

The Cartesian product of graphs with loops

Tetiana Boiko *

*Institut für Mathematische Strukturtheorie, Technische Universität Graz
Steyrergasse 30/III, 8010 Graz, Austria*

Johannes Cuno *

*Institut für Mathematische Strukturtheorie, Technische Universität Graz
Steyrergasse 30/III, 8010 Graz, Austria*

Wilfried Imrich *

*Department Mathematik und Informationstechnologie
Montanuniversität Leoben, Franz-Josef-Straße 18, 8700 Leoben, Austria*

Florian Lehner *

*Institut für Geometrie, Technische Universität Graz
Kopernikusgasse 24/IV, 8010 Graz, Austria*

Christiaan E. van de Woestijne †

*Department Mathematik und Informationstechnologie
Montanuniversität Leoben, Franz-Josef-Straße 18, 8700 Leoben, Austria*

Received dd mmmm yyyy, accepted dd mmmmm yyyy, published online dd mmmmm yyyy

Abstract

We extend the definition of the Cartesian product to graphs with loops and show that the Sabidussi–Vizing unique factorization theorem for connected finite simple graphs still holds in this context for all connected finite graphs with at least one unlooped vertex. We also prove that this factorization can be computed in $O(m)$ time, where m is the number of edges of the given graph.

Keywords: Graphs, monoids, factorizations, algorithms.

Math. Subj. Class.: 05C70, 13A05, 20M13, 05C85.

*Tetiana Boiko, Johannes Cuno, Wilfried Imrich, and Florian Lehner are supported by the Austrian Science Fund (FWF): W1230, Doctoral Program “Discrete Mathematics.”

†Christiaan E. van de Woestijne is supported by the Austrian Science Fund (FWF): S9611. This project is part of the Austrian National Research Network “Analytic Combinatorics and Probabilistic Number Theory.”

E-mail addresses: boiko@math.tugraz.at (Tetiana Boiko), cuno@math.tugraz.at (Johannes Cuno), imrich@unileoben.ac.at (Wilfried Imrich), f.lehner@tugraz.at (Florian Lehner), c.vandewoestijne@unileoben.ac.at (Christiaan E. van de Woestijne)

1 Introduction

This paper considers finite undirected graphs that may contain loops, or, put differently, symmetric binary relations on finite sets. One may define several binary operations on such graphs; these are explored in the recently revised monograph [1]. The well-known *Cartesian product* of finite undirected graphs is usually defined only for *simple* graphs, that is, for graphs that do not contain multiple edges between the same pair of vertices and, more importantly for us, do not contain loops. Here we extend this definition.

Before doing so, let us fix the notation. For us, a *graph* $G = (V, E)$ will always be a finite undirected graph without multiple edges. The edge set E is taken to be a set of ordered pairs of vertices; thus, a loop on the vertex $v \in V$ corresponds to the edge $(v, v) \in E$, and as all graphs are undirected, we have $(v, w) \in E$ if and only if $(w, v) \in E$. We will occasionally call a loop a *1-edge* and an edge that is not a loop a *2-edge*. Moreover, given a graph G , we will refer to its vertex set as $V(G)$ and to its edge set as $E(G)$.

Definition 1.1. Let G_1, \dots, G_k be graphs. The *Cartesian product* $G = G_1 \square \dots \square G_k$ is a graph with vertex set $V(G) = V(G_1) \times \dots \times V(G_k)$, and edge set $E(G)$ defined as follows: two vertices $(v_1, \dots, v_k) \in V(G)$ and $(w_1, \dots, w_k) \in V(G)$ are adjacent if there exists an index i such that $(v_i, w_i) \in E(G_i)$, and $v_j = w_j$ for all $j \neq i$.

Note that this definition extends the classical one for simple graphs. The product graph has a loop on a vertex $(v_1, \dots, v_k) \in V(G)$ if and only if there is a loop on at least one of the constituents $v_i \in V(G_i)$. Thus, the distribution of loops (or 1-edges) on the product graph is independent from the distribution of the 2-edges.

