An iterated LP approach to the exact computation of the partition function in general Markov random field models

Ikhlef Bechar

Aix Marseille Univ, Université de Toulon, CNRS, ENSAM, LSIS
83957 La Garde, France

Abstract

In a recent paper [1], we have shown that one may efficiently compute the marginals of a general higher-order Markov random field (MRF) model with respect to an arbitrary collection of point subsets by solving a single linear program. Therefore, in this paper, we build on such a work for showing that the partition function of a general higher-order MRF model may, in turn, be computed both in an exact and efficient way, by iteratively solving at most n linear programs, with n standing for the number of nodes in the graph representation of a MRF model. This is, first, achieved by establishing a recursive formula between the partition function of a given MRF model and the one of another (one-degree simpler) MRF model resulting from the pruning of any of the nodes of the graph of the former. Then, we show that the problem amounts to solving successive and increasingly simpler MRF marginal inference problems, and thus, one may use the approach developed in [1] for solving such MRF marginal inference problems.

Keywords: MRF distribution, higher-order clique-potentials, partition function, recursive formula, marginal distribution, iterated linear programming, computationally efficient solution.

Email address: ikhlef.bechar@univ-tln.fr, ikhlef.bechar@lsis.org, ikhlef.bechar@gmail.com (Ikhlef Bechar)

Preprint 9/2017

1. Introduction

Suppose $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is an undirected graph, with $\mathcal{V}$ and $\mathcal{E}$ standing for its vertex-set (or node-set) and edge-set, respectively, and denote by $\mathcal{C}$ its set of maximal cliques, in other words, $\mathcal{C}$ stands for the maximal subsets of mutually neighboring vertices of $\mathcal{G}$. Moreover, for the sake of simplicity throughout, denote $n = |\mathcal{V}|$ standing for the number of vertices of $\mathcal{G}$, and let us simply put $\mathcal{V} = \big\{1, \ldots, n\big\}$. Next, let $\mathcal{L}$ stand for a discrete label-set which we assume, without loss of generality, to be the integer set $\big\{0, \ldots, L-1\big\}$, with L standing for an integer which is greater than, or equal to 2, and whatever a hypervertex of $\mathcal{G}$ (i.e.; a subset of its vertices) denoted by s, and whatever an integer vector $x \in \mathcal{L}^n$, denote $x_s = (x_i)_{i \in s}$. Next, suppose $\mathcal{X} = \big(\mathcal{X}_i\big)_{i \in \mathcal{V}}$ is a Markov random field with respect to graph $\mathcal{G}$ taking values in $\mathcal{L}^n$ and of which probability distribution is given by [2, 3]:

$$\mathbb{P}\big(\mathcal{X} = x\big) = \tfrac{1}{Z} \Pi_{c \in \mathcal{C}}\, \phi_c(x_c),\ \forall x \in \mathcal{L}^n \tag{1}$$

where:

- $\phi_c : \mathcal{L}^{|c|} \longrightarrow \mathbb{R}, \forall c \in \mathcal{C}$ stand for strictly positive local functions (also called clique-potential functions),
- Z (called the partition function) stands for a normalization constant which is defined, in such a way that, $\sum_{x \in \mathcal{L}^n} \mathbb{P}\big(\mathcal{X} = x\big) = 1$.

Then, the problem that we propose to solve in the present paper consists in computing, under rather general assumptions about clique-potential functions $\phi_c(\cdot), \forall c \in \mathcal{C}$, such a partition function Z which is explicitly given by the formula:

$$Z = \sum_{x \in \mathcal{L}^n} \big(\Pi_{c \in \mathcal{C}}\, \phi_c(x_c)\big) \tag{2}$$

We want then to emphasize that, though explicit knowledge of the value of Z is not important for achieving some tasks, such as sampling from a general MRF distribution using the MCMC sampling algorithm [4, 5, 6, 7], or even for

computing exactly its marginals with respect to a given collection of hypervertices of $\mathcal{G}$ [1], exact knowledge of the value of Z is, nonetheless, essential for performing several other tasks, such as:

1. evaluation of the MRF pointwise probabilities $\mathbb{P}\big(\mathcal{X} = x\big), \forall x \in \mathcal{L}^n$ (of course, in practice, this may only be achieved with respect to a polynomially-sized subset of $\mathcal{L}^n$),
2. estimation of the Kullback-Leibler divergence between the distribution of MRF $\mathcal{X}$ and a target joint distribution $p(x), \forall x \in \mathcal{L}^n$ (e.g.; another MRF distribution) as:

$$\begin{aligned}\mathbf{Div}_{KL}(\mathcal{X}\|\mathcal{Y}) &= \textstyle\sum_{x\in\mathcal{L}^n} p(x)\ \log\Big(\frac{p(x)}{\mathbb{P}\big(\mathcal{X}=x\big)}\Big) \\ &= -\mathbf{Entropy}(p) - \textstyle\sum_{c\in\mathcal{C}}\sum_{x_c\in\mathcal{L}^{|c|}} p_c(x_c)\log\big(\phi_c(x_c)\big) + \log(Z)\end{aligned}$$

   where $\mathbf{Entropy}(p)$ stands for the entropy of p, and $\forall c \in \mathcal{C}$, $p_c : \mathcal{L}^{|c|} \rightarrow \mathbb{R}$ stands for the marginal of p with respect to c (assuming, of course, that one knows how to compute them efficiently),
3. estimation of the entropy of a MRF distribution, as one may write:

$$\begin{aligned}\mathbf{Entropy}(\mathbb{P}(\mathcal{X})) &= -\textstyle\sum_{x\in\mathcal{L}^n}\mathbb{P}\big(\mathcal{X}=x\big)\log\big(\mathbb{P}\big(\mathcal{X}=x\big)\big) \\ &= -\textstyle\sum_{x\in\mathcal{L}^n}\mathbb{P}\big(\mathcal{X}=x\big)\big(\sum_{c\in\mathcal{C}}\log\big(\phi_c(x_c)\big) - \log(Z)\big) \\ &= -\textstyle\sum_{c\in\mathcal{C}}\sum_{x_c\in\mathcal{L}^{|c|}}\mathbb{P}\big(\mathcal{X}_c=x_c\big)\log\big(\phi_c(x_c)\big) + \log(Z)\end{aligned}$$

   assuming, of course, that one also knows, by some means, how to compute the MRF marginals of the form $\mathbb{P}\big(\mathcal{X}_c = x_c\big), \forall x_c \in \mathcal{L}^{|c|}, \forall c \in \mathcal{C}$ (see section 4 for more details).

to name a few. As a result, an accurate estimation of either quantities above is bound to an accurate estimation of Z (and of course, of an accurate estimation of the clique-wise marginal MRF distributions as well for computation of the entropy of a MRF model instance). Therefore, the remainder of this paper is devoted to the elaboration of an original approach to the exact and efficient computation of Z in formula (2) under rather general assumptions about MRF clique-potential functions involved.

