informs
https://pubsonline.informs.org/journal/moor

MATHEMATICS OF OPERATIONS RESEARCH
Articles in Advance, pp. 1–26
ISSN 0364-765X (print), ISSN 1526-5471 (online)

Scalable Computation of Dynamic Flow Problems via Multimarginal Graph-Structured Optimal Transport

Isabel Haasler,[a,*] **Axel Ringh,**[b,c] **Yongxin Chen,**[d] **Johan Karlsson**[e]

[a] Signal Processing Laboratory 4, École Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland; [b] Department of Mathematical Sciences, Chalmers University of Technology, SE-412 96 Gothenburg, Sweden; [c] Department of Mathematical Sciences, University of Gothenburg, SE-412 96 Gothenburg, Sweden; [d] School of Aerospace Engineering, Georgia Institute of Technology, GA 30332 Atlanta, Georgia; [e] Department of Mathematics, KTH Royal Institute of Technology, SE-100 44 Stockholm, Sweden

*Corresponding author

Contact: isabel.haasler@epfl.ch, https://orcid.org/0000-0002-2484-0181 (IH); axelri@chalmers.se, https://orcid.org/0000-0002-9778-1426 (AR); yongchen@gatech.edu, https://orcid.org/0000-0002-1459-6365 (YC); johan.karlsson@math.kth.se, https://orcid.org/0000-0001-5158-9255 (JK)

Received: June 15, 2021
Revised: October 12, 2022
Accepted: May 1, 2023
Published Online in Articles in Advance: June 15, 2023

MSC2020 Subject Classifications: Primary: 49Q22, 90-08, 90C35, 90C08, 49M29

https://doi.org/10.1287/moor.2021.148

Abstract. In this work, we develop a new framework for dynamic network flow problems based on optimal transport theory. We show that the dynamic multicommodity minimum-cost network flow problem can be formulated as a multimarginal optimal transport problem, where the cost function and the constraints on the marginals are associated with a graph structure. By exploiting these structures and building on recent advances in optimal transport theory, we develop an efficient method for such entropy-regularized optimal transport problems. In particular, the graph structure is utilized to efficiently compute the projections needed in the corresponding Sinkhorn iterations, and we arrive at a scheme that is both highly computationally efficient and easy to implement. To illustrate the performance of our algorithm, we compare it with a state-of-the-art linear programming (LP) solver. We achieve good approximations to the solution at least one order of magnitude faster than the LP solver. Finally, we showcase the methodology on a traffic routing problem with a large number of commodities.

Funding: This work was supported by KTH Digital Futures, Knut och Alice Wallenbergs Stiftelse [Grants KAW 2018.0349, KAW 2021.0274, the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation], Vetenskapsrådet [Grant 2020-03454], and the National Science Foundation [Grants 1942523 and 2206576].

Keywords: multimarginal optimal transport • dynamic network flow • multicommodity network flow • Sinkhorn's method • Computational methods • traffic routing

1. Introduction

Many phenomena in today's society can be modeled as large-scale transportation or flow problems, and new technological advances create the need for solving larger and larger problems. An example is the introduction of self-driving cars to the road network, which will create both new opportunities and new challenges (Levinson et al. [44], Pasquale et al. [50]). Increasing automation and communication between vehicles will result in very large systems where all vehicles need to be routed simultaneously, taking into account destinations, vehicle properties, and urgency (Carlino et al. [13]). Another challenge is to direct large crowds in, for example, transit areas in airports, subways, or event venues (Aronson [2], Haghani and Oh [35], Yamada [61]), which is particularly critical for evacuation scenarios in the case of emergencies but also essential for everyday use.

Many of these problems can be modeled as large-scale dynamic network flow problems (Aronson [2], Bertsimas and Patterson [10], Kennington [41]). The most common strategy for handling such problems is to convert the dynamic flow problem to a static flow problem on a time-expanded network, and this strategy goes back to the classical work of Ford and Fulkerson [26]. In addition to this, there are typically several classes of groups of agents with heterogeneous properties and objectives in the system. For instance, each agent in a traffic network drives a vehicle with certain properties, and the objective is typically to reach a certain destination with a certain degree of urgency. Similar problems appear in air traffic planning, railroad traffic scheduling, communication, and logistics, and they are often treated as multicommodity flow problems over networks (Aronson [2], Bertsimas and Patterson [10], Haghani and Oh [35], Kennington [41]). Although such problems are usually formulated as linear programming (LP) problems, for real applications the corresponding optimization problems are often too large to be handled by standard methods. Specialized methods exploit the structure of multicommodity flow problems using, for example, column generation methods. These include price-directive decomposition (Jones et al. [38]), resource-directive decomposition (Kennington and Shalaby [40], McBride [48]), and basis partitioning methods (Farvolden et al. [24]). However, it has been reported that these methods typically decrease the solution time of standard (LP) solvers by at most one order of magnitude (Barnhart et al. [4], Khodayifar [42], Retvdri et al. [54]).

During the last few decades, there has been considerable development in the field of optimal transport theory. Traditionally, the optimal transport problem addresses a static scenario where one given distribution is transported to another, and this problem has been extensively used in areas such as economics and logistics (Villani [59]). There has recently been a rapid advancement of theory and applications for optimal transport, in particular toward applications in imaging, statistics, and machine learning (see Peyré and Cuturi [53] and references therein) and in systems and control (Benamou and Brenier [6], Chen et al. [15]), which has led to a mature framework with computationally efficient algorithms (Peyré and Cuturi [53]) that can be used to address a wide range of problems. The optimal transport problem is a linear program, but the number of variables often makes it intractable to solve with general-purpose optimization methods for large-size problems. However, a recent computational breakthrough in this area builds on introducing an entropic barrier term in the objective function. The resulting optimization problem can then be solved efficiently using the so-called Sinkhorn iterations (Cuturi [20]). This allows for computing an approximate solution of large transportation problems and has opened up the field for new applications where no computationally feasible method previously existed.

The optimal transport framework has in some cases been used for modeling several kinds of interacting classes (e.g., for transport of multiple species (Bacon [3], Chen et al. [16]) or flows with several phases (Benamou et al. [7])). In this paper, we will build on some of these results, and we propose to use a generalization of the optimal transport problem with several marginals to address multicommodity flow problems. This multimarginal optimal transport problem (Gangbo and Świech [28], Pass [51], Rüschendorf [55], Rüschendorf and Uckelmann [56]) is computationally challenging because the number of variables grows exponentially in the number of marginals. Even though entropy regularization methods have been derived for the multimarginal optimal transport problem (Benamou et al. [8]), the cost for each iteration still grows exponentially in the number of marginals (see Lin et al. [45] for computational complexity bounds). However, in many cases the cost function has a structure that can be utilized for efficient computations, such as, for example, in barycenter, information fusion, and tracking problems (Benamou et al. [8], Elvander et al. [23], Haasler et al. [33]).

In this paper, we show that the dynamic flow problem can be formulated as a structured multimarginal optimal transport problem. This structure can be visualized in a graph where the set of nodes corresponds to the marginals and where there is an edge between two nodes if there is a cost term or a constraint that depends jointly on the two nodes. For the single-commodity case, this structure is a path graph with one node for each time point that represents the flow in the network at that time. For the dynamic multicommodity network flow problem, there is one additional node in the graph that represents the distribution over the different commodity classes. The solution to this optimal transport problem then describes a joint distribution, which consists of the optimal flow for all commodities in the dynamic network problem.

We consider the corresponding entropy-regularized approximation of this problem, and by utilizing the structure in the cost function, we derive methods for solving this problem. Many of the classical methods for dynamic flow problems consider standard network flow methods on the time-expanded network. By instead formulating this problem as a multimarginal optimal transport problem, we can more efficiently utilize the sequential structure without explicitly setting up the time-expanded network. This results in an elegant and easily implementable method. We illustrate experimentally that this method is computationally competitive with state-of-the-art methods, and then, we apply it to a traffic routing problem.

The rest of the paper is structured as follows. Section 2 summarizes background material on dynamic multicommodity network flows and multimarginal optimal transport. In Section 3, we explain how to formulate network flow problems as structured multimarginal optimal transport problems. Based on this, we develop numerical schemes to solve the problems in Section 4. Finally, in Section 5, we compare the performance of our methods with a commercial LP solver and showcase it in a traffic routing application.

2. Background

In this section, we review background on the two central topics of this paper: dynamic multicommodity network flows and multimarginal optimal transport. We also use this section to set up notation. In particular, boldfaced letters are used throughout to denote tensors, and $\otimes$ denotes the tensor (outer) product (e.g., for vectors $v_1 \in \mathbb{R}^{n_1}$ and $v_2 \in \mathbb{R}^{n_2}$, we have that $v_1 \otimes v_2 \in \mathbb{R}^{n_1 \times n_2}$ and $(v_1 \otimes v_2)_{ij} = (v_1)_i (v_2)_j$). Moreover, by $\mathbf{1}$, we denote a column vector of ones of appropriate size; by $\mathbb{R}_+$, we denote the nonnegative real numbers, and we use $\overline{\mathbb{R}}_+ = \mathbb{R}_+ \cup \{\infty\}$ and $\overline{\mathbb{R}} =$

$\mathbb{R} \cup \{\infty\} \cup \{-\infty\}$ to denote the extended nonnegative real line and extended real line, respectively. Throughout, we will adopt the convention that $0 \cdot \infty = 0$. Finally, by $\exp(\cdot)$, $\log(\cdot)$, $\odot$, $\oslash$, and $\min(\cdot,\cdot)$, we denote elementwise exponential, logarithm, product, division, and minimum, respectively.

2.1. Minimum-Cost Network Flow Problems

A minimum-cost network flow problem is to determine a flow from sources to sinks with minimum cost (Bertsekas and Tseng [9], Ford and Fulkerson [27]). More specifically, the flow is defined on a network $\mathcal{N} = (\mathcal{V}, \mathcal{E})$ with vertices $\mathcal{V}$ and directed edges $\mathcal{E}$, and the sources and sinks are sets of edges[1] $\mathcal{S}^+ \subset \mathcal{E}$ and $\mathcal{S}^- \subset \mathcal{E}$. Let each source $e \in \mathcal{S}^+$ be equipped with a supply $r_e^+ \in \mathbb{R}_+$ and each sink $e \in \mathcal{S}^-$ with a demand $r_e^- \in \mathbb{R}_+$, and we assume that the total supply matches the total demand (i.e., that $\sum_{e\in\mathcal{S}^+} r_e^+ - \sum_{e\in\mathcal{S}^-} r_e^- = 0$). In addition, let each edge $e \in \mathcal{E}$ be assigned a cost $c_e \in \mathbb{R}_+$ of transporting a unit of flow on that edge. The goal of minimum cost-flow problems is to transport the flow from the sources to the sinks with minimal total transporting cost. We also include capacity constraints, which require that the total flow on an edge is limited by the edge capacity $d_e \in \mathbb{R}_+$ on $e \in \mathcal{E}$.

There are two standard formulations for the network flow problem. One is the arc-chain formulation, where one optimizes over a set of flow paths (arc chains) from sources to sinks (Ford and Fulkerson [27], Tomlin [57]). This is the main formulation considered in this work and is described in detail. Another common formulation is the node-edge formulation, where one seeks the optimal amount of flow over each edge while maintaining flow balance in each node. For more details on this formulation and a comparison of both formulations, we refer the reader to Ford and Fulkerson [27] and Tomlin [57].

2.1.1. The Arc-Chain Formulation.

Given a network $\mathcal{N} = (\mathcal{V}, \mathcal{E})$, a path is a sequence of edges that joins two vertices such that all edges and all visited vertices are distinct (i.e., they occur at most once in the sequence) (Diestel [21, p. 6]). A path is thus a subgraph, which we denote by p, and is defined by a list of edges $(p_1, p_2, \ldots, p_N)$, where $p_j \in \mathcal{E}$ denotes the jth element of the path for $j = 1, \ldots, N$. Here, N is called the length of the path p. Moreover, because p is a path, the edge p_j ends in the initial node of p_{j+1} for $j = 1, \ldots, N-1$.

In the arc-chain formulation, we consider the paths or arc chains, which start in a source and end in a sink. Let $\mathcal{P}$ denote the set of all such paths, where the first element lies in $\mathcal{S}^+$ and its last element lies in $\mathcal{S}^-$. Moreover, let $\mathcal{P}_e^+$ denote the paths starting from the edge $e \in \mathcal{S}^+$, and let $\mathcal{P}_e^-$ denote the paths ending in the edge $e \in \mathcal{S}^-$. The cost of a path $p \in \mathcal{P}$ is the sum of the costs of its edges $c_p = \sum_{e\in p} c_e$. Next, let x_p denote the amount of flow associated with path $p \in \mathcal{P}$. Then, the arc-chain formulation of the minimum-cost network flow problem reads

$$\begin{aligned}
\underset{x_p\in\mathbb{R}_+,\, p\in\mathcal{P}}{\text{minimize}} \quad & \sum_{p\in\mathcal{P}} c_p x_p \\
\text{subject to} \quad & \sum_{p\in\mathcal{P}_e^+} x_p = r_e^+, && \text{for } e \in \mathcal{S}^+, \\
& \sum_{p\in\mathcal{P}_e^-} x_p = r_e^-, && \text{for } e \in \mathcal{S}^-, \\
& \sum_{p\in\mathcal{P}} \delta_{e\in p} x_p \le d_e, && \text{for } e \in \mathcal{E},
\end{aligned} \tag{1}$$

where $\delta_{e\in p} = 1$ if the edge e is part of path p and $\delta_{e\in p} = 0$ otherwise. Here, the objective function corresponds to the total cost of the flow. The first two sets of constraints guarantee that the supply and demand for all sources and sinks are satisfied, and the last set of constraints enforces that the flow on each edge does not exceed the given capacity.

2.1.2. Multicommodity Network Flow.

The extension to multicommodity network flow problems deals with the case where there are multiple commodities present in the network (Ford and Fulkerson [25], Hall et al. [36], Kennington [41], Tomlin [57], Wang [60]). Here, we let L denote the number of commodities, and let c_e^ℓ denote the cost of a unit flow on edge $e \in \mathcal{E}$ of commodity ℓ, for $\ell = 1, \ldots, L$. The supply and demand typically depend on the commodity; thus, each commodity ℓ has specified sources $\mathcal{S}^{\ell,+} \in \mathcal{E}$ with supplies $r_e^{\ell,+}$ for $e \in \mathcal{S}^{\ell,+}$ and sinks $\mathcal{S}^{\ell,-} \in \mathcal{E}$ with demands $r_e^{\ell,-}$ for $e \in \mathcal{S}^{\ell,-}$. Moreover, for each commodity $\ell = 1, \ldots, L$, let $\mathcal{P}^\ell$ denote the sets of paths from the sources to the sinks, let $\mathcal{P}_e^{\ell,+}$ denote the paths starting in $e \in \mathcal{S}^{\ell,+}$, and let $\mathcal{P}_e^{\ell,-}$ denote the paths ending in $e \in \mathcal{S}^{\ell,-}$. The cost of a unit flow of commodity ℓ on a path $p \in \mathcal{P}$ is the sum of the corresponding costs of the edges in the path $c_p^\ell = \sum_{e\in p} c_e^\ell$. Next, by letting x_p^ℓ denote the amount of flow of commodity ℓ on path p, the minimum-cost multicommodity network flow problem in arc-chain formulation reads

$$
\begin{aligned}
\underset{\substack{x_p^\ell \in \mathbb{R}_+,\ p \in \mathcal{P}^\ell \\ \ell = 1, \dots, L}}{\text{minimize}} \quad & \sum_{\ell=1}^{L} \sum_{p \in \mathcal{P}^\ell} c_p^\ell x_p^\ell \\
\text{subject to} \quad & \sum_{p \in \mathcal{P}_e^{\ell,+}} x_p^\ell = r_e^{\ell,+}, \qquad \text{for } e \in \mathcal{S}^{\ell,+}, \quad \ell = 1, \dots, L, \\
& \sum_{p \in \mathcal{P}_e^{\ell,-}} x_p^\ell = r_e^{\ell,-}, \qquad \text{for } e \in \mathcal{S}^{\ell,-}, \quad \ell = 1, \dots, L, \\
& \sum_{\ell=1}^{L} \sum_{p \in \mathcal{P}^\ell} \delta_{e \in p} x_p^\ell \le d_e, \qquad \text{for } e \in \mathcal{E}.
\end{aligned} \tag{2}
$$

Here, the first two sets of constraints guarantee that the demand and supply for all commodities are satisfied. The third set of constraints enforces that the flow on each edge does not exceed the given capacity. In particular, note that the multicommodity Problem (2) with only one commodity (i.e., $L = 1$) boils down to the single-commodity Problem (1).

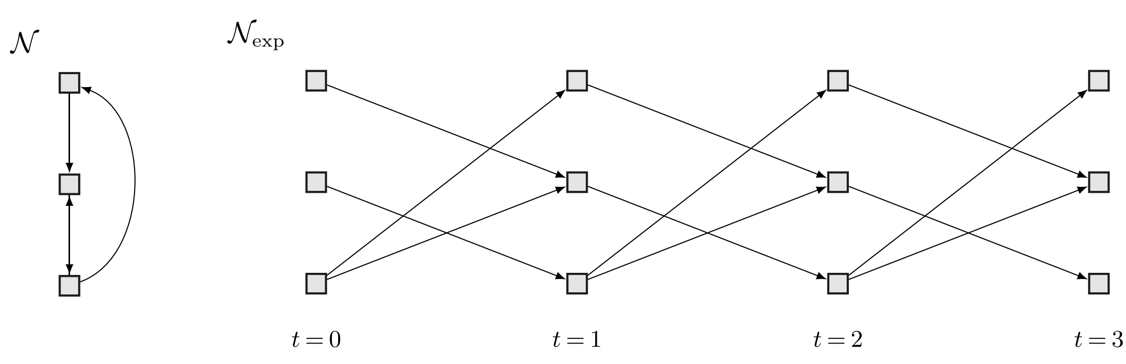
2.1.3. Dynamic Network Flow. In this work, we consider dynamic flows, also called flows over time, where the time that it takes for the flow to travel in the network is taken into account (Aronson [2], Ford and Fulkerson [26], Hall et al. [36]). In this work, we develop efficient methods that exploit the temporal structure. For this to work, we need to assume synchronous traveling times for all edges, but on the other hand, the efficient methods allow for handling problems with large networks and fine time discretization.

