

NZ-Flows in Strong Products of Graphs

Wilfried Imrich,¹ Iztok Peterin,² Simon Špacapan,³ and Cun-Quan Zhang⁴

¹MONTANUNIVERSITÄT LEOBEN
A-8700 LEOBEN, AUSTRIA
E-mail: imrich@unileoben.ac.at

²UNIVERSITY OF MARIBOR
FEECS, SMETANOVA 17
2000 MARIBOR, SLOVENIA
E-mail: iztok.peterin@uni-mb.si

³UNIVERSITY OF MARIBOR
FME, SMETANOVA 17
2000 MARIBOR, SLOVENIA
E-mail: simon.spacapan@uni-mb.si

⁴DEPARTMENT OF MATHEMATICS, WEST VIRGINIA UNIVERSITY
MORGANTOWN, WEST VIRGINIA 26506-6310
E-mail: cqzhang@math.wvu.edu

Received February 26, 2007; Revised May 29, 2009

Published online 20 October 2009 in Wiley InterScience (www.interscience.wiley.com).

DOI 10.1002/jgt.20455

Abstract: We prove that the strong product $G_1 \boxtimes G_2$ of G_1 and G_2 is \mathbb{Z}_3 -flow contractible if and only if $G_1 \boxtimes G_2$ is not $T \boxtimes K_2$, where T is a tree (we call $T \boxtimes K_2$ a K_4 -tree). It follows that $G_1 \boxtimes G_2$ admits an NZ 3-flow unless $G_1 \boxtimes G_2$ is a K_4 -tree. We also give a constructive proof that yields a polynomial algorithm whose output is an NZ 3-flow if $G_1 \boxtimes G_2$ is not a K_4 -tree, and an NZ 4-flow otherwise. © 2009 Wiley Periodicals, Inc. *J Graph Theory* 64: 267–276, 2010

MSC: 05C15; 05C75; 05C38

Keywords: integer flows; strong product; paths and cycles

Journal of Graph Theory

© 2009 Wiley Periodicals, Inc.

1. INTRODUCTION AND PRELIMINARIES

All graphs in this paper are simple and connected. A graph is *trivial* if it is isomorphic to K_1 . For the path on n vertices and the cycle of length n we use the notation P_n and C_n , respectively. A graph is *even* or *Eulerian* if all of its vertices are of even degree. Let H be a connected subgraph of G . The *quotient graph* of G by H is the graph G/H obtained from G by contracting all edges in H and deleting possible loops that might appear.

Let D be an orientation of a graph G , and f a function from $E(G)$ to \mathbb{Z} with $-k < f(e) < k$ for every $e \in E(G)$. Then the pair (D, f) is a k -flow if it satisfies the *Kirchhoff condition*

$$\sum_{e \in E^+(v)} f(e) = \sum_{e \in E^-(v)} f(e)$$

for every vertex $v \in V(G)$, where $E^+(v)$ and $E^-(v)$ denote the sets of outgoing and incoming edges of v with respect to D . A k -flow (D, f) is *nowhere-zero*, *NZ* for short, if $f(e) \neq 0$ for every $e \in E(G)$.

Analogously, an *NZ \mathbb{Z}_k -flow* is a function $f: E(G) \rightarrow \mathbb{Z}_k$, such that $\sum_{e \in E^+(v)} f(e) = \sum_{e \in E^-(v)} f(e)$ for all $v \in V(G)$, and $f(e) \neq 0$ for all $e \in E(G)$. We say that a graph G is *\mathbb{Z}_k -flow contractible* if for every supergraph G' (i.e. a graph G' that contains G as a subgraph) and every *NZ \mathbb{Z}_k -flow* f in G'/G there exists an *NZ \mathbb{Z}_k -flow* g in G' such that $g(e) = f(e)$ for all edges $e \in E(G'/G)$. This in particular implies that \mathbb{Z}_k -flow contractible graphs admit an *NZ \mathbb{Z}_k -flow* (by setting $G' = G$).

