

# IMAGE LABELING BASED ON GRAPHICAL MODELS USING WASSERSTEIN MESSAGES AND GEOMETRIC ASSIGNMENT

RUBEN HÜHNERBEIN, FABRIZIO SAVARINO, FREDDIE ÅSTRÖM, CHRISTOPH SCHNÖRR

**ABSTRACT.** We introduce a novel approach to Maximum A Posteriori 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.

## CONTENTS

|  |    |
|--|----|
| 1. Introduction  | 2  |
| 1.1. Overview and Motivation   | 2  |
| 1.2. Related Work  | 3  |
| 1.3. Contribution and Organization   | 4  |
| 2. Preliminaries   | 5  |
| 2.1. Basic Notation  | 5  |
| 2.2. The Local Polytope Relaxation of the Labeling Problem                     | 7  |
| 3. Image Labeling on the Assignment Manifold                                   | 7  |
| 3.1. The Assignment Manifold   | 8  |
| 3.2. Image Labeling on $\mathcal{W}$   | 9  |
| 3.3. Geometric Integration of Gradient Flows                                   | 10 |
| 4. Energy, Gradients and Wasserstein Messages                                  | 11 |
| 4.1. Smooth Approximation of the LP Relaxation                                 | 11 |
| 4.2. Energy Gradient $\nabla E_\tau$   | 13 |
| 4.3. Local Wasserstein Distance Gradient                                       | 14 |
| 5. Application to Graphical Models   | 16 |
| 5.1. Smooth Intergration of Minimizing and Rounding on the Assignment Manifold | 17 |
| 5.2. Wasserstein Messages  | 18 |
| 6. Implementation  | 20 |
| 6.1. Assignment Normalization  | 20 |
| 6.2. Computing Wasserstein Gradients   | 21 |
| 6.3. Termination Criterion   | 23 |
| 7. Experiments   | 23 |

*Date:* October 5, 2017.

*2010 Mathematics Subject Classification.* 62H35, 62M40, 65K10, 68U10.

*Key words and phrases.* image labeling, assignment manifold, Fisher–Rao metric, Riemannian gradient flow, Wasserstein distance, graphical models.

Support by the German Science Foundation, grant GRK 1653, is gratefully acknowledged.

|   |    |
|---|----|
| 7.1. Parameter Influence                                      | 24 |
| 7.2. Exploring all Cyclic Graphical Models on $\mathcal{K}^3$ | 27 |
| 7.3. Comparison to other Methods                              | 30 |
| 7.4. Non-Uniform (Non-Potts) Priors                           | 31 |
| 8. Conclusion   | 33 |
| Appendix A. Proofs  | 33 |
| A.1. Proof of Proposition 4.2                                 | 33 |
| A.2. Proof of Lemma 4.3                                       | 34 |
| A.3. Proof of Corollary 4.4                                   | 34 |
| A.4. Proof of Lemma 4.6                                       | 34 |
| A.5. Proof of Lemma 4.7                                       | 34 |
| A.6. Proof of Lemma 4.8                                       | 35 |
| A.7. Proof of Lemma 4.9                                       | 35 |
| Acknowledgements  | 36 |
| References  | 36 |

## 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  $\Omega$ . Each vertex  $i \in \mathcal{V}$  indexes the location of a pixel, to which a random variable

$$x_i \in \mathcal{X} = \{\ell_1, \dots, \ell_n\} \quad (1.1)$$

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

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

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 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  $d: \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ , then (1.2) is called the *metric labeling problem* [KT02]. In general, the presence of these latter terms makes image labeling a combinatorially hard task. We refer to [KAH<sup>+</sup>15] 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* [Wer07] (see Section 2.2 for details)

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

Solving the linear program (LP) (1.3) returns a globally optimal *relaxed indicator vector*  $\mu$  whose components take values in  $[0, 1]$ . If  $\mu$  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  $\mu$  has to be rounded to an integral solution in a post-processing 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 [ÅPSS17] 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. The recent paper [BFPS17] reports the application of our scheme to a range of challenging labeling problems of manifold-valued data.

Adopting this starting point, the objectives of the present paper are:

- Show how the approach [ÅPSS17] can be used to efficiently compute 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 [ÅPSS17] so as to overcome the inherent non-smoothness 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

$$E_\tau(\mu_V) = \langle \theta_V, \mu_V \rangle + \sum_{ij \in \mathcal{E}} d_{\theta_{ij}, \tau}(\mu_i, \mu_j), \quad \tau > 0 \quad (1.4)$$

of problem (1.3), where  $d_{\theta_{ij}, \tau}$  denotes the local *smoothed* Wasserstein distance between the discrete label assignment measures  $\mu_i, \mu_j$  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  $E_\tau$  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 linear program (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 [NN87, Ren95] 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 constitute subclasses of objective functions (1.2) as studied in [KZ04], in particular binary submodular functions, that enable to reformulate the labeling problem as maximum-flow problem in an associated network and the application of discrete combinatorial solvers [BVZ01, BK04].

Since the structure of such algorithms inherently limits fine-grained parallel implementations, however, *belief-propagation* and variants [YFW05] 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 [YMW06]. The theoretical deficiencies of basic belief propagation 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 [WJW05, HS10]. We refer to [KAH<sup>+</sup>15] 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  $\bar{\mu}$  to a valid labeling  $x \in \mathcal{X}^m$ , along with suboptimality bounds, were studied in [KT02, CKNZ05]. The problem to infer components  $x_i^*$  of the unknown globally optimal *combinatorial* labeling that minimizes (1.2), through partial optimality and persistency, was studied in [SSK<sup>+</sup>16]. We refer to [Wer07] for the

history and more information about the LP relaxation of labeling problems, and to [WJ08] for connections to discrete probabilistic graphical models from the variational viewpoint.

*Optimal transport* and the *Wasserstein distance* have become a major tool of signal modeling and analysis [KPT<sup>+</sup>17]. Regarding the finite-dimensional formulation in terms of linear programs, the design of efficient algorithms for large-scale problems requires sophisticated techniques [Sch16]. 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 to [Sch90] and [Bru06, Ch. 9] for basic related work and the connection to matrix scaling algorithms and the history. Smoothing of the Wasserstein distance and Sinkhorn’s algorithm has become popular in machine learning due to [Cut13a]. The authors of [Pey15, CP16] comprehensively investigated barycenters and interpolation based on the Wasserstein distance. Our approach to image labeling, in conjunction with the geometric approach of [ÅPSS17], is novel and elaborates the preliminary announcement [ÅHS<sup>+</sup>17].

**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 [ÅPSS17], in particular the so-called assignment manifold, and the general framework of [SHÅ<sup>+</sup>17] 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 the restriction of  $E_\tau$  to the assignment manifold. A key property of this set-up concerns the local polytope constraints that define the feasible set  $\mathcal{L}_G$  of the LP relaxation (1.3): by construction, they are *always* satisfied throughout the resulting iterative 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 ‘Wasserstein distances’ 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 belief propagation 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. To keep this paper at a reasonable length, we merely consider 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 [SHÅ<sup>+</sup>17] for more sophisticated numerical schemes will be explored elsewhere. Furthermore, working out any realistic application is beyond the scope of this paper. Rather, the experimental results are supposed to 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 discussed the results of four types of experiments:

- (1) The interplay between two parameters  $\tau$  and  $\alpha$  that control smoothness of the approximation (1.4) and rounding, respectively, is studied. 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  $\tau$ . Naturally, the effect of rounding has to be stronger to make the iterative process converge to an integral solution. A corresponding choice of  $\alpha$  controls the compromise between quality of integral labelings in terms of the energy (1.4) and speed of

convergence. Fortunately, the upper bound on  $\tau$  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 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  $\alpha$ .
- (3) A labeling problem of the usual size was conducted to confirm and demonstrate that the finding of the preceding points for ‘all’ models on  $\mathcal{K}^3$  also hold in a typical application. A comparison to sequential tree-reweighted message passing (TRWS) [Kol06] which defines the state of the art, and to loopy belief propagation (BP) based on the OpenGM package [ABK12], shows that our approach is on par with these methods regarding the energy level  $E(x)$  of the resulting labeling  $x$ .
- (4) A final experiment based on the graphical model with a pronounced non-uniform (non-Potts) prior demonstrates that our approach is able to perform inference for any given graphical model.

We conclude in Section 8 and relegated to an Appendix a series of proofs that otherwise would interrupt too much the line of reasoning.

## 2. PRELIMINARIES

We introduce basic notation in Section 2.1 and the common linear programming (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 [ÅPSS17] (see Section 3.1), we keep the standard notation in the literature for the former approach and the notation from [ÅPSS17] for the latter one. 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

$$\mathcal{N}(i) = \{j \in \mathcal{V} : i \sim j\} \quad (2.1)$$

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 form *ordered* pairs denoted by  $(i, j) = ij \neq ji = (j, i)$ . 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

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

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  $\langle A, B \rangle = \text{tr}(A^\top B)$ , where  $\text{tr}$  denotes the trace of a matrix, i.e.  $\text{tr}(A^\top B) = \sum_{i \in [m]} \langle A_i, B_i \rangle = \sum_{j \in [n]} \langle A^j, B^j \rangle = \sum_{i \in [m], j \in [n]} A_{ij} B_{ij}$ .

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  $\mu_{ij}$  by  $\mu_i$  and  $\mu_j$ , respectively, and the linear mapping for extracting them by

$$\mathcal{A}: \mathbb{R}^{n \times n} \rightarrow \mathbb{R}^{2n}, \quad \mu_{ij} \mapsto \mathcal{A}\mu_{ij} = \begin{pmatrix} \mu_{ij} \mathbb{1}_n \\ \mu_{ij}^\top \mathbb{1}_n \end{pmatrix} = \begin{pmatrix} \mu_i \\ \mu_j \end{pmatrix}. \quad (2.3a)$$

Its transpose is given by

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

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

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

for the *componentwise* multiplication and subdivision.

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. [RW09, Section 11.A])

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

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

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

and

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

denotes the orthogonal projection onto  $C$ . The shorthand “s.t.” means: “subject to” in connection to the specification of some constraints.

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

$$\log \exp_\varepsilon(x) = \varepsilon \log \left( \sum_{i \in [n]} e^{\frac{x_i}{\varepsilon}} \right). \quad (2.8a)$$

It uniformly approximates the function  $\operatorname{vecmax} \in \mathcal{F}_0$  [RW09, Ex. 1.30], i.e.

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

We will use the following basic result from convex analysis (cf., e.g. [RW09, Ch. 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*

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



and

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

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

**Theorem 2.2** ([Dan66, Roc91]). *Let  $f(z) = \max_{w \in W} g(z, w)$ , where  $W$  is compact and the function  $g$  is differentiable and  $\nabla_z g(z, w)$  is continuously depending on  $(z, w)$ . If in addition  $g(z, w)$  is convex in  $z$ , and if  $\bar{z}$  is a point such that  $\arg \max_{w \in W} g(\bar{z}, w) = \{\bar{w}\}$ , then  $f$  is differentiable at  $\bar{z}$  with*

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

**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*

$$\theta_i = (\theta_i(\ell_1), \dots, \theta_i(\ell_n)) \in \mathbb{R}^n, \quad \theta_{ij} = (\theta_{ij}(\ell_k, \ell_l))_{\ell_k, \ell_l \in \mathcal{X}} \in \mathbb{R}^{n \times n} \quad (2.11)$$

which merely encode the values of the discrete objective function (1.2):  $\theta_i(\ell_j) = E_i(\ell_j)$ ,  $\theta_{ij}(\ell_k, \ell_l) = E_{ij}(\ell_k, \ell_l)$ . 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

$$\theta := (\theta_{\mathcal{V}}, \theta_{\mathcal{E}}), \quad \theta_{\mathcal{V}} := (\dots, \theta_i, \dots), \quad i \in \mathcal{V}, \quad \theta_{\mathcal{E}} := (\dots, \theta_{ij}, \dots), \quad ij \in \mathcal{E}, \quad (2.12)$$

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

$$\mu_i = (\mu_i(\ell_1), \dots, \mu_i(\ell_n)) \in \{0, 1\}^n, \quad \mu_{ij} = (\mu_{ij}(\ell_k, \ell_l))_{\ell_k, \ell_l \in \mathcal{X}} \in \{0, 1\}^{n \times n} \quad (2.13)$$

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

$$\mu = (\mu_{\mathcal{V}}, \mu_{\mathcal{E}}), \quad \mu_{\mathcal{V}} := (\dots, \mu_i, \dots), \quad i \in \mathcal{V}, \quad \mu_{\mathcal{E}} := (\dots, \mu_{ij}, \dots), \quad ij \in \mathcal{E}. \quad (2.14)$$

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

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

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

As a result, the linear programming relaxation (1.3) of (1.2) reads more explicitly