More precisely, we consider a flow problem on the network $\mathcal{N} = (\mathcal{V}, \mathcal{E})$ over the time interval 0 to $\mathcal{T}$. The problem is to transport a given flow at time 0 through the network to a final flow at time $\mathcal{T}$ with minimal cost while satisfying capacity constraints at all time points. We consider the discretized problem on the time steps $0, 1, \dots, \mathcal{T}$. Dynamic flow problems are typically solved as a static problem on the time-expanded network (Ford and Fulkerson [26]). The time-expanded network $\mathcal{N}_{\text{exp}}$ is constructed by considering $\mathcal{T} + 1$ copies of the vertices $\mathcal{V}$, denoted by $\mathcal{V}_0, \dots, \mathcal{V}_\mathcal{T}$. Here, the copy $\mathcal{V}_t$ is associated with time instance t in the time-expanded network, and we denote these nodes by (t, v), where $v \in \mathcal{V}$ in the original network.

The edges of $\mathcal{N}_{\text{exp}}$ connect nodes corresponding to consecutive time instances according to the edges $\mathcal{E}$ in the original network: that is, $\mathcal{E}_{\text{exp}} = \cup_{t=1}^{\mathcal{T}} \mathcal{E}_t$, where $\mathcal{E}_t$ consists of the directed edges $((t-1, v_{t-1}), (t, v_t))$ where $(v_{t-1}, v_t) \in \mathcal{E}$ for $t = 1, \dots, \mathcal{T}$. The capacities and costs on these added edges are defined to be the same as the corresponding[2] edges in the original network $\mathcal{N}$. The time-expanded network is illustrated for a simple example in Figure 1.

To express the dynamic flow problem in arc-chain formulations similarly to (1) and (2), a path p is as before a tuple of edges $(p_1, \dots, p_\mathcal{T})$. Its element $p_t \in \mathcal{E}_t$ denotes the edge, which the paths flow takes in the time interval $[t-1, t]$. In the setting of one commodity, let $\mathcal{P}$ denote the set of feasible paths in the time-expanded network $\mathcal{N}_{\text{exp}}$ (i.e., $p \in \mathcal{P}$ if p is a path that starts in a source, $p_1 \in \mathcal{S}^+$, and ends in a sink, $p_\mathcal{T} \in \mathcal{S}^-$). The corresponding cost of unit flow on the path $p \in \mathcal{P}$ is then $c_p = \sum_{t=1}^{\mathcal{T}} c_{p_t}$. The dynamic minimum-cost network flow problem can then

Figure 1. A network with three nodes and its time-expanded network for $\mathcal{T} = 3$ time steps.



be written as

$$\underset{x_p\in\mathbb{R}_+,\, p\in\mathcal{P}}{\text{minimize}} \sum_{p\in\mathcal{P}} c_p x_p \tag{3a}$$

$$\text{subject to} \sum_{p\in\mathcal{P}} \delta_{e=p_1} x_p = r_e^+, \quad \text{for } e\in\mathcal{S}^+, \tag{3b}$$

$$\sum_{p\in\mathcal{P}} \delta_{e=p_{\mathcal{T}}} x_p = r_e^-, \quad \text{for } e\in\mathcal{S}^-, \tag{3c}$$

$$\sum_{p\in\mathcal{P}} \delta_{e=p_t} x_p \le d_e, \quad \text{for } e\in\mathcal{E}, \quad t=2,\ldots,\mathcal{T}-1. \tag{3d}$$

Note that the network flow Problem (1) on the time-expanded network $\mathcal{N}_{\text{exp}}$ corresponds to (3) line by line.

To formulate the multicommodity counterpart of the dynamic flow Problem (3), let $\mathcal{P}^\ell$ denote the set of feasible paths in the time-expanded network $\mathcal{N}_{\text{exp}}$ for commodity $\ell=1,\ldots,L$. The corresponding cost of unit flow on the path for commodity ℓ is then $c_p^\ell = \sum_{t=1}^{\mathcal{T}} c_{p_t}^\ell$ for a path $p\in\mathcal{P}^\ell$, and the dynamic minimum-cost multicommodity network flow problem reads

$$\begin{aligned}
\underset{\substack{x_p^\ell\in\mathbb{R}_+,\, p\in\mathcal{P}^\ell\\ \ell=1,\ldots,L}}{\text{minimize}} \quad & \sum_{\ell=1}^{L}\sum_{p\in\mathcal{P}^\ell} c_p^\ell x_p^\ell\\
\text{subject to} \quad & \sum_{p\in\mathcal{P}^\ell} \delta_{e=p_1} x_p^\ell = r_e^{\ell,+}, \quad \text{for } e\in\mathcal{S}^{\ell,+}, \quad \ell=1,\ldots,L,\\
& \sum_{p\in\mathcal{P}^\ell} \delta_{e=p_{\mathcal{T}}} x_p^\ell = r_e^{\ell,-}, \quad \text{for } e\in\mathcal{S}^{\ell,-}, \quad \ell=1,\ldots,L,\\
& \sum_{\ell=1}^{L}\sum_{p\in\mathcal{P}^\ell} \delta_{e=p_t} x_p^\ell \le d_e, \quad \text{for } e\in\mathcal{E}, \quad t=2,\ldots,\mathcal{T}-1;
\end{aligned} \tag{4}$$

see Khodayifar [42] for a similar problem formulation.

A problem with the arc-chain formulations is that the number of variables, corresponding to possible paths, grows exponentially with $\mathcal{T}$. Thus, standard linear programming methods are not applicable when $\mathcal{T}$ is large. A way to circumvent this issue is to use specialized solvers building on, for example, column generation or to instead consider the corresponding node-edge formulations of the problem (cf. Ford and Fulkerson [27], Tomlin [57]). In this work, we take a different approach that builds on formulating the problem as an optimal transport problem that utilizes the structure in the arc-chain formulation.

2.2. Optimal Transport

The optimal transport problem is to find a mapping that moves the mass from one distribution to another with minimal cost based on an underlying metric (Villani [59]). In this paper, we consider the discrete setting where the two distributions are represented by two nonnegative vectors $\mu_1\in\mathbb{R}_+^{n_1}$, $\mu_2\in\mathbb{R}_+^{n_2}$ with equal mass. In this setting, the transport cost is defined in terms of a underlying nonnegative cost matrix $C\in\overline{\mathbb{R}}_+^{n_1\times n_2}$, where C_{ij} denotes the cost[3] of moving a unit mass from position i to j. Analogously, a transport plan $M\in\mathbb{R}_+^{n_1\times n_2}$ is a nonnegative matrix, where M_{ij} represents the amount of mass moved from i to j. The optimal transport plan from μ_1 to μ_2 is then a minimizing solution of

$$\begin{aligned}
\underset{M\in\mathbb{R}_+^{n_1\times n_2}}{\text{minimize}} \quad & \text{trace}(C^T M)\\
\text{subject to} \quad & M\mathbf{1} = \mu_1,\\
& M^T\mathbf{1} = \mu_2.
\end{aligned} \tag{5}$$

Multimarginal optimal transport extends the concept of the classical optimal transport Problem (5) to the setting with a set of marginals $\mu_t\in\mathbb{R}_+^{n_t}$, for $t=1,\ldots,\mathcal{T}$, where $\mathcal{T}\ge 2$ (Benamou et al. [8], Elvander et al. [23], Haasler et al. [33], Pass [51]). In this setting, the transport cost and transport plan are described by tensors $\mathbf{C}\in\overline{\mathbb{R}}_+^{n_1\times n_2\ldots\times n_{\mathcal{T}}}$ and

$\mathbf{M} \in \mathbb{R}_+^{n_1 \times n_2 \cdots \times n_{\mathcal{T}}}$. Here, $\mathbf{C}_{i_1 \ldots i_{\mathcal{T}}}$ denotes the unit cost associated with the tuple $(i_1, \ldots, i_{\mathcal{T}})$, and $\mathbf{M}_{i_1 \ldots i_{\mathcal{T}}}$ denotes the amount of mass associated with this tuple. Then, the total transportation cost for a given transport plan $\mathbf{M}$ is

$$\langle \mathbf{C}, \mathbf{M} \rangle = \sum_{i_1, \ldots, i_{\mathcal{T}}} \mathbf{C}_{i_1 \ldots i_{\mathcal{T}}} \mathbf{M}_{i_1 \ldots i_{\mathcal{T}}}.$$

Moreover, $\mathbf{M}$ is a transport plan between the desired marginals if its projections on the marginals satisfy $P_t(\mathbf{M}) = \mu_t$, for $t = 1, \ldots, \mathcal{T}$, where the projection on the tth marginal is defined by

$$(P_t(\mathbf{M}))_{i_t} := \sum_{i_1, \ldots, i_{t-1}, i_{t+1}, \ldots, i_{\mathcal{T}}} \mathbf{M}_{i_1 \ldots i_{t-1} i_t i_{t+1} \ldots i_{\mathcal{T}}}. \tag{6}$$

The discrete multimarginal optimal transport problem thus reads

$$\begin{aligned} \underset{\mathbf{M} \in \mathbb{R}_+^{n_1 \times \cdots \times n_{\mathcal{T}}}}{\text{minimize}} \quad & \langle \mathbf{C}, \mathbf{M} \rangle \\ \text{subject to} \quad & P_t(\mathbf{M}) = \mu_t, \ \text{ for } t \in \Gamma. \end{aligned} \tag{7}$$

Here, Γ is an index set that describes the set of constrained marginals. In the original multimarginal optimal transport formulation, constraints are typically given on all marginals (i.e., for the index set $\Gamma = \{1, 2, \ldots, \mathcal{T}\}$). However, in this work, we typically consider the case where constraints are only imposed on a subset of marginals (i.e., $\Gamma \subset \{1, 2, \ldots, \mathcal{T}\}$) or when some of the constraints are inequality constraints.

Note that the standard bimarginal optimal transport Problem (5) is a special case of the multimarginal optimal transport Problem (7), where $\mathcal{T} = 2$ and $\Gamma = \{1, 2\}$. It is also worth noting that the bimarginal optimal transport problem can be interpreted as a minimum-cost network flow problem. However, this interpretation does in general not extend to the multimarginal case (Lin et al. [46]). In this work, we show how to formulate any dynamic network flow problem as a multimarginal optimal transport problem with a structured cost tensor.

2.2.1. Sinkhorn Iterations. Although linear, the number of variables in the multimarginal optimal transport Problem (7) is often too large to be solved directly. A popular approach for the bimarginal setting to bypass the size of the problem has been to add a regularizing entropy term to the objective (Cuturi [20]). In principle, the same approach can be used also for the multimarginal case. With the entropy term

$$D(\mathbf{M}) = \sum_{i_1, \ldots, i_{\mathcal{T}}} (\mathbf{M}_{i_1 \ldots i_{\mathcal{T}}} \log(\mathbf{M}_{i_1 \ldots i_{\mathcal{T}}}) + \mathbf{M}_{i_1 \ldots i_{\mathcal{T}}} - 1), \tag{8}$$

the entropy-regularized multimarginal optimal transport problem is defined as

$$\begin{aligned} \underset{\mathbf{M} \in \mathbb{R}_+^{n_1 \times \cdots \times n_{\mathcal{T}}}}{\text{minimize}} \quad & \langle \mathbf{C}, \mathbf{M} \rangle + \epsilon D(\mathbf{M}) \\ \text{subject to} \quad & P_t(\mathbf{M}) = \mu_t, \ \text{ for } t \in \Gamma, \end{aligned} \tag{9}$$

where $\epsilon > 0$ is a small regularization parameter. The introduction of the entropy term in Problem (9) allows for expressing the optimal solution $\mathbf{M}$ in terms of Lagrange dual variables, which may be computed by Sinkhorn iterations (Benamou et al. [8], Nenna [49]). In particular, it can be shown that the optimal solution of (9) is of the form (Elvander et al. [23])

$$\mathbf{M} = \mathbf{K} \odot \mathbf{U}, \tag{10}$$

where $\mathbf{K} = \exp(-\mathbf{C}/\epsilon)$ and where $\mathbf{U}$ can be decomposed as

$$\mathbf{U} = u_1 \otimes u_2 \otimes \cdots \otimes u_{\mathcal{T}}. \tag{11}$$

Here, the vectors $u_t \in \mathbb{R}_+^{n_t}$, for $t = 1, 2, \ldots, \mathcal{T}$, are given by

$$u_t = \begin{cases} \exp(\lambda_t/\epsilon), & \text{if } t \in \Gamma \\ \mathbf{1}, & \text{else,} \end{cases} \tag{12}$$

where $\lambda_t \in \overline{\mathbb{R}}^{n_t}$ for $t \in \Gamma$ are optimal dual variables in the dual problem of (9). This dual problem takes the form

$$\underset{\lambda_t \in \overline{\mathbb{R}}^{n_t},\, t \in \Gamma}{\text{maximize}} \quad -\epsilon \langle \mathbf{K}, \mathbf{U} \rangle + \sum_{t \in \Gamma} \lambda_t^T \mu_t, \tag{13}$$

where $\mathbf{U}$ depends on $\{\lambda_t\}_{t \in \Gamma}$ as specified in (11) and (12). For details, the reader is referred to, for example, Benamou et al. [8] and Elvander et al. [23].

The Sinkhorn scheme for finding $\mathbf{U}$ in (11) is to iteratively update u_t according to

$$u_t \leftarrow u_t \odot \mu_t \oslash P_t(\mathbf{K} \odot \mathbf{U}), \tag{14}$$

for all $t \in \Gamma$. This scheme may, for instance, be derived as Bregman projections (Benamou et al. [8]) or a block-coordinate ascent in the dual (13) (Elvander et al. [23], Karlsson and Ringh [39], Tseng [58]). As a result, global convergence of the Sinkhorn scheme (14) is guaranteed (Bauschke and Lewis [5], Luo and Tseng [47], Tseng [58]). The computational bottleneck of the Sinkhorn iterations (14) is computing the projections $P_t(\mathbf{K} \odot \mathbf{U})$, for $t \in \Gamma$, which in general, scale exponentially in $\mathcal{T}$. In fact, even storing the tensor $\mathbf{M}$ is a challenge as it consists of $\prod_{t=1}^{\mathcal{T}} n_t$ elements. However, in many cases of interest, structures in the cost tensors can be exploited to perform the sum operations in (6) in an appropriate order, which makes the computation of the projections feasible (Benamou et al. [8], Elvander et al. [23], Haasler et al. [31], Haasler et al. [33], Haasler et al. [34]). More precisely, in many applications, the tensor $\mathbf{K} \odot \mathbf{U}$ factorizes such that it can be described by a graph $G = (V, E)$, where the vertices V correspond to the tensor marginals and its dependencies are described by the set of edges E. The projections (6) can then be computed efficiently by first eliminating the variables (i.e., performing the sum operations) for the vertices that have few dependencies. For instance, when the tensor $\mathbf{K} \odot \mathbf{U}$ factorizes according to a tree structure, the projections (6) can be computed by first eliminating the variables corresponding to the trees leaves and successively moving down the branches. Computing the projections requires then only matrix-vector multiplications, where the matrices are at most of size $\max_t(n_t)$ (Haasler et al. [33], Haasler et al. [34]). In the case of more complex graphs, a similar approach can be utilized, but computations become more expensive. For instance, in case the graph is a cycle, the complexity is increased by a factor of $\max_t(n_t)$ as compared with the tree setting (Benamou et al. [8], Haasler et al. [31]).

3. Network Flow Problems via Optimal Transport

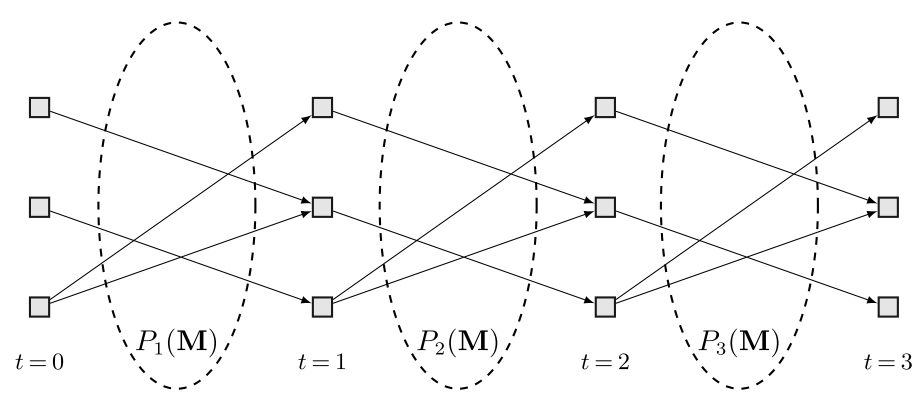
In this section, we introduce a reformulation of the dynamic minimum-cost flow problem as a multimarginal optimal transport Problem (7). In the single-commodity, case this optimal transport problem has a path structure. The multicommodity case can be expressed as several single-commodity problems, which are coupled through the capacity constraints. Alternatively, this can be set up as one multimarginal optimal transport problem, where the cost function decouples as a graph that contains cycles.

3.1. The Dynamic Minimum-Cost Flow Problem

Let $\mathcal{N}_{\text{exp}}$ be the time expansion of the network $\mathcal{N}$ for the time steps $t = 0, \ldots, \mathcal{T}$, and let $\mathcal{P}$ denote the set of feasible paths in $\mathcal{N}_{\text{exp}}$. In order to solve an arc-chain formulation of a flow problem on this network, one has to identify all paths in this set. Clearly, the set of feasible paths $\mathcal{P}$ is a subset of the set $\tilde{\mathcal{P}} = \{(i_1, \ldots, i_{\mathcal{T}}) : i_t \in \mathcal{E}_t \text{ for } t = 1, \ldots, \mathcal{T}\}$, which contains all combinations of $\mathcal{T}$ edges in $\mathcal{E}$. In fact, the set $\tilde{\mathcal{P}}$ is generally much larger than $\mathcal{P}$ because it lifts the set of feasible paths to the set of all "paths" possible from purely combinatorial considerations (ignoring the graph structure).