The definition of a flow depends on the given orientation D , but it is easily seen that a graph that admits an *NZ k -flow* with respect to an orientation D also admits an *NZ k -flow* with respect to any other orientation. Clearly a graph is Eulerian if and only if it admits a nowhere-zero 2-flow. Furthermore, if G admits an *NZ k -flow*, then G also admits an *NZ ℓ -flow* for any $\ell > k$. It is well known that a graph admits an *NZ k -flow* if and only if it admits an *NZ \mathbb{Z}_k -flow*. Therefore, \mathbb{Z}_k -flow contractibility of a graph G is stronger than existence of an *NZ k -flow* in G .

The concept of *NZ k -flows* was introduced by Tutte [7, 8]. Tutte [7] showed that a cubic graph admits an *NZ 3-flow* if and only if it is bipartite. Tutte [8] also proved that a plane graph admits an *NZ k -flow* if and only if it is face k -colorable.

This concept proved to be very fruitful. There exists an extensive literature and many famous open problems about *NZ flows*. Best known are probably three conjectures of Tutte, of which the oldest, from 1954, asserts that every bridgeless graph admits an *NZ 5-flow*. For a comprehensive general treatment of the subject, we recommend the book [9] by Zhang.

Recently several results on flows in Cartesian and direct products of graphs were published, but for both products there are cases where the minimum k for which a *NZ k -flow* exists is not known a priori. Nonetheless, for the strong product a concise complete answer is possible. It is the main result of this paper.

We now state the two recent results for the Cartesian and the direct product and show how they relate to the strong product.

In [4, 5] it is shown that the Cartesian product of nontrivial graphs admits NZ 4-flow in general and an NZ 3-flow if one factor is not an odd-circuit tree and the other does not contain a bridge. In [10] it is shown that the direct product of nontrivial graphs admits an NZ 3-flow if one factor does not have any leafs and the other is not a member of a certain family of trees.

The *strong product* $G_1 \boxtimes G_2$ of two graphs G_1 and G_2 is the graph with the vertex set $V(G_1) \times V(G_2)$, where two distinct vertices (x_1, x_2) and (y_1, y_2) are adjacent in $G_1 \boxtimes G_2$, if x_i is either equal or adjacent to y_i in G_i for $i=1,2$. Note that the edge set of the strong product is the union of the edge sets of the Cartesian and the direct product, see e.g. [3].

Let u be a vertex of G . Then the subgraph of $G \boxtimes H$ induced by all vertices that have u as the first coordinate is clearly isomorphic to H . It is called an *H-fiber* and denoted by H_u . The *G-fibers* are defined analogously. For an edge $e=uv \in E(G)$, we denote the subgraph of $G \boxtimes H$ induced by the set $\{u, v\} \times V(H)$ by $e \boxtimes H$.

Since the edge set of the strong product of two graphs G and H is the (disjoint) union of the edge sets of the Cartesian product with the direct product, the above results about the Cartesian and the direct product already cover many cases of the strong product. To see this we only have to observe that an NZ k -flow f on (V, E) and an NZ k -flow g on (V, F) yield an NZ k -flow h on $(V, E \cup F)$ if E and F are disjoint. One just has to set $h(e)=f(e)$ if $e \in E$ and $h(e)=g(e)$ if $e \in F$.

Thus, one way to solve the problem for the strong product would be to analyze the remaining cases, such as the strong product of a graph with a bridge by an arbitrary graph. In this paper we take the direct approach and prove a stronger result which gives a characterization of \mathbb{Z}_3 -flow contractible strong products. Then we give also an algorithmic proof that yields a polynomial algorithm for NZ-flows in strong products. We say that a graph is a *K₄-tree* if it is a strong product of a tree by K_2 . Our main results are the following theorems.

