

Recursive sum-product algorithm for generalized outer-planar graphs

Qiang Cheng^{a,*}, Feng Chen^a, Wenli Xu^a, Song Wang^b

^a Tsinghua National Laboratory for Information Science and Technology, Department of Automation, Tsinghua University, Beijing 100084, PR China

^b Department of Computer Science and Engineering, University of South Carolina, Columbia, SC 29208, USA

ARTICLE INFO

Article history:

Received 26 June 2011

Received in revised form 1 March 2012

Accepted 2 March 2012

Available online 6 March 2012

Communicated by J. Torán

Keywords:

Graphical models

Inference algorithm

Approximation algorithms

Generalized outer-planar graph

Recursive sum-product algorithm

ABSTRACT

Inference on cyclic graphs is one of the most important problems in the applications of graphical models. While exact inference is NP-hard on general cyclic graphs, it has been found that exact inference can be achieved with a computational complexity as low as $\mathcal{O}(Nm^3)$ on the outer-planar graph, which is a special kind of cyclic graph. In this paper, we introduce a new kind of cyclic graph, the generalized outer-planar (GOP) graph, which is more general than the outer-planar graph and show that the exact inference on the GOP graphs can be achieved in $\mathcal{O}(Nm^3)$ by a recursive sum-product (RSP) algorithm. RSP exploits the property of GOP graphs that the faces are reducible, and brings a “face elimination” procedure to compute the marginals exactly. Furthermore, RSP can be implemented on general cyclic graphs to obtain approximate marginals. Experimental results show the effectiveness of approximate RSP on various graphical models.

© 2012 Elsevier B.V. All rights reserved.

1. Introduction

Graphical models have been widely studied and used as powerful tools in many fields for modeling uncertainty, such as image processing [1], natural language processing [2], computational biology [3], and statistical machine learning [4]. Inference problems on graphical models, including computing partition functions, marginal distributions and most probable configurations, play an important role in many of these applications [5]. Exact inference on general graphs is NP-hard, and therefore it is usually infeasible to achieve in practice. However, for some special cyclic graphs, exact inference can be done with efficient algorithms. The belief propagation algorithm (BP) is an exact inference algorithm, with $\mathcal{O}(Nm^2)$ complexity (N is the number of variables, and m is the maximal number of states of the variables), for tree-structured graphical models [6]. The junction tree algorithm is an exact inference approach for graphs with cycles, with complexity exponential to the size of the maximal clique in the induced junction tree [4]. Its complexity is $\mathcal{O}(Nm^3)$ for outer-planar

graphs and may be higher for other cyclic graphs. There are many approximate algorithms, such as the loopy belief propagation algorithm, the generalized belief propagation algorithm [7] and the junction graph method [8], which can obtain approximate inference results with lower complexity compared with that of exact inference.

In this paper, we introduce a new kind of cyclic graph, the *generalized outer-planar* (GOP) graph, and develop an exact inference algorithm for such graphs. GOP graphs are more general than outer-planar graphs. The computational complexity of off-the-shelf exact inference methods may be more than $\mathcal{O}(Nm^3)$ on GOP graphs. We propose a *recursive sum-product* (RSP) algorithm for exact inferences on GOP graphs, which can keep the lowest computational complexity, i.e., $\mathcal{O}(Nm^3)$. The property of a GOP graph is that its variables can be eliminated by using two simple operations. Making use of this property, RSP is developed as a “face elimination” procedure, which eliminates the faces of GOP graphs one after another recursively. More specifically, the inference is obtained from each individual face and later integrated by using the structural overlap among different faces in RSP. RSP obtains exact inference on GOP graphs with the complexity of $\mathcal{O}(Nm^3)$, which is

* Corresponding author. Tel.: +86 1062797145; fax: +86 1062786911.

E-mail address: cheng-q09@mails.tsinghua.edu.cn (Q. Cheng).

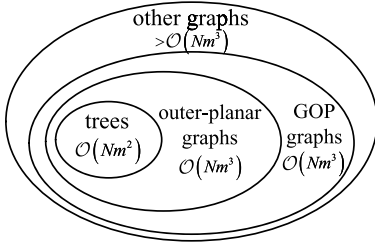


Fig. 1. Relations of different kinds of graphs and the computational complexity on these graphs for exact inference.

lower than the complexities of other exact inference methods.

The relations of different kinds of graphs and the computational complexity on these graphs for exact inference are shown in Fig. 1. Specifically, RSP can obtain exact inference on those GOP graphs, with a lower complexity.

