DISCRETE MATHEMATICS

ELSEVIER Discrete Mathematics 167/168 (1997) 393–403

Recognizing Cartesian graph bundles

Wilfried Imrich[a], Tomaž Pisanski[b,1], Janez Žerovnik[c,*,1]

[a]Montanuniversität Leoben, Institut für Mathematik und Angewandte Geometrie, Franz Jozef-Strasse 18, A-8700 Leoben, Austria
[b]University of Ljubljana, Faculty of Mathematics and Physics, Jadranska 19, Ljubljana, Slovenia
[c]Univerza v Mariboru, Strojna fakulteta, Smetanova 17, 2000 Maribor, Slovenia

Received 7 July 1995; revised 29 April 1996

Abstract

Graph bundles generalize the notion of covering graphs and graph products. In this paper we extend some of the methods for recognizing Cartesian product graphs to graph bundles. Two main notions are used. The first one is the well-known equivalence relation $\delta^\star$ defined on the edge-set of a graph. The second one is the concept of k-convex subgraphs. A subgraph H is k-convex in G, if for any two vertices x and y of distance d, $d \leqslant k$, each shortest path from x to y in G is contained entirely in H. The main result is an algorithm that finds a representation as a nontrivial Cartesian graph bundle for all graphs that are Cartesian graph bundles over a triangle-free simple base. The problem of recognizing graph bundles over a base containing triangles remains open.

1. Introduction

Knowledge of the structure of a graph often leads to faster algorithms for solving combinatorial problems on these graphs. In general, an efficient algorithm for recognizing a special class of graphs may allow us to compute certain graph invariants faster. For example, the chromatic number of a Cartesian product is the maximum of the chromatic numbers of the factors. Computing the chromatic number is in general an NP-hard problem, but factoring can be done in polynomial time. Hence, if the graph is a Cartesian product, we can save computation time by first factorizing and then computing the chromatic numbers of the factors. Here we shall be concerned with the structure of Cartesian graph bundles.

* Corresponding author. E-mail: janez.zerovnik@uni-1j.si.

[1] Also at: IMFM/TCS, Jadranska 19, Ljubljana. Supported in part by the Ministry of Science and Technology of Slovenia (grant No. P1-0105-0101 and grant No. P2-5086-0101).

PII S0012-365X(96)00242-7

Graph bundles [13, 12] generalize the notion of covering graphs and graph products. We note that they can be defined with respect to arbitrary graph products [13]. Various problems on graph bundles were studied recently, including edge coloring, maximum genus, isomorphism classes and chromatic numbers [8–12]. We shall only consider the problem of recognition of Cartesian graph bundles.

It is well known that finite connected graphs enjoy unique factorization under the Cartesian multiplication [14] and recently a number of polynomial algorithms for recognizing Cartesian product graphs have been published [3, 16, 1]. Contrarily, a graph may have more than one presentation as a graph bundle. Natural questions therefore are to find all possible presentations of a graph as a graph bundle or to decide whether a graph has at least one presentation as a nontrivial graph bundle. We will restrict our attention to cases where fibres are connected.

In this note, we present a result on recognizing Cartesian graph bundles. We begin with several definitions and recall some well-known results in Section 2. The main theorem is proved in Section 3. In the last section, we present a polynomial algorithm which finds all so-called minimal presentations of a graph as a Cartesian bundle provided the base graphs do not contain triangles.

2. Preliminaries

In this section we begin with definitions and well-known or easily proved facts. We will consider only connected simple graphs, i.e. graphs without loops and multiple edges.

We say that two edges are *adjacent* if they have a common vertex. Furthermore, $G \cong H$ denotes graph isomorphism, i.e. the existence of a bijection $b : V(G) \to V(H)$ such that vertices v_1, v_2 are connected in G exactly if $b(v_1)$, $b(v_2)$ are connected in H.

