SIAM J. IMAGING SCIENCES
Vol. 11, No. 2, pp. 1317–1362

Image Labeling Based on Graphical Models Using Wasserstein Messages and Geometric Assignment*

Ruben Hühnerbein†, Fabrizio Savarino†, Freddie Åström‡, and Christoph Schnörr†

Abstract. We introduce a novel approach to Maximum A Posteriori (MAP) inference based on discrete graphical models. By utilizing local Wasserstein distances for coupling assignment measures across edges of the underlying graph, a given discrete objective function is smoothly approximated and restricted to the assignment manifold. A corresponding multiplicative update scheme combines in a single process (i) geometric integration of the resulting Riemannian gradient flow, and (ii) rounding to integral solutions that represent valid labelings. Throughout this process, local marginalization constraints known from the established LP relaxation are satisfied, whereas the smooth geometric setting results in rapidly converging iterations that can be carried out in parallel for every edge.

Key words. image labeling, assignment manifold, Fisher–Rao metric, Riemannian gradient flow, discrete optimal transport, Wasserstein distance, entropic regularization, graphical models

AMS subject classifications. 62H35, 62M40, 65K10, 68U10

DOI. 10.1137/17M1150669

1. Introduction.

1.1. Overview and motivation.

Let $\Omega \subset \mathbb{R}^2$ be a domain where image data are observed, and let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, $|\mathcal{V}| = m$, denote a grid graph embedded into Ω. Each vertex $i \in \mathcal{V}$ indexes the location of a pixel, to which a random variable

$$x_i \in \mathcal{X} = \{\ell_1, \dots, \ell_n\} \tag{1.1}$$

is assigned, which takes values in a finite set $\mathcal{X}$ of *labels*. The *image labeling problem* is the task of assigning to each x_i a label such that the discrete *objective function*

$$\min_{x \in \mathcal{X}^m} E(x), \qquad E(x) = \sum_{i \in \mathcal{V}} E_i(x_i) + \sum_{ij \in \mathcal{E}} E_{ij}(x_i, x_j) \tag{1.2}$$

is minimized. This function comprises for each pixel $i \in \mathcal{V}$ local energy terms $E_i(x_i)$ that evaluate local label predictions for each possible value of $x_i \in \mathcal{X}$. In addition, $E(x)$ comprises for each edge $ij \in \mathcal{E}$ local distance functions $E_{ij}(x_i, x_j)$ that evaluate the joint assignment of

*Received by the editors October 4, 2017; accepted for publication (in revised form) March 1, 2018; published electronically May 24, 2018.
http://www.siam.org/journals/siims/11-2/M115066.html
Funding: The work of the authors was supported by the German Science Foundation, grant GRK 1653.

†Image and Pattern Analysis Group, Heidelberg University, Heidelberg 69120, Germany, http://ipa.math.uni-heidelberg.de (ruben.huehnerbein@iwr.uni-heidelberg.de, fabrizio.savarino@iwr.uni-heidelberg.de, schnoerr@math.uni-heidelberg.de).
‡Heidelberg Collaboratory for Image Processing, Heidelberg University, Heidelberg 69120, Germany (freddie.astroem@iwr.uni-heidelberg.de, https://hciweb.iwr.uni-heidelberg.de/user/fastroem).

labels to x_i and x_j. If the local energy functions $E_{ij}(x_i, x_j) = d(x_i, x_j)$ are defined by a metric $d\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, then (1.2) is called the *metric labeling problem* [25]. In general, the presence of these latter terms makes image labeling a combinatorially hard task. Function $E(x)$ has the common format of variational problems for image analysis comprising a data term and a regularizer. From a Bayesian perspective, therefore, minimizing E corresponds to *maximum a posteriori* inference with respect to the probability distribution $p(x) = \frac{1}{Z}\exp(-E(x))$. We refer the reader to [23] for a recent survey on the image labeling problem and on algorithms for solving either approximately or exactly problem (1.2).

A major class of algorithms for approximately solving (1.2) is based on the *linear* (programming) *relaxation* [52] (see section 2.2 for details)

$$\min_{\mu \in \mathcal{L}_\mathcal{G}} \langle \theta, \mu \rangle. \tag{1.3}$$

Solving the linear program (LP) (1.3) returns a globally optimal *relaxed indicator vector* μ, whose components take values in $[0, 1]$. If μ is a binary vector, then it corresponds to a solution of problem (1.2). In realistic applications, this is not the case, however, and the relaxed solution μ has to be rounded to an integral solution in a postprocessing step.

In this paper, we present an alternative inference algorithm that deviates from the traditional two-step process: convex relaxation and rounding. It is based on the recently proposed geometric approach [5] to image labeling. The basic idea underlying this approach is to restrict indicator vector fields to the relative interior of the probability simplex, equipped with the Fisher–Rao metric, and to regularize label assignments by iteratively computing Riemannian means (see section 3 for details). This results in a highly parallel, multiplicative update scheme that rapidly converges to an integral solution. Because this model of label assignment does not interfere with data representation, the approach applies to any data given in a metric space. Adopting this starting point, the objectives of the present paper are as follows:

- Show how the approach [5] can be used and extended to efficiently compute a high-quality (low-energy) solution for an arbitrary given instance of the labeling problem (1.2).
- Devise a novel labeling algorithm that tightly integrates both relaxation and rounding to an integral solution in a single process.
- Stick to the smooth geometric model suggested by [5] so as to overcome the inherent nonsmoothness of convex polyhedral relaxations and the slow convergence of corresponding first-order iterative methods of convex programming.

Regarding the last point, a key ingredient of our approach is a *smooth* approximation

$$E_\tau(\mu_\mathcal{V}) = \langle \theta_\mathcal{V}, \mu_\mathcal{V} \rangle + \sum_{ij \in \mathcal{E}} d_{\theta_{ij},\tau}(\mu_i, \mu_j), \qquad \tau > 0, \tag{1.4}$$

of problem (1.3), where $d_{\theta_{ij},\tau}$ denotes the local *smoothed* Wasserstein distance between the discrete label assignment measures μ_i, μ_j coupled along the edge ij of the underlying graph. Besides achieving the degree of smoothness required for our geometric setting, this approximation also properly takes into account the regularization parameters that are specified in terms of the local energy terms E_{ij} of the labeling problem (1.2). Our approach restricts the function E_τ to the so-called assignment manifold and iteratively determines a labeling by

tightly combining geometric optimization with rounding to an integral solution in a smooth fashion.

1.2. Related work. Problem sizes of LP (1.3) are large in typical applications of image labeling, which rules out the use of standard LP codes. In particular, the theoretically and practically most efficient interior point methods based on self-concordant barrier functions [31, 38] are infeasible due to the dense linear algebra steps required to determine search and update directions.

Therefore, the need for dedicated solvers for the LP relaxation (1.3) has stimulated a lot of research. A prominent example constitutes subclasses of objective functions (1.2) as studied in [28], in particular binary submodular functions, that enable reformulation of the labeling problem as a maximum-flow problem in an associated network and the application of discrete combinatorial solvers [12, 11].

Since the structure of such algorithms inherently limits fine-grained parallel implementations, however, *belief propagation (BP)* and variants [54] have been popular among practitioners. These fixed point schemes in terms of dual variables iteratively enforce the so-called local polytope constraints that define the feasible set of the LP relaxation (1.3). They can be efficiently implemented using "message passing" and exploit the structure of the underlying graph. Although convergence is not guaranteed on cyclic graphs, the performance in practice may be good [53]. The theoretical deficiencies of basic BP in turn stimulated research on *convergent* message passing schemes, either using heuristic damping or utilizing in a more principled way *convexity*. Prominent examples of the latter case are [49, 22]. We refer the reader to [23] for many more references and a comprehensive experimental evaluation of a broad range of algorithms for image labeling.

The feasible set of the relaxation (1.3) is a superset of the original feasible set of (1.2). Therefore, globally optimal solutions to (1.3) generally do *not* constitute valid labelings but comprise *non*integral components $\mu_i(x_i) \in (0,1)$, $x_i \in \mathcal{X}$, $i \in \mathcal{V}$. Randomized rounding schemes for converting a relaxed solution vector $\overline{\mu}$ to a valid labeling $x \in \mathcal{X}^m$, along with suboptimality bounds, were studied in [25, 15]. The problem of inferring components x_i^* of the unknown globally optimal *combinatorial* labeling that minimizes (1.2), through partial optimality and persistency, was studied in [47]. We refer the reader to [52] for the history and more information about the LP relaxation of labeling problems, and to [50] for connections to discrete probabilistic graphical models from the variational viewpoint.

The approach in [37] applies the mirror descent scheme [30] to the LP (1.3). This amounts to sequential proximal minimization [41], yet it uses a Bregman distance as proximity measure instead of the squared Euclidean distance [14]. A key technical aspect concerns the proper choice of entropy functions related to the underlying graphical model that qualify as convex functions of Legendre type (cf. [6]). The authors of [37] observed a fast convergence rate. However, the scheme does not scale up to the typically large problem sizes used in image analysis, especially when graphical models with higher edge connectivity are considered, due to the memory requirements when working entirely in the primal domain.

Optimal transport and the *Wasserstein distance* have become major tools of signal modeling and analysis [29]. In connection with the metric labeling problem, using the Wasserstein distance (aka optimal transport costs, earthmover metrics) was proposed in [1, 15]. These

works study bounds on the integrality gap of an "earthmover LP" and performance guarantees of rounding procedures applied as postprocessing. While the earthmover LP corresponds to our approach (1.4) *without* smoothing, authors do not specify how to solve such LPs efficiently, especially when the LP relates to large-scale graphical models as in image analysis. Moreover, the bounds derived by [1] become weak with increasing numbers of variables, which are fairly large in typical problems of image analysis. In contrast, the focus of the present paper is on a *smooth geometric* problem reformulation that scales well with both the problem size and the number of labels, and performs rounding *simultaneously*. If and how theoretical guarantees regarding the integrality gap and rounding carry over to our setting is an interesting open problem of future research.

Regarding the finite-dimensional formulation of optimal discrete transport in terms of LPs, the design of efficient algorithms for large-scale problems requires sophisticated techniques [43]. The problems of discrete optimal transport studied in this paper, in connection with the local Wasserstein distances of (1.4), have a small or moderate size (n^2: number of labels squared). We apply the standard device of enhancing convexity through entropic regularization, which increases smoothness in the dual domain. We refer the reader to [45] and [13, Chap. 9] for basic related work, the connection to matrix scaling algorithms, and history. When entropic regularization is very weak and for large problem sizes, the related fixed point iteration suffers from numerical instability, and dedicated methods for handling them have been proposed [44]. Smoothing of the Wasserstein distance and Sinkhorn's algorithm have become popular in machine learning due to [16]. The authors of [34, 17] comprehensively investigated barycenters and interpolation based on the Wasserstein distance. Our approach to image labeling, in conjunction with the geometric approach of [5], is novel and elaborates [4].

Finally, since our approach is defined on a graph and works with data on a graph, our work may be assigned to the broad class of nonlocal methods for image analysis on graphs, from a more general viewpoint. Recent major related work includes [9] on the connection between the Ginzburg–Landau functional for binary regularized segmentation and spectral clustering, and [8] on generalizing PDE-like models on graphs to manifold-valued data. We refer the reader to the bibliographies in these works and to the seminal papers [2] on regularized variational segmentation using Γ-convergence and [21, 20] on nonlocal variational image processing on graphs that initiated these rapidly evolving lines of research. The focus of the present paper, however, is on discrete graphical models and the corresponding labeling problem, in terms of any discrete objective function of the form (1.2).

1.3. Contribution and organization. We collect basic notation, background material, and details of the LP relaxation (1.3) in section 2. Section 3 summarizes the basic concepts of the geometric labeling approach of [5], in particular the so-called assignment manifold, and the general framework of [42] for numerically integrating Riemannian gradient flows of functionals defined on the assignment manifold. This section provides the basis for the two subsequent sections that contain our main contribution.

Section 4 studies the approximation (1.4) and provides explicit expressions for the Riemannian gradient of the restriction of E_τ to the assignment manifold. A key property of this setup concerns the local polytope constraints that define the feasible set $\mathcal{L}_\mathcal{G}$ of the LP relaxation (1.3): by construction, they are *always* satisfied throughout the resulting itera-

tive process of label assignment. Thus, our formulation is *both more tightly constrained and smooth*, in contrast to the established convex programming approaches based on (1.3).

Section 5 details the combination of all ingredients into a *single*, smooth, geometric approach that performs simultaneously minimization of the objective function (1.4) and rounding to an integral solution (label assignment). This tight integration is a second major property that distinguishes our approach from related work. Section 5 also explains the notion of "Wasserstein messages" in the title of this paper due to the dual variables that are numerically utilized to evaluate gradients of local Wasserstein distances, akin to how dual (multiplier) variables in basic BP schemes are used to enforce local marginalization constraints. Unlike the latter computations, they have the structure of message passing on a dataflow architecture; however, message passing induced by our approach is fully parallel along all edges of the underlying graph and hence resembles the structure of numerical solvers for PDEs.

The remaining two sections are devoted to numerical evaluations of our approach. The recent paper [7] reports a convergence analysis and the application of the scheme of [5] to a range of challenging labeling problems of manifold-valued data. The results of [7] concern the boundary of the underlying simplex domains, however, which are excluded from the assignment manifold by definition. In addition, the approach worked out in this paper extends [5] so that any convergence results regarding [5] would not directly apply to the present paper. To keep this paper at a reasonable length, we merely considered the most elementary iterative update scheme, based on the geometric integration of the Riemannian gradient flow with the (geometric) explicit Euler scheme. The potential of the framework outlined by [42] for more sophisticated numerical schemes will be explored elsewhere along with establishing bounds for parameter values that provably ensure stability of numerical integration of the underlying gradient flow. Furthermore, working out any realistic application is beyond the scope of this paper. Rather, the experimental results demonstrate major properties of our approach.

Section 6 provides all details of our implementation that are required to reproduce our computational results. Section 7 reports and discusses the results of the following four types of experiments:

1. We study the interplay between two parameters τ and α that control smoothness of the approximation (1.4) and rounding, respectively. In order to minimize efficiently (1.2), the Riemannian flow with respect to the smooth approximation (1.4) must reveal proper descent directions. This imposes an upper bound on the smoothing parameter τ. Naturally, the effect of rounding has to be stronger to make the iterative process converge to an integral solution. A corresponding choice of α controls the compromise between quality of integral labelings in terms of the energy (1.4) and speed of convergence. Fortunately, the upper bound on τ is large enough to achieve attractive convergence rates.
2. We comprehensively explore numerically the entire model space of the minimal binary graphical model on the *cyclic* triangle graph $\mathcal{K}^3$, whose relaxation in terms of the so-called *local* polytope already constitutes a superset of the *marginal* polytope as an admissible set for valid integral labelings. In this way, we explore the performance of our approach in view of the LP relaxation and established inference based on convex programming, and with respect to the (generally intractable) feasible set of integral solutions. Corresponding phase diagrams display and support quantitatively the trade-

off between accuracy of optimization and rate of convergence through the choice of the single parameter α.

3. We conducted a labeling problem of the usual size to confirm and demonstrate that the finding of the preceding points for "all" models on $\mathcal{K}^3$ also holds in a typical application. A comparison to sequential tree-reweighted message passing (TRWS) [27], which defines the state of the art, and to loopy belief propagation (loopy-BP) based on the OpenGM package [3], shows that our approach is on par with these methods regarding the energy level $E(x)$ of the resulting labeling x. Regarding runtime on an off-the-shelf PC, our (nonoptimized) research code runs as fast as the OpenGM implementation of loopy BP, whereas TRWS terminates more rapidly on such sequential machines. The TRWS algorithm, however, does not exhibit the PDE-like structure of our approach that enables massive parallel implementations.
4. In a final experiment based on the graphical model with a pronounced nonuniform (non-Potts) prior, we demonstrate that our approach is able to perform inference for any given graphical model.

We conclude in section 8 and relegate some proofs to an appendix to minimize interruption of the overall line of reasoning.

2. Preliminaries.

We introduce basic notation in section 2.1 and the common LP relaxation of the labeling problem in section 2.2. In order to clearly distinguish between the LP relaxation and our geometric approach to the labeling problem based on [5] (see section 3.1), we keep the standard notation in the literature for the former approach and the notation from [5] for the latter. Remark 3.1 below identifies variables of both approaches that play a similar role.

2.1. Basic notation.

For an *undirected* graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, the adjacency relation $i \sim j$ means that vertices i and j are connected by an undirected edge $ij \in \mathcal{E}$, where the latter denotes the *unordered* pair $\{i, j\} = ij = ji$. The neighbors of vertex i form the set

$$\mathcal{N}(i) = \{j \in \mathcal{V} \colon i \sim j\} \tag{2.1}$$

of all vertices adjacent to i, and its cardinality $d(i) = |\mathcal{N}(i)|$ is the degree of i. $\mathcal{G}$ is turned into a *directed* graph by assigning an *orientation* to every edge ij, which then forms *ordered* pairs $(i, j) \neq (j, i)$. By abuse of notation we also sometimes write $(i, j) = ij$ in the oriented case; however, the exact meaning will be clear from the context. We only consider graphs *without multiple* edges between any pair of nodes $i, j \in \mathcal{V}$.

We use the abbreviation $[n] = \{1, 2, \dots, n\}$ for $n \in \mathbb{N}$. $\overline{\mathbb{R}} = \mathbb{R} \cup \{+\infty\}$ denotes the extended real line. All vectors are regarded as column vectors, and $x^\top$ denotes transposition of a vector x. We ignore transposition, however, when vectors are explicitly specified by their components; e.g., we write $x = (y, z)$ instead of the more cumbersome $x = (y^\top, z^\top)^\top$. We set $\mathbb{1}_n = (1, 1, \dots, 1) \in \mathbb{N}^n$ and write $\mathbb{1}$ if n is clear from the context. $\langle x, y \rangle = \sum_{i \in [n]} x_i y_i$ denotes the Euclidean inner product. Given a matrix

$$A = \begin{pmatrix} A_1 \\ \vdots \\ A_m \end{pmatrix} = \begin{pmatrix} A^1 \dots A^n \end{pmatrix} \in \mathbb{R}^{m \times n}, \tag{2.2}$$

we denote the row vectors by A_i, $i \in [m]$, and the column vectors by A^j, $j \in [n]$. The canonical matrix inner product is $\langle A, B\rangle = \operatorname{tr}(A^\top B)$, where tr denotes the trace of a matrix, i.e., $\operatorname{tr}(A^\top B) = \sum_{i\in[m]}\langle A_i, B_i\rangle = \sum_{j\in[n]}\langle A^j, B^j\rangle = \sum_{i\in[m],j\in[n]} A_{ij}B_{ij}$. Superscripts in brackets, e.g. $, A_i^{(k)}$, index iterative steps.

The set of nonnegative vectors $x \in \mathbb{R}^n$ is denoted by $\mathbb{R}^n_+$ and the set of strictly positive vectors by $\mathbb{R}^n_{++}$. The probability simplex $\Delta_n = \{p \in \mathbb{R}^n_+ : \langle \mathbb{1}_n, p\rangle = 1\}$ contains all discrete distributions on $[n]$. A doubly stochastic matrix $\mu_{ij} \in \mathbb{R}^{n\times n}_+$, also called *coupling measure* in this paper in connection with discrete optimal transport, has the property $\mu_{ij}\mathbb{1}_n \in \Delta_n$ and $\mu_{ij}^\top \mathbb{1}_n \in \Delta_n$. We denote these two *marginal distributions* of μ_{ij} by μ_i and μ_j, respectively, and the linear mapping for extracting them by

$$\mathcal{A}\colon \mathbb{R}^{n\times n} \to \mathbb{R}^{2n}, \qquad \mu_{ij} \mapsto \mathcal{A}\mu_{ij} = \begin{pmatrix} \mu_{ij}\mathbb{1}_n \\ \mu_{ij}^\top \mathbb{1}_n \end{pmatrix} = \begin{pmatrix} \mu_i \\ \mu_j \end{pmatrix}. \tag{2.3a}$$

Its transpose is given by

$$\mathcal{A}^\top : \mathbb{R}^{2n} \to \mathbb{R}^{n\times n}, \qquad (\nu_i, \nu_j) \mapsto \mathcal{A}^\top \begin{pmatrix} \nu_i \\ \nu_j \end{pmatrix} = \nu_i \mathbb{1}_n^\top + \mathbb{1}_n \nu_j^\top. \tag{2.3b}$$

The kernel (nullspace) of a linear mapping $\mathcal{A}$ is denoted by $\mathcal{N}(\mathcal{A})$ and its range by $\mathcal{R}(\mathcal{A})$.

The functions exp, log apply *componentwise* to strictly positive vectors $x \in \mathbb{R}^n_{++}$, e.g., $e^x = (e^{x_1}, \dots, e^{x_n})$, and similarly for strictly positive matrices. Likewise, if $x, y \in \mathbb{R}^n_{++}$, then we simply write

$$x \cdot y = (x_1 y_1, \dots, x_n y_n), \qquad \frac{x}{y} = \left(\frac{x_1}{y_1}, \dots, \frac{x_n}{y_n}\right) \tag{2.4}$$

for the *componentwise* multiplication and division.

