European Journal of Combinatorics 30 (2009) 1119–1133

Contents lists available at ScienceDirect

European Journal of Combinatorics

journal homepage: www.elsevier.com/locate/ejc

Approximate graph products

Marc Hellmuth [a], Wilfried Imrich [b], Werner Klöckl [b], Peter F. Stadler [a,c,d,e]

[a] *Bioinformatics Group, Department of Computer Science and Interdisciplinary Center for Bioinformatics, University of Leipzig, Härtelstraße 16-18, D-04107 Leipzig, Germany*

[b] *Chair of Applied Mathematics, Montanuniversität Leoben, A-8700 Leoben, Austria*

[c] *RNomics Group, Fraunhofer Institut für Zelltherapie und Immunologie, Perlickstraße 1, D-04103 Leipzig, Germany*

[d] *Department of Theoretical Chemistry, University of Vienna, Währingerstraße 17, A-1090 Wien, Austria*

[e] *Santa Fe Institute, 1399 Hyde Park Road, Santa Fe, NM 87501, USA*

ARTICLE INFO

Article history:
Available online 30 September 2008

ABSTRACT

The problem of recognizing *approximate* graph products arises in theoretical biology. This paper presents an algorithm that recognizes a large class of approximate graph products.

The main part of this contribution is concerned with a new, local prime factorization algorithm that factorizes all strong products on an extensive class of graphs that contains, in particular, all products of triangle-free graphs on at least three vertices. The local approach is linear for graph with fixed maximal degree.

1. Introduction and motivation

The problem of computing approximate graph products was posed several years ago in a theoretical biology context [16]. The notion of a "character" (trait or *Merkmal*) can be understood as a property of an organism that in evolution can vary independently of other traits. Characters thus are not necessarily the same as observable properties such as arms, legs, fingers, spinal chord, etc, although such observables of course often are instantiations of characters. The important biological distinction is *whether* such measurable attributes (or combinations thereof) form a "coordinate axis" along which the character states (e.g. the lengths of arms or fingers) can vary independently of other traits, or whether the underlying genetics dictates dependencies among the observables [10].

This question can be represented as a graph problem in the following way: Consider a set $\mathbb{X}$ of "phenotypes", that is, representations of distinct organisms, each of which is characterized by a list of properties such as body weight, color, presence or absence of certain bones, etc. If we know about

E-mail addresses: marc@bioinf.uni-leipzig.de (M. Hellmuth), imrich@unileoben.ac.at (W. Imrich), werner.kloeckl@mu-leoben.at (W. Klöckl), studla@bioinf.uni-leipzig.de (P.F. Stadler).

0195-6698/$ – see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ejc.2008.09.006

the phylogenetic relationships between the members of $\mathbb{X}$, we can estimate which combinations of properties are interconvertible over short evolutionary time-scales. This evolutionary "accessibility relation" introduces a graph structure on $\mathbb{X}$ [6,7,1,14]. In practice, however, one will only use a subset of properties to distinguish the vertices in $\mathbb{X}$, so that sufficient information can be sampled from the observed phylogenetic transitions between many pairs of related species. Clearly, the sampled information will always be incomplete and have limited accuracy. Hence, for practical application we will have to work with a graph representation of phenotype space that will be known only approximately, and that will cover only a limited region of interest. This forces us to consider an approximate and local theory.

Traits are said to vary "quasi-independently" of each other if we can find a coordinate system that labels the states of the traits such that transitions (edges) between different states are not influenced by the other coordinates. In other words, characters are identified with the fibers of a product structure of the phenotype graph $\mathbb{X}$. In its original formulation [16,13], the approach is developed in the context of topological spaces and their products. In the finite case, this setting specializes to the reflexive direct product (of directed graphs with loops at each vertex); under the additional assumption that the accessibility relation is symmetric, this is equivalent to the strong product of undirected graphs, which we treat in this contribution.

Given a graph G that approximately represents a phenotype space we are thus given the task of finding a graph H that is a non-trivial strong product and a good approximation of G in the sense that H can be reached from G by a small number of additions or deletions of edges and vertices.

It stands to reason that this optimization problem will in general be hard, probably NP-complete, because the number of perturbed graphs with no more than k insertions and/or deletions will grow exponentially in k. Hence, even with a linear factorization algorithm (Lemma 2), an exhaustive search over all possible perturbed graphs (in all but the smallest distances) is certainly impractical for the graph sizes and perturbation levels that we have to expect in practical applications.

Therefore, our aim is a fast, preferably linear, algorithm that recognizes a large class of approximate graph products. Clearly the products in this class, as special approximate graph products, must be factorizable within this time complexity too.

The observation that neighborhoods in strong products are subproducts of the original product leads to the idea of factorizing neighborhoods and using the local factorizations for the construction of a global one, thereby lowering the complexity of decomposing a graph with respect to the strong product. If the graph G is not too disturbed, we would expect to be able to cover most of it by neighborhoods that are products and to use this for the construction of a strong product H that approximates G.

Our contribution is organized as follows: We first remind the reader of a few basic properties of graphs products and formally introduce approximate products (Section 2), and then describe a local approach to recognizing strong products. In Section 4 we derive desirable properties of thin graphs and then characterize the class of NICE graphs. Section 6 contains the main result: NICE graphs can be factorized in linear time by our local approach. In the final section we extend this results by showing that the same algorithm also recognizes interesting classes of prime graphs that can be approximated by products.

2. Products and approximate graph products

In this section we begin with the definition of the strong and the Cartesian product, prime factor decompositions, and fundamental properties of product graphs. The presentation is rather concise; for details and proofs we refer the reader to the book of Imrich and Klavžar [8]. Furthermore, we define approximate graph products and illustrate the complexity of their recognition.

We only consider finite, simple undirected graphs. To fix ideas, a graph G is a finite set $V(G)$ of vertices together with a set $E(G)$ of unordered pairs of distinct elements of V, the edges of G. We call $|V(G)|$ the order of G and $|E(G)|$ its size.

Our main product of interest is the strong product. The vertex set of the *strong product* $G_1 \boxtimes G_2$ of two graphs G_1 and G_2 is the set

$$\{(v_1, v_2) \mid v_1 \in V(G_1), v_2 \in V(G_2)\},$$

that is, the Cartesian product of the vertex sets of the factors. Two vertices $(x_1, x_2), (y_1, y_2)$ are adjacent in $G_1 \boxtimes G_2$ if one of the following conditions is satisfied:

(i) $[x_1, y_1] \in E(G_1)$ and $[x_2, y_2] \in E(G_2)$,
(ii) $[x_1, y_1] \in E(G_1)$ and $x_2 = y_2$,
(iii) $[x_2, y_2] \in E(G_2)$ and $x_1 = y_1$.