However, using this representation, the network flow can be described by a tensor $\mathbf{M} \in \mathbb{R}_+^{n^{\mathcal{T}}}$, where $n = |\mathcal{E}|$ and where the element $\mathbf{M}_{i_1, \ldots, i_{\mathcal{T}}}$ denotes the amount of flow on the path $(i_1, \ldots, i_{\mathcal{T}})$. The vector $P_t(\mathbf{M}) \in \mathbb{R}_+^n$, where the projection operator is defined as in (6), then describes the flow distribution over the edges between time $t-1$ and t, as illustrated in Figure 2. That is, its element $P_t(\mathbf{M})_i$ denotes the amount of flow over edge $i \in \mathcal{E}_t$.

Figure 2. Illustration of the optimal transport tensor $\mathbf{M}$ in the time-expanded network from Figure 1. The tensors marginal $P_t(\mathbf{M})$ describes the distribution of flow over the edges in the time interval $(t-1, t)$.



Similarly, the evolution of flow between time intervals (t_1-1, t_1) and (t_2-1, t_2) is described by the bimarginal projections $P_{t_1,t_2}(\mathbf{M}) \in \mathbb{R}_+^{n\times n}$, which are defined as

$$(P_{t_1,t_2}(\mathbf{M}))_{i_{t_1} i_{t_2}} = \sum_{i_1,\dots,i_{\mathcal{T}} \setminus \{i_{t_1}, i_{t_2}\}} \mathbf{M}_{i_1 \dots i_{\mathcal{T}}}. \tag{15}$$

That is, the element $(P_{t_1,t_2}(\mathbf{M}))_{ij}$ describes the amount of flow that is in edge i at time t_1 and that is in edge j at time t_2. Let $c \in \mathbb{R}_+^n$, where c_i denotes the cost of a unit flow on edge $i \in \mathcal{E}$, and let $C \in \overline{\mathbb{R}}_+^{n\times n}$ encode the network topology (i.e., $C_{ij} = 0$ if edge i leads to[4] edge j and $C_{ij} = \infty$ otherwise). Then, we define the cost of a transport plan $\mathbf{M}$ as

$$\sum_{t=1}^{\mathcal{T}} c^T P_t(\mathbf{M}) + \sum_{t=1}^{\mathcal{T}-1} \operatorname{trace}(C^T P_{t,t+1}(\mathbf{M})) = \langle \mathbf{C}, \mathbf{M} \rangle, \tag{16}$$

where the tensor $\mathbf{C} \in \overline{\mathbb{R}}_+^{n\times n \dots \times n}$ is defined as

$$\mathbf{C}_{i_1 \dots i_{\mathcal{T}}} = \sum_{t=1}^{\mathcal{T}} c_{i_t} + \sum_{t=1}^{\mathcal{T}-1} C_{i_t i_{t+1}}. \tag{17}$$

Note that $\langle \mathbf{C}, \mathbf{M} \rangle = \infty$ here means that the transport plan contains paths that are not consistent with the network structure (i.e., for some $t \in \{1, \dots, \mathcal{T}-1\}$ and some $(i,j) \notin \mathcal{E}$, $(P_{t,t+1}(\mathbf{M}))_{ij} > 0$).

Let μ_1 and $\mu_{\mathcal{T}}$ be the supply and demand distributions, respectively. That is, $(\mu_1)_i = r_i^+$ for $i \in \mathcal{S}^+$ and zero otherwise and $(\mu_{\mathcal{T}})_i = r_i^-$ for $i \in \mathcal{S}^-$ and zero otherwise. Moreover, let $d \in \mathbb{R}_+^n$ encode the capacity constraints of the network; that is, d_i is the flow capacity on edge $i \in \mathcal{E}$. These supply, demand, and capacity constraints can be encoded as equality and inequality constraints on the flow distributions over the edges $P_t(\mathbf{M})$. Based on this, we formulate the linear program

$$\underset{\mathbf{M} \in \mathbb{R}_+^{n^{\mathcal{T}}}}{\text{minimize}} \ \langle \mathbf{C}, \mathbf{M} \rangle \tag{18a}$$

$$\text{subject to} \ P_1(\mathbf{M}) = \mu_1, \tag{18b}$$

$$P_{\mathcal{T}}(\mathbf{M}) = \mu_{\mathcal{T}} \tag{18c}$$

$$P_t(\mathbf{M}) \le d, \qquad \text{for} \qquad t = 2, \dots, \mathcal{T}-1. \tag{18d}$$

The structure of Problem (18) with cost (17) can be illustrated by the path graph in Figure 3. This graph structure will be utilized in Section 4 to develop a numerical method for dynamic minimum-cost network flow problems. Indeed, Problem (18) with cost (17) is equivalent to the dynamic minimum-cost network flow Problem (3) in the sense described in the following theorem.

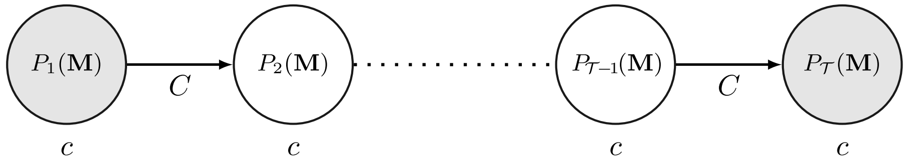
Theorem 1. *The dynamic minimum-cost network flow Problem* (3) *and Problem* (18) *correspond to each other in the following sense.*

1. *Assume that* (18) *has a feasible solution with finite objective value. Then, it has a finite optimal value, and* (3) *has the same optimal value. Moreover, if* $\mathbf{M}$ *is an optimal solution of* (18), *then there is an optimal solution* $\{x_p : p \in \mathcal{P}\}$ *of* (3) *such that*

$$\mathbf{M}_{i_1 \dots i_{\mathcal{T}}} = \begin{cases} x_p & \textit{for } (i_1, \dots, i_{\mathcal{T}}) \in \mathcal{P}, \textit{ where } p = (i_1, \dots, i_{\mathcal{T}}) \\ 0 & \textit{for } (i_1, \dots, i_{\mathcal{T}}) \in \tilde{\mathcal{P}} \setminus \mathcal{P}. \end{cases} \tag{19}$$

2. *Assume that there is a feasible solution to* (3). *Then, it has a finite optimal value, and Problem* (18) *has the same optimal value. Moreover, if* $\{x_p : p \in \mathcal{P}\}$ *is an optimal solution of* (3), *then there is an optimal solution* $\mathbf{M}$ *of* (18) *such that* (19) *holds.*

Figure 3. Illustration of the path graph for the single-commodity network flow problem. Gray and white circles describe equality and inequality constrained marginals, respectively. As described by (16), the costs c are acting on the marginals, and the costs C are acting on the bimarginals.



Proof. First, note that the amount of flow on edge $e \in \mathcal{E}$ between time $t-1$ and t is given in the optimal transport formulation (18) by

$$\sum_{i \in \tilde{\mathcal{P}},\, i_t = e} \mathbf{M}_{i_1 \ldots i_{\mathcal{T}}} = P_t(\mathbf{M})_e. \tag{20}$$

Thus, the flow distribution over $\mathcal{E}_t$ is exactly the projection $P_t(\mathbf{M})$ as defined in (6). Then, with

$$(\mu_1)_i = \begin{cases} r_i^+, & i \in \mathcal{S}^+ \\ 0, & \text{otherwise,} \end{cases} \qquad (\mu_{\mathcal{T}})_i = \begin{cases} r_i^-, & i \in \mathcal{S}^- \\ 0, & \text{otherwise,} \end{cases}$$

the sets of Constraints (3b)–(3c) and (18b)–(18c) both restrict the respective problems to paths that satisfy the supply and demand constraints. In the formulation (3), the total flow on edge $e \in \mathcal{E}_t$ is given by

$$\sum_{p \in \mathcal{P}} \delta_{e=p_t} x_p, \tag{21}$$

and thus, the inequality Constraints (3d) and (18d) restrict the flows in the respective problems to the same capacity constraints. Moreover, note that $(P_{t,t+1}(\mathbf{M}))_{ij}$ describes the amount of flow moving from edge $i \in \mathcal{E}_t$ to edge $j \in \mathcal{E}_{t+1}$. Therefore, the objective (18a) is finite if and only if $\mathbf{M}_{i_1 \ldots i_{\mathcal{T}}} = 0$ for all $(i_1, \ldots, i_{\mathcal{T}}) \in \tilde{\mathcal{P}} \setminus \mathcal{P}$. Now, by associating the amount of flow on edge

$i \in \mathcal{E}_t$

with (20) and (21), respectively, the cost of a feasible flow plan (i.e., a plan that satisfies $\mathbf{M}_{i_1 \ldots i_{\mathcal{T}}} = 0$ for all $(i_1, \ldots, i_{\mathcal{T}}) \in \tilde{\mathcal{P}} \setminus \mathcal{P}$) can be written in the two formulations as

$$\sum_{p \in \mathcal{P}} c_p x_p = \sum_{p \in \mathcal{P}} \left(\sum_{t=1}^{\mathcal{T}} \sum_{e \in \mathcal{E}} \delta_{e=p_t} c_e \right) x_p = \sum_{e \in \mathcal{E}} \sum_{t=1}^{\mathcal{T}} \left(\sum_{p \in \mathcal{P}} \delta_{e=p_t} x_p \right) c_e = \sum_{t=1}^{\mathcal{T}} \sum_{e \in \mathcal{E}} P_t(\mathbf{M})_e c_e = \sum_{t=1}^{\mathcal{T}} c^T P_t(\mathbf{M}).$$

This completes the proof. □

Comparing Problem (18) with Problem (3), we have expanded the set of optimization variables by adding a large number of infeasible paths. However, the novel formulation (18) is structured as a multimarginal optimal transport problem as in (7), which opens up for efficiently computing an approximate solution. In particular, the structure of Problem (18) can be described by the path graph in Figure 3. Although Problem (18) lifts the set of optimization variables in (3) from the set of feasible paths to the set of all combinatorially possible paths in the network, the infinite values in the tensor (17) restrict the problem to the set of feasible paths as in (3).

Remark 1. The second term in (16) is needed only to restrict the solution of Problem (18) to the set of feasible paths $\mathcal{P}$. Naturally, this could instead be imposed as a set of hard constraints $P_{t,t+1}(\mathbf{M}) \leq E$, for $t = 1, \ldots, \mathcal{T}-1$, where $E_{ij} = \infty$ if edge i leads to edge j and $E_{ij} = 0$ otherwise. Instead, we choose to use the penalty terms in (16) for computational reasons. In Section 4, we develop a scheme, which is based on the methods introduced in Section 2.2 (i.e., solving the dual of a regularization of Problem (18)). Note that adding more hard constraints to (18) leads to a larger number of dual variables, which makes it more expensive to solve the regularized dual problem. We thus impose the network structure through the penalty terms in (16), which yields a dual problem with considerably fewer variables. Moreover, infinite values in C induce sparsity to the tensor $\mathbf{K} = \exp(-\mathbf{C}/\epsilon)$ in (10), which can be exploited when computing the projections (6) needed for the Sinkhorn scheme.

3.2. The Dynamic Multicommodity Minimum-Cost Flow Problem

In this section, we extend the optimal transport formulation of the dynamic minimum-cost network flow problem from Section 3.1 to the multicommodity setting.

Assume that there are L different commodities present in the network $\mathcal{N}$, and each of these is assigned an initial distribution μ_1^ℓ and a final distribution $\mu_{\mathcal{T}}^\ell$ for $\ell = 1, \ldots, L$. For each commodity, we define a cost vector $c_\ell \in \mathbb{R}^n$, where $(c_\ell)_i$ denotes the cost of a unit flow of commodity ℓ on edge $i \in \mathcal{E}$. As in the single-commodity case in Section 3.1, the network structure is imposed by a matrix $C \in \overline{\mathbb{R}}_+^{n \times n}$, and the total flow capacity is bounded on all edges and described by a vector $d \in \mathbb{R}_+^n$. One way to formulate an optimal transport problem for the multicommodity flow is to describe each commodity flow by a mass transport tensor $\mathbf{M}^\ell$, for $\ell = 1, \ldots, L$. Then, each of these transport tensors has to satisfy the respective supply and demand Constraints (18b)–(18c), and its cost is given by $\langle \mathbf{C}^\ell, \mathbf{M}^\ell \rangle$ as defined in (16). The capacity constraints in the network need to hold for the sum of all commodity flows (i.e., the sum of the projections $P_t(\mathbf{M}^\ell)$ over all commodities $\ell = 1, \ldots, L$). The dynamic multicommodity minimum-cost flow Problem (4) can, therefore, be written as

$$\underset{\mathbf{M}^1,\ldots,\mathbf{M}^L\in\mathbb{R}_+^{n^{\mathcal{T}}}}{\text{minimize}} \quad \sum_{\ell=1}^{L}\langle \mathbf{C}^\ell, \mathbf{M}^\ell\rangle \tag{22a}$$

$$\text{subject to} \quad P_1(\mathbf{M}^\ell) = \mu_1^\ell, \quad \text{for } \ell = 1,\ldots,L, \tag{22b}$$

$$P_{\mathcal{T}}(\mathbf{M}^\ell) = \mu_{\mathcal{T}}^\ell, \quad \text{for } \ell = 1,\ldots,L, \tag{22c}$$

$$\sum_{\ell=1}^{L} P_t(\mathbf{M}^\ell) \le d, \quad \text{for } t = 2,\ldots,\mathcal{T}-1. \tag{22d}$$

Note here that the L optimal transport problems are each of the form in (18) and are coupled only through the capacity Constraint (22d).

We will now bring Problem (22) on a form similar to a multimarginal optimal transport Problem (7) (i.e., a formulation containing only one mass transport tensor). This is done by combining all information from the $\mathcal{T}$-mode transport plans $\mathbf{M}^\ell \in \mathbb{R}_+^{n\times\cdots\times n}$, for $\ell = 1,\ldots,L$, to a new mass transport tensor $\mathbf{M}\in\mathbb{R}_+^{L\times n\times\cdots\times n}$ with $\mathcal{T}+1$ modes. That is, we let its element $\mathbf{M}_{\ell,i_1\ldots i_{\mathcal{T}}}$ describe the amount of flow of commodity ℓ over the path $i_1,\ldots,i_{\mathcal{T}}$. Accordingly, for the added mode in the tensor, we introduce a marginal $\mu_0\in\mathbb{R}_+^L$, where $(\mu_0)_\ell = \mathbf{1}^T\mu_1^\ell = \mathbf{1}^T\mu_{\mathcal{T}}^\ell$ denotes the total supply and demand of commodity $\ell\in L$. The initial and final distributions for the commodities can then be summarized in two matrices $R^{(0,1)}, R^{(0,\mathcal{T})}\in\mathbb{R}_+^{L\times n}$, defined as $R^{(0,1)} = (\mu_1^1,\mu_1^2,\ldots,\mu_1^L)^T$ and $R^{(0,\mathcal{T})} = (\mu_{\mathcal{T}}^1,\mu_{\mathcal{T}}^2,\ldots,\mu_{\mathcal{T}}^L)^T$. In particular, with this construction, it holds that $R^{(0,1)}\mathbf{1} = R^{(0,\mathcal{T})}\mathbf{1} = \mu_0$. Moreover, define a matrix $C_L\in\mathbb{R}^{L\times n}$ as $C_L = (c_1,c_2,\ldots,c_L)^T$; that is, $(C_L)_{\ell,i}$ denotes the cost for commodity $\ell\in L$ to be on edge $i\in\mathcal{E}$.

Note that the objective function (22a) can be written as

$$\sum_{t=2}^{\mathcal{T}-1}\mathrm{trace}(C_L{}^T P_{0,t}(\mathbf{M})) + \sum_{t=1}^{\mathcal{T}-1}\mathrm{trace}(C^T P_{t,t+1}(\mathbf{M})) = \langle \mathbf{C},\mathbf{M}\rangle,$$

where the cost tensor $\mathbf{C}\in\overline{\mathbb{R}}_+^{L\times n^{\mathcal{T}}}$ is given by

$$\mathbf{C}_{i_0\ldots i_{\mathcal{T}}} = \sum_{t=2}^{\mathcal{T}-1}(C_L)_{i_0 i_t} + \sum_{t=1}^{\mathcal{T}-1} C_{i_t i_{t+1}}. \tag{23}$$

Thus, the dynamic multicommodity minimum-cost network flow Problem (22) can be expressed as

$$\begin{aligned}
&\underset{\mathbf{M}\in\mathbb{R}_+^{L\times n^{\mathcal{T}}}}{\text{minimize}} && \langle \mathbf{C},\mathbf{M}\rangle \\
&\text{subject to} && P_{0,1}(\mathbf{M}) = R^{(0,1)}, \\
& && P_{0,\mathcal{T}}(\mathbf{M}) = R^{(0,\mathcal{T})}, \\
& && P_t(\mathbf{M}) \le d_t, \qquad \text{for} \qquad t = 2,\ldots,\mathcal{T}-1
\end{aligned} \tag{24}$$

Utilizing the result in Theorem 1, we have now proved that the solutions to (24) and the dynamic multicommodity minimum-cost network flow Problem (4) are equivalent, as summarized in the following theorem.