We define $\mathcal{F}_0$ to be the class of proper, lower-semicontinuous, and convex functions defined on $\mathbb{R}^n$. For any function $f \in \mathcal{F}_0$, $\partial f(x)$ denotes its subdifferential at x, and the conjugate function $f^* \in \mathcal{F}_0$ of f is given by the Legendre–Fenchel transform (cf. [40, section 11.A])

$$f^*(y) := \sup_{x\in\mathbb{R}^n} \{\langle y, x\rangle - f(x)\}. \tag{2.5}$$

For a given closed convex set C, its indicator function is denoted by

$$\delta_C(x) := \begin{cases} 0 & \text{if } x \in C, \\ +\infty & \text{otherwise,} \end{cases} \tag{2.6}$$

and

$$P_C\colon \mathbb{R}^n \to C, \qquad P_C(x) := \operatorname{argmin}_{y\in C} \|x - y\| \tag{2.7}$$

denotes the orthogonal projection onto C. The shorthand "s.t." means "subject to" in connection to the specification of constraints.

The *log-exponential* function $\operatorname{logexp}_\varepsilon \in \mathcal{F}_0$ is defined as

$$\operatorname{logexp}_\varepsilon(x) := \varepsilon \log\left(\sum_{i\in[n]} e^{\frac{x_i}{\varepsilon}}\right). \tag{2.8a}$$

It uniformly approximates the function $\operatorname{vecmax} \in \mathcal{F}_0$ [40, Ex. 1.30], i.e.,

$$\lim_{\varepsilon \searrow 0} \operatorname{logexp}_\varepsilon(x) = \operatorname{vecmax}(x) = \max\{x_i\}_{i\in[n]}. \tag{2.8b}$$

We will use the following basic result from convex analysis (cf., e.g., [40, Chap. 11]), where $\partial f(x)$ denotes the subdifferential of a function $f \in \mathcal{F}_0$ at x.

Theorem 2.1 (inversion rule for subgradients). *Let $f \in \mathcal{F}_0$. Then*

$$\hat{p} \in \partial f(\hat{x}) \quad \Leftrightarrow \quad \hat{x} \in \partial f^*(\hat{p}) \quad \Leftrightarrow \quad f(\hat{x}) + f^*(\hat{p}) = \langle \hat{p}, \hat{x}\rangle. \tag{2.9}$$

We will also apply the following classical theorem of Danskin and its extension by Rockafellar.

Theorem 2.2 (see [18, 39]). *Let $f(z) = \max_{w\in W} g(z,w)$, where W is compact and the function $g(\cdot, w)$ is differentiable and $\nabla_z g(z,w)$ is continuously dependent on (z,w). If in addition $g(z,w)$ is convex in z, and if $\overline{z}$ is a point such that $\arg\max_{w\in W} g(\overline{z}, w) = \{\overline{w}\}$, then f is differentiable at $\overline{z}$ with*

$$\nabla f(\overline{z}) = \nabla_z g(\overline{z}, \overline{w}). \tag{2.10}$$

2.2. The local polytope relaxation of the labeling problem. We sketch in this section the transition from the discrete energy minimization problem (1.2) to the LP relaxation (1.3) and thereby introduce additional notation needed in subsequent sections.

The first step concerns the definition of *local model parameter vectors* and *matrices*

$$\theta_i := \big(\theta_i(\ell_k)\big)_{k\in[n]} \in \mathbb{R}^n, \qquad \theta_{ij} := \big(\theta_{ij}(\ell_k,\ell_r)\big)_{k,r\in[n]} \in \mathbb{R}^{n\times n}, \qquad \text{with} \quad \ell_k, \ell_r \in \mathcal{X}, \tag{2.11}$$

which merely encode the values of the discrete objective function (1.2): $\theta_i(\ell_k) = E_i(\ell_k)$, $\theta_{ij}(\ell_k,\ell_r) = E_{ij}(\ell_k,\ell_r)$. These local terms are commonly called *unary* and *pairwise terms* in the literature. Recall from the discussion of (1.2) that the unary terms represent the data and the pairwise terms specify a regularizer. All these local terms are indexed by the vertices $i \in \mathcal{V}$ and edges $ij \in \mathcal{E}$ of the underlying graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and assembled into the vectors

$$\theta := (\theta_\mathcal{V}, \theta_\mathcal{E}), \qquad \text{where} \qquad \theta_\mathcal{V} := (\theta_i)_{i\in\mathcal{V}}, \qquad \text{and} \qquad \theta_\mathcal{E} := (\theta_{ij})_{ij\in\mathcal{E}}, \tag{2.12}$$

where we conveniently regard $\theta_{ij} \in \mathbb{R}^{n^2}$ as either local vector or local matrix $\theta_{ij} \in \mathbb{R}^{n\times n}$, depending on the context. Next we define *local indicator vectors*

$$\mu_i := \big(\mu_i(\ell_k)\big)_{k\in[n]} \in \{0,1\}^n, \quad \mu_{ij} := \big(\mu_{ij}(\ell_k,\ell_r)\big)_{k,r\in[n]} \in \{0,1\}^{n\times n}, \quad \text{with} \quad \ell_k, \ell_r \in \mathcal{X}, \tag{2.13}$$

indexed in the same way as (2.11) and assembled into the vectors

$$\mu := (\mu_{\mathcal{V}}, \mu_{\mathcal{E}}), \qquad \text{where} \qquad \mu_{\mathcal{V}} := (\mu_i)_{i\in\mathcal{V}}, \qquad \text{and} \qquad \mu_{\mathcal{E}} := (\mu_{ij})_{ij\in\mathcal{E}}. \tag{2.14}$$

The combinatorial optimization problem (1.2) now reads $\min_\mu \langle \theta, \mu\rangle$. The corresponding LP relaxation consists of replacing the discrete feasible set of (2.13) by the convex polyhedral sets

$$\mu_i \in \Delta_n, \quad \mu_{ij} \in \Pi(\mu_i, \mu_j), \qquad i \in \mathcal{V},\ ij \in \mathcal{E}, \tag{2.15a}$$

$$\Pi(\mu_i, \mu_j) := \big\{\mu_{ij} \in \mathbb{R}_+^{n\times n} \colon \mu_{ij}\mathbb{1} = \mu_i,\ \mu_{ij}^\top \mathbb{1} = \mu_j,\ \mu_i, \mu_j \in \Delta_n\big\}. \tag{2.15b}$$

As a result, the LP relaxation (1.3) of (1.2) reads more explicitly as

$$\min_{\mu\in\mathcal{L}_\mathcal{G}} \langle \theta, \mu\rangle = \min_{\mu\in\mathcal{L}_\mathcal{G}} \langle \theta_{\mathcal{V}}, \mu_{\mathcal{V}}\rangle + \langle \theta_{\mathcal{E}}, \mu_{\mathcal{E}}\rangle, \tag{2.16}$$

where the so-called *local polytope* $\mathcal{L}_\mathcal{G}$ is the set of all vectors μ of the form (2.14) with components ranging over the sets specified by (2.15). The adjective "local" refers to the local marginalization constraints (2.15b).

3. Image labeling on the assignment manifold. This section sets the stage for our approach to solving approximately the labeling problem (1.2). We first introduce in section 3.1 in terms of the assignment manifold the setting for the smooth approach to image labeling [5], to be sketched in section 3.2. Section 3.3 summarizes the general framework of [42] for numerically integrating Riemannian gradient flows of functionals defined on the assignment manifold.

3.1. The assignment manifold. The relative interior of the probability simplex $\mathcal{S} := \operatorname{rint}(\Delta_n)$, given by $\mathcal{S} = \{p \in \mathbb{R}^n_{++} \colon \langle \mathbb{1}, p\rangle = 1\}$, is an $n-1$ dimensional smooth manifold with constant tangent space

$$T_p\mathcal{S} = \{v \in \mathbb{R}^n \colon \langle \mathbb{1}, v\rangle = 0\} =: T \subset \mathbb{R}^n, \quad p \in \mathcal{S}. \tag{3.1}$$

Due to $\langle \mathbb{1}, v\rangle = 0$ for all $v \in T$, we have the orthogonal decomposition $\mathbb{R}^n = T \oplus \mathbb{R}\mathbb{1}$. The orthogonal projection onto T is given by

$$P_T \colon \mathbb{R}^n \to T\,, \quad x \mapsto P_T(x) = x - \frac{1}{n}\langle \mathbb{1}, x\rangle \mathbb{1} = \left(I - \frac{1}{n}\mathbb{1}\mathbb{1}^\top\right) x, \tag{3.2}$$

where I denotes the $(n\times n)$ identity matrix. The manifold $\mathcal{S}$ becomes a Riemannian manifold by endowing it with the Fisher–Rao metric. At a point $p \in \mathcal{S}$, this metric is given by

$$\langle \cdot, \cdot\rangle_p \colon T_p\mathcal{S} \times T_p\mathcal{S} \to \mathbb{R}\,, \quad (u, v) \mapsto \langle u, v\rangle_p = \left\langle \frac{u}{\sqrt{p}}, \frac{v}{\sqrt{p}}\right\rangle. \tag{3.3}$$

In this setting, there is an important map, called the *lifting map* (cf. [5, Def. 4]), defined as

$$\tilde{L}_p \colon \mathbb{R}^n \to \mathcal{S}, \qquad x \mapsto \tilde{L}_p(x) := \frac{p \cdot e^x}{\langle p, e^x\rangle}, \quad p \in \mathcal{S}. \tag{3.4}$$

By restricting $\tilde{L}_p$ onto the tangent space, we obtain a diffeomorphism

$$L_p := \tilde{L}_p|_T \colon T \to \mathcal{S}, \qquad \tilde{L}_p = L_p \circ P_T. \tag{3.5}$$

This restricted lifting map L_p is also a local first-order approximation to the exponential map of the Riemannian manifold $\mathcal{S}$ (cf. [5, Prop. 3]), with the inverse mapping given by

$$L_p^{-1} \colon \mathcal{S} \to T\,, \quad q \mapsto L_p^{-1}(q) := P_T\Big(\log \frac{q}{p}\Big). \tag{3.6}$$

The *assignment manifold* is defined as the product manifold $\mathcal{W} := \prod_{i\in[m]} \mathcal{S}$ and can be identified with the space $\mathcal{W} = \{W \in \mathbb{R}_{++}^{m\times n} \colon W\mathbb{1} = \mathbb{1}\}$ of row-stochastic matrices with full support. With the Riemannian product metric, $\mathcal{W}$ also becomes a Riemannian manifold with constant tangent space

$$T_W\mathcal{W} = \prod_{i\in[m]} T = \{V \in \mathbb{R}^{m\times n} \colon V\mathbb{1} = 0\} =: T^m, \quad W \in \mathcal{W}. \tag{3.7}$$

The Fisher–Rao product metric reads

$$\langle U, V\rangle_W = \sum_{i\in[m]} \Big\langle \frac{U_i}{\sqrt{W_i}}, \frac{V_i}{\sqrt{W_i}} \Big\rangle, \quad W \in \mathcal{W}, \quad U, V \in T^m. \tag{3.8}$$

The orthogonal decomposition of T induces the orthogonal decomposition

$$\mathbb{R}^{m\times n} = T^m \oplus \{\lambda\mathbb{1}_n^\top \in \mathbb{R}^{m\times n} \colon \lambda \in \mathbb{R}^m\}, \tag{3.9}$$

together with the orthogonal projection

$$P_{T^m} \colon \mathbb{R}^{m\times n} \to T^m, \qquad X \mapsto P_{T^m}(X) = X\left(I - \frac{1}{n}\mathbb{1}\mathbb{1}^\top\right). \tag{3.10}$$

Thus, the projection of a matrix X onto T^m is just the projection (3.2) applied to every row of X. The lifting map, and the restricted lifting map and its inverse, are naturally extended to

$$\tilde{L}_W \colon \mathbb{R}^{m\times n} \to \mathcal{W}, \qquad L_W \colon T^m \to \mathcal{W}, \qquad \text{and} \qquad L_W^{-1} \colon \mathcal{W} \to T^m \tag{3.11}$$

for every $W \in \mathcal{W}$ by applying $\tilde{L}_{W_i} \colon \mathbb{R}^n \to \mathcal{S}$, $L_{W_i} \colon T \to \mathcal{S}$, and $L_{W_i}^{-1} \colon \mathcal{S} \to T$ from (3.4), (3.5), (3.6) to every row i,

$$\big(\tilde{L}_W(X)\big)_i := \tilde{L}_{W_i}(X_i), \qquad \big(L_W(V)\big)_i := L_{W_i}(V_i), \qquad \text{and} \qquad \big(L_W^{-1}(Q)\big)_i := L_{W_i}^{-1}(Q_i), \tag{3.12}$$

for $i \in [m]$, $X \in \mathbb{R}^{m\times n}$, $V \in T^m$, and $Q \in \mathcal{W}$.

3.2. Image labeling on $\mathcal{W}$. In [5] the following approach was proposed. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a graph with vertex set $\mathcal{V} = [m]$. Suppose a function is given on this graph with values in some feature space $\mathcal{F}$,

$$f\colon \mathcal{V} = [m] \to \mathcal{F}, \qquad i \mapsto f_i. \tag{3.13}$$

Furthermore, let the set $\mathcal{X} = \{\ell_1, \dots, \ell_n\}$ from (1.1) denote a set of prototypes or labels (possibly $\mathcal{X} \subset \mathcal{F}$), and assume a distance function is specified,

$$d\colon \mathcal{F} \times \mathcal{X} \to \mathbb{R}, \tag{3.14}$$

measuring how well a feature is represented by a certain prototype. We are interested in the assignment of the prototypes to the data in terms of an *assignment matrix* $W \in \mathcal{W} \subset \mathbb{R}^{m\times n}$. The elements of W can be interpreted as the *posterior probability*

$$W_{i,j} = \Pr(\ell_j | f_i), \quad i \in [m], \quad j \in [n], \tag{3.15}$$

that ℓ_j generated the observation f_i. The assignment task of determining an optimal assignment W^* can thus be interpreted as finding an "explanation" of the data in terms of the prototypes $\mathcal{X}$.

Remark 3.1 (W vs. μ). Each row vector W_i, $i \in [m]$, plays the role of a corresponding vector μ_i of the basic LP relaxation as defined by (2.13), with relaxed domain due to (2.15). Unlike μ_i, however, vectors $W_i \in \mathbb{R}^n_{++}$ always have full support and live on the manifold $\mathcal{S}$.

The objective function for measuring the quality of an assignment involves three matrices, defined next. First, all distance information between observed feature vectors and prototypes (labels) is gathered by the *distance matrix*

$$D \in \mathbb{R}^{m\times n}, \qquad D_{i,j} = d(f_i, \ell_j) \tag{3.16}$$

and then lifted onto the assignment manifold at $W \in \mathcal{W}$. By using (3.11) we obtain the *likelihood matrix*

$$L = \tilde{L}_W\Big(-\frac{1}{\rho}D\Big) = L_W\Big(-\frac{1}{\rho}P_{T^m}(D)\Big), \quad \rho > 0, \tag{3.17}$$

where each row i of L is given by $L_i = \tilde{L}_{W_i}(-\frac{1}{\rho}D_i)$ and P_{T^m} is given by (3.10). Finally, the *similarity matrix*

$$S = S(W) \in \mathcal{W} \tag{3.18}$$

is defined as a local geometric average of assignment vectors at neighboring nodes; i.e., the ith row S_i is defined to be the Riemannian mean (cf. [5, Def. 2] in the present context and [24] for the general definition)

$$S_i = \operatorname{mean}_{\mathcal{S}}\{L_j\}_{j\in\overline{\mathcal{N}}(i)} \tag{3.19}$$

of the lifted distances L_j in the neighborhood $\overline{\mathcal{N}}(i) = \mathcal{N}(i) \cup \{i\}$.

The correlation between W and the local averages defining $S(W)$, as measured by the basic matrix inner product, is used as the objective function

$$\sup_{W\in\mathcal{W}} J(W), \qquad J(W) := \langle W, S(W)\rangle \tag{3.20}$$

to be maximized. The optimization strategy is to follow the Riemannian gradient ascent flow on $\mathcal{W}$ (see section 3.3 for the formal definition of the Riemannian gradient)

$$\dot{W}(t) = \nabla_{\mathcal{W}} J(W(t)), \qquad W(0) = \frac{1}{n}\mathbb{1}_m\mathbb{1}_n^\top =: C. \tag{3.21}$$

The initialization $W_i(0) = \frac{1}{n}\mathbb{1}_n^\top$ with the barycenter of $\mathcal{S}$ constitutes an *uninformative* uniform assignment, which is not biased towards any prototype.

To obtain an efficient numerical algorithm, the Riemannian mean is approximated using the geometric mean

$$S_i(W) = \frac{\mathrm{mean}_g\{L_j\}_{j\in\overline{\mathcal{N}}(i)}}{\left\langle \mathbb{1}, \mathrm{mean}_g\{L_j\}_{j\in\overline{\mathcal{N}}(i)}\right\rangle}, \quad \mathrm{mean}_g\{L_j\}_{j\in\overline{\mathcal{N}}(i)} = \left(\prod_{j\in\overline{\mathcal{N}}(i)} L_j\right)^{\frac{1}{|\overline{\mathcal{N}}(i)|}}. \tag{3.22}$$

Based on the simplifying, plausible assumption that the mean only changes slowly, and by using the explicit Euler-method directly on $\mathcal{W}$ with a certain adaptive step-size (cf. [5, section 3.3]), the following multiplicative update scheme is obtained:

$$W_i^{(k+1)} = \frac{W_i^{(k)}\cdot S_i(W^{(k)})}{\langle W_i^{(k)}, S_i(W^{(k)})\rangle}, \quad W_i^{(0)} = \frac{1}{n}\mathbb{1}_n^\top, \quad i\in[m], k\in\mathbb{N}. \tag{3.23}$$

3.3. Geometric integration of gradient flows. In this section we collect the basic ingredients needed in the remainder of this paper of a general framework due to [42] for integrating a Riemannian gradient flow of an arbitrary function $J\colon \mathcal{W}\to\mathbb{R}$ defined on the assignment manifold.

We first recall the definition of the Riemannian gradient. Let M be a Riemannian manifold with an inner product g_x^M on each tangent space T_xM varying smoothly with $x\in M$ and $f\colon M\to\mathbb{R}$ a smooth function. Using the identification $T_r\mathbb{R}=\mathbb{R}$ for $r\in\mathbb{R}$, the Riemannian gradient $\nabla_M f(x)\in T_xM$ of f at $x\in M$ can be defined as the unique element of T_xM satisfying

$$g_x^M(\nabla_M f(x), v) = Df(x)[v] \quad \forall v\in T_xM, \tag{3.24}$$

where $Df(x): T_xM\to T_{f(x)}\mathbb{R}=\mathbb{R}$ is the differential of f.

Suppose $J\colon\mathcal{W}\to\mathbb{R}$ is a general smooth objective function modeling an assignment problem and we are interested in minimizing J by following the Riemannian gradient descent flow

$$\dot{W}(t) = -\nabla_{\mathcal{W}} J(W(t)), \quad W(0) = C\in\mathcal{W}, \tag{3.25}$$

with the barycenter $C = \frac{1}{n}\mathbb{1}_m\mathbb{1}_n^\top$. Instead of directly minimizing J on $\mathcal{W}$, the basic idea of [42] is to pull the optimization problem back onto the tangent space $T^m = T_C\mathcal{W}$ by setting

$$\overline{J} := J \circ L_C, \tag{3.26}$$

using the diffeomorphism $L_C\colon T^m \to \mathcal{W}$ given by (3.11). Furthermore, the pullback of the Fisher–Rao metric under L_C is used to equip T^m with a Riemannian metric and to turn L_C into an isometry. In this setting, the Riemannian gradient of $\overline{J}\colon T^m \to \mathbb{R}$ at $V \in T^m$ is given by [42, section 3]

$$\nabla_{T^m}\overline{J}(V) = \nabla J\big(L_C(V)\big) \in T^m\,, \tag{3.27}$$

where ∇J denotes the standard Euclidean gradient of $J\colon \mathcal{W} \to \mathbb{R}$. Based on this construction, solving the gradient flow (3.25) is equivalent to

$$W(t) = L_C(V(t)), \tag{3.28}$$

where $V(t) \in T^m$ solves

$$\dot{V}(t) = -\nabla_{T^m}\overline{J}(V(t)) = -\nabla J\big(W(t)\big)\,, \quad V(0) = 0\,. \tag{3.29}$$

Choosing the explicit Euler-method for solving this gradient flow problem on the vector space T^m results in the numerical update scheme for every row $i \in [m]$,

$$V_i^{(k+1)} = V_i^{(k)} - h\nabla J\big(L_C(V_i^{(k)})\big), \quad V_i^{(0)} = 0, \quad k \in \mathbb{N}, \tag{3.30}$$

with step-size $h \in \mathbb{R}$. Lifting this update scheme to the assignment manifold $\mathcal{W}$ yields a multiplicative update rule

$$W_i^{(k+1)} = \frac{W_i^{(k)} \cdot e^{-h\nabla J(W_i^{(k)})}}{\big\langle W_i^{(k)}, e^{-h\nabla J(W_i^{(k)})}\big\rangle}\,, \quad W_i^{(0)} = \frac{1}{n}\mathbb{1}_n, \quad i \in [m], k \in \mathbb{N}. \tag{3.31}$$

4. Energy, gradients, and Wasserstein messages.

In this section we study the smooth objective function (1.4) *restricted* to the assignment manifold, in order to prepare the application of the approach of section 3 to graphical models in section 5.

After detailing the rationale behind (1.4) in section 4.1, we compute the Euclidean gradient of the objective function in section 4.2 on which the Riemannian gradient will be based. This gradient involves the gradients of local Wasserstein distances that are considered in section 4.3. From the viewpoint of BP, these gradients can be considered as "Wasserstein messages" as discussed in section 5.