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

where the so-called *local polytope*  $\mathcal{L}_{\mathcal{G}}$  is the set of all vectors  $\mu$  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 [ÅPSS17], to be sketched in Section 3.2. Section 3.3 summarizes the general framework of [SHÅ<sup>+</sup>17] 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  $\mathcal{S} := \text{rint}(\Delta_n)$ , given by  $\mathcal{S} = \{p \in \mathbb{R}_{++}^n : \langle \mathbb{1}, p \rangle = 1\}$ , is a  $n - 1$  dimensional smooth manifold with constant tangent space

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

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

$$P_T : \mathbb{R}^n \rightarrow T, \quad x \mapsto P_T(x) = x - \frac{1}{n} \langle \mathbb{1}, x \rangle \mathbb{1} = \Pi x, \quad \Pi := I - \frac{1}{n} \mathbb{1} \mathbb{1}^\top, \quad (3.2)$$

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

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

where all operations are applied componentwise, i.e.  $\frac{u}{\sqrt{p}} = (\frac{u_1}{\sqrt{p_1}}, \dots, \frac{u_n}{\sqrt{p_n}})$ . In this setting, there is an important map, called the *lifting map* (cf. [ÅPSS17, Def. 4]), defined as

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

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

$$L := \tilde{L}|_T : T \rightarrow \mathcal{S}, \quad \tilde{L} = L \circ P_T. \quad (3.5)$$

This restricted lifting map  $L$  is also a local first order approximation to the exponential map of the Riemannian manifold  $\mathcal{S}$  (cf [ÅPSS17, Prop. 3]), with the inverse mapping given by

$$L_p^{-1} : \mathcal{S} \rightarrow T, \quad q \mapsto L_p^{-1}(q) := P_T \left( \log \frac{q}{p} \right) = \Pi \log \frac{q}{p}. \quad (3.6)$$

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

$$T_W \mathcal{W} = \prod_{i \in [m]} T = \{V \in \mathbb{R}^{m \times n} : V\mathbb{1} = 0\} =: T^m \quad \text{at } W \in \mathcal{W}. \quad (3.7)$$

The Fisher-Rao product metric reads

$$\langle U, V \rangle_W = \sum_{i \in [m]} \left\langle \frac{U_i}{\sqrt{W_i}}, \frac{V_i}{\sqrt{W_i}} \right\rangle \quad \text{at } W \in \mathcal{W}, \quad U, V \in T^m. \quad (3.8)$$

The orthogonal decomposition of  $T$  induces the orthogonal decomposition

$$\mathbb{R}^{m \times n} = T^m \oplus \{\lambda \mathbb{1}_n^\top \in \mathbb{R}^{m \times n} : \lambda \in \mathbb{R}^m\} \quad (3.9)$$

together with the orthogonal projection

$$P_{T^m} : \mathbb{R}^{m \times n} \rightarrow T^m, \quad X \mapsto P_{T^m}(X) = X\Pi, \quad (3.10)$$

where  $\Pi$  is the projection matrix from (3.2) above. 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, the restricted lifting map and its inverse are naturally extended to

$$\tilde{L}_W : \mathbb{R}^{m \times n} \rightarrow \mathcal{W}, \quad L_W : T^m \rightarrow \mathcal{W} \quad \text{and} \quad L_W^{-1} : \mathcal{W} \rightarrow T^m \quad (3.11)$$



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

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

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 [ÅPSS17] 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  $\mathcal{F}$ ,

$$f: \mathcal{V} = [m] \rightarrow \mathcal{F}, \quad i \mapsto f_i. \quad (3.13)$$

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,

$$d: \mathcal{F} \times \mathcal{X} \rightarrow \mathbb{R}, \quad (3.14)$$

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*

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

that  $\ell_j$  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.  $\mu$ ). Each row vector  $W_i$ ,  $i \in [m]$  plays the role of a corresponding vector  $\mu_i$  of the basic LP relaxation as defined by (2.13), with relaxed domain due to (2.15). Unlike  $\mu_i$ , 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) are gathered by the *distance matrix*

$$D \in \mathbb{R}^{m \times n}, \quad D_{ij} = d(f_i, \ell_j) \quad (3.16)$$

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

$$L = \tilde{L}_W \left( -\frac{1}{\rho} D \right) = L_W \left( -\frac{1}{\rho} D \Pi \right), \quad \rho > 0 \quad (3.17)$$

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

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

is defined as a local geometric average of assignment vectors at neighboring nodes, i.e. the  $i$ -th row  $S_i$  is defined to be the Riemannian mean (cf. [ÅPSS17, Def. 2])

$$S_i = \text{mean}_{\mathcal{S}} \{L_j\}_{j \in \bar{\mathcal{N}}(i)} \quad (3.19)$$

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

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

$$\max_{W \in \mathcal{W}} J(W), \quad J(W) := \langle W, S(W) \rangle \quad (3.20)$$

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)

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

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

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

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. [ÅPSS17, chap. 3.3]), the following multiplicative update scheme is obtained

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

**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 [SHÅ<sup>+</sup>17] for integrating a Riemannian gradient flow of an arbitrary function  $J: \mathcal{W} \rightarrow \mathbb{R}$  defined on the assignment manifold.

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

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

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

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

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

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 [SHÅ<sup>+</sup>17] is to pull the optimization problem back onto the tangent space  $T^m = T_C \mathcal{W}$  by setting

$$\bar{J} := J \circ L_C, \quad (3.26)$$

using the diffeomorphism  $L_C: T^m \rightarrow \mathcal{W}$  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  $\bar{J}: T^m \rightarrow \mathbb{R}$  at  $V \in T^m$  is given by [SHÅ<sup>+</sup>17, sec. 3]

$$\nabla_{T^m} \bar{J}(V) = \nabla J(L_C(V)) \in T^m, \quad (3.27)$$

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

$$W(t) = L_C(V(t)), \quad (3.28)$$

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

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

Choosing the explicit Euler method for solving this gradient flow problem on the vector space  $T^m$ , results in the numerical update scheme

$$V^{(k+1)} = V^{(k)} - h \nabla J(L_C(V^{(k)})), \quad V^{(0)} = 0, \quad (3.30)$$

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

$$W^{(k+1)} = \frac{W^{(k)} \cdot e^{-h\nabla J(W^{(k)})}}{\langle W^{(k)}, e^{-h\nabla J(W^{(k)})} \rangle}, \quad W^{(0)} = C. \quad (3.31)$$

#### 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 belief propagation, 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  $\theta$  defined by (2.12), this is straightforward to do for the *unary* terms  $\theta_i$  that typically measure a local distance to observed data. But this is less obvious for the *pairwise* terms  $\theta_{ij}$  that do not have a direct counterpart in the geometric labeling approach.

The following Lemma explains why the local Wasserstein distances

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

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  $\theta_{ij}$ .

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

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

*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.

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

□

In order to conform to our smooth geometric setting, we regularize the convex but non-smooth (piecewise-linear) local Wasserstein distances (4.1) with a general convex *smoothing function*  $F_{\tau}$ ,

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

with smoothing parameter  $\tau$ .

**Remark 4.1** (role of the smoothing). The influence of the smoothing parameter  $\tau$  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  $\tau$  weaken the nonlinear relation (4.3) and considerably speed up the convergence of numerical algorithm for iterative label assignment.

**Remark 4.2** (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, like loopy belief propagation, where these constraints are gradually enforced during the iteration and are guaranteed to hold only *after* convergence of the entire iteration process.

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

- (i) inherent smoothness and
- (ii) anytime validity of the local polytope constraints.

Based on Lemma 4.1 and the regularized local Wasserstein distances (4.3), we study in this paper 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  $F_\tau$ , we inspect the smoothed local Wasserstein distance (4.1) 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.

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

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

with the marginal vector  $\mu$  playing the role of  $\begin{pmatrix} \mu_i \\ \mu_j \end{pmatrix}$  in (2.15). The local (non-smooth) Wasserstein distance (4.1) then reads, for any edge  $ij \in \mathcal{E}$ ,

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

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

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

The corresponding dual LP of (4.6) is given by

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

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

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

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

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

with the conjugate function  $G_\tau^*$  of

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

Suitable candidates of functions  $G_\tau$  for smoothing  $d_\Theta$  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

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

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

**4.2. Energy Gradient  $\nabla E_\tau$ .** The pairwise model parameters  $\theta_{ij}$  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:  $d_{\theta_{ij}, \tau}(W_i, W_j) \neq d_{\theta_{ji}, \tau}(W_j, W_i)$ . In order to compute the Euclidean gradient  $\nabla E_\tau$  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

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

The following proposition specifies the gradient  $\nabla E_\tau$  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 4.5).

**Proposition 4.2** (objective function gradient). *Suppose the edges  $\mathcal{E}$  have an arbitrary fixed orientation. Then the Euclidean gradient of the objective function  $E_\tau: \mathcal{W} \rightarrow \mathbb{R}$  due to (1.4), at  $W \in \mathcal{W}$ , is the matrix  $\nabla E_\tau(W) \in T^m$  whose  $i$ -th row is given by*

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

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

$$d_{\theta_{ij}, \tau}(\cdot, W_j): \mathcal{S} \rightarrow \mathbb{R}, \quad d_{\theta_{ji}, \tau}(W_j, \cdot): \mathcal{S} \rightarrow \mathbb{R}. \quad (4.14)$$

*Proof.* Appendix A.1. □

We now consider after a preparatory Lemma the specific case that all pairwise model parameters  $\theta_{ij} = \theta_{ij}^\top$  are symmetric (Corollary 4.4). 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 4.3.** *Suppose the convex smoothing function  $F_\tau$  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*

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

*Proof.* Appendix A.2. □

As a consequence of Lemma 4.3, if all pairwise model parameters  $\theta_{ij}$  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

$$E_\tau(W) = \sum_{i \in V} \left( \langle \theta_i, W_i \rangle + \frac{1}{2} \sum_{j \in \mathcal{N}(i)} d_{\theta_{ij}, \tau}(W_i, W_j) \right) \quad (4.16)$$

and reformulate Proposition 4.2 accordingly.

**Corollary 4.4** (objective function gradient: symmetric case). *Suppose  $F_\tau(T) = F_\tau(T^\top)$  for all  $T \in [0, 1]^{n \times n}$  and  $\theta_{ij}$  is symmetric for all  $ij \in \mathcal{E}$ . Then the  $i$ -th row of the Euclidean gradient  $\nabla E_\tau$  is given by*

$$\nabla_i E_\tau(W) = P_T(\theta_i) + \sum_{j \in \mathcal{N}(i)} \nabla_1 d_{\theta_{ij}, \tau}(W_i, W_j). \quad (4.17)$$

*Proof.* Appendix A.3. □

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

$$g(\mu, \nu) = \langle \mu, \nu \rangle - G_\tau^*(\mathcal{A}^\top \nu - \Theta) \quad (4.18)$$

*denote the dual objective function (4.23). Then the smoothed Wasserstein distance  $d_{\Theta, \tau}: \mathcal{S} \times \mathcal{S} \rightarrow \mathbb{R}$  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*

$$\nabla d_{\Theta, \tau}(p) = \nabla d_{\Theta, \tau}(p_1, p_2) = \bar{\nu}_T := P_{T \times T}(\bar{\nu}) = \begin{pmatrix} P_T(\bar{\nu}_1) \\ P_T(\bar{\nu}_2) \end{pmatrix}, \quad (4.19)$$

where

$$\bar{\nu} = \begin{pmatrix} \bar{\nu}_1 \\ \bar{\nu}_2 \end{pmatrix} \in \operatorname{argmax}_{\nu \in \mathbb{R}^{2n}} g(p, \nu). \quad (4.20)$$

The proof follows below after some preparatory Lemmas.

**Lemma 4.6.** *Let*

$$G_\tau(M) = F_\tau(M) + \delta_{\mathbb{R}_+^{n \times n}}(M) \quad (4.21)$$

*with the convex smoothing function  $F_\tau$  of eq. (4.3), and assume the conjugate function  $G_\tau^*$  is continuously differentiable. Then the dual problem of*

$$\min_{M \in \Pi(\mu_1, \mu_2)} \{ \langle \Theta, M \rangle + F_\tau(M) \} \quad (4.22)$$

*is given by*

$$\max_{\nu_1, \nu_2} \{ \langle \mu, \nu \rangle - G_\tau^*(\mathcal{A}^\top \nu - \Theta) \}. \quad (4.23)$$

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

$$\bar{M} = \nabla G_\tau^*(\mathcal{A}^\top \bar{\nu} - \Theta), \quad \mathcal{A}^\top \bar{\nu} - \Theta \in \partial G_\tau(\bar{M}) \quad (4.24a)$$

*together with the affine constraint*

$$\mathcal{A}\bar{M} = \mu. \quad (4.24b)$$

*Proof.* Appendix A.4. □

**Remark 4.3** (smoothness of  $G_\tau^*$ ). The *smoothness* assumption with respect to  $G_\tau^*$  enables to compute conveniently the gradient of the smoothed Wasserstein distance  $d_{\Theta, \tau}$ . It corresponds to a *convexity* assumption on  $G_\tau$ . These aspects are further discussed in Section 6.2 as well.