The *Cartesian product* $G \square H$ of graphs G and H has as vertices the pairs (v, w) where $v \in V(G)$ and $w \in V(H)$. (v_1, w_1) and (v_2, w_2) are connected if $\{v_1, v_2\}$ is an edge of G and $w_1 = w_2$ or if $v_1 = v_2$ and $\{w_1, w_2\}$ is an edge of H.

Let B and F be graphs. A graph G is a (*Cartesian*) graph bundle with fibre F over the base graph B if there is a mapping $p : G \to B$ which satisfies the following conditions:

(1) It maps adjacent vertices of G to adjacent or identical vertices in B.

(2) The edges are mapped to edges or collapsed to a vertex.

(3) For each vertex $v \in V(B)$, $p^{-1}(v) \cong F$, and for each edge $e \in E(B)$, $p^{-1}(e) \cong K_2 \square F$.

A mapping satisfying just the first two conditions above is called a *graph map*. For a given graph G there may be several mappings $p_i : G \to B_i$ with the above properties. In such cases we write (G, p_i, B_i) to avoid confusion. We say an edge e is *degenerate* if $p(e)$ is a vertex. Otherwise we call it *nondegenerate*. A factorization of a graph G is a collection of spanning subgraphs H_i of G such that the edge set of G is partitioned into the edge sets of the graphs H_i. In other words, the set $E(G)$ can be written as a disjoint union of the sets $E(H_i)$. The projection p induces a factorization of G into the graph consisting of isomorphic copies of the fibre F and the graph $\tilde{G}$ consisting of all nondegenerate edges. This factorization is called the *fundamental factorization*. It can be shown that the restriction of p to $\tilde{G}$ is a covering projection of graphs; see, for instance, [12, 13] for details.

Now we introduce an equivalence relation $\delta^\star$ defined among the edges of a graph. This relation was first used by Sabidussi [14] and later by Feigenbaum et al. [3] as a starting relation in their algorithm for factoring a graph with respect to the Cartesian product. As we show later, this relation can also be used for recognizing graph bundles.

An induced cycle of four vertices is called a *chordless square*. We now define an auxiliary binary relation δ. For any $e, f \in E(G)$ we set $e \,\delta\, f$ if at least one of the following conditions is satisfied:

(1) e and f are the opposite edges of a chordless square.

(2) e and f are adjacent and there is no chordless square spanned on e and f.

By $\delta^\star$ we denote the reflexive and transitive closure of δ. Since δ is symmetric, $\delta^\star$ is an equivalence relation.

Note that any pair of adjacent edges which belong to distinct $\delta^\star$-equivalence classes span a chordless square. It is easy to see that there is exactly one such square. We say that $\delta^\star$ has the *square property*. Furthermore, any equivalence relation $R \supseteq \delta$ also has the square property.

Let R have the square property and let e be an edge. For any edge f not in the same class as e and adjacent to e we can define a *translation* of e along f, $T_f(e)$, to be the (unique) opposite edge of the chordless square spanned by the edges e and f.

Equivalence classes of R will be denoted by Greek letters, possibly equipped by indexes. In particular, the class containing the edge e_i will be denoted by φ_i. We are mainly interested in *nontrivial* equivalence relations R, i.e. equivalence relations having at least two equivalence classes.

Now we recall several well-known facts about the equivalence relation $\delta^\star$; see, for example, [3].

Lemma 1 (Lemma 1 of Feigenbaum et al. [3]). *Each vertex in a connected graph G is incident to at least one edge of each $\delta^\star$ class.*

Lemma 2 (Remark on p. 127 of Feigenbaum et al. [3]). *If the edge $\{u, v\}$ is in class φ_1, then for any other $\delta^\star$-class $\varphi_2 \neq \varphi_1$, the vertices u and v have the same φ_2-degree, and $\delta^\star$ induces a bijection between the φ_2-edges incident to u and φ_2-edges incident to v.*

3. Results

Let R be an equivalence relation on the edge set $E(G)$ of a connected graph G and let φ be an equivalence class of R. Denote by G_φ the spanning subgraph of

G containing the edges of φ and let $G_\varphi(v)$ be the connected component of G_φ that contains $v \in V(G)$.

