A Study of Optimal Transport in Deep Learning

Divyansh Khanna
Department of Computer Science
New York University
New York City, NY 100012
divyansh@nyu.edu

Sachin Shastri
Department of Computer Science
New York University
New York City, NY 100012
sds662@nyu.edu

Hans Krupakar
Department of Computer Science
New York University
New York City, NY 100012
hansk@nyu.edu

Abstract

In this report, we study the theory of Optimal Transport (OT) and its application to deep learning. OT is the study of allocation of resources between two distributions such that the overall cost is minimized. This finds use in deep learning as a way to model a distribution closer to the data distribution. OT is most commonly used as a way to compute a distance metric between distributions. Recently, Earth-Mover distance (EM) or Wasserstein-1 has brought the interest of the research community to OT based distance metrics for efficiently training generative models [1]. As a natural extension of the idea, OT has also been used for matching distributions. Optimal transport provides the tools to transform one distribution into another. This is particularly helpful in domain adaptation. A regularized optimal transportation model can be used to perform the alignment of the representations in the source and target domains [2] [3]. We touch upon gradient flows for the Wasserstein metric on the space of probability measures. This has been relevant in varied problems, like studying the global convergence of gradient descent [4] and reinforcement learning [5]. The report also covers the latest research on applications of OT based gradient flow techniques for natural language understanding [6]. OT provides a unique geometrical perspective along with optimum mass displacement which is usually lacking in other distance measures like Euclidean and Kullback–Leibler. We conclude by studying further research directions in the field and possible extensions to existing methods to further advance the application of OT in deep learning.

1 Introduction

Optimal transport (OT) is the study of allocation of resources between two distributions such that the overall cost is minimized. This finds use in deep learning as a way to model a distribution closer to the data distribution. Though the research on OT can be dated to early 1980s, we focus in particular on the recent wave of efficient algorithms that have helped OT find relevance in deep learning ([1], [2], [3]). In this paper, we discuss the fundamental of optimal transport and the concepts related to it in section 2. In section 3, we cover prominent applications of this field in deep learning. We start our exploration with generative models, covering GAN [7] and VAE [8]. Next, we discover the use of OT in the domain adaptation and sequence-to-sequence learning. We finish with the study of OT based theory on reinforcement learning.

2 Background on Optimal Transport

2.1 Preliminaries and Notations

We use calligraphic letters (i.e. $\mathcal{X}$) for sets, capital letters (i.e. X) for random variables, and lower case value (i.e. x) for their values. Probability distributions are represented using capital letters (i.e. $P(X)$) and corresponding densities with lower case letters (i.e. $p(x)$).

2.2 Primal Formulation

The problem of optimal transport was originally proposed by Monge in 1781. Consider a cost function $c : (x, y) \in \mathcal{X} \times \mathcal{Y} \mapsto c(x, y) \in \mathbb{R}^+$, and two random variables $X \sim \mu$ and $Y \sim \nu$ taking values in $\mathcal{X}$ and $\mathcal{Y}$ respectively. The Monge problem is formulated as finding a map $f : \mathcal{X} \to \mathcal{Y}$ which transports the mass from μ to ν minimizing the mass transportation cost,

$$\inf_f \ \mathbb{E}_{X\sim\mu}\left[c(X, f(Y)\right] \ \text{ subject to } \ f(X) \sim Y \tag{1}$$

To understand this mapping better, we can set up the map on discrete measures $\alpha = \sum_{i=1}^n \mathbf{a}_i \delta_{x_i}$ and $\beta = \sum_{j=1}^m \mathbf{b}_i \delta_{y_i}$, where δ_x is the Dirac at position x. The Monge problem seeks a map that associates to each point x_i a single point y_j and which must push the mass of α toward the mass of β. The map would be $T : x_1, ..., x_n \to y_1, ..., y_m$ such that,

$$\forall j \in (1, m), \ \ \mathbf{b}_j = \sum_{i:T(x_i)=y_j} \mathbf{a}_i \tag{2}$$

This 'forwarding' can be represented as a push forward operator $T_{\#}\mathcal{M}(\mathcal{X}) \to \mathcal{M}(\mathcal{Y})$, for a continuous map $T : \mathcal{X} \to \mathcal{Y}$. For discrete measures, this would be,

$$T_{\#}\alpha \stackrel{\text{def}}{=} \sum_i \mathbf{a}_i \delta_{T(x_i)} \tag{3}$$