4.1. Smooth approximation of the LP relaxation.

The starting point (3.16) for applying the labeling approach of section 3.2 to a given problem is a definition of suitable distances. Regarding problem (1.2) and the corresponding model parameter vector θ defined by (2.12), this is straightforward for the *unary* terms θ_i that typically measure a local distance to observed data. But this is less obvious for the *pairwise* terms θ_{ij} that do not have a direct counterpart in the geometric labeling approach.

The following lemma explains why the local Wasserstein distances

$$d_{\theta_{ij}}(\mu_i, \mu_j) := \min_{\mu_{ij} \in \Pi(\mu_i, \mu_j)} \langle \theta_{ij}, \mu_{ij} \rangle \ , \tag{4.1}$$

defined for every edge $ij \in \mathcal{E}$ with $\Pi(\mu_i, \mu_j)$ due to (2.15b), are natural candidates for taking into account pairwise model parameters θ_{ij}.

Lemma 4.1. *The local polytope relaxation* (2.16) *is equivalent to the problem*

$$\min_{\mu_{\mathcal{V}} \in \Delta_n^m} \left(\sum_{i \in \mathcal{V}} \langle \theta_i, \mu_i \rangle + \sum_{ij \in \mathcal{E}} d_{\theta_{ij}}(\mu_i, \mu_j) \right) \tag{4.2}$$

involving the local Wasserstein distances (4.1).

Proof. The claim follows from reformulating the LP relaxation based on the local polytope constraints (2.15) as follows:

$$\begin{aligned}
\min_{\mu \in \mathcal{L}_{\mathcal{G}}} \langle \theta, \mu \rangle &= \min_{\mu \in \mathcal{L}_{\mathcal{G}}} \langle \theta_{\mathcal{V}}, \mu_{\mathcal{V}} \rangle + \langle \theta_{\mathcal{E}}, \mu_{\mathcal{E}} \rangle \\
&= \min_{\mu_{\mathcal{V}} \in \Delta_n^m} \left(\langle \theta_{\mathcal{V}}, \mu_{\mathcal{V}} \rangle + \min_{\mu_{\mathcal{E}}} \sum_{ij \in \mathcal{E}} \left(\langle \theta_{ij}, \mu_{ij} \rangle + \delta_{\Pi(\mu_i, \mu_j)}(\mu_{ij}) \right) \right) \\
&= \min_{\mu_{\mathcal{V}} \in \Delta_n^m} \left(\sum_{i \in \mathcal{V}} \langle \theta_i, \mu_i \rangle + \sum_{ij \in \mathcal{E}} \min_{\mu_{ij} \in \Pi(\mu_i, \mu_j)} \langle \theta_{ij}, \mu_{ij} \rangle \right) \\
&= \min_{\mu_{\mathcal{V}} \in \Delta_n^m} \left(\sum_{i \in \mathcal{V}} \langle \theta_i, \mu_i \rangle + \sum_{ij \in \mathcal{E}} d_{\theta_{ij}}(\mu_i, \mu_j) \right).
\end{aligned}$$

■

In order to conform to our smooth geometric setting, we regularize the convex but nonsmooth (piecewise-linear (cf. [40, Def. 2.47])) local Wasserstein distances (4.1) with a general convex *smoothing function* F_τ,

$$d_{\theta_{ij},\tau}(\mu_i, \mu_j) = \min_{\mu_{ij} \in \Pi(\mu_i, \mu_j)} \big\{ \langle \theta_{ij}, \mu_{ij} \rangle + F_\tau(\mu_{ij}) \big\}, \quad ij \in \mathcal{E}, \quad F_\tau \in \mathcal{F}_0, \quad \tau > 0, \tag{4.3}$$

with smoothing parameter τ.

Remark 4.2 (role of the smoothing). The influence of the smoothing parameter τ will be examined in detail in the remainder of this paper. We wish to point out from the beginning, however, that the ability of our smooth geometric approach to compute *integral* labeling assignments does *not* necessarily imply values of $\tau \approx 0$ close to zero, because the rounding mechanism to integral assignments is a *different one*, as will be shown in section 5. As a consequence, larger feasible values of τ weaken the nonlinear relation (4.3) and considerably speed up the convergence of the numerical algorithm for iterative label assignment.

Remark 4.3 (local polytope constraints). Using the regularized local Wasserstein distances (4.3) implies by their definition that the local marginalization constraints (2.15) are *always* satisfied. This is in sharp contrast to alternative labeling schemes, such as loopy BP, where

these constraints are gradually enforced during the iteration and are guaranteed to hold only *after* convergence of the entire iteration process.

This elucidates the following two key properties that distinguish the manifold setting of our labeling approach from established work:

1. inherent smoothness, and
2. anytime validity of the local polytope constraints.

Based on Lemma 4.1 and the regularized local Wasserstein distances (4.3), we study the objective function (1.4), which is a *smooth* approximation of the local polytope relaxation (2.16) of the original labeling problem (1.2), with the local polytope constraints (2.15) *built in*.

In order to get an intuition about suitable smoothing functions F_τ, we inspect the smoothed local Wasserstein distance (4.3) in more detail. To this end, it will be convenient to simplify temporarily our notation in the remainder of this section by dropping indices as follows:

$$\text{(4.4a)} \qquad \text{For any edge } ij: \quad M = \mu_{ij} \in \mathbb{R}^{n\times n}, \qquad \Theta = \theta_{ij} \in \mathbb{R}^{n\times n},$$

$$\text{(4.4b)} \qquad \mu = \begin{pmatrix}\mu_1\\ \mu_2\end{pmatrix} = \begin{pmatrix}M\mathbb{1}_n\\ M^\top \mathbb{1}_n\end{pmatrix}, \qquad \nu = \begin{pmatrix}\nu_1\\ \nu_2\end{pmatrix},$$

with the marginal vector μ playing the role of $\left(\begin{smallmatrix}\mu_i\\ \mu_j\end{smallmatrix}\right)$ in (2.15). The local (nonsmooth) Wasserstein distance (4.1) then reads, for any edge $ij \in \mathcal{E}$, as

$$d_\Theta(\mu_1, \mu_2) = \min_{M\in\Pi(\mu_1,\mu_2)} \langle \Theta, M\rangle \,. \tag{4.5}$$

Using the linear map $\mathcal{A}$ defined by (2.3a), we rewrite expression (4.5) as

$$d_\Theta(\mu_1, \mu_2) = \min_{M\in\mathbb{R}^{n\times n}} \langle \Theta, M\rangle \quad \text{s.t.} \quad \mathcal{A}M = \begin{pmatrix}\mu_1\\ \mu_2\end{pmatrix}, \quad M \geq 0 \,. \tag{4.6}$$

The corresponding dual LP of (4.6) is given by

$$\max_{\nu\in\mathbb{R}^{2n}} \langle \mu, \nu\rangle \quad \text{s.t.} \quad \mathcal{A}^\top \nu \leq \Theta \,. \tag{4.7}$$

The *smoothed* local Wasserstein distance (4.3) is given by

$$\begin{aligned} d_{\Theta,\tau}(\mu_1, \mu_2) &:= \min_{M\in\mathbb{R}^{n\times n}} \langle \Theta, M\rangle + F_\tau(M) \quad \text{s.t.} \quad \mathcal{A}M = \begin{pmatrix}\mu_1\\ \mu_2\end{pmatrix}, \quad M \geq 0, \\ &= \min_{M\in\mathbb{R}^{n\times n}} \langle \Theta, M\rangle + F_\tau(M) + \delta_{\mathbb{R}_+^{n\times n}}(M) + \delta_{\{0\}}\big(\mathcal{A}M - \left(\begin{smallmatrix}\mu_1\\ \mu_2\end{smallmatrix}\right)\big), \end{aligned} \tag{4.8}$$

for $F_\tau \in \mathcal{F}_0$ and $\tau > 0$, and the dual problem to (4.8) reads as

$$\max_{\nu\in\mathbb{R}^{2n}} \langle \mu, \nu\rangle - G_\tau^*\big(\mathcal{A}^\top \nu - \Theta\big), \tag{4.9}$$

with the conjugate function G_τ^* of

$$G_\tau(M) = F_\tau(M) + \delta_{\mathbb{R}_+^{n\times n}}(M). \tag{4.10}$$

Suitable candidates of functions G_τ for smoothing d_Θ suggest themselves by comparing the dual LP (4.7) with the dual problem (4.9) of the smoothed LP. Rewriting the constraints of (4.7) in the form

$$\delta_{\mathbb{R}^{n\times n}_-}(\mathcal{A}^\top \nu - \Theta) \tag{4.11}$$

and comparing with (4.9) shows that G_τ^* should be a smooth approximation of the indicator function $\delta_{\mathbb{R}^{n\times n}_-}$. We return to this point in section 6.2.

4.2. Energy gradient ∇E_τ. The pairwise model parameters $\theta_\mathcal{E}$ may not be symmetric, $\theta_{ij} \neq \theta_{ij}^\top$, $ij \in \mathcal{E}$, in general, which implies that the smoothed local Wasserstein distances are not symmetric either: $d_{\theta_{ij},\tau}(W_i, W_j) \neq d_{\theta_{ij},\tau}(W_j, W_i)$. In order to compute the Euclidean gradient ∇E_τ of the objective function (1.4), we therefore introduce an *arbitrary fixed orientation* (i, j) (ordered pair) of all edges $ij \in \mathcal{E}$, which means $ij \in \mathcal{E} \implies ji \notin \mathcal{E}$. As a consequence, (1.4) reads as

$$E_\tau(W) = \sum_{i\in\mathcal{V}} \left(\langle \theta_i, W_i \rangle + \sum_{j\colon (i,j)\in\mathcal{E}} d_{\theta_{ij},\tau}(W_i, W_j) \right) . \tag{4.12}$$

The following proposition specifies the gradient ∇E_τ in terms of an expression that involves local gradients of the smoothed Wasserstein distances $d_{\theta_{ij},\tau}$. These latter gradients are studied in section 4.3 (Theorem 4.7).

Proposition 4.4 (objective function gradient). *Suppose the edges $\mathcal{E}$ have an arbitrary fixed orientation. Then the Euclidean gradient of the objective function $E_\tau\colon \mathcal{W} \to \mathbb{R}$ due to (1.4), at $W \in \mathcal{W}$, is the matrix $\nabla E_\tau(W) \in T^m$, whose ith row is given by*

$$\nabla_i E_\tau(W) = P_T(\theta_i) + \sum_{j\colon (i,j)\in\mathcal{E}} \nabla_1 d_{\theta_{ij},\tau}(W_i, W_j) + \sum_{j\colon (j,i)\in\mathcal{E}} \nabla_2 d_{\theta_{ji},\tau}(W_j, W_i) , \tag{4.13}$$

where $\nabla_1 d_{\theta_{ij},\tau}(W_i, W_j) \in T$ and $\nabla_2 d_{\theta_{ji},\tau}(W_j, W_i) \in T$ are the Euclidean gradients of

$$d_{\theta_{ij},\tau}(\cdot, W_j)\colon \mathcal{S} \to \mathbb{R}, \qquad d_{\theta_{ij},\tau}(W_j, \cdot)\colon \mathcal{S} \to \mathbb{R}. \tag{4.14}$$

Proof. See section A.1 of the appendix. ∎

We now consider, after a preparatory lemma, the specific case when all pairwise model parameters $\theta_{ij} = \theta_{ij}^\top$ are symmetric (Corollary 4.6). Recall definition (2.15b) of the set $\Pi(\cdot,\cdot)$ of coupling measures having its arguments as marginals and Remark 3.1 regarding notation.

Lemma 4.5. *Suppose the convex smoothing function F_τ defining the regularized local Wasserstein distances (4.3) satisfies $F_\tau(M) = F_\tau(M^\top)$ for all $M \in \Pi(W_i, W_j)$. Then*

$$d_{\theta_{ij},\tau}(W_i, W_j) = d_{\theta_{ij}^\top,\tau}(W_j, W_i). \tag{4.15}$$

Proof. Let $M_* \in \Pi(W_i, W_j)$ be a minimizer of (4.8). Then due to the assumption on F_τ, we have

$$d_{\theta_{ij},\tau}(W_i, W_j) = \langle \theta_{ij}, M_* \rangle + F_\tau(M_*) = \langle \theta_{ij}^\top, M_*^\top \rangle + F_\tau(M_*^\top) . \tag{4.16}$$

Let $\tilde{M} \in \Pi(W_j, W_i)$ be arbitrary. Then $\tilde{M}^\top \in \Pi(W_i, W_j)$, and we have

$$\langle \theta_{ij}^\top, \tilde{M} \rangle + F_\tau(\tilde{M}) = \langle \theta_{ij}, \tilde{M}^\top \rangle + F_\tau(\tilde{M}^\top) \geq \langle \theta_{ij}, M_* \rangle + F_\tau(M_*) = \langle \theta_{ij}^\top, M_*^\top \rangle + F_\tau(M_*^\top)\,. \tag{4.17}$$

This shows that $M_*^\top \in \Pi(W_j, W_i)$ is a minimizer of $d_{\theta_{ij}^\top,\tau}(W_j, W_i)$ and establishes (4.15). ∎

As a consequence of Lemma 4.5, if all pairwise model parameters θ_{ij} are symmetric, in addition to $F_\tau(M) = F_\tau(M^\top)$ for all $M \in [0,1]^{n\times n}$, then there is no need to choose an edge orientation as was done in connection with (4.12). Rather, using (2.1), we may rewrite (4.12) as

$$E_\tau(W) = \sum_{i\in\mathcal{V}} \left(\langle \theta_i, W_i \rangle + \frac{1}{2} \sum_{j\in\mathcal{N}(i)} d_{\theta_{ij},\tau}(W_i, W_j) \right) \tag{4.18}$$

and reformulate Proposition 4.4 accordingly.

Corollary 4.6 (objective function gradient: Symmetric case). *Suppose $F_\tau(T) = F_\tau(T^\top)$ for all $T \in [0,1]^{n\times n}$ and that θ_{ij} is symmetric for all $ij \in \mathcal{E}$. Then the ith row of the Euclidean gradient ∇E_τ is given by*

$$\nabla_i E_\tau(W) = P_T(\theta_i) + \sum_{j\in\mathcal{N}(i)} \nabla_1 d_{\theta_{ij},\tau}(W_i, W_j). \tag{4.19}$$

Proof. Applying the equation $\nabla_2 d_{\theta_{ji},\tau}(W_j, W_i) = \nabla_1 d_{\theta_{ij},\tau}(W_i, W_j)$ due to Lemma 4.5 to (4.13), we obtain

$$\nabla_i E_\tau(W) = P_T(\theta_i) + \sum_{j\colon (i,j)\in\mathcal{E}} \nabla_1 d_{\theta_{ij},\tau}(W_i, W_j) + \sum_{j\colon (j,i)\in\mathcal{E}} \nabla_1 d_{\theta_{ij},\tau}(W_i, W_j) \tag{4.20a}$$

$$= P_T(\theta_i) + \sum_{j\in\mathcal{N}(i)} \nabla_1 d_{\theta_{ij},\tau}(W_i, W_j), \tag{4.20b}$$

which is (4.19). ∎

4.3. Local Wasserstein distance gradient. In this section, we check differentiability of the distance functions $d_{\theta_{ij},\tau}(\mu_i, \mu_j)$, $ij \in \mathcal{E}$, given by (4.3), and specify an expression for the corresponding gradient. To formulate the main result of this section, we again use the simplified notation (4.4).

Theorem 4.7 (Wasserstein distance gradient). *Consider $\mathcal{S} \subset \mathbb{R}^n$ as a Euclidean submanifold with tangent space T defined by (3.1), and let*

$$g(\mu, \nu) = \langle \mu, \nu \rangle - G_\tau^*(\mathcal{A}^\top \nu - \Theta) \tag{4.21}$$

denote the dual objective function (4.26). Then the smoothed Wasserstein distance $d_{\Theta,\tau}\colon \mathcal{S} \times \mathcal{S} \to \mathbb{R}$ is differentiable, and the Euclidean gradient of $d_{\Theta,\tau}$ at $p = (p_1, p_2) \in \mathcal{S} \times \mathcal{S}$ is given by

$$\nabla d_{\Theta,\tau}(p) = \nabla d_{\Theta,\tau}(p_1, p_2) = \overline{\nu}_T := P_{T\times T}(\overline{\nu}) = \begin{pmatrix} P_T(\overline{\nu}_1) \\ P_T(\overline{\nu}_2) \end{pmatrix}, \tag{4.22}$$

where

$$\overline{\nu} = \begin{pmatrix} \overline{\nu}_1 \\ \overline{\nu}_2 \end{pmatrix} \in \operatorname*{argmax}_{\nu\in\mathbb{R}^{2n}} g(p, \nu). \tag{4.23}$$

The proof follows below after some preparatory lemmas, which also clarify the structure of the dual solution set. In particular, this set restricted to $\mathcal{R}(\mathcal{A})$ is a singleton (Lemma 4.9).

Lemma 4.8. *Let*

$$G_\tau(M) = F_\tau(M) + \delta_{\mathbb{R}_+^{n\times n}}(M), \tag{4.24}$$

with the convex smoothing function F_τ of (4.3), and assume the conjugate function G_τ^* is continuously differentiable. Then the dual problem of

$$\min_{M\in\Pi(\mu_1,\mu_2)} \big\{\langle \Theta, M\rangle + F_\tau(M)\big\} \tag{4.25}$$

is given by

$$\max_{\nu_1,\nu_2} \big\{\langle \mu, \nu\rangle - G_\tau^*(\mathcal{A}^\top \nu - \Theta)\big\}. \tag{4.26}$$

Furthermore, assuming that strong duality holds, the conditions for optimal primal $\overline{M}$ and dual $\overline{\nu} = (\overline{\nu}_1, \overline{\nu}_2)$ solutions are

$$\overline{M} = \nabla G_\tau^*\big(\mathcal{A}^\top \overline{\nu} - \Theta\big), \qquad \mathcal{A}^\top \overline{\nu} - \Theta \in \partial G_\tau(\overline{M}), \tag{4.27a}$$

together with the affine constraint

$$\mathcal{A}\overline{M} = \mu. \tag{4.27b}$$

Proof. Taking into account (2.15b), we write the right-hand side of (4.8) in the form

$$\min_{M\in\mathbb{R}^{n\times n}} \langle \Theta, M\rangle + G_\tau(M) \quad \text{s.t.} \quad \mathcal{A}M = \mu, \quad M \geq 0. \tag{4.28}$$

Let $\nu = (\nu_1, \nu_2) \in \mathbb{R}^{2n}$ denote the dual variables corresponding to the affine constraint of (4.28). Then problem (4.28) rewritten in Lagrangian form reads as

$$\min_{M\in\mathbb{R}^{n\times n}} \big\{\langle \Theta, M\rangle + G_\tau(M) + \max_\nu \langle \nu, \mu - \mathcal{A}M\rangle\big\} \tag{4.29a}$$

$$\Leftrightarrow \min_{M\in\mathbb{R}^{n\times n}} \big\{ \max_\nu \langle \nu, \mu\rangle + G_\tau(M) - \big\langle \mathcal{A}^\top \nu - \Theta, M\big\rangle\big\}. \tag{4.29b}$$

Since strong duality holds by assumption, interchanging min and max yields the dual problem (4.26). Moreover, the optimal primal and dual objective function values are equal, which gives, with (4.29a) and (4.26),

$$-\langle \overline{M}, \mathcal{A}^\top \overline{\nu} - \Theta\rangle + G_\tau(\overline{M}) + G_\tau^*(\mathcal{A}^\top \overline{\nu} - \Theta) = 0. \tag{4.30}$$

This implies (4.27a) by the subgradient inversion rule [40, Prop. 11.3], whereas the primal constraint (4.27b) is obvious. ∎

Remark 4.9 (smoothness of G_τ^*). The *smoothness* assumption with respect to G_τ^* enables us to compute conveniently the gradient of the smoothed Wasserstein distance $d_{\Theta,\tau}$. It corresponds to a *convexity* assumption on G_τ. These aspects are further discussed in section 6.2 as well.

Remark 4.10 (strong duality). The condition of strong duality (cf. [10, Section I.5]) made by Lemma 4.8 is crucial for what follows. This condition will be satisfied later on when working in a *geometric* setting with local measures M, μ_1, μ_2 with *full* support, as introduced in section 3.1.

Lemma 4.11. *Let the linear mapping $\mathcal{A}^\top$ be defined by (2.3b). Then*

(4.31)
$$\mathcal{N}(\mathcal{A}^\top) = \left\{ \lambda \begin{pmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{pmatrix} \in \mathbb{R}^{2n} \colon \lambda \in \mathbb{R} \right\} \quad \text{and} \quad \mathcal{N}(\mathcal{A}^\top)^\perp = \left\{ x \in \mathbb{R}^{2n} \colon \left\langle x, \begin{pmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{pmatrix} \right\rangle = 0 \right\}.$$

Proof. Let $z = \left(\begin{smallmatrix} x \\ y \end{smallmatrix}\right) \in \mathbb{R}^{2n}$ with $0 = \mathcal{A}^\top z = x\mathbb{1}_n^\top + \mathbb{1}_n y^\top$. Applying $\mathcal{A}$, we get

(4.32)
$$0 = \mathcal{A}\mathcal{A}^\top z = \mathcal{A}(x\mathbb{1}_n^\top) + \mathcal{A}(\mathbb{1}_n y^\top) = \begin{pmatrix} nx + \langle y, \mathbb{1}_n \rangle \mathbb{1}_n \\ \langle x, \mathbb{1}_n \rangle \mathbb{1}_n + ny \end{pmatrix} \quad \Leftrightarrow \quad z = \begin{pmatrix} x \\ y \end{pmatrix} = -\frac{1}{n} \begin{pmatrix} \langle y, \mathbb{1}_n \rangle \mathbb{1}_n \\ \langle x, \mathbb{1}_n \rangle \mathbb{1}_n \end{pmatrix}.$$

This implies $\langle x, \mathbb{1}_n \rangle = -\langle y, \mathbb{1}_n \rangle$, and setting $\lambda = \frac{1}{n}\langle x, \mathbb{1}_n \rangle \in \mathbb{R}$ shows that z has the form (4.31). Conversely, in view of the definition (2.3b), it is clear that any vector from the set (4.31) is in $\mathcal{N}(\mathcal{A}^\top)$. The characterization of $\mathcal{N}(\mathcal{A}^\top)^\perp$ directly follows from the definitions. ∎

The following lemma characterizes the set of optimal dual solutions to problem (4.26).