**Remark 4.4** (strong duality). The condition of strong duality made by Lemma 4.6 is crucial for what follows. This condition will be satisfied later on when working in a *geometric* setting with local measures  $M, \mu_1, \mu_2$  with *full* support, as introduced in Section 3.1.



**Lemma 4.7.** *Let the linear mapping  $\mathcal{A}^\top$  be defined by (2.3b). Then*

$$\mathcal{N}(\mathcal{A}^\top) = \left\{ \alpha \begin{pmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{pmatrix} \in \mathbb{R}^{2n} : \alpha \in \mathbb{R} \right\} \quad \text{and} \quad \mathcal{N}(\mathcal{A}^\top)^\perp = \left\{ x \in \mathbb{R}^{2n} : \left\langle x, \begin{pmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{pmatrix} \right\rangle = 0 \right\}. \quad (4.25)$$

*Proof.* Appendix A.5. □

The following Lemma characterizes the set of optimal dual solutions to problem (4.23).

**Lemma 4.8.** *Let the function  $G_\tau^*$  of the dual objective function (4.23) resp. (4.18) be continuously differentiable and strictly convex, and let  $p \in \mathbb{R}_{++}^{2n}$ . Then the set of optimal dual solutions has the form*

$$\operatorname{argmax}_{\nu \in \mathbb{R}^{2n}} g(p, \nu) = \begin{cases} \{\bar{\nu}\}, & \text{if } \langle p, \begin{pmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{pmatrix} \rangle \neq 0, \\ \bar{\nu} + \mathcal{N}(\mathcal{A}^\top), & \text{if } \langle p, \begin{pmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{pmatrix} \rangle = 0. \end{cases} \quad (4.26)$$

*Proof.* Appendix A.6. □

We next clarify the *attainment* of optimal dual solutions due to Lemma 4.8.

**Lemma 4.9.** *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  $\langle p, \begin{pmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{pmatrix} \rangle = 0$ , we have*

$$\operatorname{argmax}_{\nu_{\mathcal{R}} \in \mathcal{R}(\mathcal{A})} g(p, \nu_{\mathcal{R}}) = \{\bar{\nu}_{\mathcal{R}}\}, \quad \bar{\nu}_{\mathcal{R}} = P_{\mathcal{R}(\mathcal{A})}(\bar{\nu}) \quad \text{for any} \quad \bar{\nu} \in \operatorname{argmax}_{\nu \in \mathbb{R}^{2n}} g(p, \nu), \quad (4.27a)$$

$$g(p, \bar{\nu}_{\mathcal{R}}) = \max_{\nu_{\mathcal{R}} \in \mathcal{R}(\mathcal{A})} g(p, \nu_{\mathcal{R}}) = \max_{\nu \in \mathbb{R}^{2n}} g(p, \nu), \quad (4.27b)$$

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

*Proof.* Appendix A.7. □

We are now in a position to prove Theorem 4.5.

*Proof of Theorem 4.5.* In the following, we proceed in three steps: 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

$$P_{T \times T}(\nu_{\mathcal{R}}) = \nu_{\mathcal{R}} = P_{T \times T}(\nu). \quad (4.28)$$

At first, we show  $T \times T \subseteq \mathcal{R}(\mathcal{A})$ . For this, take an arbitrary  $v = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \in T \times T$ . 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  $\langle v, \begin{pmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{pmatrix} \rangle = 0$ , which according to Lemma 4.7 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 (4.28) follows from

$$P_{T \times T}(\nu) - P_{T \times T}(\nu_{\mathcal{R}}) = P_{T \times T}(\nu - \nu_{\mathcal{R}}) = P_{T \times T}(\nu_{\mathcal{N}}) = 0. \quad (4.29)$$

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

$$\psi: \mathbb{R}^{2(n-1)} \rightarrow (T \times T) + b, \quad x \mapsto \psi(x) := Bx + b, \quad Bx = \sum_{i=1}^{2(n-1)} x_i v_i. \quad (4.30)$$

Because  $\mathcal{S} \times \mathcal{S}$  is an open subset of  $(T \times T) + b$  and  $\psi$  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

$$\phi := \psi|_U : U \rightarrow \mathcal{S} \times \mathcal{S} \quad (4.31)$$

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

$$P_{T \times T} = BB^\top. \quad (4.32)$$

(3) Using  $\phi$  given by (4.31), we obtain the coordinate representations

$$\bar{d}_{\Theta, \tau} := d_{\Theta, \tau} \circ \phi, \quad \bar{g}(x, \nu) := g(\phi(x), \nu) \quad (4.33)$$

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 (4.22) and (4.23), 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 4.9 to

$$\bar{g}(x_p, \bar{\nu}_{\mathcal{R}}) = \max_{\nu_{\mathcal{R}} \in \mathcal{R}(\mathcal{A})} \bar{g}(x_p, \nu_{\mathcal{R}}) = \bar{g}(x_p, \bar{\nu}) = \max_{\nu \in \mathbb{R}^{2n}} \bar{g}(x_p, \nu) = \bar{d}_{\Theta, \tau}(x_p), \quad (4.34)$$

with unique maximizer  $\bar{\nu}_{\mathcal{R}} = P_{\mathcal{R}(\mathcal{A})}(\bar{\nu})$ . Let  $\mathbb{B}_\delta \subset \mathcal{R}(\mathcal{A})$  be a compact neighborhood of  $\bar{\nu}_{\mathcal{R}}$ . Then (4.34) remains valid after restricting  $\mathcal{R}(\mathcal{A})$  to  $\mathbb{B}_\delta$ . Because  $g$  given by (4.18) is linear in the first argument and the mapping  $\phi$  is affine, the function  $\bar{g}$  is convex in the first argument and differentiable, hence satisfies the assumptions of Theorem 2.2.

In order to compute the gradient  $\nabla_x \bar{g}(x, \nu_{\mathcal{R}})$ , it suffices to consider the first term  $\langle \phi(x), \nu_{\mathcal{R}} \rangle$  of  $\bar{g}$ , which only depends on  $x$ . Using (4.31), we have

$$\langle \phi(x), \nu_{\mathcal{R}} \rangle = \langle Bx + b, \nu_{\mathcal{R}} \rangle = \langle x, B^\top \nu_{\mathcal{R}} \rangle + \langle b, \nu_{\mathcal{R}} \rangle. \quad (4.35)$$

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

$$\nabla \bar{d}_{\Theta, \tau}(x_p) = \nabla_x \bar{g}(x_p, \bar{\nu}_{\mathcal{R}}) = B^\top \bar{\nu}_{\mathcal{R}}. \quad (4.36)$$

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

$$\nabla d_{\Theta, \tau}(p) = B \nabla \bar{d}_{\Theta, \tau}(x_p) = BB^\top \bar{\nu}_{\mathcal{R}} \stackrel{(4.28)}{=} P_{T \times T}(\bar{\nu}_{\mathcal{R}}) \stackrel{(4.28)}{=} \bar{\nu}_T, \quad (4.37)$$

which proves (4.19). □

## 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, 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, worked out in Section 5.1, for determining a labeling  $x$  with low energy  $E(x)$  is to combine minimization of the convex relaxation (1.3) and non-convex 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 due to Section 3.1, and by combining numerical integration of the corresponding Riemannian gradient flow due to Section 3.3 with the assignment mechanism suggested by [ÅPSS17] due to Section 3.2.

Section 5.2 complements our preliminary observations stated as Remarks 4.1 and 4.2, in order to highlight the essential properties of this process as a novel way of ‘belief propagation’ using dually computing gradients of local Wasserstein distances, that we call *Wasserstein messages*.

**5.1. Smooth Intergration of Minimizing and Rounding on the Assignment Manifold.** We recall how regularization is performed by the assignment approach of [ÅPSS17]: 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 – eq. (3.19) resp. (3.22).

Given a graphical model in terms of an energy function (1.2), regularization is already *defined* by the pairwise model parameters  $E_{ij}(\ell_i, \ell_j)$  resp.  $\theta_{ij}(\ell_i, \ell_j)$ , so that evaluating the gradient of the regularized objective function (1.4) *implies* averaging over spatial neighborhoods, as eq. (4.13) clearly displays. Taking additionally into account the simplest (explicit Euler) update rule (3.31) for geometric integration of Riemannian gradient flows on the assignment manifold, a natural definition of the similarity matrix that consistently incorporates the graphical model into the geometric approach of [ÅPSS17], is

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

where  $h$  is a stepsize 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 [ÅPSS17] leads to the update of the assignment matrix

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

These two interleaved update steps represent two objectives: (i) minimize the function  $E_\tau$  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

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

which suggests to control more flexibly the latter rounding mechanism by a *rounding parameter*  $\alpha$  and the update rule

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

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

**Proposition 5.1.** *Let  $E_\tau$  be given by (1.4) and denote the entropy of the assignment matrix  $W$  by*

$$H(W) = -\langle W, \log W \rangle. \quad (5.5)$$

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

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

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

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

where the  $i$ -th 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  $\gamma: (-\varepsilon, \varepsilon) \rightarrow \mathcal{W}$  with  $\gamma(0) = W$  and  $\dot{\gamma}(0) = X$ . Then

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

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

$$\langle \nabla H(W), X \rangle = \frac{d}{dt} H(\gamma(t)) \Big|_{t=0} = \langle -P_T \log(W), X \rangle. \quad (5.9)$$

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

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

Substitution into (5.7) gives

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

and in turn the update

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

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

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

which is (5.4).  $\square$

**Remark 5.1** (continuous DC programming). Proposition 5.1 and (5.6) admit to interpret the update rule (5.4) as a *continuous difference of convex (DC) programming* strategy. Unlike the established DC approach [PDHA97, PDHA98], however, which 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, our update rule (5.4) differs in two essential ways: *geometric optimization* by numerically integrating the Riemannian gradient flow *tightly interleaves with rounding* to an integral solution.

**5.2. Wasserstein Messages.** We get back to the informal discussion of *belief propagation* in Section 1.2 in order to highlight properties of our approach (1.4) from this viewpoint. We first sketch belief propagation and the origin of corresponding *messages*, and refer to [YFW05, WJ08] for background and more details.

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

$$\min_{\mu \in \mathcal{L}_G} \langle \theta, \mu \rangle = \min_{\mu} \langle \theta, \mu \rangle \quad \text{subject to} \quad A\mu = b, \mu \geq 0, \quad (5.13)$$

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

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

with dual (multiplier) variables

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

corresponding to the affine primal constraints. In order to obtain a condition that relates optimal vectors  $\mu$  and  $\nu$  without subdifferentials that are caused by the non-smoothness of these LPs, one considers the *smoothed* primal convex problem

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

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

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

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

$$L(u, v) = \langle \theta, \mu \rangle - H(\mu) + \langle \nu, A\nu - b \rangle, \quad (5.18)$$

yields the relations connecting  $\mu$  and  $\nu$ ,

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

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

$x_i, x_j \in \mathcal{X}$ ,  $ij \in \mathcal{E}$ , 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 to eliminate the left-hand side of (5.19) to obtain after some algebra the fixed point equations

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

solely in terms of the *dual* variables, commonly called *sum-product algorithm* or *loopy belief propagation* by *message passing*. Repeating this derivation, after weighting the entropy function  $H(\mu)$  of (5.18) by  $\varepsilon$  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 *max-product algorithm* in the literature.

In order to highlight subsequently major differences to our approach, we make the following **key observations**:

- (1) **Local non-convexity.** The negative  $-H(\mu)$  of the so-called *Bethe entropy* function  $H(\mu)$  is *non-convex* in general for graphs  $\mathcal{G}$  with cycles [WJ08, 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 limit of a *non-convex* function, due to (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 to choose  $\varepsilon$  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 unfavourable stationary points, due to the non-convexity according to (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 **Wasserstein messages** these local gradients that are ‘passed along edges’. Similarly to (5.20), each such message is given by *dual* variables through (4.19), that solve the regularized *local dual LPs* (4.18). As a consequence, local marginalization constraints are *always* satisfied, throughout the iterative process.

In addition, we make the following **observations** in correspondence to the points (1)-(4) above:

- (1) **Local convexity.** Wasserstein messages of (4.13) are defined by local *convex* programs (4.18). This contrasts with loopy belief propagation and holds true for any pairwise model parameters  $\theta_{ij}$  of the prior of the graphical model and the corresponding coupling of  $\mu_i$  and  $\mu_j$ . This removes spurious minima introduced through non-convex entropy approximations.
- (2) **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.
- (3) **Smoothness and weak nonlinearity.** The role of the smoothing parameter  $\tau$  of (1.4) *differs* from the role of the smoothing parameter  $\varepsilon$  of (5.16). While the latter has to be chosen quite close to 0 so as to achieve rounding at all,  $\tau$  merely mollifies the dual local problems (4.18) and hence should be chosen small, but may be considerably larger than  $\varepsilon$ . In particular, this does not impair rounding due to (2), which happens due to the *global* flow which is *smoothly* driven by the Wasserstein messages. This *decoupling* of smoothing and rounding enables to numerically compute labelings more efficiently. The results reported in Section 7 demonstrate this fact.
- (4) **Valid constraints.** By construction, computation of the Wasserstein messages enforces all local marginalization constraints *throughout* the iteration. This is in sharp contrast to belief propagation 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 given by (5.4),

$$W_i^{(k+1)} = \frac{(W_i^{(k)})^{1+\alpha} \cdot e^{-h\nabla_i E_\tau(W^{(k)})}}{\langle (W_i^{(k)})^{1+\alpha}, e^{-h\nabla_i E_\tau(W^{(k)})} \rangle}, \quad W_i^{(0)} = \frac{1}{n} \mathbb{1}_n, \quad i \in \mathcal{V} \quad (6.1)$$

where  $\alpha \geq 0$  is the *rounding* parameter,  $h > 0$  the step-size and  $\tau$  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  $F_\tau$  in eq. (4.3). Our criterion for convergence and terminating the iterative process (6.1) of label assignment is specified in Section 6.3.