Equation (2) can be shortened as $T_{\#}\alpha = \beta$. The minimization problem then becomes,

$$\min_T \left\{ \sum_i c(x_i, T(x_i)) \ : \ T_{\#}\alpha = \beta \right\} \tag{4}$$

Thought the formulation is elegant, it may not be always feasible to satisfy the constraint on the problem. For example, in the case of discrete measures, there doesn't exist an optimal Monge map when the target measure is supported on more points than the source measure. Also, solving the problem in its original combinatorial form may be computationally complex.

2.3 Kantorovich Relaxation and Dual Formulation

To remedy the challenges with the primal form, Kantorovich relaxed the Monge problem by casting problem (1) into a minimization over couplings $(X, Y) \sim \pi$ rather than the set of maps, where π should have marginals μ and ν,

$$\inf_\pi \ \mathbb{E}_{(X,Y)\sim\pi}\left[c(X, f(Y)\right] \ \text{ subject to } \ X \sim \mu, Y \sim \nu \tag{5}$$

This relaxation is crucial as it lets a point in μ be mapped to multiple points in ν, where as the Monge map allowed sending the whole mass to a unique location. This formulation becomes a linear program and can be solved using specialized solvers, although the computational complexity can be a hindrance for wide-scale adoption.

The Kantorovich duality of equation (5) has the following form,

$$\sup_{u \in C(\mathcal{X}), v \in C(\mathcal{Y})} \mathbb{E}_{(X,Y)\sim\mu\times\nu}[u(X) + v(Y)] \tag{6}$$

subject to $u(x) + u(y) \leq c(x, y)$ for all (x, y). An interesting case is when $(\mathcal{X}, d)$ is a metric space and $c(x, y) = d^p(x, y)$ for $p \geq 1$. In this case the infimum of equation (5), represented as W_c, has it's p-th root W_p known as the *p-Wasserstein* distance. When $c(x, y) = d(x, y)$ the following Kantorovich-Rubinstein duality holds,

$$W_1(\mu, \nu) = \sup_{f \in \mathcal{F}_L} \mathbb{E}_{X\sim\mu}[f(X)] - \mathbb{E}_{Y\sim\nu}[f(Y)] \tag{7}$$

where $\mathcal{F}_L$ is the class of all bounded 1-Lipschitz functions on $(\mathcal{X}, d)$.

2.4 Entropic Regularization

To speed up computation of the OT problem, Cuturi et al [9] proposed a way to regularize the OT problem. This is achieved by adding a negative entropy penalty *R* to the primal joint probability π of problem (5),

$$\inf_{\pi} \ \mathbb{E}_{(X,Y)\sim\pi}\ [c(X, Y)] + \epsilon\, R(\pi) \ \text{ subject to } \ X \sim \mu, Y \sim \nu \tag{8}$$

We will see in the following sections on how this regularization is particularly helpful in using the Sinkhorn algorithm. Additionally, research [10], [11] has used the negative entropic regularization in the dual form to get a more stable unconstrained maximization problem.

2.4.1 Sinkhorn Algorithm

Returning back to entropic regularization (8), Cuturi et al [12] showed an important result in the numerical calculation of the OT matrix. The work argues that this regularization is intuitive given the geometry of the optimal transportation problem. From an optimization point of view, this regularization turns the linear program into a strictly convex problem that can be solved extremely quickly with the Sinkhorn Knopp matrix scaling algorithm [13], [14].

Specifically, the Sinkhorn algorithm tries to solve the entropy regularized optimization problem,

$$\mathcal{L}(\mu, \nu) = \min_{\mathcal{T} \in \Pi(u,v)} \langle \mathcal{T}, C \rangle \ - \ \frac{1}{\epsilon} H(\mathcal{T}) \tag{9}$$

which is essentially a reformulation of (8) with $\mathcal{T}$ being then joint probability matrix and $\epsilon > 0$ is the regularization strength. This result is extremely important as it allows parallel computation of the solution, and opens the doors for application of OT based methods on high dimensional data.