Closely related is the *Cartesian product* $G_1 \square G_2$, which plays a central role in the prime factorization algorithms of strong products. It has the same vertex set as $G_1 \boxtimes G_2$, but vertices are only adjacent if they satisfy (ii) or (iii). Consequently, the edges of a strong product that satisfy (ii) or (iii) are called *Cartesian*, the others *non-Cartesian*. Clearly the Cartesian edges are those, whose endpoints differ in exactly one coordinate. This characterization also holds for products of more than two factors.

It is well known that the strong product is associative. Thus a vertex of

$$\bigboxtimes_{i=1}^{n} G_i = G_1 \boxtimes G_2 \boxtimes \cdots \boxtimes G_n$$

is properly "coordinatized" by the vector $(x_1, \ldots, x_n)$ whose entries are the vertices x_i of the factor graphs G_i.

The coordinatization of a product is equivalent to a partial edge coloring of G by n colors $c_1, c_2, \ldots, c_n$ that assigns the color c_k to all edges xy whose endpoints differ only in the kth coordinate, that is, to all edges xy with $x_i = y_i$, $i \neq k$ and $x_k \neq y_k$. In this case we set $c(xy) = c_k$.

This coloring is called *product coloring*. It colors the *Cartesian edges* of G (with respect to the *given* product representation). For every color c the set $E_c = \{e \in E | c(e) = c\}$ of edges with color c spans G. The connected components of the induced subgraph $\langle E_c \rangle$ of E_c are usually called the *fibers* or *layers* of G. These fibers are isomorphic to the factor G_k of G with $c_k = c$. For a fiber that is isomorphic to G_k and contains the vertex v we will use the notation G_k^v.

These types of colorings will play an important role in the prime factorizations of graphs.

A graph is *non-trivial* if it has at least two vertices, and it is *prime* with respect to the strong product if it is not isomorphic to the strong product of two non-trivial graphs. It is clear that every finite graph is isomorphic to a strong product of prime graphs. It has been shown by Dörfler and Imrich [2], and McKenzie [11], that the prime factorization decomposition, PFD for short, of a finite graph with respect to the strong product is unique up to isomorphisms and the order of the factors. For us it is important that the prime factorization can be found in polynomial time. This result is due to Feigenbaum and Schäffer [5].

Analogously, one defines prime graphs and the PFD with respect to the Cartesian product. Sabidussi [12], and Vizing [15] showed that it is unique for connected graphs. It can also be found in polynomial time [4]. For a linear algorithm see [9].

The PFDs for disconnected graphs need not be unique for either product.

Note that a NICE graph of order n cannot have more than $\log_2 n$ factors, because the order of a product of k non-trivial graphs is at least 3^k.

For the definition of approximate graph products we begin with the definition of the distance between two graphs. We say the *distance* $d(G, H)$ between two graphs G and H is the smallest integer k such that G and H have representations G', H' for which the sum of the symmetric differences between the vertex sets of the two graphs and between their edge sets is at most k, that is, if

$$|V(G') \triangle V(H')| + |E(G') \triangle E(H')| \leq k.$$

A graph G is a *k-approximate graph product* if there is a product H such that

$$d(G, H) \leq k.$$

Here k need not be constant, it can be a slowly growing function of $|E(G)|$. This definition is motivated by the way we imagine errors to occur in the measurements that are needed for construction of the phenotype space $\mathbb{X}$ and the corresponding phenotype graph.

The remainder of the section illustrates the complexity of recognizing approximate graph products.

We first show that k-approximate graph products can be recognized in polynomial time for constant k. To this end we begin with a bound on the number of graphs of distance k from a given connected graph G.

Lemma 1. *Let G be a connected graph on n vertices. Then the number of connected graphs of fixed distance $\leq k$ from G is $O(n^{2k})$.*

Proof. We bound the number of (connected and disconnected) graphs H of distance $\leq k$ from G. First let $V(H) = V(G)$ and $E(H) = E(G)$. Then we modify the edge set $E(H)$. There are $\binom{n}{2} = (n)(n-1)/2 = O(n^2)$ ways to select a pair of vertices in $V(G)$. If a selected pair is an edge of $E(G)$ we delete it from $E(H)$, otherwise we add the corresponding edge. Iterating this step i-times we generate $O(n^{2i})$ graphs. Summing over all i from 0 to k yields $O(n^{2k})$ graphs, and in particular all graphs with distance of at most k from G that have the same vertex set as G.

Now we allow the vertex set to change. Suppose we only add $j \leq k$ isolated vertices. We proceed with $V(H) = V(G) \cup \{v_1, \ldots, v_j\}$ and $E(H) = E(G)$. Now we have $(n+j)(n+j-1)/2 = O(n^2)$ ways to select pairs in $V(H)$. Hence we can re-use the argument above to see that this generates no more than $O(n^{2k})$ distinct graphs.

Finally, suppose we add l_1 and delete l_2 vertices. Of course, we have $l_1 + l_2 \leq k$. For fixed l_1, we know from the previous paragraph that there are no more than $O(n^{2l_1})$ distinct graphs. In each of them, we have at most $\binom{n}{l_2} \in O(n^{l_2})$ ways to delete vertices that were already there in $V(G)$. (Note that deleting a newly inserted vertex is equivalent to reducing l_1 and hence need not be considered.) For fixed l_1 and l_2, we can proceed by adding or deleting edges. Now we have $\binom{n+l_1-l_2}{2} \in O(n^2)$ ways to select, and we can repeat this no more than $i \leq k - l_1 - l_2$ times, giving us access to no more than $O(n^{2k})$ graphs. There are $O(k^2)$ ways of choosing l_1 and l_2, hence we have no more than $O(k^2 \cdot n^{2k})$.

The lemma follows by treating k as a prescribed constant. □

Lemma 2. *For fixed k all strong and Cartesian k-approximate graph products can be recognized in polynomial time in n.*

Proof. For a given graph G the number of graphs of distance at most k is $O(n^{2k})$. The observation that every one of these graphs can be factored in polynomial time completes the proof. □

Without the restriction on k the problem of finding a product of closest distance to a given graph G is NP-complete for the Cartesian product. This has been shown by Feigenbaum and Haddad [3]. They proved that the following problem is NP-complete:

Problem 1. To a given connected prime graph G find a connected product $G_1 \Box \cdots \Box G_k$ with the same number of vertices as G, such that G can be obtained from $G_1 \Box \cdots \Box G_k$ by adding a minimum number of edges only or deleting a minimum number of edges only.