The remainder of this paper is organized as follows. After a first section of related work, we present the main idea which has motivated the present work, by deriving a recursive formula of the partition function, in such a way that, exact evaluation of the MRF partition function in the general case, amounts to computing the marginal distributions with respect to the augmented neighborhood of an arbitrary MRF model instance. After that, building on the recent work in [1], we develop a method enabling to compute exactly such augmented neighborhood-wise probabilities for an arbitrary instance of a MRF model. Last but not least, we present an efficient iterative linear programming algorithm for computing the partition function of an arbitrary graphical model instance. A general conclusion regarding future work concludes this paper.

2. Related work

Markov random field models (MRFs) [2, 8, 7, 9] are an important class of factorable undirected graphical models [10, 11, 3] enjoying the locality property. Their use in science and engineering, including social network mining, language and speech processing, deep belief networks, computer vision and pattern recognition, protein folding, computational chemistry, statistical physics, to name a few, is proliferous. Nevertheless, the main difficulty with the MRF framework, and more generally, with factorable graphical models, resides in the fact that, generally speaking, inference (including MAP inference, marginal inference, computation of the MRF partition function) in their arbitrary instances is a hard combinatorial problem [12, 13, 14, 15, 16, 17, 18, 19, 20, 21]. This means that, unless $P = NP$, there exists no polynomial-time algorithm which can solve either exactly, or at least, approximately either of the aforementioned inference problems in a general graphical model. As a result, except in particular cases (e.g.; Tree-structured graphs, Associative graphs, Gaussian graphs, MRFs with regular priors,etc) [22, 23, 24, 20, 25, 26], researchers have had to settle for elaboration of heuristical approaches to inference in arbitrary large-scale graphical model instances, ranging from stochastic (using MCMC) [4, 5, 6]

to deterministic heuristical approaches [20, 27, 28].

An important category of approximate inference approaches in general graphical models which has gained tremendous interest among the statistical and the AI research communities is what is called loopy belief propagation (LBP), and often referred to as variational inference approach [1] [29, 11, 30]. Historically, BP has been conceived in order to perform exact inference in tree-structured graphical models [31], but later, it has been later extended in a way which enables it to handle loopy-structured graphical models either exactly (e.g.; in factor graphs [32]), or approximately in general non-tree-structured graphical models. In fact, such a variational version of the BP approach can achieve approximate MAP inference, approximate marginal inference, approximate conditional inference, as well as estimation of the partition function) in an arbitrary graphical model instance, to such an extent that an important amount of research in inference in graphical models is devoted to this type of inference approaches. Its main idea consists in replacing a (hardly inferable) graphical model (e.g.; a general MRF distribution) with a surrogate parametric distribution (with respect to which exact inference is known to be easy) via the minimization of an objective [2] over a set of surrogate parameters, in such a way that, approximate inference in a MRF model instance boils down to exact inference in an optimal surrogate distribution. Optimization in such a category of inference approaches is generally achieved using convex programming, particularly, linear programming [11, 33, 34]. Let us then emphasize that, although good performances on many practical problems pertaining to AI, computer vision, and beyond, have been reported in the literature [35, 36, 33], it is well known that it is possible that a BP based algorithm encounters a tautological situation in which it may run indefinitely without any possibility of further improvements of the solution, and

[1] i.e.; "variational" arises here from the fact that a sought solution is an optimal solution of an optimization problem

[2] e.g.; the Kullback-Leibler divergence between target and surrogate distributions, or any other entropy like criteria for measuring similarity between two joint distributions, possibly, plus some prior terms for enforcing prior knowledge about a solution.

thus, generally resulting in a poor solution. Therefore, several amendments to the BP approach have been proposed in the literature, and such algorithms are grouped under the name "message-passing algorithms". The latter include the tree-reweighted belief propagation (TRBP) algorithm [37], the convergent TRBP [33], convergent message-passing based on geometric programming [38], the dual decomposition algorithm [39, 34], and so forth. Furthermore, in [40, 41], it has been shown that tighter relaxations of the global marginal polytope may result in a significant improvement of the results by combining the conditional gradient method (also called Frank-Wolfe) with standard LP solvers [42, 43]. Also, in [44], it has been demonstrated that the TRW objective may be performed over the (hard) global marginal polytope using the Frank-Wolfe method, yet, it has been reported in [45] that such a method lacks convergence because of the inflation of the entropy terms of the TRW objective, and thus, its modification has also been proposed in the same paper. To cut a long story short, let us just say that more theoretical work on BP in general graphical models is, yet, to be done in order to better understand its reported good practical performances on many AI and computer vision problems, as well as its present limitations.

With that being said, further to the important recent work regarding exact marginal inference in general MRF models using linear programming (LP), the main focus in this paper is extension of such a work in a way which enables computation of the partition function of a general MRF model instance–in an exact and efficient way– by means of a linear programming approach, thereby, achieving an unprecedented characterization of MRF distributions, both in terms of their marginals and pointwise entries.

3. Basic idea of the proposed approach

We propose to begin this section by introducing a few useful definitions that we need to use throughout.

Definition 1 (Neighborhood). *Suppose a vertex $i \in \mathcal{V}$. Then, one defines*

the neighborhood of i (with respect to graph $\mathcal{G}$) as the hypervertex of $\mathcal{G}$ denoted by $ne(i)$ and given by:

$$ne(i) = \cup_{c \ni i} \left\{ j \in c / \{i\} \right\}$$

where $\cup_{c \ni i}$ is a shortcut for $\cup_{c \in \mathcal{C}, s.t., i \in c}$.