**6.1. Assignment Normalization.** The rounding mechanism addressed by Prop. 5.1 and Remark 5.1 will be effective if  $\alpha_h$  in (5.6) is chosen large enough so as to compensate the influence of the function  $F_\tau$  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 4.4. Since there is a difference between mathematical and numerical positivity, we avoid numerical problems by adopting the normalization strategy of [ÅPSS17]. After each iteration, we check all  $W_i$  and whenever an entry drops below  $\varepsilon = 10^{-10}$ , we rectify  $W_i$  by

$$W_i \leftarrow \frac{1}{\langle \mathbb{1}, \tilde{W}_i \rangle} \tilde{W}_i, \quad \tilde{W}_i = W_i - \min_{j=1,\dots,n} \{W_{ij}\} + \varepsilon, \quad \varepsilon = 10^{-10}. \quad (6.2)$$



Thus, the constant  $\varepsilon$  plays the role of 0 in our implementation. Our numerical experiments showed that this operation removed any numerical issues without affecting convergence.

**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

$$d_{\Theta, \tau}(\mu_1, \mu_2) = \min_{M \in \mathbb{R}^{n \times n}} \langle \Theta, M \rangle - \tau H(M) \quad \text{s.t.} \quad \mathcal{A}M = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \quad M \geq 0, \quad (6.3)$$

with the entropy function

$$H(M) = - \sum_{i,j} M_{ij} \log M_{ij}. \quad (6.4)$$

As shown in Section 4.3 and according to Theorem 4.5, the gradients of (6.3) are the maximizer of the corresponding dual problem. Using the notation (4.4), the dual problem of (6.3) reads

$$\max_{\nu \in \mathbb{R}^{2n}} \langle \mu, \nu \rangle - \tau \sum_{k,l} \exp \left[ \frac{1}{\tau} \left( \mathcal{A}^\top \nu - \Theta \right)_{kl} \right]. \quad (6.5)$$

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* [NN87, Ter96], 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 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 favours the formulation (6.5).

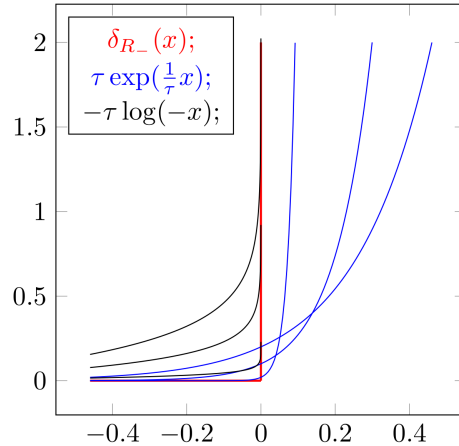


FIGURE 6.1. Approximations of the indicator function  $\delta_{\mathbb{R}_-}$  of the nonpositive orthant. The log-barrier function (black curves) strictly rule out violations of the constraints but induce a bias towards interior points. Our formulation (blue curves) is less biased and reasonable approximates the  $\delta$ -function (red curve) depending on the smoothing parameter  $\tau$ .

We now make explicit how the local Wasserstein gradients (4.19) are computed based on the formulation (6.3) and examine numerical aspects depending on the smoothing parameter  $\tau$ . It is well known that doubly stochastic matrices as solutions of convex programs like (6.3) can be computed by iterative matrix scaling [Sin64, Sch90], [Bru06, ch. 9]. This has been made popular in the field of machine learning by [Cut13b].

The optimality condition (4.24) takes the form

$$\bar{M} = \exp \left[ \frac{1}{\tau} \left( \mathcal{A}^\top \bar{\nu} - \Theta \right) \right], \quad (6.6)$$

and rearranging makes explicit the connection to matrix scaling:

$$\begin{aligned} \bar{M} &= \exp \left[ \frac{1}{\tau} \left( \mathcal{A}^\top \bar{\nu} - \Theta \right) \right] \stackrel{(2.3b)}{=} \exp \left[ \frac{1}{\tau} \left( \bar{\nu}_1 \mathbb{1}_n^\top + \mathbb{1}_n \bar{\nu}_2^\top - \Theta \right) \right] \\ &= \left( \exp(\frac{\bar{\nu}_1}{\tau}) \exp(\frac{\bar{\nu}_2}{\tau})^\top \right) \cdot \exp \left( -\frac{1}{\tau} \Theta \right) = \text{Diag} \left( \exp(\frac{\bar{\nu}_1}{\tau}) \right) \exp \left( -\frac{1}{\tau} \Theta \right) \text{Diag} \left( \exp(\frac{\bar{\nu}_2}{\tau}) \right), \end{aligned} \quad (6.7)$$

where  $\text{Diag}(\cdot)$  denotes the diagonal matrix with the argument vector as entries. For given marginals  $\mu = (\mu_1, \mu_2)$  due to (6.3) and with the shorthand  $K = \exp \left( -\frac{1}{\tau} \Theta \right)$ , the optimal dual variables  $\bar{\nu} = (\bar{\nu}_1, \bar{\nu}_2)$  can be determined by the Sinkhorn's iterative algorithm [Sin64], up to a common multiplicative constant. Specifically, we have

**Lemma 6.1** ([Cut13b, Lemma 2]). *For  $\tau > 0$ , the solution  $\bar{M}$  of (6.3) is unique and has the form  $\bar{M} = \text{diag}(v_1) K \text{diag}(v_2)$ , where the two vectors  $v_1, v_2 \in \mathbb{R}^n$  are uniquely defined up to a multiplicative factor.*

Accordingly, by setting

$$v_1 := \exp(\frac{\nu_1}{\tau}), \quad v_2 := \exp(\frac{\nu_2}{\tau}), \quad (6.8)$$

the corresponding fixed point iterations read

$$v_1^{(k+1)} = \frac{\mu_1}{K \left( \frac{\mu_2}{K^\top v_1^{(k)}} \right)}, \quad v_2^{(k+1)} = \frac{\mu_2}{K^\top \left( \frac{\mu_1}{K v_2^{(k)}} \right)}, \quad (6.9)$$

which are iterated until the change between consecutive iterates is small enough. Denoting the iterates after convergence by  $\bar{v}_1, \bar{v}_2$ , resubstitution into (6.8) determines the optimal dual variables

$$\bar{\nu}_1 = \tau \log \bar{v}_1, \quad \bar{\nu}_2 = \tau \log \bar{v}_2. \quad (6.10)$$

Due to Theorem 4.5, the local Wasserstein gradients then finally are given by

$$\nabla d_{\Theta, \tau}(\mu_1, \mu_2) = \begin{pmatrix} P_T(\bar{\nu}_1) \\ P_T(\bar{\nu}_2) \end{pmatrix}, \quad (6.11)$$

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 [Kni08], the numbers of iterations required in practice significantly depends on the smoothing parameter  $\tau$ . In addition, for smaller values of  $\tau$ , an entry of the matrix  $K = \exp \left( -\frac{1}{\tau} \Theta \right)$  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 not encounter such problems 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 for the energy function to be minimized. Figures 6.2 and 6.3 demonstrate that this indeed holds for relatively large values of  $\tau$ , e.g.  $\tau \in \{\frac{1}{5}, \frac{1}{10}, \frac{1}{15}\}$ , no matter if the number of labels is  $n = 10$  or  $n = 1000$ .

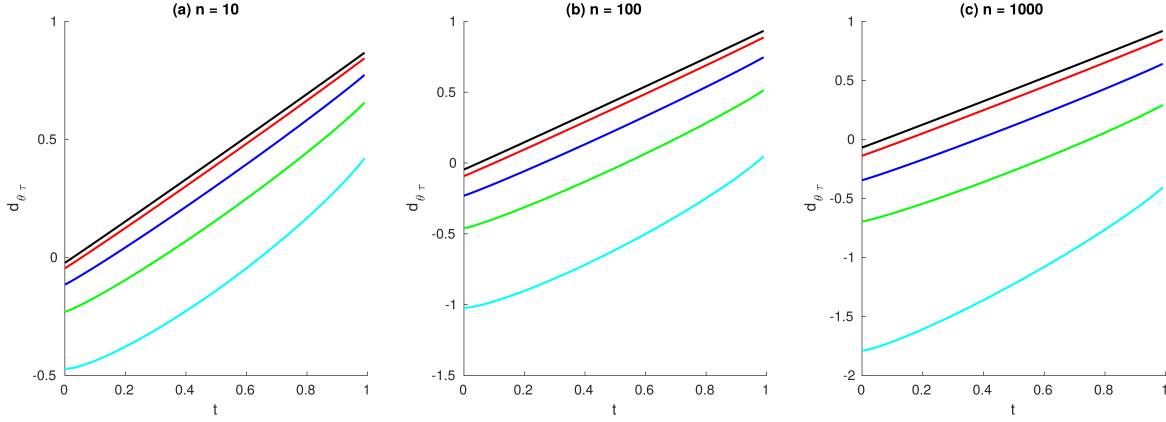


FIGURE 6.2. The plots show the entropy-regularized Wasserstein distance  $d_{\Theta, \tau}(c, \gamma(t))$  for varying parameter  $\tau$  and increasing number of labels  $n$ . Here,  $\gamma(t) = t(e_1 - c) + c \in \Delta_n$  with  $t \in [0, 1]$  is the line segment from the barycenter  $c = \frac{1}{n}\mathbb{1}$  to the vertex  $e_1$  on the simplex  $\Delta_n$ . The cost matrix  $\Theta$  is given by the Potts regularizer. In all three plots the parameter  $\tau$  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 that really matters, is approximated already for larger values of  $\tau$  quite well and uniformly for small up to large numbers  $n$  of labels.

**6.3. Termination Criterion.** In all experiments, the normalized averaged entropy

$$-\frac{1}{m \log(n)} \sum_{i \in \mathcal{V}} \sum_{k=1}^n W_{ik} \log(W_{ik}) \quad (6.12)$$

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.

## 7. EXPERIMENTS

We demonstrate in this section main properties of our approach. The dependency of label assignment on the smoothing parameter  $\tau$  and the rounding parameter  $\alpha$  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 of nonbinary vertices. The results reported in Section 7.2 exhibit a relationship between  $\alpha$  and  $\tau$  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 two established and widely applied approaches, sequential tree-reweighted message passing (TRWS) [Kol06] and loopy belief propagation reveals similar

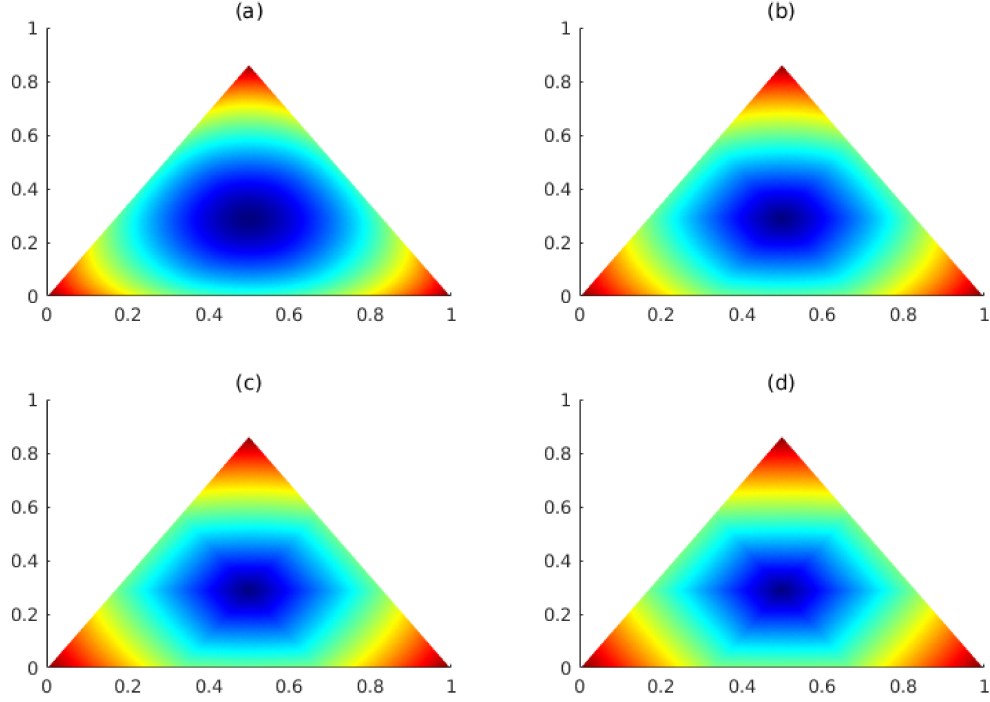


FIGURE 6.3. The plot shows the entropy-regularized Wasserstein distance with the Potts regularizer from the barycenter to every point on  $\Delta_3$  for different values of  $\tau$ . Used values are (a)  $\tau = \frac{1}{5}$ , (b)  $\tau = \frac{1}{10}$ , (c)  $\tau = \frac{1}{20}$  and (d)  $\tau = \frac{1}{50}$ . These plots confirm that even for relatively large values of  $\tau$ , e.g.  $\frac{1}{10}$  and  $\frac{1}{20}$ , the gradient of the Wasserstein distance is sufficiently accurate approximated so as to obtain valid descent directions for distance minimization.

