Discussiones Mathematicae
Graph Theory 37 (2017) 975–988
doi:10.7151/dmgt.1982

A SHARP LOWER BOUND FOR THE GENERALIZED 3-EDGE-CONNECTIVITY OF STRONG PRODUCT GRAPHS

Yuefang Sun

Department of Mathematics
Shaoxing University
Zhejiang 312000, P.R. China

e-mail: yfsun2013@gmail.com

Abstract

The generalized k-connectivity $\kappa_k(G)$ of a graph G, mentioned by Hager in 1985, is a natural generalization of the path-version of the classical connectivity. As a natural counterpart of this concept, Li *et al.* in 2011 introduced the concept of generalized k-edge-connectivity which is defined as $\lambda_k(G) = \min\{\lambda_G(S) \mid S \subseteq V(G) \text{ and } |S| = k\}$, where $\lambda_G(S)$ denote the maximum number ℓ of pairwise edge-disjoint trees $T_1, T_2, \ldots, T_\ell$ in G such that $S \subseteq V(T_i)$ for $1 \leq i \leq \ell$. In this paper we get a sharp lower bound for the generalized 3-edge-connectivity of the strong product of any two connected graphs.

Keywords: generalized connectivity, generalized edge-connectivity, strong product.

2010 Mathematics Subject Classification: 05C05, 05C40, 05C76.

1. Introduction

We refer to book [1] for graph theoretical notation and terminology not described here. For a graph G, let $V(G)$, $E(G)$ be the set of vertices, the set of edges of G, respectively. For $X \subseteq V(G)$, we denote by $G \setminus X$ the subgraph obtained by deleting from G the vertices of X together with the edges incident with them. For a set S, we use $|S|$ to denote its size. We use P_n, C_m and K_ℓ to denote a path of order n, a cycle of order m and a complete graph of order ℓ, respectively.

Connectivity is one of the most basic concepts in graph theory, both in combinatorial sense and in algorithmic sense. The connectivity of G, written

$\kappa(G)$, is the minimum size of a vertex set $X \subseteq V(G)$ such that $G \setminus X$ is disconnected or has only one vertex. This definition is called the *cut-version* definition of the connectivity. A well-known theorem of Menger provides an equivalent definition, which can be called the *path-version* definition of the connectivity. For any two distinct vertices x and y in G, the local connectivity $\kappa_G(x, y)$ is the maximum number of internally disjoint paths connecting x and y. Then $\kappa(G) = \min\{\kappa_G(x, y) | x, y \in V(G), x \neq y\}$ is defined to be the connectivity of G.

Although there are many elegant and powerful results on connectivity in graph theory, the basic notation of classical connectivity may not be general enough to capture some computational settings and so people tried to generalize this concept.

The cut-version definition of the connectivity does not concern the number of components of $G \setminus X$. Two graphs with the same connectivity may have different degrees of vulnerability in the sense that the deletion of a vertex cut-set of minimum cardinality from one graph may produce a graph with considerably more components than in the case of the other graph. For example, the star $K_{1,n-1}$ and the path P_n $(n \geq 3)$ are both trees of order n and therefore have connectivity 1, but the deletion of a cut-vertex from $K_{1,n-1}$ produces a graph with $n-1$ components while the deletion of a cut-vertex from P_n produces only two components. Chartrand *et al.* [3] generalized the cut-version definition of the connectivity as follows: For an integer $k \geq 2$ and a graph G of order $n \geq k$, the *k-connectivity* $\kappa'_k(G)$ is the smallest number of vertices whose removal from G produces a graph with at least k components or a graph with fewer than k vertices. By definition, we clearly have $\kappa'_2(G) = \kappa(G)$. Thus, the concept of k-connectivity could be seen as a generalization of the classical connectivity. For more details about this topic, we refer to [3, 5, 24, 25, 33, 34].

The generalized k-connectivity $\kappa_k(G)$ of a graph G which was mentioned by Hager [9] in 1985 is a natural generalization of the path-version definition of the connectivity. For a graph $G = (V, E)$ and a set $S \subseteq V$ of at least two vertices, an *S-Steiner tree* or a *Steiner tree connecting S* (or simply, an *S-tree*) is such a subgraph T of G that is a tree with $S \subseteq V(T)$. Two S-trees T_1 and T_2 are said to be *internally disjoint* if $E(T_1) \cap E(T_2) = \emptyset$ and $V(T_1) \cap V(T_2) = S$. The *generalized local connectivity* $\kappa_G(S)$ is the maximum number of internally disjoint S-trees in G. For an integer k with $2 \leq k \leq n$, the *generalized k-connectivity* is defined as

$$\kappa_k(G) = \min\{\kappa_G(S) | S \subseteq V(G), |S| = k\}.$$

Thus, $\kappa_k(G)$ is the minimum value of $\kappa_G(S)$ when S runs over all the k-subsets of $V(G)$. By definition, we clearly have $\kappa_2(G) = \kappa(G)$, which is the reason why one addresses $\kappa_k(G)$ as the generalized connectivity of G. By convention, for a connected graph G with less than k vertices, we set $\kappa_k(G) = 1$, and $\kappa_k(G) = 0$ when G is disconnected.

As a natural counterpart of the generalized k-connectivity, recently Li *et al.* [20] introduced the following concept of generalized edge-connectivity. Two S-trees T_1 and T_2 are said to be *edge-disjoint* if $E(T_1) \cap E(T_2) = \emptyset$. The *generalized local edge-connectivity* $\lambda_G(S)$ is the maximum number of edge-disjoint S-trees in G. For an integer k with $2 \leq k \leq n$, the *generalized k-edge-connectivity* is defined as

$$\lambda_k(G) = \min\{\lambda_G(S) | S \subseteq V(G), |S| = k\}.$$

Thus, $\lambda_k(G)$ is the minimum value of $\lambda_G(S)$ when S runs over all the k-subsets of $V(G)$. By definition, we clearly have $\lambda_2(G) = \lambda(G)$.

