Optimal Transport (OT) and heat kernels to approximate geodesic distances.

In the present project, we realize an in-depth analysis of the heat kernel to calculate Wasserstein distances for graphs. We solve an optimal transport problem on graphs for both the true kernel and the heat kernel. We implemented several tasks using Wasserstein distances on graphs. Furthermore, we implemented heat diffusion on 3D meshes. Code and resources are available on the page of the project [1].

2 Background and preliminaries

For all the following calculations we consider (same setting as in [Solomon et al. 2015]): a compact, connected Riemannian manifold M rescaled to have unit volume representing a domain such as a surface. We denote the geodesic distance function by $M \times M \to \mathbb{R}^+$, where $d(x, y)$ represents the shortest distance from x to y along M. The notation Prob(M) denotes the space of probability measures on M, and Prob$(M \times M)$ refers to probability measures on the product space of M with itself. To avoid confusion, we use the term "marginals" to refer to elements $\mu_0, \mu_1, \ldots \in$ Prob(M) and the term "couplings" to refer to joint probabilities $\pi_0, \pi_1, \ldots \in$ Prob$(M \times M)$. Consider a scenario where a distribution represented by the marginal μ_0 can be transformed into another distribution represented by the marginal μ_1 through the utilization of a transportation plan π. This transportation plan, a coupling in the space of probability measures Prob$(M \times M)$, characterizes the amount of mass $\pi(x, y)$ to be moved from the distribution μ_0 at point x to point y, resulting in the creation of the distribution μ_1 in aggregate. The set of admissible transportation plans, denoted as $\mathbf{\Pi}(\mu_0, \mu_1)$, is defined as follows:

$$\mathbf{\Pi}(\mu_0, \mu_1) = \{\pi \in \text{Prob}(M \times M) : \pi(\cdot, M), \pi(M, \cdot) = \mu_0, \mu_1\}.$$

This set captures transportation plans that satisfy mass conservation laws (incompressibility), ensuring that the transformation adheres to the specified source and target distributions.

A coupling π is considered absolutely continuous with respect to the Lebesgue measure (defined as the volume measure in the paper) when it possesses a density function p, such that $\pi(U) = \int_U p(x, y)\, dx\, dy$ for all $U \subseteq M \times M$. π will both represents the measure and its density as in the paper [Solomon et al. 2015].

After Gaspard Monge intuition, L. Kantorovich [Kantorovich 2006] was the first to formulate the optimal transport problem as an optimization problem. The objective is to minimize the average cost of transportation from one distribution to another on all the possible mapping plans:

$$\inf_{\pi \in \Pi} \left\{ \iint c(x, y) d\pi(x, y), \pi_1 = \mu_0, \pi_2 = \mu_0 \right\} \tag{1}$$

with the two values π_1 and π_2 which represents the incompressibility constraints and c the cost function.

For discretized problems, linear solvers are usually used such as the ones developped in the python library *CVXPY*. However, as the dimension of the setting increases these solvers become to slow and the problem becomes computationally too complex. Indeed, in 1979 Dimitri P. Bertsekas [Bertsekas 1988] proved that with some relaxations these problems could be solved in $\mathcal{O}(N^2)$ with

*hippolyte.pilchen@polytechnique.edu
Department of applied Mathematics, ENS Paris-Saclay
†victor.barberteguy@polytechnique.edu
Department of Computer Science, Ecole Polytechnique

[1] https://github.com/VictorBbt/Convolutional-Wassertein-Distances

N the dimension of the distributions. Following this, other algorithms were designed such as the Sinkhorn-Knopp algorithm based on the mathematical findings in [Sinkhorn 1964]. This algorithm also scales in $\mathcal{O}(N^2)$ and has been widely used in optimal transport. It uses regularization to approximate solutions for the transportation problem. While interior point methods have traditionally utilized barrier functions to convert linear programs into strictly convex problems, the use of entropic regularizers, especially in the case of optimal transportation, presents several notable advantages as outlined in [Cuturi 2013]. With entropic regularization, the resolution of optimal transportation involves an iterative scaling method as in the Sinkhorn-Knopp algorithm. Marco Cuturi democratized the use of this method by showing that it could be parallelized and thus take advantage of technological advances to accelerate calculations using GPUs [Cuturi 2013]. In this way, it became possible to solve more complex tasks, with distributions in higher-dimensional spaces: the barycenter of several 3D shapes, soft mapping between different shapes etc.

The paper studied is part of this research dynamic. It takes advantage of these iterative scaling methods and makes them even more scalable by eliminating the need to calculate geodesic distances (the cost matrix in the Kantorovitch problem) thanks to the heat method[Crane et al. 2017]. This reduction in complexity makes it possible to accomplish complex tasks (barycenters, soft mapping, skeleton layout, color histogram manipulation, etc.) with distributions supported on high-dimensional spaces (3D meshes, graphs, RGB pixel space, etc.).