performance of 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 was to label a noisy RGB-image  $f: \mathcal{V} \rightarrow [0, 1]^3$ , depicted in Fig. 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  $\mathcal{P} = \{l_1, \dots, l_8\} \subset [0, 1]^3$  (Fig. 7.2) were used as labels. The unary (or data term) was defined using the  $\|\cdot\|_1$  distance and a scaling factor  $\rho > 0$  by

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

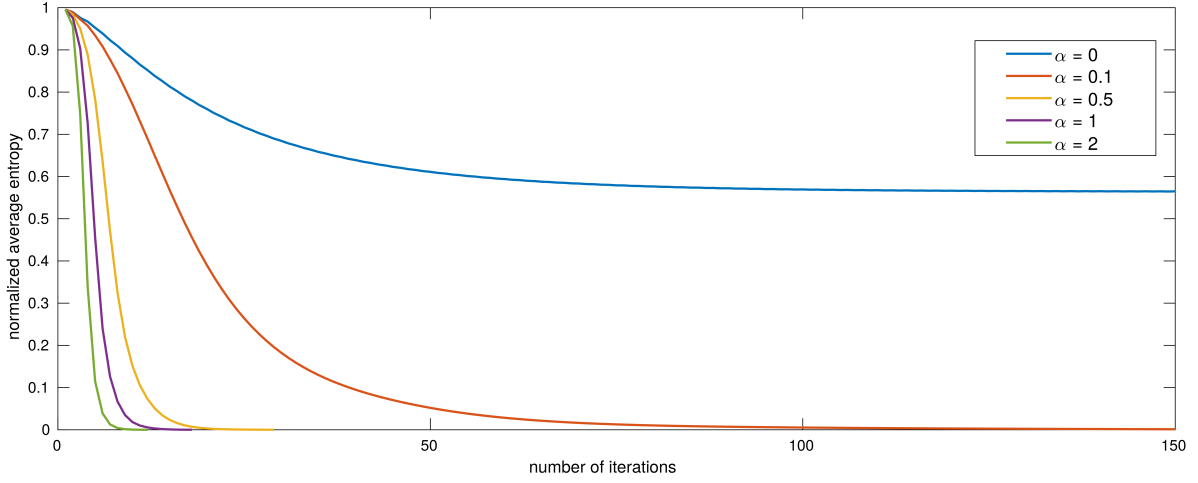


FIGURE 7.1. The normalized average entropy (6.12) as a function of iterations for the smoothing parameter value  $\tau = 0.1$ . With increasing values for  $\alpha$  the entropy drops more rapidly and hence converges faster to an integral labeling.

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

$$(\theta_{ij})_{kp} = \delta_{k,p} = \begin{cases} 1 & \text{if } k = p, \\ 0 & \text{else,} \end{cases} \quad ij \in \mathcal{E}. \quad (7.2)$$

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 (6.12) was set to  $10^{-4}$ .

Figure 7.1 displays the empirical convergence rate depending on the rounding parameter  $\alpha$ , 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. Less aggressive rounding in terms of smaller values of  $\alpha$  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. Overall, the total number of iterations is significantly smaller than the number of iterations which first-order methods of convex programming require for solving the LP relaxation [LS11]. For too small values of the rounding parameter  $\alpha$ , the algorithm does naturally not converge to an integral solution.

Fig. 7.2 shows the influence of the rounding strength  $\alpha$  and the smoothing parameter  $\tau$  for the Wasserstein distance. All images marked with an '\*' in the lower right corner do not show an integral solution, which means that the normalized average entropy (6.12) 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 Fig. 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  $\alpha$ ) leads to a faster decrease in entropy (cf. Fig. 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.

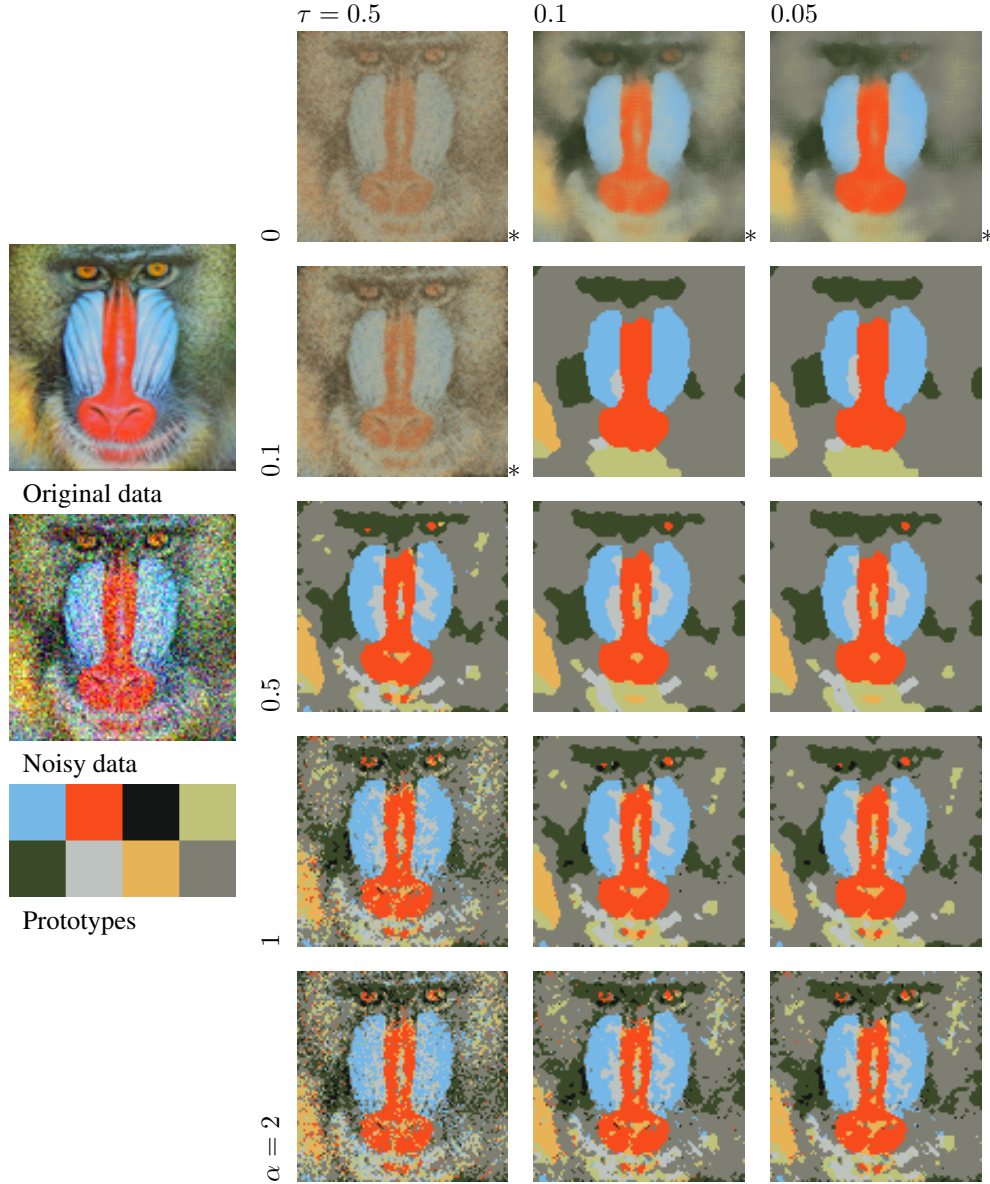


FIGURE 7.2. Influence of the rounding parameter  $\alpha$  and the smoothing parameter  $\tau$  on the assignment of 8 prototypical labels to noisy input data. All images marked with an '\*' do not show integral solutions because too large smoothing of the Wasserstein distance in terms of  $\tau$ , relative to  $\alpha$ , overcompensated the effect of rounding. Likewise, a too strong smoothing of the Wasserstein distance (left column,  $\tau = 0.5$ ) yields poor approximations of the objective function gradient and to erroneous label assignments. For the remaining parameter regime, i.e. smoothing below a still reasonably large upper bound  $\tau = 0.1$  that leads to fast numerical convergence, the label assignment can be precisely controlled by  $\alpha$ .

On the other hand, choosing the smoothing parameter  $\tau$  too large lead to poor approximations of the Wasserstein distance gradients and consequently to erroneous non-regularized labelings, as displayed in



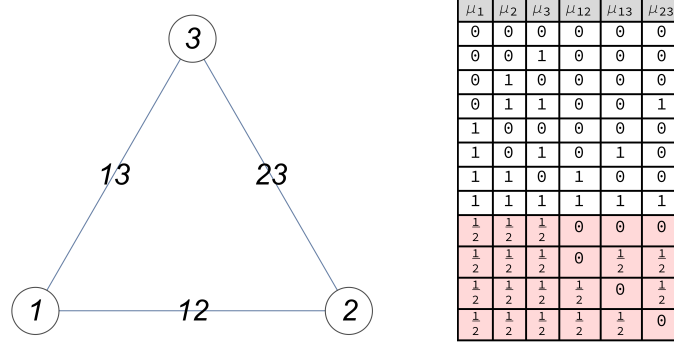


FIGURE 7.3. LEFT: The minimal binary cyclic graphical model  $\mathcal{K}^3 = (\mathcal{V}, \mathcal{E}) = (\{1, 2, 3\}, \{12, 13, 23\})$ . RIGHT: The 8 vertices (white background) of the minimally represented marginal polytope  $\mathcal{P}_{\mathcal{K}^3} \subset \mathbb{R}_+^6$  and the 4 additional non-integer vertices (red background) of the minimally represented local polytope  $\mathcal{L}_{\mathcal{K}^3} \subset \mathbb{R}_+^6$ .

the left-most column of Fig. 7.1. Once  $\tau$  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 conveniently controlled by  $\alpha$ . In particular, this upper bound on  $\tau$  is sufficiently large to ensure very rapid convergence of the fixed point iteration for computing the Wasserstein distance gradients.

**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  $\mathcal{K}^3$  (Fig. 7.3, left panel). Due to the single cycle, models exist where the LP relaxation (1.3) returns a non-binary solution (red part of the right panel of Fig. 7.3). As a consequence, evaluating such models with our geometric approach for minimizing (1.4) enables to check two properties:

- (i) Whenever solving the LP relaxation (1.3) by convex programming returns the global binary minimum of (1.2) as solution, we assess if our geometric approach based on the smooth approximation (1.4) returns this solution as well.
- (ii) Whenever the LP relaxation has a *non-binary* vector as 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 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 to [Pad89] for background and details.

