algorithms

Article

Accelerating the Sinkhorn Algorithm for Sparse Multi-Marginal Optimal Transport via Fast Fourier Transforms

Fatima Antarou Ba *, Michael Quellmalz

Institute of Mathematics, Technische Universität Berlin, Straße des 17. Juni 136, 10623 Berlin, Germany
* Correspondence: fatimaba@math.tu-berlin.de

Abstract: We consider the numerical solution of the discrete multi-marginal optimal transport (MOT) by means of the Sinkhorn algorithm. In general, the Sinkhorn algorithm suffers from the curse of dimensionality with respect to the number of marginals. If the MOT cost function decouples according to a tree or circle, its complexity is linear in the number of marginal measures. In this case, we speed up the convolution with the radial kernel required in the Sinkhorn algorithm via non-uniform fast Fourier methods. Each step of the proposed accelerated Sinkhorn algorithm with a tree-structured cost function has a complexity of $\mathcal{O}(KN)$ instead of the classical $\mathcal{O}(KN^2)$ for straightforward matrix–vector operations, where K is the number of marginals and each marginal measure is supported on, at most, N points. In the case of a circle-structured cost function, the complexity improves from $\mathcal{O}(KN^3)$ to $\mathcal{O}(KN^2)$. This is confirmed through numerical experiments.

Keywords: multi-marginal optimal transport; Sinkhorn algorithm; fast Fourier transform; image processing; optimal transport; Wasserstein barycenter

MSC: 49Q22; 65T50; 65D18; 49M20

Citation: Ba, F.A.; Quellmalz, M. Accelerating the Sinkhorn Algorithm for Sparse Multi-Marginal Optimal Transport via Fast Fourier Transforms. *Algorithms* **2022**, *15*, 311. https://doi.org/10.3390/a15090311

Academic Editor: Francisco Saldanha da Gama

Received: 5 August 2022
Accepted: 26 August 2022
Published: 30 August 2022

1. Introduction

The optimal transport (OT) problem is an optimization problem that deals with the search for an optimal map (plan) that moves masses between two or more measures at low cost [1,2]. OT appears in a wide range of applications such as image and signal processing [3–7], economics [8,9], finance [10,11], and physics [12,13]. The OT problem was first introduced in 1781 by Monge. His objective was to find a map between two probability measures μ^1, μ^2 on $\mathbb{R}^d$ that transports μ^1 to μ^2 with minimal cost, where the cost function describes the cost of transporting a mass between two points in $\mathbb{R}^d$. However, such maps do not always exist, so that Kantorovich [14] relaxed the problem in 1942 by looking for a transport plan with two prescribed marginals μ^1 and μ^2 that minimizes a certain cost functional.

Several authors have generalized the formulation to multi-marginal optimal transport (MOT) [15–17], where more than two marginal measures are given. For the given probability measures μ^k on $\Omega^k \subset \mathbb{R}^d$, $k = 1, \dots, K$, an optimal transport plan π is defined as a solution of the MOT problem

$$\min_{\pi \in \Pi(\mu^1,\dots,\mu^k)} \int_{\Omega^1 \times \dots \times \Omega^K} c(x^1, \dots, x^K) \, \mathrm{d}\pi(x^1, \dots, x^K)$$

where $\Pi(\mu^1, \cdots, \mu^K)$ is the convex set of all joint probability measures π whose marginals are μ^k, and $c \colon \Omega^1 \times \dots \times \Omega^K \to \mathbb{R}$ is the cost function.

Since the numerical computation of a transport plan is difficult in general, a regularization term such as the entropy [13,18], Kullback–Leibler divergence [19], general f-divergence [20], or L^2-regularization [21,22] can be added to make the problem strictly convex. Different approaches such as the Sinkhorn algorithm [1,13], stochastic gradient descent [23], the Gauss–Seidel method [22], or proximal splitting [24] have been used to iteratively determine a minimizing sequence of the MOT problem.

However, the problem suffers from the curse of dimensionality as the complexity grows exponentially with the number K of marginal measures. One way to circumvent this lies in incorporating additional sparsity assumptions into the model. Polynomial-time algorithms to solve certain sparse MOT problems have been studied in [13,25,26]. We will assume that the cost function decouples according to a graph, where the nodes correspond to the marginals and the cost function is the sum of functions that depend only on two variables which correspond to two marginals connected by an edge of the graph. For example, the circle-structured Euclidean cost function reads as

$$c(x^1, \dots, x^K) = \left\| x^1 - x^2 \right\|_2^2 + \cdots + \left\| x^{K-1} - x^K \right\|_2^2 + \left\| x^K - x^1 \right\|_2^2, \quad x^1, \dots, x^K \in \mathbb{R}^d.$$

MOT problems with graph-structured cost functions with a constant tree-width can be solved with the Sinkhorn algorithm in polynomial time [27]. In [25], polynomial-time algorithms were presented for the MOT problem and its regularized counterpart for the cases of a graph structure and a set-optimization structure, as well as a low-rank and sparsely structured cost function. Another sparsity assumption lies on the transport plan to be thinly spread, which is, e.g., the case for the L^2-regularized problem [21,22].

1.1. Our Contributions

In the present paper, we study the discrete, entropy-regularized MOT problem with a tree-structured [3,13] or a circle-structured cost function, where all measures are supported on a finite number of points (atoms) in $\mathbb{R}^d$. Then, the computational time of the Sinkhorn algorithm [28–30], which iteratively determines a sequence converging to the solution, depends linearly on the number K of input measures. If the numbers of atoms is large, however, the Sinkhorn algorithm still requires considerable computational time and memory, which mainly comes from computing a discrete convolution, i.e., a matrix–vector product with a kernel matrix.

This is significantly improved by Fourier-based fast summation methods [31,32]. The key idea is the approximation of the kernel function using a Fourier series, which enables the application of the non-uniform fast Fourier transform (NFFT). Although such fast summation methods are frequently used in different applications such as electrostatic particle interaction [33], tomography [34], image segmentation [35], and, very recently, also OT with two marginals [36], they have not been utilized for MOT so far. Furthermore, a method for accelerating the Sinkhorn algorithm for Wasserstein barycenters via computing the convolution with a different kernel, namely the heat kernel, was discussed in [37].

Our main contribution is the combination of the fast summation method with the sparsity of the tree- or circle-structured cost function in the MOT problem for accelerating the Sinkhorn algorithm. Each iteration step has a complexity of $\mathcal{O}(KN)$ for a tree- and $\mathcal{O}(KN^2)$ for a circle-structured cost function, compared to $\mathcal{O}(KN^2)$ and $\mathcal{O}(KN^3)$, respectively, with the straightforward matrix–vector operations, where N is an upper bound of the number of atoms for each of the K marginal measures. Our numerical tests with both tree- and circle-structured cost functions confirm a considerable acceleration, while the accuracy stays almost the same. A different acceleration of the Sinkhorn algorithm via low-rank approximation for tree-structured cost yields the same asymptotic complexity [38]. We note that MOT problems with tree-structured cost functions are used for the computation of Wasserstein barycenters [4], and with circle-structured cost for computing Euler flows [39].

1.2. Outline of the Paper

Section 2 introduces the notation. In Section 3, we focus on the discrete MOT problem with squared Euclidean norm cost functions and the numerical solution of the corresponding entropy-regularized problem, using the Sinkhorn algorithm. We investigate sparsely structured cost functions that decouple according to a tree or circle in Section 4. Then, the complexity of the Sinkhorn algorithm depends only linearly on K. Section 5 describes a fast summation method for further accelerating the Sinkhorn algorithm. Finally, in Section 6, we verify the practical performance by applying it to generalized Euler flows and for finding generalized Wasserstein barycenters. We compare the computational time of the proposed Sinkhorn algorithm based on the NFFT, with the algorithm based on direct matrix multiplication.

2. Notation

Let $K \in \mathbb{N}$ and $\boldsymbol{n} = (n_1, \ldots, n_k) \in \mathbb{N}^K$. We set $[K] := \{1, \cdots, K\}$ and consider K finite sets

$$\Omega^k = \left\{ \boldsymbol{x}^k_{i_k} : i_k \in [n_k] \right\} \subset \mathbb{R}^d, \quad k \in [K],$$

consisting of points $\boldsymbol{x}^k_{i_k} \in \mathbb{R}^d$, which are called *atoms*. We set $\boldsymbol{\Omega} := \Omega^1 \times \cdots \times \Omega^K$. Additionally, we define the index set

$$\boldsymbol{I} := \{ \boldsymbol{i} = (i_1, \ldots, i_K) : i_k \in [n_k], k \in [K] \}$$

and the set of K-dimensional matrices (tensors)

$$\mathbb{R}^{\boldsymbol{i}} := \mathbb{R}^{i_1} \times \cdots \times \mathbb{R}^{i_K}, \quad \boldsymbol{i} \in \boldsymbol{I}.$$

Let $\mathbb{P}(\Omega^k)$ denote the set of probability measures on Ω^k. In this paper, we consider K discrete probability measures, also called *marginal measures*, $\mu^k \in \mathbb{P}(\Omega^k)$, given by

$$\mu^k = \sum_{i_k=1}^{n_k} \mu^k_{i_k} \delta_{\boldsymbol{x}^k_{i_k}}, \quad k \in [K],$$

where the probabilities satisfy

$$\mu^k_{i_k} \geq 0, \quad \sum_{i_k=1}^{n_k} \mu^k_{i_k} = 1 \quad \text{for all } i_k \in [n_k],\ k \in [K],$$

and, for all $A \subset \Omega^k$, the Dirac measure is given by

$$\delta_{x^k_{i_k}}(A) := \begin{cases} 1 & \text{if } x^k_{i_k} \in A, \\ 0 & \text{otherwise.} \end{cases}$$

For $\boldsymbol{G}, \boldsymbol{H} \in \mathbb{R}^{\boldsymbol{n}}$, we denote their component-wise (Hadamard) product by

$$\boldsymbol{G} \odot \boldsymbol{H} := (G_{\boldsymbol{i}} H_{\boldsymbol{i}})_{\boldsymbol{i} \in \boldsymbol{I}} \in \mathbb{R}^{\boldsymbol{n}},$$

and similarly, their component-wise division by $\oslash$, as well as the Frobenius inner product

$$\langle \boldsymbol{G}, \boldsymbol{H} \rangle_{\mathrm{F}} := \sum_{\boldsymbol{i} \in \boldsymbol{I}} G_{\boldsymbol{i}} H_{\boldsymbol{i}} \in \mathbb{R}.$$

The tensor product (Kronecker product) of $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^m$ is denoted by $\boldsymbol{u} \otimes \boldsymbol{v} \in \mathbb{R}^{m \times m}$. Analogues can be defined for tensors of different size.

3. Multi-Marginal Optimal Transport

In the following, we consider the discrete *multi-marginal optimal transport* (MOT) between K marginal measures $\mu^k \in \mathbb{P}(\Omega^k)$, $k \in [K]$. We define the set of *admissible transport plans* by

$$\Pi(\mu^1, \ldots, \mu^k) := \left\{ \boldsymbol{\Pi} \in \mathbb{R}^{\boldsymbol{n}}_{\geq 0} : P_k(\boldsymbol{\Pi}) = \mu^k \text{ for all } k \in [K] \right\}, \tag{1}$$

where the k-th marginal projection is defined as

$$P_k(\mathbf{\Pi}) := \sum_{\ell \in [K] \setminus \{k\}} \sum_{i_\ell \in [n_\ell]} \mathbf{\Pi}_{i_1, \cdots, i_K} \in \mathbb{R}^{n_k}. \tag{2}$$

For a *cost function* $c\colon \mathbf{\Omega} \to \mathbb{R}_{\geq 0}$ and the samples $\boldsymbol{x_i} = \left(x^1_{i_1}, \ldots, x^K_{i_K}\right)$, $\boldsymbol{i} \in \boldsymbol{I}$, we define the respective *cost matrix*

$$\boldsymbol{C} := [\boldsymbol{C_i}]_{\boldsymbol{i} \in \boldsymbol{I}} = [c(\boldsymbol{x_i})]_{\boldsymbol{i} \in \boldsymbol{I}} = [c(\boldsymbol{x}^1_{i_1}, \cdots, \boldsymbol{x}^K_{i_K})]_{(i_1, \cdots, i_K) \in \boldsymbol{I}} \in \mathbb{R}^{\boldsymbol{n}}_{\geq 0}. \tag{3}$$

The discrete MOT problem reads

$$\min_{\mathbf{\Pi} \in \Pi(\mu^1, \ldots, \mu^k)} \langle \mathbf{\Pi}, \boldsymbol{C} \rangle_{\mathrm{F}}, \tag{4}$$

whose solution $\mathbf{\Pi} \in \mathbb{R}^{\boldsymbol{n}}_{\geq 0}$ is called *optimal plan*.