3 The method: Convolutional Wasserstein Distance

Optimal Transport

The Kantorovitch problem can be reformulated as a calculation of the Wasserstein distance described in the introduction. Therefore, for the second order Wasserstein distance, the optimization problem for finding the optimal transportation becomes:

$$W_2(\mu_{0,1}) = \left(\inf_{\pi \in \Pi} \int_{M \times M} d(x,y)^2 \, d\pi(x,y) \right)^{\frac{1}{2}} \tag{2}$$

In the paper the second order Wasserstein is used. Thus, we decided to prove the formula which links the Wasserstein distance with the Kullback-Leibler divergence for the first order Wasserstein distance as it had not been done in the studied paper.

First, the following functions are defined:

– The *Kullback-Leibler divergence* between a coupling π and a kernel K:

$$KL(\pi|K) = \iint_{M \times M} \pi(x,y) \left[\log\left(\frac{\pi(x,y)}{K(x,y)}\right) - 1 \right] dxdy \tag{3}$$

– The *entropy* of a coupling:

$$H(\pi) = -\iint_{M \times M} \pi(x,y) \log(\pi(x,y)) dxd \tag{4}$$

– The *entropy regularized kernel of degree p* associated with the distance $d(\cdot,\cdot)$:

$$K_\gamma^1(x,y) = e^{-\frac{d(x,y)^p}{\gamma}}, \tag{5}$$

with γ the regularization parameter.

Finally, we define the *entropy-regularized 1-Wasserstein distance* and we derive a simpler formulation as in the paper:

$$\begin{aligned} W_{1,\gamma}(\mu_0,\mu_1) &= \inf_{\pi \in \Pi} [\iint_{M \times M} d(x,y)\pi(x,y)dxdy - \gamma H(\pi)] \\ &= \inf_{\pi \in \Pi} \gamma [\iint_{M \times M} -\log(K_\gamma)\pi(x,y) \\ &\quad + \pi(x,y)\log(\pi(x,y))dxdy] \\ &= \inf_{\pi \in \Pi} \gamma [\iint_{M \times M} \pi(x,y)\log(\frac{\pi(x,y)}{K(x,y)})xdy] \\ &= \gamma[1 + \min_{\pi \in \Pi} KL(\pi|K_\gamma)] \end{aligned}$$

The last line of calculation uses $\iint_{M \times M} \pi(x,y)dxdy = 1$.
While entropy regularization enables to smooth solutions, it mostly transforms the optimization problem in a strictly convex problem. It enables to use less computationally costly algorithms to solve the Optimal Transport problem such as the Sinkhorn-Knopp's method. Moreover, the Wasserstein distance can be reformulated using a kernel rather than directly the distances. The authors of the studied paper managed to leverage this new formulation by using kernels which don't need to be completely calculated. Indeed, we just need to know how to perform a convolution with these kernels. This is where the heat kernel comes into play since we know how to apply it without calculating it entirely.

Heat kernel

The previous formula requires to compute the geodesic distance between all pairs of points from the two marginal distributions. This can be very costly, especially in high dimensional spaces. Furthermore, sometimes geodesic distances can be complex to calculate precisely depending on the shape of the manifold which supports the distribution. In 2017, the heat method [Crane et al. 2017] solved to a certain extent this issue. Indeed, according to Varadhan's formula the geodesic distance on a manifold and the heat kernel are linked as follows: $d(x,y) = \lim_{t \to 0} \sqrt{-2t \ln H_t(x,y)}$, with t the time of heat diffusion and H_t the heat kernel. The heat kernel measures the heat transferred from x to destination y after time t. The intuition behind this behavior stems from the fact that heat diffusion can be modeled as a large collection of hot particles taking random walks starting at x: any particle that reaches a distant point y after a small time t has had little time to deviate from the shortest possible path. However, computing the geodesic distance from the heat kernel leads to a lot of numerical errors. In [Crane et al. 2017], the heat method is developed which enables to accurately apply the heat diffusion on a discretized spatial distribution. Then, it is possible to perform the equivalent of a kernel convolution with the heat kernel without computing all the distances. This results in even fewer computations.

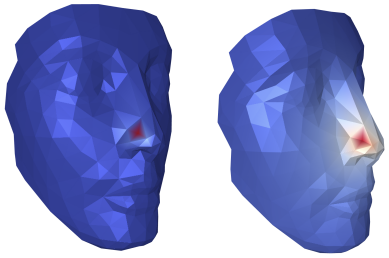

Figure 1: *Illustration of heat diffusion on a 3D shape. A dirac is applied to a mesh on the nose and then, diffusion is applied for $t = 0.01$ (left) and $t = 10$. Values go from 0 (in blue) to 1 (in red).*

The algorithm

Finally, the authors of the studied paper came up with a new method, more scalable, based on Sinkhorn-Knopp algorithm to accurately compute Wasserstein distances with faster itera-

tions. The algorithm presented in the paper is the following:

function CONVOLUTIONAL-WASSERSTEIN($\boldsymbol{\mu}_0, \boldsymbol{\mu}_1; \mathbf{H}_t, \mathbf{a}$)