Lemma 4.12. *Let the function G_τ^* of the dual objective function (4.26), respectively, (4.21), be continuously differentiable and strictly convex, and let $p \in \mathbb{R}^{2n}_{++}$. Then the set of optimal dual solutions has the form*

$$\operatorname*{argmax}_{\nu \in \mathbb{R}^{2n}} g(p, \nu) = \begin{cases} \{\overline{\nu}\} & \text{if } \left\langle p, \left(\begin{smallmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{smallmatrix}\right) \right\rangle \neq 0, \\ \overline{\nu} + \mathcal{N}(\mathcal{A}^\top) & \text{if } \left\langle p, \left(\begin{smallmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{smallmatrix}\right) \right\rangle = 0. \end{cases} \tag{4.33}$$

Proof. See section A.2 in the appendix. ∎

We next clarify the *attainment* of optimal dual solutions due to Lemma 4.12.

Lemma 4.13. *Consider the orthogonal decomposition $\mathbb{R}^{2n} = \mathcal{N}(\mathcal{A}^\top) \oplus \mathcal{R}(\mathcal{A})$ into linear subspaces, and denote the corresponding components of a vector $\nu \in \mathbb{R}^{2n}$ by $\nu = \nu_\mathcal{N} + \nu_\mathcal{R}$. Then, for $p \in \mathbb{R}^{2n}_{++}$ satisfying $\langle p, \left(\begin{smallmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{smallmatrix}\right) \rangle = 0$, we have*

$$\operatorname*{argmax}_{\nu_\mathcal{R} \in \mathcal{R}(\mathcal{A})} g(p, \nu_\mathcal{R}) = \{\overline{\nu}_\mathcal{R}\}, \qquad \overline{\nu}_\mathcal{R} = P_{\mathcal{R}(\mathcal{A})}(\overline{\nu}) \quad \text{for any} \quad \overline{\nu} \in \operatorname*{argmax}_{\nu \in \mathbb{R}^{2n}} g(p, \nu), \tag{4.34a}$$

$$g(p, \overline{\nu}_\mathcal{R}) = \max_{\nu_\mathcal{R} \in \mathcal{R}(\mathcal{A})} g(p, \nu_\mathcal{R}) = \max_{\nu \in \mathbb{R}^{2n}} g(p, \nu); \tag{4.34b}$$

that is, a unique dual maximizer exists in the subspace $\mathcal{R}(\mathcal{A})$.

Proof. We first show (4.34b). Let $\overline{\nu}$ be an optimal dual solution. Since $\left\langle p, \left(\begin{smallmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{smallmatrix}\right) \right\rangle = 0$, Lemma 4.12 yields $\operatorname{argmax}_{\nu \in \mathbb{R}^{2n}} g(p, \nu) = \overline{\nu} + \mathcal{N}(\mathcal{A}^\top) = \overline{\nu}_\mathcal{N} + \overline{\nu}_\mathcal{R} + \mathcal{N}(\mathcal{A}^\top)$. This shows $\overline{\nu}_\mathcal{R} \in \overline{\nu} + \mathcal{N}(\mathcal{A}^\top)$, that is, $\overline{\nu}_\mathcal{R} \in \mathcal{R}(\mathcal{A})$ is a maximizer, which implies (4.34b).

Let $\overline{\nu}'_\mathcal{R} \in \mathcal{R}(\mathcal{A})$ be another maximizer. As before, we have the representation $\overline{\nu}'_\mathcal{R} \in \overline{\nu} + \mathcal{N}(\mathcal{A}^\top)$, that is, $\overline{\nu}'_\mathcal{R} = \overline{\nu}_\mathcal{N} + \overline{\nu}_\mathcal{R} + \tilde{\nu}_\mathcal{N}$ for some $\tilde{\nu}_\mathcal{N} \in \mathcal{N}(\mathcal{A}^\top)$, which implies $\overline{\nu}'_\mathcal{R} = \overline{\nu}_\mathcal{R}$, i.e., uniqueness (4.34a) of the dual maximizer in $\mathcal{R}(\mathcal{A})$. ∎

We are now in position to prove Theorem 4.7.

Proof of Theorem 4.7. We proceed by subsequently proving the following: First, we relate the orthogonal decomposition $\mathbb{R}^{2n} = \mathcal{N}(\mathcal{A}^\top) \oplus \mathcal{R}(\mathcal{A})$ to the tangent space $T_p(\mathcal{S}\times\mathcal{S}) = T\times T \subset \mathbb{R}^{2n}$ for any $p = (p_1, p_2) \in \mathcal{S}\times\mathcal{S}$. Second, the existence of a global isometric chart for the manifold $\mathcal{S}\times\mathcal{S}$ is shown in order to represent the smoothed Wasserstein distance $d_{\Theta,\tau}$ and the dual objective function $g(\mu,\nu)$ in a convenient way. Third, we apply Theorem 2.2.

1. Consider the unique decomposition $\nu = \nu_{\mathcal{N}} + \nu_{\mathcal{R}} \in \mathcal{N}(\mathcal{A}^\top) \oplus \mathcal{R}(\mathcal{A})$ of any point $\nu \in \mathbb{R}^{2n}$. Then we have

$$P_{T\times T}(\nu_{\mathcal{R}}) = \nu_T = P_{T\times T}(\nu). \tag{4.35}$$

At first, we show $T\times T \subseteq \mathcal{R}(\mathcal{A})$. For this, take an arbitrary $v = \left(\begin{smallmatrix} v_1 \\ v_2 \end{smallmatrix}\right) \in T\times T$. Due to the definition of T, we have $\langle \mathbb{1}_n, v_1\rangle = \langle \mathbb{1}_n, v_2\rangle = 0$ and thus $\langle v, \left(\begin{smallmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{smallmatrix}\right)\rangle = 0$, which according to Lemma 4.11 means $v \in \mathcal{N}(\mathcal{A}^\top)^\perp = \mathcal{R}(\mathcal{A})$. As a consequence of $T\times T \subseteq \mathcal{R}(\mathcal{A})$ we have $P_{T\times T}(\nu_{\mathcal{N}}) = 0$, and therefore statement (4.35) follows from

$$P_{T\times T}(\nu) - P_{T\times T}(\nu_{\mathcal{R}}) = P_{T\times T}(\nu - \nu_{\mathcal{R}}) = P_{T\times T}(\nu_{\mathcal{N}}) = 0. \tag{4.36}$$

2. There exist an open subset $U \subset \mathbb{R}^{2(n-1)}$ and an isometry $\phi\colon U \to \mathcal{S}\times\mathcal{S}$ such that ϕ^{-1} is a global isometric chart of the manifold $\mathcal{S}\times\mathcal{S}$. ϕ can be constructed as follows. Choose an orthonormal basis $\{v_1, \dots, v_{2(n-1)}\}$ of the tangent space $T\times T$, set $b = \frac{1}{n}\left(\begin{smallmatrix} \mathbb{1}_n \\ \mathbb{1}_n \end{smallmatrix}\right)$, and define the isometry

$$\psi\colon \mathbb{R}^{2(n-1)} \to \big(T\times T\big) + b, \quad x \mapsto \psi(x) := Bx + b, \quad Bx = \sum_{i=1}^{2(n-1)} x_i v_i. \tag{4.37}$$

Because $\mathcal{S}\times\mathcal{S}$ is an open subset of $\big(T\times T\big) + b$ and ψ an isometry, we have that the set $U := \psi^{-1}(\mathcal{S}\times\mathcal{S}) \subset \mathbb{R}^{2(n-1)}$ is also open and

$$\phi := \psi|_U\colon U \to \mathcal{S}\times\mathcal{S} \tag{4.38}$$

is the desired isometric mapping. Furthermore, since the basis $\{v_i\}_{i=1}^{2(n-1)}$ is orthonormal, the orthogonal projection reads as

$$P_{T\times T} = BB^\top. \tag{4.39}$$

3. Using ϕ given by (4.38), we obtain the coordinate representations

$$\overline{d}_{\Theta,\tau} := d_{\Theta,\tau} \circ \phi, \qquad \overline{g}(x,\nu) := g\big(\phi(x),\nu\big) \tag{4.40}$$

of the smoothed Wasserstein distance $d_{\Theta,\tau}$ and the dual objective function $g(p,\nu)$. Since we assume strong duality, that is, equality of the optimal values of (4.25) and (4.26), we have $d_{\Theta,\tau}(p) = \max_{\nu\in\mathbb{R}^{2n}} g(p,\nu)$. Setting $x_p = \phi^{-1}(p)$, this equation translates, in view of Lemma 4.13, into

$$\overline{g}(x_p, \overline{\nu}_{\mathcal{R}}) = \max_{\nu_{\mathcal{R}}\in\mathcal{R}(\mathcal{A})} \overline{g}(x_p,\nu_{\mathcal{R}}) = \overline{g}(x_p,\overline{\nu}) = \max_{\nu\in\mathbb{R}^{2n}} \overline{g}(x_p,\nu) = \overline{d}_{\Theta,\tau}(x_p), \tag{4.41}$$

with unique maximizer $\overline{\nu}_{\mathcal{R}} = P_{\mathcal{R}(\mathcal{A})}(\overline{\nu})$. Let $\mathbb{B}_\delta \subset \mathcal{R}(\mathcal{A})$ be a compact neighborhood of $\overline{\nu}_{\mathcal{R}}$. Then (4.41) remains valid after restricting $\mathcal{R}(\mathcal{A})$ to $\mathbb{B}_\delta$. Because g given by (4.21) is linear in the first argument and the mapping ϕ is affine, the function $\overline{g}$ is convex in the first argument and differentiable, and hence satisfies the assumptions of Theorem 2.2.

In order to compute the gradient $\nabla_x \overline{g}(x, \nu_{\mathcal{R}})$, it suffices to consider the first term $\langle \phi(x), \nu_{\mathcal{R}} \rangle$ of $\overline{g}$, which depends only on x. Using (4.38), we have

$$\langle \phi(x), \nu_{\mathcal{R}} \rangle = \langle Bx + b, \nu_{\mathcal{R}} \rangle = \langle x, B^\top \nu_{\mathcal{R}} \rangle + \langle b, \nu_{\mathcal{R}} \rangle. \tag{4.42}$$

Thus, $\nabla_x \overline{g}(x, \nu_{\mathcal{R}}) = B^\top \nu_{\mathcal{R}}$, which continuously depends on $\nu_{\mathcal{R}}$. As a consequence, we may apply Theorem 2.2 and obtain, due to (2.10),

$$\nabla \overline{d}_{\Theta,\tau}(x_p) = \nabla_x \overline{g}(x_p, \overline{\nu}_{\mathcal{R}}) = B^\top \overline{\nu}_{\mathcal{R}}. \tag{4.43}$$

Using the differential $D\phi(x) = B$, we finally get

$$\nabla d_{\Theta,\tau}(p) = B \nabla \overline{d}_{\Theta,\tau}(x_p) = B B^\top \overline{\nu}_{\mathcal{R}} \overset{(4.35)}{=} P_{T\times T}(\overline{\nu}_{\mathcal{R}}) \overset{(4.35)}{=} \overline{\nu}_T, \tag{4.44}$$

which proves (4.22). ∎

5. Application to graphical models. This section explains how the labeling approach on the assignment manifold of section 3 can be applied to a graphical model by using the global and local gradients derived in section 4. The graphical model is given in terms of an energy function $E(x)$ of the form (1.2). The basic idea for determining a labeling x with low energy $E(x)$, worked out in section 5.1, is to combine minimization of the convex relaxation (1.3) and nonconvex rounding to an integral solution in a *single smooth process*. This idea is realized by restricting the smooth approximation (1.4) of the objective function to the assignment manifold from section 3.1, and by combining numerical integration of the corresponding Riemannian gradient flow from section 3.3 with the assignment mechanism suggested by [5] from section 3.2.

Section 5.2 complements our preliminary observations stated as Remarks 4.2 and 4.3 and highlights the essential properties of smooth process as a novel way of BP using dually computed gradients of local Wasserstein distances, which we call *Wasserstein messages*.

5.1. Smooth integration of minimizing and rounding on the assignment manifold. We recall how regularization is performed by the assignment approach of [5]: distance vectors (3.16) representing the data term of classical variational approaches are lifted to the assignment manifold by (3.17) and geometrically averaged over spatial neighborhoods; see (3.19) and (3.22).

Given a graphical model in terms of an energy function (1.2), regularization is already *defined* by the pairwise model parameters $E_{ij}(\ell_k, \ell_r)$, respectively, $\theta_{ij}(\ell_k, \ell_r)$, so that evaluating the gradient of the regularized objective function (1.4) *implies* averaging over spatial neighborhoods, as (4.13) clearly displays. Additionally, taking into account the simplest (explicit Euler) update rule (3.31) for geometric integration of Riemannian gradient flows on the assignment manifold, we find that a natural definition of the similarity matrix for the kth

iterate, $k \in \mathbb{N}$, that consistently incorporates the graphical model into the geometric approach of [5], is

$$S_i(W^{(k)}) = \frac{W_i^{(k)} \cdot e^{-h\nabla_i E_\tau(W^{(k)})}}{\langle W_i^{(k)}, e^{-h\nabla_i E_\tau(W^{(k)})}\rangle}, \quad i \in [m], \qquad h > 0, \qquad W^{(0)} = \frac{1}{n}\mathbb{1}_m\mathbb{1}_n^\top, \tag{5.1}$$

where h is a step-size parameter and the partial gradients $\nabla_i E_\tau(W^{(k)})$ are given by (4.13). The sequence $(W^{(k)})$ is initialized in an unbiased way at the barycenter $W^{(0)} \in \mathcal{W}$. Adopting the fixed point iteration proposed by [5] leads to the update of the assignment matrix

$$W_i^{(k+1)} = \frac{W_i^{(k)} \cdot S_i(W^{(k)})}{\langle W_i^{(k)}, S_i(W^{(k)})\rangle}, \quad i \in [m]. \tag{5.2}$$

These two interleaved update steps represent two objectives: (i) minimize the function E_τ on the assignment manifold $\mathcal{W}$ (section 3.3), and (ii) converge to an integral solution, i.e., a valid labeling. Plugging (5.1) into (5.2) gives

$$W_i^{(k+1)} = \frac{(W_i^{(k)})^2 \cdot e^{-h\nabla_i E_\tau(W^{(k)})}}{\langle (W_i^{(k)})^2, e^{-h\nabla_i E_\tau(W^{(k)})}\rangle}, \tag{5.3}$$

which suggests that the latter rounding mechanism can be more flexibly controlled by a *rounding parameter* α and the update rule

$$W_i^{(k+1)} = \frac{(W_i^{(k)})^{1+\alpha} \cdot e^{-h\nabla_i E_\tau(W^{(k)})}}{\langle (W_i^{(k)})^{1+\alpha}, e^{-h\nabla_i E_\tau(W^{(k)})}\rangle}, \quad \alpha \geq 0. \tag{5.4}$$

The following proposition reveals the *continuous* gradient flow that is approximated by the sequence (5.4).

Proposition 5.1. *Let E_τ be given by (1.4), and denote the entropy of the assignment matrix W by*

$$H(W) = -\langle W, \log W\rangle. \tag{5.5}$$

Then the sequence of updates (5.4) are geometric Euler-steps for numerically integrating the Riemannian gradient flow of the extended objective function

$$f_{\tau,\alpha}(W) := E_\tau(W) + \alpha_h H(W), \qquad \alpha_h = \frac{\alpha}{h}. \tag{5.6}$$

Proof. An Euler-step for minimizing $f_{\tau,\alpha}$ on the tangent space reads (with $\nabla_i = \nabla_{W_i}$) as

$$V_i^{(k+1)} = V_i^{(k)} - h\nabla_i f(W^{(k)}) = V_i^{(k)} - h\nabla_i E_\tau(W^{(k)}) - \alpha\nabla_i H(W^{(k)}), \qquad i \in [m], \tag{5.7}$$

where the ith row of $W^{(k)}$ is given by $W_i^{(k)} = L_c(V_i^{(k)})$, $c = \frac{1}{n}\mathbb{1}_n$. In order to compute the gradient of the entropy, consider a smooth curve $\gamma\colon (-\varepsilon, \varepsilon) \to \mathcal{W}$ with $\gamma(0) = W$ and $\dot\gamma(0) = X$. Then

$$\frac{d}{dt}H(\gamma(t))\big|_{t=0} = -\langle X, \log(W)\rangle - \left\langle W, \frac{1}{W}\cdot X\right\rangle = -\langle X, \log(W)\rangle - \langle \mathbb{1}\mathbb{1}^\top, X\rangle. \tag{5.8}$$

Since $\langle \log(W), X\rangle = \langle P_{T^m}(\log(W)), X\rangle$ and $\langle \mathbb{1}\mathbb{1}^\top, X\rangle = \langle \mathbb{1}, X\mathbb{1}\rangle = \langle \mathbb{1}, 0\rangle = 0$, we have

$$\langle \nabla H(W), X\rangle = \frac{d}{dt} H(\gamma(t))\big|_{t=0} = \langle -P_{T^m}(\log(W)), X\rangle. \tag{5.9}$$

Thus, using $P_T(\log(W_i)) = L_c^{-1}(W_i)$ from (3.6), we obtain

$$\nabla_i H(W^{(k)}) = -P_T\left(\log(W_i^{(k)})\right) = -L_c^{-1}\big(L_c(V_i^{(k)})\big) = -V_i^{(k)}. \tag{5.10}$$

Substitution into (5.7) gives

$$V_i^{(k+1)} = (1+\alpha)V_i^{(k)} - h\nabla_i E_\tau(W^{(k)}) \tag{5.11}$$

and in turn the update

$$W_i^{(k+1)} = L_c(V_i^{(k+1)}) = \frac{e^{(1+\alpha)V_i^{(k)}} \cdot e^{-h\nabla_i E_\tau(W^{(k)})}}{\langle \mathbb{1}_n, e^{(1+\alpha)V_i^{(k)}} \cdot e^{-h\nabla_i E_\tau(W^{(k)})}\rangle} \tag{5.12a}$$

$$= \frac{(e^{V_i^{(k)}})^{(1+\alpha)} \cdot e^{-h\nabla_i E_\tau(W^{(k)})}}{\langle \mathbb{1}_n, (e^{V_i^{(k)}})^{1+\alpha} \cdot e^{-h\nabla_i E_\tau(W^{(k)})}\rangle} = \frac{(W_i^{(k)})^{(1+\alpha)} \cdot e^{-h\nabla_i E_\tau(W^{(k)})}}{\langle \mathbb{1}_n, (W_i^{(k)})^{1+\alpha} \cdot e^{-h\nabla_i E_\tau(W^{(k)})}\rangle} \tag{5.12b}$$

$$= \frac{(W_i^{(k)})^{(1+\alpha)} \cdot e^{-h\nabla_i E_\tau(W^{(k)})}}{\langle (W_i^{(k)})^{1+\alpha}, e^{-h\nabla_i E_\tau(W^{(k)})}\rangle}, \tag{5.12c}$$

which is (5.4). ∎

Remark 5.2 (continuous DC programming). Proposition 5.1 and (5.6) interpret the update rule (5.4) as a *continuous difference of convex (DC) programming* strategy. The established DC approach [35, 36] takes *large steps* by solving to optimality a sequence of convex programs in connection with updating an affine upper bound of the concave part of the objective function, but our update rule (5.4) differs from this approach in two essential ways: *geometric optimization* by numerically integrating the Riemannian gradient flow *tightly interleaves with rounding* to an integral solution. The rounding effect is achieved by minimizing the entropy term of (5.6), which steadily sparsifies the assignment vectors comprising W.

5.2. Wasserstein messages. We return to the informal discussion of *belief propagation (BP)* from section 1.2 in order to highlight properties of our approach (1.4) from this viewpoint. We first sketch BP and the origin of corresponding *messages*, and we refer the reader to [54, 50] for background and more details.