Entropy Regularization

As the MOT problem (4) is numerically unfeasible, we consider for $\eta > 0$ the entropy-regularized multi-marginal optimal transport (MOT$_\eta$) problem

$$\min_{\mathbf{\Pi} \in \Pi(\mu^1, \ldots, \mu^K)} \langle \mathbf{\Pi}, \boldsymbol{C} \rangle_{\mathrm{F}} + \eta \langle \mathbf{\Pi}, \log \mathbf{\Pi} - \mathbf{1}_{\boldsymbol{n}} \rangle_{\mathrm{F}} = \min_{\mathbf{\Pi} \in \Pi(\mu^1, \ldots, \mu^k)} \sum_{\boldsymbol{i} \in \boldsymbol{I}} \mathbf{\Pi}_{\boldsymbol{i}} \boldsymbol{C_i} + \eta \sum_{\boldsymbol{i} \in \boldsymbol{I}} \mathbf{\Pi}_{\boldsymbol{i}} \left(\log \mathbf{\Pi}_{\boldsymbol{i}} - 1\right), \tag{5}$$

which is a convex optimization problem. It is possible to numerically deduce the optimal transport plan $\hat{\mathbf{\Pi}}$ of (5) from the solution of the corresponding Lagrangian dual problem. The following theorem is a special case of [3] for a constant entropy function.

Theorem 1. *The Lagrangian dual formulation of the discrete MOT$_\eta$ problem* (5) *states*

$$\sup_{\phi^k \in \mathbb{R}^{n_k}_{\geq 0}, k \in [K]} \mathcal{S}\left(\phi^1, \cdots, \phi^K\right) := \sup_{\phi^k \in \mathbb{R}^{n_k}_{\geq 0}, k \in [K]} \eta \sum_{k \in [K]} \sum_{j \in [n_k]} \mu^k_j \log \phi^k_j - \eta \sum_{\boldsymbol{i} \in \boldsymbol{I}} \boldsymbol{\mathcal{K}_i} \mathbf{\Phi}_{\boldsymbol{i}}, \tag{6}$$

where the kernel matrix $\boldsymbol{\mathcal{K}} \in \mathbb{R}^{\boldsymbol{n}}$ is defined by

$$\boldsymbol{\mathcal{K}_i} := \exp\left(-\frac{\boldsymbol{C_i}}{\eta}\right), \quad \boldsymbol{i} \in \boldsymbol{I},$$

and the dual tensor $\mathbf{\Phi} = \bigotimes_{k=1}^K \phi^k$ by

$$\mathbf{\Phi}_{\boldsymbol{i}} := \prod_{k=1}^K \phi^k_{i_k}, \quad \boldsymbol{i} \in \boldsymbol{I}.$$

The functional $\mathcal{S}\colon \mathbb{R}^{\boldsymbol{n}}_{\geq 0} \to \mathbb{R}$ is called the Sinkhorn function.

The solutions of the dual and the primal problem are generally not equal. The solution of (6) is generally a lower bound to the solution of the primal problem (5). Equality holds if the cost function c is lower semi-continuous; i.e., $\liminf c(\boldsymbol{x}) \geq c(\boldsymbol{x_0})$ as $\boldsymbol{x} \to \boldsymbol{x_0}$ for every $\boldsymbol{x_0} \in \mathbb{R}^{\boldsymbol{n}}$, or Borel measurable and bounded [40]. This is obviously the case for the squared Euclidean norm cost function

$$c\colon \mathbb{R}^{\boldsymbol{n}} \to [0, \infty), \quad c(\boldsymbol{x}) = \sum_{\substack{k_1, k_2 \in [K] \\ k_1 \neq k_2}} \left\| x^{k_1}_{i_{k_1}} - x^{k_2}_{i_{k_2}} \right\|_2^2$$

that we study here.

Proposition 1 ([41])**.** *An optimal plan of the MOT$_\eta$ problem* (5) *is given by*

$$\hat{\boldsymbol{\Pi}} = \boldsymbol{\mathcal{K}} \odot \hat{\boldsymbol{\Phi}},$$

where $\hat{\boldsymbol{\Phi}} = \bigotimes_{k\in[K]} \hat{\phi}^k$ *and* $\hat{\phi}^k$, $k \in [K]$ *are the optimal solutions of dual problem* (6)*.*

A sequence converging to the optimal dual vectors $\hat{\phi}^k$, $k \in [K]$ in (6) can be iteratively determined by the *Sinkhorn algorithm* [13,42] presented in Algorithm 1, where we note that line 4 is obtained by deriving the Sinkhorn function $\mathcal{S}$ with respect to ϕ^k, $k \in [K]$. The complexity of the algorithm mainly comes from the computation of the marginal $P_k(\boldsymbol{\mathcal{K}} \odot \boldsymbol{\Phi})$, where the projection P_k is defined in (2). In general, the number of operations depends exponentially on K.

Algorithm 1 Sinkhorn iterations for the MOT$_\eta$ problem

```
 1: Input: Initial values $(\phi^k)^{(0)} \in \mathbb{R}^{n_k}$, $k \in [K]$, regularization parameter $\eta > 0$, threshold
    $\delta > 0$
 2: Set $r \leftarrow 0$
 3: do
 4:     for $k = 1, \ldots, K$ do
 5:         Compute $\left(\tilde{\boldsymbol{\Phi}}\right)_k^{(r+1)} := \bigotimes_{\ell\in[k-1]} \left(\phi^\ell\right)^{(r+1)} \otimes \bigotimes_{\ell\in[K]\setminus[k-1]} \left(\phi^\ell\right)^{(r)}$
 6:         Compute dual vectors
                $\left(\phi^k\right)^{(r+1)} := \left(\mu^k \odot \left(\phi^k\right)^{(r)}\right) \oslash P_k\left(\boldsymbol{\mathcal{K}} \odot \left(\tilde{\boldsymbol{\Phi}}\right)_k^{(r+1)}\right)$
 7:         Increment $r \leftarrow r + 1$
 8:     end for
 9: while $\left|\mathcal{S}\left(\left(\phi^1\right)^{(r)}, \ldots, \left(\phi^K\right)^{(r)}\right) - \mathcal{S}\left(\left(\phi^1\right)^{(r-1)}, \ldots, \left(\phi^K\right)^{(r-1)}\right)\right| \geq \delta$
10: return optimal plan $\hat{\boldsymbol{\Pi}} = \boldsymbol{\mathcal{K}} \odot \hat{\boldsymbol{\Phi}}$ and $\hat{\boldsymbol{\Phi}} = \bigotimes_{k\in[K]} \left(\phi^k\right)^{(r+1)}$
```

4. Sparse Cost Functions

In this section, we take a look at sparsely structured cost functions, for which the Sinkhorn algorithm becomes much faster and we overcome the curse of dimensionality. Let $G = (\mathcal{V}, \mathcal{E})$ be an undirected graph with vertices $\mathcal{V}$ and edges $\mathcal{E}$. We say that the MOT$_\eta$ problem has the graph G structure if $\mathcal{V} = [K]$ and the cost matrix (3) decouples according to

$$\boldsymbol{C}_{\boldsymbol{i}} = \sum_{\{k_1,k_2\}\in\mathcal{E}} \left\| x_{i_{k_1}}^{k_1} - x_{i_{k_2}}^{k_2} \right\|_2^2, \quad \boldsymbol{i} = (i_1, \ldots, i_K) \in \boldsymbol{I}.$$

This implies that the kernel $\boldsymbol{\mathcal{K}} \in \mathbb{R}_{\geq 0}^{\boldsymbol{n}}$ satisfies

$$\boldsymbol{\mathcal{K}}_{\boldsymbol{i}} = \exp\left(-\frac{\boldsymbol{C}_{\boldsymbol{i}}}{\eta}\right) = \prod_{\{k_1,k_2\}\in\mathcal{E}} \boldsymbol{\mathcal{K}}_{i_{k_1},i_{k_2}}^{(k_1,k_2)}, \quad \boldsymbol{i} \in \boldsymbol{I}, \tag{7}$$

where the kernel matrix $\boldsymbol{\mathcal{K}}^{(k_1,k_2)} \in \mathbb{R}_{\geq 0}^{n_{k_1} \times n_{k_2}}$ for $\{k_1, k_2\} \in \mathcal{E}$ is given by

$$\boldsymbol{\mathcal{K}}_{i_{k_1},i_{k_2}}^{(k_1,k_2)} := \exp\left(-\frac{1}{\eta}\left\| \boldsymbol{x}_{i_{k_1}}^{k_1} - \boldsymbol{x}_{i_{k_2}}^{k_2} \right\|_2^2\right). \tag{8}$$

We use the indices $k \in \mathcal{V} = [K]$ to identify the marginal measures μ^k in the rest of the paper.

The discrete, dual formulation (6) of the MOT$_\eta$ problem has the same form independently of the structure of the graph G; only the marginals $P_k(\boldsymbol{\mathcal{K}} \odot \boldsymbol{\Phi})$ differ. If the graph G is complete, i.e., each of the two vertices in $\mathcal{V}$ are connected with an edge, then the computational complexity of the Sinkhorn Algorithm 1 depends exponentially on the number K of marginal measures (vertices). For larger values of K, it is practically impossible to numerically compute an optimal plan for the MOT$_\eta$ problem. We consider two sparsity assumptions of the MOT$_\eta$ problem, each of them yielding that the Sinkhorn algorithm has a linear complexity in the number K of the nodes. It was shown in [25] that MOT$_\eta$ problems with graphically structured cost functions of constant tree-width can be implemented in polynomial time. This is the case for the tree and circle, whose tree-widths are 1 and 2, respectively. In the following, we give an explicit scheme to efficiently compute the Sinkhorn iterations for the tree and circle structures.

4.1. Tree Structure

We consider the MOT$_\eta$ problem with the structure of a *tree* $(\mathcal{V}, \mathcal{E})$, which is a connected and circle-free graph with $|\mathcal{E}| = |\mathcal{V}| - 1$. We define the (non-empty) *neighbour* set $\mathcal{N}_k$ of $k \in \mathcal{V}$ as the set of all nodes $\ell \in \mathcal{V}$, such that $\{k, \ell\} \in \mathcal{E}$. Furthermore, we denote by $\mathcal{L} := \{k \in \mathcal{V} : |\mathcal{N}_k| = 1\}$ the set of all *leaves* of the tree.

We call $1 \in \mathcal{V}$ the *root* of the tree. For every $k \in \mathcal{V}$, there is a unique path between k and the root. For $k \in \mathcal{V} \setminus \{1\}$, we define the *parent* $p(k)$ as the node in $\mathcal{N}_k$ such that $p(k)$ lies in the path between k and the root 1. The root has no parent. Without loss of generality, we assume that $p(k) < k$ holds for all $k \in \mathcal{V} \setminus \{1\}$. We define the set of *children* $\mathcal{C}_k := \{\ell \in \mathcal{N}_k : \ell > k\}$. Then, we can derive a recursive formula for the k-th marginal of the tensor $\boldsymbol{\mathcal{K}} \odot \boldsymbol{\Phi} \in \mathbb{R}^{\boldsymbol{n}}$.