We define a graph B_φ and a projection $p_\varphi : G \to B_\varphi$ by the following rules:

(1) Let the vertex set of B_φ be $V(B_\varphi) = \{G_\varphi(v) \mid v \in V(G)\}$.

(2) For each vertex $v \in V(G)$ let $p_\varphi(v) = G_\varphi(v)$ and for each edge $e = \{u, v\} \in E(G)$, let $p_\varphi(\{u,v\}) = \{G_\varphi(u), G_\varphi(v)\}$.

(3) There are no other edges in B_φ except those forced by rule (2).

Note that in general B_φ has no parallel edges but it may have loops: at most one per vertex.

Proposition 1. *B_φ has no loops if and only if each connected component is an induced subgraph of G.*

Proof. Obvious. □

We call the triple (G, p, B) a *pre-bundle* if G is connected, $p : G \to B$ is a graph map, B is simple and if for each $e \in E(B)$, $p^{-1}(e)$ is a matching in G.

Let H be a connected subgraph of G. We say that H is *k-convex in G* if for any pair of vertices $u, v \in V(H)$ of distance $d_G(u, v) \leqslant k$ the set of all shortest paths $I_G(u, v)$ from u to v in G is also contained in H: $I_G(u,v) \subseteq I_H(u,v)$. The usual convexity is the same as ∞-convexity and a subgraph is induced if and only if it is 1-convex. Here we are only interested in 1- and 2-convex subgraphs. Note that 2-convex graphs have been studied, for instance, in [5]. For general H, define: H is k-convex in G if and only if each of its connected components is k-convex. Let R be an equivalence relation on $E(G)$ and let φ be an equivalence class of R. We say φ is *k-convex* if G_φ is k-convex. Furthermore, we define R to be *k-convex* if each equivalence class of R is k-convex. R is *weakly k-convex* if at least one equivalence class of R is k-convex.

Proposition 2. *φ is 1-convex if and only if each connected component of G_φ is an induced subgraph of G.*

Proof. Obvious. □

Note that B_φ can, by definition, have no multiple edges. Thus, 1-convexity of equivalence class φ implies that B_φ is a simple graph.

Proposition 3. *φ is 2-convex if and only if $(G, p_\varphi, G_\varphi)$ is a pre-bundle.*

Proof. Assume φ is 2-convex. Hence, G_φ is 2–convex. Since 2-convexity implies 1-convexity, G_φ is an induced subgraph and therefore B_φ is simple by Proposition 2. Furthermore, because of 2-convexity of G_φ, any vertex of G_φ can have at most one neighbor in any other connected component of G_φ. Hence, $p_\varphi^{-1}(e)$ is a matching for any e and $(G, p_\varphi, B_\varphi)$ is a pre-bundle.

Now assume $(G, p_\varphi, B_\varphi)$ is a pre-bundle. Since B_φ is simple, the connected components of G_φ must be induced subgraphs of G (by Proposition 2). It remains to show that the graph G_φ (and hence φ) is 2-convex. Assume that G_φ is not 2-convex. Then there must be a vertex u and a connected component $G_\varphi(v)$ not containing u such that u has at least two neighbours $x, y \in G_\varphi(v)$. Since $p_\varphi(\{x,u\}) = p_\varphi(\{y,u\}) = \{G_\varphi(v), G_\varphi(u)\}$, $p_\varphi^{-1}(\{G_\varphi(v), G_\varphi(u)\}) \supseteq \{\{x,u\}, \{y,u\}\}$ is not a matching which contradicts the assumption that $(G, p_\varphi, B_\varphi)$ is a pre-bundle. Hence G_φ must be 2-convex. □

