Theoretical Computer Science 615 (2016) 1–11

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Fast factorization of Cartesian products of (directed) hypergraphs

Marc Hellmuth [a,b,*], Florian Lehner [c,*]

[a] Department of Mathematics and Computer Science, University of Greifswald, Walther-Rathenau-Straße 47, 17487 Greifswald, Germany
[b] Center for Bioinformatics, Saarland University, Building E 2.1, Room 413, P.O. Box 15 11 50, D-66041 Saarbrücken, Germany
[c] Department of Mathematics, University of Hamburg, Bundesstraße 55, 20146 Hamburg, Germany

ARTICLE INFO

Article history:
Received 28 August 2015
Accepted 17 November 2015
Available online 2 December 2015
Communicated by D.-Z. Du

Keywords:
Directed hypergraph
Cartesian product
Prime factor decomposition
Factorization algorithm
2-section

ABSTRACT

Cartesian products of graphs and hypergraphs have been studied since the 1960s. For (un)directed hypergraphs, unique *prime factor decomposition (PFD)* results with respect to the Cartesian product are known. However, there is still a lack of algorithms, that compute the PFD of directed hypergraphs with respect to the Cartesian product.

In this contribution, we focus on the algorithmic aspects for determining the Cartesian prime factors of a finite, connected, directed hypergraph and present a first polynomial time algorithm to compute its PFD. In particular, the algorithm has time complexity $O(|E||V|r^2)$ for hypergraphs $H = (V, E)$, where the rank r is the maximum number of vertices contained in a hyperedge of H. If r is bounded, then this algorithm performs even in $O(|E|\log^2(|V|))$ time. Thus, our method additionally improves also the time complexity of PFD-algorithms designed for undirected hypergraphs that have time complexity $O(|E||V|r^6\Delta^6)$, where Δ is the maximum number of hyperedges a vertex is contained in.

1. Introduction

Products are a common way in mathematics of constructing larger objects from smaller building blocks. For graphs, hypergraphs, and related set systems several types of products have been investigated, see [18,14] for recent overviews.

In this contribution we will focus on the *Cartesian product* of *directed hypergraphs* that are the common generalization of both directed graphs and (undirected) hypergraphs. In particular, we present a fast and conceptually very simple algorithm to find the decomposition of directed hypergraphs into *prime* hypergraphs (its so-called prime factors), where a (hyper)graph is called prime if it cannot be presented as the product of two nontrivial (hyper)graphs, that is, as the product of two (hyper)graphs with at least two vertices.

Graphs and the Cartesian product A graph is a tuple $G = (V, E)$ with non-empty set of vertices V and a set of edges E containing two-element subsets of V. If the edges are ordered pairs, then G is called directed and undirected, otherwise. The Cartesian *graph* product was introduced by Gert Sabidussi [26]. As noted by Szamkołowicz [29] also Shapiro introduced a notion of Cartesian products of graphs in [27]. Sabidussi and independently V.G. Vizing [30] showed that connected *undirected* graphs have a representation as the Cartesian product of prime graphs that is unique up to the order and isomorphisms of the factors. The question whether one can find the prime factorization of connected undirected graphs in polynomial time

* Corresponding authors.
E-mail addresses: mhellmuth@mailbox.org (M. Hellmuth), mail@florian-lehner.net (F. Lehner).

http://dx.doi.org/10.1016/j.tcs.2015.11.038
0304-3975/

was answered about two decades later by Feigenbaum et al. [13] who presented an $O(|V|^{4.5})$ time algorithm. From then on, a couple of factorization algorithms for undirected graphs have been developed [1,11,13,21,31]. The fastest one is due to Imrich and Peterin [21] and runs in linear-time $O(|V|+|E|)$.

For connected *directed* graphs, Feigenbaum showed that the Cartesian product satisfies the unique prime factorization property [12]. Additionally, she provided a polynomial-time algorithm to determine the prime factors which was improved to a linear time approach by Crespelle et al. [9].

Hypergraphs and the Cartesian product Hypergraphs are a natural generalization of graphs in which "edges" may consist of more than two vertices. More precisely, a hypergraph is a tuple $H=(V,E)$ with non-empty set of vertices V and a set of hyperedges E, where each $e \in E$ is an ordered pair of non-empty sets of vertices $e=(t(e),h(e))$. If $t(e)=h(e)$ for all $e \in E$ the hypergraph is called undirected and directed, otherwise. Products of hypergraphs have been investigated by several authors since the 1960s [2,3,5–8,10,16,19,20,22,23,25,28,32]. It was shown by Imrich [19] that connected *undirected* hypergraphs have a unique prime factor decomposition (PFD) w.r.t. to the Cartesian product, up to isomorphism and the order of the factors. A first polynomial-time factorization algorithm for undirected hypergraphs was proposed by Bretto et al. [8].

Unique prime factorization properties for *directed* hypergraphs were derived by Ostermeier et al. [25]. However, up to our knowledge, no algorithm to determine the Cartesian prime factors of a connected directed hypergraph is established, so-far.