Theorem 2. *The dynamic minimum-cost network flow Problem* (4) *and Problem* (24) *correspond to each other in the following sense.*

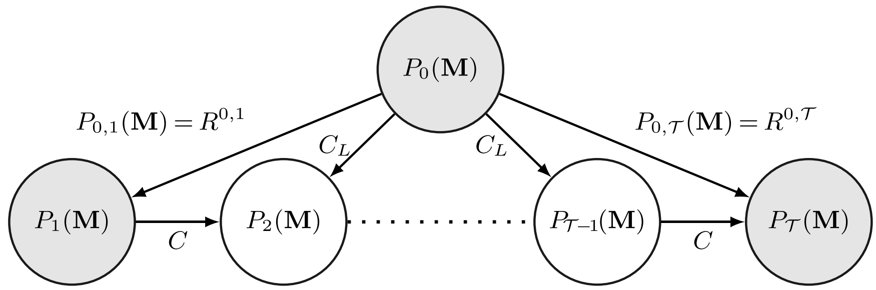
1. *Assume that* (24) *has a feasible solution with finite objective value. Then,* (24) *has a finite optimal value, and* (4) *has the same optimal value. Moreover, if* $\mathbf{M}$ *is an optimal solution of* (24), *then there is an optimal solution* $\{x_p^\ell : p\in\mathcal{P}^\ell, \ell = 1,\ldots,L\}$ *of* (4) *such that*

$$\mathbf{M}_{\ell,i_1\ldots i_{\mathcal{T}}} = \begin{cases} x_p^\ell & \text{for } (i_1,\ldots,i_{\mathcal{T}})\in\mathcal{P}^\ell, \text{ where } p = (i_1,\ldots,i_{\mathcal{T}}) \\ 0 & \text{for } (i_1,\ldots,i_{\mathcal{T}})\in\tilde{\mathcal{P}}\setminus\mathcal{P}^\ell. \end{cases} \tag{25}$$

2. *Assume that there is a finite feasible solution to* (4). *Then, it has a finite optimal value, and Problem* (24) *has the same optimal value. Moreover, if* $\{x_p^\ell : p\in\mathcal{P}^\ell, \ell = 1,\ldots,L\}$ *is an optimal solution of* (4), *then there is an optimal solution* $\mathbf{M}$ *of* (24) *such that* (25) *holds.*

The structure of Problem (24) with cost (23) can also be illustrated as a graph, as seen in Figure 4. The two graphs in Figures 3 and 4 motivate our interest in a general framework for graph-structured optimal transport problems, which will be developed in Section 4.

Figure 4. Illustration of the dynamic multicommodity minimum-cost flow Problem (24). Gray and white circles describe equality and inequality constrained marginals, respectively.



3.3. Generalizations

In this section, we have introduced novel formulations for dynamic minimum-cost network flow problems based on the optimal transport framework. We will now discuss a few modifications and generalizations of the proposed Problems (18) and (24) and show that the proposed formulations in fact provide a highly flexible framework for dynamic network flow problems.

An advantage of our framework is that, because of the fact that the network structure is imposed by the cost matrix C, a time-varying network can be modeled in a straightforward way. Namely, the matrix C can simply be replaced by a set of time-dependent matrices C_t, for $t = 1, \ldots, \mathcal{T} - 1$, where C_t encodes the network topology in the interval $(t, t+1)$. Moreover, based on the formulation (22), where each commodity is described by a separate transport tensor, one can extend the problem to the setting, where different commodities enter and leave the network at different times. In fact, the computational methods derived in this work can easily be modified to this setting, as we will argue in Remark 5.

In some applications, for instance in traffic flow problems, where edges and nodes describe streets and junctions, respectively, it is natural to allow for intermediate storage on the edges. This can be easily incorporated in our framework by letting C_{ii} denote the cost for staying on edge $i \in \mathcal{E}$. It should be noted that in this case, the cost c_i^ℓ denotes the cost for commodity ℓ to use edge $i \in \mathcal{E}$ and not the cost for traveling between the two vertices. That is, the cost accumulates if flow remains on an edge for several time intervals, which is useful (e.g., in traffic routing problems, where the cost model should take the travel time of agents into account). However, we can achieve a cost that does not accumulate in the case where all commodities are described by the same cost $c_i = c_i^\ell$ for all $\ell = 1, \ldots, L$ by defining a negative cost $C_{ii} = -c_i$ for staying on the edge $i \in \mathcal{E}$.

A more classical setting in network flow problems is to allow for storage in the vertices. One way to include this in the presented framework is to augment the support of the modes of the mass transport tensor by the set of vertices (i.e., by letting $n = |\mathcal{E}| + |\mathcal{V}|$). In particular, in the multicommodity Problem (24), the mass transport tensor is then of the size $\mathbf{M} \in \mathbb{R}_+^{L\times(|\mathcal{E}|+|\mathcal{V}|)^{\mathcal{T}}}$, and the distributions are of the size $\mu_t \in \mathbb{R}_+^{|\mathcal{E}|+|\mathcal{V}|}$, for $t = 1, \ldots, \mathcal{T}$. Analogously to before, the network structure is imposed by the cost matrices $C \in \mathbb{R}_+^{(|\mathcal{E}|+|\mathcal{V}|)\times(|\mathcal{E}|+|\mathcal{V}|)}$ (i.e., we define $C_{ij} = 0$ if $i \in \{\mathcal{E} \cup \mathcal{V}\}$ is adjacent[5] to $j \in \{\mathcal{E} \cup \mathcal{V}\}$ and $C_{ij} = \infty$ otherwise). Similarly, the definition of the cost C_L and the capacities d can be extended to the vertices. It is worth noting that this extension of the state space also allows for defining the set of sinks and sources on the vertices instead of the edges.

Another extension of the formulation, of particular interest for traffic routing problems, is the setting where the sinks and sources are defined on nodes but where intermediate storage is only allowed in the sinks and sources and agents are not permitted to enter sources or leave sinks. In this case, we let $n = |\mathcal{E}| + |\mathcal{S}^+| + |\mathcal{S}^-|$ and define the network structure through the cost matrix as follows:

$$C_{ij} = \begin{cases} 0, & \text{if } i \in \{\mathcal{E} \cup \mathcal{S}^+\} \text{ is adjacent to } j \in \{\mathcal{E} \cup \mathcal{S}^+ \cup \mathcal{S}^-\} \\ 0, & \text{if } i \in \{\mathcal{E} \cup \mathcal{S}^+ \cup \mathcal{S}^-\} \text{ is adjacent to } j \in \{\mathcal{E} \cup \mathcal{S}^-\} \\ \infty, & \text{otherwise.} \end{cases} \tag{26}$$

A final extension worth mentioning is the possibility of introducing commodity-dependent capacity constraints (Gendron et al. [29], Kennington [41]). This may be done by introducing the set of constraints $P_{0,t}(\mathbf{M}) \leq D^{(0,t)}$ for $t = 2, \ldots, \mathcal{T} - 1$, with capacity matrices $D^{(0,t)} \in \overline{\mathbb{R}}_+^{L\times n}$, where $D_{\ell i}^{(0,t)}$ denotes the capacity of commodity ℓ on edge $i \in \mathcal{E}$.

4. The Graph-Structured Multimarginal Optimal Transport Problem

Motivated by the two graph-structured optimal transport Problems (17)–(18) and (23)–(24), we will in this section define the general graph-structured optimal transport problem and develop methods to solve the corresponding entropy-regularized problem. This generalizes some of the results in Altschuler and Boix-Adsera [1], Benamou et al. [8], Haasler et al. [33], and Haasler et al. [34]. We will also consider the dynamic flow problems in Section 3 in detail and show how to exploit the graph structures in order to derive efficient methods.

We have noted that the network flow Problems (18) and (24) can be seen as multimarginal optimal transport problems with the underlying graph structures in Figures 3 and 4. In particular, we let each mode of the transport tensor $\mathbf{M}$ be associated with a vertex and let interaction terms be described by edges. This defines a graph $G = (V, E)$ with vertices V and edges E. The interaction terms defining the edges are given by bimarginal constraints, as in (24), or by bimarginal cost terms in the cost tensor: that is, $\mathbf{C} \in \mathbb{R}^{n_1 \times \cdots \times n_{\mathcal{T}}}$ with

$$\mathbf{C}_{i_1 \ldots i_{\mathcal{T}}} = \sum_{(t_1, t_2) \in E} C^{(t_1, t_2)}_{i_{t_1} i_{t_2}}.$$

We denote the set of marginals that are constrained by equality and inequality constraints by $\tilde{V}_= \subset V$ and $\tilde{V}_\leq \subset V$, respectively. Moreover, the set of tuples that are associated with a bimarginal constraint is denoted by $\tilde{E}$. Thus, the dynamic network flow Problems (18) and (24) are special cases of the graph-structured optimal transport problem

$$\begin{aligned} \underset{\mathbf{M} \in \mathbb{R}_+^{n_1 \times \cdots \times n_{\mathcal{T}}}}{\text{minimize}} \quad & \langle \mathbf{C}, \mathbf{M} \rangle \\ \text{subject to} \quad & P_t(\mathbf{M}) = \mu_t, && \text{for } t \in \tilde{V}_= \\ & P_t(\mathbf{M}) \leq d_t, && \text{for } t \in \tilde{V}_\leq \\ & P_{t_1, t_2}(\mathbf{M}) = R^{(t_1, t_2)}, && \text{for } (t_1, t_2) \in \tilde{E}, \end{aligned} \tag{27}$$

where $\mu_t, d_t \in \mathbb{R}_+^{n_t}$, and $R^{(t_1, t_2)} \in \mathbb{R}_+^{n_{t_1} \times n_{t_2}}$. Following the approach presented in Section 2.2.1, we develop a scheme for approximately solving optimal transport problems of this form. It is worth noting that the results in Theorems 3 and 4 and Proposition 1 are only based on the structure of the constraints in (27), and thus, they hold for arbitrary cost tensors $\mathbf{C}$. However, to derive the efficient schemes presented in Section 4.2, the graph structures in the objective function have to be exploited.

4.1. Sinkhorn's Method

In order to apply the approach in Section 2.2.1, we regularize (27) with an entropy term (8), which yields the regularized problem

$$\begin{aligned} \underset{\mathbf{M} \in \mathbb{R}_+^{n_1 \times \cdots \times n_{\mathcal{T}}}}{\text{minimize}} \quad & \langle \mathbf{C}, \mathbf{M} \rangle + \epsilon D(\mathbf{M}) \\ \text{subject to} \quad & P_t(\mathbf{M}) = \mu_t, && \text{for } t \in \tilde{V}_= \\ & P_t(\mathbf{M}) \leq d_t, && \text{for } t \in \tilde{V}_\leq \\ & P_{t_1, t_2}(\mathbf{M}) = R^{(t_1, t_2)}, && \text{for } (t_1, t_2) \in \tilde{E}. \end{aligned} \tag{28}$$

Similarly to the standard multimarginal optimal transport problem, the solution to (28) can be expressed in terms of its optimal dual variables, as the following theorem describes.

Theorem 3. *Assume* $\mathbf{C}$ *is finite and the prescribed marginals* μ_t *for* $t \in \tilde{V}_=$, d_t *for* $t \in \tilde{V}_\leq$, *and* $R^{(t_1, t_2)}$ *for* $(t_1, t_2) \in \tilde{E}$ *are strictly positive. Moreover, assume that* (28) *has a feasible solution. Let* $\tilde{V} = \tilde{V}_= \cup \tilde{V}_\leq$. *Then, the optimal solution to* (28) *has the structure* $\mathbf{M} = \mathbf{K} \odot \mathbf{U}$, *where* $\mathbf{K} = \exp(-\mathbf{C}/\epsilon)$ *and*

$$\mathbf{U}_{i_1 \ldots i_{\mathcal{T}}} = \left(\prod_{t \in \tilde{V}} (u_t)_{i_t} \right) \left(\prod_{(t_1, t_2) \in \tilde{E}} U^{(t_1, t_2)}_{i_{t_1} i_{t_2}} \right), \tag{29}$$

where $u_t \in \mathbb{R}_+^{n_t}$, *for* $t \in \tilde{V}$, *and* $U^{(t_1, t_2)} \in \mathbb{R}_+^{n_{t_1} \times n_{t_2}}$, *for* $(t_1, t_2) \in \tilde{E}$.

In particular, $u_t = \exp(-\lambda_t/\epsilon)$ *and* $U^{(t_1,t_2)} = \exp(-\Lambda^{(t_1,t_2)}/\epsilon)$, *where* $\lambda_t \in \mathbb{R}^{n_t}$ *and* $\Lambda^{(t_1,t_2)} \in \mathbb{R}^{n_{t_1}\times n_{t_2}}$, *for* $t \in \tilde{V}$ *and* $(t_1,t_2) \in \tilde{E}$, *respectively, are optimal variables for the dual problem of* (28), *which is given by*

$$\underset{\substack{\Lambda^{(t_1,t_2)} \in \mathbb{R}^{n_{t_1}\times n_{t_2}},\, (t_1,t_2)\in\tilde{E},\\ \lambda_t\in\mathbb{R}^{n_t},\, t\in\tilde{V}_=\\ \lambda_t\in\mathbb{R}^{n_t}_+,\, t\in\tilde{V}_\le}}{\text{maximize}} \quad -\epsilon\langle \mathbf{K},\mathbf{U}\rangle - \sum_{(t_1,t_2)\in\tilde{E}} \langle \Lambda^{(t_1,t_2)}, R^{(t_1,t_2)}\rangle - \sum_{t\in\tilde{V}}\langle \lambda_t, \mu_t\rangle. \tag{30}$$

Proof. Define Lagrange multipliers $\Lambda^{(t_1,t_2)} \in \mathbb{R}^{n_{t_1}\times n_{t_2}}$, for $(t_1,t_2)\in\tilde{E}$, and $\lambda_t \in \mathbb{R}^{n_t}$, for $t\in\tilde{V}$. Moreover, let $\lambda := (\lambda_t)_{t\in\tilde{V}}$ and $\Lambda := (\Lambda^{(t_1,t_2)})_{(t_1,t_2)\in\tilde{E}}$. With these, a Lagrangian of (28) is

$$\mathcal{L}(\mathbf{M},\lambda,\Lambda) := \langle \mathbf{C},\mathbf{M}\rangle + \epsilon D(\mathbf{M}) + \sum_{(t_1,t_2)\in\tilde{E}} \langle \Lambda^{(t_1,t_2)}, P_{t_1,t_2}(\mathbf{M}) - R^{(t_1,t_2)}\rangle + \sum_{t\in\tilde{V}}\langle \lambda_t, P_t(\mathbf{M}) - \mu_t\rangle. \tag{31}$$

The minimum of (31) with respect to $\mathbf{M}_{i_1\dots i_\mathcal{T}}$ is achieved when its derivative vanishes: that is, when

$$\mathbf{C}_{i_1\dots i_\mathcal{T}} + \epsilon\log(\mathbf{M}_{i_1\dots i_\mathcal{T}}) + \sum_{(t_1,t_2)\in\tilde{E}} \Lambda^{(t_1,t_2)}_{i_{t_1} i_{t_2}} + \sum_{t\in\tilde{V}} (\lambda_t)_{i_t} = 0.$$

Thus, the optimal transport tensor is of the form $\mathbf{M} = \mathbf{K}\odot\mathbf{U}$ with $\mathbf{K}$ and $\mathbf{U}$ as defined in the theorem. Note that the entropy term $D(\mathbf{K}\odot\mathbf{U})$ reads

$$\begin{aligned}&\sum_{i_1,\dots,i_\mathcal{T}} \left(\mathbf{K}_{i_1\dots i_\mathcal{T}}\mathbf{U}_{i_1\dots i_\mathcal{T}} \frac{1}{\epsilon}\left(-\mathbf{C}_{i_1\dots i_\mathcal{T}} - \sum_{(t_1,t_2)\in\tilde{E}} \Lambda^{(t_1,t_2)}_{i_{t_1} i_{t_2}} - \sum_{t\in\tilde{V}} (\lambda_t)_{i_t}\right) - \mathbf{K}_{i_1\dots i_\mathcal{T}}\mathbf{U}_{i_1\dots i_\mathcal{T}} + 1\right)\\ &= -\frac{1}{\epsilon}\langle \mathbf{K}\odot\mathbf{U},\mathbf{C}\rangle - \frac{1}{\epsilon}\sum_{(t_1,t_2)\in\tilde{E}} \langle \Lambda^{(t_1,t_2)}, P_{t_1,t_2}(\mathbf{K}\odot\mathbf{U})\rangle - \frac{1}{\epsilon}\sum_{t\in\tilde{V}}\langle \lambda_t, P_t(\mathbf{K}\odot\mathbf{U})\rangle - \langle \mathbf{K},\mathbf{U}\rangle + \prod_{t=1}^{\mathcal{T}} n_t.\end{aligned}$$

Thus, plugging $\mathbf{M} = \mathbf{K}\odot\mathbf{U}$ into $\mathcal{L}(\mathbf{M},\lambda,\Lambda)$ in (31) and removing constants yield

$$-\epsilon\langle \mathbf{K},\mathbf{U}\rangle - \sum_{(t_1,t_2)\in\tilde{E}} \langle \Lambda^{(t_1,t_2)}, R^{(t_1,t_2)}\rangle - \sum_{t\in\tilde{V}}\langle \lambda_t, \mu_t\rangle. \tag{32}$$

The dual to (28) is to maximize (32) with respect to $\Lambda^{(t_1,t_2)}$ for $(t_1,t_2)\in\tilde{E}$ and λ_t for $t\in\tilde{V}$. Finally, given the assumptions, strong duality holds between the primal and the dual problem (see, e.g., Boyd and Vandenberghe [11, p. 226]). □

The assumptions in Theorem 3 are typically not satisfied for the network flow Problems (18) and (24). If the underlying network is not a complete graph, the cost tensor has infinite entries. Moreover, in most flow problems, the sources and sinks are a strict subset of the set of edges, which is modeled by zero entries in the prescribed marginals μ_t, or $R^{(t_1,t_2)}$. The following theorem extends Theorem 3 to these cases.

Theorem 4. *Let* $\mathbf{C} \in \overline{\mathbb{R}}_+^{n_1\times\dots\times n_\mathcal{T}}$, *and assume that there is a feasible solution* $\mathbf{M}$ *of* (28) *for which* $\mathbf{M}_{i_1\dots i_\mathcal{T}} > 0$ *if and only if* $\mathbf{C}_{i_1\dots i_\mathcal{T}} < \infty, (\mu_t)_{i_t} > 0, (d_t)_{i_t} > 0$, *and* $R^{(t_1,t_2)}_{i_{t_1} i_{t_2}} > 0$. *Then, the optimal solution to* (28) *has the structure* $\mathbf{M} = \mathbf{K}\odot\mathbf{U}$, *where* $\mathbf{K} = \exp(-\mathbf{C}/\epsilon)$ *and* $\mathbf{U}$ *factorizes as in* (29).