// Sinkhorn iterations
$\mathbf{v}, \mathbf{w} \leftarrow \mathbf{1}$
for $i = 1, 2, 3, \ldots$
$\mathbf{v} \leftarrow \boldsymbol{\mu}_0 \oslash \mathbf{H}_t(\mathbf{a} \otimes \mathbf{w})$
$\mathbf{w} \leftarrow \boldsymbol{\mu}_1 \oslash \mathbf{H}_t(\mathbf{a} \otimes \mathbf{v})$

// KL *divergence*
return $\gamma\, \mathbf{a}^\top [(\boldsymbol{\mu}_0 \otimes \ln \mathbf{v}) + (\boldsymbol{\mu}_1 \otimes \ln \mathbf{w})]$

Figure 2: *Sinkhorn iterations for convolutional Wasserstein distances.* $\otimes$ *and* $\oslash$ *denote elementwise multiplication and division, respectively.*

4 Experiments

In the mathematical field of graph theory, the distance between two vertices in a graph is the number of edges in a shortest path (also called a graph geodesic) connecting them. This is also known as the geodesic distance or shortest-path distance. Is is worth noting that there may be more than one shortest path between two vertices. If there is no path connecting the two vertices, i.e., if they belong to different connected components, then conventionally the distance is defined as infinite. To avoid numerical issues due to the second point, we choose to test our algorithms on connected graphs which means that every pair of vertices in the graph is linked by a path (not necessarily by one edge). Furthermore, in order to ensure the symmetry of our geodesic distance ($d(u, v) = d(v, u)$), we use undirected graphs. To perform all our experiments on graphs we use *NetworkX* [2] library.
Studied graphs are either *m-ary trees* or random graphs generated according to Watts-Strogatz model. *m-ary trees* are arborescences in which each node has no more than m children. The Watts-Strogatz model is a random graph model that generates small-world networks. Introduced by Watts and Strogatz in 1998 [Watts and Strogatz 1998], the model starts with a regular lattice and introduces randomness by rewiring edges with a probability p. This process results in networks with a small average path length and high clustering coefficient, capturing the small-world phenomenon observed in various real-world systems. We set p relatively low to keep visibility but imposed the graph to be connected. Lastly, we used two types of distributions on nodes of graphs: uniform distributions on a fixed number of nodes (for clustered distributions mostly) and a Dirichlet distribution of varying size (to model spread distributions over most of the nodes of the graph). Our setting is summarized in Figure 3:

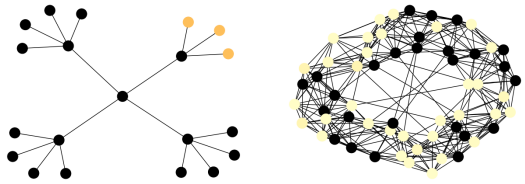

Figure 3: *A 4-rary tree of 20 nodes with a uniform distribution on* 3 *nodes and a randomly generated Watts-Strogatz graph of 60 nodes and with a parameter* $p = 0.1$ *with a Dirichlet distribution (the size of alpha is sampled uniformly at random between 1 and* N *the number of nodes). Values on nodes go from* 0 *(in black) to* 1 *(in red), as soon as the value on a node exceeds* 10^{-3} *the color goes from yellow to red.*

[2] https://networkx.org

4.1 Diffusion: Heat Kernel

Computation of kernels on graphs

As geodesic distance between two nodes on a graph can be calculated (shortest path length), the true geodesic distance kernels can be computed. The heat kernel can also be computed. However, we did not manage to obtain an accurate diffusion algorithm for graphs. We implemented one for 3D meshes which works (Figure 1) but didn't succeed in implementing one on graphs even by traducing in Python the MATLAB code of the studied paper. Maybe a special type of Laplacian matrix should be used. Nevertheless, we compare true kernel and heat kernels by coming back to the definition of heat kernels on graphs which is: $H_t = e^{-t*L}$, with L the Laplacian matrix of the graph ($= D - A$, with D the degree matrix and A the adjacency matrix) and t the diffusion time. Furthermore, small regularization provokes numerical issues during experiments. The smaller the regularization term, the more the kernel calculation is subject to numerical errors. As the $\frac{d(x,y)}{\gamma}$ term increases, the value of the kernel $e^{-\frac{d(x,y)}{\gamma}}$ becomes very low and thus very exposed to numerical errors.
First, we compare the entropy-regularized kernel of degree 1 (Equation 5) and the heat kernel. According to Varadhan's formula 3, for small t, these two matrices should be linked by the equation $\exp\left(\sqrt{-t * \log(H_t)}\right)$. The Frobenius distance between the true kernel of first degree and the previous quantity is computed as a function of t the time of diffusion in Figure 4. It can be observed that indeed the distance between the two matrices is relatively low close to $t = 0$ but begins to diverge slightly before t exceeds the regularization imposed in the true kernel. Therefore, parameters should be carefully set to avoid numerical issues in computing kernels, which requires high regularization, and to have a heat kernel close to the true kernel (which requires small time of diffusion).