The generalized k-connectivity and generalized k-edge-connectivity are also called *tree connectivities* in the literature. There are many results on this type of generalized connectivity, see [9, 14, 15, 17–22, 28–32, 34, 35].

In addition to being a natural combinatorial measure, the tree connectivity can be motivated by its interesting interpretation in practice. For example, suppose that G represents a network. If one considers to connect a pair of vertices of G, then a path is used to connect them. However, if one wants to connect a set S of vertices of G with $|S| \geq 3$, then a tree has to be used to connect them. This kind of tree with minimum order for connecting a set of vertices is usually called a Steiner tree, and popularly used in the physical design of VLSI (see [7, 8, 27]) and computer communication networks (see [6]). Usually, one wants to consider how tough a network can be, for the connection of a set of vertices. Then, the number of totally independent ways to connect them is a measure for this purpose. The generalized k-connectivity can serve for measuring the capability of a network G to connect any k vertices in G. For the topic of generalized connectivities and their applications, the reader is referred to a recent survey [16].

Products of graphs occur naturally in discrete mathematics as tools in combinatorial constructions; they give rise to important classes of graphs and deep structural problems. Many researchers have investigated the topic of graph products in the past several decades, such as [2, 4, 10–13, 23, 26, 36, 37, 38].

The *Cartesian product* of two graphs G and H, denoted by $G \Box H$, is defined to have the vertex set $V(G) \times V(H)$ such that (u, u') and (v, v') are adjacent if and only if either $u = v$ and $u'v' \in E(H)$, or $u' = v'$ and $uv \in E(G)$. The *strong product* of G and H is the graph $G \boxtimes H$ whose vertex set is $V(G) \times V(H)$ and whose edge set is the set of all pairs $(u, u')(v, v')$ such that either $u = v$ and $u'v' \in E(H)$, or $u' = v'$ and $uv \in E(G)$, or $uv \in E(G)$ and $u'v' \in E(H)$. Clearly, both of these two products are commutative, that is, $G \Box H = H \Box G$ and $G \boxtimes H = H \boxtimes G$. By definition, we also know that the graph $G \Box H$ is a spanning subgraph of the graph $G \boxtimes H$ for any two graphs G and H. The *lexicographic product* of two graphs G and H, written as $G \circ H$, is defined as follows: $V(G \circ H) = V(G) \times V(H)$, and two distinct vertices (u, v) and (u', v') of $G \circ H$ are adjacent if and only if either $(u, u') \in E(G)$ or $u = u'$ and $(v, v') \in E(H)$.

For the Cartesian product graphs, the exact formula of $\kappa(G\Box H)$ was obtained.

Theorem 1 [23, 36]**.** *Let G and H be graphs on at least two vertices. Then*

$$\kappa(G\Box H) = \min\{\kappa(G)|H|, \kappa(H)|G|, \delta(G)+\delta(H)\}.$$

This theorem was first stated by Liouville [23]. However, the proof never appeared. In the meantime, several partial results were obtained until Špacapan [36] provided the proof. Theorem 1 in particular implies the following result of Sabidussi [26].

Theorem 2 [26]**.** *Let G and H be connected graphs. Then $\kappa(G\Box H) \geq \kappa(G) + \kappa(H)$.*

Li, Li and Sun [15] investigated the generalized 3-connectivity of the Cartesian product graphs and obtain the following result which can be seen as an extension of Theorem 2.

Theorem 3 [15]**.** *Let G and H be connected graphs such that $\kappa_3(G) \geq \kappa_3(H)$.*

(a) *If $\kappa_3(G) < \kappa(G)$, then $\kappa_3(G\Box H) \geq \kappa_3(G) + \kappa_3(H)$. Moreover, the bound is sharp.*

(b) *If $\kappa_3(G) = \kappa(G)$, then $\kappa_3(G\Box H) \geq \kappa_3(G) + \kappa_3(H) - 1$. Moreover, the bound is sharp.*

Li and Mao derived a sharp upper bound for $\kappa_3(G\Box H)$.

Theorem 4 [18]**.** *Let G and H be two connected graphs. Then $\kappa_3(G\Box H) \leq \min\big\{\lfloor\frac{4}{3}\kappa_3(G) + r_1 - \frac{4}{3}\lceil\frac{r_1}{2}\rceil\rfloor|V(H)|, \lfloor\frac{4}{3}\kappa_3(H) + r_2 - \frac{4}{3}\lceil\frac{r_2}{2}\rceil\rfloor|V(G)|, \delta(G)+\delta(H)\big\}$, where $r_1 \equiv \kappa(G) \pmod 4$ and $r_2 \equiv \kappa(H) \pmod 4$. Moreover, the bound is sharp.*

In [29], we obtained a sharp lower bound for the generalized 3-edge-connectivity of Cartesian product graph.

Theorem 5 [29]**.** *If G and H are connected graphs, then $\lambda_3(G\Box H) \geq \lambda_3(G) + \lambda_3(H)$. Moreover, the bound is sharp.*

The following result concerns a sharp upper bound for the generalized 3-edge-connectivity of Cartesian product graph.

Theorem 6 [32]**.** *Let G and H be two graphs with at least two vertices. Then $\lambda_3(G\Box H) \leq \min\left\{\lfloor\frac{4}{3}\lambda_3(G) + r_1 - \frac{4}{3}\lceil\frac{r_1}{2}\rceil\rfloor|V(H)|, \lfloor\frac{4}{3}\lambda_3(H) + r_2 - \frac{4}{3}\lceil\frac{r_2}{2}\rceil\rfloor|V(G)|, \delta(G)+\delta(H)\right\}$, where $r_1 \equiv \lambda(G) \pmod 4$ and $r_2 \equiv \lambda(H) \pmod 4$. Moreover, the bound is sharp.*

Note that the minimum in Theorem 6 can be realized by any of three terms. For the lexicographic product graphs, Li and Mao obtained the following bounds for $\kappa_3(G \circ H)$.