Proof. Define the set of tuples

$$I = \left\{ (i_1,\dots,i_\mathcal{T}) | i_t \in \{1,\dots,n\}, \mathbf{C}_{i_1\dots i_\mathcal{T}} < \infty, (\mu_t)_{i_t} > 0, (d_t)_{i_t} > 0, R^{(t_1,t_2)}_{i_{t_1} i_{t_2}} > 0 \right\}.$$

For $(i_1,\dots,i_\mathcal{T})\in I$, we define $\hat{C}_{i_1\dots i_\mathcal{T}} = C_{i_1\dots i_\mathcal{T}}$, $(\hat{\mu}_t)_{i_t} = (\mu_t)_{i_t}$, $(\hat{d}_t)_{i_t} = (d_t)_{i_t}$, and $\hat{R}^{(t_1,t_2)}_{i_{t_1} i_{t_2}} = R^{(t_1,t_2)}_{i_{t_1} i_{t_2}}$. Consider the problem

$$\begin{aligned} \underset{\hat{\mathbf{M}}_{i_1\dots i_\mathcal{T}},(i_1,\dots,i_\mathcal{T})\in I}{\text{minimize}} \quad & \sum_{(i_1,\dots,i_\mathcal{T})\in I} \tilde{\mathbf{C}}_{i_1\dots i_\mathcal{T}} \hat{\mathbf{M}}_{i_1\dots i_\mathcal{T}} + \epsilon D(\hat{\mathbf{M}}) \\ \text{subject to} \quad & P_t(\hat{\mathbf{M}}) = \tilde{\mu}_t, \qquad \text{for } t\in\tilde{V}_=, \\ & P_t(\hat{\mathbf{M}}) \le \hat{d}_t, \qquad \text{for } t\in\tilde{V}_\le, \\ & P_{t_1,t_2}(\hat{\mathbf{M}}) = \hat{R}^{(t_1,t_2)}, \quad \text{for } (t_1,t_2)\in\tilde{E}, \end{aligned} \tag{33}$$

where the definition of $D(\mathbf{M}), P_t(\mathbf{M})$ and $P_{t_1,t_2}(\mathbf{M})$ is relaxed to the case where the argument is not a tensor. The proof of Theorem 3 can be mirrored for the case where the variable is not a tensor. Thus, the optimal solution to

(33) can be written as $\hat{\mathbf{M}}_{i_1 \ldots i_\mathcal{T}} = \hat{\mathbf{K}}_{i_1 \ldots i_\mathcal{T}} \hat{\mathbf{U}}_{i_1 \ldots i_\mathcal{T}}$, where $\hat{\mathbf{K}}_{i_1 \ldots i_\mathcal{T}} = \exp(-\hat{\mathbf{C}}_{i_1 \ldots i_\mathcal{T}}/\epsilon)$, and

$$\hat{\mathbf{U}}_{i_1 \ldots i_\mathcal{T}} = \left(\prod_{t \in \tilde{V}} (\hat{u}_t)_{i_t}\right)\left(\prod_{(t_1,t_2) \in \tilde{E}} \hat{U}^{(t_1,t_2)}_{i_{t_1} i_{t_2}}\right),$$

where $(i_1, \ldots, i_\mathcal{T}) \in I$. Now, define the tensors $\mathbf{K} = \exp(-\mathbf{C}/\epsilon)$ and $\mathbf{U} \in \mathbb{R}_+^{n_1 \times \cdots \times n_\mathcal{T}}$, which are constructed as in (29), where

$$(u_t)_{i_t} = \begin{cases} (\hat{u}_t)_{i_t}, & \text{if it is defined,} \\ 0, & \text{otherwise,} \end{cases} \qquad U^{(t_1,t_2)}_{i_{t_1} i_{t_2}} = \begin{cases} \hat{U}^{(t_1,t_2)}_{i_{t_1} i_{t_2}}, & \text{if it is defined,} \\ 0, & \text{otherwise.} \end{cases} \tag{34}$$

Then, by construction, $\mathbf{M} = \mathbf{K} \odot \mathbf{U}$ is an optimal solution to (28). □

The Sinkhorn iterations for Problem (28) can be derived as a block-coordinate ascend method in the dual Problem (30), as summarized in the following proposition.

Proposition 1. *Assume* (28) *has a feasible solution as in the assumptions of Theorem* 4. *Let* $\mathbf{K} = \exp(-\mathbf{C}/\epsilon)$ *and* $\mathbf{U}$ *as defined in* (29). *Then, the iterative scheme*

$$U^{(t_1,t_2)} \leftarrow U^{(t_1,t_2)} \odot R^{(t_1,t_2)} \oslash P_{t_1,t_2}(\mathbf{K} \odot \mathbf{U}), \qquad \text{for } (t_1,t_2) \in \tilde{E} \tag{35a}$$

$$u_t \leftarrow u_t \odot \mu_t \oslash P_t(\mathbf{K} \odot \mathbf{U}), \qquad \text{for } t \in \tilde{V}_= \tag{35b}$$

$$u_t \leftarrow \min\big(u_t \odot d_t \oslash P_t(\mathbf{K} \odot \mathbf{U}), \mathbf{1}\big), \qquad \text{for } t \in \tilde{V}_\le, \tag{35c}$$

converges linearly, and in the limit point, the optimal solution of (28) *is given by* $\mathbf{M} = \mathbf{K} \odot \mathbf{U}$.

Proof. We first assume that the stronger assumptions from Theorem 3 hold. The scheme is derived as a block-coordinate ascent method in the dual (30). This is to maximize the objective with respect to one set of dual variables while keeping the other dual variables fixed: that is, to perform the updates

$$\Lambda^{(t_1,t_2)} \leftarrow \underset{\Lambda^{(t_1,t_2)} \in \mathbb{R}^{n_{t_1} \times n_{t_2}}}{\arg\max} \; -\epsilon\langle \mathbf{K}, \mathbf{U}\rangle - \langle \Lambda^{(t_1,t_2)}, R^{(t_1,t_2)}\rangle, \qquad \text{for } (t_1,t_2) \in \tilde{E} \tag{36a}$$

$$\lambda_t \leftarrow \underset{\lambda_t \in \mathbb{R}^{n_t}}{\arg\max} \; -\epsilon\langle \mathbf{K}, \mathbf{U}\rangle - \langle \lambda_t, \mu_t\rangle, \qquad \text{for } t \in \tilde{V}_= \tag{36b}$$

$$\lambda_t \leftarrow \underset{\lambda_t \in \mathbb{R}_+^{n_t}}{\arg\max} \; -\epsilon\langle \mathbf{K}, \mathbf{U}\rangle - \langle \lambda_t, d_t\rangle, \qquad \text{for } t \in \tilde{V}_\le. \tag{36c}$$

The objectives of the unconstrained Problems (36a) and (36b) are strictly concave, and thus, a necessary and sufficient condition for optimality is that the respective gradient vanishes. Note that for each $(t_1,t_2) \in E$, the gradient of (36a) with respect to $\Lambda^{(t_1,t_2)}$ is

$$\exp(-\Lambda^{(t_1,t_2)}/\epsilon) \odot \left(\sum_{i_0, \ldots, i_\mathcal{T} \setminus \{i_{t_1}, i_{t_2}\}} \mathbf{K}_{i_0 \ldots i_\mathcal{T}} \left(\prod_{t \in \tilde{V}} (u_t)_{i_t}\right)\left(\prod_{(\tau_1,\tau_2) \in \tilde{E} \setminus (t_1,t_2)} U^{(\tau_1,\tau_2)}_{i_{\tau_1} i_{\tau_2}}\right)\right) - R^{(t_1,t_2)},$$

and setting it to zero gives (35a). Similarly, for $t \in V_=$, the gradient of (36b) with respect to λ_t is

$$\exp(-\lambda_t/\epsilon) \odot \left(\sum_{i_0, \ldots, i_{t-1}, i_{t+1}, \ldots, i_\mathcal{T}} \mathbf{K}_{i_0 \ldots i_\mathcal{T}} \left(\prod_{\tau \in \tilde{V} \setminus \{t\}} (u_\tau)_{i_\tau}\right)\left(\prod_{(t_1,t_2) \in \tilde{E}} U^{(t_1,t_2)}_{i_{t_1} i_{t_2}}\right)\right) - \mu_t,$$

which yields (35b). Finally, note that the objective in (36c) can be written as

$$\sum_{i_t} \left(-\epsilon e^{-(\lambda_t)_{i_t}/\epsilon} \left(\sum_{i_0, \ldots, i_{t-1}, i_{t+1}, \ldots, i_\mathcal{T}} \mathbf{K}_{i_0 \ldots, i_\mathcal{T}} \left(\prod_{\tau \in \tilde{V} \setminus \{t\}} (u_\tau)_{i_\tau}\right)\left(\prod_{(t_1,t_2) \in \tilde{E}} U^{(t_1,t_2)}_{i_{t_1} i_{t_2}}\right)\right) - (\lambda_t)_{i_t} d_{i_t} \right).$$

Thus, the maximization in (36c) can be performed in each element of λ_t individually. If the derivative of the objective in (36c) with respect to $(\lambda_t)_{i_t}$ vanishes for a feasible (i.e., nonnegative) point, then this is the global maximizer. Otherwise, the maximizer is the projection on the feasible set (i.e., $(\lambda_t)_{i_t} = 0$). This yields (35c). The linear convergence of the scheme follows from Luo and Tseng [47] (cf. Haasler et al. [33, theorem 3.5]).

In case only the assumptions in Theorem 4 are satisfied, we perform a block-coordinate ascent in the dual of (33). The dual variables can then be augmented by zero entries as in (34) to arrive at the scheme in the proposition. □

Computing the projections of $\mathbf{K} \odot \mathbf{U}$ in Proposition 1 is in general still expensive because computing the sums in (6) and (15) requires $\mathcal{O}(n^{\mathcal{T}})$ operations. However, in the dynamic minimum-cost flow problems, there are additional structures in the cost tensor $\mathbf{C}$ and thus, in the tensor $\mathbf{K}$. Namely, these tensors decouple according to the graphs in Figures 3 and 4. The next subsections describe how these structures can be utilized in order to efficiently compute the projections needed to apply the scheme in Proposition 1.

4.2. Sinkhorn's Method for the Dynamic Minimum-Cost Flow Problem

Recall that the dynamic minimum-cost flow Problem (18) is a multimarginal optimal transport problem. In particular, it can be written on the form (27), where $\tilde{V}_= = \{1, \mathcal{T}\}$, $\tilde{V}_\leq = \{2, \ldots, \mathcal{T}-1\}$, and $E = \emptyset$. Adding the entropy term (8) yields then an entropy-regularized Problem (28), which in this case, explicitly reads

$$\begin{aligned} \underset{\mathbf{M} \in \mathbb{R}_+^{n^{\mathcal{T}}}}{\text{minimize}} \quad & \langle \mathbf{C}, \mathbf{M} \rangle + \epsilon D(\mathbf{M}) \\ \text{subject to} \quad & P_t(\mathbf{M}) \leq d_t, \qquad \text{for} \qquad t = 2, \ldots, \mathcal{T}-1 \\ & P_1(\mathbf{M}) = \mu_1, \\ & P_{\mathcal{T}}(\mathbf{M}) = \mu_t, \end{aligned} \tag{37}$$

where $\mathbf{C}$ is defined by

$$\mathbf{C}_{i_1 \ldots i_{\mathcal{T}}} = \sum_{t=1}^{\mathcal{T}} c_{i_t} + \sum_{t=1}^{\mathcal{T}-1} C_{i_t i_{t+1}}.$$

Remark 2. Without the inequality constraints $P_t(\mathbf{M}) \leq d_t$ and with zero cost on the edges, $c = \mathbf{0}$, the entropy-regularized Problem (37) is a discrete Schrödinger bridge problem (Haasler et al. [32], Haasler et al. [33], Pavon and Ticozzi [52]). The Schrödinger bridge problem is tightly connected to optimal transport (Chen et al. [14], Léonard [43]). It is a popular tool in ensemble control applications, as it provides a framework for steering a given distribution (i.e., an ensemble of agents) to a target one (Brockett [12], Chen et al. [15]). In particular, network flow problems of this form have previously been considered in Chen et al. [17], Chen et al. [18], and Chen et al. [19]. This connection to the Schrödinger bridge problem gives another motivation for adding the regularizing entropy term to the objective of (18). Namely, the Schrödinger bridge problem on a network can be interpreted as an ensemble of agents, which are each evolving according to a Markov chain (Haasler et al. [32], Pavon and Ticozzi [52]). The entropy term thus induces a stochastic component to the problem, which yields a more smoothed out solution. Therefore, the solutions to the regularized Problem (37) can be understood as robust transport plans (Chen et al. [17], Chen et al. [18], Chen et al. [19]).

According to Theorem 4, the solution to the regularized Problem (37) is of the form $\mathbf{M} = \mathbf{K} \odot \mathbf{U}$, where

$$\mathbf{K}_{i_1 \ldots i_{\mathcal{T}}} = \left(\prod_{t=1}^{\mathcal{T}} k_{i_t} \right) \left(\prod_{t=1}^{\mathcal{T}-1} K_{i_t i_{t+1}} \right),$$

with $k = \exp(-c/\epsilon)$ and $K = \exp(-C/\epsilon)$, and $\mathbf{U} = u_1 \otimes \cdots \otimes u_{\mathcal{T}}$. The components of the tensor $\mathbf{U}$ can be found utilizing Proposition 1. In particular, the solution is found by iterating

$$\begin{aligned} u_t &\leftarrow u_t \odot \mu_t \oslash P_t(\mathbf{K} \odot \mathbf{U}), && \text{for } t = 1, \mathcal{T}, \\ u_t &\leftarrow \min\big(u_t \odot d \oslash \big(P_t(\mathbf{K} \odot \mathbf{U})\big), \mathbf{1}\big), && \text{for } t = 2, \ldots, \mathcal{T}-1. \end{aligned} \tag{38}$$

In this case, where the cost decouples according to a path graph, the projections can be computed efficiently (Elvander et al. [23, proposition 2]). Namely, the projections for this problem are of the form

$$P_t(\mathbf{K} \odot \mathbf{U}) = u_t \odot k_t \odot \hat{\varphi}_t \odot \varphi_t, \tag{39}$$

for $t = 1, \ldots, \mathcal{T}$, where

$$\hat{\varphi}_t = K^T \mathrm{diag}(u_{t-1} \odot k_{t-1}) K^T \ldots \mathrm{diag}(u_2 \odot k_2) K^T (u_1 \odot k_1), \tag{40a}$$

$$\varphi_t = K \mathrm{diag}(u_{t+1} \odot k_{t+1}) K \ldots \mathrm{diag}(u_{\mathcal{T}-1} \odot k_{\mathcal{T}-1}) K (u_{\mathcal{T}} \odot k_{\mathcal{T}}). \tag{40b}$$

The Sinkhorn algorithm (38) is summarized in Algorithm 1.

Algorithm 1 (Scheme for Solving the Dual of the Regularized Dynamic Flow Problem (37))

Initialize $u_1, \dots, u_{\mathcal{T}}$, $t = 1$, $\hat{\varphi}_1 = \mathbf{1}$, $\varphi_{\mathcal{T}} = \mathbf{1}$
while Not converged **do**
  for $t = \mathcal{T} - 1, \dots, 1$ **do**
    Update $\varphi_t \leftarrow K(u_{t+1} \odot k_{t+1} \odot \varphi_{t+1})$
  end for
  Update $u_1 \leftarrow \mu_1 \oslash \varphi_1$
  for $t = 2, \dots, \mathcal{T} - 1$ **do**
    Update $\hat{\varphi}_t \leftarrow K^T(u_{t-1} \odot k_{t-1} \odot \hat{\varphi}_{t-1})$
    Update $u_t \leftarrow \min(d_t \oslash (\varphi_t \odot \hat{\varphi}_t \odot k_t), \mathbf{1})$
  end for
  Update $\hat{\varphi}_{\mathcal{T}} \leftarrow K^T(u_{\mathcal{T}-1} \odot k_{\mathcal{T}-1} \odot \hat{\varphi}_{\mathcal{T}-1})$
  Update $u_{\mathcal{T}} \leftarrow \mu_{\mathcal{T}} \oslash \hat{\varphi}_{\mathcal{T}}$
end while
return $u_1, \dots, u_{\mathcal{T}}$

Note that intermediate results of (40a) and (40b) are stored and that the updates in (38) are scheduled such that for each update, only one matrix-vector multiplication needs to be performed. Thus, in the case of a dense matrix K, one iteration sweep (i.e., once updating all vectors u_t) for $t = 1, \dots, \mathcal{T}$ is of complexity $\mathcal{O}(\mathcal{T} n^2)$. However, for sparse networks, the matrix K is also sparse, and thus, the matrix multiplications required to compute the projections (39) via (40) become even more efficient, as discussed in the following remark.

Remark 3. Note that $K_{ij} > 0$ if the edges i and j are adjacent and $K_{ij} = 0$ otherwise. Thus, multiplication with a vector $v \in \mathbb{R}^n$ can be performed as

$$(Kv)_i = \sum_{j \in N(i)} K_{ij} v_j.$$

This multiplication is of order $\mathcal{O}(\Delta(\mathcal{N}) \cdot n)$, where $\Delta(\mathcal{N})$ is the maximum degree of $\mathcal{N}$ (i.e., the highest number of neighboring nodes among the nodes $\mathcal{V}$). The complexity of one iteration sweep in Algorithm 1 is thus $\mathcal{O}(\mathcal{T} n \Delta(\mathcal{N}))$. The algorithm converges linearly by Proposition 1 to an optimal set of vectors $u_1, \dots, u_{\mathcal{T}}$, which make up the components in the tensor $\mathbf{U}$ that defines the optimal solution tensor $\mathbf{M} = \mathbf{K} \odot \mathbf{U}$ for (37).