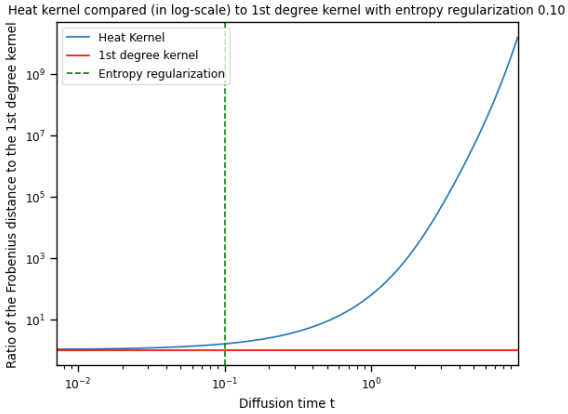

Figure 4: *Frobenius distance of* $\exp\left(\sqrt{-t * \log(H_t)}\right)$, *with* H_t *the heat kernel matrix, and the first degree regularized kernel matrix as a function of the time diffusion* t. *The ratio of the Frobenius matrix on the norm of first degree regularized kernel is computed. The graphs used are 4-rary trees with* 20 *nodes. The plot is in log scale. The entropy of the kernel is drawn in green dot line.*

Secondly, we do the same experiment but for the second degree kernel of distances. This time we compare directly H_t to K_γ^2. The dot green line illustrates the optimal time t of diffusion according to the studied paper. For this t we can obtain this approximation $K_\gamma^2 \underset{\gamma \to 0}{\backsimeq} H_{\frac{\gamma}{2}}$. However, we observe in Figure 5 that even at this t the error with the true kernel remains relatively high and that a t smaller than $\frac{\gamma}{2}$ would be even better.

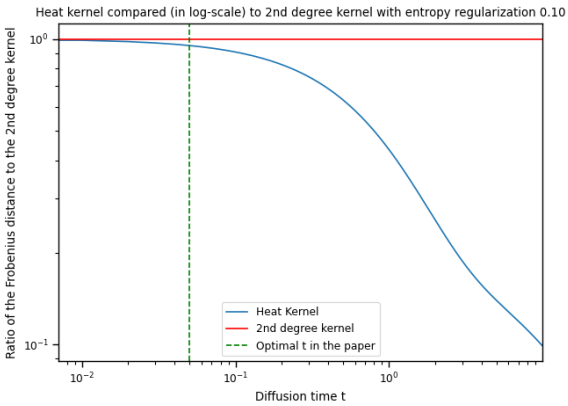

Figure 5: *Frobenius distance of H_t the heat kernel matrix, and the second degree regularized kernel matrix as a function of the time diffusion t. The ratio of the Frobenius matrix on the norm of second degree regularized kernel is computed. The graphs used are 4-rary trees with 20 nodes. The plot is in log scale. The entropy divided by two (optimal value of diffusion in the paper [Solomon et al. 2015]) of the kernel is drawn in green dot line.*

Impact of the heat kernel on the Wasserstein distance computation

We realize an analysis of the computation of the entropy-regularized order 2 Wasserstein distance. The convergence of the algorithm is studied in two configurations: one using the true distance kernel (2^{nd} degree) and the heat kernel.

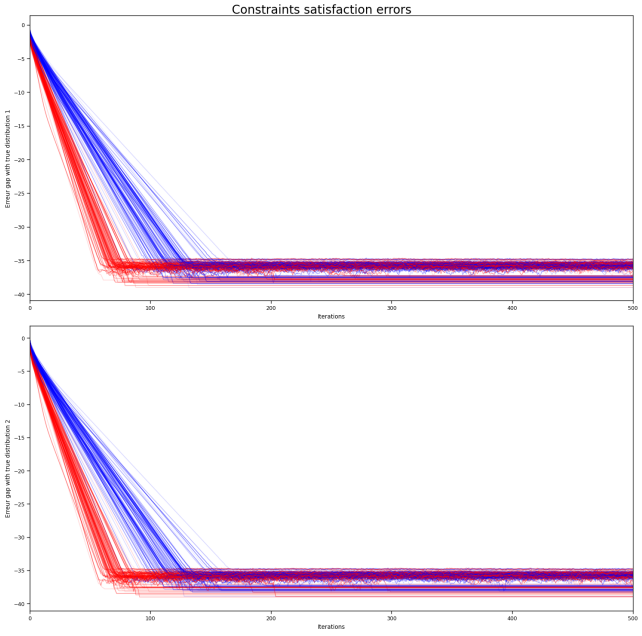

Figure 6: *Evolution of the constraints satisfaction error (logarithmic evolution), $\|\pi * \mathbb{1} - \mu_0\|$ and $\|\pi^T * \mathbb{1} - \mu_1\|$, for the true 2nd degree kernel (in red) and its approximation with heat diffusion (in blue). The convolutional Wasserstein algorithm is used to calculate the 2nd order Wasserstein distance of two Dirichlet distributions (with random uniformly sampled between 0 and 1 alpha parameters) on graphs generated with the Watts-Strogatz model. The experiment is run 100 time with a random generation of graphs with 60 nodes. The time of diffusion is set to $\gamma/2$ with the entropy regularization sets to $\gamma = 0.1$.*