Theorem 7 [18]**.** *Let G and H be two connected graphs. If G is non-trivial and non-complete, then $\kappa_3(G \circ H) \leq \lfloor \frac{4}{3}\kappa_3(G) + r - \frac{4}{3}\lceil \frac{r}{2} \rceil \rfloor |V(H)|$, where $r \equiv \kappa(G) \pmod 4$. Moreover, the bound is sharp.*

Theorem 8 [18]**.** *Let G and H be two connected graphs. Then $\kappa_3(G \circ H) \geq \kappa_3(G)|V(H)|$. Moreover, the bound is sharp.*

Li, Yue, and Zhao studied $\lambda_3(G \circ H)$ and provided both sharp lower and upper bounds.

Theorem 9 [22]**.** *Let G and H be two non-trivial graphs such that G is connected. Then $\lambda_3(H) + \lambda_3(G)|V(H)| \leq \lambda_3(G \circ H) \leq \min\left\{ \left\lfloor \frac{4\lambda_3(G)+2}{3} \right\rfloor |V(H)|^2, \delta(H) + \delta(G)|V(H)| \right\}$. Moreover, both bounds are sharp.*

For the strong product graphs, we obtained the following upper bound for $\lambda(G \boxtimes H)$.

Theorem 10 [32]**.** *Let G and H be two connected graphs. Then $\lambda(G \boxtimes H) \leq \min\{ \lfloor \frac{4}{3}\lambda_3(G) + r_1 - \frac{4}{3}\lceil \frac{r_1}{2} \rceil \rfloor (|V(H)| + 2|E(H)|), \lfloor \frac{4}{3}\lambda_3(H) + r_2 - \frac{4}{3}\lceil \frac{r_2}{2} \rceil \rfloor (|V(G)| + 2|E(G)|), \delta(G) + \delta(H) + \delta(G)\delta(H)\}$. Moreover, the bound is sharp.*

Note that the minimum in Theorem 10 can be realized by any of three terms.

In this paper, we continue the research on tree connectivities of the product graphs and get a sharp lower bound for the generalized 3-edge-connectivity of the strong product graph (Theorem 16). The proof of Theorem 16 consists of Lemmas 13, 14 and 15. In order to prove these lemmas we need a few preliminary results which will be given in the next section.

2. Preliminaries

We need the following useful notion which was used in [11, 12]. The mappings $p_G : (u, v) \mapsto u$ and $p_H : (u, v) \mapsto v$ from $V(G \square H)$ into $V(G)$, respectively $V(H)$ are weak homomorphisms from $G \square H$ onto the factors G, respectively H. They are called *projections* in the literatures.

Let G and H be two graphs with $V(G) = \{u_i | 1 \leq i \leq n\}$ and $V(H) = \{v_j | 1 \leq j \leq m\}$. We use $G(v_j)$ to denote the subgraph of $G \square H$ induced by the vertex set $\{(u_i, v_j) | 1 \leq i \leq n\}$ where $1 \leq j \leq m$, and use $H(u_i)$ to denote the subgraph of $G \square H$ induced by the vertex set $\{(u_i, v_j) | 1 \leq j \leq m\}$ where

$1 \le i \le n$. Clearly, we have $G(v_j) \cong G$ and $H(u_i) \cong H$. For example, as shown in Figure 1, $G(v_j) \cong G$ for $1 \le j \le 4$ and $H(u_i) \cong H$ for $1 \le i \le 3$. For $1 \le j_1 \ne j_2 \le m$, (u_i, v_{j_1}) and (u_i, v_{j_2}) belong to the same graph $H(u_i)$ where $u_i \in V(G)$, and we call (u_i, v_{j_2}) the *vertex corresponding to* (u_i, v_{j_1}) in $G(v_{j_2})$; for $1 \le i_1 \ne i_2 \le n$, we call (u_{i_2}, v_j) the vertex corresponding to (u_{i_1}, v_j) in $H(u_{i_2})$ [15]. Similarly, we can define the path and the tree corresponding to some path and tree, respectively. For example, in the graph (c) of Figure 1, let P_1, respectively P_2, be the paths whose edges are labelled 1, respectively 2, in $H(u_1)$, respectively $H(u_2)$. Then P_2 is called the path corresponding to P_1 in $H(u_2)$. Clearly, P_1 and P_2 correspond to the path v_1, v_2, v_3, v_4 in H.

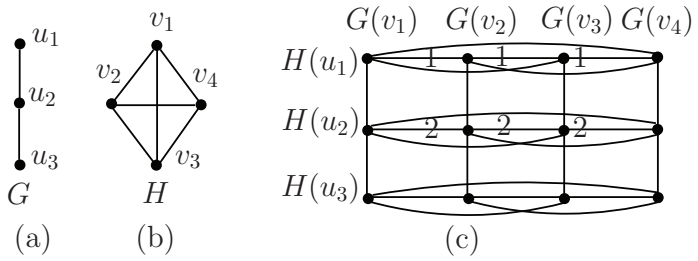

Figure 1. Graphs G, H and their Cartesian product.

Lemma 11 [21]**.** *Let G be a connected graph of order n. If there exist two adjacent vertices of degree $\delta(G)$, then $\lambda_k(G) \le \delta(G) - 1$ for every integer k with $3 \le k \le n$, and moreover the bound is sharp.*

For simplicity, we set $u_{i,j} = (u_i, v_j)$ in the sequel. The following result concerns the strong product of P_2 and P_n where $n \ge 3$.