Definition 2 (Augmented-neighborhood). *Suppose a vertex $i \in \mathcal{V}$. Then, we call the augmented-neighborhood of i (with respect to graph $\mathcal{G}$) the hypervertex of $\mathcal{G}$ denoted by $ne^{+}(i)$ and given by $ne^{+}(i) = \{i\} \cup ne(i)$.*

Furthermore, denote:

$$\Psi_i(x_{\mathrm{ne}^{+}(i)}) = \Pi_{c \ni i} \, \phi_c(x_c), \, \forall x_{\mathrm{ne}^{+}(i)} \in \mathcal{L}^{|\mathrm{ne}^{+}(i)|}, \, \forall i \in \mathcal{V}$$

and suppose an arbitrary vertex $i \in \mathcal{V}$. Then, one may write:

$$\begin{array}{rcl} \mathbb{P}\big(\mathcal{X} = x\big) & = & \frac{1}{Z} \Big(\Pi_{c \ni i} \, \phi_c(x_c) \Big) \Big(\Pi_{c \not\ni i} \, \phi_c(x_c) \Big) \\ & = & \frac{1}{Z} \Psi_i(x_{\mathrm{ne}^{+}(i)}) \Big(\Pi_{c \not\ni i} \, \phi_c(x_c) \Big), \, \forall x \in \mathcal{L}^n, , \, \forall x \in \mathcal{L}^n \end{array} \tag{3}$$

where $\Pi_{c \not\ni i}$ is a shortcut for $\Pi_{c \in \mathcal{C}, \mathrm{s.t.}, i \notin c}$, in such a way that, one may also write:

$$\Big(\frac{\mathbb{P}\big(\mathcal{X} = x\big)}{\Psi_i(x_{\mathrm{ne}^{+}(i)})} \Big) \, Z = \Big(\Pi_{c \not\ni i} \, \phi_c(x_c) \Big), \, \forall x \in \mathcal{L}^n$$

Finally, by taking the margin of both hand sides the latter formula with respect to $\mathrm{ne}(i)$, one derives:

$$Z = \sum_{x \in \mathcal{L}^n} \Big(\Pi_{c \in \mathcal{C}} \, \phi_c(x_c) \Big) = \frac{\sum_{x_{\neq i} \in \mathcal{L}^{n-1}} \Big(\Pi_{c \not\ni i} \, \phi_c(x_c) \Big)}{\sum_{x_{\mathrm{ne}(i)} \in \mathcal{L}^{|\mathrm{ne}(i)|}} \frac{\mathbb{P}\left(\mathcal{X}_{\mathrm{ne}^{+}(i)} = x_{\mathrm{ne}^{+}(i)} \right)}{\Psi_i(x_{\mathrm{ne}^{+}(i)})}}, \, \forall x_i \in \mathcal{L} \tag{4}$$

Moreover, one finds that the quantity:

$$\sum_{x_{\mathrm{ne}(i)} \in \mathcal{L}^{|\mathrm{ne}(i)|}} \frac{\mathbb{P}\big(\mathcal{X}_{\mathrm{ne}^{+}(i)} = x_{\mathrm{ne}^{+}(i)} \big)}{\Psi_i(x_{\mathrm{ne}^{+}(i)})} \tag{5}$$

is a scalar, which means that it does not depend on x_i, in such a way, that a replacement of x_i in formula (5) with any value in $\mathcal{L}$ always yields the same value of such a quantity (5).

In the remainder of this section, without loss of generality, one may assume that $i = 1$ (i.e.; i is the first vertex of $\mathcal{G}$), and let us consider, accordingly, the random field $\mathcal{X}^{(1)} = (\mathcal{X}_2^{(1)}, \ldots, \mathcal{X}_n^{(1)})$ taking values in $\mathcal{L}^{n-1}$ of which probability distribution is given by:

$$\mathbb{P}\big(\mathcal{X}^{(1)} = x_{\neq 1}\big) = \tfrac{1}{Z_1}\Pi_{c \not\ni 1}\, \phi_c(x_c),\ \forall x_{\neq 1} \in \mathcal{L}^{n-1}$$

where Z_1 stands for the partition function of MRF $\mathcal{X}^{(1)}$ which is given by the formula:

$$Z_1 = \sum_{x_{\neq 1} \in \mathcal{L}^{n-1}} \big(\Pi_{c \not\ni 1}\, \phi_c(x_c)\big)$$

where $x_{\neq 1}$ denotes any vector in $\mathcal{L}^{n-1}$ with respect to the indices in $\mathcal{V}/\{1\}$. Then, by using formula (3) above, one finds that the partition function of MRF $\mathcal{X}$ (i.e.; Z) and the partition function of MRF $\mathcal{X}^{(1)}$ (i.e.; Z_1) are related to each other via the formula:

$$Z = \frac{Z_1}{\sum_{x_{\mathrm{ne}(1)} \in \mathcal{L}^{|\mathrm{ne}(1)|}} \frac{\mathbb{P}\left(\mathcal{X}_{\mathrm{ne}^+(1)} = x_{\mathrm{ne}^+(1)}\right)}{\Psi_1(x_{\mathrm{ne}^+(1)})}},\ \forall x_1 \in \mathcal{L} \tag{6}$$

which also writes (so as to better exhibit its iterative nature) as:

$$\sum_{x \in \mathcal{L}^n} \big(\Pi_{c \in \mathcal{C}}\, \phi_c(x_c)\big) = \frac{\sum_{x_{\neq 1} \in \mathcal{L}^{n-1}} \big(\Pi_{c \not\ni 1}\, \phi_c(x_c)\big)}{\sum_{x_{\mathrm{ne}(1)} \in \mathcal{L}^{|\mathrm{ne}(1)|}} \frac{\mathbb{P}\left(\mathcal{X}_{\mathrm{ne}^+(1)} = x_{\mathrm{ne}^+(1)}\right)}{\Psi_1(x_{\mathrm{ne}^+(1)})}},\ \forall x_1 \in \mathcal{L} \tag{7}$$

in such a way that, if one knew (by some oracle) what the following neighborhood-wise marginal probabilities of MRF $\mathcal{X}$ with respect to the first vertex of its graph:

$$\mathbb{P}\big(\mathcal{X}_{\mathrm{ne}^+(1)} = x_{\mathrm{ne}^+(1)}\big),\ \forall x_{\mathrm{ne}^+(1)} \in \mathcal{L}^{|\mathrm{ne}^+(1)|} \tag{8}$$

are, then, clearly, computing the partition function (Z) of MRF $\mathcal{X}$ would amount, first, to computing the partition function (Z_1) of MRF $\mathcal{X}^{(1)}$, then, to using formula (15) above for computing Z. Thus, denote by $\mathcal{G}^{(1)} = \big(\mathcal{V}^{(1)}, \mathcal{E}^{(1)}\big)$ the graph of MRF $\mathcal{X}^{(1)}$. Then, one finds that $\mathcal{G}^{(1)}$ is nothing else than $\mathcal{G}$ deprived of its first vertex (indexed by 1) and, hence, of all its edges involving its first vertex, said otherwise, one has:

$$\mathcal{V}^{(1)} = \mathcal{V}/\{1\}, \quad \mathcal{E}^{(1)} = \mathcal{E}/\big\{e \in \mathcal{E}, \text{s.t.}, 1 \in e\big\}$$

As a result, if there existed a black-box (or an oracle) which can provide the neighborhood-wise probabilities of the form (8) of an arbitrary instance of a MRF model, one might iteratively compute Z efficiently by using formula (15) above, as it will be further described in more details in section 5. Beforehand, we propose to first describe in section 4 below the approach to the exact computation of the MRF neighborhood-wise marginal probabilities of the form (8).