2.5 Wasserstein Gradient Flows

An important topic in the field of optimization is gradient flows. The key idea of a gradient flow is to minimize a given function $F : \mathbb{R}^n \to \mathbb{R}$ defined on a vector space X by starting with an initial point x0, and solving equations of the form $x(t) = \Delta F(x(t)))$ to minimize F in as few steps as possible. [15] explains gradient flow in the Euclidean space, shows ways to generalize it in the metric space and relates it to optimal transport by bringing it to the probability measure space. In the Euclidean space, the simplest case is when F is differentiable. Then this boils down to a standard Cauchy problem with unique solution if ΔF is Lipschitz continuous. A standard numerical solution to this is to use Minimizing Movement Scheme (MMS) [16] which given point x_k iteratively obtains the next sequence of points x_{k+1} for small steps along the gradient of F.

Gradient flow can be extended by bringing it to the probability measure space, denoted by $P(\Omega)$ with $\Omega \subset \mathbb{R}^d$. In Wasserstein Gradient Flow, we endow a Riemannian geometry on $P(\Omega)$ where the distance between two elements is given by 2nd order Wasserstein distance -

$$W_2^2(\mu, \upsilon) = \inf_T \int_\Omega c(x, T(x)) d\mu(x) \tag{10}$$

Where T is a mapping $\mathbb{R}^d \to \mathbb{R}^d$ pushing marginal μ onto marginal υ satisfying $T_{\#\mu} = \upsilon$, c is the cost function associated with transporting x in μ to y in υ. If $\{\mu_\tau\}_{\tau \in [0,1]}$ is an absolutely continuous curve in $P(\Omega)$ with finite second-order moments, [17] shows that by considering W_2^2 for u_τ and $u_{\tau+h}$ i.e considering $\upsilon_\tau(x) = \lim_{h \to 0}(T(x_\tau) - x_\tau)/h$ as the velocity of the particle, then a gradient flow can be described on $P(\Omega)$.

This has been used in multiple applications including policy optimization in reinforcement learning and global convergence of gradient descent.

3 Applications

3.1 Generative models

Generative modelling is a major sub-field of machine learning that studies the problem of how to learn models that generate images, audio or text. The primary advantage of generative modelling is that it can be trained on unlabelled data, which is almost endlessly available. The central problem in generative modelling is training the generative model such that the distribution of the generated data matches the distribution of the training data. There are many successful generative models which try to solve this problem either using auto-encoders such as VAE [8]), or *generator* and *discriminator* pair such as GANs [7].

The Optimal Transport cost ([18], [19]) is a way to measure a distance between probability distributions and provides a much weaker topology than many others, including $f-$divergences associated with the original GAN algorithms [20]. This is particularly important in applications where data is usually supported on low dimensional manifolds in the input space $\mathcal{X}$.

3.1.1 GAN based techniques

The *discriminator* or *critic* based techniques are particularly powerful as they define a distance between the model distribution and the data distribution which the generative model can optimize to produce data that more closely resembles the training data. Optimal Transport theory provides a related approach to measuring such a distance. Framed differently, the problem of optimally transporting one set of data points to another represents an alternate method of specifying a metric over probability distributions.

The first introduction of Optimal Transport in generative models came in the work of Arjovky et al.s [1] which re-interpreted GANs as a metric minimization problem. They proposed the *Earth-Mover distance* or *Wasserstein-1 distance* as a good objective for generative modelling,

$$D_{EMD}(p, g) = \inf_{\gamma \in \Pi(p,g)} \mathbb{E}_{x,y \sim \gamma} c(x, y) \tag{11}$$

As is the trend, this is another reformulation of the primal form of OT we have seen before. Here, $\Pi(p, g)$ is the set of all distributions $\gamma(x, y)$ with marginals $p(x)$, $g(y)$, and $c(x, y)$ is a cost function that Arjovsky et al. [1] take as Euclidean distance. Since solving over γ is generally intractable, their proposed model, *Wasserstein GAN* use the dual formulation of OT,

$$D_{EMD}(p, g) = \sup_{||f||_L \leq 1} \mathbb{E}_{x \sim p} f(x) - \mathbb{E}_{y \sim g} f(y) \tag{12}$$