Theorem 1. *Let G be a strong product of two nontrivial connected graphs. Then*

- (i) *G is \mathbb{Z}_3 -flow contractible if and only if G is not a K_4 -tree.*
- (ii) *G admits a nowhere-zero 3-flow if and only if G is not a K_4 -tree.*

Let G_1 and G_2 be graphs. We denote by n_i , m_i , and Δ_i the number of vertices, the number of edges, and the maximum degree of G_i , respectively. The number of edges of $G_1 \boxtimes G_2$ will be denoted by m .

Theorem 2. *There exists an algorithm whose input are two connected nontrivial graphs G_1 and G_2 and whose output is an NZ k -flow in $G_1 \boxtimes G_2$, where*

- (a) *$k=2$ if G_1 and G_2 are even graphs. The algorithm runs in $O(m)$ time.*
- (b) *$k=4$ if $G_1 \boxtimes G_2$ is a K_4 -tree with $G_2=K_2$. The algorithm runs in $O(m_1)$ time.*
- (c) *$k=3$ if G_1 and G_2 are trees different from K_2 . The algorithm runs in $O(m_1+m_2)$ time.*
- (d) *$k=3$ and one of the factors, say G_1 , is not a tree. The algorithm runs in $O(m_1+n_2m_2\Delta_2)$ time.*

If only the product $G = G_1 \boxtimes G_2$ is given, but not the decomposition, then we can use the algorithm of Feigenbaum and Schäffer [2, 3] to decompose G into its prime factors with respect to the strong product and subsequently apply Theorems 1 and 2. The algorithm of Feigenbaum and Schäffer runs in $O(n^5)$ time, where $n = |G|$, but is almost linear for graphs of bounded degree.

We will prove the above theorems in the following two sections.

2. \mathbb{Z}_3 -FLOW CONTRACTIBILITY OF STRONG PRODUCTS

In this section we prove Theorem 1. Since \mathbb{Z}_3 -flow contractibility of G is stronger than the existence of an NZ 3-flow in G , it suffices to prove the (\Leftarrow) part of (i) and the (\Rightarrow) part of (ii) of Theorem 1.

Let H_1 and H_2 be connected graphs. A 2-sum of H_1 and H_2 is a graph G obtained from H_1 and H_2 by identifying an edge e_1 of H_1 with an edge e_2 of H_2 (note that by doing this the endvertices of e_1 are identified with the endvertices of e_2). If G is a 2-sum of H_1 and H_2 we write $G = H_1 \oplus_2 H_2$. In particular, if $G = W \oplus_2 H$ for an odd wheel W , we say that W is a *separable odd wheel* of G . We say that G is *triangularly connected* if for any two edges e_1 and e_2 in G , there is a sequence of triangles T_1, \dots, T_k , such that T_{i-1} and T_i share a common edge and $e_1 \in E(T_1)$, $e_2 \in E(T_k)$. Note that every strong product of connected graphs is triangularly connected. A wheel W_k is the graph obtained from a k -cycle by adding a new vertex, called the center of the wheel, which is joined to every vertex of the k -cycle. W_k is an odd (even) wheel if k is odd (even). For technical reasons, a single edge is regarded as 1-cycle, and thus W_1 is a triangle, called the trivial wheel. We first state two results proved in [1].

Theorem 3 (Theorem 1.4, Fan et al. [1]). *Let G be a triangularly connected graph with $|V(G)| \geq 3$. Then G is not \mathbb{Z}_3 -flow contractible if and only if there is an odd wheel W and a subgraph H such that $G = W \oplus_2 H$, where H is triangularly connected and not \mathbb{Z}_3 -flow contractible.*

Theorem 4 (Theorem 1.5, Fan et al. [1]). *Let G be a triangularly connected graph with $|V(G)| \geq 3$. Then G has no nowhere-zero 3-flow if and only if there is a nontrivial odd wheel W and a subgraph H such that $G = W \oplus_2 H$, where H is a triangularly connected graph without an NZ 3-flow.*

We also use a lemma which follows from results in [6].