Theorem 2. *Let $(\mathcal{V}, \mathcal{E})$ be a tree with leaves $\mathcal{L}$, and let c be the MOT$_\eta$ cost function associated. Furthermore, let $k \in \mathcal{V}$ be an arbitrary node. Then, the k-th marginal of the transport plan $\boldsymbol{\mathcal{K}} \odot \boldsymbol{\Phi}$ is given by*

$$P_k(\boldsymbol{\mathcal{K}} \odot \boldsymbol{\Phi}) = \phi^k \odot \bigodot_{\ell \in \mathcal{N}_k} \alpha^{(k,\ell)},$$

where the vectors $\alpha^{(k,\ell)} \in \mathbb{R}^{n_k}$ are recursively defined as

$$\alpha^{(k,\ell)} = \begin{cases} \mathcal{K}^{(k,\ell)} \phi^\ell & \textit{if } \ell \in \mathcal{L}, \\ \mathcal{K}^{(k,\ell)} \left(\phi^\ell \odot \bigodot_{t \in \mathcal{N}_\ell \setminus \{k\}} \alpha^{(\ell,t)} \right) & \textit{otherwise}. \end{cases}$$

A proof of Theorem 2 can be found in [13] (Theorem 3.2). The main idea is to split the rooted tree at the node k into $|\mathcal{N}_k|$ subtrees. Therefore, the kernel matrix holds

$$\boldsymbol{\mathcal{K}}_{\boldsymbol{i}} = \prod_{\{k_1,k_2\} \in \mathcal{E}} \mathcal{K}^{(k_1,k_2)}_{i_{k_1}, i_{k_2}} = \prod_{\ell \in \mathcal{N}_k} \prod_{t \in \mathcal{D}_\ell} \mathcal{K}^{(p(t),t)}_{i_{p(t)}, i_t}, \quad \boldsymbol{i} \in \boldsymbol{I},$$

where $\mathcal{D}_\ell$ is the set of descendants of ℓ, i.e., the nodes $t \in \mathcal{V}$ such that ℓ lies in the path between k and t. Inserting $\boldsymbol{\mathcal{K}}_{\boldsymbol{i}}$ into $P_k(\boldsymbol{\mathcal{K}} \odot \boldsymbol{\Phi})$ and recalling the definition of P_k in (2) yields the result.

In order to efficiently perform the Sinkhorn algorithm, we compute iteratively for $\ell = K, \ldots, 2$ the vectors $\beta_\ell := \alpha^{(p(\ell),\ell)} \in \mathbb{R}^{n_{p(\ell)}}$. From Theorem 2, we obtain

$$\beta_\ell = \begin{cases} \mathcal{K}^{(p(\ell),\ell)} \phi^\ell & \text{if } \ell \in \mathcal{L}, \\ \mathcal{K}^{(p(\ell),\ell)} \left(\phi^\ell \odot \bigodot_{t \in \mathcal{C}_\ell} \beta_t \right) & \text{otherwise.} \end{cases}$$

Since we assumed that $p(k) < k$, the computation of β_ℓ requires only β_t for $t > \ell$. Similarly, the vectors $\gamma_\ell := \alpha^{(\ell,p(\ell))} \in \mathbb{R}^{n_\ell}$ for $\ell > 1$ and $\gamma_1 := \mathbf{1}_{n_1}$ can be iteratively computed by

$$\gamma_\ell = \left(\boldsymbol{\mathcal{K}}^{(p(\ell),\ell)}\right)^\intercal \left(\phi^{p(\ell)} \odot \gamma_{p(\ell)} \odot \bigodot_{t\in\mathcal{C}_{p(\ell)}\setminus\{\ell\}} \beta_t\right), \quad \ell = 2,\dots,K,$$

where the vectors $\beta_t \in \mathbb{R}^{n_{p(\ell)}}$ are assumed to be known from above. Then, the marginals are

$$P_k(\boldsymbol{\mathcal{K}} \odot \boldsymbol{\Phi}) = \phi^k \odot \gamma_k \odot \beta^{\odot \mathcal{C}_k}, \quad k \in [K],$$

where for any subset $\mathcal{U} \subseteq \mathcal{C}_k$, we define

$$\mathbb{R}^{n_k} \ni \beta^{\odot \mathcal{U}} := \begin{cases} \mathbf{1}_{n_k} & \text{if } \mathcal{U} = \emptyset, \\ \bigodot_{t\in\mathcal{U}} \beta_t & \text{otherwise.} \end{cases}$$

The resulting procedure is summarized in Algorithm 2.

Algorithm 2 Sinkhorn algorithm for tree structure

```
1:  Input: Tree $T(\mathcal{V},\mathcal{E})$ with leaves $\mathcal{L}$ and root $1 \in \mathcal{V} = [K]$, initialization $(\phi^k)^{(0)}, k \in [K]$,
    parameters $\eta, \delta > 0$
2:  Initialize $r \leftarrow 0$
3:  for $k = K,\dots,2$ do
4:      $\beta_k^{(0)} := \begin{cases} \boldsymbol{\mathcal{K}}^{(p(k),k)}\left(\phi^k\right)^{(0)} & \text{if } k \in \mathcal{L}, \\ \boldsymbol{\mathcal{K}}^{(p(k),k)}\left(\left(\phi^k\right)^{(0)} \odot \left(\beta^{\odot\mathcal{C}_k}\right)^{(0)}\right) & \text{otherwise} \end{cases}$
5:  end for
6:  do
7:      for $k = 1,\dots,K$ do
8:          $\gamma_k^{(r)} := \begin{cases} \mathbf{1} & \text{if } k = 1, \\ \left(\boldsymbol{\mathcal{K}}^{(p(k),k)}\right)^\intercal \left(\left(\phi^{p(k)}\right)^{(r+1)} \odot \gamma_{p(k)}^{(r)} \odot \left(\beta^{\odot\mathcal{C}_{p(k)}\setminus\{k\}}\right)^{(r)}\right) & \text{otherwise} \end{cases}$
9:          Compute the dual vector $\left(\phi^k\right)^{(r+1)} := \mu^k \oslash \left(\gamma_k^{(r)} \odot \left(\beta^{\odot\mathcal{C}_k}\right)^{(r)}\right)$
10:     end for
11:     for $k = K,\dots,2$ do
12:         Compute $\beta_k^{(r+1)}$ according to step 4.
13:     end for
14:     Set $\mathcal{S}^{(r)} := \eta\left(\sum_{k=1}^{K}\left(\mu^k\right)^\intercal \log\left(\phi^k\right)^{(r)} - \left(\mu^1 \oslash \left(\phi^1\right)^{(r+1)}\right)^\intercal \left(\phi^1\right)^{(r)}\right)$
15:     Increment $r \leftarrow r + 1$
16: while $|\mathcal{S}^{(r)} - \mathcal{S}^{(r-1)}| \geq \delta$
17: return optimal plan $\hat{\boldsymbol{\Pi}} = \boldsymbol{\mathcal{K}} \odot \hat{\boldsymbol{\Phi}}$, where $\hat{\boldsymbol{\Phi}} = \bigotimes_{k\in[K]}\left(\phi^k\right)^{(r)}$
```

4.2. Circle Structure

We consider the MOT$_\eta$ problem where the graph $(\mathcal{V},\mathcal{E})$ is a circle. We assume for each $k \in \mathcal{V} = [K]$ that $\{k, k+1\} \in \mathcal{E}$ is an edge of the circle, where we set $k+1 = 1$ if $k = K$ and $k - 1 = K$, if $k = 1$. Thus, we can define the distance between two nodes $k_1, k_2 \in \mathcal{V}$ as

$$d(k_1,k_2) := \begin{cases} k_2 - k_1 & \text{if } k_2 \geq k_1, \\ K - k_1 + k_2 & \text{otherwise.} \end{cases}$$

Theorem 3. *Let $(\mathcal{V},\mathcal{E})$ be a circle and c be the MOT cost function associated. Furthermore, let $k \in \mathcal{V}$ be an arbitrary node. The k-th marginal of the transport plan $\boldsymbol{\mathcal{K}} \odot \boldsymbol{\Phi}$ is given by*

$$P_k(\boldsymbol{\mathcal{K}} \odot \boldsymbol{\Phi}) = \left(\left(\phi^k \odot \boldsymbol{\mathcal{K}}^{(k,k+1)}\right) \odot \left(\phi^{k+1} \odot \alpha^{(k+1,k)}\right)^{\mathsf{T}}\right)\mathbf{1}_{k+1},$$

where the matrices $\alpha^{(\ell,t)} \in \mathbb{R}^{n_\ell \times n_t}$, $\ell, t \in \mathcal{V}$ recursively satisfy

$$\alpha^{(\ell,t)} = \begin{cases} \boldsymbol{\mathcal{K}}^{(\ell,t)} & \text{if } d(\ell,t) = 1, \\ \boldsymbol{\mathcal{K}}^{(\ell,\ell+1)}\left(\phi^{\ell+1} \odot \alpha^{(\ell+1,t)}\right) & \text{otherwise,} \end{cases} \tag{9}$$

and we set $k+1 = 1$ if $k = K$ and $k-1 = N$, if $k = 1$.

Proof. The kernel $\boldsymbol{\mathcal{K}}$ can be decomposed as (7). Let $k \in [K]$ and $i_k \in [n_k]$. It holds then, that

$$\begin{aligned} [P_k(\boldsymbol{\mathcal{K}} \odot \boldsymbol{\Phi})]_{i_k} &= \sum_{\ell \in \mathcal{V}\setminus\{k\}} \sum_{i_\ell \in [n_\ell]} \boldsymbol{\mathcal{K}}_{\boldsymbol{i}} \odot \boldsymbol{\Phi}_{\boldsymbol{i}} = \phi^k_{i_k} \sum_{\ell \in \mathcal{V}\setminus\{k\}} \sum_{i_\ell \in [n_\ell]} \prod_{\{k_1,k_2\}\in\mathcal{E}} \boldsymbol{\mathcal{K}}^{(k_1,k_2)}_{i_{k_1},i_{k_2}} \prod_{j \in \mathcal{V}\setminus\{k\}} \phi^j_{i_j} \\ &= \phi^k_{i_k} \sum_{i_1} \phi^1_{i_1} \sum_{i_2} \boldsymbol{\mathcal{K}}^{(1,2)}_{i_1,i_2} \phi^2_{i_2} \cdots \sum_{i_{k-2}} \boldsymbol{\mathcal{K}}^{(k-3,k-2)}_{i_{k-3},i_{k-2}} \phi^{k-2}_{i_{k-2}} \sum_{i_{k-1}} \boldsymbol{\mathcal{K}}^{(k-2,k-1)}_{i_{k-2},i_{k-1}} \phi^{k-1}_{i_{k-1}} \boldsymbol{\mathcal{K}}^{(k-1,k)}_{i_{k-1},i_k} \\ &\quad \sum_{i_{k+1}} \boldsymbol{\mathcal{K}}^{(k,k+1)}_{i_k,i_{k+1}} \phi^{k+1}_{i_{k+1}} \cdots \sum_{i_K} \boldsymbol{\mathcal{K}}^{(K-1,K)}_{i_{K-1},i_K} \phi^K_{i_K} \boldsymbol{\mathcal{K}}^{(K,1)}_{i_K,i_1} \\ &= \phi^k_{i_k} \sum_{i_{k+1}} \boldsymbol{\mathcal{K}}^{(k,k+1)}_{i_k,i_{k+1}} \phi^{k+1}_{i_{k+1}} \cdots \sum_{i_K} \boldsymbol{\mathcal{K}}^{(K-1,K)}_{i_{K-1},i_K} \phi^K_{i_K} \sum_{i_1} \boldsymbol{\mathcal{K}}^{(K,1)}_{i_K,i_1} \phi^1_{i_1} \sum_{i_2} \boldsymbol{\mathcal{K}}^{(1,2)}_{i_1,i_2} \phi^2_{i_2} \cdots \\ &\quad \sum_{i_{k-2}} \boldsymbol{\mathcal{K}}^{(k-3,k-2)}_{i_{k-3},i_{k+2}} \phi^{k-2}_{i_{k-2}} \sum_{i_{k-1}} \boldsymbol{\mathcal{K}}^{(k-2,k-1)}_{i_{k-2},i_{k-1}} \phi^{k-1}_{i_{k-1}} \boldsymbol{\mathcal{K}}^{(k-1,k)}_{i_{k-1},i_k}. \end{aligned}$$