The constraints are more conveniently stated using the so-called *minimal representation* of binary graphical models [WJ08, Sect. 3.2], that involves the variables<sup>1</sup>

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

and encodes the local vectors (2.15) by

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

<sup>1</sup>We reuse the symbol  $\mu$  for simplicity and only ‘overload’ in this subsection the symbols  $\mu_i, \mu_{ij}$  for local vectors (2.15) by the variables on the left-hand sides of (7.3)

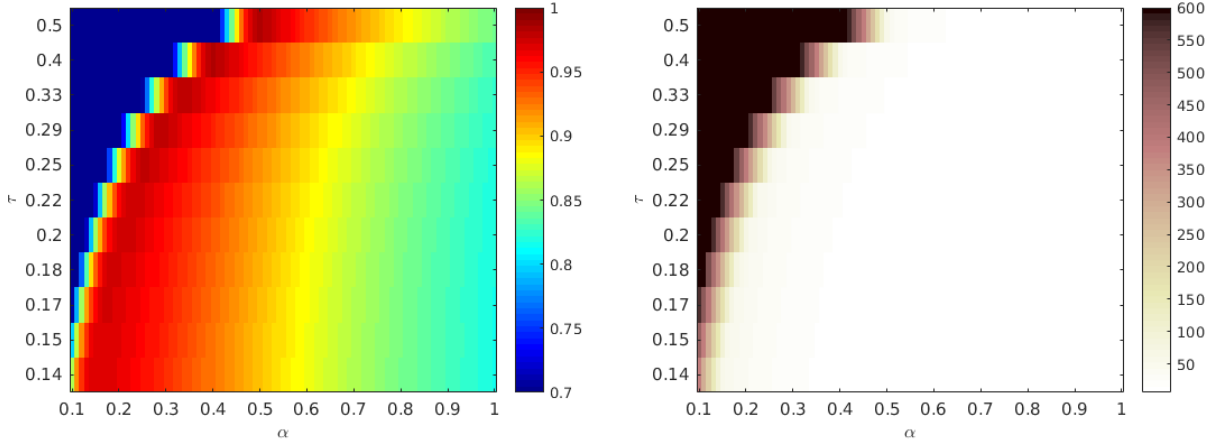


FIGURE 7.4. EVALUATION OF THE MINIMAL GRAPHICAL MODEL  $\mathcal{K}^3$ : For every pair of parameters  $(\tau, \alpha)$ , we evaluated  $10^5$  models, which were generated as explained in the text. In all experiments, we terminate the algorithm if the average entropy dropped below  $10^{-3}$  or if 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. RIGHT: This plot shows the corresponding average number of iterations. The black color indicates that the maximum number of 600 iterations was reached, because too strong smoothing of the Wasserstein distance (large  $\tau$ ) overcompensates the effect of rounding (small  $\alpha$ ), so that the convergence criterion (6.12) which measures the distance to integral solutions, cannot be satisfied. In the remaining large parameter regime, the choice of  $\alpha$  enables to control the trade-off between high-quality (low-energy) solutions and computational costs.

Thus, it suffices to use a single variable  $\mu_i$  for every node  $i \in \mathcal{V}$  instead of two variables  $\mu_i(0), \mu_i(1)$ , and also a single variable  $\mu_{ij}$  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

$$0 \leq \mu_{ij}, \quad \mu_{ij} \leq \mu_i, \quad \mu_{ij} \leq \mu_j, \quad \mu_i + \mu_j - \mu_{ij} \leq 1, \quad \forall ij \in \mathcal{E}. \quad (7.5)$$

The *marginal* polytope constraints additionally involve the so-called triangle inequalities [DL97]

$$\sum_{i \in \mathcal{V}} \mu_i - \sum_{jk \in \mathcal{E}} \mu_{jk} \leq 1, \quad (7.6a)$$

$$\mu_{12} + \mu_{13} - \mu_{23} \leq \mu_1, \quad \mu_{12} - \mu_{13} + \mu_{23} \leq \mu_2, \quad -\mu_{12} + \mu_{13} + \mu_{23} \leq \mu_3. \quad (7.6b)$$

Figure 7.3, right panel, lists the 8 vertices of  $\mathcal{P}_{\mathcal{K}^3}$  and the 4 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

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

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 make inference more difficult. Suppose, for example, that the diagonal terms of

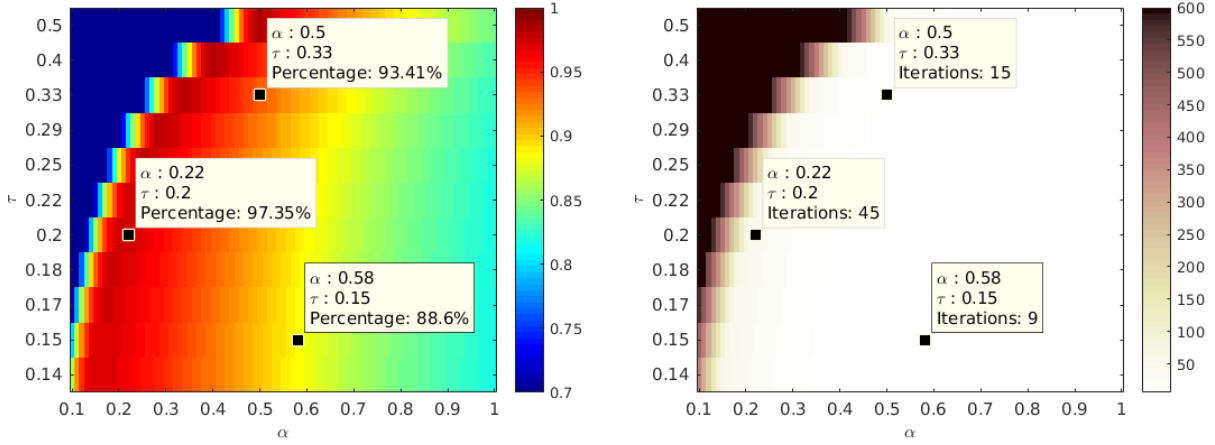


FIGURE 7.5. These plots show the same results as Fig. 7.4 together with additional data boxes and information for three different configurations of parameter values. Comparing the success rate (left panel) and the number of iterations until convergence (right panel) clearly demonstrate the trade-off between the accuracy of optimization and the convergence rate, depending on the *rounding* variable  $\alpha$  and the *smoothing* parameter  $\tau$ . Overall, the number of iterations is significantly smaller than for first-order methods of convex programming for solving the LP relaxation, that additionally require rounding as a post-processing step to obtain an integral solution.

$\theta_{ij}$  are large, which favours the assignment of *different* labels to the nodes  $1, 2, 3 \in \mathcal{V}$ . Then assigning say the 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 by Fig. 7.3, right panel, is the global optimum of both the linear relaxation (1.3) and the original objective function (1.2) in approximately  $\approx 11.94\%$  of the  $10^5$  scenarios, whereas every *non-binary* vertex is optimal in approximately  $\approx 1.12\%$ .

Fig. 7.4 presents the results of the experiments for the minimal cyclic graphical model  $\mathcal{K}^3$ . To get an intuition of our *rounding* parameter  $\alpha$  and the *smoothing* parameter  $\tau$ , we evaluated all  $10^5$  models for *each* pair of  $(\tau, \alpha)$ , where  $\tau \in \{\frac{1}{2}, \frac{1}{2.5}, \dots, \frac{1}{6.5}, \frac{1}{7}\}$  and  $\alpha \in \{0.1, 0.11, \dots, 0.99, 1\}$ .

Fig. 7.5 presents exactly the same results as Fig. 7.4, except that we include data boxes for three different configurations of parameter. E.g., for  $\alpha = 0.22$  and  $\tau = \frac{1}{5}$  our algorithm found in 97.35% of the experiments an energy with relative error smaller than 1% compared to the optimal energy. At the same time, the algorithm required 45 iterations on average to converge. As shown by the following parameter,  $\alpha = 0.58$  and  $\tau = 0.15$ , a larger value of  $\alpha$  means that the rounding mechanism in each iteration step (5.4) is increased. The average number of iterations is reduced to 9, but the accuracy is 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  $\alpha$ , for all values of  $\tau$  below a reasonably large upper bound,
- a small or very small number of iterations required to converge, depending on the choice of  $\alpha$ .

**7.3. Comparison to other Methods.** We compared our geometric approach to sequential tree-reweighted message passing (TRWS) [Kol06] and loopy belief propagation [Wei01] (Loopy-BP) based on the OpenGM package [ABK12].

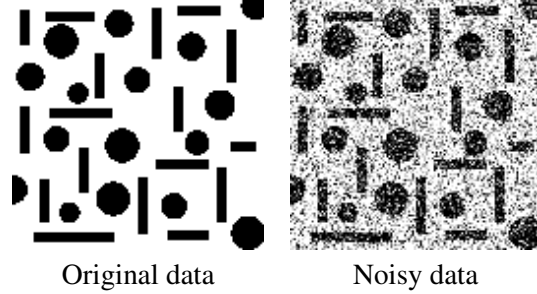


FIGURE 7.6. Noisy image labeling problem: a binary ground truth image (left) to be recovered from noisy input data (right).

For this comparison, we evaluate the performance of the methods for a noisy binary labeling scenario depicted by Fig. 7.6. Let  $f: \mathcal{V} \rightarrow [0, 1]$  denote the noisy image data given on the grid graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  with a 4-neighborhood and  $\mathcal{X} = \{0, 1\}$  be the prototypes. Then the used data term and Potts prior are given by

$$\theta_i = \begin{pmatrix} f(i) \\ 1 - f(i) \end{pmatrix} \quad \text{for } i \in \mathcal{V} \quad \text{and} \quad \theta_{ij} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad \text{for } ij \in \mathcal{E}. \quad (7.8)$$

In the experiments, the threshold for the normalized average entropy termination criterion (6.12) was  $10^{-4}$ . Figure 7.7 shows the visual reconstruction as well as the corresponding energy values and percentage of correct labels for all three methods. Our method has the same accuracy and returns the same optimal energy level as TRWS and Loopy-BP.

To investigate once more the influence of the *rounding* mechanism, we repeated the same experiment as explained above, but for different values of the rounding parameter  $\alpha \in \{0.1, 1, 2, 5\}$ . As shown by Fig. 7.8, the results confirm the finding of the experiments of the preceding section: A more aggressive rounding scheme ( $\alpha$  large) leads to faster convergence but yields less regularized results with higher energy values.

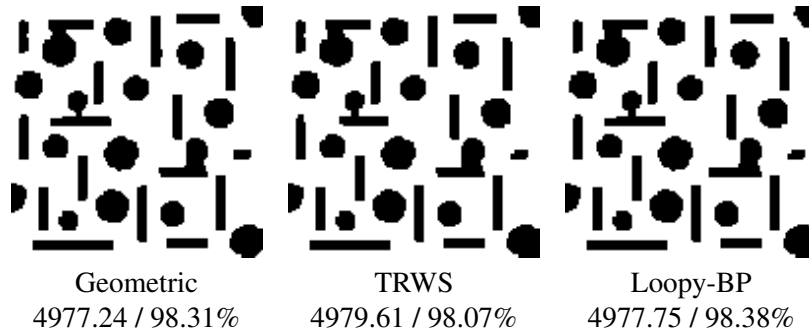


FIGURE 7.7. Results for the noisy labeling problem from Fig. 7.6 using a standard data term with Potts prior, with Energy / Accuracy values. Parameter values for the geometric approach: smoothing  $\tau = 0.1$ , step-size  $h = 0.2$  and rounding strength  $\alpha = 0.1$ . The threshold for the termination criterion was  $10^{-4}$ . All methods show equal performance.

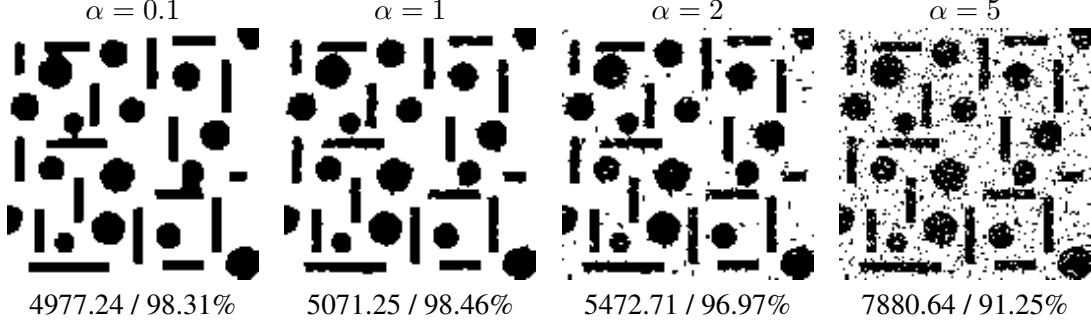


FIGURE 7.8. Results for the noisy labeling problem from Fig. 7.6 using different values of the rounding parameter  $\alpha \in \{0.1, 1, 2, 5\}$  with Energy / Accuracy values: more aggressive rounding scheme ( $\alpha$  large) leads to less regularized results with higher energy values. Parameter values of the geometric approach: smoothing  $\tau = 0.1$ , step-size  $h = 0.2$  and the threshold for the termination criterion was  $10^{-4}$ .

**7.4. Non-Uniform (Non-Potts) Priors.** We examined the behavior of our approach for a non-Potts prior by applying it to a non-binary labeling problem with noisy input image, as depicted by Fig. 7.9. Our objective is to demonstrate that pre-specified pairwise model parameters (regularization) by a graphical model are properly taken into account.

Assume additional information about a labeling problem is 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" (Fig. 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 non-uniform pairwise model parameters accordingly, as specified below.

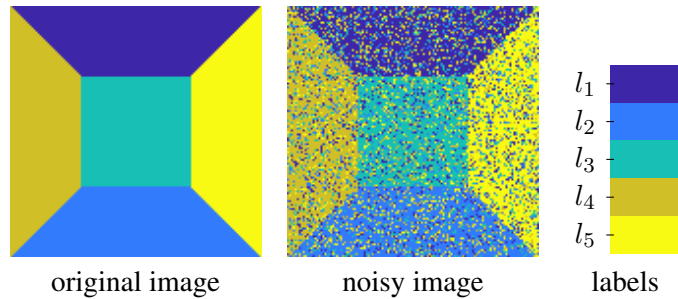


FIGURE 7.9. Original image (left), encoding the image directions "top", "bottom", "center", "left" and "right" by the RGB-color labels  $l_1, l_2, l_3, l_4$  and  $l_5$  (right). The noisy test image (middle) was created by randomly selecting 40% of the original image pixels and then uniformly sampling a label at each chosen positions. The label constraints suggested by the labeling order leads to a non-Potts prior (7.11).