4. Exact computation of neighborhood-wise MRF marginal probabilities

Following the arguments that we have developed in section 3 (see also section 5 for more details), thus, in this section, we describe a linear programming approach to the efficient computation of the neighborhood-wise marginal probabilities of the form (8) in an arbitrary MRF model instance. Therefore, without loss of generality, one may assume MRF $\mathcal{X}$, and the first vertex of $\mathcal{G}$ (i.e.; the one being indexed by 1), and we propose to efficiently evaluate the following marginal probabilities:

$$\mathbb{P}\big(\mathcal{X}_{\mathrm{ne}^+(1)} = x_{\mathrm{ne}^+(1)}\big),\ \forall x_{\mathrm{ne}^+(1)} \in \mathcal{L}^{|\mathrm{ne}^+(1)|} \tag{9}$$

then, by induction, one may be able to also compute them for an arbitrary MRF instance, and for any vertex i of the graph of the latter. For such a purpose, thus, in subsection 4.1, we introduce some useful definitions, whereas subsection 4.2 is devoted to the presentation of a linear programming approach to the exact computation of the probabilistic quantities (9).

4.1. Some definitions and notations

Definition 3 (Neighborhood-wise conditional probabilities). *Let us introduce the local functions:*

$$\begin{aligned}\pi_i(x_{ne^+(i)}) &= \mathbb{P}\big(\mathcal{X}_i = x_i / \mathcal{X}_{ne(i)} = x_{ne(i)}\big) \\ &= \frac{\Pi_{c\in\mathcal{C},s.t.,i\in c}\phi_c(x_c)}{\sum_{x_i\in\mathcal{L}}\big(\Pi_{c\in\mathcal{C},s.t.,i\in c}\phi_c(x_c)\big)},\ \forall x_{ne^+(i)} \in \mathcal{L}^{|ne^+(i)|},\ \forall i \in \mathcal{V}\end{aligned}$$

called the neighborhood-wise conditional probabilities of MRF $\mathcal{X}$.

Moreover, in the light of the work in [1], let us introduce:

- the hypervertex-set:

$$\mathcal{S}_1^+ = \left\{ \mathrm{ne}^+(1) \cup \mathrm{ne}^+(i),\, \forall i \in \mathcal{V} \right\}$$

standing for the set of hypervertices of $\mathcal{G}$ consisting of all the unions of any of its augmented-neighborhoods $\mathrm{ne}^+(i)$, $\forall i \in \mathcal{V}$ with the augmented-neighborhood $\mathrm{ne}^+(1)$ of vertex 1. Furthermore, let us convene, henceforth, that, $\forall t \in \mathcal{S}^+$, the notation $t \doteq \mathrm{ne}^+(1) \cup \mathrm{ne}^+(i)$ is a shortcut for "t is a hypervertex which is originated from the union of hypervertices $\mathrm{ne}^+(1)$ and $\mathrm{ne}^+(i)$, with $\mathrm{ne}^+(i)$ standing for the augmented-neighborhood of some vertex i of $\mathcal{G}$, and $\mathrm{ne}^+(1)$ standing for the augmented-neighborhood of the first vertex of $\mathcal{G}$".

- the real-valued functions $\theta_t : \mathcal{L}^{|t|} \to \mathbb{R}$, $\forall t \in \mathcal{S}_1^+$ defined as:

$$\theta_t(x_t) = \frac{-1 + L\, \pi_i(x_{\mathrm{ne}^+(i)})}{L^{|t|}},\ \forall x_t \in \mathcal{L}^{|t|},\ \forall t \doteq \mathrm{ne}^+(1) \cup \mathrm{ne}^+(i) \in \mathcal{S}_1^+$$

and please note that, $\forall t \doteq \mathrm{ne}^+(1) \cup \mathrm{ne}^+(i) \in \mathcal{S}_1^+$, function $\theta_t(x_t)$ solely depends on $x_{\mathrm{ne}^+(i)}$,

- the signed Boolean functions $\nu_t : \mathcal{L}^{|t|} \to \{-1, 0, +1\}$, $\forall t \in \mathcal{S}_1^+$ defined as:

$$\forall t \in \mathcal{S}_1^+,\ \nu_t(x_t) = \begin{cases} 0, & \text{if } \theta_t(x_t) = 0, \\ +1, & \text{if } \theta_t(x_t) > 0, \\ -1, & \text{if } \theta_t(x_t) < 0. \end{cases} \tag{10}$$

- the four (local) sets of integer vectors, respectively, denoted by Λ_t^0, Λ_t^+, Λ_t^-, and $\Lambda_t^\pm$ and defined as:

$$\forall t \in \mathcal{S}_1^+,\ \begin{cases} \Lambda_t^0 = \left\{ x_t \in \mathcal{L}^{|t|},\ \text{s.t.}, \theta_t(x_t) = 0 \right\}, \\ \Lambda_t^+ = \left\{ x_t \in \mathcal{L}^{|t|},\ \text{s.t.}, \theta_t(x_t) > 0 \right\}, \\ \Lambda_t^- = \left\{ x_t \in \mathcal{L}^{|t|},\ \text{s.t.}, \theta_t(x_t) < 0 \right\}, \\ \Lambda_t^\pm = \Lambda_t^+ \cup \Lambda_t^-. \end{cases}$$

4.2. Exact computation of the neighborhood-wise marginal probabilities

Assume MRF $\mathcal{X}$, and consider the following linear program:

$$\min\Big\{\tau+\sum_{t\doteq \mathrm{ne}^+(1)\cup\mathrm{ne}^+(i)\in\mathcal{S}_1^+}\sum_{x_t\in\mathcal{L}^{|t|}}\nu_t(x_t)\big(q_t(x_t)-\pi_i(x_{\mathrm{ne}^+(i)})\sum_{x_i\in\mathcal{L}}q_t(x_t)\big)\Big\}$$
$$\begin{cases}\tau\geq 1\\ \nu_t(x_t)\big(q_t(x_t)-\pi_i(x_{\mathrm{ne}^+(i)})\sum_{x_i\in\mathcal{L}}q_t(x_t)\big)\geq 0,\ \forall x_t\in\Lambda_t^{\pm},\forall t\doteq \mathrm{ne}^+(1)\cup\mathrm{ne}^+(i)\in\mathcal{S}_1^+\\ q_t(x_t)-\pi_i(x_{\mathrm{ne}^+(i)})\sum_{x_i\in\mathcal{L}}q_t(x_t)=0,\ \forall x_t\in\Lambda_t^0,\forall t\doteq \mathrm{ne}^+(1)\cup\mathrm{ne}^+(i)\in\mathcal{S}_1^+\\ \sum_{x_t\in\mathcal{L}^{|t|}}q_t(x_t)=\tau,\ \forall t\in\mathcal{S}_1^+\\ q_t(x_t)\geq 0,\ \forall x_t\in\mathcal{L}^{|t|},\ \forall t\in\mathcal{S}_1^+\\ \sum_{i\in c/t}\sum_{x_i\in\mathcal{L}}q_t(x_t)=\sum_{i\in t/c}\sum_{x_i\in\mathcal{L}}q_c(x_c),\forall x_{t\cap c}\in\mathcal{L}^{|t\cap c|},\forall t,c\in\mathcal{S}_1^+,\text{s.t.},t\cap c\neq\emptyset\end{cases}\tag{11}$$