Setting $\psi^\ell := \phi^{(k+\ell-1)\text{mod } K}$ and $\tilde{\boldsymbol{\mathcal{K}}}^{(j,\ell+1)} := \boldsymbol{\mathcal{K}}^{\left((k+\ell-1)\text{mod } K,(k+\ell)\text{mod } K\right)}$ for every $\ell \in [K]$, we obtain

$$\begin{aligned} [P_k(\boldsymbol{\mathcal{K}} \odot \boldsymbol{\Phi})]_{i_k} &= \phi^k_{i_k} \sum_{j_2} \tilde{\boldsymbol{\mathcal{K}}}^{(1,2)}_{i_k,j_2} \psi^2_{j_2} \cdots \sum_{j_{K-1}} \tilde{\boldsymbol{\mathcal{K}}}^{(K-2,K-1)}_{j_{K-2},j_{K-1}} \psi^{K-1}_{j_{K-1}} \sum_{j_K} \tilde{\boldsymbol{\mathcal{K}}}^{(K-1,K)}_{j_{K-1},j_K} \psi^K_{j_K} \tilde{\boldsymbol{\mathcal{K}}}^{(K,1)}_{j_K,i_k} \\ &= [P_1\left(\tilde{\boldsymbol{\mathcal{K}}} \odot \tilde{\boldsymbol{\Phi}}\right)]_{i_k}. \end{aligned}$$

We define

$$\tilde{\alpha}^{(K,1)} := \tilde{\boldsymbol{\mathcal{K}}}^{(K,1)} = \alpha^{(k-1,k)} \in \mathbb{R}^{n_{k-1} \times n_k}$$

and recursively for all $\ell = K-1, \ldots, 1$, the matrix $\tilde{\alpha}^{(\ell,1)} \in \mathbb{R}^{((k+\ell-1)\text{mod } K) \times n_k}$ by

$$\tilde{\alpha}^{(\ell,1)}_{j_l,i_k} := \left[\tilde{\boldsymbol{\mathcal{K}}}^{(\ell,\ell+1)}\left(\psi^{\ell+1} \odot \tilde{\alpha}^{(\ell+1,1)}\right)\right]_{j_l,i_k} = \sum_{j_{\ell+1}} \tilde{\boldsymbol{\mathcal{K}}}^{(\ell,\ell+1)}_{j_\ell,j_{\ell+1}} \psi^{\ell+1}_{j_{\ell+1}} \tilde{\alpha}^{(\ell+1,1)}_{j_{\ell+1},i_k}.$$

Inserting this into the marginal yields

$$\begin{aligned} [P_1\left(\tilde{\boldsymbol{\mathcal{K}}} \odot \tilde{\boldsymbol{\Phi}}\right)]_{i_k} &= \phi^k_{i_k} \sum_{j_2} \tilde{\boldsymbol{\mathcal{K}}}^{(1,2)}_{i_k,j_2} \psi^2_{j_2} \cdots \sum_{j_{K-2}} \tilde{\boldsymbol{\mathcal{K}}}^{(K-2,K-1)}_{j_{K-2},j_{K-1}} \psi^{K-1}_{j_{K-1}} \sum_{j_{K-1}} \tilde{\boldsymbol{\mathcal{K}}}^{(K-2,K-1)}_{j_{K-2},j_{K-1}} \psi^{K-1}_{j_{K-1}} \tilde{\alpha}^{(K-1,1)}_{j_{K-1},i_k} \\ &= \psi^1_{i_k} \sum_{j_2} \tilde{\boldsymbol{\mathcal{K}}}^{(1,2)}_{i_k,j_2} \psi^2_{j_2} \tilde{\alpha}^{(2,1)}_{j_2,i_k} = \psi^1_{i_k} \left[\tilde{\boldsymbol{\mathcal{K}}}^{(1,2)}_{i_k,j_2}\right]^{\mathsf{T}}_{j_2 \in [n_{k+1}]} \left(\psi^2 \odot \left[\tilde{\alpha}^{(2,1)}_{j_2,i_k}\right]_{j_2 \in [n_{k+1}]}\right). \end{aligned}$$

Henceforth,

$$P_1\left(\tilde{\boldsymbol{\mathcal{K}}} \odot \tilde{\boldsymbol{\Phi}}\right) = \left(\left(\psi^1 \odot \tilde{\boldsymbol{\mathcal{K}}}^{(1,2)}\right) \odot \left(\psi^2 \odot \tilde{\alpha}^{(2,1)}\right)^{\mathsf{T}}\right)\mathbf{1}.$$

Finally, defining $\tilde{\alpha}^{(\ell,1)} = \alpha^{((k+\ell)\mathrm{mod}\, K,k)}$ yields the assumption. □

To efficiently compute the marginal optimal transport as for the tree structure, we choose $k = 1$ as the starting point and decompose the marginal into two matrices, which can be computed recursively as follows. For matrices $\boldsymbol{G}, \boldsymbol{H} \in \mathbb{R}^{n \times m}$, we define the inner product with respect to the second dimension by

$$\langle \boldsymbol{G}, \boldsymbol{H} \rangle := \left(\sum_{j=1}^{m} G_{ij} H_{ij} \right)_{i \in [n]} \in \mathbb{R}^n.$$

Theorem 4. *Under the assumptions of Theorem 3, we have*

$$P_k(\boldsymbol{\mathcal{K}} \odot \boldsymbol{\Phi}) = \begin{cases} \phi^1 \odot \left\langle \mathcal{K}^{(1,2)}, \left(\phi^2 \odot \alpha^{(2,1)}\right)^{\mathrm{T}} \right\rangle, & k = 1, \\ \phi^k \odot \left\langle \alpha^{(k,1)}, \left(\phi^1 \odot \lambda^{(1,k)}\right)^{\mathrm{T}} \right\rangle, & k = 2, \ldots, K, \end{cases}$$

where $\alpha^{(k,1)}$ *is given in* (9) *and*

$$\lambda^{(1,k)} := \begin{cases} \mathcal{K}^{(1,2)}, & k = 2, \\ \lambda^{(1,k-1)} \left(\phi^{(k-1)} \odot \mathcal{K}^{(k-1,k)} \right), & k = 3, \ldots, K. \end{cases} \tag{10}$$

Proof. Let $k \in [K]$ and $i_k \in [n_k]$. For $k = 1$, the assertion follows from Theorem 3. For $k > 2$, we have

$$\begin{aligned} \left[P_k(\boldsymbol{\mathcal{K}} \odot \boldsymbol{\Phi}) \right]_{i_k} &= \phi^k_{i_k} \sum_{i_1} \phi^1_{i_1} \sum_{i_2} \mathcal{K}^{(1,2)}_{i_1,i_2} \phi^2_{i_2} \cdots \sum_{i_{k-1}} \mathcal{K}^{(k-2,k-1)}_{i_{k-2},i_{k-1}} \phi^{k-1}_{i_{k-1}} \mathcal{K}^{(k-1,k)}_{i_{k-1},i_k} \\ &\quad \sum_{i_{k+1}} \mathcal{K}^{(k,k+1)}_{i_k,i_{k+1}} \phi^{k+1}_{i_{k+1}} \cdots \sum_{i_K} \mathcal{K}^{(K-1,K)}_{i_{K-1},i_K} \phi^K_{i_K} \mathcal{K}^{(K,1)}_{i_K,i_1} \\ &= \phi^k_{i_k} \sum_{i_1} \phi^1_{i_1} \underbrace{\sum_{i_2} \mathcal{K}^{(1,2)}_{i_1,i_2} \phi^2_{i_2} \cdots \sum_{i_{k-1}} \mathcal{K}^{(k-2,k-1)}_{i_{k-2},i_{k-1}} \phi^{k-1}_{i_{k-1}} \mathcal{K}^{(k-1,k)}_{i_{k-1},i_k}}_{\mathrm{I}:=} \alpha^{(k,1)}_{i_k,i_1}. \end{aligned}$$

Furthermore, the term I can be rewritten as

$$\mathrm{I} = \sum_{i_{k-1}} \mathcal{K}^{(k-1,k)}_{i_{k-1},i_k} \phi^{k-1}_{i_{k-1}} \sum_{i_{k-2}} \mathcal{K}^{(k-2,k-1)}_{i_{k-2},i_{k-1}} \phi^{k-2}_{i_{k-2}} \cdots \sum_{i_2} \mathcal{K}^{(1,2)}_{i_1,i_2} \phi^2_{i_2} \mathcal{K}^{(2,3)}_{i_2,i_3}$$

By (10), we have

$$\lambda^{(1,3)}_{i_1,i_3} = \sum_{i_2} \mathcal{K}^{(1,2)}_{i_1,i_2} \phi^2_{i_2} \mathcal{K}^{(2,3)}_{i_2,i_3}, \quad i_1 \in [n_1],\ i_3 \in [n_3].$$

This implies that

$$\begin{aligned} \mathrm{I} &= \sum_{i_{k-1}} \mathcal{K}^{(k-1,k)}_{i_{k-1},i_k} \phi^{k-1}_{i_{k-1}} \sum_{i_{k-2}} \mathcal{K}^{(k-2,k-1)}_{i_{k-2},i_{k-1}} \phi^{k-2}_{i_{k-2}} \cdots \sum_{i_3} \mathcal{K}^{(3,4)}_{i_3,i_4} \phi^3_{i_3} \lambda^{(1,3)}_{i_1,i_3}, \\ &= \cdots = \sum_{i_{k-1}} \mathcal{K}^{(k-1,k)}_{i_{k-1},i_k} \phi^{k-1}_{i_{k-1}} \lambda^{(1,k-1)}_{i_1,i_{k-1}} = \left[\lambda^{(1,k-1)} \left(\phi^{k-1} \odot \mathcal{K}^{(k-1,k)} \right) \right]_{i_1,i_k} = \lambda^{(1,k)}_{i_1,i_k}. \end{aligned}$$

Hence, the hypothesis is true for all $k > 2$. The case $k = 2$ can be proven using the same procedure. □

In order to efficiently compute a maximizing sequence of the dual MOT$_\eta$ problem (6), we set for $k \in [K]$ the dual matrices

$$\beta_k := \alpha^{(k,1)} \in \mathbb{R}^{n_k \times n_1},$$
$$\gamma_k := \lambda^{(k,1)} \in \mathbb{R}^{n_1 \times n_k},$$

given in Theorems 3 and 4, respectively. The method is shown in Algorithm 3.