Lemma 3. *Let R be a weakly 2-convex equivalence relation on $E(G)$ with the square property and let φ be a 2-convex equivalence class of R. Let $e = \{u,v\}$ be an edge from $E(G)\setminus\varphi$. Then e induces a unique isomorphism between $G_\varphi(u)$ and $G_\varphi(v)$.*

Proof. Define the set M_e connecting $G_\varphi(u)$ and $G_\varphi(v)$ as follows:
- $e \in M_e$,
- *if* $e' \in M_e$, $f \in E(G_\varphi(u))$ then $T_f(e') \in M_e$,
- *if* $e' \in M_e$, $f \in E(G_\varphi(v))$ then $T_f(e') \in M_e$,

where $T_f(e)$ is the translation of e along f. Since φ is 2-convex, M_e is a matching. Because $G_\varphi(u)$ and $G_\varphi(v)$ are connected, M_e is a perfect matching on $G_\varphi(u) \cup G_\varphi(v)$ and hence defines a 1–1 map $\alpha : V(G_\varphi(u)) \to V(G_\varphi(v))$. By Lemma 2, we can verify that $\alpha : G_\varphi(u) \to G_\varphi(v)$ is a local isomorphism which in turn implies that it is an isomorphism. □

Theorem 1. *Let G be any graph and R any nontrivial weakly 2-convex equivalence relation having the square property with φ being a 2-convex equivalence class of R. Then $(G, p_\varphi, B_\varphi)$ is a graph bundle.*

Proof. By Proposition 3, $(G, p_\varphi, B_\varphi)$ is a pre-bundle. It remains to show that for each $e = \{a,b\} \in E(B_\varphi)$ the matching $p^{-1}(e)$ induces an isomorphism between two connected components $G_\varphi(u)$ and $G_\varphi(v)$ such that $p(u) = a$ and $p(v) = b$. Since $p^{-1}(e)$ is M_e of the previous lemma this concludes the proof. □

The theory developed so far can now be used for representing graph G as a graph bundle. We start with $\delta^\star$ and then glue some equivalence classes together as long as the resulting equivalence relation R does not satisfy the conditions of the theorem. We will later give an algorithm which will use this approach for recognizing graph bundles. Unfortunately, this approach does not recognize all graph bundles. For example, take the complete bipartite graph $K_{3,3}$. It has trivial $\delta^\star$ but it is a graph bundle with fibre K_2 over base K_3. The reason is that K_3 contains a triangle. As we show later, the existence of triangles in the base graph is the only case in which our approach may fail.

On the other hand, if (G, p, B) is a graph bundle whose base graph B has no triangles, then each $\delta^\star$ equivalence class either contains only degenerate edges or only nondegenerate edges.

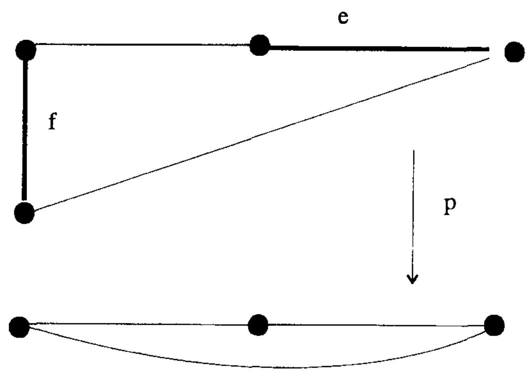

Fig. 1. Degenerate and nondegenerate edge in relation $\delta^\star$.

To show this, let R_1 be the union of $\delta^\star$ classes containing degenerate edges and let R_2 be the union of $\delta^\star$ classes containing nondegenerate edges. We claim that R_1 and R_2 have empty intersection. Assume there is a $\delta^\star$ equivalence class contaning a degenerate edge e' and a nondegenerate edge f'. Then there must be a pair e, f of edges such that e is degenerate, f is nondegenerate and $e\delta f$.