Lemma 5. *If G_1 and G_2 are connected nontrivial graphs, then every vertex separator in $G_1 \boxtimes G_2$ of size two is of the form $V(G_1) \times \{y\}$ or $\{x\} \times V(G_2)$, where $\{x\}$ and $\{y\}$ are vertex separators in G_1 and G_2 , respectively.*

Next we prove Theorem 1.

Proof of the (\Leftarrow) part of (i): Suppose that the assertion is not true, and let $G = G_1 \boxtimes G_2$ be a strong product of two connected nontrivial graphs such that G is not

a K_4 -tree, not \mathbb{Z}_3 -flow contractible, and $|E(G_1)| + |E(G_2)|$ is minimum possible. Note that this implies that $|E(G_1)| + |E(G_2)| > 2$, for otherwise G is a K_4 -tree. By Theorem 3 there is a separable odd wheel W in G . Clearly, a separable odd wheel yields a vertex separator of size two. It follows from Lemma 5 and the fact that the only strong product which is a wheel is K_4 , that $G_1 = K_2$, and that $W = K_4$ is a subproduct in G . In particular, all separable odd wheels are subproducts isomorphic to K_4 . Since G is not \mathbb{Z}_3 -flow contractible $G = H \oplus_2 K_4$, where $W = K_4$ is a separable wheel and H is not \mathbb{Z}_3 -flow contractible (see Theorem 3). By the minimality of G , $H = G_1 \boxtimes (G_2 - v)$ is a K_4 -tree. Since v is a vertex of degree one in G_2 , this yields a contradiction to the assumption that G is not a K_4 -tree.

Proof of the (\Rightarrow) part of (ii): We prove that a K_4 -tree does not admit an NZ 3-flow. Suppose the contrary, and let G be a smallest (with respect to the number of vertices) K_4 -tree that admits an NZ 3-flow. Assume that $G = G_1 \boxtimes G_2$, where $G_1 = K_2$ (and note that G_2 has at least three vertices). Clearly $G = K_4 \oplus_2 H$, where $H = G_1 \boxtimes (G_2 - v)$ and v is a leaf of G_2 . Since H is a K_4 -tree smaller than G , we infer that H does not admit an NZ 3-flow. This contradicts Theorem 4. ■

3. PROOF OF THEOREM 2

This section provides a construction of NZ flows according to Theorem 2. First let us deal with the easy part of the theorem. Since every even graph with m edges can be decomposed into an edge-disjoint union of cycles in m steps, the (a) part of Theorem 2 follows. Now we provide an algorithm for the other cases.

A. Flows If One Factor Is Not a Tree

We first introduce the generalized strong product of graphs G and H . The vertex sets of graphs G and H are partitioned into two subsets each, say G_a, G_i and H_a, H_i . The vertices of G_a and H_a are called *active* and vertices of G_i and H_i are called *inactive*. The *generalized strong product* $G \boxtimes_w H$ of G and H is the graph obtained from $G \boxtimes H$ by deletion of all edges $(x_1, y_1)(x_2, y_2) \in E(G \boxtimes H)$, where $x_1 = x_2 \in G_i$ or $y_1 = y_2 \in H_i$.

Lemma 6. *Let G be a connected graph that is not a tree and P a path of length n . If all vertices of G are active and at least the endvertices of P , then the generalized strong product $G \boxtimes_w P$ admits an NZ 3-flow. The construction of the NZ 3-flow in $G \boxtimes_w P$ requires $O(m+n)$ time, where m is the number of edges of G .*

Proof. We will define a 3-flow h on $E(G \boxtimes P)$ such that $h(e) = 0$ if and only if $e \in E(G \boxtimes P) - E(G \boxtimes_w P)$. Thus the restriction of h to $E(G \boxtimes_w P)$ is an NZ 3-flow on $G \boxtimes_w P$. Let $K_2 = \{u, v\}$.