and please note that, whatever a feasible solution of LP (11) denoted by $\big(\tau,(q_t(x_t))_{\forall x_t\in\mathcal{L}^t,\forall t\in\mathcal{S}_1^+}\big)$, the set of local nonnegative functions $\frac{q_t(\cdot)}{\tau}:\mathcal{L}^{|t|}\to\mathbb{R}^+$, $\forall t\in\mathcal{S}_1^+$ are locally consistent (i.e.; they satisfy the local-consistency constraints, and they all sum up to a unity), thereby, defining a nonnegative pseudo-marginals-set [37, 11, 1]. Then, Theorem 1 below guarantees that one may be able to compute exactly such neighborhood-wise marginal probabilities (9) by means of LP (11).

Theorem 1. *Denote by* $\big(\tau^*,(q_t^*)_{\forall t\in\mathcal{S}_1^+}\big)$ *the optimal solution of LP (11). Then,* $\forall t\doteq ne^+(1)\cup ne^+(i)\in\mathcal{S}_1^+$*, one has :*

$$\sum_{j\in t/ne^+(1)}\sum_{x_j\in\mathcal{L}}q_t^*(x_t)=\mathbb{P}(\mathcal{X}_{ne^+(1)}=x_{ne^+(1)}),\ \forall x_{ne^+(1)}\in\mathcal{L}^{|ne^+(1)|}\tag{12}$$

The proof of Theorem 1 is sketched in appendix section Appendix A.

Let us then note that, in plain language, formula (12) means that $\forall i\in\mathcal{V}$, if one takes the margin of the local function $q^*_{\mathrm{ne}^+(1)\cup\mathrm{ne}^+(i)}(x_{\mathrm{ne}^+(1)\cup\mathrm{ne}^+(i)})$, $\forall x_{\mathrm{ne}^+(1)\cup\mathrm{ne}^+(i)}\in\mathcal{L}^{|\mathrm{ne}^+(1)\cup\mathrm{ne}^+(i)|}$ with respect to $\mathrm{ne}^+(1)$ (i.e.; by summing it up with respect to all the variables of which indices belong to $\mathrm{ne}^+(i)$ but not to $\mathrm{ne}^+(1)$), one obtains $\mathbb{P}(\mathcal{X}_{\mathrm{ne}^+(1)}=x_{\mathrm{ne}^+(1)})$ which is, thus, nothing else than the marginal distribution of MRF $\mathcal{X}$ with respect to the augmented neighborhood of the first vertex of graph $\mathcal{G}$ (being indexed by 1).

We conclude this section by just noting that the system of local consistency constraints in LP (11) of the form:

$$\begin{cases} \sum_{x_t \in \mathcal{L}^{|t|}} q_t(x_t) = \tau, \, \forall t \in \mathcal{S}_1^+ \\ \sum_{i \in c/t} \sum_{x_i \in \mathcal{L}} q_t(x_t) = \sum_{i \in t/c} \sum_{x_i \in \mathcal{L}} q_c(x_c), \forall x_{t \cap c} \in \mathcal{L}^{|t \cap c|}, \forall t, c \in \mathcal{S}_1^+, \text{s.t.}, t \cap c \neq \emptyset \end{cases} \tag{13}$$

may present many redundancies, and thus, may further undergo simplifications for the purpose of reducing the number of the linear constraints defining it (i.e.; while ensuring total equivalence between such a reduced system and system (13)), thereby, obtaining a more optimized LP in terms of number of linear constraints. Nevertheless, for the sake of keeping the present paper reader-friendly, we refer the interested reader to [1] for more hints on the latter point.

5. Algorithm for computing the MRF partition function

In this section, we describe an algorithm which allows to compute the partition function of MRF $\mathcal{X}$ by iteratively linear programs with decreasing complexity. Therefore, assume MRF $\mathcal{X}$ and its graph $\mathcal{G}$ which we have already defined in the introductory section, and introduce the graph series $\mathcal{G}^{(i)} = (\mathcal{V}^{(i)}, \mathcal{E}^{(i)})$, $\forall i = 0, \ldots, n-1$ which may be iteratively defined as:

$$\begin{cases} \mathcal{V}^{(0)} = \mathcal{V}, \quad \mathcal{E}^{(0)} = \mathcal{E} \\ \forall i = 1, \ldots, n-1, \begin{cases} \mathcal{V}^{(i)} = \mathcal{V}^{(i-1)} / \{i\} = \{i+1, \ldots, n\} \\ \mathcal{E}^{(i)} = \mathcal{E}^{(i-1)} / \{e \in \mathcal{E}^{(i)}, \text{s.t.}, i \in e\} = \{e \in \mathcal{E}, \text{s.t.}, j \notin e, \forall j = 1, \ldots, i\} \end{cases} \end{cases}$$

and please note the recursion in the latter system, in the sense that, graph $\mathcal{G}^{(i)}$ is defined as the graph resulting from the pruning of vertex i from graph $\mathcal{G}^{(i-1)}$, $\forall i = 0, \ldots, n-1$. Moreover, denote $\mathcal{C}^{(i)}$ the maximal-clique system of $\mathcal{G}^{(i)}$, $\forall i = 0, \ldots, n-1$ which may also be given iteratively by:

$$\begin{cases} \mathcal{C}^{(0)} = \mathcal{C} \\ \mathcal{C}^{(i)} = \{c \in \mathcal{C}^{(i-1)}, \text{s.t.}, i \notin c\} = \{c \in \mathcal{C}, \text{s.t.}, j \notin c, \forall j = 1, \ldots, i\}, \, \forall i = 1, \ldots, n-1 \end{cases}$$

Finally, denote respectively by $ne_i^{(i)}$ and by $ne_i^{(i+)}$ the neighborhood and the augmented-neighborhood of vertex i with respect to graph $\mathcal{G}^{(i)}$, $\forall i = 0, \ldots, n-1$.