We have two different cases to consider. First, if e and f are adjacent then there must be a chordless square spanned by e and f since two adjacent fibres induce a Cartesian product with K_2. But this is not possible by the definition of δ.

The second case occurs when e and f are opposite edges of a chordless square. Then it is easily seen that there must be a triangle in the base graph; see Fig. 1. Therefore no $\delta^\star$ class can contain both degenerate and nondegenerate edges. We formulate this as a lemma.

Lemma 4. *Let (G, p, B) be a graph bundle whose base graph B has no triangles. Then each $\delta^\star$ equivalence class contains either only degenerate edges or only nondegenerate edges. In particular, $\delta^\star$ is not trivial.*

Let R be any equivalence relation with the square property and let φ be any of its classes. We define the closure $\mathscr{C}_2(\varphi,R)$ as the subset ρ of the edge set $E(G)$, such that ρ is the minimal union of equivalence classes of R, that satisfies the following two conditions: (1) $\varphi \subseteq \rho$ and (2) ρ is 2-convex in G. In order to justify the above definition we must show that the 2-convex closure is well defined. It suffices to prove that the intersection of 2-convex subgraphs is 2-convex.

Lemma 5. *If two subgraphs C_1 and C_2 are 2-convex, then the intersection $C_1 \cap C_2$ is 2-convex.*

Proof. Let $u, v \in C_1 \cap C_2$. If u and v are adjacent in G, then the edge $\{u, v\}$ must be in both C_1 and C_2. If u and v are at distance 2 apart, then any 2-path between u and v must be both in C_1 and in C_2 because C_1 and C_2 are 2-convex. □

Let H be a subgraph of G. An edge e from $G \setminus H$ that belongs to a shortest 1- or 2-path of G having both endpoints in the same connected component of H is called an *obstruction* of H. Let $\mathcal{O}(H)$ denote the set of all obstructions of H. Clearly, H is 2-convex if and only if $\mathcal{O}(H) = \emptyset$.

Let R be an equivalence relation on $E(G)$ and $\rho \subseteq E(G)$ an arbitrary set of edges. If an equivalence class φ of R contains at least one edge from ρ we say that φ *meets* ρ. Define two operators S and T as follows:

$$S(\rho, R) := \bigcup_{\varphi_j \cap \rho \neq \emptyset} \varphi_j; \qquad T(\rho, R) := \{S(\rho, R)\} \cup \{\varphi_k \mid \varphi_k \cap \rho = \emptyset\}.$$

Clearly, $S = S(\rho, R)$ represents the union of all equivalence classes of R that meet ρ and T is a new equivalence relation on $E(G)$ obtained from R by merging the equivalence classes that meet ρ into a single class S. $T(\rho, R)$ is called the *R-closure* of ρ. The set ρ is *R-closed* if $T(\rho, R) = R$.

Here is an algorithm A for computing $\mathscr{C}_2(\varphi, R) = \rho := \mathbf{A}(G, R, \varphi)$ for any graph G and an arbitrary set of edges $\varphi \subseteq E(G)$. Later it is used only in the case when φ is an equivalence class of R.

Algorithm A:

Input: G: graph,

R: equivalence relation with the square property on $E(G)$,
given as a partition of $E(G)$ into equivalence classes,

φ: subset of $E(G)$

Output: $C_2(\varphi, R)$

1. $k := 0$
2. $\rho_k := \varphi$
3. $\gamma_k := \mathcal{O}(\rho_k)$
4. $R_k := R$
5. **while** $\gamma_k \neq \emptyset$ **do**
   - 5.1 $\rho_{k+1} := S(\rho_k \cup \gamma_k, R_k)$
   - 5.2 $R_{k+1} := T(\rho_k \cup \gamma_k, R_k)$
   - 5.3 $\gamma_{k+1} := \mathcal{O}(\rho_{k+1})$
   - 5.4 $k := k + 1$

end-while

6. **return**(ρ_k)