Lemma 12. *Let $G = P_2 \boxtimes P_n$ with $V(G) = \{u_{i,j}\}$, where $1 \le i \le 2$ and $1 \le j \le n$. If $x = u_{1,1}, y = u_{1,j_0}, z = u_{1,n}$ for some $1 \le j_0 \le n$, then there are two edge-disjoint trees connecting S in G, where $S = \{x, y, z\}$.*

Proof. We get the first tree T_1 by letting $V(T_1) = V(G) \setminus \{u_{2,n}\}$ and $E(T_1) = \{u_{1,j}u_{2,j}, u_{2,j}u_{1,j+1} | 1 \le j \le n-1\}$, then we construct the second tree T_2 by letting $V(T_2) = V(G) \setminus \{u_{2,1}\}$ and $E(T_2) = \{u_{2,j}u_{2,j+1} | 2 \le j \le n-1\} \cup \{u_{1,1}u_{2,2}\} \cup \{u_{1,j_0}u_{2,j_0+1}\} \cup \{u_{1,n}u_{2,n}\} \cup \{u_{1,j}u_{2,j+1} | 1 \le j \le n-1\}$. Clearly, T_1 and T_2 are two edge-disjoint S-trees. ■

Note that in the proof of Lemma 12, we have $E(T_i) \cap E(P_n(u_1)) = \emptyset$ for $i \in \{1, 2\}$. We need the following definition. Let $G = P_m \boxtimes P_n$ with $n, m \ge 2$ and $C := u_{1,1}, u_{1,2}, \ldots, u_{1,n}, u_{2,n}, \ldots, u_{m,n}, u_{m,n-1}, \ldots, u_{m,1}, u_{m-1,1}, \ldots, u_{1,1}$ be a cycle of G; we call an edge of C an *outer edge* of G and an edge of $E(G) \setminus E(C)$

an *inner edge* of G. A path is called an *inner path* if all edges of it are inner edges. Let $x = u_{1,1}, y = u_{m,n}$; there is an inner $x - y$ path. We take $P_3 \boxtimes P_4$ for an example as shown in Figure 2, edges labelled with 1 are outer edges of G. Each edge of the $x - y$ path labelled 2 is an inner edge.

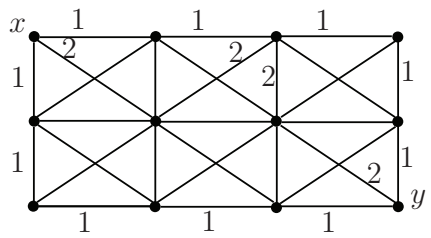

Figure 2. $P_3 \boxtimes P_4$.

Note that in the sequel, for a set $S = \{x, y, z\}$, we assume that every S-tree T is *minimal*, that is, the subgraph which is obtained by deleting any set of vertices or edges of T will not be an S-tree. We know T must be one of the following two types: (a) T is a path whose two end vertices belong to S; (b) T is a tree with exactly three end vertices and these end vertices are x, y, z. In the second type, there is exactly one vertex u with $deg_T(u) = 3$. The above assumption will not affect our results.

3. Main Results

The following fact which will be useful is clear: If H is a spanning subgraph of G, then $\lambda_3(G) \geq \lambda_3(H)$. Let $S = \{x, y, z\}$, where $x \in V(G(v_\alpha)), y \in V(G(v_\beta)), z \in V(G(v_\gamma))$ for some $1 \leq \alpha, \beta, \gamma \leq m$. Without loss of generality, we assume that $\lambda_3(G) = k \geq \ell = \lambda_3(H)$. Thus, $\min\{2\lambda_3(G) + \lambda_3(H), \lambda_3(G) + 2\lambda_3(H)\} = \lambda_3(G) + 2\lambda_3(H) = k + 2\ell$. In order to prove our main result, we need the following three lemmas.

Lemma 13. *If α, β, γ are distinct, then there are at least $k + 2\ell$ edge-disjoint S-trees.*

Proof. Without loss of generality, we assume that $x = u_{1,1}, y \in V(G(v_2)), z \in V(G(v_3))$. Furthermore, let y', z' be the vertices corresponding to y, z in $G(v_1)$, x', z'' be the vertices corresponding to x, z in $G(v_2)$ and x'', y'' be the vertices corresponding to x, y in $G(v_3)$.

Case 1. $p_G(x) = p_G(y) = p_G(z)$. Now we have that x, y', z' are the same vertex in $G(v_1)$. Let x_1 be a neighbor of x in $G(v_1)$. Without loss of generality, we can assume that $x_1 \in H(u_2)$. Let x_1' and x_1'' be the corresponding vertices of

x_1 in $G(v_2)$ and $G(v_3)$, respectively. Clearly, $yx_1' \in E(G(v_2)), zx_1'' \in E(G(v_3))$. Let T_1' be a $\{x_1, x_1', x_1''\}$-tree in $H(u_2)$. If T_1' is a path, then by Lemma 12, we can find two edge-disjoint S-trees.

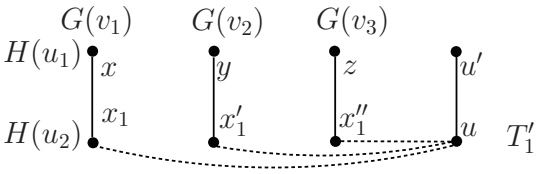

Figure 3. The graph of Case 1.

In the following of this case, we assume that T_1' is a tree of type (b) introduced in the last paragraph of Section 2. Let u be the vertex of degree 3 in T_1' and u' be the vertex corresponding to it in $H(u_1)$ as shown in Figure 3. The unique $x_1 - x_1'$ path in T_1' is a path of order at least 3, redhence we can construct a $\{x, y, u'\}$-tree $T_{x,y}$ which is similar to T_1 in the proof of Lemma 12. Similarly, the unique $x_1 - x_1''$ path in T_1' is a path of order at least 3, and we can construct a $\{x, z, u'\}$-tree $T_{x,z}$ similarly to T_1 in the proof of Lemma 12. Now we can get an S-tree from $T_{x,y}$ and $T_{x,z}$. Similarly, we can get a $\{x, y, u'\}$-tree $T_{x,y}'$ and a $\{x, z, u'\}$-tree $T_{x,z}'$. Thus, another S-tree can be constructed from $T_{x,y}'$ and $T_{x,z}'$.