This formulation can be solved by approximating using a class of neural network discriminators coupled with gradient clipping to account for the 1-Lipschitz constraint. There has been a follow-up work by Gulrajani et al. [21] suggesting another way to bound the gradients for critics. These class

Algorithm 1 Optimal Transport GAN (OT-GAN) training algorithm with step size α, using mini-batch SGD for simplicity

Require: n_{gen}, the number of iterations of the generator per critic iteration
Require: η_0, initial critic parameters. θ_0, initial generator parameters
1: **for** $t = 1$ to N **do**
2: Sample $\mathbf{X}, \mathbf{X}'$ two independent mini-batches from real data, and $\mathbf{Y}, \mathbf{Y}'$ two independent mini-batches from the generated samples
3: $\mathcal{L} = \mathcal{W}_c(\mathbf{X}, \mathbf{Y}) + \mathcal{W}_c(\mathbf{X}, \mathbf{Y}') + \mathcal{W}_c(\mathbf{X}', \mathbf{Y}) + \mathcal{W}_c(\mathbf{X}', \mathbf{Y}') - 2\,\mathcal{W}_c(\mathbf{X}, \mathbf{X}') - 2\,\mathcal{W}_c(\mathbf{Y}, \mathbf{Y}')$
4: **if** $t \bmod n_{gen} + 1 = 0$ **then**
5: $\eta \leftarrow \eta + \alpha \cdot \nabla_\eta \mathcal{L}$
6: **else**
7: $\theta \leftarrow \theta - \alpha \cdot \nabla_\theta \mathcal{L}$
8: **end if**
9: **end for**

of GANs are the first which try to solve the mode collapse problem which is pretty evident in most classical GAN architectures. In most GAN based networks, the model is able to learn only certain modes and hence not able to generalize properly. The authors of WGAN report that no experiment suffered from mode collapse. We refer the reader to Bousquet et al. [22] and Genevay et al. [9] which further explore the connection between GANs and the dual form of optimal transport.

Another approach to generative modelling is lead by research trying to solve for a close approximate to the primal formulation of optimal transport. A foundational work to this end is by Genevay et al. [23] which use an entropic regularization to the *Earth-Mover distance*, called the *Sinkhorn distance* [12], where the set of allowed joint distribution Π_β is now restricted to distributions with entropy of at least some constant β. The distance is evaluated on mini-batches of data X, Y, with the coupling distribution γ replaced by matrix M of *soft matchings* between the elements of X, Y. The resulting distance, evaluated on a minibatch, is then

$$\mathcal{W}_c(X, Y) = \inf_M Tr[MC^T] \tag{13}$$

This approximation method has the benefit of being calculated efficiently on a GPU using the earlier discussed Sinkhorn algorithm. This method is known as *Sinkhorn AutoDiff*.

The recent work of Salimans et al. ([24]) explores the idea of defining distance functions between distributions over *mini-batches* of data. They propose a new distance over mini-batch distributions, called *Mini-batch Energy Distance* combining optimal transport in primal form with an energy distance defined in an adversarially learned feature space. The distance between mini-batches are measured using the entropy-regularized Wasserstein distance, or *Sinkhorn* distance, as defined for mini-batches in equation (12).

$$D^2_{MED}(p, g) = 2\mathbb{E}[\mathcal{W}_c(X, Y)] - \mathbb{E}[\mathcal{W}_c(X, X')] - \mathbb{E}[\mathcal{W}_c(Y, Y')] \tag{14}$$

where X, X' are independently sampled mini-batches from distribution p and Y, Y' are independent mini-batches from g. The authors propose learning the transport cost function $c(x, y)$ adversarially, so that it can adapt to the generator distribution g and become more discriminative. The work defines c to be the cosine distance between vectors $\nu_\eta(x)$ and $\nu_\eta(y)$, where ν_η is a deep neural network that maps the images in the mini-batch into a learned latent space. This algorithm is called OT-GAN and is described above.

3.1.2 Auto-encoders

In the previous section we covered OT based GAN techniques. Now, we take a look at another very popular sub-field of generative models, probabilistic autoencoders. Primarily, we focus our review to the recently proposed Wasserstein Auto-encoders [25], which are conceptually similar to VAE [8], but make use of optimal transport based metric to measure the distance between probability distributions.