Definition 1.2. Let G_1, \dots, G_k be graphs, and $G = G_1 \square \dots \square G_k$. The *i th projection* $p_i : V(G) \rightarrow V(G_i)$ is given by $(v_1, \dots, v_k) \mapsto v_i$.

Using Definition 1.1, we preserve the property that the projections $p_i : V(G) \rightarrow V(G_i)$ are weak homomorphisms from G to G_i . Recall that a *weak homomorphism* between graphs G and H is a map $\varphi : V(G) \rightarrow V(H)$ such that, whenever $(v, w) \in E(G)$, either $(\varphi(v), \varphi(w)) \in E(H)$ or $\varphi(v) = \varphi(w)$. In particular, the presence of loops in G or H does not impose any restriction on a weak homomorphism from G to H .

Definition 1.3. Let G_1, \dots, G_k be graphs, and $G = G_1 \square \dots \square G_k$. For every vertex $a = (a_1, \dots, a_k) \in V(G)$, the *G_i -layer through a* is the induced subgraph

$$\begin{aligned} G_i^a &= \langle \{x \in V(G) \mid p_j(x) = a_j \text{ for } j \neq i\} \rangle \\ &= \langle \{(a_1, a_2, \dots, x_i, \dots, a_k) \mid x_i \in V(G_i)\} \rangle. \end{aligned}$$

Note that $G_i^a = G_i^b$ if and only if $p_j(a) = p_j(b)$ for each index $j \neq i$. With the usual Cartesian product, the restrictions $p_i|V(G_i^a) : V(G_i^a) \rightarrow V(G_i)$ are isomorphisms between G_i^a and G_i [1, Section 4.3]. Under Definition 1.1, we obtain a dichotomy, as follows.

Lemma 1.4. Let G_1, \dots, G_k be graphs, and $G = G_1 \square \dots \square G_k$. Then, the following two conditions hold for every vertex $a = (a_1, \dots, a_k) \in V(G)$ and every $i \in \{1, \dots, k\}$:

- (i) If $a_j \in V(G_j)$ is unlooped for every $j \neq i$, then $p_i|V(G_i^a) : V(G_i^a) \rightarrow V(G_i)$ is an isomorphism between G_i^a and G_i .
- (ii) Otherwise, G_i^a is isomorphic to G_i with a loop attached to every vertex.

Proof. Easy from the definitions. □

2 Matrix and semiring properties

From the definition of the Cartesian product we infer that it is commutative and distributive over the disjoint union. Moreover, the trivial graph K_1 , that is, a vertex without edges, is a unit. As the Cartesian product is also associative, see below, the set Γ_0 of isomorphism classes of finite undirected graphs with loops is a commutative semiring.

To prove associativity we could adapt the proof of [1, Proposition 4.1] for associativity of the Cartesian product of graphs without loops, or we could modify the multiplication table method of [1, Exercise 4.15], which was introduced for the classification of associative products. However, we follow a different path and use the fact that the adjacency matrix $A(G \square H)$ of the Cartesian product of two simple graphs is the Kronecker sum of the adjacency matrices $A(G)$ and $A(H)$ of the factors, see [1, Section 33.3].

Let us first recall that the Kronecker sum $A \oplus B$ of an $n \times n$ matrix A by an $m \times m$ matrix B is defined as $I_n \otimes B + A \otimes I_m$. Here, I_n and I_m denote the identity matrices of size n and m , respectively, and $P \otimes Q$ denotes the Kronecker product. In our situation, the first factor $P = (p_{ij})$ is always an $n \times n$ matrix and the Kronecker product is defined by

$$P \otimes Q = \begin{bmatrix} p_{11}Q & \cdots & p_{1n}Q \\ \vdots & \ddots & \vdots \\ p_{n1}Q & \cdots & p_{nn}Q \end{bmatrix}.$$

Notice that both the Kronecker sum and the Kronecker product are associative but not commutative.

For simple graphs G and H we have $A(G \square H) = A(G) \oplus A(H)$. For graphs with loops we find that the diagonal entries take positive integer values that are not restricted to $\{0, 1\}$. If we agree on the convention that a positive diagonal entry in the adjacency matrix means a loop, whereas a 0 means no loop, then the product given in Definition 1.1 still corresponds to the Kronecker sum. It follows that, up to isomorphism of graphs, this product is associative.