To study the convergence of the algorithm, we first focus on the evolution of Wasserstein distance value as Sinkhorn iterations are performed. After the first iteration, updates of the Wasserstein distance calculated value are on the order of 10^{-7}. Another check for convergence can be studied: the convergence of the incompressibility constraints. As the optimal plan for transportation from μ_0 to μ_1 is updated, the euclidean distances $\|\pi * \mathbb{1} - \mu_0\|$ and $\|\pi^T * \mathbb{1} - \mu_1\|$ are computed. This test of convergence is realized for 100 randomly generated graph and Dirichlet distributions. These two plots (Figure 6) gives strong insight on the rate of convergence of the Sinkhorn-Knopp algorithm. Around 100 iterations, constraints are satisfied for both kernels. However, the Convolutional Wasserstein algorithm converges in more iterations than the Sinkhorn-Knopp algorithm with the true second degree kernel. In addition to only approximating geodesic distances, the heat kernel influences the speed of convergence of the Sinkhorn iterations.

Then, the dependency of the entropy-regularized order 2 Wasserstein distance on the entropy parameter is explored. For all these experiments we set the heat kernel to the optimal time of diffusion according to the paper $\gamma/2$. In Figure 7, the distance value for clustered uniform distributions on a small number of nodes is studied. It can be verified that the Wasserstein distance for a heat kernel with $t = \gamma/2 = 0$ is 0. This is due to the fact that with 0 diffusion we have no information about geodesic distances in the heat kernel and the heat kernel matrix is full of ones. Inversely, as γ goes close to 0 numerical errors become too significant in the computation of the true kernel which brings about some non-linearities. An interesting behavior is that the Wasserstein distance seems to decrease linearly in γ which is not obvious since there is also the dependency in γ with the kernel. Furthermore, distances computed with the two different kernels cross for values of $\gamma \in [0.4, 0.7]$. Randomness in distributions and graphs generation leads to this large range of values but it seems that the behavior is similar for most of the runs. This insight could contribute to the study of an optimal value of the entropy-regularization depending on the setting to minimize the approximation error with the heat kernel when calculating the Wasserstein distance.

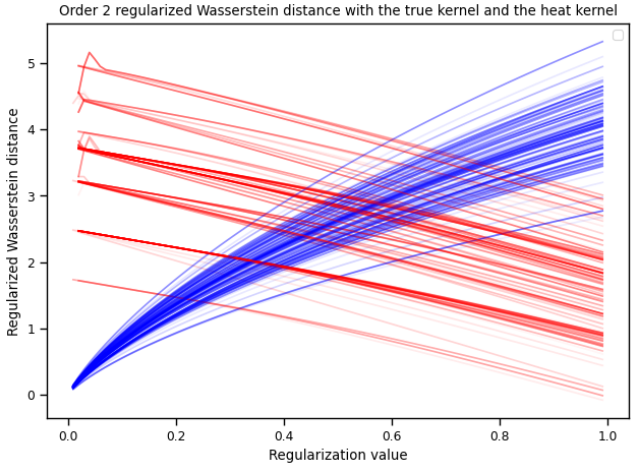

Figure 7: *Order 2 Wasserstein distance as a function of the regularization parameter for the true 2nd degree kernel (in red) and its approximation with heat diffusion (in blue). The convolutional Wasserstein algorithm is used to calculate the 2nd order Wasserstein distance of two uniform distributions on 4 nodes on graphs generated with the Watts-Strogatz model. The experiment is run 100 time with a random generation of graphs with 60 nodes. The time of diffusion is set to $\frac{\gamma}{2}$ with γ the entropy regularization.*

This intuition seems even more accurate when we see the following

graph, numerical issues lead to negative value in the Sinkhorn-Knopp algorithm so the value of the distance should not be taken directly. However, we see that for more spread distributions among nodes the different runs are more similar and have a behavior close to the one observed above.