We first define an NZ 3-flow on $G \boxtimes K_2$. Let T be a rooted spanning tree of G , such that the root r lies on a cycle of G and $e \notin E(T)$ be an edge incident with r . We orient all edges of T away from r and e toward r (see Fig. 2). For every edge $t \in E(T) \cup \{e\}$, we put the flow f (defined on Fig. 1) on $t \boxtimes K_2$ in $G \boxtimes K_2$. If $t' \notin E(T) \cup \{e\}$ we put the

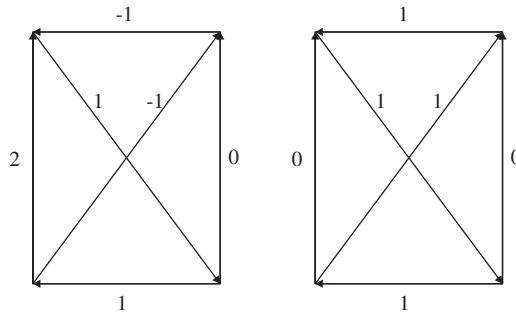


FIGURE 1. 3-flows f (left) and g (right) on $K_2 \boxtimes K_2$.

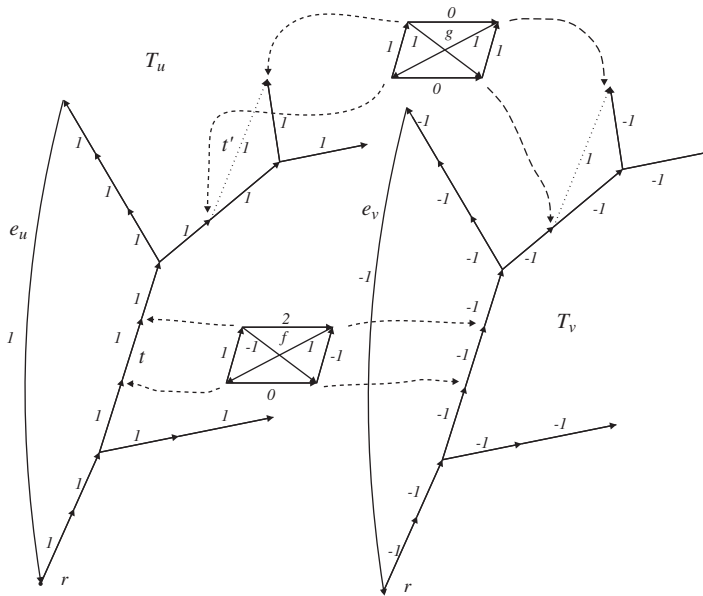


FIGURE 2. The product $G \boxtimes K_2$. The edge t' is marked on the left copy of G .

flow g (also defined on Fig. 1) on $t' \boxtimes K_2$ in the product $G \boxtimes K_2$; see Figure 2. Adding the values of different flows defined on an edge we obtained an NZ 3-flow on $G \boxtimes K_2$, where the flow values of every edge in $E(T_u) \cup \{e_u\}$ is 1 and the flow value of edges in $E(T_v) \cup \{e_v\}$ is -1 . The flow value of the other edges of G_u and G_v is 1. Note that for the edges $t' \notin E(T) \cup \{e\}$ we can also use $-g$ to obtain an NZ 3-flow on $G \boxtimes K_2$. Since finding a spanning tree in G requires m steps (it suffices to run a BFS algorithm on G), this part can be done in linear time.