We note in passing that the fact that the Kronecker sum is not commutative does not contradict the commutativity of the Cartesian product: $A(G) \oplus A(H)$ and $A(H) \oplus A(G)$ represent adjacency matrices of $G \square H$ for different vertex numberings.

Finally, we briefly call a graph *entirely looped* if every vertex has a loop. For any graph G , we let $\mathcal{N}(G)$ be G with its loops removed.

Lemma 2.1. *Let G, H, H_1, H_2 be graphs. Assume that G is entirely looped. Then $G \square H$ is entirely looped as well. Moreover, if $\mathcal{N}(H_1) \cong \mathcal{N}(H_2)$, then $G \square H_1 \cong G \square H_2$.*

Proof. The first statement follows directly from Definition 1.1. As remarked earlier, the 2-edges of the products $G \square H_i$ do not depend on the loops of either factor. Thus

$$\begin{aligned} \mathcal{N}(G \square H_1) &= \mathcal{N}(G) \square \mathcal{N}(H_1) \\ &\cong \mathcal{N}(G) \square \mathcal{N}(H_2) \\ &= \mathcal{N}(G \square H_2). \end{aligned}$$

Next, we insert the loops on the product; but, as every vertex of G has a loop, it follows that every vertex of either product $G \square H_i$ has a loop as well, and the two products are obviously isomorphic. \square

It follows that the subset Γ_{00} of Γ_0 given by the isomorphism classes of entirely looped graphs constitutes an *ideal* of the semiring Γ_0 . It is obviously closed under the disjoint union and the Cartesian product, and, since the loop K_1^* is a unit for the Cartesian product inside Γ_{00} , it is a semiring itself. The loop-removing map \mathcal{N} constitutes an isomorphism of semirings between Γ_{00} and the set of simple graphs Γ .

3 Unique factorization

One fundamental property of the Cartesian product, proved independently by Sabidussi [5] and Vizing [6] in the 1960s, is the unique factorization of connected simple graphs into irreducibles with respect to this product. We will extend this result to graphs with loops, where we will have to exclude the set of entirely looped graphs (Lemma 2.1 suggests why). Algebraically speaking, we might want to form the *quotient semiring* Γ_0/Γ_{00} , so that also any fully looped components in disconnected graphs are annulled. However, since we will only consider connected graphs in what follows, this is not of great consequence.

Definition 3.1. A nontrivial, connected graph G with at least one unlooped vertex is called *irreducible* with respect to the Cartesian product if, for every factorization $G = H \square L$, either H or L is trivial.

Recall that a graph is called *trivial* if it is a vertex without edges. Consider a nontrivial, connected graph G with at least one unlooped vertex. One can easily check that, if G is not irreducible, it can be expressed as Cartesian product of two factors each of which is, again, a nontrivial, connected graph with at least one unlooped vertex. Iteration of this procedure yields a representation of G as a product of irreducible graphs. It is occasionally called a *prime factorization*.

Another way to prove the existence of a prime factorization is the following: Any factorization of G with a maximum number of nontrivial factors must be a product of irreducible graphs. If G has n vertices, this maximum number is at most $\log_2(n)$.

Our main results are the following.

Theorem 3.2. *Every nontrivial, connected graph with at least one unlooped vertex has a representation as a product of irreducible graphs with respect to the Cartesian product. The representation is unique up to isomorphisms and the order of the factors.*

Theorem 3.3. *The unique prime factorization with respect to the Cartesian product of a nontrivial, connected graph G with at least one unlooped vertex can be computed in $O(m)$ time, where m is the number of edges of G .*

To prove Theorem 3.2, we follow the method of [1, Section 6.1], for Theorem 3.3 we extend the ideas of [4]. First, let us define convex subgraphs and boxes.

Definition 3.4. A subgraph H of a graph G is *convex* in G if every shortest path in G that connects two vertices of H is completely contained in H . A subgraph H of a Cartesian product $G = G_1 \square \cdots \square G_k$ is called a *box* or *subproduct* if there are subgraphs $H_i \subseteq G_i$ such that

$$H = H_1 \square \cdots \square H_k.$$

In order to determine whether a subgraph is convex or not, only the 2-edges need to be concerned. In particular, a subgraph H is convex in G if and only if the subgraph $\mathcal{N}(H)$ is convex in $\mathcal{N}(G)$.