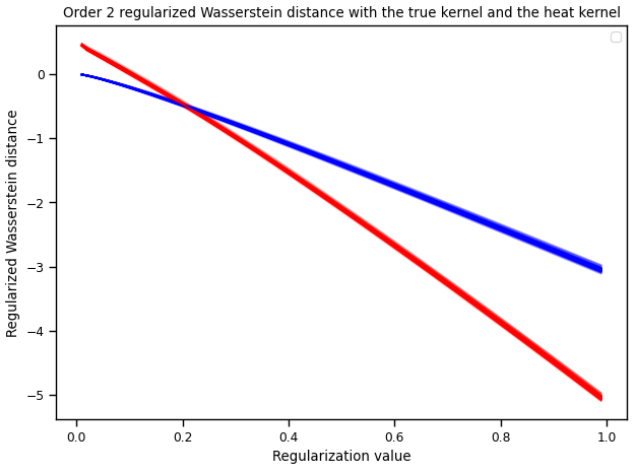

Figure 8: *Order 2 Wasserstein distance as a function of the regularization parameter for the true 2nd degree kernel (in red) and its approximation with heat diffusion (in blue). The convolutional Wasserstein algorithm is used to calculate the 2nd order Wasserstein distance of two Dirichlet distributions (with random uniformly sampled between* 0 *and* 1 *alpha parameters) on graphs generated with the Watts-Strogatz model. The experiment is run* 100 *time with a random generation of graphs with* 60 *nodes. The time of diffusion is set to* $\frac{\gamma}{2}$ *with* γ *the entropy regularization.*

4.2 Entropy regularization term

As we have seen before entropy regularization enables to solve a strictly convex problem. It smoothes solutions but also enables to reduce numerical errors in computation. However, this fuzziness can be a problem when one want to have a sharp distribution. Here, we test the impact on the sharpness of distributions by performing an interpolation between two diracs, Figure 9. The barycenter distribution of two dirac distributions on nodes taken far away in the graph is different depending on the entropy-regularization (γ) and the type of kernel used (heat kernel or ground-truth kernel). Indeed, high entropy-regularization leads to more diffuse distributions. By lowering this parameter a sharper distribution can be obtained. However, this parameter can't be too small otherwise numerical errors appear and the barycenter distribution becomes spurious. 19 additional nodes support the barycentric distribution in the high-entropy case compared to the low-entropy case. The same goes for the approximation with the heat kernel. Furthermore, using diffusion with the heat kernel to estimate geodesic distances generates more diffuse distributions. Indeed, at a same level of entropy-regularization the barycentric distribution is more diffuse with the heat kernel than with the ground-truth kernel. In the studied paper diffuse distributions are described as a result of the entropy regularization. However, the Convolutional Wasserstein algorithm as described in the paper intrinsically involves diffuse distribution. The entropy-regularization term enables to control the more or less diffuse nature of the barycentric distribution, but does not completely eliminate it. Experiments in the appendix 11 illustrate this diffuse barycenter distribution on images on which H_t corresponds to convolving with a Gaussian kernel.

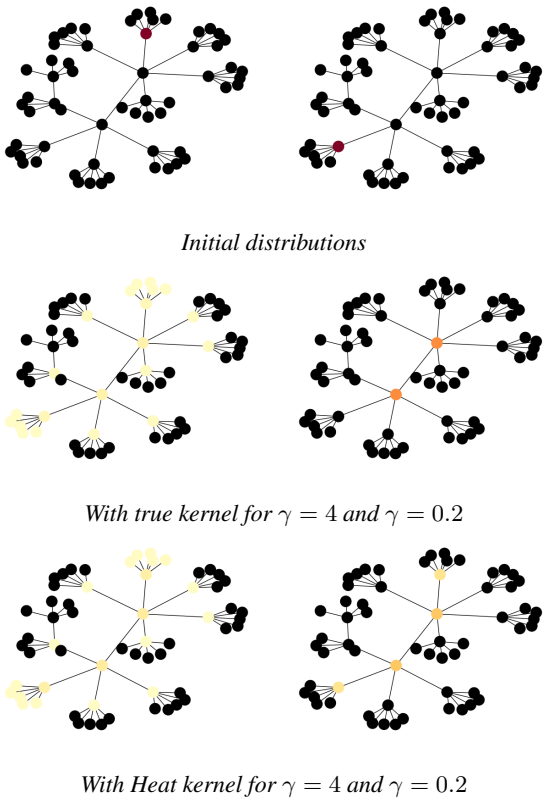

Figure 9: *First row: 5-ary tree with a two dirac distributions (red). Second row: barycenter distribution of the two diracs with a high entropy regularization* $\gamma = 4$ *(left) and with a low regularization* $\gamma = 0.2$ *(right) using the true second degree regularized kernel with Convolutional-Wasserstein barycenter algorithm. Third row: same experiment as second row but with heat kernels with a diffusion time* $t = \gamma/2$

5 Limitations and Conclusion

Convolutional Wasserstein Distances: Efficient Optimal Transportation on Geometric Domains [Solomon et al. 2015] proposes an efficient and scalable method for computing an accurate approximation of the entropy-regularized 2-Wasserstein distance. It leverages existing Optimal Transport (OT) algorithm, Sinkhorn-Knopp iterations, and advances in graphic, with the heat method for solving multiple-source shortest path problem on manifolds, to perform computationally intensive OT tasks. Various tasks can be performed using the Convolutional Wasserstein algorithm: shape interpolation, BRDF design, color histogram manipulation, soft mapping ...