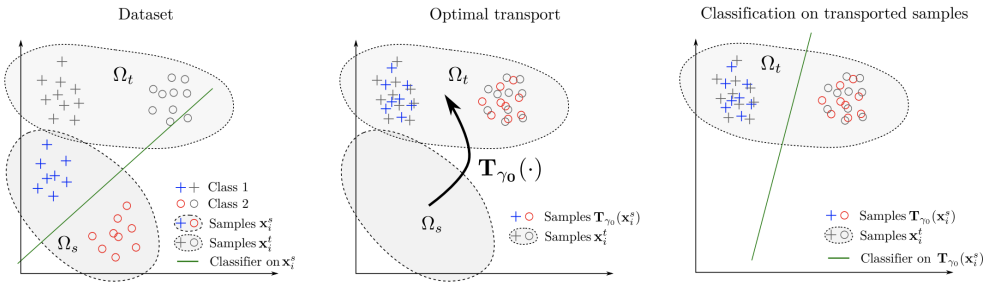

Figure 1: Transportation map T translates data from source domain to target domain.

The work aims at minimizing $\mathcal{W}_c(P_X, P_G)$ between the true data distribution P_X and a latent variable model P_G specified by the prior distribution P_Z of latent codes $Z \in \mathcal{Z}$ and the generative model $P_G(X|Z)$ of the data points $X \in \mathcal{X}$ given Z. Just like the VAE, this method also consists of two terms; the ground cost c based reconstruction cost and a regularizer $\mathcal{D}(P_Z, P(G))$ penalizing a discrepancy between two distributions in $\mathcal{Z}$: P_Z and a distribution of encoded data points $Q_Z = \mathbb{E}_{P_X}[Q(Z|X)]$. An important point to note here is this method is agnostic of the cost function c. The paper highlights that this method is equivalent to adversarial auto-encoders [26], for a squared cost function and $\mathcal{D}_Z$ as the GAN objective.

Together with the reconstruction term coming from the primal of the optimal transport formulation, we get what looks like the components of a generative model like the VAE – a reconstruction term, plus, a regularizer term. The regularizer gives the model its generative characteristics, in that without it we would get a regular auto-encoder which will know how to reconstruct input, but will have 'holes' in $\mathcal{Z}$ in those places that don't have training data. In other words, we won't be able to draw from a latent representation. This finally leads us to the WAE objective for a given map $G : \mathcal{Z} \to \mathcal{X}$:

$$\mathcal{D}_{WAE}(P_x, P_G) = \inf_{Q(Z|X) \in Q} \mathbb{E}_{P_X} \mathbb{E}_{Q(Z|X)}[c(X, G(Z))] + \lambda \, \mathcal{D}_Z(Q_Z, P_Z) \tag{15}$$

where Q is any set of probabilistic encoders, D_Z is any divergence between Q_Z and P_Z. The authors highly recommend the reader to the original paper Tolstikhin et al. [25] and Bousquet et al. [27].

3.2 Domain Adaptation

In practical applications, we often find our data to be comprised of multiple distributions. Discrepancies in data distribution are due to several reasons and are application dependent. This problem is often referred to as domain adaptation, where the use is challenged with transferring information from one source (domain) to another, with minimal loss and high consistency.

The domain adaptation problem boils down to finding a transformation of the input data matching the source and target distributions, and learning a new classifier from the transformed source samples. Optimal Transport distance make a strong case for such problems because they can be evaluated directly on empirical estimates of the distributions, and they exploit the geometry of the underlying metric space.

A common strategy to tackle unsupervised domain adaptation is to propose methods that aim at finding representations in which domains match in some sense ([2], [3]). Regularization is key in solving OT linear problems, as it induces some properties of the solution; and importantly reduces overfitting. The entropic regularization [12] is crucial since most elements of the transport should be zero with high probability, one can look for a smoother version of the transport, thus lowering its sparsity by increasing its entropy. As the parameter controlling the entropic regularization term increases, the sparsity of the plan decreases and source points tend to distribute their probability masses toward more target points.