Our starting point is the primal linear program (LP) (1.3) written in the form

$$\min_{\mu \in \mathcal{L}_\mathcal{G}} \langle \theta, \mu\rangle = \min_{\mu} \langle \theta, \mu\rangle \quad \text{s.t.} \quad A\mu = b,\ \mu \geq 0, \tag{5.13}$$

where the constraints represent the feasible set $\mathcal{L}_\mathcal{G}$, which is explicitly given by the local marginalization constraints (2.15). The corresponding dual LP reads as

$$\max_{\nu} \langle b, \nu\rangle = \max_{\nu} \langle \mathbb{1}, \nu_\mathcal{V}\rangle, \quad A^\top \nu \leq \theta, \tag{5.14}$$

with dual (multiplier) variables

$$\nu = (\nu_{\mathcal{V}}, \nu_{\mathcal{E}}) = (\dots, \nu_i, \dots, \nu_{ij}(x_i), \dots, \nu_{ij}(x_j), \dots), \quad i \in \mathcal{V}, \quad ij \in \mathcal{E}, \tag{5.15}$$

corresponding to the affine primal constraints. In order to obtain a condition that relates optimal vectors μ and ν without subdifferentials that are caused by the nonsmoothness of these LPs, one should consider the *smoothed* primal convex problem

$$\min_{\mu \in \mathcal{L}_{\mathcal{G}}} \langle \theta, \mu \rangle - \varepsilon H(\mu), \quad \varepsilon > 0, \qquad H(\mu) = \sum_{ij \in \mathcal{E}} H(\mu_{ij}) - \sum_{i \in \mathcal{V}} \big(d(i) - 1\big) H(\mu_i), \tag{5.16}$$

with smoothing parameter $\varepsilon > 0$, degree $d(i)$ of vertex i, and local entropy functions

$$H(\mu_i) = -\sum_{x_i \in \mathcal{X}} \mu_i(x_i) \log \mu_i(x_i), \qquad H(\mu_{ij}) = -\sum_{x_i, x_j \in \mathcal{X}} \mu_{ij}(x_i, x_j) \log \mu_{ij}(x_i, x_j). \tag{5.17}$$

Setting temporarily $\varepsilon = 1$ and evaluating the optimality condition $\nabla_\mu L(\mu, \nu) = 0$ based on the corresponding Lagrangian

$$L(\mu, \nu) = \langle \theta, \mu \rangle - H(\mu) + \langle \nu, A\nu - b \rangle \tag{5.18}$$

yields the relations connecting μ and ν,

$$\mu_i(x_i) = e^{\nu_i} e^{-\theta_i(x_i)} \prod_{j \in \mathcal{N}(i)} e^{\nu_{ij}(x_i)}, \qquad x_i \in \mathcal{X},\ i \in \mathcal{V}, \tag{5.19a}$$

$$\mu_{ij}(x_i, x_j) = e^{\nu_i + \nu_j} e^{-\theta_{ij}(x_i, x_j) - \theta_i(x_i) - \theta_j(x_j)} \prod_{k \in \mathcal{N}(i) \setminus \{j\}} e^{\nu_{ik}(x_i)} \prod_{k \in \mathcal{N}(j) \setminus \{i\}} e^{\nu_{jk}(x_j)}, \tag{5.19b}$$

with $x_i, x_j \in \mathcal{X}$, $ij \in \mathcal{E}$, where the terms $e^{\nu_i}, e^{\nu_i + \nu_j}$ normalize the expressions on the right-hand side, whereas the so-called *messages* $e^{\nu_{ij}(x_i)}$ enforce the local marginalization constraints $\mu_{ij} \in \Pi(\mu_i, \mu_j)$. Invoking these latter constraints enables us to eliminate the left-hand side of (5.19) to obtain, after some algebra, the fixed point equations

$$e^{\nu_{ij}(x_i)} = e^{\nu_j} \sum_{x_j \in \mathcal{X}} \Big(e^{-\theta_{ij}(x_i, x_j) - \theta_j(x_j)} \prod_{k \in \mathcal{N}(j) \setminus \{i\}} e^{\nu_{ik}(x_j)} \Big), \qquad ij \in \mathcal{E}, \quad x_i \in \mathcal{X}, \tag{5.20}$$

solely in terms of the *dual* variables, commonly called the *sum-product algorithm* or *loopy BP* by *message passing*. Repeating this derivation, after weighting the entropy function $H(\mu)$ of (5.18) by ε as in (5.16), and taking the limit $\lim_{\varepsilon \searrow 0}$ yields relation (5.20), with the sum replaced by the max operation as a consequence of taking the log of both sides and relation (2.8). This fixed point iteration is called the *max-product algorithm* in the literature.

From this viewpoint, our alternative approach (5.6) emerges as follows, starting at the

smoothed primal LP (5.16) and following the idea of the proof of Lemma 4.1:

$$\min_{\mu\in\mathcal{L}_\mathcal{G}} \langle\theta,\mu\rangle - \varepsilon H(\mu) \tag{5.21a}$$

$$= \min_{\mu\in\mathcal{L}_\mathcal{G}} \langle\theta,\mu\rangle - \varepsilon\Bigg(\sum_{ij\in\mathcal{E}} H(\mu_{ij}) - \sum_{i\in\mathcal{V}} \big(d(i)-1\big)H(\mu_i)\Bigg) \tag{5.21b}$$

$$= \min_{\mu\in\mathcal{L}_\mathcal{G}} \langle\theta_\mathcal{V},\mu_\mathcal{V}\rangle + \langle\theta_\mathcal{E},\mu_\mathcal{E}\rangle - \varepsilon\sum_{ij\in\mathcal{E}} H(\mu_{ij}) + \varepsilon\sum_{i\in\mathcal{V}} \big(d(i)-1\big)H(\mu_i) \tag{5.21c}$$

$$= \min_{\mu_\mathcal{V}\in\Delta_n^m} E_\varepsilon(\mu_\mathcal{V}) + \varepsilon\sum_{i\in\mathcal{V}} \big(d(i)-1\big)H(\mu_i). \tag{5.21d}$$

Formulation (5.6) results from replacing ε by a smoothing parameter τ, which can be set to a value *not very* close to 0 (cf. Remark 4.2), and we absorb the second nonnegative factor, weighting the entropy term by a second parameter α. As demonstrated in section 7, this latter parameter enables us to control precisely the trade-off between accuracy of labelings in terms of the given objective function E_τ of (5.6), which approximates the original discrete objective function (1.2), and the speed of convergence to an integral (labeling) solution.

Regarding the resulting term E_τ, a key additional step is to use the reformulation (1.4), because all edge-based variables are *locally* "dualized away" as done *globally* with *all* variables when using established BP (cf. (5.20)). In this way, we can work in the primal domain, and with graphs having higher connectivity, without suffering from the enormous memory requirements that would arise from merely smoothing the LP and solving (5.16) in the primal domain. Furthermore, the "messages" defined by our approach have a clear interpretation in terms of the smoothed Wasserstein distance between local marginal measures.

We summarize this discussion by contrasting directly established BP with our approach in terms of the following key observations:

1. **Local nonconvexity.** The negative $-H(\mu)$ of the so-called *Bethe entropy* function $H(\mu)$ is *nonconvex* in general for graphs $\mathcal{G}$ with cycles [50, section 4.1] due to the negative sign of the second sum of (5.16).
2. **Local rounding at each step.** The max-product algorithm performs *local rounding* at *every* step of the iteration so as to obtain integral solutions, i.e., a *labeling* after convergence. This operation results as a limit of a *nonconvex* function, due to observation 1.
3. **Either nonsmoothness or strong nonlinearity.** The latter max-operation is inherently nonsmooth. Preferring instead a smooth approximation with $0 < \varepsilon \ll 1$ necessitates choosing ε very small so as to ensure rounding. This, however, leads to *strongly nonlinear* functions of the form (2.8) that are difficult to handle numerically.
4. **Invalid constraints.** Local marginalization constraints are only satisfied *after* convergence of the iteration. Intuitively it is plausible that, by only *gradually* enforcing constraints in this way, the iterative process becomes more susceptible to getting stuck in unfavorable stationary points, due to the nonconvexity according to observation 1.

Our *geometric approach* removes each of these issues. *Message passing* with respect to vertex $i \in \mathcal{V}$ is defined by evaluating the local Wasserstein gradients of (4.13) for all edges incident

to i. We therefore call these local gradients *Wasserstein messages,* which are "passed along edges." Similarly to (5.20), each such message is given by *dual* variables through (4.22) that solve the regularized *local* dual LPs (4.21). As a consequence, local marginalization constraints are *always* satisfied throughout the iterative process.

In addition, we make the following observations in correspondence to points 1–4 above:

5. **Local convexity.** Wasserstein messages of (4.13) are defined by local *convex* programs (4.21). This contrasts with loopy BP and holds true for any pairwise model parameters θ_{ij} of the prior of the graphical model and the corresponding coupling of μ_i and μ_j. This removes spurious minima introduced through nonconvex entropy approximations.
6. **Smooth global rounding after convergence.** Rounding to integral solutions is *gradually* enforced through the Riemannian flow induced by the extended objective function (5.6). In particular, repeated "aggressive" local max operations of the max-product algorithm are replaced by a *smooth* flow.
7. **Smoothness and weak nonlinearity.** The role of the smoothing parameter τ of (1.4) *differs* from the role of the smoothing parameter ε of (5.16). While the latter has to be chosen quite close to 0 so as to achieve rounding at all, τ merely mollifies the dual local problems (4.21), and hence should be chosen small, but may be considerably larger than ε. In particular, this does not impair rounding due to observation 6, which happens due to the *global* flow which is *smoothly* driven by the Wasserstein messages. This *decoupling* of smoothing and rounding enables us to numerically compute labelings more efficiently. The results reported in section 7 demonstrate this fact.
8. **Valid constraints.** By construction, computation of the Wasserstein messages enforces all local marginalization constraints *throughout* the iteration. This is in sharp contrast to BP, where this generally holds after convergence only. Intuitively, it is plausible that our *more tightly* constrained iterative process is less susceptible to getting stuck in poor local minima. The results reported in section 7.2 provide evidence of this conjecture.

6. Implementation. In this section we discuss several aspects of the implementation of our approach. The numerical update scheme used in our implementation is the one given by (5.4),

$$W_i^{(k+1)} = \frac{(W_i^{(k)})^{1+\alpha} \cdot e^{-h\nabla_i E_\tau(W^{(k)})}}{\langle (W_i^{(k)})^{1+\alpha}, e^{-h\nabla_i E_\tau(W^{(k)})} \rangle}, \quad W_i^{(0)} = \frac{1}{n}\mathbb{1}_n, \quad i \in \mathcal{V}, k \in \mathbb{N},$$

where $\alpha \geq 0$ is the *rounding* parameter, $h > 0$ the step-size, and τ the *smoothing* parameter for the local Wasserstein distances.

Section 6.1 details a strategy for maintaining in a *numerically stable* way strict positivity of all variables defined on the assignment manifold. Numerical aspects of computing local Wasserstein gradients are discussed in section 6.2, and the natural role of the entropy function is highlighted for assuming the role of the smoothing function F_τ in (4.3). Our criterion for convergence and terminating the iterative process (5.4) of label assignment is specified in section 6.3.

6.1. Assignment normalization. The rounding mechanism addressed by Proposition 5.1 and Remark 5.2 will be effective if α_h in (5.6) is chosen large enough to compensate for the influence of the function F_τ that regularizes the local Wasserstein distances (4.3).

In this case, each vector W_i approaches some vertex e_i of the simplex, and thus some entries of W_i converge to zero. However, due to our optimization scheme, every vector W_i evolves on the interior of the simplex $\mathcal{S}$; that is, all entries of W_i have to be positive all the time—see also Remark 4.10. Since there is a limit to the precision of representing small positive numbers on a computer, we avoid numerical problems by adopting the normalization strategy of [5]. After each iteration, we check all W_i, and whenever an entry drops below $\varepsilon = 10^{-10}$, we rectify W_i by

$$W_i \leftarrow \frac{1}{\langle \mathbb{1}, \tilde{W}_i \rangle} \tilde{W}_i \,, \quad \tilde{W}_i = W_i - \min_{j=1,\dots,n} \{W_{i,j}\} + \varepsilon \,, \quad \varepsilon = 10^{-10} \,. \tag{6.1}$$

Thus, the constant ε plays the role of 0 in our implementation. Our numerical experiments showed that this operation avoids numerical issues.

6.2. Computing Wasserstein gradients. A core subroutine of our approach concerns the computation of the local Wasserstein gradients as part of the overall gradient (4.13). We argue in this section why the *negative entropy function* that we use in our implementation for smoothing the local Wasserstein distances plays a distinguished role. To this end, we adopt again in this section the notation (4.4).

Using this notation, the *smooth* entropy-regularized Wasserstein distance (4.3) reads as

$$d_{\Theta,\tau}(\mu_1, \mu_2) = \min_{M \in \mathbb{R}^{n \times n}} \langle \Theta, M \rangle - \tau H(M) \quad \text{s.t.} \quad \mathcal{A}M = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \quad M \geq 0 \,, \tag{6.2}$$

with the entropy function

$$H(M) = -\sum_{i,j} M_{i,j} \log M_{i,j}. \tag{6.3}$$

As shown in section 4.3 and according to Theorem 4.7, the gradients of (6.2) are the maximizer of the corresponding dual problem. Using the notation (4.4), the dual problem of (6.2) reads as

$$\max_{\nu \in \mathbb{R}^{2n}} \langle \mu, \nu \rangle - \tau \sum_{k,l} \exp\left[\frac{1}{\tau}\Big(\mathcal{A}^\top \nu - \Theta\Big)_{k,l}\right]. \tag{6.4}$$

In particular, in view of the general form (4.9) of this dual problem, the indicator function (4.11) is smoothly approximated by the function $\tau \exp(\frac{1}{\tau}x)$. Figure 6.1 compares this approximation with the classical logarithmic barrier $-\log(-x)$ function for approximating the indicator function $\delta_{\mathbb{R}_-}$ of the nonpositive orthant. Log-barrier penalty functions are the method of choice for *interior point methods* [31, 48], which *strictly* rule out violations of the constraints. While this is essential for many applications where constraints represent physical properties that cannot be violated, it is *not* essential in the present case for calculating the

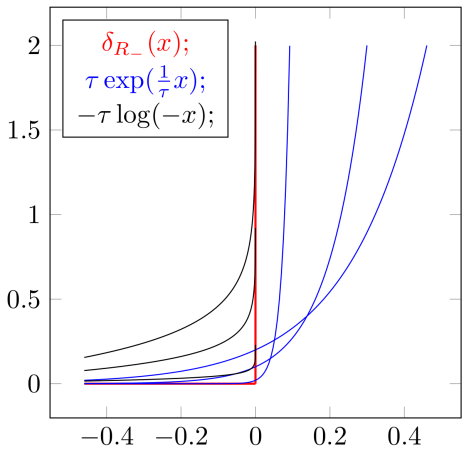


Figure 6.1. *Approximations of the indicator function $\delta_{\mathbb{R}_-}$ of the nonpositive orthant. The log-barrier function (black curves) strictly rules out violations of the constraints but induces a bias towards interior points. Our formulation (blue curves) is less biased and reasonably approximates the δ-function (red curve) depending on the smoothing parameter τ. Displayed are the approximations of $\delta_{\mathbb{R}_-}$ for $\tau = \frac{1}{5}, \frac{1}{10}, \frac{1}{50}$.*

Wasserstein messages. Moreover, the bias towards interior points by log-barrier functions, as Figure 6.1 clearly shows, is detrimental in the present context and favors the formulation (6.4).

We now derive how the local Wasserstein gradients (4.22) are computed based on the formulation (6.2) and examine numerical aspects depending on the smoothing parameter τ. It is well known that doubly stochastic matrices as solutions of convex programs like (6.2) can be computed by iterative matrix scaling [46, 45], [13, Chap. 9]. This has been made popular in the field of machine learning by [16].

The optimality condition (4.27) takes the form

$$\overline{M} = \exp\Big[\frac{1}{\tau}\Big(\mathcal{A}^\top \overline{\nu} - \Theta\Big)\Big], \tag{6.5}$$

and rearranging yields the connection to matrix scaling:

$$\begin{aligned}\overline{M} = \exp\Big[\frac{1}{\tau}\Big(\mathcal{A}^\top \overline{\nu} - \Theta\Big)\Big] &\overset{(2.3b)}{=} \exp\Big[\frac{1}{\tau}\Big(\overline{\nu}_1 \mathbb{1}_n^\top + \mathbb{1}_n \overline{\nu}_2^\top - \Theta\Big)\Big] \\ = \quad \Big(\exp(\tfrac{\overline{\nu}_1}{\tau})\exp(\tfrac{\overline{\nu}_2}{\tau})^T\Big) \cdot \exp\big(-\tfrac{1}{\tau}\Theta\big) \quad &= \quad \mathrm{Diag}\big(\exp(\tfrac{\overline{\nu}_1}{\tau})\big) \exp\big(-\tfrac{1}{\tau}\Theta\big) \mathrm{Diag}\big(\exp(\tfrac{\overline{\nu}_2}{\tau})\big),\end{aligned} \tag{6.6}$$

where $\mathrm{Diag}(\cdot)$ denotes the diagonal matrix with the argument vector as entries. For given marginals $\mu = (\mu_1, \mu_2)$ due to (6.2) and with the shorthand $K = \exp\big(-\frac{1}{\tau}\Theta\big)$, the optimal dual variables $\overline{\nu} = (\overline{\nu}_1, \overline{\nu}_2)$ can be determined by Sinkhorn's iterative algorithm [46], up to a common multiplicative constant. Specifically, we have the following lemma.

Lemma 6.1 (see [16, Lemma 2]). *For $\tau > 0$, the solution $\overline{M}$ of (6.2) is unique and has the form $\overline{M} = \mathrm{diag}(v_1) K \mathrm{diag}(v_2)$, where the two vectors $v_1, v_2 \in \mathbb{R}^n$ are uniquely defined up to a multiplicative factor.*

Accordingly, by setting

$$v_1 := \exp(\tfrac{\nu_1}{\tau}), \qquad v_2 := \exp(\tfrac{\nu_2}{\tau}), \tag{6.7}$$

the corresponding fixed point iterations read

$$v_1^{(k+1)} = \frac{\mu_1}{K\left(\frac{\mu_2}{K^\top v_1^{(k)}}\right)}, \qquad v_2^{(k+1)} = \frac{\mu_2}{K^\top\left(\frac{\mu_1}{K v_2^{(k)}}\right)}, \tag{6.8}$$

which are iterated until the change between consecutive iterates is small enough. Denoting the iterates after convergence by $\overline{v}_1, \overline{v}_2$, resubstitution into (6.7) determines the optimal dual variables

$$\overline{\nu}_1 = \tau \log \overline{v}_1, \qquad \overline{\nu}_2 = \tau \log \overline{v}_2. \tag{6.9}$$

Due to Theorem 4.7, the local Wasserstein gradients then finally are given by

$$\nabla d_{\Theta,\tau}(\mu_1, \mu_2) = \begin{pmatrix} P_T(\overline{\nu}_1) \\ P_T(\overline{\nu}_2) \end{pmatrix}, \tag{6.10}$$

where the projection P_T due to (3.2) removes the common multiplicative constant resulting from Sinkhorn's algorithm.

While the linear convergence rate of Sinkhorn's algorithm is known theoretically [26], the numbers of iterations required in practice significantly depends on the smoothing parameter τ. In addition, for smaller values of τ, an entry of the matrix $K = \exp\big(-\frac{1}{\tau}\Theta\big)$ might be too small to be represented on a computer, due to machine precision. As a consequence, the matrix K might have entries which are numerically treated as zeros, and Sinkhorn's algorithm does not necessarily converge to the true optimal solution.

Fortunately, our approach does allow larger values of τ because merely a sufficiently accurate approximation of the *gradient* of the Wasserstein distance is required, rather than an approximation of the Wasserstein distance itself, to obtain valid *descent* directions. Figures 6.2 and 6.3 demonstrate that this indeed holds for relatively large values of τ, e.g., $\tau \in \{\frac{1}{5}, \frac{1}{10}, \frac{1}{15}\}$, no matter if the number of labels is $n = 10$ or $n = 1000$.

6.3. Termination criterion. In all experiments, the normalized averaged entropy

$$\frac{1}{m\log(n)} H(W) = -\frac{1}{m\log(n)} \sum_{i\in\mathcal{V}} \sum_{k=1}^{n} W_{i,k} \log\big(W_{i,k}\big) \quad \text{for} \quad W \in \mathcal{W}, \tag{6.11}$$

was used as a termination criterion; i.e., if the value drops below a certain threshold, the algorithm is terminated. Due to this normalization, the value does not depend on the number of labels, and thus the threshold is comparable across different models with a varying number of pixels and labels.

For example, a threshold of 10^{-4} means in practice that, up to a small fraction of nodes $i \in \mathcal{V}$, all rows W_i of the assignment matrix W are very close to unit vectors and thus indicate an almost unique assignment of the prototypes or labels to the observed data.