Limits of the method

However, we underline that entropy regularization and the heat method cause several numerical approximations which could be detrimental to dealing with tasks requiring high precision. The optimal time of diffusion to approximate ground-truth kernels with heat kernel could be even smaller than $\frac{\gamma}{2}$ in order to minimize approximation errors. This value in the paper has been set to obtain this approximation $H_{\frac{\gamma}{2}} \approx K_\gamma^2$. This approximation is valid for $\gamma \to 0$. As γ too small leads to numerical errors and a small t enables a more accurate approximation, the method would be even more accurate for smaller t ($\frac{\gamma}{10}$ for instance). Furthermore, the heat method slows down convergence of the Sinkhorn iterations. For distributions supported on high dimensional spaces it could contribute to much longer algorithm run times.

Avenues for future work

Nevertheless, interesting phenomenology has been highlighted. Depending on the geometry of the problem distributions and manifolds, it empirically seems that an optimal entropy regularization value for calculating Wasserstein distance with heat method exists. Indeed, the entropy-regularized 2-Wasserstein distance and the approximation with heat method match around a specific value of γ. As written in the studied paper, $W^2_{2,H_{\frac{\gamma}{2}}}$ convergence as $\gamma \to 0$ has not been proven. However, an alternative could be to prove that for each problem there exist γ^* such that $W^2_{2,H_{\frac{\gamma}{2}}} = W^2_{2,K^2_\gamma}$ which would also prove the previous convergence. In practice, it has been shown that entropy regularization leads to more diffuse distributions (in the computation of a barycenter distribution for instance). We underline here that entropy-regularization is not the only source of diffused distribution but the heat method intrinsically generates diffuse distributions. We have not studied the entropic-sharpening method described in the paper so we don't know if this process would also limit the diffuseness due to heat method or only the one due to entropy-regularization.

To conclude, we have conducted an empirical study on the validity of the heat method for calculating Wasserstein distance to accomplish specific tasks in the field of Optimal Transport. While this analysis is not exhaustive, we hope to have provided an unbiased overview of the Convolutional Wasserstein method presented in the paper [Solomon et al. 2015].

References

BERTSEKAS, D. P. 1988. The auction algorithm: A distributed relaxation method for the assignment problem. *Annals of Operations Research 14*, 1, 105–123.

CRANE, K., WEISCHEDEL, C., AND WARDETZKY, M. 2017. The heat method for distance computation. *Commun. ACM 60*, 11 (Oct.), 90–99.

CUTURI, M., 2013. Sinkhorn distances: Lightspeed computation of optimal transportation distances.

KANTOROVICH, L. 2006. On the translocation of masses. *Journal of Mathematical Sciences 133*, 1381–1382.

SINKHORN, R. 1964. A relationship between arbitrary positive matrices and doubly stochastic matrices. *Annals of Mathematical Statistics 35*, 876–879.

SOLOMON, J., DE GOES, F., PEYRÉ, G., CUTURI, M., BUTSCHER, A., NGUYEN, A., DU, T., AND GUIBAS, L. 2015. Convolutional wasserstein distances. *ACM Transactions on Graphics 34*, 4, 66:1–66:11.

WATTS, D. J., AND STROGATZ, S. H. 1998. Collective dynamics of 'small-world'networks. *Nature 393*, 6684, 440–442.