Adding on the prior work on domain adaptation, new work of Sequy et al. [11] presents a novel two step approach for the fundamental problem of learning an optimal map from one distribution to another. First, the authors propose an algorithm to compute the optimal transport plan using dual

stochastic gradient for solving regularized dual form of OT. Second, they learn an optimal map (Monge map) as a neural network by approximating the projection of the OT plan obtained in the first step. Parameterizing using a neural network allows efficient learning and provides generalization outside the support of the input measure. This work provides strong theoretical background for learning better projections.

3.3 Sequence Learning

Generative models use optimal transport to define metrics with weaker topology to try and learn an efficient way to transport distance between two distributions. With recent work [6], we can see the use of optimal transport in sequence models to alleviate the issues of maximum likelihood estimation.

The research presents a novel Seq2Seq learning scheme that leverages OT to construct sequence-level loss. The objective aims to find an optimal matching (see figure below) of similar words and phrases between two sequences. The OT loss allows end-to-end supervised training and acts as an effective sequence-level regularization to the MLE loss.

The use of OT is motivated by the use of *soft matchings* to capture the semantic similarity between words. The traditional method to match the 'key words' between synthesized and reference sequences (in machine translation tasks) is constrained because two different words can be close to each other in semantic space. The authors propose a *soft bipartite matching* loss $\mathcal{L}_{SBM}$. Specifically, $\mathcal{L}_{SBM} = \sum_k c(w_{i_k}, w_{j_k})$, for $k \in [1, K]$ and $K \leq n$. The ground cost used is cosine distance $c(x, y) = 1 - \frac{x^T y}{||x||_2 \, ||y||_2}$, where x and y are word embedding vectors.

The authors use the Inexact Proximal Point method for Optimal Transport (IPOT) algorithm [28] to solve the OT optimization problem. Though the Sinkhorn algorithm can be used, IPOT was empirically found to be better and less sensitive to regularization hyper-parameter. The loss function for the model is,

$$\mathcal{L} = \mathcal{L}_{MLE} + \gamma_1 \mathcal{L}_{copy} + \gamma_2 \mathcal{L}_{seq} \tag{16}$$

where $\mathcal{L}_{MLE}$ is the standard maximum likelihood loss, $\mathcal{L}_{copy}$ is feature matching OT loss between source and target sequence embeddings, and $\mathcal{L}_{seq}$ is the sequence level OT matching loss between the ground-truth and model prediction. The essential idea here is to use the OT losses as a regularizer to the MLE loss. Importantly, the research shows that this formulation can be interpreted Wasserstein gradient flows (introduced in earlier section).

3.4 Reinforcement Learning

As mentioned in 2.5, a direct application of Wasserstein Gradient flow can be in finding an optimal policy in reinforcement learning. The general problem setting of reinforcement learning involves an agent picking an action $a \in A$ conditioned on a state variable $s \in S$ using a conditional distribution policy $\pi(a|s)$. $P_s(s'|s, a)$ is the transition probability of going to a new state s' and r(s,a) is the immediate reward of this action. The goal is to obtain an optimal policy that maximizes the total expected reward -

$$J(\pi) = \mathbb{E}_{s \sim \rho_\pi, a \sim \pi}[r(s, a)] \tag{17}$$

where $\rho_\pi = \sum_{t=1}^{\infty} \gamma^{t-1} P_r(s_t = s)$, $\gamma \in [0, 1]$ is the discount factor regularizing future rewards, and $P_r(s)$ is the state marginal distribution induced by π.

[5] shows we can use Wasserstein gradient flow by considering the policy to form a Riemannian manifold on the space of probability measures characterized by an energy functional. The policy can be optimized indirectly, by treating the uncertainty of a policy as parameter distributions or directly by defining gradient flow over actions.

In the indirect case, the policy π is parameterized by θ and we learn it's posterior distribution $p(\theta)$ in response to the expected total reward. The objective function is defined as -

$$max_P\{\mathbb{E}_{p(\theta)}[J(\pi_\theta)] - \alpha KL(p||p_0)\} \tag{18}$$

where $p_0(\theta)$ is the prior of θ, α is the temperature hyper-parameter to balance exploitation and exploration in the policy.