We conjecture that this also holds for the strong product.

3. A local approach to the recognition of approximate graph products

The basis for the approach in this paper is the fact that neighborhoods in strong products are products of neighborhoods in the factors. We therefore begin with the definition of neighborhoods and related concepts.

The neighborhood $N(v)$ of a vertex $v \in V$ is the set of all vertices that are adjacent to v. The closed neighborhood $N[v]$ of v is then the union $N(v) \cup \{v\}$. Fig. 1 illustrates that induced closed neighborhoods are products of induced closed neighborhoods of each factor, the so-called *subproducts*. For a proof we start from the following lemma; see [8], p.149:

Lemma 3. *Given two graphs G and H; then the distance between (g_1, h_1) and $(g_2, h_2))$ in $G \boxtimes H$ is* $\max\{d^G(g_1, g_2), d^H(h_1, h_2)\}$.

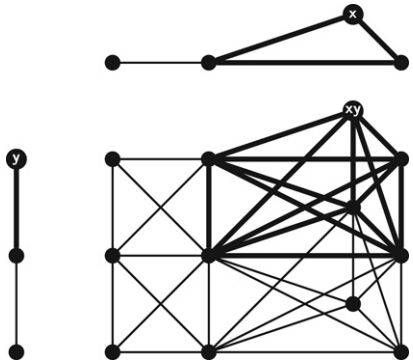

Fig. 1. $\langle N[(x, y)]\rangle = \langle N[x]\rangle \boxtimes \langle N[y]\rangle$.

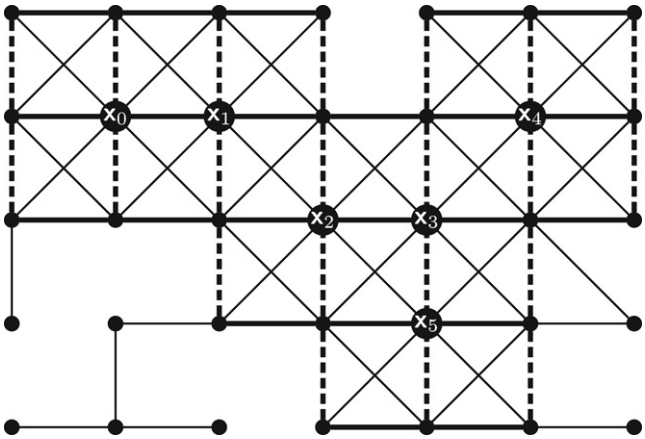

Fig. 2. A graph obtained by deleting edges from $H = P_4 \boxtimes P_6$.

Corollary 1. *Given two graphs G and H; then* $\langle N^{G\boxtimes H}[(x, y)]\rangle = \langle N^G[x]\rangle \boxtimes \langle N^H[y]\rangle$.

Induction on the number of factors shows that induced closed neighborhoods are subproducts.

Now, consider the graph G of Fig. 2. It approximates $P_4 \boxtimes P_6$. Suppose we are unaware of this fact. If G is a product, then the closed neighborhood of every vertex is also a product. We factor every neighborhood and try to use the information to find a product that is either identical to G or approximates it.

To this end we choose the vertices $x_0, x_1, \ldots, x_5$, factor their neighborhoods and consider the Cartesian edges in the factorizations. There are two factors for every such neighborhood and thus two colors for the Cartesian edges in every neighborhood. If two neighborhoods have a Cartesian edge in common, we identify their colors. In this way we end up with two colors altogether, one for the horizontal Cartesian edges and one for the vertical ones. If G is a product, then the edges of the same color span a subgraph with isomorphic components, and all of them are isomorphic to one and the same factor.

Clearly the components are not isomorphic in our example. But, under the assumption that G is an approximate graph product, we take a component of maximal size for each color, say the one consisting of the horizontal edges through x_2, and the vertical ones through x_3. They induce a P_6 and a P_4. It is now easily seen that G can be obtained from $P_4 \boxtimes P_6$ by the deletion of edges.

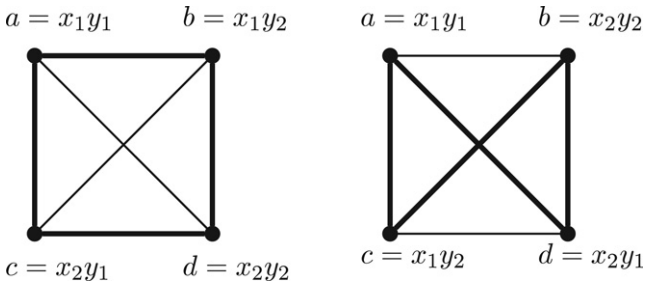

Fig. 3. The edge (a, b) is Cartesian in the left and non-Cartesian in the right coordinatization.

The main questions that remain pertain to the choice of the vertices $x_0, x_1, \ldots, x_5$. They were chosen, because the neighborhoods of the other vertices were either prime, or they were not prime and their Cartesian edges were not uniquely defined.

It is clear why such neighborhoods do not help us in our approach, but we wish to explain when the Cartesian edges in a product are not uniquely defined. Consider Fig. 3. It shows two copies of $K_2 \boxtimes K_2$ that are coordinatized differently, which means that the diagonals are non-Cartesian in the left graph, but Cartesian in the right one.

Thus, we need a closer look at non-unique coordinatizations in products. Let us say a product is *uniquely coordinatizable* if for any PFD and its coordinatization the sets of vertices that coincide in all but in one coordinate are the same. In other words, a product is uniquely coordinatizable if the property of being a Cartesian edge only depends on G, but not on the choice of the PFD.

What we clearly need for our approach is a covering of a significant part of the graph by factorable neighborhoods with unique coordinatizations. Moreover, the neighborhoods have to overlap enough, such that after merging colors there remains only a reasonable number of colors, that is, factors.

4. Thin graphs

Fig. 3 shows that the reason for the non-unique coordinatizations is the existence of automorphisms that interchange the vertices a and b, but fix all the others. This is possible because a and b have the same closed neighborhoods.