Since x has at least k neighbors in $G(v_1)$, we can get $2k$ such trees. Then by adding the ℓ edge-disjoint S-trees in $H(u_1)$, we can find at least $2k + \ell \geq k + 2\ell$ S-trees. It is not hard to show that any two of these trees are edge-disjoint by the definition of the strong product.

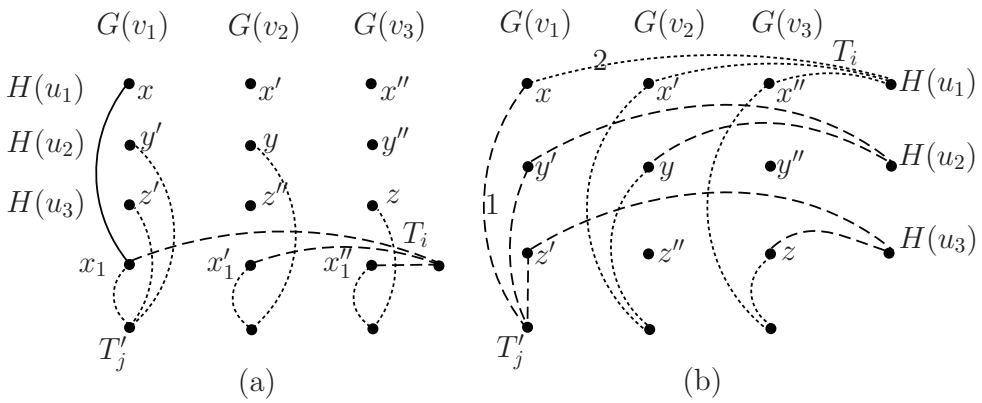

Figure 4. Graphs in the proof of Claim 1.

Case 2. $p_G(x), p_G(y), p_G(z)$ are three distinct vertices. In this case we have that x, y', z' are three distinct vertices in $G(v_1)$. Without loss of generality, we can assume that $y' \in V(H(u_2)), z' \in V(H(u_3))$. As $\lambda_3(G(v_1)) = k$, there are k edge-

disjoint $\{x, y', z'\}$-trees in $G(v_1)$, say T'_j, where $1 \le j \le k$. Let $\{T_i | 1 \le i \le \ell\}$ be a set of ℓ edge-disjoint $\{v_1, v_2, v_3\}$-trees in H since $\lambda_3(H) = \ell$. Let $k_0, k_1, \ldots, k_\ell$ be integers such that $0 = k_0 < k_1 < \cdots < k_\ell = k$ since $k \ge \ell$.

Subcase 2.1. $xy', xz' \notin E(G(v_1))$. We need the following claim.

Claim 1. *If $xy', xz' \notin E(G(v_1))$, then there are $k_i - k_{i-1} + 2$ edge-disjoint S-trees in $\left(\bigcup_{j=k_{i-1}+1}^{k_i} T'_j\right) \boxtimes T_i$ for each $1 \le i \le \ell$.*

Proof. Let $1 \le i \le \ell$. For the case $k_{i-1} + 1 \le j \le k_i - 1$, we can construct an S-tree in $T'_j \boxtimes T_i$ as shown in the graph (a) of Figure 4. Here $x_1 \in V(T'_j)$ is a neighbor of x in $G(v_1)$, while x'_1 and x''_1 are vertices corresponding to x_1 in $G(v_2)$ and $G(v_3)$, respectively. For simplicity, we also use T_i and T'_j to denote the $\{x_1, x'_1, x''_1\}$-tree and $\{x, y', z'\}$-tree, respectively, as shown in the graph (a) of Figure 4.

For the case $j = k_i$, in $T'_{k_i} \boxtimes T_i$ we first construct two S-trees as shown in the graph (b) of Figure 4. In $H(u_1)$, let $P_{xx'}$ denote the $x - x'$ path which belongs to T_i; in $G(v_1)$, let $P_{xy'}$ denote the $x - y'$ path which belongs to T'_{k_i}. Then there is an inner $x - y$ path, denoted by P_{xy}, in the subgraph $P_{xx'} \boxtimes P_{xy'}$ of $T'_{k_i} \boxtimes T_i$. Similarly, we can find an inner $y - z$ path P_{yz}. By combining P_{xy} and P_{yz}, we can obtain the third S-tree in $T'_{k_i} \boxtimes T_i$. Hence, for each $1 \le i \le \ell$, in $\left(\bigcup_{j=k_{i-1}+1}^{k_i} T'_j\right) \boxtimes T_i$, we can get $k_i - k_{i-1} + 2$ trees in total and it is not hard to show that any two of these trees are edge-disjoint by the definition of the strong product. $\square$

Now for the case that $xy', xz' \notin E(G(v_1))$, we can construct $\sum_{i=1}^{\ell}(k_i - k_{i-1} + 2) = k + 2\ell$ trees in total by Claim 1 and any two of these trees are edge-disjoint.

Subcase 2.2. $xy', xz' \in E(G(v_1))$. If both xy' and xz' belong to the same tree, say $\overline{T} \in \left\{T'_j | 1 \le j \le k\right\}$, then we reorder these trees such that $T'_{k_1} = \overline{T}$. With a similar argument to that of Subcase 2.1, we can construct $k + 2\ell$ edge-disjoint S-trees.