[29] shows that by taking the derivative of the objective function, the optimal distribution has a simple closed form of $p(\theta) \propto \exp(J(\pi_\theta)/\alpha)$ similar to a Bayesian formulation of θ. The posterior distribution for θ denoted by $\mu(\theta)$ is learned by solving it as a gradient flow problem. The energy functional characterizing the similarity between the current parameter distribution and true distribution induced by the total reward is then defined as :

$$F(\mu) := -\int J(\pi_\theta)\mu(\theta)d\theta + \int \mu(\theta)\log\mu(\theta)d\theta = KL(\mu||p_\theta) \tag{19}$$

Zhang et al. [5] propose that 19 converges to p_θ in the infinite time limit.

For the direct case, we formulate the problem as policy-distribution based gradient flows. Here the energy functional has to depend on the learned policy π and hence even the states. For that, we define functional Q where

$$Q(a_t, s_t) := r(a_t = a, s_t = s) + \mathbb{E}_{(s_{t+1}, a_{t+1}, \ldots)\sim(p_\pi,\pi)} \sum_{l=1}^{\infty} \gamma^l r(s_{t+l}, a_{t+l}) \tag{20}$$

By integrating out the action a, we can define an energy functional that indicates the similarity between the current and the optimal policy, $p_{s,\pi}(a|s) \propto \exp^{Q(a,s)}$, by -

$$F_s(\pi) := -\int Q(a, s)\pi(a|s)da + \int \pi(a|s)log\pi(a|s)da = KL(\pi||p_{s,\pi}) \tag{21}$$

Authors of [5] again propose that WGF with 21 converges to an optimal policy with Q(a,s) satisfying modified Bellman equation,

$$Q(a_t, s_t) = r(a_t, s_t) + \gamma\mathbb{E}_{s_{t+1}\sim p_\pi}[log \int_A \exp(Q(a, s_{t+1}))da] \tag{22}$$

Both the direct-policy and indirect-policy learning energy functional equations can be solved to convergence using the Jordan-Kinderlehrer-Otto (JKO) scheme [30] which is beyond the scope of this paper.

Zhaang et al. [5] show that Wasserstein gradient flow with direct policy learning solved with the JKO scheme was shown to outperform various other algorithms such as TRPO-GAE [31] and DDPG [32] with respect to average returns over episodes on different MuJoCo tasks in OpenAI rllab and Gym [33].

3.5 Other applications

Bach et al. [4] describe a method that utilizes Wasserstein gradient flow in minimizing the convex function of a measure. They show that by proper initialization and using many-particle limit, the gradient flow they obtain by discretizing the measure to a mixture of particles and using continuous-time gradient descent is non-convex but can still converge to a global minimum. Some of the application of this is in sparse spikes deconvolution and in training a neural network with a single hidden layer.

4 Conclusion

In this survey paper, we have explored the theory of optimal transport, topics associated with it, and how these topics do or potentially can play a role in the different areas of deep learning. The theory of optimal transport provides an elegant approach to moving information across distributions. This provides an open platform for research and application to deep learning. With its strong theoretical foundations, we have seen recent work exploiting application specific regularization techniques successfully. The research community is actively working on better modelling techniques to overcome the computational shortcomings of the OT problem.

References

[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017.

[2] Nicolas Courty, Rémi Flamary, Devis Tuia, and Alain Rakotomamonjy. Optimal transport for domain adaptation. *CoRR*, abs/1507.00504, 2015.

[3] Nicolas Courty, Rémi Flamary, Amaury Habrard, and Alain Rakotomamonjy. Joint distribution optimal transportation for domain adaptation. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 3730–3739. Curran Associates, Inc., 2017.

[4] Lenaic Chizat and Francis Bach. On the global convergence of gradient descent for over-parameterized models using optimal transport. In *Advances in neural information processing systems*, pages 3040–3050, 2018.

[5] Ruiyi Zhang, Changyou Chen, Chunyuan Li, and Lawrence Carin. Policy optimization as wasserstein gradient flows. *CoRR*, abs/1808.03030, 2018.

[6] Liqun Chen, Yizhe Zhang, Ruiyi Zhang, Chenyang Tao, Zhe Gan, Haichao Zhang, Bai Li, Dinghan Shen, Changyou Chen, and Lawrence Carin. Improving sequence-to-sequence learning via optimal transport. *CoRR*, abs/1901.06283, 2019.

[7] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, pages 2672–2680, Cambridge, MA, USA, 2014. MIT Press.