Lemma 6. *Let G be a graph bundle whose base graph contains no triangles and let φ be any equivalence class of $\delta^\star$ containing only degenerate edges. If $\rho := C_2(\varphi, \delta^\star) \neq E(G)$, then G is a graph bundle with fibres being the connected components of G_ρ.*

Proof. Since each connected component of G_ρ is an induced subgraph of G, every edge of $E(G)\setminus\rho$ has its endpoints in distinct connected components of G_ρ. The equivalence relation with two equivalence classes $\{\rho, E(G)\setminus\rho\}$ is weakly 2-convex. Therefore, by Lemma 3, all pairs of connected components of G_ρ are pairwise isomorphic. □

Lemma 7. *Let G be a graph bundle with fibre F. Assume each equivalence class of $\delta^\star$ contains either only degenerate or nondegenerate edges and let γ be any equivalence class of $\delta^\star$. If a connected component of the graph determined by γ is contained in a fibre, then also the connected component of the 2-convex closure $\mathscr{C}_2(\gamma, R)$ is contained in a fibre. In particular, the graph determined by the 2-convex closure of γ has at least two connected components.*

Proof. We show that if a connected component of an arbitrary subgraph $H\subseteq G$ is contained in a fibre, then also the connected component of the 2-convex closure is contained in the same fibre. Since the argument is valid for each connected component of H, we may assume without loss of generality, that H is connected. Let H be a connected subgraph of a fibre F_1. Let H' be obtained from H by a step of the algorithm A. The obstructions are either edges or 2-paths.

(1) If an edge was added, then this edge must also be in F_1 because fibres are induced subgraphs. Furthermore, any other edge of the same $\delta^\star$-equivalence class adjacent to a vertex of H' must be in F_1. If not, then this $\delta^\star$-equivalence class would contain both degenerate and nondegenerate edges which we have assumed not to be the case.

(2) If a 2-path (with a new vertex $v\notin F_1$) was added to H, then we have a vertex v in another fibre, say F_2 connected to a pair of vertices of H and hence of F_1. But since G is a graph bundle a vertex cannot have more than one neighbour in another fibre. Hence, no edge not belonging to F_1 can be added and H' must also be a subgraph of F_1.

Thus, all obstructions are in F and the edges of the obstructions degenerate. Since each class of $\delta^\star$ contains by assumption either only degenerate or only nondegenerate edges, the $\delta^\star$ closure contains only degenerate edges. □

If there is a graph B with no triangles, such that (G, p, B) is a graph bundle for some p, we can now give a polynomial algorithm which finds at least one representation of G as a bundle. In fact, by computing the closures of all $\delta^\star$ equivalence classes, we can find all *minimal* representations of G as a graph bundle.

Algorithm B:
Input: G: graph;
Output: C: set of degenerate edges of some bundle representation.
1. compute $\delta^\star$
2. **for all** equivalence classes φ of $\delta^\star$ **do**
    2.1 **if** $C := \mathscr{C}_2(\varphi, \delta^\star) \neq E(G)$ **then return** (C)
end-for
3. **return** ('G is not a bundle over a K_3-free base.')

Theorem 8. *Let G be a graph which can be represented as a graph bundle with triangle-free base. Then algorithm B returns* $\mathscr{C}_2(\varphi,\delta^\star)$, *the set of degenerate edges of* $(G, p_\varphi, G_\varphi)$, *for at least one representation of G as a graph bundle.*

Proof. For any representation with a triangle-free base, the equivalence classes of the relation $\delta^\star$ contain either only degenerate or only nondegenerate edges by Lemma 4. Let φ be an equivalence class of $\delta^\star$ with degenerate edges. Each connected component must be contained in one fibre and by Lemmas 6 and 7 the closure $\mathscr{C}_2(\varphi,\delta^\star)$ is the set of degenerate edges for a representation of G as a graph bundle. □

Remark. Algorithm B may also produce a representation with a base containing a triangle. Indeed, the example where degenerate and nondegenerate edges are in the relation $\delta^\star$ is $K_{3,3}\setminus e$, i.e. a $K_{3,3}$ from which an edge has been deleted. A more precise characterization of the graph bundles, not recognized by the algorithm B is the following: There must be a triangle in the base graph and the composition of the three isomorphisms between fibres over that triangle (which is an automorphism on one copy of fibre) must map at least one vertex to one of its neighbors.