Algorithm 3 Sinkhorn algorithm for circle structure

```
1:  Input: Initialization $(\phi^k)^{(0)}$, $k \in [K]$, parameters $\eta, \delta > 0$
2:  Initialize $r \leftarrow 0$
3:  for $k = K, \cdots, 2$ do
4:      $(\beta_k)^{(0)} := \begin{cases} \mathcal{K}^{(K,1)} & \text{if } k = K, \\ \mathcal{K}^{(k,k+1)}\left(\left(\phi^{k+1}\right)^{(0)} \odot (\beta_{k+1})^{(0)}\right) & \text{otherwise} \end{cases}$
5:  end for
6:  do
7:      $\left(\phi^1\right)^{(r+1)} := \mu^1 \oslash \left\langle \mathcal{K}^{(1,2)}, \left(\left(\phi^2\right)^{(r)} \odot (\beta_2)^{(r)}\right)^\top \right\rangle$
8:      for $k = 2, \cdots, K$ do
9:          $(\gamma_k)^{(r)} := \begin{cases} \mathcal{K}^{(1,2)} & \text{if } k = 2, \\ (\gamma_{k-1})^{(r)}\left(\left(\phi^{k-1}\right)^{(r+1)} \odot \mathcal{K}^{(k-1,k)}\right) & \text{otherwise} \end{cases}$
10:         Compute dual vector $\left(\phi^k\right)^{(r+1)} := \mu^k \oslash \left\langle (\beta_k)^{(r)}, \left(\left(\phi^1\right)^{(r+1)} \odot (\gamma_k)^{(r)}\right)^\top \right\rangle$
11:     end for
12:     for $k = K, \cdots, 2$ do
13:         Compute $(\beta_k)^{(r+1)}$ according to step 4
14:     end for
15:     Compute $\mathcal{S}^{(r)} := \eta\left(\sum_{k=1}^{K}\left(\mu^k\right)^\top \log\left(\phi^k\right)^{(r)} - \left(\mu^1 \oslash \left(\phi^1\right)^{(r+1)}\right)^\top \left(\phi^1\right)^{(r)}\right)$
16:     Increment $r \leftarrow r + 1$
17: while $|S^{(r)} - S^{(r-1)}| \geq \delta$
18: return Optimal plan $\hat{\boldsymbol{\Pi}} = \mathcal{K} \odot \hat{\boldsymbol{\Phi}}$, where $\hat{\boldsymbol{\Phi}} = \bigotimes_{k \in [K]}\left(\phi^k\right)^{(r+1)}$
```

The tree-structured and circle-structured MOT$_\eta$ problems both have a sparse cost function that considerably improves the computational complexity of the Sinkhorn algorithm. In each iteration step, Algorithm 2 requires only $2(K-1)$ matrix–vector products, which have a complexity of $\mathcal{O}(N^2)$, where $N := \|\boldsymbol{n}\|_\infty$, and Algorithm 3 requires $2(K-1)$ matrix–matrix products, which have a complexity of $\mathcal{O}(N^3)$. This can be considerably improved by employing fast Fourier techniques, as we will see in the next section.

5. Non-Uniform Discrete Fourier Transforms

The main computational cost of the Sinkhorn algorithm comes from the matrix–vector product with the kernel matrix (8). Let $k, \ell \in [K]$ and $\alpha \in \mathbb{R}^{n_\ell}$. We briefly describe a fast summation method for the computation of $\beta = \mathcal{K}^{(k,\ell)}\alpha$, i.e.,

$$\beta_{i_k} = \sum_{i_\ell=1}^{n_\ell} \alpha_{i_\ell} \exp\left(-\tfrac{1}{\eta}\left\|x_{i_k}^k - x_{i_\ell}^\ell\right\|_2^2\right), \quad i_k \in [n_k]. \tag{11}$$

We refer to [32] for a detailed derivation and error estimates. The main idea is to approximate the kernel function

$$\kappa(x) := \exp(-\tfrac{1}{\eta}x^2), \quad x \in \mathbb{R}, \tag{12}$$

using a Fourier series. In order to ensure the fast convergence of the Fourier series, we extend κ to a periodic function of certain smoothness $p \in \mathbb{N}$. Let $\varepsilon_{\mathrm{B}} > 0$ and $\tau > \varepsilon_{\mathrm{B}} + \max\Big\{ \|x^k_{i_k} - x^\ell_{i_\ell}\|_2^2 : i_k \in [n_k], i_\ell \in [n_\ell] \Big\}$. For $x \in \mathbb{R}$, we define the regularized kernel

$$\kappa_{\mathrm{R}}(x) := \begin{cases} \kappa(x), & |x| \le \tau - \varepsilon_{\mathrm{B}}, \\ \kappa_{\mathrm{B}}(x), & \tau - \varepsilon_{\mathrm{B}} < |x| \le \tau, \\ \kappa_{\mathrm{B}}(\tau), & |x| > \tau, \end{cases} \tag{13}$$

where κ_{B} is a polynomial of degree p that fulfills the Herminte interpolation conditions $\kappa_{\mathrm{B}}^{(r)}(\tau - \varepsilon_{\mathrm{B}}) = \kappa^{(r)}(\tau - \varepsilon_{\mathrm{B}})$ for $r = 0, \dots, p-1$, and $\kappa_{\mathrm{B}}^{(r)}(\tau) = 0$ for $r = 1, \dots, p-1$; see Figure 1.

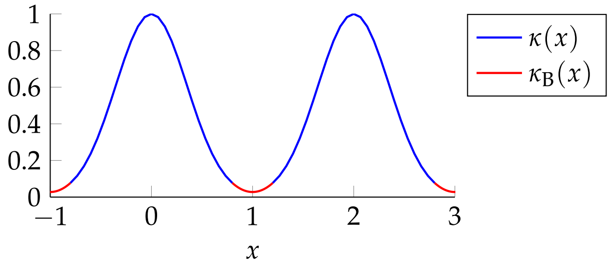

Figure 1. Regularized kernel κ_{R} for $\eta = 1/4$, periodicity length $\tau = 1$, boundary interval $\varepsilon_{\mathrm{B}} = 0.2$, and smoothness $p = 1$.

Then, we define a 2τ-periodic function on $\mathbb{R}^d$ by

$$\tilde{\kappa}(x) := \kappa_{\mathrm{R}}(\|x\|), \quad x \in [-\tau, \tau)^d. \tag{14}$$

By construction, $\tilde{\kappa}$ is p times continuously differentiable and we have $\tilde{\kappa}(x^k_{i_k} - x^\ell_{i_\ell}) = \exp(-\frac{1}{\eta}\|x^k_{i_k} - x^\ell_{i_\ell}\|_2^2)$ for all $i_k \in [n_k]$, $i_\ell \in [n_\ell]$.

Let $M \in \mathbb{N}$. We approximate $\tilde{\kappa}$ using the 2τ-periodic Fourier expansion of degree $2M$,

$$\tilde{\kappa}(x) \approx \sum_{m \in \{-M,\dots,M-1\}^d} \hat{\kappa}(m) \exp(\mathrm{i}\tfrac{\pi}{\tau} m^\intercal x), \quad x \in \mathbb{R}^d,$$

with the discrete Fourier coefficients $\hat{\kappa}(m) \in \mathbb{C}$, which can be efficiently approximated by the fast Fourier transform (FFT)

$$\hat{\kappa}(m) := \frac{1}{(2M)^d} \sum_{x \in \frac{\tau}{M}\{-M,\dots,M-1\}^d} \tilde{\kappa}(x) \exp(-\mathrm{i}\tfrac{\pi}{\tau} m^\top x), \quad m \in \{-M, \dots, M-1\}^d. \tag{15}$$

This yields an approximation of (11) by

$$\begin{aligned} \beta_{i_k} = \sum_{j=1}^{n_\ell} \tilde{\kappa}(x^k_{i_k} - x^\ell_{i_\ell})\, \alpha_{i_\ell} &\approx \sum_{i_\ell=1}^{n_\ell} \sum_{m \in \{-M,\dots,M-1\}^d} \hat{\kappa}(m) \exp\Big(\mathrm{i}\tfrac{\pi}{\tau} m^\intercal (x^k_{i_k} - x^\ell_{i_\ell})\Big)\, \alpha_{i_\ell} \\ &= \sum_{m \in \{-M,\dots,M-1\}^d} \hat{\kappa}(m) \left(\sum_{i_\ell=1}^{n_\ell} \alpha_{i_\ell} \exp\Big(-\mathrm{i}\tfrac{\pi}{\tau} m^\intercal x^\ell_{i_\ell}\Big) \right) \exp\Big(\mathrm{i}\tfrac{\pi}{\tau} m^\intercal x^k_{i_k}\Big). \end{aligned} \tag{16}$$

The *non-uniform discrete Fourier transform* (NDFT) of $\hat{\boldsymbol{\kappa}} := [\hat{\kappa}(m)]_{m\in\{-M,\dots,M-1\}^d}$ at the nodes $\Omega^k \subset \mathbb{R}^d$ is defined by

$$[\boldsymbol{F}_k\hat{\boldsymbol{\kappa}}]_{i_\ell} := \sum_{m\in\{-M,\dots,M-1\}^d} \hat{\kappa}(m)\, \exp\Big(-\mathrm{i}\tfrac{\pi}{\tau} m^\intercal x^k_{i_\ell}\Big), \quad i_\ell \in [n_\ell], \tag{17}$$

and the adjoint NDFT of $\alpha \in \mathbb{R}^{n_\ell}$ on the set Ω^ℓ is given by

$$[\boldsymbol{F}^*_\ell\alpha]_m := \sum_{i_\ell=1}^{n_\ell} \alpha_{i_\ell}\, \exp\Big(-\mathrm{i}\tfrac{\pi}{\tau} m^\intercal x^\ell_{i_\ell}\Big), \quad m \in \{-M,\dots,M-1\}^d, \tag{18}$$

cf. [43] (Section 7). Therefore, the approximation (16) can be written as

$$\beta = \boldsymbol{\mathcal{K}}^{(k,\ell)}\alpha \approx \boldsymbol{F}_k(\hat{\boldsymbol{\kappa}} \odot \boldsymbol{F}^*_\ell\alpha). \tag{19}$$

The procedure is summarized in Algorithm 4.

Algorithm 4 NFFT-based fast summation

```
Input: Vector $\alpha \in \mathbb{R}^{n_\ell}$, kernel function $\kappa$ in (12), parameters $\varepsilon_{\mathrm{B}} > 0$, $M \in \mathbb{N}$
precomputation
    Compute the regularized kernel $\kappa_{\mathrm{R}}$ by (13)
    Compute the periodized kernel $\tilde{\kappa}$ by (14)
    Compute the discrete Fourier coefficients $\hat{\kappa}(m)$, $m \in \{-M,\dots,M-1\}^d$, by an FFT, see (15)
end
Compute the adjoint NDFT $\boldsymbol{F}_\ell\alpha$, see (18)
Compute the pointwise product $\hat{\beta} := \hat{\boldsymbol{\kappa}} \odot \boldsymbol{F}_\ell\alpha$
Compute the NDFT $\beta := \boldsymbol{F}_k\hat{\beta}$, see (17)
return $\beta \approx \boldsymbol{\mathcal{K}}^{(k,\ell)}\alpha$
```

There are fast algorithms, known as *non-uniform fast Fourier transform* (NFFT), allowing for the computation of an NDFT (17) and its adjoint (18) in $\mathcal{O}(M^d \log M + N)$ steps up to arbitrary numeric precision; see e.g., [44,45] and [43] (Section 7), where $N = \|\boldsymbol{n}\|_\infty$. Note that the direct implementation of (16) requires $\mathcal{O}(M^d N)$ operations. We call the Sinkhorn algorithm where the matrix–vector multiplication is performed via (19) the *NFFT-Sinkhorn algorithm*. If we fix the Fourier expansion degree M, which is possible because of the smoothness of κ, then we end up at a numerical complexity of $\mathcal{O}(KN)$ for each iteration step of the NFFT-Sinkhorn algorithm for trees. In the case of a circle (Algorithm 3), we can apply the fast summation column by column for the matrix–matrix product with $\boldsymbol{\mathcal{K}}^{(k,k+1)}$, yielding a complexity of $\mathcal{O}(KN^2)$ for each iteration step.

6. Numerical Examples

We illustrate the results from Section 5 concerning the Sinkhorn algorithm and its accelerated version, the NFFT-Sinkhorn. First, we investigate the effect of parameter choices in some artificial examples. Then, we look at the one-dimensional Euler flow problem of incompressible fluids and the fixed support barycenter problem of images (the code for our examples is available at https://github.com/fatima0111/NFFT-Sinkhorn, accessed on 8 August 2022). All computations were performed on an eight-core Intel Core i7-10700 CPU with 32 GB memory. For computing the NFFT of Section 5, we rely on the implementation [46].