Now, suppose i_0 is any integer number which is greater than, or equal to 1, which we, moreover, assume to be much smaller than n, i.e.; either i_0 is a constant (equal to 1, 2, etc) which is independent of n, or at most, i_0 is some polylogarithmic function of n which means that there exists a nonnegative real constant k such that $i_0 = O\big(\log^k(n)\big)$. Shortly, we will see, that i_0 stands for the number of nodes of the smallest MRF graph which will be considered in the iterative process to the computation of Z, hence, the constraints above regarding its desirable values, and in practice, the choice of the value i_0 is, simply, a matter of convenience. Next, consider the MRF series $\mathcal{X}^{(0)} := \mathcal{X}$, $\mathcal{X}^{(i)}, \forall i = 1, \ldots, n - i_0$ taking values in $\mathcal{L}^{n-i}, \forall i = 0, \ldots, n - i_0$ defined with respect to the graph series $\mathcal{G}^{(i)}, \forall i = 0, \ldots, n - i_0$, and of which probability distributions are, respectively, defined as:

$$\mathbb{P}\big(\mathcal{X}^{(i)} = x_{\{i+1,\ldots,n\}}\big) = \tfrac{1}{Z_i}\Pi_{c\in\mathcal{C}^{(i)}}\,\phi_c(x_c),\ \forall x_{\{i+1,\ldots,n\}} \in \mathcal{L}^{n-i},\ \forall i = 0, \ldots, n - i_0$$

where $\forall i = 0, \ldots, n-i_0$, Z_i stands for the partition function of MRF $\mathcal{X}^{(i)}$ which is given by the formula:

$$Z_i = \sum_{x_{\{i+1,\ldots,n\}}\in\mathcal{L}^{n-i}} \big(\Pi_{c\in\mathcal{C}^{(i)}}\,\phi_c(x_c)\big)$$

where $x_{\{i+1,\ldots,n\}}$ denotes any vector $\in \mathcal{L}^{n-i}$ with respect to the indices in $\{i + 1, \ldots, n\}$. Therefore, $\forall i = 0, \ldots, n - i_0$, also put:

$$\Psi^{(i)}(x_{\mathrm{ne}_i^{(i+)}}) = \Pi_{c\in C^{(i)},\mathrm{s.t.},i\in c}\,\phi_c(x_c),\ \forall x_{\mathrm{ne}_i^{(i+)}} \in \mathcal{L}^{|\mathrm{ne}_i^{(i+)}|}$$

Then, by using formula (15) we have established in section 3, one derives:

$$Z_i = \frac{Z_{i+1}}{\sum_{x_{\mathrm{ne}_i^{(i)}}\in\mathcal{L}^{|ne_i^{(i)}|}} \frac{\mathbb{P}\left(\mathcal{X}^{(i)}_{ne_i^{(i+)}} = x_{ne_i^{(i+)}}\right)}{\Psi^{(i)}(x_{ne_i^{(i+)}})}},\ \forall x_i \in \mathcal{L},\ \forall i = 0, \ldots, n - i_0 - 1 \quad (14)$$

In addition, for the same reasons as already evoked in section 3, one finds that the quantities:

$$\sum_{x_{\mathrm{ne}_i^{(i)}}\in\mathcal{L}^{|ne_i^{(i)}|}} \frac{\mathbb{P}\big(\mathcal{X}^{(i)}_{ne_i^{(i+)}} = x_{ne_i^{(i+)}}\big)}{\Psi^{(i)}(x_{ne_i^{(i+)}})},\ \forall x_i \in \mathcal{L},\ \forall i = 0, \ldots, n - i_0 \quad (15)$$

are scalar (i.e.; constant) quantities. Also, one notes, in particular, the recursive nature of formula (14), in the sense that, $\forall i = 0, \ldots, n-i_0-1$, for computing the partition function Z_i of MRF $\mathcal{X}^{(i)}$, one may first compute the partition function Z_{i+1} of MRF $\mathcal{X}^{(i+1)}$, as well as the neighborhood-wise marginal distributions of MRF $\mathcal{X}^{(i)}$ with respect to the first vertex of its graph $\mathcal{G}^{(i)}$ (assumed, here, to be the one corresponding to the vertex indexed by i in graph $\mathcal{G}$)–as it has been described in section 4– by solving a linear program of the form (11) (after doing the necessary changes), finally, one may use formula (14) above for computing exactly Z_i.

Then, the algorithm for computing $Z_0 = Z$ proceeds as follows. One spans the values of i in decreasing order, starting from $i = n-i_0$, then decreasing i until termination when $i = 0$ is reached. Moreover, at the first iteration of the process (then, $i = n - i_0$), one computes the partition function Z_{n-i_0} of MRF $\mathcal{X}^{(n-i_0)}$, then at each subsequent iteration of the process ($i = n - i_0 - 1, \ldots, 0$), one computes the neighborhood-wise marginal distributions of MRF $\mathcal{X}^{(i)}$ by solving an instance of LP (11), and one uses formula (14) for computing the partition function Z_i of MRF $\mathcal{X}^{(i)}$, in such a way that at termination of the algorithm, one recovers $Z_0 = Z$ which is nothing else than the partition function of MRF $\mathcal{X}$.

6. Conclusion

By building on a recent work on exact marginal inference in general MRF models using linear programming [1], and by deriving a recursive formula of the MRF partition function, we have been able to develop an iterative linear programming approach which enables to compute the partition function of an arbitrary MRF distribution both in an exact and efficient way. Roughly speaking (see section 5 for a more accurate statement) the proposed method needs solve as many instances as number of vertices of a MRF graph, yet, with minus one vertex at each new iteration of the iterative process. Therefore, we believe that it is also worthwhile investigating, in the near future, further method

acceleration schemes. This may concern, for instance, exploitation, somehow, of the existing redundancy (or consistency) between neighborhood-wise MRF marginal distributions, as well as the (logarithmic-like) reduction of number of LP variables in non-binary MRF models (in terms of number of labels), overall, for bigger values of the number of labels L, by converting an instance of a non-binary MRF model into an equivalent instance of a binary MRF model.

References

[1] I. Bechar, Exact marginal inference in general higher-order graphical models using linear programming, Preprint, available in PDF format at: http://vixra.org/abs/1709.0243.

[2] J. Hammersley, P. Clifford, Markov fields on finite graphs and lattices, Unpublished.

[3] D. Koller, N. Friedman, Probabilistic graphical models: principles and techniques, MIT Press.