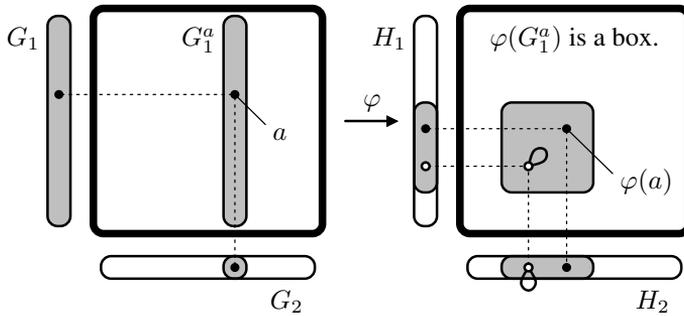


Figure 1: An isomorphism between factored graphs with loops.

Lemma 3.5. *Let H be a subgraph of a Cartesian product $G = G_1 \square \cdots \square G_k$. Then the following are equivalent:*

- (i) H is an induced and convex subgraph of G ;
- (ii) There are induced and convex subgraphs $H_i \subseteq G_i$ such that $H = H_1 \square \cdots \square H_k$.
In other words, H is a box whose factors are induced and convex.

Proof. As far as only the 2-edges are concerned, all convex subgraphs are induced and the assertion is Lemma 6.5 of [1]. This means that $p_1(V(H)) \times \cdots \times p_k(V(H)) = V(H)$. Now, let H_i be the subgraph of G_i induced by $p_i(V(H))$, where $i \in \{1, \dots, k\}$. Then the lemma follows by the definition of the Cartesian product. \square

As remarked after Definition 3.1 every finite graph has a factorization into irreducibles. Thus we only have to show that it is unique in order to prove Theorem 3.2. The next lemma and its corollary makes this precise; the situation is illustrated in Figure 1.

Lemma 3.6. *Let φ be an isomorphism between nontrivial, connected graphs G and H with at least one unlooped vertex. Assume that G and H are representable as products $G = G_1 \square \cdots \square G_k$ and $H = H_1 \square \cdots \square H_\ell$ of irreducible graphs. Then $k = \ell$ and, for every unlooped vertex $a \in V(G)$, there is a permutation π of $\{1, \dots, k\}$ such that*

$$\varphi(G_i^a) = H_{\pi(i)}^{\varphi(a)} \text{ for every } i \in \{1, \dots, k\}.$$

Formally, φ is a bijection between the vertex sets $V(G)$ and $V(H)$. But since φ is a homomorphism of graphs, it induces a well-defined mapping between the edge sets $E(G)$ and $E(H)$. In the above theorem, we slightly abuse notation and denote the image of the subgraph G_i^a , including vertices and edges, by $\varphi(G_i^a)$.

Proof. Fix an unlooped vertex $a = (a_1, \dots, a_k) \in V(G)$, and set $(b_1, \dots, b_\ell) := \varphi(a)$. By Lemma 1.4 we infer that $G_i^a \cong G_i$ and $H_j^{\varphi(a)} \cong H_j$ for every i and j . Every layer G_i^a is induced and, as a consequence of Lemma 3.5, convex in G . So, its image $\varphi(G_i^a)$ is induced and convex in H . Again, as a consequence of Lemma 3.5, $\varphi(G_i^a) = U_1 \square \cdots \square U_\ell$, where every U_j is induced and convex in H_j . But $\varphi(G_i^a) \cong G_i^a \cong G_i$ is irreducible. Since

$(b_1, \dots, b_\ell) = \varphi(a) \in \varphi(G_i^a)$, we conclude that $V(U_j) = \{b_j\}$ for all indices but one, say $\pi(i)$. In other words, $\varphi(G_i^a) \subseteq H_{\pi(i)}^{\varphi(a)}$. But then

$$G_i^a \subseteq \varphi^{-1} \left(H_{\pi(i)}^{\varphi(a)} \right).$$

Because the latter graph is induced and convex, it is a box; and because it is irreducible, it must be contained in G_i^a . Therefore, $\varphi(G_i^a) = H_{\pi(i)}^{\varphi(a)}$.