For $a, b \in \{-1, 1\}$ let $h_{a,b}$ be the NZ 3-flow defined on $G \boxtimes K_2$ such that for every edge of $T_u \cup \{e_u\}$, $h_{a,b} = a$, for every edge of $T_v \cup \{e_v\}$, $h_{a,b} = -a$, and $h_{a,b} = b$ for all the other edges of G_u and G_v . Let $P = v_1 v_2 \dots v_n$. Put the flow $h_{1,1}$ on $G \boxtimes \{v_1, v_2\}$ and for the other edges of P proceed inductively as follows. If the flow $h_{a,b}$ was assigned

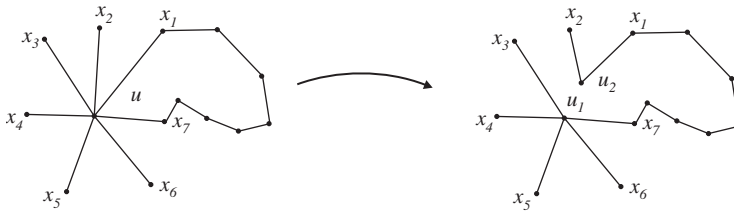


FIGURE 3. The operation $*$ if there is a path from x_1 to x_k that does not meet u .

to $G \boxtimes \{v_i, v_{i+1}\}$ and v_{i+1} is an active vertex then put the flow $h_{-a,b}$ on $G \boxtimes \{v_{i+1}, v_{i+2}\}$. Otherwise, if v_{i+1} is an inactive vertex, then put the flow $h_{a,-b}$ on $G \boxtimes \{v_{i+1}, v_{i+2}\}$. Then sum the values of different flows defined on a G -fiber. Clearly, the so-defined flow is non-zero on all edges of $G \boxtimes_w P$ and all flow values are between -2 and $+2$. The second part of the construction requires n steps, where n is the length of P . Since every step requires constant time, the total time complexity is $O(m_1 + m_2)$. ■

Next we define an operation $*$ on a graph G . Suppose that $u \in G$ is an active vertex of degree ≥ 3 . Then we split the vertex u into two vertices u_1 and u_2 , where u_1 is of degree $d(u) - 2$ and u_2 is of degree 2. Moreover let u_1 be active and u_2 inactive. Observe that if G^* is obtained from G by applying the operation $*$, then if $G^* \boxtimes_w H$ admits an NZ k -flow, then also $G \boxtimes_w H$.

For graphs G and H let $|E(G)| = m_1, |E(H)| = m_2, |V(H)| = n_2$, and let Δ_2 be the maximum degree of H . We use this notation in the following theorem.

Theorem 7. *Let G and H be connected nontrivial graphs, where G is not a tree. The construction of an NZ 3-flow in $G \boxtimes H$ requires at most $O(m_1 + m_2 n_2 \Delta_2)$ time.*

Proof. Suppose that G is not a tree and that H has a vertex of odd degree. Let all vertices of H be active. If H is a path, then the assertion of the theorem holds by Lemma 6. If H is not a path, we claim that we can repeatedly apply the operation $*$ to H such that the resulting graph is a disjoint union of paths with active endvertices.

Let $u \in V(H)$ be a vertex of odd degree. If $d(u) = 1$, we proceed with a nearest vertex v of u with $d(v) > 2$. Such a vertex exists, because H is not a path. Let x_1, \dots, x_k be the neighbors of u or v , respectively (see Figs. 3 and 4). For simplicity we renamed v to u . There are two cases.

Case 1. There is a path from x_i to an x_j that does not contain u for some $i \neq j$. In this case we can assume without loss of generality that there is a path from x_1 to $x_k, 2 < k$, that does not meet u . Then we apply $*$ by splitting u such that u_2 is adjacent to x_1 and x_2 and u_1 to the other neighbors of u . Note that the graph stays connected.

Case 2. Every path from x_i to x_j contains u for any pair $i \neq j$; compare Figure 4. We decompose by repeated application of $*$ as indicated in the figure. Note that if $d(u)$ is odd, then the vertex $u_{(k+1)/2}$ is of degree 1 and active. We call the connected components of the decomposed graph shown in Figure 4 as *bridge graphs*.