The remainder of this paper is organized as follows. The background and some notations are introduced in Section 2. The sum-product procedure for one cycle is introduced in Section 3. Generalized outer-planar graphs are introduced in Section 4. The recursive sum-product algorithm (RSP) on GOP graphs is developed in Section 5. In Section 6, we develop an approximate version of RSP for general cyclic graphs. In Section 7, experiments on a variety of graphs are conducted to show the effectiveness of approximate RSP.

2. Preliminaries

In this section, we overview the concepts of *outer-planar graph* and *Markov random field* that are important for our algorithm development. We denote an undirected graph by $G = (V, E)$, where $V = \{1, 2, \dots, n\}$ is the set of nodes and $E \subset V \times V$ is the set of edges. When G is used for probabilistic inference, each node in this graph represents a random variable.

2.1. Face and outer-planar graph

We first give the description of the *face* of a planar graph. A planar graph G partitions the rest of the plane into a number of connected open sets, which are called the **faces** of G . Each planar graph has exactly one unbounded face, called the **outer face** or **external face** [9]. In our paper, “face” is used to denote the bounded face if there is no confusion, and “face” also denotes the set of variables on its boundary. Fig. 2 shows four faces f_1, f_2, f_3, f_4 and an outer face f_0 in a planar graph. The outer-planar graph is defined as follows.

An **outer-planar graph** G is a planar graph, where all nodes lie on the outer face [10]. The graph shown in Fig. 5(a) is an outer-planar graph, while the graph in Fig. 5(b) is not an outer-planar graph, since node 5 does not lie on the outer face.

2.2. Graphical models

An undirected graphical model (also known as a Markov random field (MRF)) is a set of random variables (as well

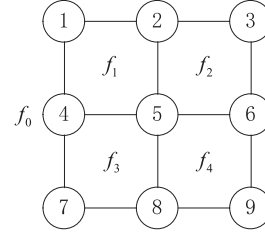


Fig. 2. A planar graph and its faces.

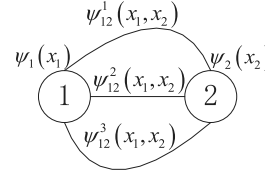


Fig. 3. Illustrations of parallel edges.

as a set of undirected links) that have a Markov property described by the graph G . Let X_s be the random variable related to node $s \in V$ in graph G , and x_s take values in the discrete space $\mathcal{X}_s = \{0, 1, \dots, m_s - 1\}$. Let $\mathbf{x} = \{x_s \mid s \in V\}$ take values in the product space $\mathcal{X}^N = \mathcal{X}_1 \times \mathcal{X}_2 \times \dots \times \mathcal{X}_N$, where $N = |V|$ is the number of nodes in G . $\mathbf{x}$ is called a state of $\mathbf{X}$, where $\mathbf{X} = \{X_s \mid s \in V\}$. A *clique* of a graph G is a subset of nodes in which there exists an edge between each pair of nodes in the subset, and a clique potential is a non-negative real valued function defined on the set of nodes in a clique [5]. The pairwise MRF is an MRF where all the cliques have at most two variables. The probability distribution of a pairwise MRF $G = (V, E)$ can be expressed as a product of clique potentials:

$$p(\mathbf{x}) = \frac{1}{Z_\psi} \prod_{s \in V} \psi_s(x_s) \prod_{(s,t) \in E} \psi_{st}(x_s, x_t), \quad (1)$$

where Z_ψ denotes the partition function, $\psi_s(x_s)$ and $\psi_{st}(x_s, x_t)$ denote the potentials corresponding to the clique $\{x_s\}$ and $\{x_s, x_t\}$.

3. Sum-product on one cycle

If a graph contains only one face (not including the outer face), there is just one cycle in the graph. When using the sum-product procedure for the graph with only one cycle, we first reduce the cycle to parallel edges between two variables of this cycle, and then compute the distribution of these two variables. If there are two or more edges connecting the same two nodes, such edges are called **parallel edges**. One example of parallel edges is shown in Fig. 3, where there are three parallel edges between nodes 1 and 2. The potential functions on parallel edges can be reduced to one potential function on one edge by the dot product:

$$\psi_{st}(x_s, x_t) = \prod_{i \in PE(st)} \psi_{st}^i(x_s, x_t), \quad (2)$$

Download English Version:

<https://daneshyari.com/en/article/427740>

Download Persian Version:

<https://daneshyari.com/article/427740>

[Daneshyari.com](https://daneshyari.com)