We claim that the map $\pi : \{1, \dots, k\} \rightarrow \{1, \dots, \ell\}$ is injective. If $\pi(i) = \pi(j)$, then

$$\varphi(G_i^a) = H_{\pi(i)}^{\varphi(a)} = H_{\pi(j)}^{\varphi(a)} = \varphi(G_j^a).$$

But φ is an isomorphism, and therefore the above equation implies $G_i^a = G_j^a$. Since every layer contains at least two vertices, we obtain $i = j$. So, π is injective, and $k \leq \ell$. Repetition of the above argument for φ^{-1} yields $\ell \leq k$. So, $k = \ell$ and π is a permutation. \square

Corollary 3.7. $G_i \cong H_{\pi(i)}$ for every $i \in \{1, \dots, k\}$.

Proof. Since a is unlooped, $G_i \cong G_i^a$ and $H_j \cong H_j^{\varphi(a)}$ for every i and j . By Lemma 3.6 the corollary follows. \square

Clearly Lemma 3.6 and Corollary 3.7 prove the validity of Theorem 3.2.

A remark about automorphisms

In Lemma 3.6 the permutation π of $\{1, \dots, k\}$ is constructed to a fixed unlooped vertex $a \in V(G)$. Actually π is independent of the choice of a , and one can extend Lemma 3.6 to the following description of the automorphisms of G .

Theorem 3.8. *Suppose φ is an automorphism of a nontrivial, connected graph G with at least one unlooped vertex and prime factorization $G = G_1 \square \dots \square G_k$. Then there are a permutation π of $\{1, \dots, k\}$ and isomorphisms $\varphi_i : G_{\pi(i)} \rightarrow G_i$ for which*

$$\varphi(x_1, \dots, x_k) = (\varphi_1(x_{\pi(1)}), \dots, \varphi_k(x_{\pi(k)})).$$

The proof of this theorem can be led on the same lines as that of [1, Theorem 6.10]. Among other consequences this implies that the automorphism group of G is isomorphic to the automorphism group of the disjoint union of the prime factors G_1, \dots, G_k .

4 Algorithms

In this section we present two algorithms for the decomposition of a nontrivial, connected graph G with at least one unlooped vertex into its prime factors. One is straightforward and has complexity $O(mn)$, where m is the number of edges and n the number of vertices of G . The other one is linear in the number of edges of G and depends on the algorithm of Imrich and Peterin [4] for the prime factorization of graphs without loops.

Let $G = G_1 \square \dots \square G_k$ be the prime factorization of a nontrivial, connected graph G with at least one unlooped vertex. Then also $\mathcal{N}(G) = \mathcal{N}(G_1) \square \dots \square \mathcal{N}(G_k)$. Clearly

the graphs $\mathcal{N}(G_i)$, $i \in \{1, \dots, k\}$, need not be irreducible with respect to the Cartesian product. Let $\mathcal{N}(G_i) = H_{i,1} \square \dots \square H_{i,\ell(i)}$ be their prime factorizations. Thus

$$\mathcal{N}(G) = \prod_{i=1}^k \prod_{j=1}^{\ell(i)} H_{i,j}$$

is a representation of $\mathcal{N}(G)$ as a Cartesian product of irreducible graphs. Because the prime factorization is unique, it is the prime factorization of $\mathcal{N}(G)$, up to the order and isomorphisms of the factors. In other words, if $\prod_{j \in J} Z_j$ is a prime factorization of $\mathcal{N}(G)$, then there is a partition $J = J_1 \cup \dots \cup J_k$ such that $\mathcal{N}(G_i) = \prod_{j \in J_i} Z_j$. Our task is to find this partition. We begin with a straightforward approach and prove the following lemma.