Let us say two adjacent vertices x, y of a graph are in the relation S if they have the same closed neighborhoods. A graph is *S-thin*, or *thin*, for short, if no two vertices are in the relation S. This property guarantees that the property of being a Cartesian edge does not depend on the choice of the coordinates.

Theorem 1 ([2,11]). *Every finite, undirected, simple connected graph G has a unique PFD with respect to the strong product. If G is S-thin, then the coordinatization is unique.*

It is easy to see that a product is thin if and only if all factors are thin. Since triangle-free graphs different from K_2 are obviously thin it is natural to begin the investigation of approximate graph products with products of triangle-free graphs.

We remark that connected, triangle-free graphs must be prime, because every strong product of two non-trivial connected graphs contains triangles. Since all subgraphs of a triangle-free graph are triangle-free they are also prime.

Lemma 4. *Let $G = \boxtimes_{i=1}^{n} G_i$ be the strong product of n triangle-free graphs different from K_2. Then G is thin. Furthermore, for all $v \in V(G)$ the PFD of $\langle N[v] \rangle$ has exactly n prime factors.*

Proof. From the assumptions we know that for any two adjacent vertices v, w there is a vertex u that is adjacent to exactly one of the vertices v, w. Hence no two vertices have the same closed neighborhood and therefore G is thin. By the above remark and Corollary 1, we conclude that the PFD of any closed neighborhood in G consists of n factors, each of which is a subgraph of a G_j. □

The last result is remarkable because it will allow us to recover the prime factorization of a product of triangle-free graphs with the method outlined in Section 3. In fact, let us heuristically follow the method before we make it precise in the following sections. Let G be a product of n triangle-free graphs. We first note that the factors are prime; our task is to recover them from G. By Corollary 1, each neighborhood $\langle N[x]\rangle$, $x \in V(G)$, is a product of exactly n prime factors. Thus the Cartesian edges are uniquely determined. Clearly every Cartesian edge of G is Cartesian in every neighborhood to which it belongs and every Cartesian edge of a neighborhood is Cartesian in G.

Finally, it is clear that to any two Cartesian edges e, f in G that have the same color, say c, there is a chain $e = e_0, e_1, \ldots, e_k = f$ such that there are neighborhoods N_i in G to every i, $0 \leq i < k$, such that $e_i, e_{i+1} \in N_i$. This ensures that by merging colors as described in Section 3 we recover the original coloring of the Cartesian edges of G, albeit with possibly different names for the colors.

5. NICE graphs

In this section we formalize the preceding arguments and present the prerequisites for a procedure that recognizes approximate graph products for a large class of graphs that includes all strong products of connected, triangle-free graphs and factors them, if they are products. We begin with several formal definitions that will be used in our lemmas.

Definition 1. A *partial coloring* of a graph G is a mapping F_G from a subset E' of $E(G)$ into a set C of colors.

To make the concept of "merging colors" that we previously used more precise we introduce the terms "color continuation" and "combined coloring" in the following definition.

Definition 2. Let $H_1, H_2 \subset G$ and F_{H_1}, resp. F_{H_2}, be partial colorings of H_1, resp. H_2. Then F_{H_2} is a *color continuation* of F_{H_1} if for every color c in the image of F_{H_2} there is an edge in H_2 with color c that is also in the domain of F_{H_1}.

The *combined coloring* on $H_1 \cup H_2$ uses the colors of F_{H_1} on H_1 and those of F_{H_2} on $H_2 \setminus H_1$.

In other words, for all newly colored edges with color c in H_2 we have to find a representative that was already colored in H_1. Notice that F_{H_1} is not necessarily a color continuation of F_{H_2}, even if F_{H_2} is a color continuation F_{H_1}. The next definition extends that of a product coloring in Section 2.

Definition 3. A *partial product coloring* of a graph $G = \boxtimes_{i=1}^{n} G_i$ is a mapping P_G from a subset E' of the set of Cartesian edges of G into a set $C = \{1, \ldots, n\}$ of colors, such that all edges in G_i-fibers receive the same color i.

Definition 4. A graph G is *thin-N coverable* if there is a finite sequence $\sigma = (v_i)_{i=1}^{k}$ of vertices v_i with thin induced neighborhoods $\langle N[v_i]\rangle$, such that for all $v \in V(G)$ there exists a vertex $v_j \in \sigma$ with $v \in N[v_j]$.

Definition 5. We call σ a *covering sequence* if in addition for all $i > 0$ the product coloring of $\langle N[v_{i+1}]\rangle$ is a color continuation of the combined coloring of $\bigcup_{j=1}^{i} E(\langle N[v_j]\rangle)$ defined by the product colorings of each $\langle N[v_j]\rangle$.

G is *thin-N intersection coverable*, in short *NICE*, if it has a covering sequence.

Remark 1. Thin-N intersection coverable does not imply that all edges of G are covered by thin induced neighborhoods. A counterexample is shown in Fig. 4.

We will show that the product of thin-N coverable graphs, resp. of intersection coverable graphs, is a thin-N coverable, resp. NICE graph. A useful tool for our proofs is the following well-known lemma.

Lemma 5 ([8]). *A graph is thin if and only if all of its factors with respect to the strong product are thin.*

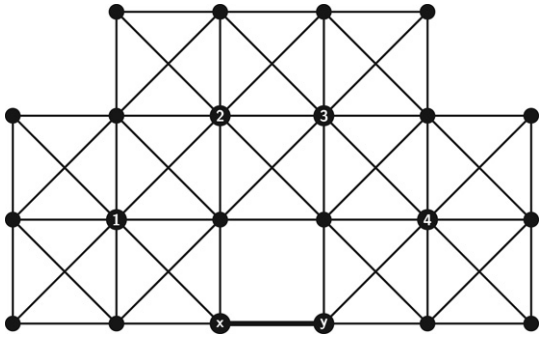

Fig. 4. A NICE graph with covering sequence $\sigma = \{1, 2, 3, 4\}$. The thick edge (x, y) cannot be covered by thin neighborhoods, since neither $\langle N[x]\rangle$ nor $\langle N[y]\rangle$ is thin.

Lemma 6. *Let $G = \boxtimes_{i=1}^{n} G_i$ be a strong product. G is thin-N coverable if and only if all of its factors are thin-N coverable.*

Proof. By induction we can assume that the number of factors is 2.