[4] W. Hastings, Monte carlo sampling methods using markov chains and their applications, Biometrika 97-109 (1970) 57.

[5] P. Green, Reversible jump markov chains monte carlo computation and bayesian model determination, Biometrika 82 (1995) 711–732.

[6] A. Gelfand, A. Smith, Sampling-based approaches to calculating marginal densities, Journal of the American Statistical Association 85 (1990) 398–409.

[7] S. Geman, D. Geman, Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images, IEEE Trans. Patt. Anal. Mach. Intell. (T-PAMI) 6 (1984) 721–741.

[8] J. Besag, Spatial interaction and the statistical analysis of lattice systems (with discussion), J. Roy. Statist. Soc. Ser. B 36 (1974) 192–236.

[9] J. Besag, On the statistical analysis of dirty pictures, J. Roy. Statist. Soc. Ser. B 48 (3) (1986) 259–302.

[10] S. L. Lauritzen, Graphical models, Oxford Statistical Science Series, Oxford Science Publications 17.

[11] M. Wainwright, M. Jordan, Graphical models, exponential families, and variational inference, Foundations and Trends in Machine Learning 1 (1–2) (2008) 1–305.

[12] R. Karp, Reducibility among combinatorial problems, Complexity of Computer Computations, Plenum Press, New York (1972) 85–103.

[13] G. Cooper, The computational complexity of probabilistic inference using Bayesian belief networks, Artificial Intelligence 42 (1990) 393–405.

[14] P. Dagum, M. Luby, Approximating probabilistic inference in bayesian belief networks is NP–hard, Artificial Intelligence 60 (1993) 141–153.

[15] Y. Shimony, Finding the maps for belief networks is NP–hard, Aritifical Intelligence 68 (2) (1994) 399–410.

[16] D. Roth, On the hardness of approximate reasoning, Artificial Intelligence 82 (273–302).

[17] D. M. Chickering, Learning Bayesian networks is NP–Complete, Fisher, D. and Lenz, H., editors, Learning from Data: Artificial Intelligence and Statistics, Springer-Verlag 5 (121130).

[18] B. A. Cipra, The ising model is NP–Complete, SIAM News 33 (6).

[19] A. Megretski, Relaxations of quadratic programs in operator theory and system analysis, Systems, approximation, singular integral operators, and related topics 3.

[20] Y. Boykov, O. Veksler, R. Zabih, Fast approximate energy minimization via graph cuts, IEEE Trans. Patt. Anal. Mach. Intell. (T-PAMI) 23 (11) (2001) 1222–1239.

[21] D. Cohen, M. Cooper, P. Jeavons, A. Krokhin, The complexity of soft constraint satisfaction, Artificial Intelligence 170 (2006) 983–1016.

[22] D. Greig, B. Porteous, A. Seheult, Exact a posteriori maximum estimation in binary images, J. Royal Statistical Soc. Series B 5 (1989) 271–279.

[23] Y. Boykov, O. Veksler, R. Zabih, Markov random fields with efficient approximations, in: Proceedings of CVPR, 1998.

[24] H. Ishikawa, Exact optimization for markov random fields with convex priors, IEEE Trans. Pattern Anal. Mach. Intell. 25 (2003) 1333–1336.

[25] D. Schlesinger, Exact solution of permuted submodular minsum problems, in: Proceedings of EMMCVPR, 2007.

[26] A. Osokin, D. Vetrov, V. Kolmogorov, Sub-modular decomposition framework for inference in associative markov networks with global constraints, in: Proceedings of CVPR, 2011.

[27] D. Cremers, T. T. Pock, K. Kolev, A. Chambolle, Convex Relaxation Techniques for Segmentation, Stereo and Multiview Reconstruction, MIT Press, 2011.

[28] O. Veksler, Multi-label moves for mrfs with truncated convex priors, International Journal of Computer Vision 98 (2012) 1–14.

[29] J. Yedidia, W. Freeman, Y. Weiss, Understanding Belief Propagation and its generalizations, Exploring Artificial Intelligence in the New Millennium (Chap. 8) (2003) 239–269.

[30] J. Yedidia, Y. Wang, S. Draper, Divide & Concur and difference-map BP decoders for ldpc codes, IEEE Trans. on Information Theory 57 (2) (2011) 786–802.

[31] J. Pearl, Reverend bayes on inference engines: A distributed hierarchical approach, Proc. Sec. Nat. Conf. Artif. Intell., AAAI-82, Pittsburgh, PA. Menlo Park, California (1982) 133–136.

[32] F. Kschischang, J. Frey, H. Loeliger, Factor graphs and the sum-product algorithm, Journal of the American Statistical Association 47 (2001) 498519.

[33] V. Kolmogorov, Convergent tree-reweighted message passing for energy minimization, IEEE Trans. Patt. Anal. Mach. Intell. (T-PAMI) 28 (10) (2006) 1568–1583.

[34] N. Komodakis, N. Paragios, G. Tziritas, Mrf energy minimization and beyond via Dual Decomposition, IEEE Trans. Patt. Anal. Mach. Intell. (T-PAMI) 33 (3) (2011) 531–552.

[35] A. Baddeley, M. V. Lieshout, Stochastic geometry models in high-level vision, Journal of Applied Statistics 1 (1993) 231–256.

[36] G. Winkler, Image Analysis, Random Fields and Markov Chain Monte Carlo Methods: A Mathematical Introduction, Springer, 1995.

[37] M. Wainwright, T. Jaakkola, A. Willsky, Map estimation via agreement on (hyper)trees: Message-passing and linear-programming approaches, IEEE Trans. on Information Theory 51 (11) (2005) 3697–3717.

[38] A. Globerson, T. Jaakkola, Convergent propagation algorithms via oriented trees, Uncertainty in Artificial Intelligence (UAI).

[39] J. Jancsary, G. Matz, Convergent decomposition solvers for tree-reweighted free energies, AISTATS.

[40] D. Sontag, T. Jaakkola, New outer bounds on the marginal polytope, Advances in Neural Information Processing Systems, MIT Press 21.

[41] D. Sontag, T. Meltzer, A. Globerson, T. Jaakkola, Y. Weiss, Tightening lp relaxations for MAP using Message Passing, Uncertainty in Artificial Intelligence (UAI).

[42] M. Jaggi, Revisiting Frank-Wolfe: Projection-free sparse convex optimization, Int. Conf. Mach. Learning (ICML).

[43] S. Lacoste-Julien, M. Jaggi, On the global linear convergence of Frank-Wolfe optimization variants, Adv. Neur. Inf. Proc. Sys. (NIPS).

