Information Processing Letters 112 (2012) 238–242

Contents lists available at SciVerse ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

Computing hypergraph width measures exactly

Lukas Moll [a], Siamak Tazari [b,*,1], Marc Thurley [c,2]

[a] Humboldt Universität zu Berlin, Berlin, Germany
[b] Massachusetts Institute of Technology, Cambridge, MA, United States
[c] Centre de Recerca Matemàtica, Bellaterra, Spain

ARTICLE INFO

Article history:
Received 25 June 2011
Received in revised form 28 November 2011
Accepted 2 December 2011
Available online 6 December 2011
Communicated by B. Doerr

Keywords:
Design of algorithms
Graph algorithms
Generalized hypertree-width
Fractional hypertree-width
Exact exponential algorithms

ABSTRACT

Hypergraph width measures are important in studying the complexity of constraint satisfaction problems (CSPs). We present a general exact exponential algorithm for a large variety of these measures. As a consequence, we obtain algorithms which, for a hypergraph H on n vertices and m hyperedges, compute its generalized hypertree-width in time $O^*(2^n)$ and its fractional hypertree-width in time $O(1.734601^n \cdot m)$.[3]

1. Introduction

Hypergraph width measures originated in studying the complexity of *structural* restrictions of constraint satisfaction problems (CSPs). This approach is different from the well-known approach of restricting CSPs to certain types of constraints (see, for example [3]), in so far as it studies the structure of CSP *instances* [7,9,12]. This structure is usually captured by a hypergraph of the instance. Then, for a class of hypergraphs $\mathcal{H}$ the restricted problem CSP($\mathcal{H}$) allows only instances with hypergraphs in $\mathcal{H}$.

Although in the so-called *bounded arity case* the concept of *bounded tree-width* captures the complexity of CSP($\mathcal{H}$) [10,8], the unbounded arity case is more delicate (cf. [12]). Here, several hypergraph width measures yield larger classes of $\mathcal{H}$ such that CSP($\mathcal{H}$) is tractable such as those of bounded *(generalized) hypertree-width* [7] and bounded *fractional hypertree-width* [9].

We present the first non-trivial exact algorithm for computing any hypertree-width measure defined by some *monotone width function* f. This implies an algorithm for both fractional and generalized hypertree-width by essentially the same means.

Theorem 1. *Let H be a hypergraph on n vertices and m hyperedges.*

(i) *The generalized hypertree-width of H can be computed in $O^*(2^n)$ time.*
(ii) *Its fractional hypertree-width can be computed in $O(m1.734601^n)$ time.*

We will show that for computing the width of a hypergraph H it is sufficient to compute a tree decomposition of its primal graph while measuring the width in terms of H. This enables us to almost seamlessly adapt the combinatorial and algorithmic results of [4,5] and [6] for tree decompositions of graphs. The proof of Theorem 1 will be

* Corresponding author.
E-mail address: siamak.tazari@gmail.com (S. Tazari).

1 Supported by a fellowship within the Postdoc-Programme of the German Academic Exchange Service (DAAD).
2 Supported in part by Marie Curie Intra-European Fellowship 271959.

3 We omit factors polynomial in n whenever the base of the exponent is rounded. This is justified as $c^n \cdot n^{O(1)} = O((c+\epsilon)^n)$ for every $\epsilon > 0$. We also use the notation O^* to suppress polynomial factors.

0020-0190/$ – see front matter
doi:10.1016/j.ipl.2011.12.002

given in Section 3.[4] Note that all of these algorithms require exponential space in the worst case.

2. Preliminaries

Graphs and hypergraphs. A *hypergraph* is a pair $H = (V(H), E(H))$ consisting of a set of *vertices* $V(H)$ and a set $E(H)$ of subsets of $V(H)$, the *hyperedges* of H. Two vertices are *adjacent* if there exists an edge that contains both of them. Unless otherwise mentioned, our hypergraphs have n vertices and m edges and do not contain isolated vertices.

We use standard notation for graphs. As these are special cases of hypergraph we shall define some concepts only for hypergraphs. Let $G = (V(G), E(G))$ be a *graph*. For a set $S \subseteq V$ we define $S^2 = \{\{u,v\} \mid u, v \in S,\ u \neq v\}$. The *primal graph* of a hypergraph H is the graph $\underline{H}$ on $V(H)$ with $E(\underline{H}) := \{\{u,v\} \mid u, v \in e,\ \text{for some } e \in E(H)\}$.

Tree decompositions and width functions. A *tree decomposition* of a hypergraph H is a pair $(T, \mathcal{B})$, where T is a tree and $\mathcal{B} = \{B_t \mid t \in V(T)\}$ is a family of subsets of $V(H)$, called *bags*, such that (i) every vertex of H appears in some bag of $\mathcal{B}$; (ii) for every $e \in E(H)$ there is a $t \in V(T)$ such that $e \subseteq B_t$; and (iii) for every vertex of H the set of bags containing it forms a subtree T. A *width function* on the vertex set V is a monotone function $f : 2^V \to \mathbb{R}_0^+$, i.e. with $f(X) \leqslant f(Y)$ for $X \subseteq Y$. We define $\mathcal{F}(V)$ to be the set of all width functions on V. The f-width *of a tree decomposition* $\mathcal{T}$ is $\max\{f(B_t) \mid t \in V(T)\}$. Recall also that $\mathcal{T}$ is *small*, if for all $t, t' \in V(T)$ with $t \neq t'$ we have $B_t \nsubseteq B_{t'}$. It is easy to see that there is always a small tree decomposition of minimal f-width.