Thus, in the following we assume that xy' and xz' belong to distinct trees, say $\overline{T}$ and $\widetilde{T}$, respectively. Both of them are paths by the assumption of the note in the end of Section 2, that is, $\overline{T} = xy' \cup Q$ and $\widetilde{T} = xz' \cup Q'$ where Q is either an $x - z'$ path or a $y' - z'$ path and Q' is either an $x - y'$ path or a $z' - y'$ path. In all cases we can set two new paths $\overline{T'}$ and $\widetilde{T'}$ where $\overline{T'} = y'xz'$ and $\widetilde{T'}$ is obtained from $Q \cup Q'$. Again we get a Subcase 2.1 and we are done.

The remaining subcase is that exactly one of xy', xz' belongs to $E(G(v_1))$. Without loss of generality, we can assume that $xy' \in E(G(v_1))$.

Subcase 2.3. $xy' \in E(G(v_1))$. Let xy' belong to a tree of $\left\{T'_j | 1 \le j \le k\right\}$, without loss of generality, we assume that $xy' \in T'_{k_1}$. Hence, we can construct $k + 2\ell$ edge-disjoint S-trees with a similar argument to that of Subcase 2.1.

Case 3. Two of $p_G(x), p_G(y), p_G(z)$ are the same vertex. Now we have that two of x, y', z' are the same vertex in $G(v_1)$. Without loss of generality, we can assume that $y' = z'$. Since $\lambda(G(v_1)) \geq \lambda_3(G(v_1)) = \lambda_3(G) = k$, there exist k edge-disjoint $x - y'$ paths P_i in $G(v_1)$ where $1 \leq i \leq k$. With a similar argument to that of Case 2, we get at least $k + 2\ell$ edge-disjoint S-trees. ■

Lemma 14. *If exactly two of α, β, γ are the same, then there are at least $k + 2\ell$ edge-disjoint S-trees.*

Proof. Without loss of generality, we can assume that $\alpha = \beta = 1$ and $\gamma = 2$. Furthermore, we assume that $x = u_{1,1}$ and $y = u_{2,1}$. For the case that $z = u_{\tau,2}$ where $3 \leq \tau \leq n$, we can construct $k + 2\ell$ edge-disjoint S-trees since it is similar to Case 3 of Lemma 13 by the commutativity of the strong product. It suffices to consider the case $\tau = 1$ since the case that $\tau = 2$ is similar.

As $\lambda(G) \geq \lambda_3(G) = k$ and $\lambda(H) \geq \lambda_3(H) = \ell$, there are at least k edge-disjoint $x - y$ paths $P_i : x = a_{i,0}, a_{i,1}, \ldots, a_{i,t_i} = y$ in $G(v_1)$ and ℓ edge-disjoint $x - z$ paths $Q_j : x = b_{j,0}, b_{j,1}, \ldots, b_{j,t_j} = z$ in $H(u_1)$ where $1 \leq i \leq k, 1 \leq j \leq \ell$. We set $P_k := x, y$ if $xy \in E(G(v_1))$ and $Q_\ell := x, z$ if $xz \in E(H(u_1))$. Note that $a_{i_1,1} \neq a_{i_2,1}$ for $1 \leq i_1 \neq i_2 \leq k$ and $b_{j_1,1} \neq b_{j_2,1}$ for $1 \leq j_1 \neq j_2 \leq k$.

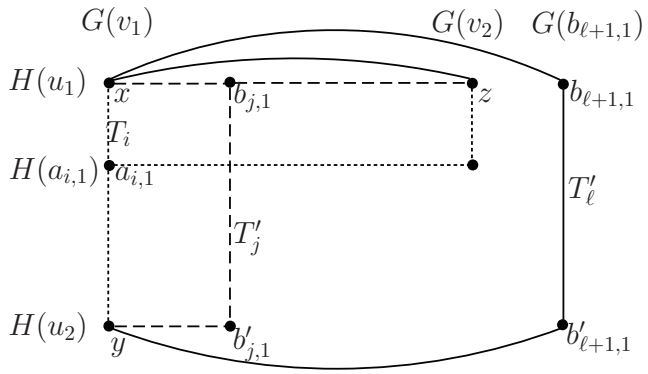

Figure 5. Trees in the graph.

For $1 \leq i \leq k - 1$, we can construct a tree $T_i := P_i \cup Q_1^i \cup \{a'_{i,1}, z\}$, where $a'_{i,1}$ is the vertex corresponding to $a_{i,1}$ in $G(v_2)$ and Q_1^i is the $a_{i,1} - a'_{i,1}$ path corresponding to Q_1 in $H(a_{i,1})$. See the lines labelled by T_i in Figure 5, and here $H(a_{i,1}) \cong H$ is a subgraph which contains $a_{i,1}$. Similarly, for $1 \leq j \leq \ell - 1$, we can construct a tree $T'_j := Q_j \cup P_1^j \cup \{b'_{j,1}, y\}$, where $b'_{j,1}$ is the vertex corresponding to $b_{j,1}$ in $H(u_2)$ and P_1^j is the $b_{j,1} - b'_{j,1}$ path corresponding to P_1 in $G(b_{j,1})$. Note that here $G(b_{j,1}) \cong G$ is a subgraph which contains $b_{j,1}$ as shown in Figure 5. We further assume that $xy \notin E(G(v_1))$ since the argument of the case that $xy \in E(G(v_1))$ is similar. We can construct a tree T_k from P_k which is similar

to T_i for $1 \leq i \leq k-1$. So far, $k+\ell-1$ S-trees have been constructed: $\left\{T_i, T'_j | 1 \leq i \leq k, 1 \leq j \leq \ell-1\right\}$. Thus, we need to construct the remaining $\ell+1$ trees by considering the following two cases.