Summary of the results In this contribution, we present an algorithm to compute the PFD of connected directed hypergraphs in $O(|V||E|r^2)$ time, where the rank r denotes the maximum number of vertices contained in the hyperedges. In addition, if we assume to have hypergraphs with bounded rank the algorithm runs in $O(|E|\log^2(|V|))$ time. Note, as directed hypergraphs are a natural generalization of undirected hypergraphs, our method generalizes and significantly improves the time-complexity of the method by Bretto et al. [8]. In fact, the algorithm of Bretto et al. has time complexity $O(|V||E|\Delta^6 r^6)$, where Δ is the maximum number of hyperedges a vertex is contained in. Assuming that given hypergraphs have bounded rank r and bounded maximum degree Δ this algorithm runs therefore in $O(|V||E|)$ time.

We shortly outline our method. Given an arbitrary connected directed hypergraph $H=(V,E)$ we first compute its so-called 2-section $[H]_2$, that is, roughly spoken the underlying *undirected* graph of H. This allows us to use the algorithm of Imrich and Peterin [21] in order to compute the PFD of $[H]_2$ w.r.t. the Cartesian *graph* product. As we will show, this provides enough information to compute the Cartesian prime factors of the directed hypergraph H. In distinction from the method of Bretto et al. our algorithm is in a sense conceptually simpler, as *(1)* we do not need the transformation of the hypergraph H into its so-called L2-section and back, where the L2-section is an edge-labeled version of the 2-section $[H]_2$, and *(2)* the test which (collections) of the factors of the 2-section are prime factors of H follows a complete new idea based on increments of fixed vertex-coordinate positions, that allows an easy and efficient check to determine the PFD of H.

2. Preliminaries

2.1. Basic definitions

A *directed hypergraph* $H=(V,E)$ consists of a finite *vertex* set $V(H):=V$ and a set of *directed hyperedges* or *(hyper)arcs* $E(H):=E$. Each arc $e \in E$ is an ordered pair of non-empty sets of vertices $e=(t(e),h(e))$. The sets $t(e) \subseteq V$ and $h(e) \subseteq V$ are called the *tail* and *head* of e, respectively. The set of vertices, that are contained in an arc will be denoted by $V(e) := t(e) \cup h(e)$. If $t(e)=h(e)$ holds for all $e \in E$, we identify e with $V(e)$, and we call $H=(V,E)$ an *undirected hypergraph*. An undirected hypergraph is an *undirected graph* if $|V(e)|=2$ for all $e \in E$. The elements of E are called simply edges, if we consider an undirected graph. The hypergraph with $|V|=1$ and $E=\emptyset$ is denoted by K_1 and is called *trivial*.

Throughout this contribution, we only consider hypergraphs without multiple hyperedges and thus, being E a usual set, and without loops, that is, $|V(e)|>1$ holds for all $e \in E$. However, we allow to have hyperedges being properly contained in other ones, i.e., we might have arcs $e, f \in E$ with $t(e) \subseteq t(f)$ and $h(e) \subseteq h(f)$.

A *partial hypergraph* or *sub-hypergraph* $H'=(V',E')$ of a hypergraph $H=(V,E)$, denoted by $H' \subseteq H$, is a hypergraph such that $V' \subseteq V$ and $E' \subseteq E$. The partial hypergraph $H'=(V',E')$ is *induced (by V')* if $E'=\{e \in E \mid V(e) \subseteq V'\}$. Induced hypergraphs will be denoted by $\langle V' \rangle$.

A *weak path P (joining the vertices $v_0, v_k \in V$)* in a hypergraph $H=(V,E)$ is a sequence $P=(v_0,e_1,v_1,e_2,\dots,e_k,v_k)$ of distinct vertices and arcs of H, such that $v_0 \in V(e_1)$, $v_k \in V(e_k)$ and $v_j \in V(e_j) \cap V(e_{j+1})$. A hypergraph H is said to be *weakly connected* or simply *connected* for short, if any two vertices of H can be joined by a weak path. A *connected component* of a hypergraph H is a connected sub-hypergraph $H' \subseteq H$ that is maximal w.r.t. inclusion, i.e., there is no other connected sub-hypergraph $H'' \subseteq H$ with $H' \subsetneq H''$. Usually, we identify connected components $H'=(V',E')$ of H simply by their vertex set V', since $\langle V' \rangle = H'$.

A *homomorphism* from $H_1=(V_1,E_1)$ into $H_2=(V_2,E_2)$ is a mapping $\phi: V_1 \to V_2$ such that $\phi(e)$ is an arc in H_2 whenever e is an arc in H_1 with the property that $\phi(t(e))=t(\phi(e))$ and $\phi(h(e))=h(\phi(e))$. A bijective homomorphism ϕ whose inverse function is also a homomorphism is called an *isomorphism*.

The *rank* of a hypergraph $H=(V,E)$ is $r(H)=\max_{e \in E}|V(e)|$.