The *f-hypertree-width of a hypergraph H*, denoted by f-htw(H), is the minimum f-width of all tree decompositions of H. We call such a tree decomposition an *f-optimal tree decomposition*. When considering graphs, we use the analogous notion of *f-tree-width* and denote it by f-tw(G).

Let H be a hypergraph and $X \subseteq V(H)$. An *edge cover* (w.r.t. H) of X is a subset $E' \subseteq E(H)$ such that $X \subseteq \bigcup_{e \in E'} e$. Define $\rho_H(X)$ as the size of the smallest edge cover of X w.r.t. H. $X \subseteq V(H)$ a mapping $\gamma : E(H) \to [0,1]$ is a fractional edge cover of X (w.r.t. H), if $\sum_{v \in e} \gamma(e) \geqslant 1$ for all $v \in X$. Then $\rho_H^*(X)$ is the minimum of $\sum_{e \in E(H)} \gamma(e)$ taken over all fractional edge covers of X w.r.t. H.

Definition 1. Let H be a hypergraph.

- The tree-width of H is $\text{tw}(H) := s$-htw(H) where $s(X) = |X| - 1$.
- The *generalized hypertree-width* of H is $\text{ghw}(H) := \rho_H$-htw(H).
- The *fractional hypertree-width* of H is $\text{fhw}(H) := \rho_H^*$-htw$(H)$.

Separators. For two non-adjacent vertices u, v of a graph G, a set $S \subseteq V(G)$ is a *u, v-separator* if u and v are in different components of $G - S$ and S is *minimal* if it does not properly contain a u, v-separator. By Δ_G we denote the set of all *minimal separators* of G, i.e. all S which are minimal u, v-separators for some u, v. Since a minimal separator can be contained in another one, we single out the set of *inclusion-minimal* separators Δ_G^*, i.e. those not containing another one.

Let $\mathcal{C}_G(S)$ denote the set of connected components of $G - S$. A component $C \in \mathcal{C}_G(S)$ is *full* w.r.t. S, if its neighborhood satisfies $N(C) = S$. By $\mathcal{C}_G^*(S)$ we denote the set of all full connected components of $G - S$. A *block* associated with an $S \in \Delta_G$ is a pair (S, C) for some component $C \in \mathcal{C}_G(S)$. A block is called *full* if C is full w.r.t. S. By definition, the set S of a block (S, C) is required to be a minimal separator. By a slight abuse of terminology, we call a block (S, C) *inclusion minimal* if S is an inclusion minimal separator. The *realization* $R(S, C)$ of a block is the graph obtained from $G[S \cup C]$ by turning S into a clique.

Triangulations, potential maximal cliques. A graph G is *triangulated* or *chordal* if every cycle of length at least 4 in G has a chord. A *triangulation* of G is a chordal graph I on $V(G)$ such that $E(G) \subseteq E(I)$. Furthermore, I is a *minimal triangulation* if there is no chordal graph I' on $V(G)$ with $E(G) \subseteq E(I') \subset E(I)$. A set $\Omega \subseteq V(G)$ is a *potential maximal clique* of G, if there is a minimal triangulation I of G having Ω as a maximal clique. The set of all potential maximal cliques of G is denoted by Π_G. Let Ω be a potential maximal clique with the components $\mathcal{C}(\Omega)$ of $G - \Omega$ and $C \in \mathcal{C}(\Omega)$; then $(N(C), C)$ is called a *block associated with Ω*.

The *f-clique-number of G* is

$$f\text{-}\omega(G) := \max_{\Omega \text{ is a clique of } G} f(\Omega).$$

Let $\mathcal{K}_G$ be the set of maximal cliques of G. A *tree on $\mathcal{K}_G$* is a labeled tree $\mathcal{T} := (T, (\Omega_t)_{t \in V(T)})$ such that there is a bijection between $\mathcal{K}_G$ and T. $\mathcal{T}$ is a *clique-tree of G*, if it additionally satisfies the *clique-intersection property*: For all distinct $\Omega, \Omega' \in \mathcal{K}_G$ the clique $\Omega \cap \Omega'$ is contained in every clique on the unique path connecting Ω and Ω' in $\mathcal{T}$.

Lemma 1. *Let G be a graph and $f \in \mathcal{F}(V(G))$ a width function. Then*

$$f\text{-tw}(G) = \min_{I \text{ is a triangulation of } G} f\text{-}\omega(I). \tag{1}$$

Furthermore, the minimum on the right-hand side is attained by a minimal triangulation of G.

Proof. Let I be any triangulation of G. Since $E(G) \subseteq E(I)$, every tree decomposition of I is also a tree decomposition of G and so, f-tw$(G) \leqslant f$-tw(I). Further, by the well-known fact that a graph I is chordal if and only if it has a clique tree, it is not hard to see that f-tw$(I) = f$-$\omega(I)$ and thus f-tw$(G) \leqslant f$-$\omega(I)$.

For the other direction, let $\mathcal{T} = (T, (B_t)_{t \in V(T)})$ be a small f-optimal tree decomposition of G, i.e. f-width$(\mathcal{T}) = f$-tw(G). We construct a triangulation $I := (V(G), E(I))$

[4] The proofs of some lemmas are straightforward adaptions from the cited references and are omitted. For the sake of completeness, we have made all proofs available online at http://arxiv.org/abs/1106.4719.