Lemma 4.1. *Let G be a nontrivial, connected graph with at least one unlooped vertex. Then its prime factorization can be found in $O(mn)$ time.*

Proof. If G has n vertices, then this is also true for $\mathcal{N}(G)$, and so the number of factors of $\mathcal{N}(G)$, say r , is at most $\log_2(n)$. This also bounds the size of J and implies that the number s of subsets of J is at most $2^{\log_2(n)}$, i. e. $s \leq n$. Notice that the factors of $\mathcal{N}(G)$ can be found in $O(m)$ time by [4].

Let J_1, J_2, \dots, J_s be all subsets of J , ordered in such a way that $|J_i| \leq |J_j|$ whenever $1 \leq i \leq j \leq s$. For every $i \in \{1, \dots, s\}$ set $Y_i := \prod_{j \in J_i} Z_j$ and $Y_i^* := \prod_{j \in J \setminus J_i} Z_j$. Let $\langle Y_i^a \rangle_G$ denote the subgraph of G induced by the layer Y_i^a of Y_i through a , and define $\langle (Y_i^*)^a \rangle_G$ analogously. If the partition $J_i \cup (J \setminus J_i)$ of J leads to a factorization of G , then $\langle Y_i^a \rangle_G$ is isomorphic to a factor of G .

We begin the algorithm by scanning the J_i in the given order. For every J_i and every vertex $v \in V(G)$ we consider the projections $p_{Y_i}(v)$ and $p_{Y_i^*}(v)$ into $\langle Y_i^a \rangle_G$ and $\langle (Y_i^*)^a \rangle_G$. If $v = (v_1, \dots, v_r)$, then $p_{Y_i}(v) = (w_1, \dots, w_r)$, where $w_j = v_j$ if $j \in J_i$, and $w_j = a_j$ otherwise. Notice that $p_{Y_i}(v)$ is the vertex of shortest distance from v in $\langle Y_i^a \rangle_G$. The other projection $p_{Y_i^*}(v)$ is defined analogously. Again, $p_{Y_i^*}(v)$ is the vertex of shortest distance from v in $\langle (Y_i^*)^a \rangle_G$. Clearly $G = \langle Y_i^a \rangle_G \square \langle (Y_i^*)^a \rangle_G$ if and only if for every vertex $v \in V(G)$ the following two conditions are satisfied:

1. If v is unlooped, then both $p_{Y_i}(v)$ and $p_{Y_i^*}(v)$ are unlooped.
2. If v has a loop then at least one of the vertices $p_{Y_i}(v)$, $p_{Y_i^*}(v)$ has a loop.

The time necessary to compute $p_{Y_i}(v)$ and $p_{Y_i^*}(v)$ for a given v is proportional to r . As one can check in constant time whether $p_{Y_i}(v)$ or $p_{Y_i^*}(v)$ has a loop, one can check in $O(nr)$ time whether $G = \langle Y_i^a \rangle_G \square \langle (Y_i^*)^a \rangle_G$.

Notice that r is the number of factors of $\mathcal{N}(G)$, which is also bounded by the minimum degree δ of $\mathcal{N}(G)$. This is easily seen, since every vertex meets every layer and, in a connected graph, is incident with at least one edge of that layer. Hence the number of factors cannot exceed the degree of any vertex, and $nr \leq n\delta \leq m$.

For a given J_i one can thus check in $O(m)$ time whether $\langle Y_i^a \rangle_G$ is a factor of G . If it is, and if J_i is minimal with respect to inclusion, then it clearly is an irreducible factor. Hence, this is true for the first factor that we encounter, because of having ordered the J_i by size. We now continue the scan, omitting the J_j that are not disjoint from J_i , to find the next factor. Clearly it will also be irreducible. We continue until we have found all irreducible factors. Since there are no more than n subsets of J , we can find them in $O(nm)$ time. \square

In order to reduce the complexity to $O(m)$, we need some more preparation. So let a be an unlooped vertex of G and L_i be the levels of a BFS-ordering of the vertices of G with respect to the root a . That is, L_i consists of all vertices of distance i from a . Furthermore, we enumerate the vertices of G by giving them so-called BFS-numbers that satisfy $\text{BFS}(v) > \text{BFS}(u)$ if the distance from a to v is larger than the one from a to u .