Suppose every factor is thin-N coverable. Then there are sets of vertices $S_i \subset V(G_i)(i \in \{1, 2\})$ such that every vertex in S_i has a thin neighborhood and $\bigcup_{v \in S_i} N[v] = V(G_i)$.

Then the neighborhoods of $S_1 \times S_2$ cover G. To see this we choose $v = (v_1, v_2) \in V(G)$ arbitrarily. By the choice of S_i there are thin neighborhoods $N[v_i']$ that contain v_i and from Lemma 5 we know that $N[(v_1', v_2')]$ is a thin neighborhood containing v.

For the converse let $v_i \in V(G_i)$ be arbitrarily chosen. Let $v \in V(G)$ with the ith coordinate v_i. By assumption it is in the thin closed neighborhood of some vertex v'. Thus v_i is contained in $N[v_i']$, the neighborhood of the ith coordinate of v' in G_i. By Lemma 5 $N[v_i']$ is thin. □

Lemma 7. *Let $G = \boxtimes_{i=1}^{n} G_i$ be a strong product graph, for which all factors are NICE. Then G is NICE.*

Proof. Since the strong product is commutative and associative it suffices to show this for the product $G' = G_1 \boxtimes G_2$ of two NICE (not necessarily prime) graphs.

Let $\sigma_1 = (x_i)_{i=1}^{k}$ and $\sigma_2 = (y_i)_{i=1}^{m}$ be covering sequences of G_1, resp. G_2. We will now construct a covering sequence for G'.

First take the vertex $v_1 \in V(G')$ with coordinates (x_1, y_1). Lemma 5 implies that $\langle N[v_1]\rangle$ is thin. Now we choose the vertex v_2 with the coordinates (x_2, y_1). Notice that a product coloring of $\langle N[v]\rangle$, which is a partial product coloring of G, colors edges that are either in a G_1- or a G_2-fiber of G'. We now recall that G_k^x denotes the fiber with respect to the kth factor that passes through vertex x in G and note that $\langle N[v_1]\rangle \cap G_2^{v_1}$ and $\langle N[v_2]\rangle \cap G_2^{v_2}$ are both isomorphic to $\langle N^{G_2}[y_1]\rangle$, and hence $\langle N[v_1]\rangle \cap G_2^{v_1} \cong \langle N[v_2]\rangle \cap G_2^{v_2}$. Thus, we will find for each colored edge $((x_2, y_1), (x_2, y'))$ in $\langle N[v_2]\rangle$ a corresponding edge $((x_1, y_1), (x_1, y'))$ that was already colored in $\langle N[v_1]\rangle$. In consequence of the definition of the strong product there is a non-Cartesian edge $((x_2, y_1), (x_1, y'))$ and thus $((x_1, y_1), (x_1, y'))$ is also an edge in $\langle N[v_2]\rangle$.

Furthermore, since G_1 is NICE, it follows that for every color c on some edge e in $\langle N[v_2]\rangle \cap G_1^{v_2}$ there is an edge e' in $\langle N[v_2]\rangle$ with color c that was also colored in $\langle N[v_1]\rangle$.

By repeating this procedure we will cover the whole G_1-fibers. On the other hand we can repeat the same procedure with all G_2-fibers by choosing the ordered vertices with coordinates $(x_l, y_1), \ldots, (x_l, y_m)$ where l is fixed.

Hence the sequence $(v_1, \ldots, v_{k \cdot m})$ of vertices in the product graph with coordinates (x_i, y_j) for $v_{(j-1)\cdot k+i}$ is a covering sequence of G'. □

Now we prove a lemma that provides a large class of thin-N coverable and NICE graphs.

Lemma 8. *Every triangle-free non-trivial connected graph G different from K_2 is NICE.*

Proof. The same argument as in the proof of Lemma 4 implies that for each vertex v with $\deg(v) > 1$ the induced closed neighborhood $\langle N[v]\rangle$ is thin.

For every edge there is at least one end vertex v_i with degree greater than 1. Hence, there are thin neighborhoods $\langle N[v_i]\rangle$ such that $\bigcup N[v_i] = V(G)$. Furthermore the ordering of the v_i can be chosen such that

$$v_{i+1} \in X_i := \left(\bigcup_{j=1}^{i} N[v_j]\right) \quad \text{and} \quad N[v_{i+1}] \not\subseteq X_i.$$

Since G is triangle-free, all $N[v_i]$ are prime and all edges in $N[v_i]$ get the same color c_i. By the ordering of the v_k we know that at least one of the edges incident with v_i has an end vertex in $\{v_j \mid 1 \leq j < i\}$ and was therefore colored earlier. □

Remark 2. Note that the proof of the last lemma works for any choice of $v_{i+1} \in X_i$ with $N[v_{i+1}] \not\subseteq X_i$.

If G is the product of n prime factors, at least one of which contains a triangle, then G contains a K_m with $m \geq 3 \cdot 2^{n-1}$.

Theorem 2. *Let $\boxtimes_{i=1}^{n} G_i$ be the PFD of the connected thin graph G. If G does not contain a clique K_m with $m \geq 3 \cdot 2^{n-1}$, then G is NICE and therefore thin-N coverable.*

Proof. By the thinness of G we know that no prime factor of G is isomorphic to K_2, moreover every prime factor is triangle-free. Hence Lemmas 6–8 imply that G is NICE. □

We show now that in thin product graphs, where no prime factor contains triangles, the covering sequence σ can be chosen arbitrarily, as long as each vertex is located on the border of the previously covered vertex set:

Lemma 9. *Let G be a strong product of n triangle-free graphs $G_j (1 \leq j \leq n)$ different from K_2. Then every sequence $\sigma = (x_i)_{i=1}^{k}$ that satisfies the following conditions is a covering sequence:*

(i) *$N[x_i]$ is thin,*
(ii) *$x_{i+1} \in X_i = \bigcup_{j=1}^{i} N[x_j]$, $N[x_{i+1}] \not\subseteq X_i$, and*
(iii) *$X_k = V(G)$.*

Proof. For $n = 1$ the assertion of the lemma is equivalent to Remark 2.

Now to the general case: Let G satisfy the conditions of the lemma. By Lemma 4 the product coloring in every $N[x_i]$ has exactly n different colors.

Now choose any x_i $(i > 1)$. By condition (ii) we know that there is a $j < i$ with $d(x_i, x_j) = 1$. The PFD of G leads to the following coordinatization: $x_i = (x_i(1), x_i(2), \ldots, x_i(n))$, resp. $x_j = (x_j(1), x_j(2), \ldots, x_j(n))$. We denote the n edge colors of $N[x_i]$ by $c(1), c(2), \ldots, c(n)$, in correspondence to its prime factors and coordinates: every Cartesian edge in a G_j-fiber incident with x_i gets the color $c(j)$.