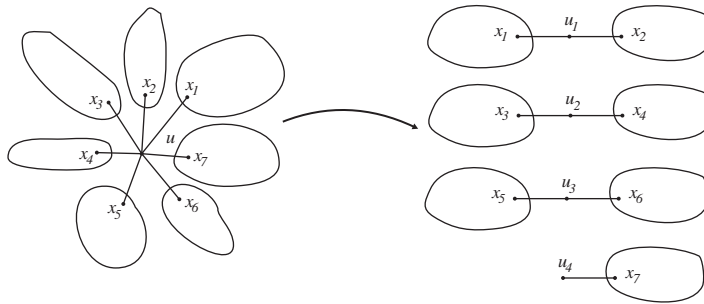


FIGURE 4. The operation $*$ in Case 2. The vertex u_4 is active.

In the first case we apply $*$ until the vertex of odd degree has degree 1 or until Case 2 occurs. Thus H will decompose into bridge graphs (if the second case does not occur, then, there will be only one bridge graph—the last one).

To each bridge graph we further apply $*$ if there are vertices of odd degree ≥ 3 . Finally, the resulting graph will decompose into a disjoint union of paths. This requires another application of the operation $*$. Note that the only active vertices in the decomposed graph are of degree 2. Since both endvertices of each one of these paths are active and since G is not a tree, we can define an NZ 3-flow on each generalized strong product $G \boxtimes_w P$ by Lemma 6. Hence, $G \boxtimes H$ admits an NZ 3-flow.

Now we discuss the time complexity of the above algorithm. When we fix an active vertex u of degree ≥ 3 , we need m_2 steps to find out if Case 1 or Case 2 applies. This is done by deleting the vertex u and finding the connected components of $G - v$. This needs to be repeated for the vertex u until its degree is reduced to one or two. So we need at most $\deg(u)m_2/2$ steps for the vertex u . To find the decomposition of G on paths with active endvertices, the operation $*$ must apply to all vertices of degree ≥ 3 . So all together $O(n_2m_2\Delta_2)$ steps are sufficient to obtain the decomposition (here Δ_2 is the maximum degree of H). According to Lemma 6 the time needed to construct an NZ 3-flow for the generalized strong product of G by a path is $O(m+n)$, where n is the length of the path. The sum of lengths of paths obtained by the decomposition is bounded by $n_2(\lceil \Delta_2/2 \rceil)$. So both parts of the construction run in $O(m_1 + m_2n_2\Delta_2)$ time. ■

B. Flows If Both Factors Are Trees

In this section we show that the strong product of two trees on at least three vertices admits an NZ 3-flow and that the strong product of a nontrivial tree by an edge does admit an NZ 4-flow, but not an NZ 3-flow.

Lemma 8. *Let T be a tree. Then the strong product $T \boxtimes uv$ admits a 3-flow h with the property that $h(z) = 0$ if and only if $z \in E(T_u)$.*

Proof. The proof is by induction on number of edges in T . If T has only one edge, then $T \boxtimes K_2$ is $K_2 \boxtimes K_2$ and f from Figure 1 is the desired flow. Now suppose that T is

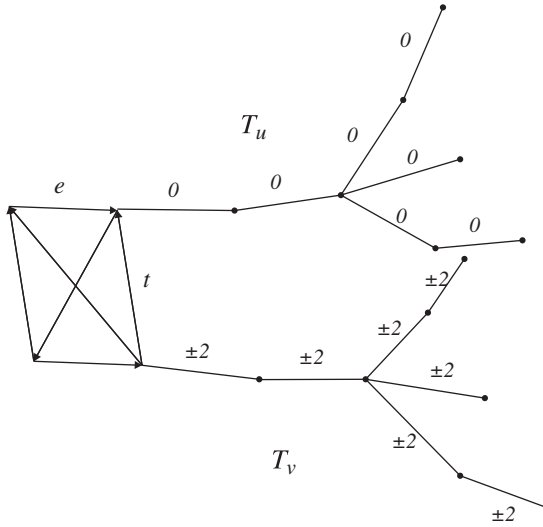


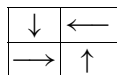
FIGURE 5. Situation from the proof of Lemma 8.