Case 1. $xz \notin E(H(u_1))$. In this case, we first get a tree T'_ℓ from Q_ℓ with a similar construction method to that of $T'_j (1 \leq j \leq \ell-1)$ in the above argument. For each $1 \leq i \leq \ell$, we can use P_i and Q_i to construct a tree $T_{i,i}$ as follows: Let P_{xy} be an inner $x-y$ path in the subgraph $P_i \boxtimes Q_{xb_{j,1}}$ and P_{xz} be an inner $x-z$ path in the subgraph $Q_i \boxtimes P_{xa_{i,1}}$, where $Q_{xb_{j,1}}$ is the edge $xb_{j,1}$ and $Q_{xa_{i,1}}$ is the edge $xa_{i,1}$; then by combining P_{xy} and P_{xz}, we can get an S-tree.

For example, we consider two graphs $G \cong C_4, H \cong P_4$. Let $S = \{x, y, z\}$ where $x = u_{1,1}, y = u_{4,1}, z = u_{1,4}$. By the above method, we can find three edge-disjoint S-trees in $C_4 \boxtimes P_4$ as shown in Figure 6.

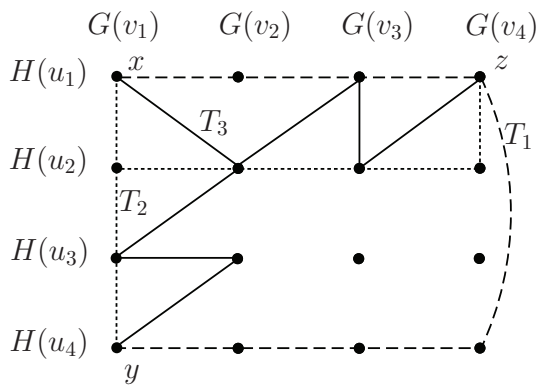

Figure 6. Three edge-disjoint S-trees in $C_4 \boxtimes P_4$.

Case 2. $xz \in E(H(u_1))$. If $deg_{H(u_1)}(x) = deg_{H(u_1)}(z) = \delta(H(u_1))$, then we have $\ell = \lambda_3(H(u_1)) \leq \delta(H(u_1)) - 1$ by Lemma 11, it means that x has a neighbor, say $b_{\ell+1,1}$, which is distinct from $b_{j,1}$ in $H(u_1)$, where $1 \leq j \leq \ell$. We can construct a tree $T'_\ell := \{xz\} \cup \{xb_{\ell+1,1}\} \cup P_1^{\ell+1} \cup \{yb'_{\ell+1,1}\}$, where $b'_{\ell+1,1}$ is the vertex corresponding to $b_{\ell+1,1}$ in $H(u_2)$ and $P_1^{\ell+1}$ is the path corresponding to P_1 in $G(b_{\ell+1,1})$. Note that here $G(b_{\ell+1,1}) \cong G$ is a subgraph of $G \boxtimes H$ which contains $b_{\ell+1,1}$ as shown in Figure 5.

If $deg_{H(u_1)}(x) > \delta(H(u_1))$ or $deg_{H(u_1)}(z) > \delta(H(u_1))$, then we can also get a tree T'_ℓ with a similar argument. The remaining ℓ trees can be found with a similar argument to that of Case 1.

Thus, there are $k+2\ell$ trees in total and it is not hard to show that any two of them are edge-disjoint by the definition of the strong product. ■

In the final case that α, β, γ are the same we can get at least $k+2\ell$ edge-disjoint trees by Lemma 13 and by the commutativity of the strong product.

Lemma 15. *If α, β, γ are the same, then there are at least $k + 2\ell$ edge-disjoint trees connecting S.*

According the above three lemmas, we can get our main result.

Theorem 16. *If G and H are two connected graphs, then $\lambda_3(G \boxtimes H) \geq \min\{2\lambda_3(G) + \lambda_3(H), \lambda_3(G) + 2\lambda_3(H)\}$. Moreover, the bound is sharp.*

Proof. By Lemmas 13, 14 and 15, we have $\lambda_3(G \boxtimes H) \geq k + 2\ell = \min\{2\lambda_3(G) + \lambda_3(H), \lambda_3(G) + 2\lambda_3(H)\}$. Thus, for any pair of two trees T_1 and T_2 with orders at least 3, $\lambda_3(T_1 \boxtimes T_2) \geq 3 = k + 2\ell$ since $\lambda_3(T_1) = \lambda_3(T_2) = 1$. Since we also have $\lambda_3(T_1 \boxtimes T_2) \leq 3 = \delta(T_1 \boxtimes T_2)$, the bound is sharp. ■

Acknowledgements

This research was supported by National Natural Science Foundation of China (No. 11401389) and China Scholarship Council (No. 201608330111). We would like to thank two anonymous referees for helpful comments and suggestions which indeed help us greatly to improve the quality of our paper.

References

[1] J.A. Bondy and U.S.R. Murty, Graph Theory, GTM 244 (Springer, Berlin, 2008).

[2] B. Brešar and S. Špacapan, *Edge-connectivity of strong products of graphs*, Discuss. Math. Graph Theory **27** (2007) 333–343.
doi:10.7151/dmgt.1365

[3] G. Chartrand, S.F. Kappor, L. Lesniak and D.R. Lick, *Generalized connectivity in graphs*, Bull. Bombay Math. Colloq. **2** (1984) 1–6.

[4] W.S. Chiue and B.S. Shieh, *On connectivity of the Cartesian product of two graphs*, Appl. Math. Comput. **102** (1999) 129–137.
doi:10.1016/S0096-3003(98)10041-3

[5] D.P. Day, O.R. Oellermann and H.C. Swart, *The ℓ-connectivity function of trees and complete multipartite graphs*, J. Combin. Math. Combin. Comput. **10** (1991) 183–192.