A design of an optimal algorithm for graph bundle recognition problem is an interesting research topic as there exist excellent algorithms with known complexity $O(m \log n)$ [1] or $O(mn)$ [2] for the special case, the recognition of Cartesian product graphs.

We implemented an early version of the algorithms A and B in Mathematica It is now part of the Vega Package [15], a system for doing discrete mathematics. Although we did not pay too much attention to speeding up the running time we can easily prove that our algorithm B computes all minimal fibres in polynomial time.

Step 1 of algorithm B, computing the relation $\delta^\star$ is well known to be polynomial; see, for instance, [3].

The number of iterations of the loop 2 of algorithm B is equal to the number of $\delta^\star$ equivalence classes, which is, by Lemma 1, bounded by the minimal degree of a vertex of a graph.

Algorithm A is called in each iteration of the loop 2.

Polynomial running time of the algorithm A follows from the following observations:

- Obstructions are edges, therefore the total number of obstructions is bounded by the number of edges in G.
- Every obstruction is 'used' at most once. This bounds the number of iterations of the **while** loop in step 5.
- Updating the set of obstructions $\mathcal{O}$ needs at most checking each edge and each 2-path in G and is definitely polynomial.
- Computing S and T is computing a union of two sets and then computing a transitive closure of a certain relation.

This shows that the time complexities of algorithms A and B are bounded by a polynomial in n, the number of vertices of G.

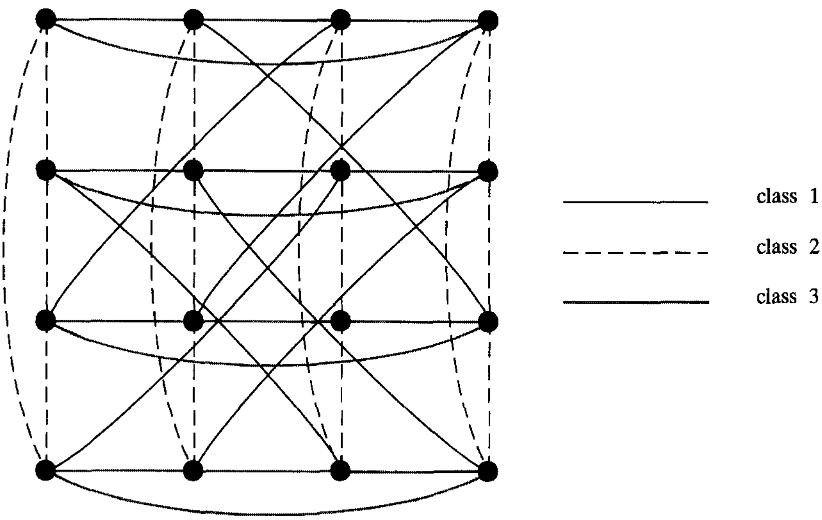

Fig. 2. Union of fibres is not a fibre.