a tree with n edges. Let e be a leaf (pendant edge) of T (see Fig. 5). By the induction hypothesis $T - e$ admits a 3-flow h , such that $h(z) = 0$ if and only if $z \in E((T - e)_u)$. For the edge t from Figure 5 we thus infer $h(t) \in \{-1, -2, 1, 2\}$. Clearly, if we put the flow f or $-f$ on $e \boxtimes K_2$ such that the edge $e \in T_u$ receives the value 0 and the edge t either 1 or -1 (depending on $h(t)$), then we obtain the desired flow on $T \boxtimes K_2$. ■

Note that the flow from the above theorem requires m steps for its construction, where m is the number of edges of T (A BFS algorithm is applied to find an orientation of the tree so that the edges are directed toward the root).

Theorem 9. *Let G and H be trees on at least three vertices. Then there is an algorithm which gives an NZ 3-flow in $G \boxtimes H$ and runs in $m_1 + m_2$ steps.*

Proof. Suppose that G and H are trees with at least three vertices. Then both of them have P_3 as a subgraph. An NZ 3-flow on $P_3 \boxtimes P_3$ is depicted on the following scheme:



The square in this scheme represents the flow f , where the arrow shows the position of the edge with flow value 0. It is not hard to extend the argument to obtain an NZ 3-flow of $G \boxtimes P_3$ by successively amalgamating $K_2 \boxtimes P_3$ equipped with the flow of Lemma 8. This part requires m_1 steps (an orientation of the edges of G toward the root is needed).

Now the argument is completed by repeatedly amalgamating $G \boxtimes P_3$ with strong products $G \boxtimes K_2$ (note that H is a tree) that are endowed with a 3-flow of Lemma 8. This part requires additional m_2 steps. ■

Theorem 10. *An NZ 4-flow in $T \boxtimes K_2$ is constructed in $|E(T)|$ steps.*

Proof. Such a flow is easily found for $K_2 \boxtimes K_2$. We then extend it to an NZ 4-flow of $T \boxtimes K_2$ with a construction that is similar to the one indicated in Figure 2. As before we choose a root r and put the flow f from Figure 1 on $t \boxtimes K_2$ for every edge of T except one edge e that is incident to r ; see Figure 2. For the edge e we put an NZ 4-flow on $e \boxtimes K_2$. ■

REFERENCES

- [1] G. Fan, H. Lai, R. Xu, C.-Q. Zhang, and C. Zhou, Nowhere-zero 3-flows in triangularly connected graphs, *J Combin Theory Ser B* 98 (2008), 1325–1336.
- [2] J. Feigenbaum and A. A. Schäffer, Finding the prime factors of strong direct product graphs in polynomial time, *Discrete Math* 109 (1992), 77–102.
- [3] W. Imrich and S. Klavžar, *Product Graphs*, Wiley-Interscience, New York, NY, 2000.
- [4] W. Imrich and R. Škrekovski, A theorem on integer flows on Cartesian product of graphs, *J Graph Theory* 43 (2003), 93–98.
- [5] J. L. Shu and C. Q. Zhang, Nowhere-zero 3-flows in products of graphs, *J Graph Theory* 50 (2005), 79–89.
- [6] S. Špacapan, Connectivity of strong products of graphs, submitted.
- [7] W. T. Tutte, On the imbedding of linear graphs in surfaces, *Proc London Math Soc* 51 (1954), 474–483.
- [8] W. T. Tutte, A contribution to the theory of chromatic polynomials, *J Canad Math Soc* 6 (1954), 80–91.
- [9] C. Q. Zhang, *Integer Flows and Cycle Covers of Graphs*, Marcel Dekker, New York, 1997.
- [10] Z. Zhang, Y. Zheng, and A. Mamut, Nowhere-zero flows in Tensor products of graphs, *J Graph Theory*, DOI: 10.1002/jgt.20211.