Consider the color $c(l)$. We distinguish two cases:

In the first case x_i and x_j have different lth coordinates. We consider the vertices x_i and $(x_i(1), x_i(2), \ldots, x_j(l), x_i(l+1), \ldots, x_i(n)) =: y$. The edge $[x_i, y]$ is in a G_l-fiber. Thus it is assigned the color $c(l)$ when $N[x_i]$ is colored. Since x_i and x_j are adjacent, $[x_i, y]$ is also contained in $E(N[x_j])$, but this implies that the Cartesian edge $[x_i, y]$ already has a color when $N[x_j]$ is colored.

In the second case x_i and x_j have the same lth coordinate. We can choose any vertex z adjacent to x_i that is in the same G_l-fiber as x_i. Similarly to the first case $[x_i, z]$ has the color $c(l)$, is contained in $\langle N[x_j]\rangle$ and was therefore colored before. □

Notice that we can conclude from Lemma 9 that for thin product graphs G, for which every prime factor is triangle-free, we can find a covering sequence by scanning the vertices of G in Breadth First Search (BFS) order and deleting all vertices that do not satisfy condition (i) or (ii). We believe that this holds for NICE graphs in general, as many examples show, but the proof is still open.

We conclude this section with the observation that thin graphs need neither be NICE nor thin-N coverable. For examples compare Figs. 5 and 6.

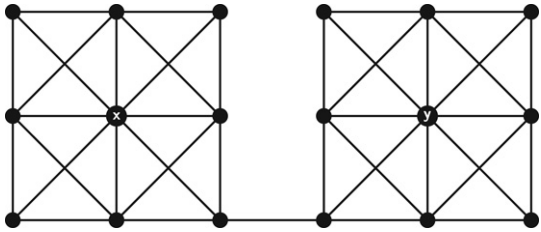

Fig. 5. A graph G that can be covered by thin neighborhoods $\langle N[x]\rangle$ and $\langle N[y]\rangle$. The graph is thin-N coverable, but not NICE because there is no covering sequence.

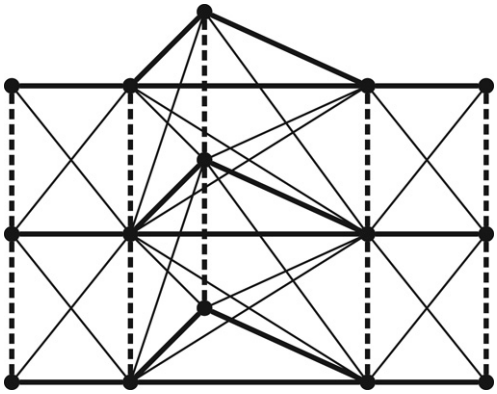

Fig. 6. A thin graph with the property that all induced neighborhoods are not thin, consequently no covering sequence σ exists. The fibers of the prime factors are marked with thick and dashed edges.

6. Fast factorization of NICE graphs

Suppose we are given a NICE graph G with a covering sequence and wish to find its PFD with respect to the strong product. We could factor every induced neighborhood of the vertices in the covering sequence, combine all the product colorings of the induced neighborhoods in order to obtain a partial product coloring of G, and try to use it to find the prime factors of G.

If G has n vertices and if its maximal degree is Δ, then the order of every neighborhood $\langle N[v]\rangle$ is at most $\Delta + 1$. With the algorithm of Feigenbaum and Schäffer [5] it can be factored in $O(\Delta^5)$ time. This algorithm also computes the product coloring of $\langle N[v]\rangle$.

Therefore the factorizations of the neighborhoods of the vertices in the covering sequence have complexity $n \cdot O(\Delta^5) = O(n)$ for constant Δ. If we manage to keep the complexity of all the other operations under control, then this will be the factorization complexity. This is indeed possible, as Algorithm 1 shows.

Lemma 10. *Let G be a NICE graph with given covering sequence σ. Then the first part of Algorithm 1 computes a partial product coloring with at most $\log \Delta$ colors, where Δ is the maximal degree of G. Moreover, the colored edges, together with their endpoints, form a connected, spanning subgraph of G.*

Proof. The algorithm scans the vertices of the covering sequence in their natural order, computes their prime factor decompositions and product colorings, and then combines the colors. In order to keep track of the colors an auxiliary graph, the color graph Γ, is introduced. It is constructed iteratively from the local edge colorings of the induced neighborhoods $\langle N[v]\rangle$, $v \in \sigma$. Its vertices are all pairs (x, c). Two vertices (x', c') and (x'', c'') of Γ are connected by an edge if and only if there is an edge $e \in E$ such that $e \in E_{c'}(x') \cap E_{c''}(x'')$. Edges in Γ therefore "connect" Cartesian edges of local PFDs that belong to the same global prime factor. The connected components Q of the color graph define edge

Algorithm 1 NICE graph decomposition

INPUT: a NICE graph G with a covering sequence σ
initialize a graph $\Gamma = \emptyset$; {called "color graph"}
while $\sigma \neq \emptyset$ **do**
  take the first vertex v of σ;
  factor $\langle N[v]\rangle$ and properly color its Cartesian edges;
  set *num_factors* = the number of factors of $\langle N[v]\rangle$;
  add *num_factors* new vertices to Γ;
  for every colored edge e in $\langle N[v]\rangle$ **do**
    if e was already colored in G **then**
      x = old color of e; y = new color of e;
      join the vertices x and y in Γ;
    end if
  end for
  delete v from σ;
end while
set *num_comp* = number of connected components of Γ;
$I = \{1, \ldots, num\_comp\}$;
$J = I$;
for $k = 1$ to *num_comp* **do**
  for each $S \subset J$ with $|S| = k$ **do**
    compute two connected components H_1, H_2 of G induced by the colored edges of G, which are in components $i \in S$, and $i \in I \backslash S$ of Γ, resp;
    if $\langle H_1\rangle \boxtimes \langle H_2\rangle \backsimeq G$ **then**
      save $\langle H_1\rangle$ as prime factor;
      $J = J \backslash S$;
    end if
  end for
end for
OUTPUT: The prime factors of G;

sets

$$E_Q = \bigcup_{(x,c)\in Q} E_c(x)$$

of G. The edge sets E_Q can be interpreted as the the maximal edge sets to which the local coloring can be extended. We define the global coloring χ of the edges of G by giving unique colors to these sets. In other words, $\chi(e) = Q$ if and only if $e \in E_Q$.