4.3. Sinkhorn's Method for the Dynamic Multicommodity Minimum-Cost Flow Problem

Similarly to the previous section, the multicommodity Problem (24) is also a multimarginal optimal transport problem of the form (27). In particular, here the constraint sets are $\tilde{V}_= = \emptyset$, $\tilde{V}_\leq = \{2, \dots, \mathcal{T} - 1\}$, and $\tilde{E} = \{(0,1), (0, \mathcal{T})\}$. Regularizing the problem with an entropy term, it is of the form (28), which in this case, reads

$$\begin{aligned} \underset{\mathbf{M} \in \mathbb{R}_+^{L \times n^{\mathcal{T}}}}{\text{minimize}} \quad & \langle \mathbf{C}, \mathbf{M} \rangle + \epsilon D(\mathbf{M}) \\ \text{subject to} \quad & P_{0,1}(\mathbf{M}) = R^{(0,1)}, \\ & P_{0,\mathcal{T}}(\mathbf{M}) = R^{(0,\mathcal{T})}, \\ & P_t(\mathbf{M}) \leq d_t, \qquad \text{for } t = 2, \dots, \mathcal{T} - 1, \end{aligned} \tag{41}$$

where $\mathbf{C}$ is defined by

$$\mathbf{C}_{i_0 \dots i_{\mathcal{T}}} = \sum_{t=2}^{\mathcal{T}-1} (C_L)_{i_0 i_t} + \sum_{t=1}^{\mathcal{T}-1} C_{i_t i_{t+1}}.$$

The solution to (41) can again be expressed in terms of its dual variables, as described in Theorem 4. In particular, the optimal mass transport plan is of the form $\mathbf{M} = \mathbf{K} \odot \mathbf{U}$, where $\mathbf{K}$ factorizes as

$$\mathbf{K}_{i_0 \dots i_{\mathcal{T}}} = \left(\prod_{t=2}^{\mathcal{T}-1} (K_L)_{i_0 i_t} \right) \left(\prod_{t=1}^{\mathcal{T}-1} K_{i_t i_{t+1}} \right), \tag{42}$$

where $K_L = \exp(-C_L/\epsilon)$ and $K = \exp(-C/\epsilon)$. Moreover, the tensor $\mathbf{U}$ is of the form

$$\mathbf{U}_{i_0 \dots i_{\mathcal{T}}} = U^{(0,1)}_{i_0 i_1} U^{(0,\mathcal{T})}_{i_0 i_{\mathcal{T}}} \prod_{t=2}^{\mathcal{T}-1} (u_t)_{i_t}, \tag{43}$$

and its components can be found according to Proposition 1 by iteratively updating

$$
\begin{aligned}
U^{(0,t)} &\leftarrow U^{(0,t)} \odot R^{(0,t)} \oslash P_{0,t}(\mathbf{K} \odot \mathbf{U}), && \text{for } t = 1, \mathcal{T}, \\
u_t &\leftarrow \min\big(u_t \odot d \oslash P_t(\mathbf{K} \odot \mathbf{U}), \mathbf{1}\big), && \text{for } t = 2, \ldots, \mathcal{T} - 1.
\end{aligned} \tag{44}
$$

Again, the tensor $\mathbf{K} \odot \mathbf{U}$ has a graph structure, which is illustrated in Figure 4. This graph contains cycles, and thus, the results from Haasler et al. [33] cannot be utilized. Nevertheless, the projections can be computed relatively efficiently, as demonstrated by the next theorem.

Theorem 5. *Consider the tensors $\mathbf{K} = \exp(-\mathbf{C}/\epsilon)$, with $\mathbf{C}$ defined as in (23) and $\epsilon > 0$ and $\mathbf{U}$ in (43). With the matrices $K_L = \exp(-C_L/\epsilon)$ and $K = \exp(-C/\epsilon)$, define*

$$
\hat{\Psi}_t = \begin{cases} U^{(0,1)}K, & t = 2, \\ (\hat{\Psi}_{t-1} \odot K_L)\mathrm{diag}(u_{t-1})K, & t = 3, \ldots, \mathcal{T}, \end{cases} \tag{45}
$$

and

$$
\Psi_t = \begin{cases} U^{0,\mathcal{T}}K^T, & t = \mathcal{T} - 1, \\ (\Psi_{t+1} \odot K_L)\mathrm{diag}(u_{t+1})K^T, & t = 1, \ldots, \mathcal{T} - 2. \end{cases} \tag{46}
$$

Then, the bimarginal projections of the tensor $\mathbf{K} \odot \mathbf{U}$ are

$$
\begin{aligned}
P_{0,1}(\mathbf{K} \odot \mathbf{U}) &= U^{(0,1)} \odot \Psi_1 \\
P_{0,\mathcal{T}}(\mathbf{K} \odot \mathbf{U}) &= U^{(0,\mathcal{T})} \odot \hat{\Psi}_{\mathcal{T}} \\
P_{0,t}(\mathbf{K} \odot \mathbf{U}) &= (\hat{\Psi}_t \odot \Psi_t \odot K_L)\mathrm{diag}(u_t), \quad \textit{for } t = 2, \ldots, \mathcal{T} - 1.
\end{aligned} \tag{47}
$$

Proof. Note that the tensor $\mathbf{K} = \exp(-\mathbf{C}/\epsilon)$ is element-wise defined as in (42); thus, the bimarginal projections of the tensor $\mathbf{K} \odot \mathbf{U}$ on the marginals 0 and t, where $t \in \{2, \ldots, \mathcal{T} - 1\}$, are given by

$$
\begin{aligned}
P_{0,t}(\mathbf{K} \odot \mathbf{U}) &= \sum_{\substack{i_1, \ldots, i_{t-1} \\ i_{t+1}, \ldots, i_{\mathcal{T}}}} \left(\prod_{s=1}^{\mathcal{T}-1} K_{i_s i_{s+1}}\right)\left(\prod_{s=2}^{\mathcal{T}-1} (K_L)_{i_0 i_s}\right) U^{(0,1)}_{i_0 i_1} U^{(0,\mathcal{T})}_{i_0 i_{\mathcal{T}}} \prod_{s=2}^{\mathcal{T}-2} (u_s)_{i_s} \\
&= (u_t)_{i_t} (K_L)_{i_0 i_t} (\hat{\Psi}_t)_{i_0 i_t} (\Psi_t)_{i_0 i_t},
\end{aligned}
$$

where

$$
\hat{\Psi}_t = \sum_{i_1, \ldots, i_{t-1}} U^{(0,1)}_{i_0 i_1} K_{i_1 i_2} \left(\prod_{s=2}^{t-1} (K_L \mathrm{diag}(u_s))_{i_0 i_s} K_{i_s i_{s+1}}\right)
$$

and

$$
\Psi_t = \sum_{i_{t+1}, \ldots, i_{\mathcal{T}}} U^{(0,\mathcal{T})}_{i_0 i_{\mathcal{T}}} K_{i_{\mathcal{T}-1} i_{\mathcal{T}}} \left(\prod_{s=t+1}^{\mathcal{T}-1} (K_L \mathrm{diag}(u_s))_{i_0 i_s} K_{i_{s-1} i_s}\right).
$$

These terms lead to the recursive definitions of $\hat{\Psi}_t$ and Ψ_t in (45) and (46). The projections $P_{0,1}(\mathbf{K} \odot \mathbf{U})$ and $P_{0,\mathcal{T}}(\mathbf{K} \odot \mathbf{U})$ are derived similarly. □

The projections on one marginal can then be found by projecting the bimarginal projections in (47) on one of the marginals, which yields the following.

Corollary 1. *The marginals of the tensor $\mathbf{K} \odot \mathbf{U}$ in Theorem 5 are given by*

$$
\begin{aligned}
P_t(\mathbf{K} \odot \mathbf{U}) &= u_t \odot (\hat{\Psi}_t \odot \Psi_t \odot K_L)^T \mathbf{1}, \quad \textit{for } t = 2, \ldots, \mathcal{T} - 1, \\
P_0(\mathbf{K} \odot \mathbf{U}) &= (\hat{\Psi}_t \odot \Psi_t \odot K_L) u_t.
\end{aligned}
$$

Theorem 5 and Corollary 1 describe an efficient way to compute the projections required for the Sinkhorn scheme (44), and the resulting computational method is summarized in Algorithm 2. Similarly to the algorithm for the single-commodity setting, intermediate results can be stored and utilized.

Algorithm 2 (Scheme for Solving the Dual of the Regularized Dynamic Multicommodity Flow Problem (41))
Initialize $u_2, \ldots, u_{\mathcal{T}-1}, U^{(0,1)}, U^{(0,\mathcal{T})}$
Compute Ψ_t, for $t = 1, \ldots, \mathcal{T}$
while Not converged **do**
  Update $U^{(0,1)} \leftarrow R^{(0,1)} \oslash \Psi_1$
  Update $\hat{\Psi}_2 \leftarrow U^{(0,1)} K$
  for $t = 2, \ldots, \mathcal{T} - 1$ **do**
    Update $u_t \leftarrow \min\big(d \oslash ((\hat{\Psi}_t \odot \Psi_t \odot K)^T \mathbf{1}), \mathbf{1}\big)$
    Update $\hat{\Psi}_{t+1} \leftarrow (\hat{\Psi}_t \odot K_L)\mathrm{diag}(u_t) K$
  end for
  $U^{(0,\mathcal{T})} \leftarrow R^{(0,\mathcal{T})} \oslash \hat{\Psi}_{\mathcal{T}}$
  Update $\Psi_{\mathcal{T}-1} \leftarrow U^{(0,\mathcal{T})} K^T$
  for $t = \mathcal{T} - 1, \ldots, 2$ **do**
    Update $\Psi_{t-1} \leftarrow (\Psi_t \odot K_L)\mathrm{diag}(u_t) K^T$
  end for
end while
return $u_2, \ldots, u_{\mathcal{T}-1}, U^{(0,1)}, U^{(0,\mathcal{T})}$

Remark 4. The computational bottleneck of the Sinkhorn iterations lies in computing the projections. One iteration sweep of the Sinkhorn iterations requires updating each of the matrices in (45) and (46) once. For dense matrices K, each of these updates is of complexity $\mathcal{O}(Ln^2)$, and thus, one full iteration sweep can be done in $\mathcal{O}(\mathcal{T}Ln^2)$. However, as noted in Remark 3, the matrix K inherits the sparsity of the network, and this can be exploited to perform the matrix multiplications in (45) and (46) more efficiently. Thus, the complexity of the matrix-matrix multiplication is decreased to $\mathcal{O}(\Delta(\mathcal{N}) \cdot Ln)$, and one full iteration sweep can be done in $\mathcal{O}(\mathcal{T}\Delta(\mathcal{N})Ln)$. Moreover, by Proposition 1, Algorithm 2 converges linearly to an optimal set of components of the tensor $\mathbf{U}$ in (43), which defines the optimal solution tensor $\mathbf{M} = \mathbf{K} \odot \mathbf{U}$ for (41).

Remark 5. In Section 3.2, we have formulated the multitensor Problem (22) as the one-tensor Problem (24) in order to bring it on the form of a graph-structured optimal transport Problem (27) and then solve it. Alternatively, we could have regularized each of the L optimal transport problems in (22) separately, yielding the regularized problem

$$\begin{aligned} \underset{\mathbf{M}^1, \ldots, \mathbf{M}^L \in \mathbb{R}_+^{n^{\mathcal{T}}}}{\text{minimize}} \quad & \sum_{\ell=1}^{L} \big(\langle \mathbf{C}, \mathbf{M}^\ell \rangle + \epsilon D(\mathbf{M}^\ell)\big) \\ \text{subject to} \quad & P_1(\mathbf{M}^\ell) = \mu_1^\ell, \quad \ell = 1, \ldots, L, \\ & P_{\mathcal{T}}(\mathbf{M}^\ell) = \mu_{\mathcal{T}}^\ell, \quad \ell = 1, \ldots, L, \\ & \sum_{\ell=1}^{L} P_t(\mathbf{M}^\ell) \le d \quad t = 2, \ldots, \mathcal{T} - 1, \end{aligned}$$

where $\mathbf{C}$ is defined as in (16). In fact, this problem is equivalent to the regularized Problem (41). Moreover, in this representation, the Sinkhorn iterations are given by

$$\begin{aligned} u_1^\ell &\leftarrow u_1^\ell \odot \mu_1^\ell \oslash P_1(\mathbf{K}^\ell \odot \mathbf{U}^\ell), && \text{for } \ell = 1, \ldots, L \\ u_t &\leftarrow \min\left(u_t \odot d \oslash \left(\sum_{\ell \in L} P_t(\mathbf{K}^\ell \odot \mathbf{U}^\ell)\right), \mathbf{1}\right), && \text{for } t = 2, \ldots, \mathcal{T} - 1, \\ u_{\mathcal{T}}^\ell &\leftarrow u_{\mathcal{T}}^L \odot \mu_{\mathcal{T}}^\ell \oslash (P_{\mathcal{T}}(\mathbf{K}^\ell \odot \mathbf{U}^\ell)), && \text{for } \ell = 1, \ldots, L \end{aligned}$$

and these are equivalent to the Sinkhorn iterations derived (cf. (44)). Recall from Section 3.3 that one convenient feature of formulation (22) is that it can be easily extended to allow for commodities that enter and leave the network at different times. Therefore, as can be seen here, such problems can also be solved efficiently.

5. Simulations

In this section, we illustrate the computational efficiency of our proposed framework. First, we compare its performance with the state-of-the-art LP solvers CPLEX and Gurobi on two different types of networks.[6] Finally, we illustrate it in a traffic routing problem with a large number of commodities.

5.1. Performance Study on a Sparse Grid Network

We first consider a dynamic multicommodity minimum-cost network flow problem on a sparse network. To this end, let $\mathcal{N}$ be a grid of nodes, let the source $\mathcal{S}^+$ for all commodities be an incoming edge to one corner of the square, and let the sink $\mathcal{S}^-$ be an outgoing edge from the opposite corner. In this setup, the sink and source can be understood as the two corner vertices. We let the total flow of each commodity be one: that is, $\mu_0 = \mathbf{1}$. Moreover, the capacity vector $d \in \mathbb{R}_+^n$ is defined as $d_i = L$ for $i \in \{\mathcal{S}^+ \cup \mathcal{S}^-\}$ and $d_i = 1$ otherwise. The cost for a unit flow of each commodity on each edge is randomly assigned from a uniform distribution on [0,1]; that is, we let $c_e^\ell \sim \mathrm{Unif}([0,1])$, for $\ell = 1, \ldots, L$, and $e \in \mathcal{E}$. Here, we do not allow for intermediate storage on the vertices or the edges, except in the sink and source.

We consider several combinations of the following three parameters: number of edges, number of commodities, and number of time steps. The dynamic multicommodity network flow problem is then solved utilizing Algorithm 2 with different values of the regularization parameter ϵ. The output of the algorithm is used to compute an approximate transport plan as $\mathbf{M} = \mathbf{K} \odot \mathbf{U}$, where the components are given by (42) and (43). To evaluate the solution $\mathbf{M}$, we plug it into the unregularized objective of (24), $\langle \mathbf{C}, \mathbf{M} \rangle$, and compute the constraint violation

$$\mathrm{violation}(\mathbf{M}) = \|P_{0,1}(\mathbf{M}) - R^{(0,1)}\|_1 + \|P_{0,\mathcal{T}}(\mathbf{M}) - R^{(0,\mathcal{T})}\|_1 + \sum_{t=2,\ldots,\mathcal{T}} \|\max(0, P_t(\mathbf{M}) - d_t)\|_1.$$

Because Algorithm 2 is a block-coordinate ascent in the dual of the regularized objective in (41), the algorithm converges to the fixed point of (44) as the constraint violation goes to zero. Therefore, the constraint violation of the iterates is a measure for the distance between the current and the optimal solution. As a baseline, we also solve the problem in node-edge formulation (cf. Ford and Fulkerson [27], Tomlin [57]) in the time-expanded network using the solvers Gurobi Optimization LLC [30] and IBM ILOG CPLEX [37]. The results are presented in Tables 1–3.

One can see that our method achieves close approximations to the optimal solution. The value of $\langle \mathbf{C}, \mathbf{M} \rangle$ converges to a value that is slightly larger than the true optimal value because our method solves the regularized Problem (41). Moreover, we achieve close approximations of the optimal objective value even before the method is converged. The smaller the regularization parameter ϵ, the closer we get to a true optimal solution.[7] In all experiments, our proposed algorithm reaches a close to optimal solution significantly faster than the LP solvers. For the smallest problem in Table 1, it converges in less than 3% of the run time of the fastest LP solver, CPLEX's dual simplex method. For the largest problem in Table 3, we are able to achieve a close to optimal solution in less than 0.4% of the fastest LP solvers run time.

5.2. Performance Study on a Dense Random Network

Next, we study the performance of Algorithm 2 in a less favorable setting.

Here, we consider a dense network with $n = 50$ nodes, where with probability 1/2, a directed edge is created between each (ordered) pair of nodes. The expected value of the number of edges in the network is thus $\binom{50}{2} = 1225$. Moreover, we allow for intermediate storage in the nodes but not in the edges. Therefore, we augment the state space by the set of nodes as described in Section 3.3. We equip each of the L commodities with a random source and sink on the set of nodes. The total flow of each commodity is set to one (i.e., $\mu_0 = \mathbf{1}$), and we consider the dynamic problem with $\mathcal{T}$ time intervals. The capacity vector $d \in \mathbb{R}_+^n$ is defined as $d_i = L/(N\mathcal{T})$, if $i \in \mathcal{E}$, and $d_i = L$, if $i \in \mathcal{V}$. As in the previous example, the cost for each commodity and each edge is assigned from a uniform distribution on [0,1]. Moreover, the cost for intermediate storage on the nodes is zero.

Table 1. Performance of Algorithm 2 on a 5×5 grid (i.e., $n = 84$ edges) for $T = 60$ time steps and $L = 50$ commodities.