6.1. Uniformly Distributed Points

We consider the MOT_η problem for uniform measures μ^k of uniformly distributed points on $\Omega = [-1/2, 1/2]$. We chose the entropy regularization parameter $\eta = 0.1$, so that a boundary regularization (13) for the fast summation method is necessary. For the tree-structured cost function, we set the boundary regularization $\varepsilon_{\mathrm{B}} = 1/16$, the Fourier expansion degree $M = 156$, and the smoothness parameters $p = 3$ (see Section 5). In Figure 2 left, we see the linear dependence of the computational time on the number K of marginals. For a growing number N of points, the NFFT-Sinkhorn algorithm, which requires $\mathcal{O}(KN)$ steps, clearly outperforms the standard method, which requires $\mathcal{O}(KN^2)$ steps (see Figure 2 right).

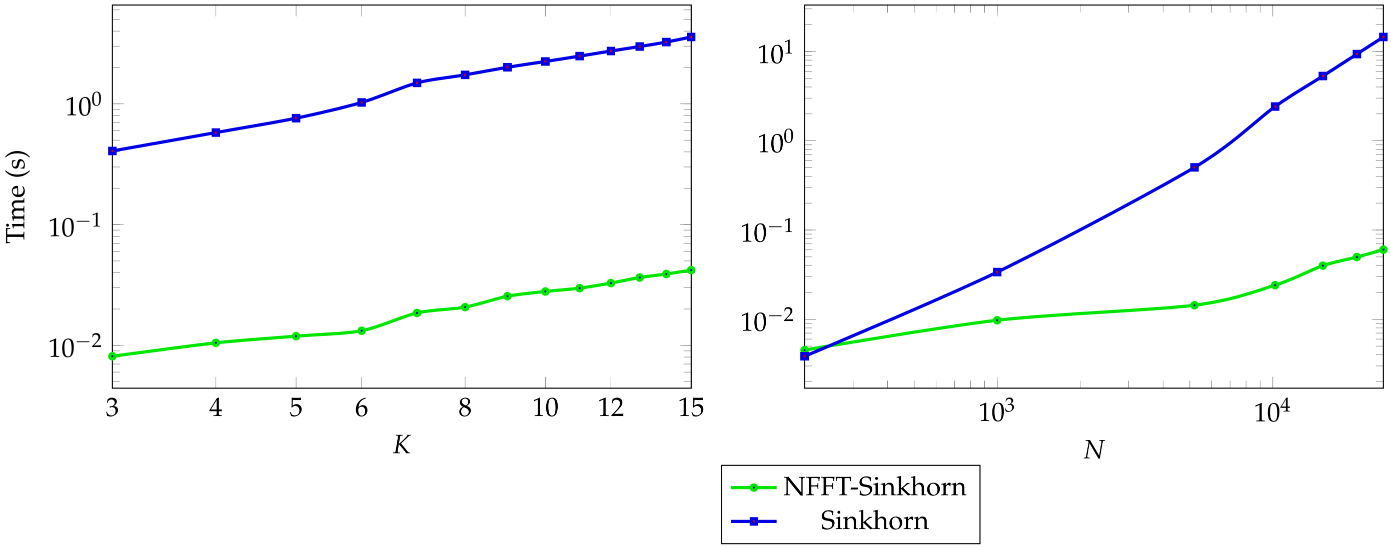

Figure 2. Computation time in seconds of MOT_η with tree-structured cost function with regularization parameter $\eta = 0.1$. **Left**: fixed $N = 10^4$. **Right**: fixed $K = 10$.

In Figure 3, we show the computation times for the circle-structured cost function, with the parameters $\eta = 0.1$, $\varepsilon_{\mathrm{B}} = 3/32$, $M = 2000$, and $p = 3$. As the Sinkhorn iteration requires matrix–matrix products, it is more costly than for the tree-structured cost function, and so we used a lower number of points N. The advantage of the NFFT-Sinkhorn is smaller than for the tree, but still considerable. We point out here that the fast summation method is applied column by column to the matrix–matrix product, and the Fourier expansion degree M is larger.

Finally, we investigate how the approximation error between the Sinkhorn and NFFT-Sinkhorn algorithm, $|\mathcal{S}^{(r)} - \tilde{\mathcal{S}}^{(r)}|$, depends on the entropy regularization parameter η and the Fourier expansion degree M at a fixed iteration $r = 10$, where $\mathcal{S}^{(r)}$ denotes the evaluation with the Sinkhorn algorithm and $\tilde{\mathcal{S}}^{(r)}$ its evaluation with the NFFT-Sinkhorn algorithm. Since the time differences for different M in the one-dimensional case are very small, we consider the two-dimensional uniform marginal measures for the MOT_η problems with tree- or circle-structured cost functions. In Figures 4 and 5, we see that for smaller η, we need a larger expansion degree M to achieve a good accuracy. The error stagnates at a certain level and does not decrease anymore for increasing M. This could be improved by increasing the approximation parameter p and the cutoff parameter of the NFFT, cf. [43] (Section 7). The parameter choice methods for the NFFT-based summation were discussed in [47]. For an appropriately chosen M, the NFFT-Sinkhorn is usually much faster than the Sinkhorn algorithm. However, for very small η, the kernel function (12) is concentrated over a small interval, and therefore, a simple truncation of the sum (11) might be beneficial to the NFFT approximation.

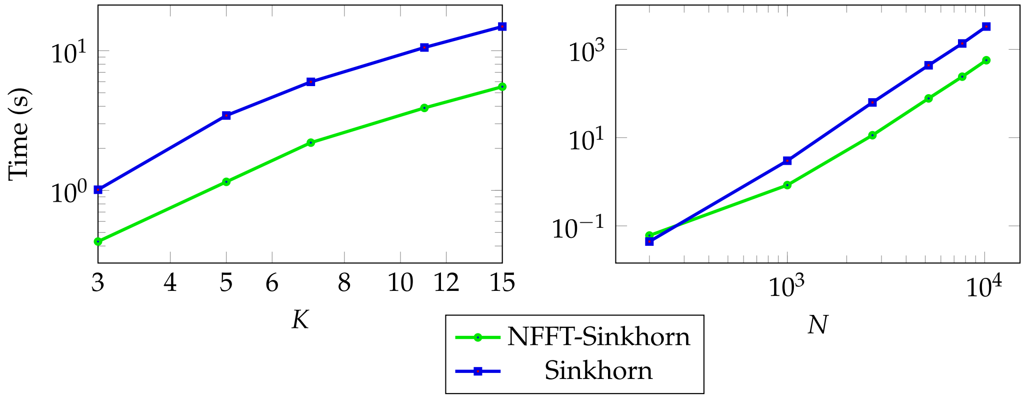

Figure 3. Computation time in seconds of MOT$_\eta$ with a circle-structured cost function with regularization parameter $\eta = 0.1$. **Left**: fixed $N = 700$. **Right**: fixed $K = 3$.

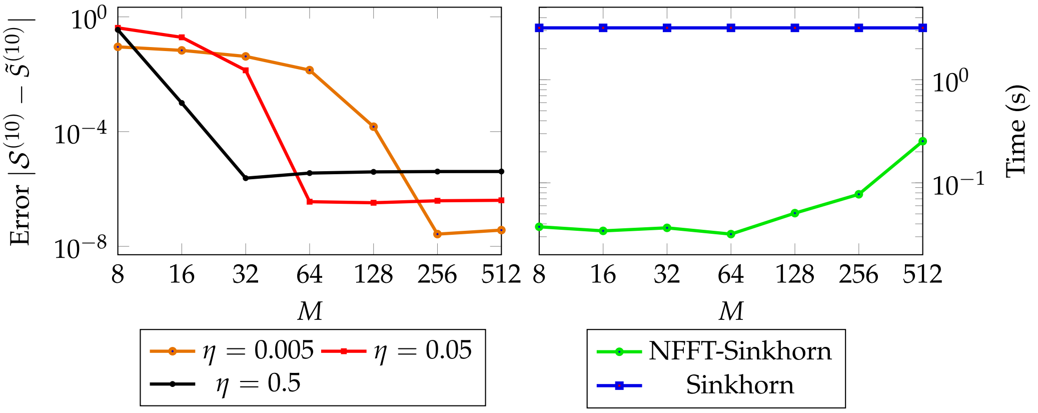

Figure 4. MOT$_\eta$ with tree-structured cost function, where $d = 2$, $K = 10$, $N = 10,000$, and $p = 3$. **Left**: Approximation error of $\mathcal{S}^{(10)}$ between the Sinkhorn and the NFFT-Sinkhorn algorithm, depending on the number of Fourier coefficients M of the NFFT. **Right**: Computation time in seconds.

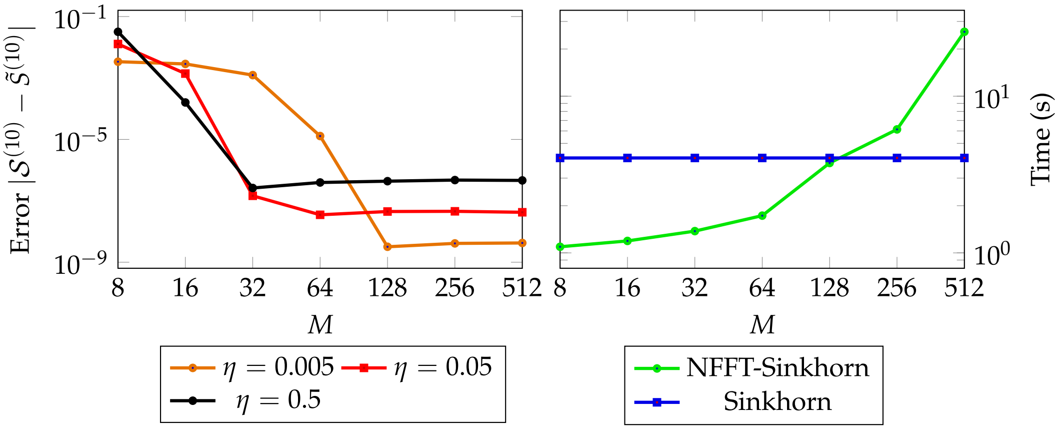

Figure 5. MOT$_\eta$ with circle-structured cost function, where $d = 2$, $K = 3$, $N = 1000$, and $p = 3$. Left: Approximation error of $\mathcal{S}^{(10)}$ between Sinkhorn and NFFT-Sinkhorn algorithm depending on the number of Fourier coefficients M. Right: Computation time in seconds.