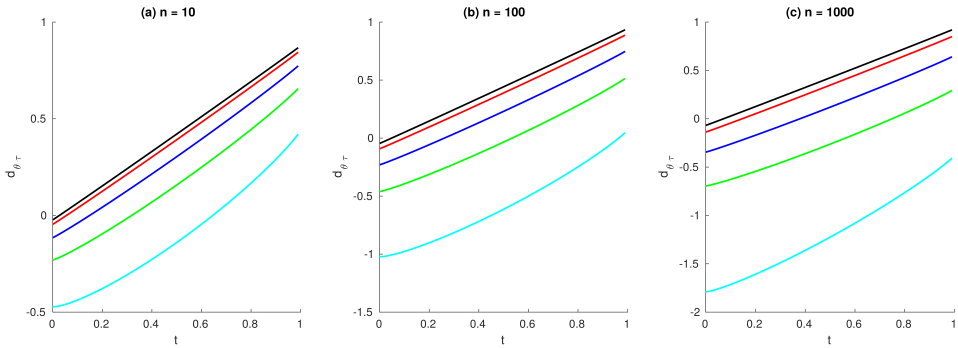


Figure 6.2. *The plots show the entropy-regularized Wasserstein distance* $d_{\Theta,\tau}(c, \gamma(t))$ *for varying parameter* τ *and increasing numbers* n *of labels. Here,* $\gamma(t) = t(e_1 - c) + c \in \Delta_n$*, with* $t \in [0, 1]$*, is the line segment connecting the barycenter* $c = \frac{1}{n}\mathbb{1}$ *to the vertex* e_1 *on the simplex* Δ_n*. The cost matrix* Θ *is given by the Potts prior (7.2). In all three plots the parameter* τ *has been chosen as* $\tau = \frac{1}{5}$ *(cyan),* $\tau = \frac{1}{10}$ *(green),* $\tau = \frac{1}{20}$ *(blue),* $\tau = \frac{1}{50}$ *(red), and* $\tau = \frac{1}{100}$ *(black). Even though the values of the approximation of the distance itself differ considerably, the slope of the distance is already approximated quite well for larger values of* τ*, uniformly for small up to large numbers* n *of labels.*

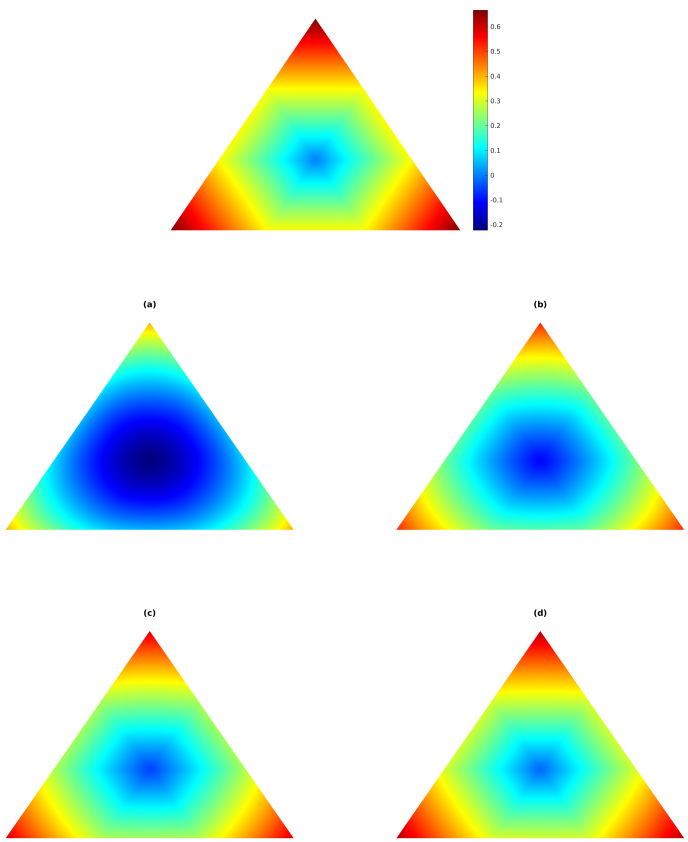


Figure 6.3. *The plot shows the exact Wasserstein distance (top) compared to the entropy-regularized Wasserstein distance with the Potts prior (7.2) from the barycenter to every point on* Δ_3 *for different values of* τ*:* (a) $\tau = \frac{1}{5}$, (b) $\tau = \frac{1}{10}$, (c) $\tau = \frac{1}{20,}$ *and* (d) $\tau = \frac{1}{50}$*. These plots confirm that even for relatively large values of* τ*, e.g..* $\frac{1}{10}$ *and* $\frac{1}{20}$*, the gradient of the Wasserstein distance is a sufficiently accurate approximation to obtain valid descent directions for distance minimization.*

7. Experiments. We demonstrate in this section the main properties of our approach. The dependency of label assignment on the smoothing parameter τ and the rounding parameter α is illustrated in section 7.1. We comprehensively explored the space of binary graphical models defined on the minimal cyclic graph: the complete graph with three vertices $\mathcal{K}^3$ whose LP relaxation is known to have a substantial part consisting of nonbinary vertices. The results reported in section 7.2 exhibit a relationship between α and τ so that in fact a single effective parameter only controls the trade-off between accuracy of optimization and the computational costs. A competitive evaluation of our approach in section 7.3, together with the two established and widely applied approaches of sequential TRWS [27] and loopy BP, reveals a similar performance to our approach. Finally, section 7.4 demonstrates for a graphical model with pronounced *non*uniform pairwise model parameters (non-Potts prior) that our geometric approach accurately takes them into account.

All experiments have been selected to illustrate properties of our approach rather than to demonstrate and work out a particular application, which will be the subject of follow-up work.

7.1. Parameter influence. We assessed the parameter influence of our geometric approach by applying it to a labeling problem. The task is to label a noisy RGB image $f\colon \mathcal{V} \to [0,1]^3$, depicted in Figure 7.2, on the grid graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with minimal neighborhood size $|\mathcal{N}(i)| = 3 \times 3$, $i \in \mathcal{V}$. Prototypical colors $\mathcal{P} = \{l_1, \dots, l_8\} \subset [0,1]^3$ (Figure 7.2) were used as labels. The unary (or data term) is defined, using the $\|\cdot\|_1$ distance and a scaling factor $\rho > 0$, by

$$\theta_i = \frac{1}{\rho}\big(\|f(i) - l_1\|_1, \dots, \|f(i) - l_8\|_1\big), \quad i \in \mathcal{V}, \tag{7.1}$$

and Potts regularization is used for defining the pairwise parameters of the model

$$\big(\theta_{ij}\big)_{k,r} = 1 - \delta_{k,r}, \quad \text{where} \quad \delta_{k,r} = \begin{cases} 1 & \text{if } k = r \\ 0 & \text{else} \end{cases} \quad \text{for} \quad ij \in \mathcal{E}. \tag{7.2}$$

The feature scaling factor was set to $\rho = 0.3$, the step-size $h = 0.1$ was used for numerically integrating the Riemannian descent flow, and the threshold for the normalized average entropy termination criterion (6.11) was set to 10^{-4}.

Figure 7.1, top, displays the empirical convergence rate depending on the rounding parameter α, for a fixed value of the smoothing parameter $\tau = 0.1$ that ensures a sufficiently accurate approximation of the Wasserstein distance gradients and hence of the Riemannian descent flow. Figure 7.1, bottom, shows the interplay between minimizing the smoothed energy E_τ (1.4) and the rounding mechanism induced by the entropy H (5.5) in $f_{\tau,\alpha}$ (5.6). Less aggressive rounding in terms of smaller values of α leads to a more accurate numerical integration of the flow using a larger number of iterations, and thus to higher quality label assignments with a lower energy of the objective function. This latter aspect is demonstrated quantitatively in section 7.2. For too small values of the rounding parameter α, the algorithm naturally does not converge to an integral solution.

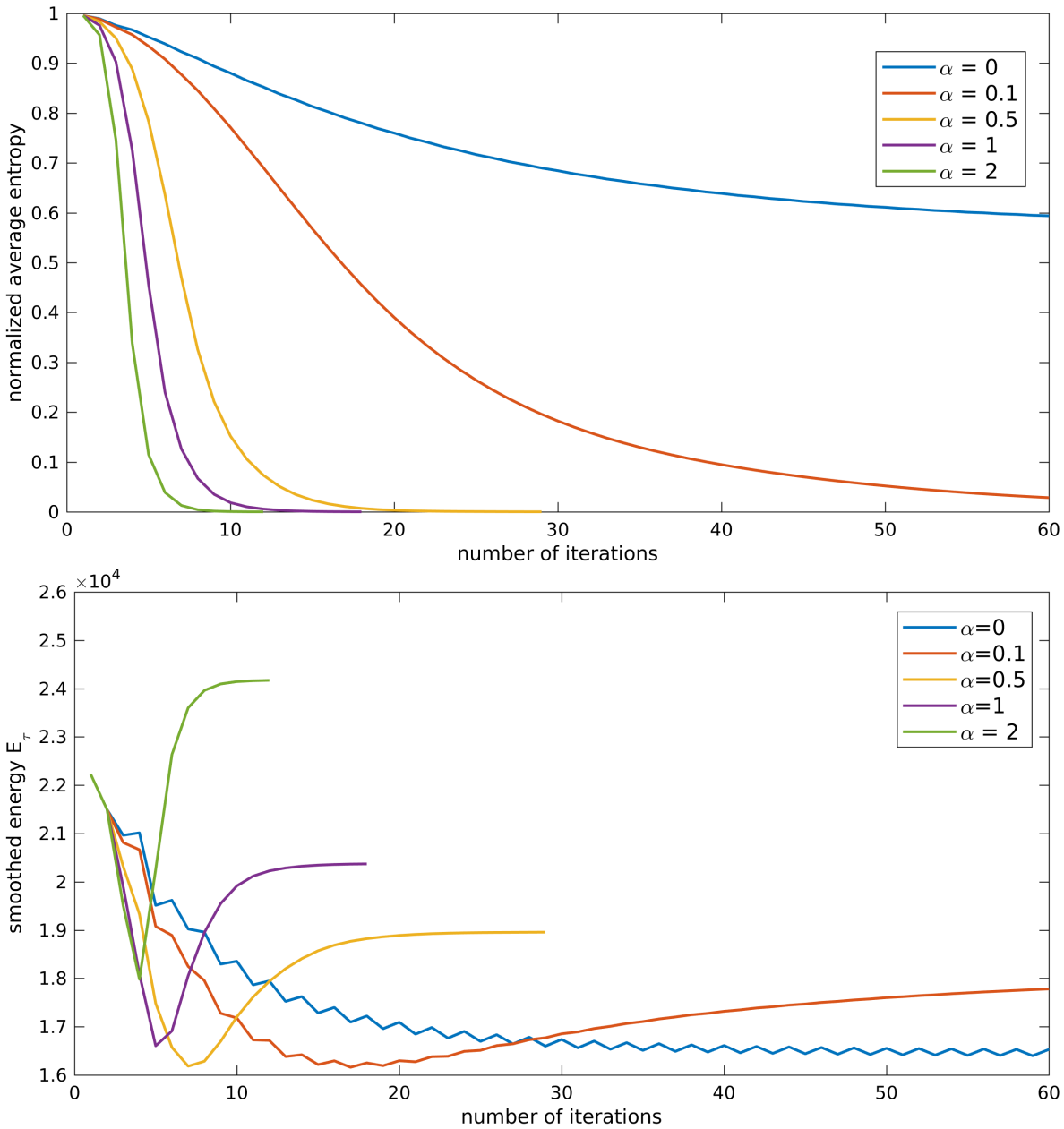


Figure 7.1. *The normalized average entropy* (6.11) *(top) and the smoothed energy* E_τ (1.4) *(bottom) are shown, for the smoothing parameter value* $\tau = 0.1$*, depending on the number of iterations. Top: With increasing values of the rounding parameter* α*, the entropy drops more rapidly and hence converges faster to an integral labeling. Bottom: Two phases of the algorithm depending on the values for* α *are clearly visible. In the first phase, the smoothed energy* E_τ *is minimized up to the point where rounding takes over in the second phase. Accordingly, the sequence of energy values first drops down to lower values corresponding to the problem relaxation and then adopts a higher energy level corresponding to an integral solution. For smaller values of the rounding parameter* α*, the algorithm spends more time on minimizing the smoothed energy. This generally results in lower energy values even after rounding, i.e., in higher quality labelings.*

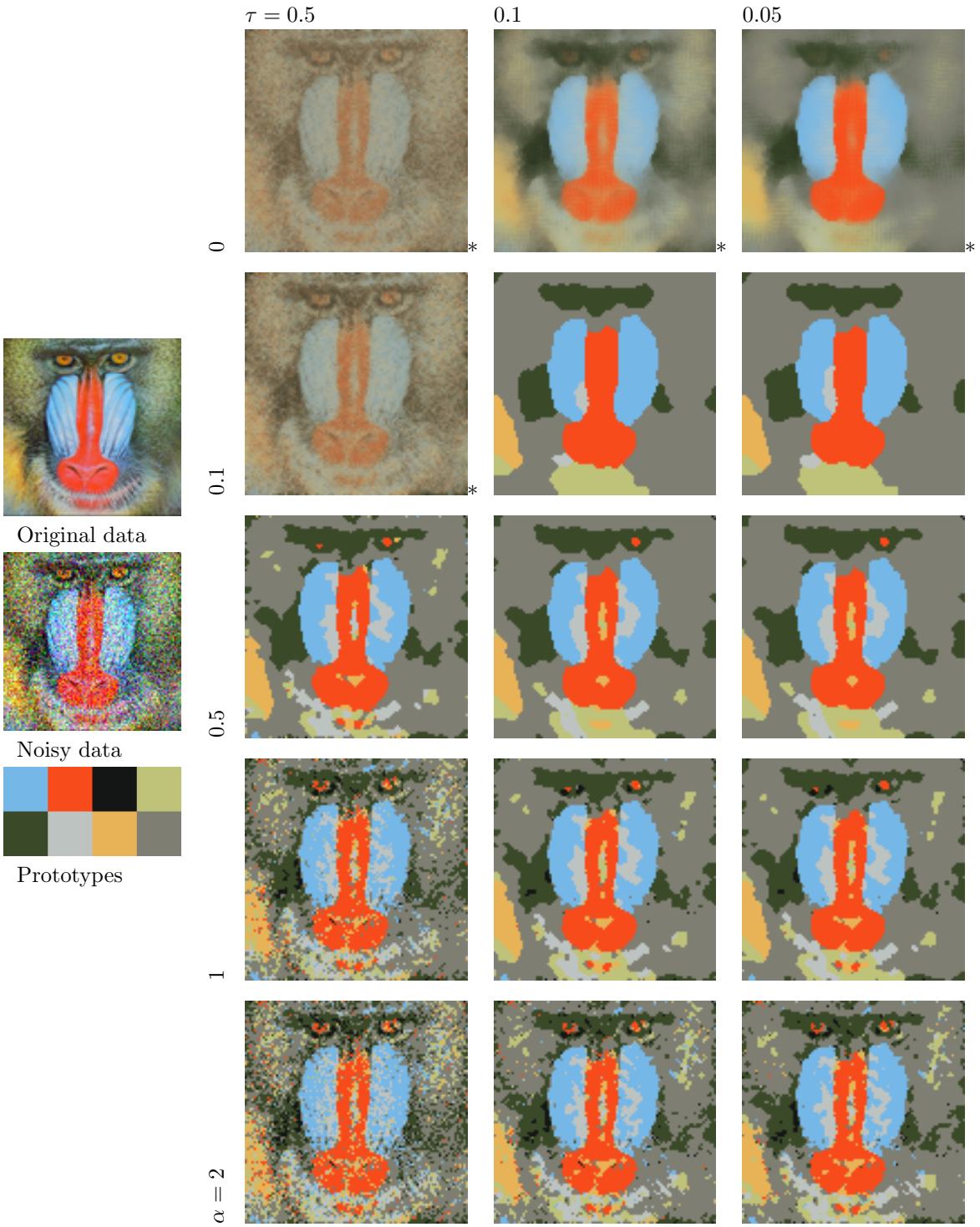


Figure 7.2. *Influence of the rounding parameter α and the smoothing parameter τ on the assignment of eight prototypical labels to noisy input data. All images marked with "$*$" do not show integral solutions due to smoothing too strongly the Wasserstein distance in terms of τ relative to α, which overcompensates for the effect of rounding. Likewise, smoothing too strongly the Wasserstein distance (left column, $\tau = 0.5$) yields poor approximations of the objective function gradient and leads to erroneous label assignments. The remaining parameter regime, i.e., smoothing below a reasonably large upper bound $\tau = 0.1$, leads to fast numerical convergence, and the label assignment can be precisely controlled by the rounding parameter α.*

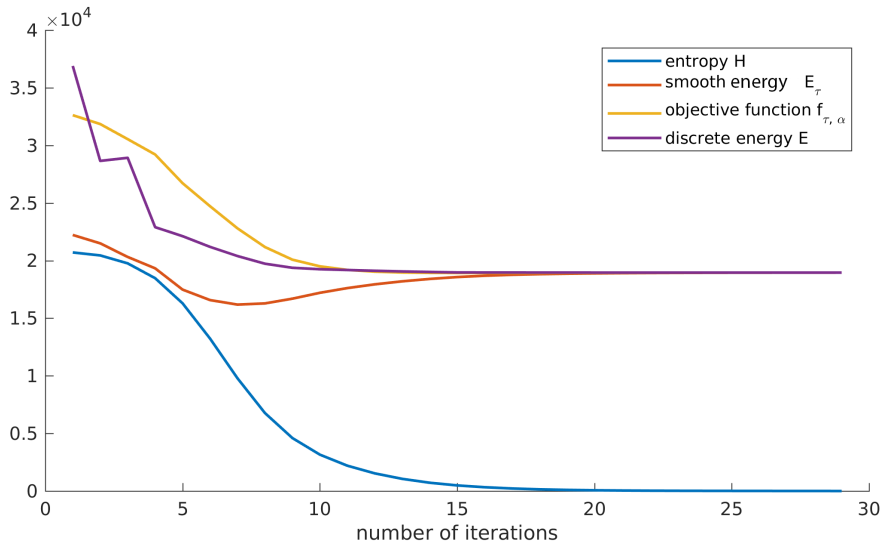


Figure 7.3. *Connection between the objective function $f_{\tau,\alpha}$ (5.6) and the discrete energy E (1.2) of the underlying graphical model, for a fixed value $\alpha = 0.5$. Minimizing $f_{\tau,\alpha}$ (yellow) by our approach also minimizes E (violet), which was calculated for this illustration by rounding the assignment vectors at every iterative step. Additionally, as already discussed in more detail in connection with Figure 7.1, the interplay between the two terms of $f_{\tau,\alpha} = E_\tau + \alpha H$ is shown, where E_τ (orange) denotes the smoothed energy (1.4) and H (blue) the entropy (5.5) causing rounding.*

Figure 7.2 shows the influence of the rounding strength α and the smoothing parameter τ for the Wasserstein distance. All images marked with "$*$" in the lower right corner do not show an integral solution, which means that the normalized average entropy (6.11) of the assignment vectors W_i did not drop below the threshold during the iteration and thus, even though the assignments show a clear tendency, they stayed far from integral solutions. As just explained for Figure 7.1, this is not a deficiency of our approach but must happen if either no rounding is performed ($\alpha = 0$) or if the influence of rounding is too small compared to the smoothing of the Wasserstein distance (e.g., $\alpha = 0.1$ and $\tau = 0.5$). Increasing the strength of rounding (larger α) leads to a faster decrease in entropy (cf. Figure 7.1 for the case of $\tau = 0.1$) and therefore to an earlier convergence of the process to a specific labeling. Thus, a more aggressive rounding scheme yields a less regularized result due to the rapid decision for a labeling at an early stage of the algorithm.

On the other hand, choosing the smoothing parameter τ too large leads to poor approximations of the Wasserstein distance gradients and consequently to erroneous nonregularized labelings, as displayed in the left column of Figure 7.2 corresponding to $\tau = 0.5$. Once τ is small enough (in our experiments, $\tau < 0.1$), the Wasserstein distance gradients are properly approximated, and the label assignment is regularized as expected and can be controlled by α. In particular, this upper bound on τ is sufficiently large to ensure very rapid convergence of the fixed point iteration for computing the Wasserstein distance gradients.

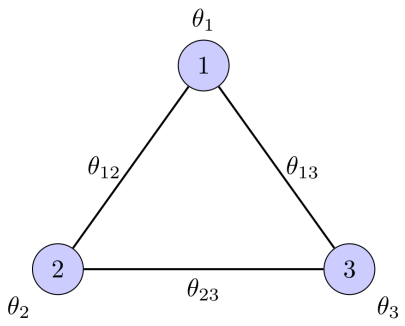
Figure 7.3 shows the connection between the objective function $f_{\tau,\alpha}$ (5.6) and the discrete energy E (1.2) of the underlying graphical model. Minimizing $f_{\tau,\alpha}$ (yellow curve) using our

approach also minimizes the discrete energy E (violet curve), which was calculated by rounding the assignment vectors after each iterative step. Figure 7.3 also shows the interplay between the two terms in $f_{\tau,\alpha} = E_\tau + \alpha H$, with smoothed energy (1.4) E_τ plotted as the orange curve and with the entropy (5.5) plotted as the blue curve. These curves illustrate (i) the smooth combination of optimization and rounding into a single process, and (ii) that the original discrete energy (1.2) is effectively minimized by this smooth process.

7.2. Exploring all cyclic graphical models on $\mathcal{K}^3$. In this section, we report an exhaustive exploration of all possible binary models, $\mathcal{X} = \{0, 1\}$, on the minimal cyclic graph $\mathcal{K}^3$ (Figure 7.4, left panel). Due to the single cycle, models exist where the LP relaxation (1.3) returns a nonbinary solution (red part of the right panel of Figure 7.4). As a consequence, evaluating such models with our geometric approach for minimizing (1.4) enables us to check the following two properties:

1. Whenever solving the LP relaxation (1.3) by convex programming returns the global binary minimum of (1.2) as solution, we assess whether our geometric approach based on the smooth approximation (1.4) returns this solution as well.
2. Whenever the LP relaxation has a *nonbinary* vector as a global solution, which therefore is *not* optimal for the labeling problem (1.2), we assess the rounding property of our approach by comparing the result with the *correct* binary labeling globally minimizing (1.2).