	$\epsilon = 0.04$		$\epsilon = 0.02$		$\epsilon = 0.01$	
Run time (seconds)	$\langle \mathbf{C}, \mathbf{M} \rangle$	Violation ($\mathbf{M}$)	$\langle \mathbf{C}, \mathbf{M} \rangle$	Violation ($\mathbf{M}$)	$\langle \mathbf{C}, \mathbf{M} \rangle$	Violation ($\mathbf{M}$)
0.03125	153.56	1.22 e+02	152.41	2.54 e+01	152.03	2.52 e+01
0.0625	153.72	1.75 e+01	152.42	2.85 e+00	152.05	8.68 e−01
0.125	154.13	1.67 e−15	152.37	3.77 e−15	152.05	1.78 e−15

Optimal objective value	Run time (seconds)	Primal simplex	Dual simplex
151.89	CPLEX	99.14	4.53
	Gurobi	8.51	22.65

Table 2. Performance of Algorithm 2 on a 5×5 grid (i.e., n = 84 edges) for T = 110 time steps and L = 100 commodities.

	$\epsilon = 0.04$		$\epsilon = 0.02$		$\epsilon = 0.01$	
Run time (seconds)	$\langle \mathbf{C}, \mathbf{M} \rangle$	Violation ($\mathbf{M}$)	$\langle \mathbf{C}, \mathbf{M} \rangle$	Violation ($\mathbf{M}$)	$\langle \mathbf{C}, \mathbf{M} \rangle$	Violation ($\mathbf{M}$)
0.03125	302.05	5.74 e+02	300.60	4.61 e+02	300.26	4.62 e+02
0.125	303.74	1.22 e+02	301.23	1.08 e+02	300.47	9.68 e+01
0.5	306.13	3.07 e−04	302.12	3.89 e−15	300.57	6.11 e−15

Optimal objective value	Run time (seconds)	Primal simplex	Dual simplex
300.13	CPLEX	>3,600	38.86
	Gurobi	187.7917	67.5105

We solve the problem using Algorithm 2 and solve its node-edge formulation in the time-expanded network with the dual simplex algorithm in CPLEX and Gurobi. The performance measures introduced in the previous subsection are presented in Tables 4–7. As in the previous set of experiments, we see that the value of $\langle \mathbf{C}, \mathbf{M} \rangle$ is closest to the optimal objective value for the smallest tested regularization parameter ϵ. In three of the experiments (see Tables 4, 6, and 7), the method also converges fastest for the small regularization parameter. However, we can see that for the simulation with the smallest number of commodities, which is presented in Table 5, a larger regularization leads to faster convergence of the method. In fact, in this setting, all of our variants of Algorithm 2 take more time than the fastest LP solver. Nevertheless, in the experiments where the number of commodities is larger, our method is faster than the LP solvers. For the setting in Table 6, the proposed algorithm finds a close approximation to the optimal solution in 3.5% of the LP solver's run time. Here, we only compare with CPLEX's dual simplex method because this outperforms all the other LP methods in the previous experiments (i.e., Tables 1–5). Moreover, it is worth noting that for small-enough ϵ, the run time of our method does not significantly increase when the number of commodities is increased further; see Table 7. In the latter, we only show the results from the proposed method because the LP solvers do not converge within a reasonable time.

To conclude, even in the less favorable setting of a dense network with intermediate storage on the nodes, we may still get an approximate solution in less than an order of magnitude of CPLEX's run time (e.g., when large numbers of commodities are present in the network).

5.3. Traffic Routing Problem with a Large Number of Commodities

We apply our framework to a synthetic traffic routing problem in the street network illustrated in Figure 5, which consists of 57 nodes and 150 directed edges. Let every node be both a sink and a source, and as described in Section 3.3, we thus let the state space be of size $n = 150 + 2 \cdot 57 = 264$. An agent in the network is modeled by a unit of flow in a dynamic multicommodity network flow problem. Assume that there is an equal number of 10 agents traveling between every pair of nodes. This can be modeled by defining a commodity as the set of agents that share the same final destination. That means that every commodity has a total amount of flow (i.e., agents) $10|\mathcal{V}| = 570$, which is initially uniformly distributed on the set of sources and in the end, has traveled to the associated sink node. That is, for each commodity ℓ, the set of sources is defined as $\mathcal{S}_\ell^+ = \mathcal{V}$, and the sink is one node $\mathcal{S}_\ell^- \in \mathcal{V}$. Moreover, the set of sinks of all commodities is a disjoint set, and its union is $\mathcal{V}$. In particular, this means that the number of commodities is $L = 57$, and the two matrix constraints in (24) are defined by the matrices $R^{(0,1)}, R^{(0,\mathcal{T})} \in \mathbb{R}_+^{L \times n}$ with entries

Table 3. Performance of Algorithm 2 on a 10×10 grid (i.e., n = 364 edges) for T = 120 time steps and L = 100 commodities.

	$\epsilon = 0.08$		$\epsilon = 0.04$		$\epsilon = 0.02$	
Run time (seconds)	$\langle \mathbf{C}, \mathbf{M} \rangle$	Violation ($\mathbf{M}$)	$\langle \mathbf{C}, \mathbf{M} \rangle$	Violation ($\mathbf{M}$)	$\langle \mathbf{C}, \mathbf{M} \rangle$	Violation ($\mathbf{M}$)
0.125	607.24	6.26 e+02	579.51	4.20 e+02	572.50	4.34 e+02
0.5	588.95	7.95 e+01	576.20	6.68 e+01	572.18	6.33 e+01
2	595.93	7.24 e−02	578.05	2.42 e−02	572.37	4.88 e−15

Optimal objective value	Run time (seconds)	Primal simplex	Dual simplex
570.02	CPLEX	>3,600	427.04
	Gurobi	>3,600	>3,600

Table 4. Performance of Algorithm 2 on a dense network with $N = 50$, $T = 50$, $L = 500$.

	$\epsilon = 5 \cdot 10^{-3}$		$\epsilon = 2.5 \cdot 10^{-3}$		$\epsilon = 1.25 \cdot 10^{-3}$	
Run time (seconds)	$\langle \mathbf{C}, \mathbf{M} \rangle$	Violation ($\mathbf{M}$)	$\langle \mathbf{C}, \mathbf{M} \rangle$	Violation ($\mathbf{M}$)	$\langle \mathbf{C}, \mathbf{M} \rangle$	Violation ($\mathbf{M}$)
8	94.33	8.93 e+00	91.34	3.68 e+01	90.15	4.10 e+02
32	92.56	9.39 e−02	90.23	5.66 e−02	89.91	1.01 e−01
128	92.59	2.66 e−11	90.23	7.97 e−15	89.87	1.83 e−15

Optimal objective value	Run time (seconds)	Primal simplex	Dual simplex
89.75	CPLEX	>3,600	447.94
	Gurobi	>3,600	769.23

Table 5. Performance of Algorithm 2 on a dense network with $N = 50$, $T = 100$, and $L = 250$.

	$\epsilon = 5 \cdot 10^{-3}$		$\epsilon = 2.5 \cdot 10^{-3}$		$\epsilon = 1.25 \cdot 10^{-3}$	
Run time (seconds)	$\langle \mathbf{C}, \mathbf{M} \rangle$	Violation ($\mathbf{M}$)	$\langle \mathbf{C}, \mathbf{M} \rangle$	Violation ($\mathbf{M}$)	$\langle \mathbf{C}, \mathbf{M} \rangle$	Violation ($\mathbf{M}$)
8	50.61	1.21 e+02	46.81	2.42 e+02	45.39	3.47 e+02
32	48.19	2.52 e+00	46.58	2.28 e+00	45.62	5.69 e+00
128	48.49	5.07 e−04	45.24	1.23 e−02	44.86	1.66 e−01
256	48.49	7.88 e−08	45.24	1.01 e−04	44.85	6.62 e−02
512	48.49	4.72 e−14	45.24	9.46 e−06	44.85	2.79 e−02

Optimal objective value	Run time (seconds)	Primal simplex	Dual simplex
44.76	CPLEX	>3,600	454.77
	Gurobi	>3,600	744.20

Table 6. Performance of Algorithm 2 on a dense network with $N = 50$, $T = 100$, and $L = 500$.

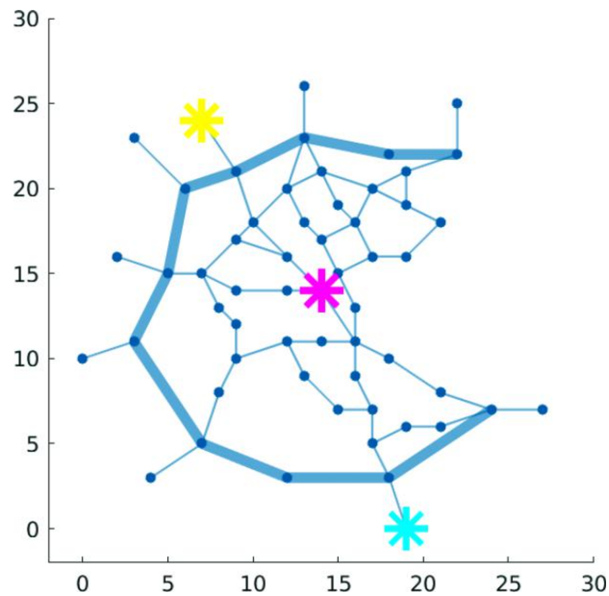
	$\epsilon = 5 \cdot 10^{-3}$		$\epsilon = 2.5 \cdot 10^{-3}$		$\epsilon = 1.25 \cdot 10^{-3}$	
Run time (seconds)	$\langle \mathbf{C}, \mathbf{M} \rangle$	Violation ($\mathbf{M}$)	$\langle \mathbf{C}, \mathbf{M} \rangle$	Violation ($\mathbf{M}$)	$\langle \mathbf{C}, \mathbf{M} \rangle$	Violation ($\mathbf{M}$)
8	95.58	6.05 e+02	91.71	7.05 e+02	90.27	7.99 e+02
32	96.14	2.25 e+00	92.59	7.99 e+00	90.70	3.49 e+01
128	97.96	4.34 e−01	90.66	1.95 e−02	89.98	9.22 e−02
256	97.96	2.28 e−05	90.68	7.76 e−09	89.97	4.70 e−08
512	97.96	4.69 e−13	90.68	5.65 e−15	89.97	1.35 e−15

Optimal objective value	Run time (seconds)	Dual simplex
89.75	CPLEX	14,810

Table 7. Performance of Algorithm 2 on a dense network with $N = 50$, $T = 100$, and $L = 1{,}000$.

	$\epsilon = 5 \cdot 10^{-3}$		$\epsilon = 2.5 \cdot 10^{-3}$		$\epsilon = 1.25 \cdot 10^{-3}$	
Run time (seconds)	$\langle \mathbf{C}, \mathbf{M} \rangle$	Violation ($\mathbf{M}$)	$\langle \mathbf{C}, \mathbf{M} \rangle$	Violation ($\mathbf{M}$)	$\langle \mathbf{C}, \mathbf{M} \rangle$	Violation ($\mathbf{M}$)
8	184.28	5.01 e+02	180.00	5.70 e+02	178.58	5.86 e+02
32	185.83	9.30 e+02	181.15	8.68 e+02	178.88	1.22 e+03
128	186.77	3.01 e−01	181.75	1.10 e+00	179.43	7.34 e+00
256	190.34	5.00 e+00	180.12	4.95 e−02	178.86	3.01 e−01
512	198.58	2.82 e−01	180.67	1.16 e−10	178.59	1.55 e−15

Figure 5. (Color online) Map of a street network. Every edge represents two directed edges, one in each direction. Broader edges represent highways. The three stars represent three different commodities' sinks.



$$R_{\ell,i}^{(0,1)} = \begin{cases} 10, & \text{if } i \in \mathcal{S}_\ell^+ = \mathcal{V}, \\ 0, & \text{otherwise,} \end{cases} \qquad R_{\ell,i}^{(0,\mathcal{T})} = \begin{cases} 570, & \text{if } i \in \mathcal{S}_\ell^-, \\ 0, & \text{otherwise.} \end{cases}$$

We consider the scenario with intermediate storage in the edges but without storage on the nodes. However, agents are permitted to stay in their respective sink and source, but once they leave their source, they may not return to it; also, once they reach their sink, they may not leave it. This structure is imposed by the cost matrix C as defined in (26). The wider streets in Figure 5 describe highways, and we denote the set of highways as $\mathcal{H}$. Because our framework assumes uniform travel time on all edges, the fact that the roads in $\mathcal{H}$ are longer than the other roads models that agents can drive faster on the highway. Let l_i denote the Euclidean length of road $i \in \mathcal{E}$. We define the capacities for each state as

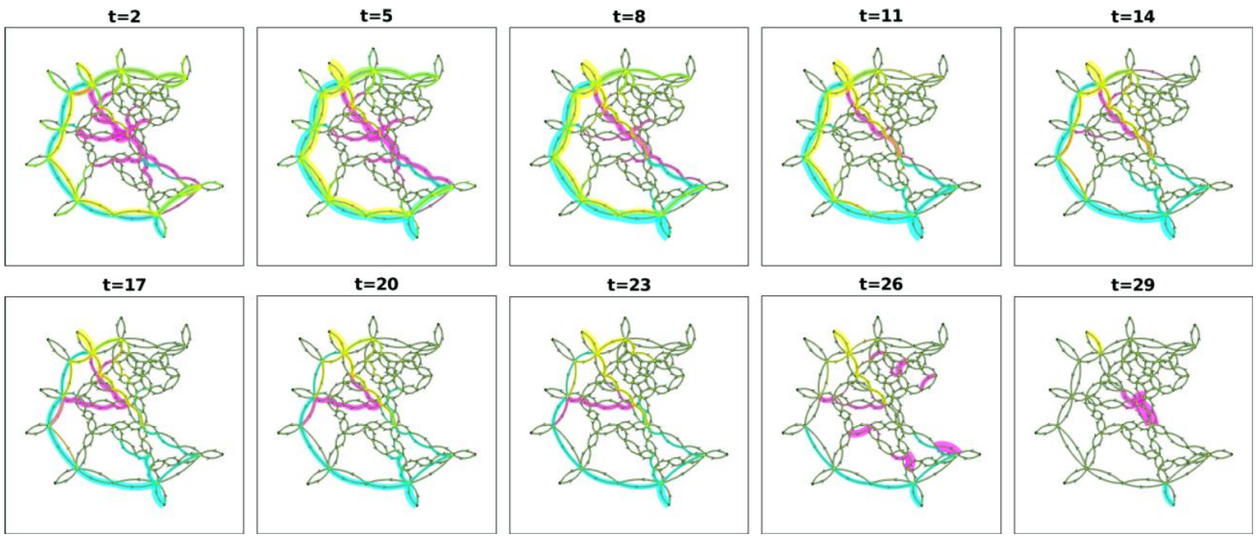
$$d_i = \begin{cases} 100 l_i, & \text{if } i \in \mathcal{H}, \\ 20 l_i, & \text{if } i \in \mathcal{E} \setminus \mathcal{H}, \\ 100L, & \text{if } i \in \mathcal{S}. \end{cases}$$

The cost for an agent to be in any of the states is defined in the matrix C_L. The costs are assumed equal for all agents and defined for all commodities $\ell = 1, \ldots, L$ as

$$(C_L)_{\ell i} = \begin{cases} 0.01, & \text{if } i \in \mathcal{S}^+, \\ 0.1, & \text{if } i \in \mathcal{E}, \\ 0, & \text{if } i \in \mathcal{S}^-. \end{cases} \tag{48}$$

Thus, the central controller aims to minimize the time agents spend inside the network and makes them reach the sink early rather than wait in the source. We consider the problem with final time $\mathcal{T} = 30$. The problem is solved using Algorithm 2 with regularization parameter $\epsilon = 0.01$. To illustrate the solution of the traffic routing problem, we will have a closer look at the flow of three commodities, namely the ones that correspond to the three sinks that are highlighted in Figure 5. The optimal flows for these three commodities are visualized in Figure 6. One can see that traffic is sent at all places in the network and finally concentrates toward the three sinks. For each commodity, we count the number of agents in the sources, roads, and sinks at each time interval. The number of agents in these three groups and for the three commodities is plotted over time in Figure 7(a). We also look at the total flow distribution (i.e., the accumulated distribution of the agents of all 57 commodities). The total flows distribution over source, roads, and sink over time can be seen by the blue lines in Figure 7(b). At the first time instance, many agents are sent from the sources into the network. Toward the end of the time interval, fewer and fewer agents are on the roads.

Figure 6. (Color online) The optimal traffic flow over time for three of the commodities.



We now vary the cost for agents to stay in the source; see Figure 7(b). Clearly, if the cost for being in a source is increased to $(C_L)_{\ell i} = 0.1$, for $i \in \mathcal{S}^+$ and $\ell = 1, \ldots, L$, more agents are sent into the network early on. If the cost for being in a source is equal to being in a sink (i.e., $(C_L)_{\ell i} = 0$) for $i \in \mathcal{S}^+$ and $\ell = 1, \ldots, L$, the amount of flow on the roads over time looks very symmetric.

Finally, we consider a scenario where a second type of commodity is present in the network. Therefore, the total number of commodities is increased to $2L = 114$. We interpret the first set of L commodities as cars and denote them as $\mathcal{L}_C$. The second set of L commodities is interpreted as trucks and denoted by $\mathcal{L}_T$. For each set of commodities, the initial and final distributions are defined as before, but the number of agents in each commodity is halved in order to get the same total number of agents. That is, we define the new constraint matrices $\hat{R}^{(0,1)}, \hat{R}^{(0,\mathcal{T})} \in \mathbb{R}_+^{2L\times n}$ as

$$\hat{R}^{(0,1)} = \frac{1}{2}\begin{bmatrix} R^{(0,1)} \\ R^{(0,1)} \end{bmatrix}, \qquad \hat{R}^{(0,\mathcal{T})} = \frac{1}{2}\begin{bmatrix} R^{(0,\mathcal{T})} \\ R^{(0,\mathcal{T})} \end{bmatrix}.$$

Figure 7. (Color online) Agents status over time. (a) Number of agents in sources, roads, and sinks over time for three commodities, where each color corresponds to one of the commodities in Figure 6. (b) Blue curves correspond to all agents in the scenario in Figure 6. Green and red curves describe the scenario, where the cost for staying in a source is equal to the cost on the roads (0.1) and to the cost for staying in a sink (0), respectively.