Supplementary experiments

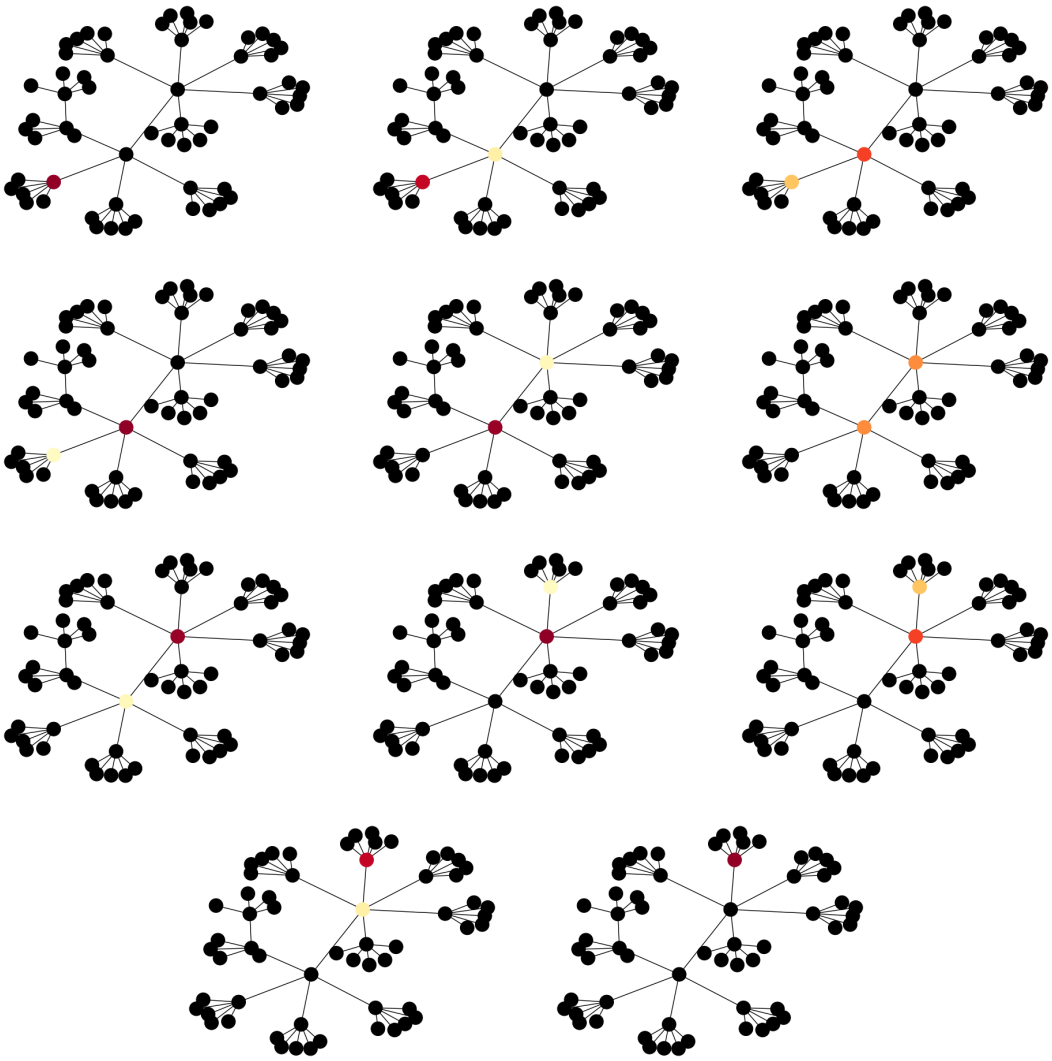

Figure 10: *Interpolated distributions from one dirac distribution to another with the true kernel and* $\gamma = 0.2$ *(left to right and up to down)*

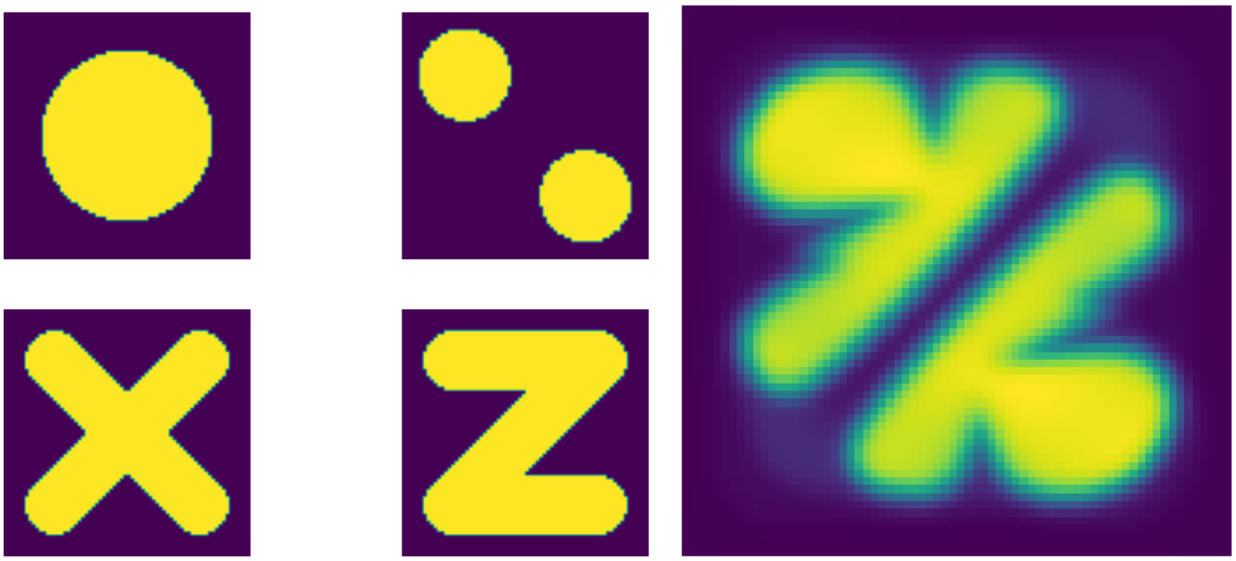

Figure 11: *Barycenter (right) of four images using Convolutional Wasserstein algorithm with* $\gamma = 0.0001$*(dataset from Computational Optimal Transport class of Gabriel Peyré)*