We conclude with some observations on the structure of all representations of G as a graph bundle. By starting with different equivalence classes of $\delta^\star$ in the algorithm B we obtain some fibres, which we call minimal fibres. Of course, there may be more representations of G as a Cartesian graph bundle. Clearly, given a graph G, the set of all possible fibres is partially ordered by inclusion (because they are all unions of $\delta^\star$ equivalence classes). Hence, we can speak of minimal and maximal fibres. The union of two fibres is not neccessarily a fibre. For example, the graph on Fig. 3 has three $\delta^\star$ equivalence classes. It can be represented as a graph bundle taking the edges of class 1 or edges of class 2 or edges of class 3 as fibres. However, if we take the union of any two classes, the graph obtained has only one connected component and is not a fibre. There are also examples where the union of fibres has more than one connected component, but it is not an induced subgraph any more. Let H be the graph in Fig. 3 and define $G = H\square K_2$. Now the union of classes 1 and 3 (of $\delta^\star$ in $E(G)$) has two connected components, but it is not an induced subgraph, as class 2 edges have to be added to get two fibres isomorphic to H in the (product) bundle $H\square K_2$.

It can be shown that the intersection (if nonvoid) of two fibres is a fibre. Hence, if we know all maximal fibres, we probably have information on all possible fibres. It seems that the maximal fibres are more difficult to find than the minimal fibres.

Since the union of two fibres is not neccessarily a fibre we may try to extend each of the known fibres with any other equivalence class of $\delta^\star$ and compute the closure defined above. However, the time complexity of such an algorithm is no more polynomial, since we may have to repeat it too many times.

We gave a rather simple algorithm for recognizing Cartesian bundles with triangle-free base. It is natural to pose:

Problem 1. How complicated is it to recognize Cartesian bundles over arbitrary base graphs?

We know it is no problem for our algorithm to recognize graphs which have no induced $K_{3,3}\setminus\{e\}$. A straightforward approach therefore would be to detect $K_{3,3}\setminus\{e\}$ in G and then in some way 'disable' the edges involved so that any pair of degenerate and nondegenerate edges would not be related.

Problem 2. How difficult is recognition of graph bundles with respect to strong or other graph products?

References

[1] F. Aurenhammer, J. Hagauer and W. Imrich, Cartesian graph factorization at logarithmic cost per edge, Comput. Complexity 2 (1992) 331–349.

[2] T. Feder, Product graph representations, J. Graph Theory 16 (1992) 467–488.

[3] J. Feigenbaum, J. Hershberger and A.A. Schäffer, A polynomial time algorithm for finding the prime factors of Cartesian-product graphs, Discrete Appl. Math. 12 (1985) 123–138.

[4] R.L. Graham and P.M. Winkler, On isometric embeddings of graphs, Trans. Amer. Math. Soc. 288 (1985) 527–536.

[5] J. Hagauer, W. Imrich and S. Klavžar, Recognizing graphs of windex 2, preprint, 1993.

[6] W. Imrich, Embedding graphs into Cartesian products, graph theory and applications: East and West, Ann. New York Acad. Sci. 576 (1989) 266–274.

[7] W. Imrich and J. Žerovnik, Factoring Cartesian-product graphs, J. Graph Theory 18 (1994) 557–567.

[8] S. Klavžar and B. Mohar, Coloring graph bundles, J. Graph Theory 19 (1995) 145–155.

[9] S. Klavžar and B. Mohar, The chromatic numbers of graph bundles over cycles, Discrete Math. 138 (1995) 301–314.

[10] J.H. Kwak and J. Lee, Isomorphism classes of graph bundles, Canadian J. Math. 42 (1990) 747–761.

[11] B. Mohar, T. Pisanski and M. Škoveira, The maximum genus of graph bundles, European J. Combin. 9 (1988) 301–314.

[12] T. Pisanski, J. Shawe-Taylor and J. Vrabec, Edge-colorability of graph bundles, J. Combin. Theory Ser. B 35 (1983) 12–19.

[13] T. Pisanski and J. Vrabec, Graph bundles, unpublished manuscript, 1982.

[14] G. Sabidussi, Graph multiplication, Math. Z. 72 (1960) 446–457.

[15] Vega Version 0.2; Quick Reference Manual and Vega Graph Gallery (T. Pisanski, ed.), IMFM, Ljubljana, 1995.

[16] P.M. Winkler, Factoring a graph in polynomial time, European J. Combin. 8 (1987) 209–212.