The label indices corresponding to the five RGB-colors of the original image (Fig. 7.9 right) are

$$\mathcal{X} = \{\ell_1 = \text{"dark blue"}, \ell_2 = \text{"light blue"}, \ell_3 = \text{"cyan"}, \ell_4 = \text{"orange"}, \ell_5 = \text{"yellow"}\} \subset [0, 1]^3. \quad (7.9)$$

Let  $f: \mathcal{V} \rightarrow [0, 1]^3$  denote the noisy input image (Fig. 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

$$\theta_i = \frac{1}{\rho} (\|f(i) - l_1\|_1, \dots, \|f(i) - l_5\|_1), \quad i \in \mathcal{V}. \quad (7.10)$$

The non-Potts prior representing the prior knowledge of our problem was chosen as

$$\theta_{ij} = \frac{1}{10} \begin{pmatrix} 0 & 10 & 1 & 1 & 1 \\ 10 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 10 \\ 1 & 1 & 1 & 10 & 0 \end{pmatrix}, \quad ij \in \mathcal{E}, \quad (7.11)$$

that is, every entry of  $\theta_{ij}$  corresponding to a label configuration  $l_1 = \text{"dark blue" ("top")}$  next to  $l_2 = \text{"light blue" ("bottom")}$  or  $l_4 = \text{"orange" ("left")}$  next to  $l_5 = \text{"yellow" ("right")}$  penalizes such label transitions with a large weight of 1, while all other "natural" configurations are treated as with the Potts prior and lower weights of 0 resp. 0.1 and are therefore less penalized.

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

$$(\theta'_{ij})_{kp} = \frac{1}{10} \delta_{k,p} = \begin{cases} \frac{1}{10} & \text{if } k = p, \\ 0 & \text{else,} \end{cases} \quad ij \in \mathcal{E}. \quad (7.12)$$

For 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 (6.12).

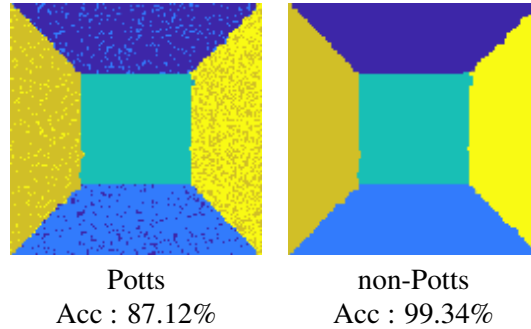


FIGURE 7.10. Results of the labeling problem using the Potts and non-Potts prior model together with the Accuracy (Acc) values. Parameters for this experiment are  $\rho = 15$ , smoothing  $\tau = 0.01$ , step-size  $h = 0.1$  and rounding strength  $\alpha = 0.01$ . The threshold for the termination criterion (6.12) was  $10^{-4}$ .

The results depicted in Fig. 7.10, clearly show the influence of the non-Potts prior (labeling accuracy 99.34%) whereas using the Potts prior lowers accuracy to only 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 a Potts model. The additional prior information about valid label configurations encoded by (7.11) suffices to overcome this difficulty 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 marginalization constraints are satisfied throughout the process. A single parameter enables to trade-off accuracy of optimization and speed of convergence.

Our work motivates to address applications using graphical models with higher edge connectivity, where established inference algorithms based on convex programming noticeably slow down. In addition, 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 multiple spatial scales.

## APPENDIX A. PROOFS

**A.1. Proof of Proposition 4.2.** Let  $\gamma: (-\varepsilon, \varepsilon) \rightarrow \mathcal{W}$  be a smooth curve, with  $\varepsilon > 0$ ,  $\gamma(0) = W$  and  $\dot{\gamma}(0) = V$ . We then have

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

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

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

the r.h.s. of (A.1) becomes

$$\langle \nabla E_\tau(W), V \rangle = \sum_{i \in V} \left( \langle P_T(\theta_i), V_i \rangle + \sum_{j: (i,j) \in \mathcal{E}} \langle \nabla_1 d_{\theta_{ij}, \tau}(W_i, W_j), V_i \rangle \right) + \sum_{i \in V} \sum_{j: (i,j) \in \mathcal{E}} \langle \nabla_2 d_{\theta_{ij}, \tau}(W_i, W_j), V_j \rangle, \quad (\text{A.3})$$

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

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

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

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

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

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

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

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

which proves (4.13).

**A.2. Proof of Lemma 4.3.** Let  $M_* \in \Pi(W_i, W_j)$  be a minimizer of (4.8). Then due to the assumption on  $F_\tau$ , we have

$$d_{\theta_{ij}, \tau}(W_i, W_j) = \langle \theta_{ij}, M_* \rangle + F_\tau(M_*) = \langle \theta_{ij}^\top, M_*^\top \rangle + F_\tau(M_*^\top). \quad (\text{A.6})$$

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

$$\langle \theta_{ij}^\top, \tilde{M} \rangle + F_\tau(\tilde{M}) = \langle \theta_{ij}, \tilde{M}^\top \rangle + F_\tau(\tilde{M}^\top) \geq \langle \theta_{ij}, M_* \rangle + F_\tau(M_*) = \langle \theta_{ij}^\top, M_*^\top \rangle + F_\tau(M_*^\top). \quad (\text{A.7})$$

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 equation (4.15).

**A.3. Proof of Corollary 4.4.** Applying the equation  $\nabla_2 d_{\theta_{ij}, \tau}(W_j, W_i) = \nabla_1 d_{\theta_{ij}, \tau}(W_i, W_j)$  due to Lemma 4.3 to Eqn. (4.13), we obtain

$$\nabla_i E_\tau(W) = P_T(\theta_i) + \sum_{j: (i,j) \in \mathcal{E}} \nabla_1 d_{\theta_{ij}, \tau}(W_i, W_j) + \sum_{j: (j,i) \in \mathcal{E}} \nabla_1 d_{\theta_{ij}, \tau}(W_i, W_j) \quad (\text{A.8a})$$

$$= P_T(\theta_i) + \sum_{j \in \mathcal{N}(i)} \nabla_1 d_{\theta_{ij}, \tau}(W_i, W_j), \quad (\text{A.8b})$$

which is (4.17).

**A.4. Proof of Lemma 4.6.** Taking into account (2.15b), we write the right-hand side of (4.8) in the form

$$\min_M \langle \Theta, M \rangle + G_\tau(M) \quad \text{s.t.} \quad \mathcal{A}M = \mu, \quad (\text{A.9})$$

Let  $\nu = (\nu_1, \nu_2) \in \mathbb{R}^{2n}$  denote the dual variables corresponding to the affine constraint of (A.9). Then problem (A.9) rewritten in Lagrangian form reads

$$\min_M \{ \langle \Theta, M \rangle + G_\tau(M) + \max_\nu \langle \nu, \mu - \mathcal{A}M \rangle \} \quad (\text{A.10a})$$

$$\Leftrightarrow \min_M \{ \max_\nu \langle \nu, \mu \rangle + G_\tau(M) - \langle \mathcal{A}^\top \nu - \Theta, M \rangle \}. \quad (\text{A.10b})$$

Since strong duality holds by assumption, interchanging min and max yields the dual problem (4.23). Moreover, the optimal primal and dual objective function values are equal, which gives with (A.10a) and (4.23)

$$\langle \overline{M}, \mathcal{A}^\top \overline{\nu} - \Theta \rangle + G_\tau(\overline{M}) + G_\tau^*(\mathcal{A}^\top \overline{\nu} - \Theta) = 0. \quad (\text{A.11})$$

This implies (4.24a) by the subgradient inversion rule [RW09, Prop. 11.3], whereas the primal constraint (4.24b) is obvious.

**A.5. Proof of Lemma 4.7.** Let  $z = \begin{pmatrix} x \\ y \end{pmatrix} \in \mathbb{R}^{2n}$  with  $0 = \mathcal{A}^\top z = x \mathbb{1}_n^\top + \mathbb{1}_n y^\top$ . Applying  $\mathcal{A}$ , we get

$$0 = \mathcal{A} \mathcal{A}^\top z = \mathcal{A}(x \mathbb{1}_n^\top) + \mathcal{A}^\top(\mathbb{1}_n y^\top) = \begin{pmatrix} nx + \langle y, \mathbb{1}_n \rangle \mathbb{1}_n \\ \langle x, \mathbb{1}_n \rangle \mathbb{1}_n + ny \end{pmatrix} \Leftrightarrow z = \begin{pmatrix} x \\ y \end{pmatrix} = -\frac{1}{n} \begin{pmatrix} \langle y, \mathbb{1}_n \rangle \mathbb{1}_n \\ \langle x, \mathbb{1}_n \rangle \mathbb{1}_n \end{pmatrix}. \quad (\text{A.12})$$

This implies  $\langle x, \mathbb{1}_n \rangle = -\langle y, \mathbb{1}_n \rangle$ , and setting  $\alpha = \frac{1}{n} \langle x, \mathbb{1}_n \rangle \in \mathbb{R}$  shows that  $z$  has the form (4.25). Conversely, in view of the definition (2.3b), it is clear that any vector from the set (4.25) is in  $\mathcal{N}(\mathcal{A}^\top)$ . The characterization of  $\mathcal{N}(\mathcal{A}^\top)^\perp$  directly follows from the definitions.

**A.6. Proof of Lemma 4.8.** We first show that, if  $\bar{\nu}$  is an optimal dual solution, then

$$\operatorname{argmax}_{\nu \in \mathbb{R}^{2n}} g(p, \nu) \subseteq \bar{\nu} + \mathcal{N}(\mathcal{A}^\top). \quad (\text{A.13})$$

Let  $\bar{\nu}' \neq \bar{\nu}$  be another optimal dual solution, that is  $g(p, \bar{\nu}) = g(p, \bar{\nu}')$ . By (4.18), this equation reads

$$G_\tau^*(\mathcal{A}^\top \bar{\nu} - \Theta) - G_\tau^*(\mathcal{A}^\top \bar{\nu}' - \Theta) = \langle p, \bar{\nu} - \bar{\nu}' \rangle. \quad (\text{A.14})$$

Moreover, due to the optimality conditions (4.24),  $\bar{\nu}'$  satisfies

$$\bar{M}' = \nabla G_\tau^*(\mathcal{A}^\top \bar{\nu}' - \Theta), \quad \mathcal{A} \bar{M}' = p, \quad (\text{A.15})$$

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

$$\langle p, \bar{\nu} - \bar{\nu}' \rangle = \langle \mathcal{A} \bar{M}', \bar{\nu} - \bar{\nu}' \rangle = \langle \bar{M}', \mathcal{A}^\top (\bar{\nu} - \bar{\nu}') \rangle \stackrel{(\text{A.15})}{=} \langle \nabla G_\tau^*(\mathcal{A}^\top \bar{\nu}' - \Theta), \mathcal{A}^\top (\bar{\nu} - \bar{\nu}') \rangle. \quad (\text{A.16})$$

Using the shorthands

$$\bar{w} = \mathcal{A}^\top \bar{\nu} - \Theta, \quad \bar{w}' = \mathcal{A}^\top \bar{\nu}' - \Theta, \quad (\text{A.17})$$

we have

$$\bar{w}' - \bar{w} = \mathcal{A}^\top (\bar{\nu}' - \bar{\nu}) \quad (\text{A.18})$$

and therefore

$$G_\tau^*(\bar{w}') - G_\tau^*(\bar{w}) \stackrel{(\text{A.14})}{=} \langle p, \bar{\nu}' - \bar{\nu} \rangle \stackrel{(\text{A.16})}{=} \langle \nabla G_\tau^*(\bar{w}'), \bar{w}' - \bar{w} \rangle. \quad (\text{A.19})$$

Since  $G_\tau^*$  is strictly convex, this equality can only hold if

$$0 = \bar{w}' - \bar{w} \stackrel{(\text{A.18})}{=} \mathcal{A}^\top (\bar{\nu}' - \bar{\nu}). \quad (\text{A.20})$$

This shows that  $\bar{\nu}$  and  $\bar{\nu}'$  can only differ by a nullspace vector, i.e. we have shown relation (A.13). It remains to show the reverse inclusion, that is vectors characterized by the right-hand side of (4.26) maximize the dual objective function  $g(p, \nu)$ .