[8] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[9] Aude Genevay, Gabriel Peyré, and Marco Cuturi. Gan and vae from an optimal transport point of view. *arXiv preprint arXiv:1706.01807*, 2017.

[10] Mathieu Blondel, Vivien Seguy, and Antoine Rolet. Smooth and sparse optimal transport. *arXiv preprint arXiv:1710.06276*, 2017.

[11] Vivien Seguy, Bharath Bhushan Damodaran, Remi Flamary, Nicolas Courty, Antoine Rolet, and Mathieu Blondel. Large scale optimal transport and mapping estimation. In *International Conference on Learning Representations*, 2018.

[12] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in neural information processing systems*, pages 2292–2300, 2013.

[13] Richard Sinkhorn and Paul Knopp. Concerning nonnegative matrices and doubly stochastic matrices. *Pacific Journal of Mathematics*, 21(2):343–348, 1967.

[14] Philip A Knight. The sinkhorn–knopp algorithm: convergence and applications. *SIAM Journal on Matrix Analysis and Applications*, 30(1):261–275, 2008.

[15] Filippo Santambrogio. {Euclidean, metric, and Wasserstein} gradient flows: an overview. *Bulletin of Mathematical Sciences*, 7(1):87–154, 2017.

[16] Massimo Gobbino. Minimizing movements and evolution problems in euclidean spaces. *Annali di Matematica Pura ed Applicata*, 176:29–48, 01 1999.

[17] Gigli N. Ambrosio L. and Savaré G. Gradient flows in metric spaces and in the space of probability measures. 2005.

[18] Cédric Villani. *Topics in optimal transportation*. Number 58. American Mathematical Soc., 2003.

[19] Gabriel Peyré and Marco Cutur. *Computational Optimal Transport*. ArXiv:1803.00567, 2018.

[20] Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-gan: Training generative neural samplers using variational divergence minimization. In *Advances in neural information processing systems*, pages 271–279, 2016.

[21] Ishaan Gulrajani, Faruk Ahmed, Martín Arjovsky, Vincent Dumoulin, and Aaron C. Courville. Improved training of wasserstein gans. *CoRR*, abs/1704.00028, 2017.

[22] Olivier Bousquet, Sylvain Gelly, Ilya Tolstikhin, Carl-Johann Simon-Gabriel, and Bernhard Schoelkopf. From optimal transport to generative modeling: the vegan cookbook. *arXiv preprint arXiv:1705.07642*, 2017.

[23] Aude Genevay, Gabriel Peyré, and Marco Cuturi. Sinkhorn-autodiff: Tractable wasserstein learning of generative models. 2017.

[24] Tim Salimans, Han Zhang, Alec Radford, and Dimitris N. Metaxas. Improving gans using optimal transport. *CoRR*, abs/1803.05573, 2018.

[25] Ilya Tolstikhin, Olivier Bousquet, Sylvain Gelly, and Bernhard Schoelkopf. Wasserstein auto-encoders. *arXiv preprint arXiv:1711.01558*, 2017.

[26] Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, and Ian J. Goodfellow. Adversarial autoencoders. *CoRR*, abs/1511.05644, 2015.

[27] Olivier Bousquet, Sylvain Gelly, Ilya Tolstikhin, Carl-Johann Simon-Gabriel, and Bernhard Schoelkopf. From optimal transport to generative modeling: the vegan cookbook. 2017. *URL http://arxiv. org/abs/1705.07642*.

[28] Yujia Xie, Xiangfeng Wang, Ruijia Wang, and Hongyuan Zha. A fast proximal point method for computing wasserstein distance. *arXiv preprint arXiv:1802.04307*, 2018.

[29] Yang Liu, Prajit Ramachandran, Qiang Liu, and Jian Peng. Stein variational policy gradient. *arXiv preprint arXiv:1704.02399*, 2017.

[30] Richard Jordan, David Kinderlehrer, and Felix Otto. The variational formulation of the fokker-planck equation. *SIAM Journal on Mathematical Analysis*, 29, 04 2000.

[31] John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015.

[32] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

[33] Yan Duan, Xi Chen, Rein Houthooft, John Schulman, and Pieter Abbeel. Benchmarking deep reinforcement learning for continuous control. *CoRR*, abs/1604.06778, 2016.