6.2. Fixed-Support Wasserstein Barycenter for General Trees

Let $\mathcal{T} = (\mathcal{V}, \mathcal{E})$ be a tree with set of leaves $\mathcal{L}$; see Section 4.1. For $k \in \mathcal{L}$, let measures μ^k and weights $\tilde{w}_k \in [0,1]$ be given that satisfy $\sum_{k\in\mathcal{L}} \tilde{w}_k = 1$. For any edge $e \in \mathcal{E}$, we set

$$w_e := \begin{cases} 1 & \text{if } |e \cap \mathcal{L}| \neq 1, \\ \tilde{w}_k & \text{if } e \cap \mathcal{L} = \{k\}. \end{cases}$$

The generalized barycenters are the minimizers μ^k, $k \in \mathcal{V} \setminus \mathcal{L}$, of

$$\inf_{\mu^k \in \mathbb{P}(\Omega^k), k \in \mathcal{V} \setminus \mathcal{L}} \sum_{e=\{e_1,e_2\} \in \mathcal{E}} w_e \mathrm{W}_2^2(\mu^{e_1}, \mu^{e_2}), \tag{20}$$

where $\mathrm{W}_2^2(\mu^{e_1}, \mu^{e_2})$ is the squared Wasserstein distance [48,49] between the measures μ^{e_1} and μ^{e_2}. The well-known Wasserstein barycenter problem [50,51] is a special case of (20), where the tree is star-shaped and the barycenter corresponds to the unique internal node. We consider the fixed-support barycenter problem [26,52], where the nodes $\boldsymbol{x}^k$, $k \in \mathcal{V} \setminus \mathcal{L}$ are also given, so that we need to optimize (20) only for $\mu^k_{i_k}$, $k \in \mathcal{V} \setminus \mathcal{L}$. This yields an MOT problem with the tree-structured cost

$$C_{\boldsymbol{i}} = \sum_{e=\{e_1,e_2\} \in \mathcal{E}} w_e \left\| \boldsymbol{x}^{e_1}_{i_{e_1}} - \boldsymbol{x}^{e_2}_{i_{e_1}} \right\|_2^2, \quad \boldsymbol{i} \in \boldsymbol{I},$$

where the marginal constraints of (1) are only set for the known measures μ^k, $k \in \mathcal{L}$ (see [53]). This barycenter problem can be solved approximately using a modification of the Sinkhorn Algorithm 2, in which we replace line 12 by

$$(\phi^k)^{(r+1)} = \begin{cases} \mathbf{1} & \text{if } k \in \mathcal{L}, \\ \mu^k \oslash \left(\gamma_k^{(r)} \odot \left(\beta^{\odot c_k} \right)^{(r)} \right) & \text{otherwise.} \end{cases}$$

We test our algorithm with a tree consisting of $K = 7$ nodes; see Figure 6. The four given marginals μ^k, $k \in \mathcal{L}$, are dithered images in $\mathbb{R}^2$ with uniform weights $\mu^k_{i_k} = 1/N$. As support points of the barycenters μ^k, $k \in \mathcal{V} \setminus \mathcal{L}$, we take the union over all support points $\boldsymbol{x}^k_{i_k}$ of all four input measures μ^k, $k \in \mathcal{L}$. Furthermore, we use the barycenter weights $\tilde{w}^k = 1/4$. The given images and the computed barycenters are show in Figure 7, where we executed $r = 150$ iterations of the Sinkhorn algorithm and its accelerated version. We chose the regularization parameter $\eta = 5 \cdot 10^{-3}$ and the fast summation parameters $M = 156$, $p = 3$.

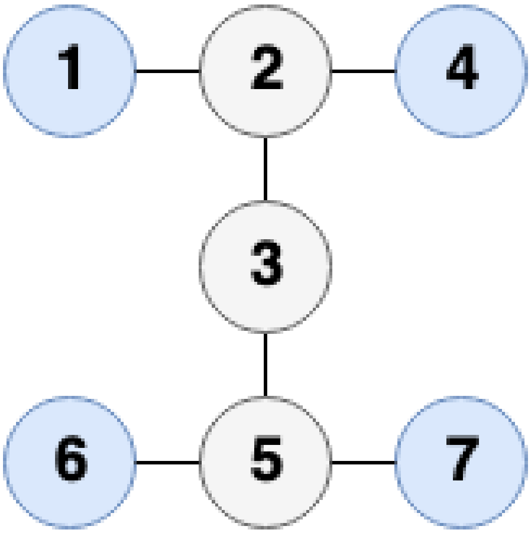

Figure 6. The tree graph of the barycenter problem, with leaves $\mathcal{L}$ marked in blue.

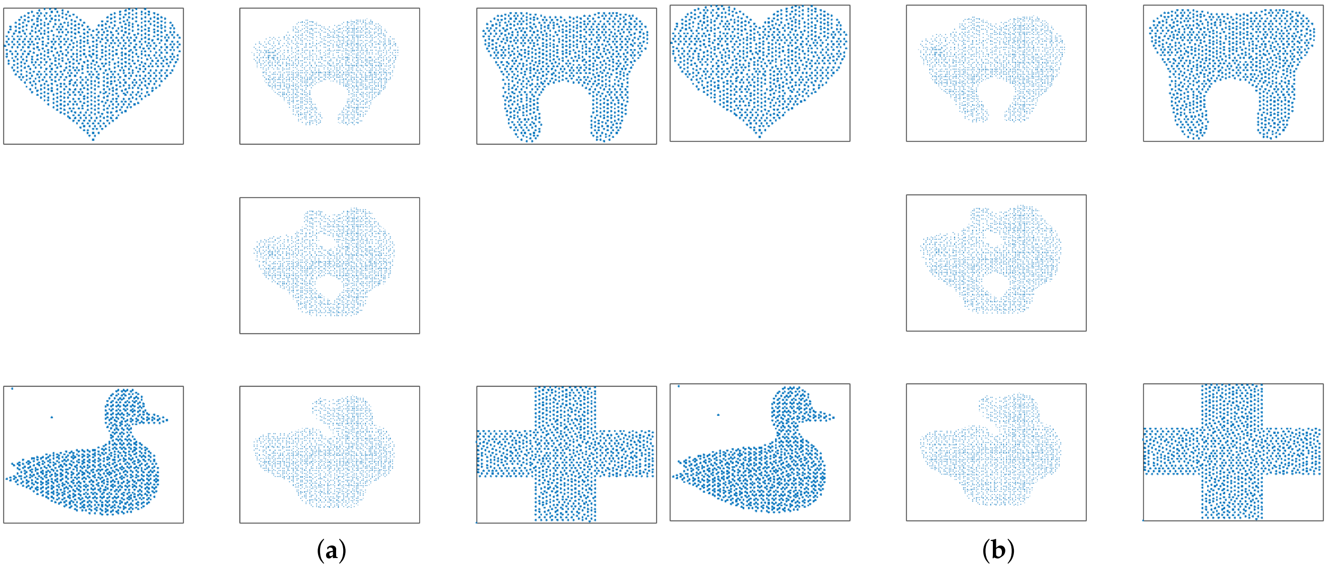

Figure 7. Given measures: $\{1,4,6,7\}$, entropy regularization parameter $\eta = 5 \cdot 10^{-3}$, $r = 150$, $\tilde{w} = \frac{1}{4}(1,1,1,1)$. NFFT-Sinkhorn parameters: $M = 156$, $p = 3$. The test images are adapted with permission (MIT License) from https://github.com/PythonOT/POT, see [54]. 2016, Rémi Flamary. (**a**) Sinkhorn. (**b**) NFFT-Sinkhorn.

6.3. Generalized Euler Flows

We consider the motion of N particles of an incompressible fluid, e.g., water, in a bounded domain Ω in discrete time steps $t_k := (k-1)/(K-1)$, $k \in [K]$. We assume that we know the function $\sigma\colon \Omega \to \Omega$, which connects N initial positions $x_{i_1} \in \Omega$ of particles with their final positions $x_{i_K} \in \Omega$. At each time step t_k, we know an image of the particle distribution, which is described by the discrete marginal measure μ^k, $k \in [K]$, with uniform weights $\mu^k_{i_k} = 1/N$. We want to find out how the single particles move, i.e., their trajectories. Due to the least-action principle, this problem can be formulated as the MOT problem (4) with the circle-structured cost

$$C_{\boldsymbol{i}} = \left\| x^K_{i_K} - \sigma\left(x^1_{k_1}\right) \right\|_2^2 + \sum_{k=1}^{K-1} \left\| x^{k+1}_{i_{k+1}} - x^k_{i_k} \right\|_2^2, \quad \boldsymbol{i} \in \boldsymbol{I},$$

see [55–57]. Then, the pair marginal

$$\boldsymbol{\Pi}_{1,k} := \sum_{\ell \in [K]\setminus\{1,k\}} \sum_{i_\ell \in [n_\ell]} \boldsymbol{\Pi}_{i_1,\ldots,i_K}$$

of the optimal plan $\boldsymbol{\Pi}$ provides the (discrete) probability that a particle that was initially at position $x_{i_1} \in \Omega$ is in position $x_{i_k} \in \Omega$ at time t_k, $k = 2,\ldots,K-1$. The one-dimensional problem has been studied by several authors [25,26,39], where the particles are assumed to be on a grid. Here, we consider the case where the positions are uniformly distributed. We draw $N = 400$ uniformly distributed points on $\Omega = [0,1]$. We use $K = 5$ marginal constraints and the entropy regularization parameter $\eta = 0.05$. Figures 8 and 9 display the probability matrix $\boldsymbol{\Pi}_{1,k}$ describing the motion of the particles from initial time $t_1 = 0$ to time t_k, where we use $r = 50$ iterations for both the NFFT-Sinkhorn and the Sinkhorn algorithms, and two different connection functions σ.

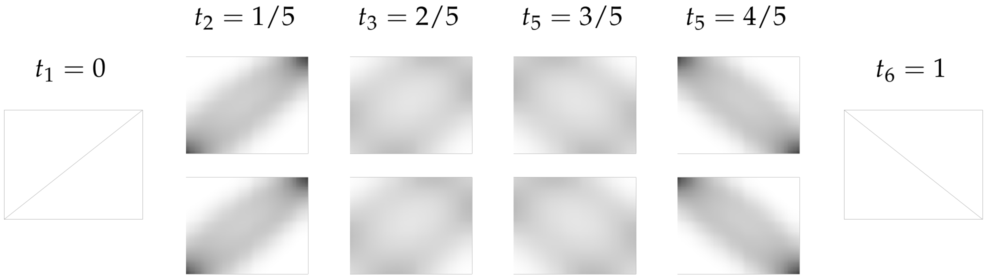

Figure 8. Joint measures $\mathbf{\Pi}_{1,k}$, $k = 2, \cdots, 5$ representing the movement of the particles from initial position $x \in [0, 1]$ (x-axis) to position $x_{t_k} \in [0, 1]$ (y-axis) at time t_k, where $\sigma(x) = 1 - x$. **First row:** Sinkhorn algorithm. **Second row:** NFFT-Sinkhorn algorithm.

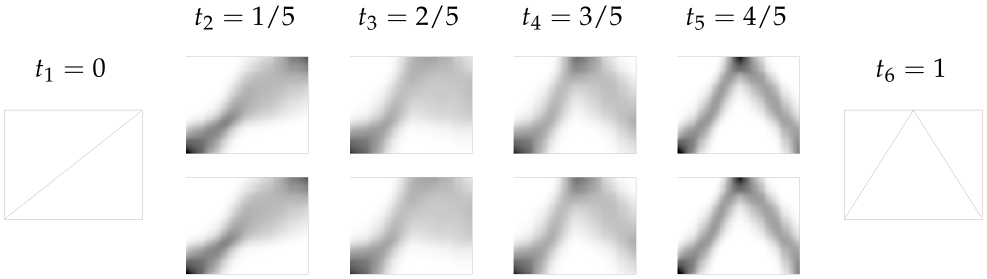

Figure 9. Joint measures as in Figure 8, but with the function $\sigma(x) = \min(2x, 1 - 2x)$.

7. Conclusions

We have proposed the NFFT-Sinkhorn algorithm to solve the MOT$_\eta$ problem efficiently. Assuming that the cost function of the multi-marginal optimal transport decouples according to a tree or a circle, we obtain a linear complexity in K. The complexity of the algorithm with respect to the numbers n_k, $k \in [K]$, of atoms of the discrete marginal measures is further improved using the non-uniform fast Fourier transform. This results in a considerable acceleration in our numerical experiments compared to the usual Sinkhorn algorithm. The tree-structured MOT$_\eta$ problem gives a much better numerical complexity than the circle-structured MOT$_\eta$ problem due to the fact that in the latter case, matrix–matrix products are required in Algorithm 3 instead of just the matrix–vector products of Algorithm 2.

Author Contributions: Both authors have contributed equally to this article. All authors have read and agreed to the published version of the manuscript.

Funding: We gratefully acknowledge funding by the German Federal Ministry of Education and Research BMBF 01|S20053B project SAℓE. Furthermore, we gratefully acknowledge funding by the German Research Foundation DFG (STE 571/19-1, project number 495365311).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data is available at https://github.com/fatima0111/NFFT-Sinkhorn, accessed on 4 August 2022.