By construction χ is constant on all sets $E_c(x)$, that is, the local colorings of the $\langle N[v]\rangle$ are refinements of the restrictions of the global coloring χ and χ is a partial product coloring.

By the color continuation property of the input sequence we know that the color graph Γ can have at most as many components as there are colors for the first neighborhood. This number is at most $\log_2(\Delta)$, because every product of k non-trivial factors must have at least 2^k vertices.

Finally, the Cartesian edges of every $\langle N[v]\rangle$, $v \in \sigma$, together with their endpoints, form a connected spanning subgraph of $\langle N[v]\rangle$, $v \in \sigma$. Since any two vertices of G are connected via the covering sequence the edges in the domain of χ, together with their endpoints, form a connected spanning subgraph of G. □

In general, the coloring returned by the first part of the algorithm is finer than the edge coloring of the global PFD of G. Thus colors may need to be combined to determine the factors of the global PFD. This is performed in the second part of the algorithm. Figs. 7 and 8 give two examples of prime graphs for which the local coloring algorithm returns more than a single color.

In the graph in Fig. 8 every induced neighborhood $\langle N[x]\rangle$ is a strong product of two factors, but the graph itself is prime. We call this a local product.

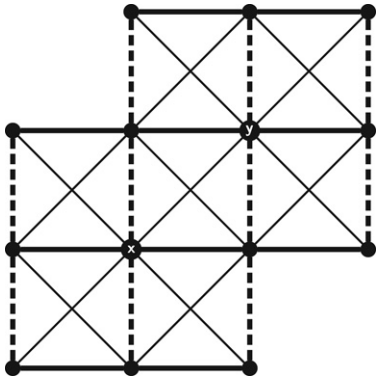

Fig. 7. A NICE graph with covering sequence $\sigma = \{x, y\}$. After running the first part of Algorithm 1 the color graph Γ has two components, although the graph is prime. The components are combined in the second part.

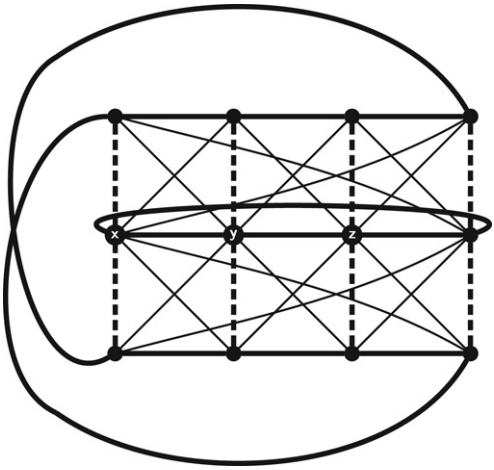

Fig. 8. The so-called twisted product with covering sequence $\sigma = \{x, y, z\}$, with product coloring (induced by thick and dashed edges) after running the first part of Algorithm 1. Again, the components are combined in the second part.

Lemma 11. *Let G be a NICE graph with a given covering sequence σ. Then Algorithm 1 determines the prime factors of G.*

Proof. For every prime factor G_i of the input graph G we have to show that it is returned by our algorithm. It is trivial that for some subset S of components of Γ, S will contain all colors that occur in a particular G_i-fiber G_i^a which contains vertex a. Every vertex $y \in N(x)$ is incident to an edge with every color used in the PFD of $\langle N[x]\rangle$, and hence also with every color of χ on the same edge set. Thus the set of S-colored edges in G_i^a spans G_i^a.

Since the global PFD induces a local decomposition, every fiber in an induced closed neighborhood with respect to a local prime factor is a subset of a fiber with respect to a global prime factor. Thus we never identify colors that occur in copies of different global prime factors. In other words, the coloring χ is a refinement of the product coloring of the global PFD.

This guarantees that a connected component of the graph induced by all edges with a color in S induces a graph that is isomorphic to G_i. The same arguments show that the colors not in S lead to the appropriate cofactor. Thus G_i will be recognized. □

Lemma 12. *For a NICE graph with a given a covering sequence and bounded maximum degree Algorithm 1 determines the prime factors in linear time in the number of vertices.*

Proof. We first consider the coloring part and begin with the while loop.

With the algorithm of Feigenbaum and Schäffer [5] Line 5 takes $O(\Delta^5)$ time. Line 6 takes constant time, and Line 7 $O(\log(\Delta))$ time. Furthermore, since the number of edges in a neighborhood is bounded by $\Delta(\Delta+1)$ and since checking, whether an edge was already colored, can be performed in constant time, the complexity of the for loop is $O(\Delta^2)$.

Thus the total complexity of the while loop is $(n \cdot \Delta^5) = O(n)$. Therefore, the coloring part is linear in the number of vertices.

For the second part we observe that the size of I is the number of components of Γ, which is bounded by $\log \Delta$ by Lemma 10. Hence we also have at most Δ sets S, i.e. color combinations to consider. In Line 21 we have to find connected components of graphs and in Line 22 we have to perform an isomorphism test for a fixed bijection. Both tasks take linear time in the size of the graphs. Thus the total complexity of this part is $O(n \cdot \Delta^2) = O(n)$ again. □

We wish to note that the algorithm of Feigenbaum and Schäffer [5] has complexity $O(n \log n)$ for graphs with bounded degree and order n. This is only slightly weaker than that of our approach, but unfortunately, we could not adapt their method to the recognition of approximate graph products.

7. Approximate graph products

Clearly the graph in Fig. 7 is an approximate graph product to $P_3 \boxtimes P_3$ and the one in Fig. 2 an approximation to $P_4 \boxtimes P_6$. These cases are special in the sense that the approximations can be obtained just by deletion of edges and several vertices from the product in the first case, and by deletion of edges only in the second.

To recognize such approximate products G, Algorithm 1 has to be modified. First note that we might possibly find fibers of the original prime factors, even if we cannot cover the whole input graph by our algorithm. Deleting or adding edges in a product graph usually makes the graph prime and also the neighborhoods $\langle N^G[v]\rangle$ that are different from $\langle N^H[v]\rangle$. We say that such neighborhoods are *disturbed*. In our algorithm we therefore only use thin neighborhoods that are not prime. Thus we require a covering sequence σ of G or of a large subgraph G' of G such that for all $v \in \sigma$ the induced closed neighborhood $\langle N[v]\rangle$ is thin, undisturbed and not prime.