The graph $\mathcal{K}^3$ enables us to specify the so-called *marginal polytope* $\mathcal{P}_{\mathcal{K}^3}$, whose vertices (extreme points) are the feasible binary combinatorial solutions that correspond to valid labelings (cf. section 1.1), and to examine the difference to the local polytope $\mathcal{L}_{\mathcal{K}^3}$, whose representation only involves a subset of the constraints corresponding to $\mathcal{P}_{\mathcal{K}^3}$. We refer the reader to [32] for background and details.



μ_1	μ_2	μ_3	μ_{12}	μ_{13}	μ_{23}
0	0	0	0	0	0
0	0	1	0	0	0
0	1	0	0	0	0
0	1	1	0	0	1
1	0	0	0	0	0
1	0	1	0	1	0
1	1	0	1	0	0
1	1	1	1	1	1
$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	0	0	0
$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	0	$\frac{1}{2}$	$\frac{1}{2}$
$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	0	$\frac{1}{2}$
$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	0

Figure 7.4. *Left: The minimal binary cyclic graphical model $\mathcal{K}^3 = (\mathcal{V}, \mathcal{E}) = (\{1, 2, 3\}, \{12, 13, 23\})$. Right: The eight vertices (white background) of the minimally represented marginal polytope $\mathcal{P}_{\mathcal{K}^3} \subset \mathbb{R}^6_+$ and the four additional noninteger vertices (red background) of the minimally represented local polytope $\mathcal{L}_{\mathcal{K}^3} \subset \mathbb{R}^6_+$.*

The constraints are more conveniently stated using the so-called *minimal representation* of binary graphical models [50, section 3.2], which involves the variables[1]

$$\mu_i := \mu_i(1),\ i \in \mathcal{V}, \qquad \mu_{ij} := \mu_i(1)\mu_j(1),\ ij \in \mathcal{E}, \tag{7.3}$$

and encodes the local vectors (2.15) by

$$\begin{pmatrix} 1-\mu_i \\ \mu_i \end{pmatrix} \leftarrow \begin{pmatrix} \mu_i(0) \\ \mu_i(1) \end{pmatrix}, \qquad \begin{pmatrix} (1-\mu_i)(1-\mu_j) \\ (1-\mu_i)\mu_j \\ \mu_i(1-\mu_j) \\ \mu_{ij} \end{pmatrix} \leftarrow \begin{pmatrix} \mu_{ij}(0,0) \\ \mu_{ij}(0,1) \\ \mu_{ij}(1,0) \\ \mu_{ij}(1,1) \end{pmatrix}. \tag{7.4}$$

Thus, it suffices to use a single variable μ_i for every node $i \in \mathcal{V}$ instead of two variables $\mu_i(0), \mu_i(1)$, and also a single variable μ_{ij} for every edge $ij \in \mathcal{E}$ instead of four variables $\mu_{ij}(0,0), \mu_{ij}(0,1), \mu_{ij}(1,0), \mu_{ij}(1,1)$. The *local* polytope constraints (2.15) then take the form

$$0 \le \mu_{ij}, \qquad \mu_{ij} \le \mu_i, \qquad \mu_{ij} \le \mu_j, \qquad \mu_i + \mu_j - \mu_{ij} \le 1 \qquad \forall ij \in \mathcal{E}. \tag{7.5}$$

The *marginal* polytope constraints additionally involve the so-called triangle inequalities [19]

$$\sum_{i \in \mathcal{V}} \mu_i - \sum_{jk \in \mathcal{E}} \mu_{jk} \le 1, \tag{7.6a}$$

$$\mu_{12} + \mu_{13} - \mu_{23} \le \mu_1, \qquad \mu_{12} - \mu_{13} + \mu_{23} \le \mu_2, \qquad -\mu_{12} + \mu_{13} + \mu_{23} \le \mu_3. \tag{7.6b}$$

Figure 7.4, right panel, lists the eight vertices of $\mathcal{P}_{\mathcal{K}^3}$ and the four additional vertices of $\mathcal{L}_{\mathcal{K}^3}$ that arise when dropping the subset of constraints (7.6).

We evaluated 10^5 models generated by randomly sampling the model parameters (2.11): with $\mathcal{U}[a,b]$ denoting the uniform distribution on the interval $[a,b] \subset \mathbb{R}$, we set

$$\theta_i = \begin{pmatrix} 1-p \\ p \end{pmatrix} - \frac{1}{2}\begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad p \sim \mathcal{U}[0,1], \qquad \theta_{ij} = \begin{pmatrix} p_1 & p_2 \\ p_3 & p_4 \end{pmatrix}, \quad p_i \sim \mathcal{U}[-2,2],\ i \in [4]. \tag{7.7}$$

Note the different scale, $\theta_i \in [-\frac{1}{2}, +\frac{1}{2}]^2$, $\theta_{ij} \in [-2,+2]^{2\times 2}$, which results in a larger influence of the pairwise terms and hence makes inference more difficult. Suppose, for example, that the diagonal terms of θ_{ij} are large, which favors the assignment of *different* labels to the nodes $1,2,3 \in \mathcal{V}$. Then assigning, say, labels 0 and 1 to the vertices 1 and 2, respectively, will inherently lead to a large energy contribution due to the assignment to node 3, no matter if this third label is 0 or 1, because it must agree with the assignment either to node 1 or to 2.

Every *binary* vertex listed in Figure 7.4, right panel, is the global optimum of both the linear relaxation (1.3) and the original objective function (1.2) in approximately 11.94% of the 10^5 scenarios, whereas every *nonbinary* vertex is optimal in approximately 1.12%.

[1] We reuse the symbol μ for simplicity and only "overload" in this subsection the symbols μ_i, μ_{ij} for local vectors (2.15) by using the variables on the left-hand side of (7.3).

An example where a *nonbinary* vertex is optimal for the linear relaxation (1.3) is given by the model parameter values

$$\begin{aligned} \theta_1 &= \begin{pmatrix} -0.2261 \\ 0.2261 \end{pmatrix}, & \theta_{12} &= \begin{pmatrix} -0.9184 & -1.6252 \\ -1.8891 & -0.9807 \end{pmatrix}, \\ \theta_2 &= \begin{pmatrix} -0.4449 \\ 0.4449 \end{pmatrix}, & \theta_{13} &= \begin{pmatrix} 0.3590 & 0.0958 \\ -1.8668 & 1.5193 \end{pmatrix}, \\ \theta_3 &= \begin{pmatrix} -0.3202 \\ 0.3202 \end{pmatrix}, & \theta_{23} &= \begin{pmatrix} 1.2147 & -1.5215 \\ -0.3302 & -0.0459 \end{pmatrix}. \end{aligned} \tag{7.8}$$

The corresponding solutions of the marginal polytope $\mathcal{M}_\mathcal{G}$, the local polytope $\mathcal{L}_\mathcal{G}$, and our method are listed in Table 7.1. Due to the nonbinary solution returned by the LP relaxation, rounding in a postprocessing step amounts to random guessing. In contrast, our method is able to determine the optimal solution because rounding is smoothly integrated into the overall optimization process.

Table 7.1

Solutions $\mu = (\mu_1, \mu_2, \mu_2)$ of the marginal polytope $\mathcal{M}_\mathcal{G}$, the local polytope $\mathcal{L}_\mathcal{G}$, and our method, for the triangle model with parameter values (7.8). Our method was applied with threshold 10^{-3} as termination criterion (6.11), step-size $h = 0.5$, smoothing parameter $\tau = 0.1$, and three values of the rounding parameter $\alpha \in \{0.2, 0.5, 0.9\}$. By definition, minimizing over the marginal polytope returns the globally optimal discrete solution. The local polytope relaxation has a fractional solution for this model, so that rounding in a postprocessing step amounts to random guessing. Our approach returns the global optimum in each case, up to numerical precision.

		μ_1	μ_2	μ_3	Iterations
Marginal polytope $\mathcal{M}_\mathcal{G}$		1	0	0	-
Local polytope $\mathcal{L}_\mathcal{G}$		0.5	0.5	0.5	-
Our method ($\tau = \frac{1}{10}$)	$\alpha = 0.2$	0.999	0.258e−3	0.205e−3	108
	$\alpha = 0.5$	0.999	0.161e−3	0.114e−4	14
	$\alpha = 0.9$	0.999	0.239e−4	0.546e−6	8

Figure 7.5 presents the results of the experiments for the minimal cyclic graphical model $\mathcal{K}^3$. In order to assess clearly the influence of the *rounding* parameter α and the *smoothing* parameter τ, we evaluated all 10^5 models for *each pair* of (τ, α), where $\tau \in \{\frac{1}{2}, \frac{1}{2.5}, \ldots, \frac{1}{6.5}, \frac{1}{7}\}$ and $\alpha \in \{0.1, 0.11, \ldots, 0.99, 1\}$. These statistics show that our algorithm converges to integral solutions, except for very unbalanced parameter values: strong smoothing with large τ, weak rounding with small α. Within the remaining broad parameter regime, parameter α enables us to control the influence of rounding. In particular, in agreement with Figure 7.1 (bottom), less aggressive rounding computed labelings closer to the global optimum.

Table 7.2 displays success rate and number of iterations for three different parameter configurations from Figure 7.5. For instance, using $\alpha = 0.22$ and $\tau = 0.2$, our algorithm found in 97.35% of the experiments an energy with relative error smaller than 1% with respect to the optimal energy. In addition, the algorithm required on average 45 iterations to converge. Using instead $\alpha = 0.58$ and $\tau = 0.15$, that is, more aggressive rounding in each iteration step

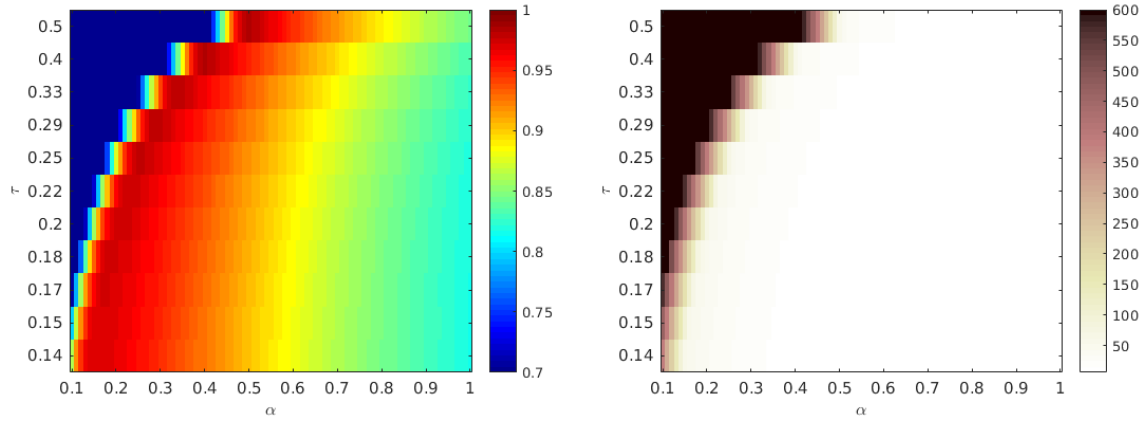


Figure 7.5. *Evaluation of the minimal cyclic graphical model $\mathcal{K}^3$. For every pair of parameter values (τ, α), we evaluated 10^5 models, which were generated as explained in the text. In each experiment, we terminated the algorithm if the average entropy dropped below 10^{-3} or if the maximum number of 600 iterations was reached. In addition, we chose a constant step-size $h = 0.5$. Left: The plot shows the percentage of experiments where the energy returned by our algorithm had a relative error smaller than 1% compared to the minimal energy of the globally optimal integral labeling. In agreement with Figure 7.1 (bottom), less aggressive rounding yielded labelings closer to the global optimum. Right: This plot shows the corresponding average number of iterations. The black region indicates experiments where the maximum number of 600 iterations was reached, because too strong smoothing of the Wasserstein distance (large τ) overcompensated for the effect of rounding (small α), so that the convergence criterion (6.11), which measures the distance to integral solutions, cannot be satisfied. In the remaining large parameter regime, the choice of α enables us to control the trade-off between high-quality (low-energy) solutions and computational costs.*

Table 7.2

Three different parameter configurations extracted from Figure 7.5. The comparison of the success rate and the number of iterations until convergence clearly demonstrates the trade-off between accuracy of optimization and convergence rate, depending on the rounding variable α and the smoothing parameter τ. Overall, the number of iterations is significantly smaller than for first-order methods of convex programming for solving the LP relaxation, which additionally require rounding as a postprocessing step to obtain an integral solution.

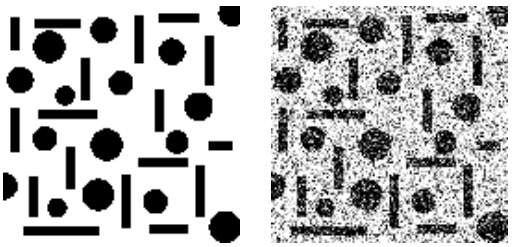
α	τ	Success rate	Iterations
0.22	0.2	97.35%	45
0.5	0.33	93.41%	15
0.58	0.15	88.6%	9

(5.4), the average number of iterations reduced to 9, but the accuracy also dropped to 88.6%.

Overall, these experiments clearly demonstrate

- the ability to control the trade-off between high-quality (low-energy) labelings and computational costs in terms of α for all values of τ below a reasonably large upper bound; and
- a small or very small number of iterations required to converge, depending on the choice of α.

7.3. Comparison to other methods. We compared our geometric approach to sequential tree-reweighted message passing (TRWS) [27] and loopy belief propagation (loopy-BP) [51]



Original data Noisy data

Figure 7.6. *Noisy image labeling problem: a binary ground truth image (left) to be recovered from noisy input data (right).*

Geometric
4977.24 / 98.38%
82.4 sec

TRWS
4979.61 / 98.07%
0.206 sec

Loopy-BP
4977.75 / 98.38%
81.4 sec

Figure 7.7. *Results for the noisy labeling problem from Figure 7.6 using a standard data term with Potts prior, with discrete energy/accuracy/runtime. Parameter values for the geometric approach: smoothing $\tau = 0.1$, step-size $h = 0.2$, and rounding strength $\alpha = 0.1$. The threshold for the termination criterion was 10^{-3}. All methods show similar performance.*

based on the OpenGM package [3].

For this comparison, we evaluated the performance of the methods for a noisy binary labeling scenario depicted in Figure 7.6. Let $f\colon \mathcal{V} \to [0,1]$ denote the noisy image data given on the grid graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with a 4-neighborhood and $\mathcal{X} = \{0,1\}$ as prototypes (labels). The following data term and Potts prior were used:

$$\theta_i = \begin{pmatrix} f(i) \\ 1 - f(i) \end{pmatrix} \quad \text{for} \quad i \in \mathcal{V} \quad \text{and} \quad \theta_{ij} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad \text{for} \quad ij \in \mathcal{E}\,. \tag{7.9}$$

The threshold 10^{-3} was used for the normalized average entropy termination criterion (6.11). Figure 7.7 shows the visual reconstruction as well as the corresponding discrete energy values and percentage of correct labels for all three methods. Our method has similar accuracy and returns a slightly better optimal discrete energy level than TRWS and loopy-BP.

We investigated again the influence of the *rounding* mechanism by repeating the same experiment but using different values of the rounding parameter $\alpha \in \{0.1, 1, 2, 5\}$. As shown in Figure 7.8, the results confirm the finding of the experiments of the preceding section: a more aggressive rounding scheme (α large) leads to faster convergence but yields less regularized results with higher energy values.

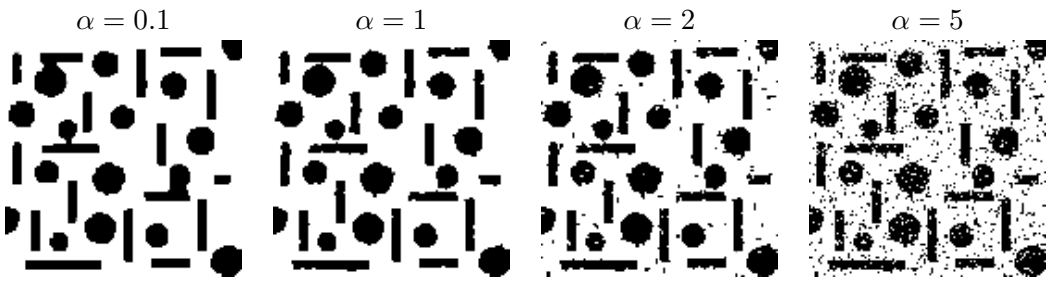


Figure 7.8. *Results for the noisy labeling problem from Figure 7.6 using different values of the rounding parameter $\alpha \in \{0.1, 1, 2, 5\}$ with discrete energy/accuracy/runtime: more aggressive rounding scheme (α large) leads to less regularized results with higher energy values. Parameter values of the geometric approach: smoothing $\tau = 0.1$, step-size $h = 0.2$, threshold 10^{-3} for termination.*

7.4. Nonuniform (non-Potts) priors. We examined the behavior of our approach for a non-Potts prior by applying it to a nonbinary labeling problem with noisy input data, as depicted in Figure 7.9. Our objective is to demonstrate that prespecified pairwise model parameters (regularization) by a graphical model are properly taken into account.

The label indices corresponding to the five RGB colors of the original image (Figure 7.9, right) are

$$\mathcal{X} = \{\ell_1 = \text{dark blue},\ \ell_2 = \text{light blue},\ \ell_3 = \text{cyan},\ \ell_4 = \text{orange},\ \ell_5 = \text{yellow}\} \subset [0,1]^3\ . \tag{7.10}$$

Let $f\colon \mathcal{V} \to [0,1]^3$ denote the noisy input image (Figure 7.9, center panel) given on the grid graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with a 4-neighborhood. This image was created by randomly selecting 40% of the original image pixels and then uniformly sampling a label at each chosen position. The unary term was defined using the $\|\cdot\|_1$ distance and a scaling factor $\rho > 0$ by

$$\theta_i = \frac{1}{\rho}\big(\|f(i) - \ell_1\|_1, \ldots, \|f(i) - \ell_5\|_1\big), \quad i \in \mathcal{V}. \tag{7.11}$$

Now assume that additional information about a labeling problem was available. For example, let the RGB color dark blue in the image represent the direction "top," light blue "bottom," yellow "right," orange "left," and cyan "center" (Figure 7.9, left). Suppose it is known beforehand that "top" and "bottom," as well as "left" and "right," cannot be adjacent to each other but are separated by another label corresponding to the center. This prior knowledge about the labeling problem was taken into account by specifying nonuniform pairwise model parameters that penalize these unlikely label transitions by a factor of 10:

$$\theta_{ij} = \frac{1}{10}\begin{pmatrix} 0 & 10 & 1 & 1 & 1 \\ 10 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 10 \\ 1 & 1 & 1 & 10 & 0 \end{pmatrix}, \qquad ij \in \mathcal{E}. \tag{7.12}$$

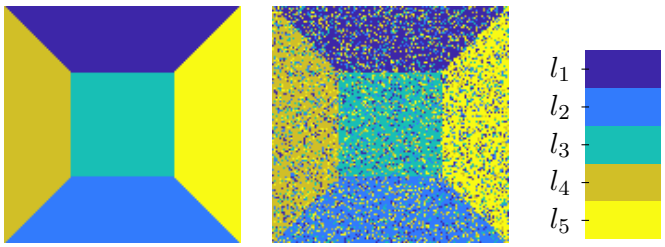


Figure 7.9. *Original image (left), encoding the image directions top, bottom, center, left, and right by the RGB color labels $\ell_1, \ell_2, \ell_3, \ell_4$, and ℓ_5 (right). The noisy test image (middle) was created by randomly selecting 40% of the original image pixels and then uniformly sampling a label at each position. Unlikely label transitions $\ell_1 \leftrightarrow \ell_2$ and $\ell_4 \leftrightarrow \ell_5$ are represented by color (feature) vectors that are close to each other, and hence can be easily confused.*

In other words, every entry of θ_{ij} corresponding to a label transition ℓ_1 = "dark blue" ("top") next to ℓ_2 = "light blue" ("bottom") or ℓ_4 = "orange" ("left") next to ℓ_5 = "yellow" ("right") has the large penalty value 1, whereas all other "natural" configurations are treated as with the Potts prior and smaller penalty values of 0 and 0.1, respectively. We point out that no color vectors or any other embedding was used to facilitate this regularization task or to represent it in a more application-specific way. Rather, the nonuniform prior (7.12) was considered as *given* in terms of some discrete graphical models and its energy function (1.2). On the other hand, the pairs of labels (ℓ_1, ℓ_2) and (ℓ_4, ℓ_5) forming unlikely label transitions can be easily confused by the data term, due to the small distance of the color (feature) vectors representing these labels.

To demonstrate how these nonuniform model parameters influence label assignments, we compared the evaluation of this model against a model with a uniform Potts prior

$$(\theta'_{ij})_{k,r} = \tfrac{1}{10}(1 - \delta_{k,r}), \quad \text{where} \quad \delta_{k,r} = \begin{cases} 1 & \text{if } k = r \\ 0 & \text{else} \end{cases} \qquad \text{for} \quad ij \in \mathcal{E}. \tag{7.13}$$

In our experiments, we used the scaling factor $\rho = 15$ for the unaries, step-size $h = 0.1$, rounding parameter $\alpha = 0.01$, smoothing parameter $\tau = 0.01$, and 10^{-4} as threshold for the normalized average entropy termination criterion (6.11).