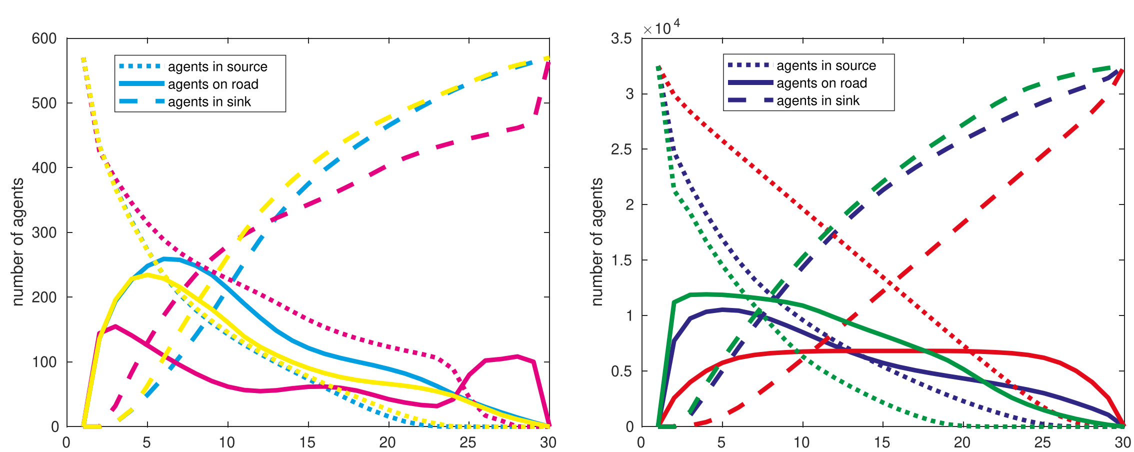
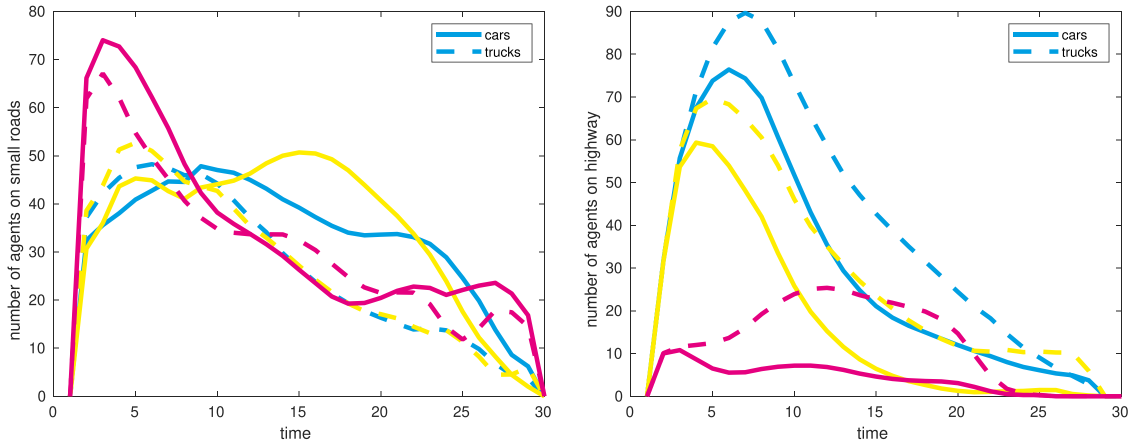


Figure 8. (Color online) Distribution of the six commodities on small roads and highways over time, where each color corresponds to the two commodities (cars and trucks) associated with the sink highlighted in the same color in Figure 5.



For the agents in $\mathcal{L}_C$, the costs to be on an edge, sink, or source are defined as before (i.e., for $\ell \in \mathcal{L}_C$, it is given by (48)). Trucks are incentivized to use highways as much as possible by an increased cost for agents in $\mathcal{L}_T$ to be on small roads. Thus, we define the modified cost matrix $\hat{C}_L \in \mathbb{R}_+^{2L\times n}$ by

$$(\hat{C}_L)_{\ell i} = \begin{cases} (C_L)_{\ell i}, & \text{if } \ell \in \mathcal{L}_C \\ 0.01, & \text{if } \ell \in \mathcal{L}_T, i \in \mathcal{S}^+, \\ 0.1, & \text{if } \ell \in \mathcal{L}_T, i \in \mathcal{H}, \\ 0.7, & \text{if } \ell \in \mathcal{L}_T, i \in \mathcal{E}\setminus\mathcal{H}, \\ 0, & \text{if } \ell \in \mathcal{L}_T, i \in \mathcal{S}^-. \end{cases}$$

The rest of the problem is set up as before, and we solve it with Algorithm 2 and regularization parameter $\epsilon = 0.01$. For each of the three sinks highlighted in Figure 5, we consider the two associated commodities and show the number of agents on the small roads and highways over time in Figure 8. As expected, the trucks avoid the small roads and mainly use the highways. In order to not exceed the capacity constraints on the highways, the cars are thus forced to the small roads.

6. Conclusion

We have developed a novel framework for dynamic network flow problems, which is based on formulating the problem as a structured multimarginal optimal transport problem. Regularizing the problem with an entropy term opens up for efficiently finding an approximate solution. By taking advantage of the graph structure in the optimal transport formulations, we derived a scheme that is computationally highly efficient, as well as easy to implement. Its competitiveness with state-of-the-art methods for network flow problems is experimentally illustrated in performance studies and on a traffic routing problem with a huge number of commodities.

Endnotes

[1] Often, the sources and sinks are defined on the nodes $\mathcal{V}$, not the edges $\mathcal{E}$. In this work, we consider the latter case; however, the framework introduced herein can easily be modified to define the sources and sinks on the nodes $\mathcal{V}$ instead.

[2] Note that there is a canonical bijection $(v_1, v_2) \leftrightarrow ((t-1, v_1), (t, v_2))$ between the edges $\mathcal{E}$ and the edges $\mathcal{E}_t$.

[3] If transport of mass is not allowed from position i to position j, then we let $C_{ij} = \infty$.

[4] That is, the second vertex of edge i is the first vertex of edge j in the network $\mathcal{N}$.

[5] A vertex is adjacent to all edges it connects to and to itself.

[6] We are not aware of any open-source solvers that are tailored for multicommodity network flow problems. However, state-of-the-art methods for multicommodity flows can typically not be expected to improve the run time by more than an order of magnitude as compared with standard LP solvers (Barnhart et al. [4], Khodayifar [42], Retvdri et al. [54]).

[7] Note that the standard Sinkhorn iterations typically converge slowly for small values of ϵ (Cuturi [20], Dvurechensky et al. [22]). Surprisingly, in these numerical experiments, the constraint violation decreases to machine precision faster for small values of ϵ. However, when ϵ becomes too small, the algorithm runs into numerical issues.

References

[1] Altschuler JM, Boix-Adsera E (2023) Polynomial-time algorithms for multimarginal optimal transport problems with structure. *Math. Programming* 199(1–2):1107–1178.

[2] Aronson J (1989) A survey of dynamic network flows. *Ann. Oper. Res.* 20(1):1–66.

[3] Bacon X (2020) Multi-species optimal transportation. *J. Optim. Theory Appl.* 184(2):315–337.

[4] Barnhart C, Krishnan N, Vance PH, Floudas C, Pardalos P (2009) Multicommodity flow problems. Floudas CA, Pardalos PM, eds. *Encyclopedia of Optimization* (Springer, New York), 2354–2362.

[5] Bauschke H, Lewis A (2000) Dykstras algorithm with Bregman projections: A convergence proof. *Optimization* 48(4):409–427.

[6] Benamou JD, Brenier Y (2000) A computational fluid mechanics solution to the Monge-Kantorovich mass transfer problem. *Numer. Math.* 84(3):375–393.

[7] Benamou JD, Brenier Y, Guittet K (2004) Numerical analysis of a multi-phasic mass transport problem. *Contemporary Math.* 353:1–18.

[8] Benamou JD, Carlier G, Cuturi M, Nenna L, Peyré G (2015) Iterative Bregman projections for regularized transportation problems. *SIAM J. Sci. Comput.* 37(2):A1111–A1138.

[9] Bertsekas DP, Tseng P (1988) Relaxation methods for minimum cost ordinary and generalized network flow problems. *Oper. Res.* 36(1):93–114.

[10] Bertsimas D, Patterson SS (2000) The traffic flow management rerouting problem in air traffic control: A dynamic network flow approach. *Transportation Sci.* 34(3):239–255.

[11] Boyd S, Vandenberghe L (2004) *Convex Optimization* (Cambridge University Press, Cambridge, UK).

[12] Brockett RW (2012) Notes on the control of the Liouville equation. Cannarsa P, Coron J-M, eds. *Control of Partial Differential Equations*, Chapter 2 (Springer, Berlin), 101–129.

[13] Carlino D, Depinet M, Khandelwal P, Stone P (2012) Approximately orchestrated routing and transportation analyzer: Large-scale traffic simulation for autonomous vehicles. *2012 15th Internat. IEEE Conf. Intelligent Transportation Systems* (IEEE, Piscataway, NJ), 334–339.

[14] Chen Y, Georgiou TT, Pavon M (2016) On the relation between optimal transport and Schrödinger bridges: A stochastic control viewpoint. *J. Optim. Theory Appl.* 169(2):671–691.

[15] Chen Y, Georgiou TT, Pavon M (2016) Optimal steering of a linear stochastic system to a final probability distribution. Part I. *IEEE Trans. Automatic Control* 61(5):1158–1169.

[16] Chen Y, Georgiou TT, Tannenbaum A (2018) Vector-valued optimal mass transport. *SIAM J. Appl. Math.* 78(3):1682–1696.

[17] Chen Y, Georgiou TT, Pavon M, Tannenbaum A (2016) Robust transport over networks. *IEEE Trans. Automatic Control* 62(9):4675–4682.

[18] Chen Y, Georgiou TT, Pavon M, Tannenbaum A (2017) Efficient robust routing for single commodity network flows. *IEEE Trans. Automatic Control* 63(7):2287–2294.

[19] Chen Y, Georgiou TT, Pavon M, Tannenbaum A (2019) Relaxed Schrödinger bridges and robust network routing. *IEEE Trans. Control Network Systems* 7(2):923–931.

[20] Cuturi M (2013) Sinkhorn distances: Lightspeed computation of optimal transport. *Adv. Neural Inform. Processing Systems* (Curran Associates, Inc., Red Hook, NY), 2292–2300.

[21] Diestel R (2017) *Graph Theory* (Springer, Berlin).

[22] Dvurechensky P, Gasnikov A, Kroshnin A (2018) Computational optimal transport: Complexity by accelerated gradient descent is better than by Sinkhorn's algorithm. *Internat. Conf. Machine Learn*. (PMLR, New York), 1367–1376.

[23] Elvander F, Haasler I, Jakobsson A, Karlsson J (2020) Multi-marginal optimal mass transport using partial information with applications in robust localization and sensor fusion. *Signal Processing* 171:107474.

[24] Farvolden JM, Powell WB, Lustig IJ (1993) A primal partitioning solution for the arc-chain formulation of a multicommodity network flow problem. *Oper. Res.* 41(4):669–693.

[25] Ford LR, Fulkerson DR (1958) A suggested computation for maximal multi-commodity network flows. *Management Sci.* 5(1):97–101.

[26] Ford LR, Fulkerson DR (1958) Constructing maximal dynamic flows from static flows. *Oper. Res.* 6(3):419–433.

[27] Ford LR, Fulkerson DR (1962) *Flows in Networks* (Princeton University Press, Princeton, NJ).

[28] Gangbo W, Święch A (1998) Optimal maps for the multidimensional Monge-Kantorovich problem. *Comm. Pure Appl. Math. Courant Inst. Math. Sci.* 51(1):23–45.

[29] Gendron B, Crainic TG, Frangioni A (1999) Multicommodity capacitated network design. Sansò B, Soriano P, eds. *Telecommunications Network Planning*, Chapter 1 (Springer, New York), 1–19.

[30] Gurobi Optimization LLC (2022) Gurobi optimizer reference manual. Accessed June 5, 2023, https://www.gurobi.com.

[31] Haasler I, Chen Y, Karlsson J (2020) Optimal steering of ensembles with origin-destination constraints. *IEEE Control Systems Lett.* 5(3):881–886.

[32] Haasler I, Ringh A, Chen Y, Karlsson J (2019) Estimating ensemble flows on a hidden Markov chain. *2019 IEEE 58th Conf. Decision Control (CDC)* (IEEE, Piscataway, NJ), 1331–1338.

[33] Haasler I, Ringh A, Chen Y, Karlsson J (2021) Multi-marginal optimal transport with a tree-structured cost and the Schrödinger bridge problem. *SIAM J. Control Optim.* 59(4):2428–2453.

[34] Haasler I, Singh R, Zhang Q, Karlsson J, Chen Y (2021) Multi-marginal optimal transport and probabilistic graphical models. *IEEE Trans. Inform. Theory* 67(7):4647–4668.

[35] Haghani A, Oh SC (1996) Formulation and solution of a multi-commodity, multi-modal network flow model for disaster relief operations. *Transportation Res. Part A Policy Pract.* 30(3):231–250.

[36] Hall A, Hippler S, Skutella M (2007) Multicommodity flows over time: Efficient algorithms and complexity. *Theoret. Comput. Sci.* 379(3):387–404.

[37] IBM ILOG CPLEX (2019) Optimization Studio 12.10.0: CP optimizer online documentation. Acccessed June 5, 2023, https://www.ibm.com/docs/en/icos/12.10.0.

[38] Jones K, Lustig I, Farvolden J, Powell W (1993) Multicommodity network flows: The impact of formulation on decomposition. *Math. Programming* 62(1–3):95–117.

[39] Karlsson J, Ringh A (2017) Generalized Sinkhorn iterations for regularizing inverse problems using optimal mass transport. *SIAM J. Imaging Sci.* 10(4):1935–1962.

[40] Kennington J, Shalaby M (1977) An effective subgradient procedure for minimal cost multicommodity flow problems. *Management Sci.* 23(9):994–1004.

[41] Kennington JL (1978) A survey of linear cost multicommodity network flows. *Oper. Res.* 26(2):209–236.

[42] Khodayifar S (2021) Minimum cost multicommodity network flow problem in time-varying networks: By decomposition principle. *Optim. Lett.* 15(2):1009–1026.

[43] Léonard C (2014) A survey of the Schrödinger problem and some of its connections with optimal transport. *Discrete Continuous Dynamical Systems* 34(4):1533–1574.

[44] Levinson J, Askeland J, Becker J, Dolson J, Held D, Kammel S, Kolter J, et al. (2011) Toward fully autonomous driving: Systems and algorithms. *2011 IEEE Intelligent Vehicles Sympos. (IV)* (IEEE, Piscataway, NJ), 163–168.

[45] Lin T, Ho N, Cuturi M, Jordan MI (2022) On the complexity of approximating multimarginal optimal transport. *J. Machine Learn. Res.* 23(65):1–43.

[46] Lin T, Ho N, Chen X, Cuturi M, Jordan M (2020) Fixed-support Wasserstein barycenters: Computational hardness and fast algorithm. Larochelle H, Ranzato M, Hadsell R, Balcan MF, Lin H, eds. *Adv. Neural Inform. Processing Systems*, vol. 33 (Curran Associates, Inc., Red Hook, NY), 5368–5380.

[47] Luo ZQ, Tseng P (1992) On the convergence of the coordinate descent method for convex differentiable minimization. *J. Optim. Theory Appl.* 72(1):7–35.

[48] McBride R (1998) Progress made in solving the multicommodity flow problem. *SIAM J. Optim.* 8(4):947–955.

[49] Nenna L (2016) Numerical methods for multi-marginal optimal transportation. PhD thesis, PSL, Paris.

[50] Pasquale C, Sacone S, Siri S, Ferrara A (2019) Traffic control for freeway networks with sustainability-related objectives: Review and future challenges. *Annual Rev. Control* 48:312–324.

[51] Pass B (2015) Multi-marginal optimal transport: Theory and applications. *ESAIM Math. Model. Numer. Anal.* 49(6):1771–1790.

[52] Pavon M, Ticozzi F (2010) Discrete-time classical and quantum Markovian evolutions: Maximum entropy problems on path space. *J. Math. Phys.* 51(4):042104.

[53] Peyré G, Cuturi M (2019) Computational optimal transport: With applications to data science. *Foundations Trends Machine Learn.* 11(5–6):355–607.

[54] Retvdri G, Bíró J, Cinkler T (2004) A novel Lagrangian-relaxation to the minimum cost multicommodity flow problem and its application to ospf traffic engineering. *Proc. ISCC 2004. Ninth Internat. Sympos. Comput. Comm.*, vol. 2 (IEEE, Piscataway, NJ), 957–962.

[55] Rüschendorf L (1995) Optimal solutions of multivariate coupling problems. *Appl. Math.* 23(3):325–338.

[56] Rüschendorf L, Uckelmann L (2002) On the n-coupling problem. *J. Multivariate Anal.* 81(2):242–258.

[57] Tomlin JA (1966) Minimum-cost multicommodity network flows. *Oper. Res.* 14(1):45–51.

[58] Tseng P (1990) Dual ascent methods for problems with strictly convex costs and linear constraints: A unified approach. *SIAM J. Control Optim.* 28(1):214–242.

[59] Villani C (2008) *Optimal Transport: Old and New* (Springer, Berlin).

[60] Wang IL (2018) Multicommodity network flows: A survey. Part I. Applications and formulations. *Internat. J. Oper. Res.* 15(4):145–153.

[61] Yamada T (1996) A network flow approach to a city emergency evacuation planning. *Internat. J. Systems Sci.* 27(10):931–936.