It is important to observe that the projection $p_{Y_i}(v)$ is a vertex of $\langle Y_i^a \rangle_G$ and always closer to a than v , unless v already is a vertex of $\langle Y_i^a \rangle_G$, because then $p_{Y_i}(v) = v$.

Proof of Theorem 3.3. Let $\prod_{j \in J} Z_j$ be a prime factorization of $\mathcal{N}(G)$. We begin with the trivial partition of J and wish to check, whether it already leads to a factorization of G . We scan the vertices v of G in BFS-order and, given v , check the validity of Conditions

- (i) If v is unlooped, then all $p_{Y_i}(v)$ are unlooped.
- (ii) If v has a loop, then at least one of the projections $p_{Y_i}(v)$ has a loop.

If one of these conditions is not satisfied, then the partition of J is obviously inconsistent with the loop structure. In either case we have too many factors and have to make the partition of J coarser. Before we go on, notice that in L_1 these conditions are trivially satisfied for any partition of J , because all projections $p_{Y_i}(v)$ are a , except one, which is v .

Suppose we arrive at a vertex v where one of the conditions (i) or (ii) is violated for the first time. Assume first that Condition (i) is violated, that is, v is unlooped, but $p_{Y_i}(v)$ has a loop for an index i . In the end, all projections have to be unlooped. We must combine the set J_i with one or more other sets of the partition. Using the fact that we proceed in BFS-order, it is easy to see that we have to make v a unit layer vertex, that is, we combine all those sets J_j for which $p_{Y_j}(v) \neq a$. Assume now that Condition (ii) is violated, that is, v has a loop, but no $p_{Y_i}(v)$ does. In the end, at least one of the projections has to have a loop. As above, the only way to achieve this is to make v a unit layer vertex, that is, we combine all factors J_j for which $p_{Y_j}(v) \neq a$.

In both cases we arrive at a coarser partition of J than the one we started out with. By associativity of the Cartesian product with loops, we need not recheck the vertices we have already considered and continue in BFS-order.

Notice that this process yields a factorization, because both (i) and (ii) are satisfied. For every finer partition of J one of these conditions is violated, hence the factorization is the unique prime factorization we are looking for.

Considering the computational cost of these operations, we observe that all projections that we need for the n vertices can be computed, in $O(n|J|)$ time. Since we can check in constant time whether a vertex has a loop or not, the checks for conditions (i) and (ii) can also be done in $O(n|J|)$ time. As $|J| \leq \delta$, we have $O(n|J|) = O(n\delta) = O(m)$.

Finally, recomputing the partition needs at most $O(|J|)$ time, and this has to be done only at most $|J|$ times, so the cost is $O(\delta^2)$. \square

5 Remarks

In [2] it was shown that connected set systems, or hypergraphs, as they are called now, also have unique prime factorizations with respect to the Cartesian product if one-element sets, or loops in our terminology, are excluded. Our result also extends to hypergraphs with loops: Connected hypergraphs have unique prime factorization with respect to the Cartesian product, if there is a least one vertex without a loop. Furthermore, the same

arguments yield unique prime factorization for connected infinite graphs or hypergraphs with respect to the weak Cartesian product; compare [3].

References

- [1] R. Hammack, W. Imrich, and S. Klavžar, *Handbook of product graphs, Second Edition*, Discrete Mathematics and its Applications (Boca Raton), CRC Press, Boca Raton, FL, 2011.
- [2] W. Imrich, Kartesisches Produkt von Mengensystemen und Graphen, *Studia Sci. Math. Hungar.* 81 (1967), 285–290.
- [3] W. Imrich, Über das schwache Kartesische Produkt von Graphen, *J. Combinatorial Theory Ser. B*, 11 (1971), 1–16.
- [4] W. Imrich and I. Peterin, Recognizing Cartesian products in linear time, *Discrete Math.* 307 (2007), 472–483.
- [5] G. Sabidussi, Graph multiplication, *Math. Z.*, 72 (1960), 446–457.
- [6] V. Vizing, The Cartesian product of graphs (Russian), *Vychisl. Sistemy*, 9 (1963), 30–43. English translation in *Comp. El. Syst.* 2 (1966), 352–365.