The results depicted in Figure 7.10 clearly show the positive influence of the non-Potts prior (labeling accuracy 99.34%), whereas using the Potts prior lowers the accuracy to 87.12%. This is due to the fact that the color labels ℓ_4 and ℓ_5, as well as ℓ_1 and ℓ_2, have a relatively small $\|\cdot\|_1$ distance and are therefore not easy to distinguish using both the data term and a Potts prior. On the other hand, the additional prior information about valid label configurations encoded by (7.12) was sufficient to overcome this difficulty, despite using the same data term, and to separate the regions correctly.

8. Conclusion. We presented a novel approach to the evaluation of discrete graphical models in a smooth geometric setting. The novel inference algorithm propagates in parallel "Wasserstein messages" along edges. These messages are lifted to the assignment manifold and drive a Riemannian gradient flow that terminates at an integral labeling. Local marginal-

Potts
Acc: 87.12%

non-Potts
Acc: 99.34%

Figure 7.10. *Results of the labeling problem using the Potts and non-Potts prior models together with the accuracy (Acc) values. Parameters for this experiment are* $\rho = 15$*, smoothing* $\tau = 0.01$*, step-size* $h = 0.1$*, and rounding strength* $\alpha = 0.01$*. The threshold for the termination criterion* (6.11) *was* 10^{-4}*.*

ization constraints are satisfied throughout the process. A single parameter enables a trade-off between accuracy of optimization and speed of convergence.

Our work motivates applications using graphical models with higher edge connectivity, where established inference algorithms based on convex programming noticeably slow down. Likewise, generalizing our approach to tighter relaxations based on hypergraphs and corresponding entropy approximations [54, 33] seems worth additional investigation. Our future work will leverage the inherent smoothness of our mathematical setting for designing more advanced numerical schemes based on higher-order geometric integration and using multiple spatial scales.

Appendix A. Proofs.

A.1. Proof of Proposition 4.4. Let $\gamma\colon (-\varepsilon, \varepsilon) \to \mathcal{W}$ be a smooth curve, with $\varepsilon > 0$, $\gamma(0) = W$, and $\dot\gamma(0) = V$. We then have

(A.1)
$$\langle \nabla E_\tau(W), V\rangle = \frac{d}{dt} E_\tau\big(\gamma(t)\big)\Big|_{t=0} \overset{(4.12)}{=} \sum_{i\in\mathcal{V}} \Big(\langle P_T(\theta_i), V_i\rangle + \sum_{j\colon (i,j)\in\mathcal{E}} \frac{d}{dt} d_{\theta_{ij},\tau}(\gamma_i(t), \gamma_j(t))\Big|_{t=0} \Big) ,$$

where $\gamma_k(t)$ denotes the kth row of the matrix $\gamma(t) \in \mathcal{W} \subset \mathbb{R}^{m\times n}$. Since

$$\frac{d}{dt} d_{\theta_{ij},\tau}(\gamma_i(t), \gamma_j(t))\Big|_{t=0} = \langle \nabla_1 d_{\theta_{ij},\tau}(W_i, W_j), V_i\rangle + \langle \nabla_2 d_{\theta_{ij},\tau}(W_i, W_j), V_j\rangle , \tag{A.2}$$

the right-hand side of (A.1) becomes

$$\begin{aligned} \langle \nabla E_\tau(W), V\rangle = \sum_{i\in\mathcal{V}} \Big(\langle P_T(\theta_i), V_i\rangle + \sum_{j\colon (i,j)\in\mathcal{E}} \langle \nabla_1 d_{\theta_{ij},\tau}(W_i, W_j), V_i\rangle \Big) \\ + \sum_{i\in\mathcal{V}} \sum_{j\colon (i,j)\in\mathcal{E}} \langle \nabla_2 d_{\theta_{ij},\tau}(W_i, W_j), V_j\rangle , \end{aligned} \tag{A.3}$$

where we deliberately separated the outer sum into two parts. Let $\delta_{(k,l)\in\mathcal{E}}$ be the function with value 1 if $(k,l) \in \mathcal{E}$ and 0 if $(k,l) \notin \mathcal{E}$. Then the second sum of the expression above

reads as

$$\sum_{i\in\mathcal{V}}\sum_{j\colon (i,j)\in\mathcal{E}}\langle\nabla_2 d_{\theta_{ij},\tau}(W_i,W_j),V_j\rangle=\sum_{i\in\mathcal{V}}\sum_{j\in\mathcal{V}}\delta_{(i,j)\in\mathcal{E}}\langle\nabla_2 d_{\theta_{ij},\tau}(W_i,W_j),V_j\rangle \tag{A.4a}$$

$$=\sum_{j\in\mathcal{V}}\sum_{i\in\mathcal{V}}\delta_{(i,j)\in\mathcal{E}}\langle\nabla_2 d_{\theta_{ij},\tau}(W_i,W_j),V_j\rangle \tag{A.4b}$$

$$=\sum_{j\in\mathcal{V}}\sum_{i\colon (i,j)\in\mathcal{E}}\langle\nabla_2 d_{\theta_{ij},\tau}(W_i,W_j),V_j\rangle \tag{A.4c}$$

$$=\sum_{i\in\mathcal{V}}\sum_{j\colon (j,i)\in\mathcal{E}}\langle\nabla_2 d_{\theta_{ji},\tau}(W_j,W_i),V_i\rangle\,, \tag{A.4d}$$

where the last equation follows by renaming the indices of summation. Substitution into (A.3) gives

$$\langle\nabla E_\tau(W),V\rangle=\sum_{i\in\mathcal{V}}\Big\langle P_T(\theta_i)+\sum_{j\colon (i,j)\in\mathcal{E}}\nabla_1 d_{\theta_{ij},\tau}(W_i,W_j)+\sum_{j\colon (j,i)\in\mathcal{E}}\nabla_2 d_{\theta_{ji},\tau}(W_j,W_i),V_i\Big\rangle \tag{A.5a}$$

$$=\sum_{i\in\mathcal{V}}\langle\nabla_i E_\tau(W),V_i\rangle, \tag{A.5b}$$

which proves (4.13).

A.2. Proof of Lemma 4.12. We first show that if $\overline{\nu}$ is an optimal dual solution, then

$$\operatorname*{argmax}_{\nu\in\mathbb{R}^{2n}} g(p,\nu)\subseteq\overline{\nu}+\mathcal{N}(\mathcal{A}^\top). \tag{A.6}$$

Let $\overline{\nu}'\neq\overline{\nu}$ be another optimal dual solution, that is, $g(p,\overline{\nu})=g(p,\overline{\nu}')$. By (4.21), this equation reads as

$$G_\tau^*(\mathcal{A}^\top\overline{\nu}-\Theta)-G_\tau^*(\mathcal{A}^\top\overline{\nu}'-\Theta)=\langle p,\overline{\nu}-\overline{\nu}'\rangle\,. \tag{A.7}$$

Moreover, due to the optimality conditions (4.27), $\overline{\nu}'$ satisfies

$$\overline{M}'=\nabla G_\tau^*(\mathcal{A}^\top\overline{\nu}'-\Theta),\qquad \mathcal{A}\overline{M}'=p, \tag{A.8}$$

with a corresponding primal optimal solution $\overline{M}'$. Hence

$$\langle p,\overline{\nu}-\overline{\nu}'\rangle=\langle\mathcal{A}\overline{M}',\overline{\nu}-\overline{\nu}'\rangle=\langle\overline{M}',\mathcal{A}^\top(\overline{\nu}-\overline{\nu}')\rangle\overset{(A.8)}{=}\langle\nabla G_\tau^*(\mathcal{A}^\top\overline{\nu}'-\Theta),\mathcal{A}^\top(\overline{\nu}-\overline{\nu}')\rangle\,. \tag{A.9}$$

Using the shorthand

$$\overline{w}=\mathcal{A}^\top\overline{\nu}-\Theta,\qquad \overline{w}'=\mathcal{A}^\top\overline{\nu}'-\Theta, \tag{A.10}$$

we have

$$\overline{w}'-\overline{w}=\mathcal{A}^\top(\overline{\nu}'-\overline{\nu}), \tag{A.11}$$

and therefore

$$G_\tau^*(\overline{w}') - G_\tau^*(\overline{w}) \overset{(A.7)}{=} \langle p, \overline{\nu}' - \overline{\nu} \rangle \overset{(A.9)}{=} \langle \nabla G_\tau^*(\overline{w}'), \overline{w}' - \overline{w} \rangle. \tag{A.12}$$

Since G_τ^* is strictly convex, this equality can hold only if

$$0 = \overline{w}' - \overline{w} \overset{(A.11)}{=} \mathcal{A}^\top(\overline{\nu}' - \overline{\nu}). \tag{A.13}$$

This shows that $\overline{\nu}$ and $\overline{\nu}'$ can differ only by a nullspace vector; i.e., we have shown relation (A.6). It remains to show the reverse inclusion; that is, vectors characterized by the right-hand side of (4.33) maximize the dual objective function $g(p, \nu)$.

Again let $\overline{\nu}$ be an optimal dual solution, and let $\overline{\nu}' \in \overline{\nu} + \mathcal{N}(\mathcal{A}^\top)$ be an arbitrary vector. Lemma 4.11 implies that $\overline{\nu}'$ takes the form

$$\overline{\nu}' = \overline{\nu} + \alpha \begin{pmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{pmatrix}, \qquad \alpha \in \mathbb{R}. \tag{A.14}$$

Now suppose $\left\langle p, \begin{pmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{pmatrix} \right\rangle = 0$. Then, since $\mathcal{A}^\top \overline{\nu}' = \mathcal{A}^\top \overline{\nu}$, we have

$$g(a, \overline{\nu}') = \left\langle p, \overline{\nu} + \alpha \begin{pmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{pmatrix} \right\rangle - G_\tau^*\left(\mathcal{A}^\top\left(\overline{\nu} + \alpha \begin{pmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{pmatrix}\right) - \Theta\right) \tag{A.15a}$$

$$= \langle p, \nu \rangle - G_\tau^*(\mathcal{A}^\top \overline{\nu} - \Theta) = g(a, \overline{\nu}), \tag{A.15b}$$

that is, $\overline{\nu}' \in \operatorname{argmax}_{\nu \in \mathbb{R}^{2n}} g(p, \nu)$.

Finally, suppose $\left\langle p, \begin{pmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{pmatrix} \right\rangle \neq 0$, $\overline{\nu}$ is an optimal dual solution, and $\overline{\nu}'$ is another optimal dual vector, which has the form (A.14) as just shown. Inserting (A.14) into (A.7) yields

$$0 = \langle p, \overline{\nu}' - \overline{\nu} \rangle = \alpha \left\langle p, \begin{pmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{pmatrix} \right\rangle. \tag{A.16}$$

Since $\left\langle p, \begin{pmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{pmatrix} \right\rangle \neq 0$, this can hold only if $\alpha = 0$. Thus, $\overline{\nu}' = \overline{\nu}$ by (A.14), which shows uniqueness of $\overline{\nu}$ as claimed by (4.33).

Acknowledgment. We thank Jan Kuske for sharing with us his framework for running series of experiments efficiently.

REFERENCES

[1] A. Aaron, J. Fakcharoenphol, C. Harrelson, R. Krauthgamer, K. Talwar, and E. Tardos, *Approximate classification via earthmover metrics*, in Proceedings of the Fifteenth Annual ACM-SIAM Symposium on Discrete Algorithms, ACM, SIAM, 2004, pp. 1079–1087.

[2] L. Ambrosio, *Variational problems in SBV and image segmentation*, Acta Appl. Math., 17 (1989), pp. 1–40.

[3] B. Andres, T. Beier, and J. H. Kappes, *OpenGM: A C++ Library for Discrete Graphical Models*, preprint, https://arxiv.org/abs/1206.0111, 2012.

[4] F. Åström, R. Hühnerbein, F. Savarino, J. Recknagel, and C. Schnörr, *MAP image labeling using Wasserstein messages and geometric assignment*, in Proc. International Conference on Scale Space and Variational Methods in Computer Vision (SSVM 2017): Scale Space and Variational Methods in Computer Vision, Lecture Notes in Comput. Sci. 10302, Springer, 2017, pp. 373–385.

[5] F. Åström, S. Petra, B. Schmitzer, and C. Schnörr, *Image labeling by assignment*, J. Math. Imaging Vision, 58 (2017), pp. 211–238.

[6] H. H. Bauschke and J. M. Borwein, *Legendre functions and the method of random Bregman projections*, J. Convex Anal., 4 (1997), pp. 27–67.
[7] R. Bergmann, J. Fitschen, J. Persch, and G. Steidl, *Iterative multiplicative filters for data labeling*, Int. J. Comput. Vis., 123 (2017), pp. 435–453.
[8] R. Bergmann and D. Tenbrinck, *A Graph Framework for Manifold-Valued Data*, preprint, https://arxiv.org/abs/1702.05293, 2017.
[9] A. L. Bertozzi and A. Flenner, *Diffuse interface models on graphs for classification of high dimensional data*, Multiscale Model. Simul., 10 (2012), pp. 1090–1118, https://doi.org/10.1137/11083109X.
[10] S. Boyd and L. Vandenberghe, *Convex Optimization*, 7th ed., Cambridge Univ. Press, 2009.
[11] Y. Boykov and V. Kolmogorov, *An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision*, IEEE Trans. Pattern Anal. Mach. Intell., 26 (2004), pp. 1124–1137.
[12] Y. Boykov, O. Veksler, and R. Zabih, *Fast approximate energy minimization via graph cuts*, IEEE Trans. Pattern Anal. Mach. Intell., 23 (2001), pp. 1222–1239.
[13] R. Brualdi, *Combinatorial Matrix Classes*, Cambridge Univ. Press, 2006.
[14] Y. Censor and S. Zenios, *Proximal minimization algorithm with D-functions*, J. Optim. Theory Appl., 73 (1992), pp. 451–464.
[15] C. Chekuri, S. Khanna, J. Naor, and L. Zosin, *A linear programming formulation and approximation algorithms for the metric labeling problem*, SIAM J. Discrete Math., 18 (2005), pp. 608–625, https://doi.org/10.1137/S0895480101396937.
[16] M. Cuturi, *Sinkhorn distances: Lightspeed computation of optimal transport*, in Advances in Neural Information Processing Systems 26, C. J. C. Burges et al., eds., Curran Associates, Inc., 2013, pp. 2292–2300.
[17] M. Cuturi and G. Peyré, *A smoothed dual approach for variational Wasserstein problems*, SIAM J. Imaging Sci., 9 (2016), pp. 320–343, https://doi.org/10.1137/15M1032600.
[18] J. M. Danskin, *The theory of max-min, with applications*, SIAM J. Appl. Math., 14 (1966), pp. 641–664, https://doi.org/10.1137/0114053.
[19] M. Deza and M. Laurent, *Geometry of Cuts and Metrics*, Springer, 1997.
[20] A. Elmoataz, O. Lezoray, and S. Bougleux, *Nonlocal discrete regularization on weighted graphs: A framework for image and manifold processing*, IEEE Trans. Image Process., 17 (2008), pp. 1047–1059.
[21] G. Gilboa and S. Osher, *Nonlocal operators with applications to image processing*, Multiscale Model. Simul., 7 (2008), pp. 1005–1028, https://doi.org/10.1137/070698592.
[22] T. Hazan and A. Shashua, *Norm-product belief propagation: Primal-dual message-passing for approximate inference*, IEEE Trans. Inform. Theory, 56 (2010), pp. 6294–6316.
[23] J. Kappes, B. Andres, F. Hamprecht, C. Schnörr, S. Nowozin, D. Batra, S. Kim, B. Kausler, T. Kröger, J. Lellmann, N. Komodakis, B. Savchynskyy, and C. Rother, *A comparative study of modern inference techniques for structured discrete energy minimization problems*, Int. J. Comput. Vis., 115 (2015), pp. 155–184.
[24] H. Karcher, *Riemannian center of mass and mollifier smoothing*, Comm. Pure Appl. Math., 30 (1977), pp. 509–541.
[25] J. Kleinberg and E. Tardos, *Approximation algorithms for classification problems with pairwise relationships: Metric labeling and Markov random fields*, J. ACM, 49 (2002), pp. 616–639.
[26] P. A. Knight, *The Sinkhorn–Knopp algorithm: Convergence and applications*, SIAM J. Matrix Anal. Appl., 30 (2008), pp. 261–275, https://doi.org/10.1137/060659624.
[27] V. Kolmogorov, *Convergent tree-reweighted message passing for energy minimization*, IEEE Trans. Pattern Anal. Mach. Intell., 28 (2006), pp. 1568–1583.
[28] V. Kolmogorov and R. Zabih, *What energy functions can be minimized via graph cuts?*, IEEE Trans. Pattern Anal. Mach. Intell., 26 (2004), pp. 147–159.
[29] S. Kolouri, S. Park, M. Thorpe, D. Slepcev, and G. Rohde, *Optimal mass transport: Signal processing and machine-learning applications*, IEEE Signal Process. Mag., 34 (2017), pp. 43–59.
[30] A. Nemirovsky and D. Yudin, *Problem Complexity and Method Efficiency in Optimization*, John Wiley & Sons, 1983.
[31] Y. Nesterov and A. Nemirovskii, *Interior-Point Polynomial Algorithms in Convex Programming*, SIAM Stud. Appl. Math. 13, SIAM, 1994, https://doi.org/10.1137/1.9781611970791.
[32] M. Padberg, *The Boolean Quadratic Polytope: Some Characteristics, Facets and Relatives*, Math. Progr., 45 (1989), pp. 139–172.

[33] P. Pakzad and V. Anantharam, *Estimation and marginalization using Kikuchi approximation methods*, Neural Comput., 17 (2005), pp. 1836–1873.
[34] G. Peyré, *Entropic approximation of Wasserstein gradient flows*, SIAM J. Imaging Sci., 8 (2015), pp. 2323–2351, https://doi.org/10.1137/15M1010087.
[35] T. Pham Dinh and L. Hoai An, *Convex analysis approach to D.C. programming: Theory, algorithms and applications*, Acta Math. Vietnamica, 22 (1997), pp. 289–355.
[36] T. Pham Dinh and L. T. Hoai An, *A D.C. optimization algorithm for solving the trust-region subproblem*, SIAM J. Optim., 8 (1998), pp. 476–505, https://doi.org/10.1137/S1052623494274313.
[37] P. Ravikumar, A. Agarwal, and M. J. Wainwright, *Message-passing for graph-structured linear programs: Proximal methods and rounding schemes*, J. Mach. Learn. Res., 11 (2010), pp. 1043–1080.
[38] J. Renegar, *Linear programming, complexity theory and elementary functional analysis*, Math. Progr., 70 (1995), pp. 279–351.
[39] R. Rockafellar, *On a special class of functions*, J. Optim. Theory Appl., 70 (1991), pp. 619–621.
[40] R. Rockafellar and R. J.-B. Wets, *Variational Analysis*, 2nd ed., Springer, 2009.
[41] R. T. Rockafellar, *Monotone operators and the proximal point algorithm*, SIAM J. Control Optim., 14 (1976), pp. 877–898, https://doi.org/10.1137/0314056.
[42] F. Savarino, R. Hühnerbein, F. Åström, J. Recknagel, and C. Schnörr, *Numerical integration of Riemannian gradient flows for image labeling*, in Proc. International Conference on Scale Space and Variational Methods in Computer Vision (SSVM 2017): Scale Space and Variational Methods in Computer Vision, Lecture Notes in Comput. Sci. 10302, Springer, 2017, pp 361–372.
[43] B. Schmitzer, *A sparse multiscale algorithm for dense optimal transport*, J. Math. Imaging Vision, 56 (2016), pp. 238–259.
[44] B. Schmitzer, *Stabilized Sparse Scaling Algorithms for Entropy Regularized Transport Problems*, preprint, https://arxiv.org/abs/1610.06519, 2016.
[45] M. Schneider, *Matrix scaling, entropy minimization, and conjugate duality (*II*): The dual problem*, Math. Progr., 48 (1990), pp. 103–124.
[46] R. Sinkhorn, *A relationship between arbitrary positive matrices and doubly stochastic matrices*, Ann. Math. Statist., 35 (1964), pp. 876–879.
[47] P. Swoboda, A. Shekhovtsov, J. Kappes, C. Schnörr, and B. Savchynskyy, *Partial optimality by pruning for MAP-inference with general graphical models*, IEEE Trans. Pattern Anal. Mach. Intell., 38 (2016), pp. 1370–1382.
[48] T. Terlaky, ed., *Interior Point Methods of Mathematical Programming*, Springer, 1996.
[49] M. Wainwright, T. Jaakola, and A. Willsky, *MAP estimation via agreement on trees: Message-passing and linear programming*, IEEE Trans. Inform. Theory, 51 (2005), pp. 3697– 3717.
[50] M. Wainwright and M. Jordan, *Graphical models, exponential families, and variational inference*, Found. Trends Mach. Learn., 1 (2008), pp. 1–305.
[51] Y. Weiss, *Comparing the mean field method and belief propagation for approximate inference in MRFs*, in Advanced Mean Field Methods: Theory and Practice, MIT Press, 2001, pp. 229–240.
[52] T. Werner, *A linear programming approach to max-sum problem: A review*, IEEE Trans. Pattern Anal. Mach. Intell., 29 (2007), pp. 1165–1179.
[53] C. Yanover, T. Meltzer, and Y. Weiss, *Linear programming relaxations and belief propagation—An empirical study*, J. Mach. Learn. Res., 7 (2006), pp. 1887–1907.
[54] J. Yedidia, W. Freeman, and Y. Weiss, *Constructing free-energy approximations and generalized belief propagation algorithms*, IEEE Trans. Inform. Theory, 51 (2005), pp. 2282–2312.