Let again  $\bar{\nu}$  be an optimal dual solution, and let  $\bar{\nu}' \in \bar{\nu} + \mathcal{N}(\mathcal{A}^\top)$  be an arbitrary vector. Lemma 4.7 implies that  $\bar{\nu}'$  takes the form

$$\bar{\nu}' = \bar{\nu} + \alpha \begin{pmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{pmatrix}, \quad \alpha \in \mathbb{R}. \quad (\text{A.21})$$

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

$$g(p, \bar{\nu}') = \langle p, \bar{\nu} + \alpha \begin{pmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{pmatrix} \rangle - G_\tau^*(\mathcal{A}^\top (\bar{\nu} + \alpha \begin{pmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{pmatrix}) - \Theta) \quad (\text{A.22a})$$

$$= \langle p, \bar{\nu} \rangle - G_\tau^*(\mathcal{A}^\top \bar{\nu} - \Theta) = g(p, \bar{\nu}), \quad (\text{A.22b})$$

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

Finally, suppose  $\langle p, \begin{pmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{pmatrix} \rangle \neq 0$ ,  $\bar{\nu}$  is an optimal dual solution and  $\bar{\nu}'$  is another optimal dual vector, which has the form (A.21) as just shown. Inserting (A.21) into (A.14) yields

$$0 = \langle p, \bar{\nu}' - \bar{\nu} \rangle = \alpha \langle p, \begin{pmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{pmatrix} \rangle. \quad (\text{A.23})$$

Since  $\langle p, \begin{pmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{pmatrix} \rangle \neq 0$ , this can only hold if  $\alpha = 0$ . Thus,  $\bar{\nu}' = \bar{\nu}$  by (A.21), which shows uniqueness of  $\bar{\nu}$  as claimed by (4.26).

**A.7. Proof of Lemma 4.9.** We first shown (4.27b). Let  $\bar{\nu}$  be an optimal dual solution. Since  $\langle p, \begin{pmatrix} \mathbb{1}_n \\ -\mathbb{1}_n \end{pmatrix} \rangle = 0$ , Lemma 4.8 yields  $\operatorname{argmax}_{\nu \in \mathbb{R}^{2n}} g(p, \nu) = \bar{\nu} + \mathcal{N}(\mathcal{A}^\top) = \bar{\nu}_\mathcal{N} + \bar{\nu}_\mathcal{R} + \mathcal{N}(\mathcal{A}^\top)$ . This shows  $\bar{\nu}_\mathcal{R} \in \bar{\nu} + \mathcal{N}(\mathcal{A}^\top)$ , that is  $\bar{\nu}_\mathcal{R} \in \mathcal{R}(\mathcal{A})$  is a maximizer, which implies (4.27b).

Let  $\bar{\nu}'_\mathcal{R} \in \mathcal{R}(\mathcal{A})$  be another maximizer. As before, we have the representation  $\bar{\nu}'_\mathcal{R} \in \bar{\nu} + \mathcal{N}(\mathcal{A}^\top)$ , that is  $\bar{\nu}'_\mathcal{R} = \bar{\nu}_\mathcal{N} + \bar{\nu}_\mathcal{R} + \tilde{\nu}_\mathcal{N}$  for some  $\tilde{\nu}_\mathcal{N} \in \mathcal{N}(\mathcal{A}^\top)$ , which implies  $\bar{\nu}'_\mathcal{R} = \bar{\nu}_\mathcal{R}$ , i.e. uniqueness (4.27a) of the dual maximizer in  $\mathcal{R}(\mathcal{A})$ .

## ACKNOWLEDGEMENTS

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

## REFERENCES

- [ABK12] B. Andres, T. Beier, and J.H. Kappes, *OpenGM: A C++ Library for Discrete Graphical Models*, CoRR **abs/1206.0111** (2012).
- [ÅHS<sup>+</sup>17] F. Åström, R. Hühnerbein, F. Savarino, J. Recknagel, and C. Schnörr, *MAP Image Labeling Using Wasserstein Messages and Geometric Assignment*, Proc. SSVM, LCNS 10302, Springer, 2017.
- [ÅPSS17] F. Åström, S. Petra, B. Schmitzer, and C. Schnörr, *Image Labeling by Assignment*, J. Math. Imag. Vision **58** (2017), no. 2, 211–238.
- [BFPS17] R. Bergmann, J.H. Fitschen, J. Persch, and G. Steidl, *Iterative multiplicative filters for data labeling*, Int. J. Computer Vision **123** (2017), no. 3, 435–453.
- [BK04] Y. Boykov and V. Kolmogorov, *An Experimental Comparison of Min-Cut/Max-Flow Algorithms for Energy Minimization in Vision*, IEEE Trans. Patt. Anal. Mach. Intell. **26** (2004), no. 9, 1124–1137.
- [Bru06] R.A. Brualdi, *Combinatorial Matrix Classes*, Cambridge Univ. Press, 2006.
- [BVZ01] Y. Boykov, O. Veksler, and R. Zabih, *Fast Approximate Energy Minimization via Graph Cuts*, IEEE Trans. Patt. Anal. Mach. Intell. **23** (2001), no. 11, 1222–1239.
- [CKNZ05] C. Chekuri, S. Khanna, J. Naor, and L. Zosin, *Linear Programming Formulation and Approximation Algorithms for the Metric Labeling Problem*, SIAM J. Discr. Math. **18** (2005), no. 3, 608–625.
- [CP16] M. Cuturi and G. Peyré, *A Smoothed Dual Approach for Variational Wasserstein Problems*, SIAM J. Imag. Sci. **9** (2016), no. 1, 320–343.
- [Cut13a] M. Cuturi, *Sinkhorn Distances: Lightspeed Computation of Optimal Transport*, Proc. NIPS, 2013.
- [Cut13b] ———, *Sinkhorn Distances: Lightspeed Computation of Optimal Transport*, Advances in Neural Information Processing Systems 26 (C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, eds.), Curran Associates, Inc., 2013, pp. 2292–2300.
- [Dan66] J.M. Danskin, *The Theory of Max-Min, with Applications*, SIAM J. Appl. Math. (1966).
- [DL97] M.M. Deza and M. Laurent, *Geometry of Cuts and Metrics*, Springer, 1997.
- [HS10] T. Hazan and A. Shashua, *Norm-Product Belief Propagation: Primal-Dual Message-Passing for Approximate Inference*, IEEE Trans. Inf. Theory **56** (2010), no. 12, 6294–6316.
- [KAH<sup>+</sup>15] J.H. Kappes, B. Andres, F.A. Hamprecht, C. Schnörr, S. Nowozin, D. Batra, S. Kim, B.X. 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. Comp. Vision **115** (2015), no. 2, 155–184.
- [Kni08] P.A. Knight, *The Sinkhorn-Knopp Algorithm: Convergence and Applications*, SIAM J. Matrix Anal. Appl. **30** (2008), no. 1, 261–275.
- [Kol06] V. Kolmogorov, *Convergent Tree-Reweighted Message Passing for Energy Minimization*, IEEE Trans. Patt. Anal. Mach. Intell. **28** (2006), no. 10, 1568–1583.
- [KPT<sup>+</sup>17] S. Kolouri, S. Park, M. Thorpe, D. Slepcev, and G.K. Rohde, *Optimal mass transport: Signal processing and machine-learning applications*, IEEE Signal Proc. Mag. **34** (2017), no. 4, 43–59, preprint: <https://arxiv.org/abs/1609.04767>.
- [KT02] J. Kleinberg and E. Tardos, *Approximation algorithms for classification problems with pairwise relationships: metric labeling and Markov random fields*, Journal of the ACM **49** (2002), no. 5, 616–639.
- [KZ04] V. Kolmogorov and R. Zabih, *What Energy Functions Can Be Minimized via Graph Cuts?*, IEEE Trans. Patt. Analysis Mach. Intell. **26** (2004), no. 2, 147–159.
- [LS11] J. Lellmann and C. Schnörr, *Continuous Multiclass Labeling Approaches and Algorithms*, SIAM J. Imaging Science **4** (2011), no. 4, 1049–1096.
- [NN87] Y. Nesterov and A. Nemirovskii, *Interior Point Polynomial Algorithms in Convex Programming*, Studies in Applied Mathematics, Society for Industrial and Applied Mathematics, 1987.
- [Pad89] M. Padberg, *The Boolean Quadratic Polytope: Some Characteristics, Facets and Relatives*, Math. Progr. **45** (1989), 139–172.
- [PDHA97] T. Pham Dinh and L. Hoai An, *Convex Analysis Approach to D.C. Programming: Theory, Algorithms and Applications*, Acta Math. Vietnamica **22** (1997), no. 1, 289–355.
- [PDHA98] T. Pham Dinh and L.T. Hoai An, *A D.C. Optimization Algorithm for Solving the Trust-Region Subproblem*, SIAM J. Optimization **8** (1998), no. 2, 476–505.
- [Pey15] G. Peyré, *Entropic Approximation of Wasserstein Gradient Flows*, SIAM J. Imag. Sci. **8** (2015), no. 4, 2323–2351.

- [Ren95] J. Renegar, *Linear Programming, Complexity Theory and Elementary Functional Analysis*, Math. Progr. **70** (1995), 279–351.
- [Roc91] R.T. Rockafellar, *On a Special Class of Functions*, J. Opt. Theory Appl. **70** (1991), no. 3, 619–621.
- [RW09] R.T. Rockafellar and R. J.-B. Wets, *Variational Analysis*, 2nd ed., Springer, 2009.
- [Sch90] M.H. Schneider, *Matrix Scaling, Entropy Minimization, and Conjugate Duality (II): The Dual Problem*, Math. Progr. **48** (1990), 103–124.
- [Sch16] B. Schmitzer, *A sparse multiscale algorithm for dense optimal transport*, J. Math. Imag. Vision **56** (2016), no. 2, 238–259.
- [SHÅ<sup>+</sup>17] F. Savarino, R. Hühnerbein, F. Åström, J. Recknagel, and C. Schnörr, *Numerical integration of Riemannian gradient flows for image labeling*, In Proc. SSVMLCNS (2017).
- [Sin64] R. Sinkhorn, *A Relationship Between Arbitrary Positive Matrices and Doubly Stochastic Matrices*, Ann. Math. Statist. **35** (1964), no. 2, 876–879.
- [SSK<sup>+</sup>16] P. Swoboda, A. Shekhovtsov, J.H. Kappes, C. Schnörr, and B. Savchynskyy, *Partial Optimality by Pruning for MAP-Inference with General Graphical Models*, IEEE Trans. Patt. Anal. Mach. Intell. **38** (2016), no. 7, 1370–1382.
- [Ter96] T. Terlaky (ed.), *Interior Point Methods of Mathematical Programming*, Kluwer Acad. Publ., 1996.
- [Wei01] Yair Weiss, *Comparing the mean field method and belief propagation for approximate inference in MRFs*, Advanced Mean Field Methods: Theory and Practice, MIT Press, 2001, pp. 229–240.
- [Wer07] T. Werner, *A Linear Programming Approach to Max-sum Problem: A Review*, IEEE Trans. Patt. Anal. Mach. Intell. **29** (2007), no. 7, 1165–1179.
- [WJ08] M.J. Wainwright and M.I. Jordan, *Graphical Models, Exponential Families, and Variational Inference*, Found. Trends Mach. Learning **1** (2008), no. 1-2, 1–305.
- [WJW05] M.J. Wainwright, T.S. Jaakola, and A.S. Willsky, *MAP Estimation via Agreement on Trees: Message-Passing and Linear Programming*, IEEE Trans. Inform. Theory **51** (2005), no. 11, 3697–3717.
- [YFW05] J.S. Yedidia, W.T. Freeman, and Y. Weiss, *Constructing Free-Energy Approximations and Generalized Belief Propagation Algorithms*, IEEE Trans. Information Theory **51** (2005), no. 7, 2282–2312.
- [YMW06] C. Yanover, T. Meltzer, and Y. Weiss, *Linear Programming Relaxations and Belief Propagation - An Empirical Study*, J. Mach. Learning Res. **7** (2006), 1887–1907.

(R. Hühnerbein) IMAGE AND PATTERN ANALYSIS GROUP, HEIDELBERG UNIVERSITY, GERMANY  
*E-mail address:* ruben.huehnerbein@iwr.uni-heidelberg.de  
*URL:* <http://ipa.math.uni-heidelberg.de>

(F. Savarino) IMAGE AND PATTERN ANALYSIS GROUP, HEIDELBERG UNIVERSITY, GERMANY  
*E-mail address:* fabrizio.savarino@iwr.uni-heidelberg.de  
*URL:* <http://ipa.math.uni-heidelberg.de>

(F. Åström) HEIDELBERG COLLABORATORY FOR IMAGE PROCESSING, HEIDELBERG UNIVERSITY, GERMANY  
*E-mail address:* freddie.astroem@iwr.uni-heidelberg.de  
*URL:* <https://hciweb.iwr.uni-heidelberg.de/user/fastroem>

(C. Schnörr) IMAGE AND PATTERN ANALYSIS GROUP, HEIDELBERG UNIVERSITY, GERMANY  
*E-mail address:* schnoerr@math.uni-heidelberg.de  
*URL:* <http://ipa.math.uni-heidelberg.de>