[44] D. Belanger, D. Sheldon, A. McCallum, Marginal inference in mrfs using Frank-Wolfe, NIPS Workshop on Greedy Optimization, Frank-Wolfe and Friends.

[45] R. G. Krishnan, S. Lacoste-Julien, D. Sontag, Barrier Frank-Wolfe for Marginal Inference, Adv. Neur. Inf. Proc. Sys. (NIPS) 28.

Appendix A. Proof of Theorem 1

We need the result of Lemma 1 below for proving Theorem 1. Therefore, first, without loss of generality, one may assume the Markov random field $\mathcal{X}$ we have already introduced in the introductory section of this paper. After that, suppose $\mathcal{S}$ is an arbitrary hypervertex-set with respect to graph $\mathcal{G}$, and denote:

$$\mathcal{T}^+ = \big\{s \cup \mathrm{ne}^+(i),\, \forall s \in \mathcal{S},\, \forall i \in \mathcal{V}\big\}$$

standing for the set of hypervertices of $\mathcal{G}$ consisting of all the unions of any of its augmented-neighborhoods $\mathrm{ne}^+(i)$, $\forall i \in \mathcal{V}$ with any hypervertex s of $\mathcal{S}$, and introduce:

- the real-valued functions $\theta'_t : \mathcal{L}^{|t|} \to \mathbb{R}$, $\forall t \in \mathcal{T}^+$ defined as:

$$\theta'_t(x_t) = \tfrac{-1+L\,\pi_i(x_{\mathrm{ne}^+(i)})}{L^{|t|}},\ \forall x_t \in \mathcal{L}^{|t|},\ \forall t \in \mathcal{T}^+$$

  and please note that, $\forall t \in \mathcal{T}^+$, function $\theta'_t(x_t)$ solely depends on $x_{\mathrm{ne}^+(i)}$,

- the signed Boolean functions $\nu'_t : \mathcal{L}^{|t|} \to \{-1, 0, +1\}$, $\forall t \in \mathcal{T}^+$ defined as:

$$\forall t \in \mathcal{T}^+,\ \nu'_t(x_t) = \begin{cases} 0, & \text{if } \theta'_t(x_t) = 0, \\ +1, & \text{if } \theta'_t(x_t) > 0, \\ -1, & \text{if } \theta'_t(x_t) < 0. \end{cases} \tag{A.1}$$

- the four sets of (local) integer vectors, respectively, denoted by $\Lambda_t^{0\prime}$, $\Lambda_t^{+\prime}$, $\Lambda_t^{-\prime}$, and $\Lambda_t^{\pm\prime}$ and defined as:

$$\forall t \in \mathcal{T}^+, \begin{cases} \Lambda_t^{0\prime} = \big\{x_t \in \mathcal{L}^{|t|},\ \text{s.t.}, \theta_t'(x_t) = 0\big\}, \\ \Lambda_t^{+\prime} = \big\{x_t \in \mathcal{L}^{|t|},\ \text{s.t.}, \theta_t'(x_t) > 0\big\}, \\ \Lambda_t^{-\prime} = \big\{x_t \in \mathcal{L}^{|t|},\ \text{s.t.}, \theta_t'(x_t) < 0\big\}, \\ \Lambda_t^{\pm\prime} = \Lambda_t^{+\prime} \cup \Lambda_t^{-\prime}. \end{cases}$$

Furthermore, convene that $\forall t \in \mathcal{T}^+$, the notation $t \doteq s \cup \text{ne}^+(i)$ stands for "t is a hypervertex which is originated from the union of hypervertices s and $\text{ne}^+(i)$, with s standing for some hypervertex in $\mathcal{S}$, and $\text{ne}^+(i)$ standing for the augmented-neighborhood of some vertex i of $\mathcal{G}$".

Then, consider the following LP:

$$\begin{aligned} &\min \Big\{\tau + \textstyle\sum_{t \doteq s \cup \text{ne}^+(i) \in \mathcal{T}^+} \sum_{x_t \in \Lambda_t^{\pm\prime}} \nu_t'(x_t)\big(q_t(x_t) - \pi_i(x_{\text{ne}^+(i)}) \sum_{x_i \in \mathcal{L}} q_t(x_t)\big)\Big\} \\ &\begin{cases} \nu_t'(x_t)\big(q_t(x_t) - \pi_i(x_{\text{ne}^+(i)}) \sum_{x_i \in \mathcal{L}} q_t(x_t)\big) \geq 0,\ \forall x_t \in \Lambda_t^{\pm\prime}, \forall t \doteq s \cup \text{ne}^+(i) \in \mathcal{T}^+ \\ q_t(x_t) - \pi_i(x_{\text{ne}^+(i)}) \sum_{x_i \in \mathcal{L}} q_t(x_t) = 0,\ \forall x_t \in \Lambda_t^{0\prime}, \forall t \doteq s \cup \text{ne}^+(i) \in \mathcal{T}^+ \\ \tau \geq 1 \\ q_t(x_t) \geq 0,\ \forall x_t \in \mathcal{L}^{|t|},\ \forall t \in \mathcal{T}^+ \\ \sum_{x_t \in \mathcal{L}^{|t|}} q_t(x_t) = \tau,\ \forall t \in \mathcal{T}^+ \\ q_t(x_t) \geq 0,\ \forall x_t \in \mathcal{L}^{|t|},\ \forall t \in \mathcal{T}^+ \\ \sum_{i \in c/t} \sum_{x_i \in \mathcal{L}} q_t(x_t) = \sum_{i \in t/c} \sum_{x_i \in \mathcal{L}} q_c(x_c), \forall x_{t \cap c} \in \mathcal{L}^{|t \cap c|}, \forall t, c \in \mathcal{T}^+, \text{s.t.}, t \cap c \neq \emptyset \end{cases} \end{aligned} \tag{A.2}$$

Then, we have the following Lemma of which proof is detailed in the paper [1] (Theorem 4).

Lemma 1. *Denote by* $\big(\tau^*, (q_t^*)_{\forall t \in \mathcal{T}^+}\big)$ *the optimal solution of LP (A.2). Then,* $\forall t \doteq s \cup ne^+(i) \in \mathcal{T}^+$*, one has:*

$$\textstyle\sum_{j \in t/s} \sum_{x_j \in \mathcal{L}} q_t^*(x_t) = \mathbb{P}(\mathcal{X}_s = x_s),\ \forall x_s \in \mathcal{L}^{|s|}, \forall s \in \mathcal{S}$$

Then, the proof of Theorem 1 is established immediately by applying Lemma 1 with as hypervertex-set $\mathcal{S} = \big\{\text{ne}^+(1)\big\}$ (hence with $\mathcal{S}^+ = \mathcal{S}_1^+$).