[6] D. Du and X. Hu, Steiner Tree Problems in Computer Communication Networks (World Scientific, 2008).
doi:10.1142/6729

[7] M. Grötschel, *The Steiner tree packing problem in VLSI design*, Math. Program. **78** (1997) 265–281.
doi:10.1007/BF02614374

[8] M. Grötschel, A. Martin and R. Weismantel, *Packing Steiner trees: A cutting plane algorithm and commputational results*, Math. Program. **72** (1996) 125–145.
doi:10.1007/BF02592086

[9] M. Hager, *Pendant tree-connectivity*, J. Combin. Theory Ser. B **38** (1985) 179–189. doi:10.1016/0095-8956(85)90083-8

[10] R. Hammack, W. Imrich and S. Klavžar, Handbook of Product Graphs, Second Edition (CRC Press, 2011).

[11] W. Imrich and S. Klavžar, Product Graphs—Structure and Recognition (Wiley, New York, 2000).

[12] W. Imrich, S. Klavžar and D.F. Rall, Topics in Graph Theory: Graphs and Their Cartesian Product (A K Peters, 2008).

[13] S. Klavžar and S. Špacapan, *On the edge-connectivity of Cartesian product graphs*, Asian-Eur. J. Math. **1** (2008) 93–98.

[14] H. Li, X. Li, Y. Mao and Y. Sun, *Note on the generalized connectivity*, Ars Combin. **114** (2014) 193–202.

[15] H. Li, X. Li and Y. Sun, *The generalized* 3*-connectivity of Cartesian product graphs*, Discrete Math. Theor. Comput. Sci. **14** (2012) 43–54.

[16] X. Li and Y. Mao, *A survey on the generalized connectivity of graphs.* arXiv:1207.1838[math.CO]

[17] X. Li and Y. Mao, *On extremal graphs with at most ℓ internally disjoint Steiner trees connecting any $n-1$ vertices*, Graphs Combin. **31** (2015) 1–29. doi:10.1007/s00373-014-1465-6

[18] X. Li and Y. Mao, *The generalized* 3*-connectivity of lexicographic product graphs*, Discrete Math. Theor. Comput. Sci. **16** (2014) 339–354. doi:10.1007/978-3-319-12691-3_31

[19] X. Li and Y. Mao, *Nordhaus-Gaddum-type results for the generalized edge-connectivity of graphs*, Discrete Appl. Math. **185** (2015) 102–112. doi:10.1016/j.dam.2014.12.009

[20] X. Li, Y. Mao and Y. Sun, *On the generalized* (*edge-*)*connectivity of graphs*, Australas. J. Combin. **58** (2014) 304–319.

[21] X. Li, Y. Mao and L. Wang, *Graphs with large generalized* 3*-edge-connectivity.* arXiv:1201.3699v1 [math.CO]

[22] X. Li, J. Yue, and Y. Zhao, *The generalized* 3*-edge-connectivity of lexicographic product graphs*, Proc. COCOA 2014, Lecture Notes in Comput. Sci. **8881** (2014) 412–425. doi:10.1007/978-3-319-12691-3_31

[23] B. Liouville, *Sur la connectivité des produits de graphes*, C.R. Acad. Sci. Paris Sér. A–B **286** (1978) A363–A365.

[24] O.R. Oellermann, *On the ℓ-connectivity of a graph*, Graphs Combin. **3** (1987) 285–299. doi:10.1007/BF01788551

[25] O.R. Oellermann, *A note on the ℓ-connectivity function of a graph*, Congr. Numer. **60** (1987) 181–188.

[26] G. Sabidussi, *Graphs with given group and given graph theoretical properties*, Canadian J. Math. **9** (1957) 515–525.
doi:0.4153/CJM-1957-060-7

[27] N.A. Sherwani, Algorithms for VLSI Physical Design Automation, 3rd Edition (Kluwar Acad. Pub., London, 1999).

[28] Y. Sun, *Generalized* 3-(*edge*)-*connectivity for undirected double-loop networks*, J. Discrete Math. Sci. Cryptogr. **17** (2014) 19–28.
doi:10.1080/09720529.2013.867672

[29] Y. Sun, *Generalized* 3-*edge-connectivity of Cartesian product graphs*, Czechoslovak Math. J. **65** (2015) 107–117.
doi:10.1007/s10587-015-0162-9

[30] Y. Sun, *Generalized* 3-*connectivity and* 3-*edge-connectivity for the Cartesian products of some graph classes*, J. Combin. Math. Combin. Comput. **94** (2015) 215–225.

[31] Y. Sun, *Maximum generalized local connectivities of cubic Cayley graphs on Abelian groups*, J. Combin. Math. Combin. Comput. **94** (2015) 227–236.

[32] Y. Sun, *Sharp upper bounds for generalized edge-connectivity of product graphs*, Discuss. Math. Graph Theory **36** (2016) 833–843.
doi:10.7151/dmgt.1924

[33] Y. Sun, *On the maximum and minimum sizes of a graph with given k-connectivity*, Discuss. Math. Graph Theory **37** (2017) 623–632.
doi:10.7151/dmgt.1941

[34] Y. Sun and X. Li, *On the difference of two generalized connectivities of a graph*, J. Comb. Optim. **33** (2017) 283–291.
doi:10.1007/s10878-015-9956-9.

[35] Y. Sun and S. Zhou, *Tree connectivities of Cayley graphs on Abelian groups with small degrees*, Bull. Malays. Math. Sci. Soc. **39** (2016) 1673–1685.
doi:10.1007/s40840-015-0147-8.

[36] S. Špacapan, *Connectivity of Cartesian products of graphs*, Appl. Math. Lett. **21** (2008) 682–685.
doi:10.1016/j.aml.2007.06.010

[37] S. Špacapan, *Connectivity of strong products of graphs*, Graphs Combin. **26** (2010) 457–467.
doi:10.1007/s00373-010-0919-8

[38] J. Xu and C. Yang, *Connectivity of Cartesian product graphs*, Discrete Math. **306** (2006) 159–165.
doi:10.1016/j.disc.2005.11.010

Received 13 February 2016
Revised 17 August 2016
Accepted 17 August 2016