We then use the first part of Algorithm 1. For many kinds of approximate products the connected components of graphs induced by the edges in one component of the color graph Γ will be not isomorphic. In our case, where the approximate product was obtained by deleting edges, it is easy to see that we should take the maximal connected component of each color. Instead of the isomorphism tests of Algorithm 1 we compare the order of the input graph with that of the product of the maximal components. If we assume that the orders of G and H are the same, and if there is no equality, we will merge components of the color graph Γ similarly to Algorithm 1. Otherwise we can relax this equality condition. For the recognition of the graph in Fig. 7 this will be essential.

A special case is the following variant of Problem 1 by Feigenbaum and Haddad [3]:

Problem 2. Let G be an approximate product of $H = \boxtimes_{i=1}^{n} H_i$ and G' be a NICE subgraph of G with a given covering sequence σ. Find the PFD of H.

It can be solved for a large class of graphs by our modified algorithm as the next theorem shows:

Theorem 3. *If G contains a NICE graph G' with a covering sequence of undisturbed vertices, and if G' contains at least one entire fiber of every prime factor of H, then we can determine the prime factors of H by the modification of Algorithm 1 as described above.*

Proof. Let $\boxtimes_{i=1}^{n} H_i$ be the PFD of H. By our modified Algorithm 1 all copies of a prime factor H_i of H that are entirely contained in G' will be colored with the same set of colors S (components of Γ). Hence, a component of the graph induced by the edges with a color in S has the same vertex set as an H_i-fiber. This means that the graph induced by this component is isomorphic to H_i. □

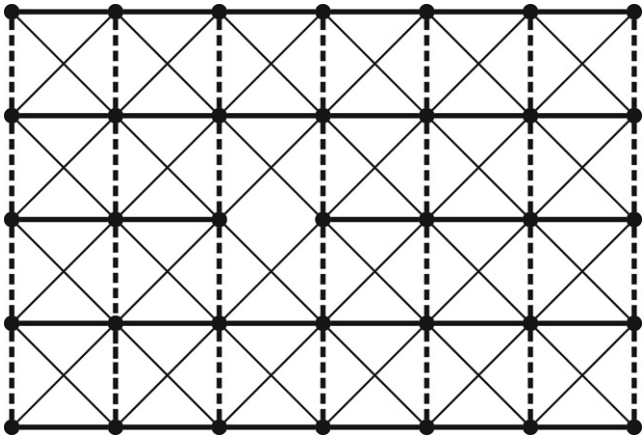

Fig. 9. A graph G obtained by deleting only one edge of $H = P_4 \boxtimes P_6$. Since no NICE subgraph G' of undisturbed neighborhoods of G exists s.t. there is an entire fiber of all factors in G', the modified Algorithm 1 cannot be used to recognize the prime factors of H.

8. Summary and outlook

This paper provides a fast algorithm for the recognition of a large class of approximate graph products. In many cases quite massive perturbations do not hinder us from recognizing the prime factors, e.g. the graph of Fig. 2.

On the other hand, even small perturbations of a product graph can lead to a graph that cannot be treated by our approach, for example the graph of Fig. 9. In part, this is due to our adherence to thin neighborhoods, which we think is not necessary, and which we hope to be able to remove.

Furthermore, due to our modeling of the origin of our approximate products as products that are perturbed by errors in the data or loss of them, it will be a matter of importance to begin a probabilistic study of the recoverability of a product from its perturbed form. The authors share the view that this will show that products of large graphs that are moderately perturbed will be recoverable with very high probability.

Acknowledgements

This work was supported by a collaborative grant of the Austrian *FWF* (Proj. No. P18969-N15) and the German *DFG* (Proj. No. STA850/5-1).

References

[1] J. Cupal, S. Kopp, P.F. Stadler, RNA shape space topology, Artificial Life 6 (2000) 3–23.
[2] W. Dörfler, W. Imrich, Über das starke Produkt von endlichen Graphen, Österreich. Akad. Wiss. Math.-Natur. Kl. Sitzungsber. 178 (1969) 247–262.
[3] J. Feigenbaum, R.A. Haddad, On factorable extensions and subgraphs of prime graphs, SIAM J. Discrete Math. 2 (1989) 197–218.
[4] J. Feigenbaum, J. Hershberger, A.A. Schäffer, A polynomial time algorithm for finding the prime factors of Cartesian-product graphs, Discrete Appl. Math. 12 (1985) 123–138.
[5] J. Feigenbaum, A.A. Schäffer, Finding the prime factors of strong direct product graphs in polynomial time, Discrete Math. 109 (1992) 77–102.
[6] W. Fontana, P. Schuster, Continuity in evolution: On the nature of transitions, Science 280 (1998) 1451–1455.
[7] W. Fontana, P. Schuster, Shaping space: The possible and the attainable in RNA genotype-phenotype mapping, J. Theoret. Biol. 194 (1998) 491–515.
[8] W. Imrich, S. Klavžar, Product graphs, Wiley-Interscience Series in Discrete Mathematics and Optimization, Wiley-Interscience, New York, 2000.
[9] W. Imrich, I. Peterin, Recognizing Cartesian products in linear time, Discrete Math. 307 (2007) 472–483.
[10] R.C. Lewontin, Adaptation, Sci. Am. 239 (1978) 156–169.
[11] R. McKenzie, Cardinal multiplication of structures with a reflexive relation, Fund. Math. 70 (1971) 59–101.

[12] G. Sabidussi, Graph multiplication, Math. Z. 72 (1959/1960) 446–457.
[13] B.M.R. Stadler, P.F. Stadler, The topology of evolutionary biology, in: Ciobanu (Ed.), Modeling in Molecular Biology, in: Natural Computing Series, Springer Verlag, 2004, pp. 267–286.
[14] B.M.R. Stadler, P.F. Stadler, G.P. Wagner, W. Fontana, The topology of the possible: Formal spaces underlying patterns of evolutionary change, J. Theoret. Biol. 213 (2001) 241–274.
[15] V.G. Vizing, The cartesian product of graphs, Vyčisl. Sistemy 9 (1963) 30–43.
[16] G.P. Wagner, P.F. Stadler, Quasi-independence, homology and the unity of type: A topological theory of characters, J. Theoret. Biol. 220 (2003) 505–527.