Acknowledgments: We would like to thank Gabriele Steidl for insightful discussions about optimal transport. We also thank the anonymous reviewers for making valuable suggestions to improve the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Peyré, G.; Cuturi, M. Computational Optimal Transport: With Applications to Data Science. *Found. Trends Mach. Learn.* **2019**, *11*, 355–607. [CrossRef]
2. Villani, C. *Optimal Transport: Old and New*; Springer: Berlin/Heidelberg, Germany, 2009. [CrossRef]
3. Beier, F.; von Lindheim, J.; Neumayer, S.; Steidl, G. Unbalanced multi-marginal optimal transport. *arXiv* **2021**, arXiv:2103.10854.
4. Bonneel, N.; Peyré, G.; Cuturi, M. Wasserstein barycentric coordinates: Histogram regression using optimal transport. *ACM Trans. Graph.* **2016**, *35*, 71. [CrossRef]
5. Tartavel, G.; Peyré, G.; Gousseau, Y. Wasserstein loss for image synthesis and restoration. *SIAM J. Imaging Sci.* **2016**, *9*, 1726–1755. [CrossRef]
6. Thorpe, M.; Park, S.; Kolouri, S.; Rohde, G.K.; Slepčev, D. A transportation L^p distance for signal analysis. *J. Math. Imaging Vis.* **2017**, *59*, 187–210. [CrossRef] [PubMed]
7. Vogt, T.; Lellmann, J. Measure-valued variational models with applications to diffusion-weighted imaging. *J. Math. Imaging Vis.* **2018**, *60*, 1482–1502. [CrossRef]
8. Carlier, G.; Oberman, A.; Oudet, E. Numerical methods for matching for teams and Wasserstein barycenters. *ESAIM M2AN* **2015**, *49*, 1621–1642. [CrossRef]
9. Galichon, A. *Optimal Transport Methods in Economics*; Princeton University Press: Princeton, NJ, USA, 2016. [CrossRef]
10. Dolinsky, Y.; Soner, H.M. Martingale optimal transport and robust hedging in continuous time. *Probab. Theory Relat. Fields* **2014**, *160*, 391–427. [CrossRef]
11. Dolinsky, Y.; Soner, H.M. Robust hedging with proportional transaction costs. *Financ. Stoch.* **2014**, *18*, 327–347. [CrossRef]
12. Frisch, U.; Matarrese, S.; Mohayaee, R.; Sobolevski, A.N. A reconstruction of the initial conditions of the universe by optimal mass transportation. *Nature* **2002**, *417*, 260–262. [CrossRef]
13. Haasler, I.; Ringh, A.; Chen, Y.; Karlsson, J. Multimarginal optimal transport with a tree-structured cost and the Schrödinger bridge problem. *SIAM J. Control Optim.* **2021**, *59*, 2428–2453. [CrossRef]
14. Kantorovich, L. On the translocation of masses. *Manag. Sci.* **1958**, *5*, 1–4. [CrossRef]
15. Lin, T.; Ho, N.; Cuturi, M.; Jordan, M.I. On the complexity of approximating multimarginal optimal transport. *J. Mach. Learn. Res.* **2022**, *23*, 1–43.
16. Pass, B. Multi-marginal optimal transport and multi-agent matching problems: Uniqueness and structure of solutions. *Discret. Contin. Dyn. Syst.* **2014**, *34*, 1623–1639. [CrossRef]
17. Pass, B. Multi-marginal optimal transport: Theory and applications. *ESAIM M2AN* **2015**, *49*, 1771–1790. [CrossRef]
18. Benamou, J.-D.; Carlier, G.; Nenna, L. A numerical method to solve multi-marginal optimal transport problems with Coulomb cost. In *Splitting Methods in Communication, Imaging, Science, and Engineering*; Springer: Cham, Switzerland, 2016; pp. 577–601.
19. Neumayer, S.; Steidl, G. From optimal transport to discrepancy. In *Handbook of Mathematical Models and Algorithms in Computer Vision and Imaging: Mathematical Imaging and Vision*; Chen, K., Schönlieb, C.-B., Tai, X.-C., Younces, L., Eds.; Springer: Cham, Switzerland, 2021; pp. 1–36. [CrossRef]
20. Terjék, D.; González-Sánchez, D. Optimal transport with f-divergence regularization and generalized Sinkhorn algorithm. *arXiv* **2021**, arXiv:2105.14337.
21. Blondel, M.; Seguy, V.; Rolet, A. Smooth and sparse optimal transport. In Proceedings of Machine Learning Research, Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics, Playa Blanca, Spain, 9–11 April 2018; Volume 84, pp. 880–889.
22. Lorenz, D.A.; Manns, P.; Meyer, C. Quadratically regularized optimal transport. *Appl. Math. Optim.* **2021**, *83*, 1919–1949. [CrossRef]
23. Genevay, A.; Cuturi, M.; Peyré, G.; Bach, F. Stochastic optimization for large-scale optimal transport. In Proceedings of the 30th International Conference on Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016; Curran Associates Inc.: Red Hook, NY, USA, 2016; pp. 3440–3448. [CrossRef]
24. Ammari, H.; Garnier, J.; Millien, P. Backpropagation imaging in nonlinear harmonic holography in the presence of measurement and medium noises. *SIAM J. Imaging Sci.* **2014**, *7*, 239–276. [CrossRef]
25. Altschuler, J.M.; Boix-Adsera, E. Polynomial-time algorithms for multimarginal optimal transport problems with structure. *Math. Program.* **2022**, *in press*. [CrossRef]
26. Benamou, J.-D.; Carlier, G.; Cuturi, M.; Nenna, L.; Peyré, G. Iterative bregman projections for regularized transportation problems. *SIAM J. Sci. Comput.* **2015**, *37*, A1111–A1138. [CrossRef]
27. Koller, D.; Friedman, N. *Probabilistic Graphical Models: Principles and Techniques*; Adaptive Computation and Machine Learning; The MIT Press: Cambridge, MA, USA, 2009.
28. Alaya, M.Z.; Bérar, M.; Gasso, G.; Rakotomamonjy, A. Screening Sinkhorn algorithm for regularized optimal transport. In Proceedings of the 33rd International Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; Curran Associates Inc.: Red Hook, NY, USA, 2019. [CrossRef]
29. Cuturi, M. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems*; Burges, C.J.C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K.Q., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2013; Volume 26.

30. Knopp, P.; Sinkhorn, R. Concerning connegative matrices and doubly stochastic matrices. *Pac. J. Math.* **1967**, *21*, 343–348. [CrossRef]
31. Potts, D.; Steidl, G. Fast summation at nonequispaced knots by NFFTs. *SIAM J. Sci. Comput.* **2003**, *24*, 2013–2037. [CrossRef]
32. Potts, D.; Steidl, G.; Nieslony, A. Fast convolution with radial kernels at nonequispaced knots. *Numer. Math.* **2004**, *98*, 329–351. [CrossRef]
33. Nestler, F.; Pippig, M.; Potts, D. Fast Ewald summation based on NFFT with mixed periodicity. *J. Comput. Phys.* **2015**, *285*, 280–315. [CrossRef]
34. Hielscher, R.; Quellmalz, M. Optimal mollifiers for spherical deconvolution. *Inverse Probl.* **2015**, *31*, 085001. [CrossRef]
35. Alfke, D.; Potts, D.; Stoll, M.; Volkmer, T. NFFT meets Krylov methods: Fast matrix-vector products for the graph Laplacian of fully connected networks. *Front. Appl. Math. Stat.* **2018**, *4*, 61. [CrossRef]
36. Lakshmanan, R.; Pichler, A.; Potts, D. Fast Fourier transform boost for the Sinkhorn algorithm. *arXiv* **2022**, arXiv:2201.07524.
37. Solomon, J.; de Goes, F.; Peyré, G.; Cuturi, M.; Butscher, A.; Nguyen, A.; Du, T.; Guibas, L. Convolutional Wasserstein distances: efficient optimal transportation on geometric domains. *ACM Trans. Graph.* **2015**, *34*, 1–11. [CrossRef]
38. Strössner, C.; Kressner, D. Low-rank tensor approximations for solving multi-marginal optimal transport problems. *arXiv* **2022**, arXiv:2202.07340.
39. Benamou, J.-D.; Carlier, G.; Nenna, L. Generalized incompressible flows, multi-marginal transport and Sinkhorn algorithm. *Numer. Math.* **2019**, *142*, 33–54. [CrossRef]
40. Beiglböck, M.; Léonard, C.; Schachermayer, W. A general duality theorem for the Monge-Kantorovich transport problem. *Studia Math.* **2012**, *209*, 151–167. [CrossRef]
41. Elvander, F.; Haasler, I.; Jakobsson, A.; Karlsson, J. Multi-marginal optimal transport using partial information with applications in robust localization and sensor fusion. *Signal Process.* **2020**, *171*, 107474. [CrossRef]
42. Marino, S.D.; Gerolin, A. An optimal transport approach for the Schrödinger bridge problem and convergence of Sinkhorn algorithm. *J. Sci. Comput.* **2020**, *85*, 27. [CrossRef]
43. Plonka, G.; Potts, D.; Steidl, G.; Tasche, M. *Numerical Fourier Analysis*; Applied and Numerical Harmonic Analysis; Birkhäuser: Basel, Switzerland, 2018. [CrossRef]
44. Dutt, A.; Rokhlin, V. Fast Fourier transforms for nonequispaced data II. *Appl. Comput. Harmon. Anal.* **1995**, *2*, 85–100. [CrossRef]
45. Beylkin, G. On the fast Fourier transform of functions with singularities. *Appl. Comput. Harmon. Anal.* **1995**, *2*, 363–381. [CrossRef]
46. Keiner, J.; Kunis, S.; Potts, D. Using NFFT3—A software library for various nonequispaced fast Fourier transforms. *ACM Trans. Math. Softw.* **2009**, *36*, 19. [CrossRef]
47. Nestler, F. Parameter tuning for the NFFT based fast Ewald summation. *Front. Phys.* **2016**, *4*, 28. [CrossRef]
48. Bassetti, F.; Gualandi, S.; Veneroni, M. On the computation of Kantorovich-Wasserstein distances between 2d-histograms by uncapacitated minimum cost flows. *arXiv* **2018**, arXiv:1804.00445.
49. Cuturi, M.; Doucet, A. Fast computation of wasserstein barycenters. In Proceedings of the 31st International Conference on International Conference on Machine Learning, Beijing, China, 21–26 June 2014; Volume 32, pp. II–685–II–693.
50. Rabin, J.; Peyre, G.; Delon, J.; Bernot, M. Wasserstein barycenter and its application to texture mixing. In *Scale Space and Variational Methods in Computer Vision*; Bruckstein, A.M., Romeny, B.M.t.H., Bronstein, A.M., Bronstein, M.M., Eds.; Springer: Berlin/Heidelberg, Germany, 2012; pp. 435–446.
51. von Lindheim, J. Approximative algorithms for multi-marginal optimal transport and free-support Wasserstein barycenters. *arXiv* **2022**, arXiv:2202.00954.
52. Takezawa, Y.; Sato, R.; Kozareva, Z.; Ravi, S.; Yamada, M. Fixed support tree-sliced Wasserstein barycenter. *arXiv* **2021**, arXiv:2109.03431.
53. Agueh, M.; Carlier, G.. Barycenters in the Wasserstein space. *SIAM J. Math. Anal.* **2011**, *43*, 904–924. [CrossRef]
54. Flamary, R.; Courty, N.; Gramfort, A.; Alaya, M.Z.; Boisbunon, A.; Chambon, S.; Chapel, L.; Corenflos, A.; Fatras, K.; Fournier, N.; et al. Pot: Python optimal transport. *J. Mach. Learn. Res.* **2021**, *22*, 1–8.
55. Brenier, Y. The least action principle and the related concept of generalized flows for incompressible perfect fluids. *J. Amer. Math. Soc.* **1989**, *2*, 225–255. [CrossRef]
56. Brenier, Y. The dual least action problem for an ideal, incompressible fluid. *Arch. Ration. Mech. Anal.* **1993**, *122*, 323–351. [CrossRef]
57. Brenier, Y. Minimal geodesics on groups of volume-preserving maps and generalized solutions of the euler equations. *Comm. Pure Appl. Math* **1997**, *52*, 411–452. [CrossRef]