Every DFS Tree of a 3-Connected Graph Contains a Contractible Edge

Amr Elmasry* Kurt Mehlhorn† Jens M. Schmidt‡

Abstract

A well-known theorem of Tutte states that every 3-connected graph G on more than 4 vertices contains a contractible edge. In this paper, we strengthen this result by showing that every depth-first-search tree of G contains a contractible edge. Moreover, we show that every spanning tree of G contains a contractible edge if G is 3-regular or if G does not contain two disjoint pairs of adjacent degree-3 vertices. Additionally, we provide several families of graphs for which not every spanning tree contains a contractible edge.

1 Introduction

Let $G = (V, E)$ be a simple undirected graph with $n := |V|$ and $m := |E|$. We denote an edge between vertices x and y by xy. If $xy \in E(G)$, we say that x and y are *neighbors* in G. The *degree* of $x \in V(G)$, denoted by $deg(x)$, is the number of neighbors of x in G.

A graph is *connected* if there is a path between every two of its vertices. For any subset of vertices $V' \subseteq V$, let $G \setminus V'$ denote the graph generated from G by removing the vertices in V' and all their incident edges. A set of vertices whose removal disconnects the graph is called a *vertex cut*. If V' is a vertex cut of G, the maximal connected subgraphs of $G \setminus V'$ are called the *components* of $G \setminus V'$. Vertex cuts of size one, two and three are called *separation vertices*, *separation pairs* and *separation triples*, respectively. Analogously, an *edge cut* of G is a subset of E whose removal disconnects G. For $k > 1$, G is *k-connected* if $n > k$ and removing any $k-1$ of its vertices leaves a connected graph.

The *contraction* of an edge $xy \in E(G)$ generates a graph $G' = G/xy$ with vertex set $V(G') = V(G) \setminus \{x, y\} \cup \{v_{xy}\}$, where v_{xy} is a new vertex. The edge xy is removed and for all edges having exactly one end-vertex in $\{x, y\}$, this vertex is replaced by v_{xy}. Finally, only one edge of each set of parallel edges is kept. Let an edge in a 3-connected graph be *contractible* (also called 3-*contractible*) if its contraction results in a 3-connected graph.

Fifty years ago Tutte [22] proved the fundamental result that every 3-connected graph on more than 4 vertices contains a contractible edge. Since then, the distribution of contractible edges in 3-connected graphs has been intensively studied. Many papers establish

*Computer Science Department, University of Copenhagen, Denmark. This research was partially done when the author was at MPI für Informatik. Supported by an Alexander von Humboldt Fellowship.

†MPI für Informatik, Campus E1 4, 66123 Saarbrücken, Germany

‡Department of Computer Science, FU Berlin, Germany. This research was supported by the Deutsche Forschungsgemeinschaft within the research training group "Methods for Discrete Structures" (GRK 1408).

lower bounds on the number of contractible edges. For example, there are at least $n/2$ contractible edges [1] in every 3-connected graph with $n > 4$, and at least $(2m+12)/7$ contractible edges [18] in every 3-connected graph with $m > 27$. Bounds for the number of contractible edges in longest cycles [3], the number of vertices incident to contractible edges [26] and results on entire contractible subgraphs [11] have also been investigated; see [10] for an excellent survey. Bounds have also been settled for the number of *removable* edges in 3-connected graphs [6, 8] (edges whose removal leave a 3-connected graph).

A *depth-first-search tree* of a graph G is a spanning tree of G produced by the depth-first-search algorithm [12, 21]. Starting from an arbitrary prescribed vertex r, the depth-first-search algorithm traverses the graph by repeatedly visiting an unvisited neighbor of the last visited vertex. If all the neighbors of the current vertex have already been visited, the search backtracks until it finds a vertex with an unvisited neighbor to continue (or all vertices have been visited). By employing a stack of the visited vertices, this algorithm can be implemented to run in $O(n+m)$ time [2]. Alternatively, a *depth-first-search tree* of G can be characterized as a spanning tree T rooted at r with the property that the end-vertices of every edge in $E(G) \setminus E(T)$ are contained in a path in T that ends at r.

We strengthen Tutte's result by investigating the existence of contractible edges in spanning trees of 3-connected graphs. Let T be a spanning tree of a 3-connected graph G with $n > 4$. We show that if T does not contain a contractible edge, there are two disjoint edges $x_1x_2, x_3x_4 \in E(G)$, where each of the 4 vertices has degree 3 and is the end-vertex of exactly one edge of T, such that for $i = 1, 3$, x_i and x_{i+1} have the same neighbor in T. The above property of depth-first-search trees will then immediately ensure that T cannot be a depth-first-search tree. We thus conclude that every depth-first-search tree of a 3-connected graph with $n > 4$ contains a contractible edge.

In addition, we exhibit 3-connected graphs with a depth-first-search tree containing exactly one contractible edge, and 3-connected graphs with a spanning tree containing no contractible edge. The 3-connected graphs with $n > 4$ that admit spanning trees with no contractible edges are rather cunning and tricky; we call such graphs *foxes*. The wheel graphs, W_i for $i \geq 5$, are an example for a family of foxes. We present additional infinite families of foxes, and give conditions under which a 3-connected graph is not a fox.

A *certifying algorithm* is an algorithm that produces, with each output, a *certificate* that the particular output has not been compromised by a bug (see [9] and [15, Section 2.14] for a general discussion of certifying algorithms and [14] for an extensive survey). A certificate for the 3-connectivity of a graph would be the sequence of contractible edges that is generated by repeatedly applying Tutte's theorem until a K_4 is produced.

When we started this research, the best certifying algorithms for 3-connectivity [16, 19] of general graphs had a running time of $O(n^2)$; in contrast, the best non-certifying decision algorithms [7, 17, 24, 25] have a running time of $O(n+m)$ time. We hope that the results of this paper lead to a linear-time certifying algorithm for 3-connectivity. As a first step in this direction, we used this result –the existence of a contractible edge in every depth-first-search tree of a 3-connected graph– to establish an $O(n+m)$-time certifying algorithm for the 3-connectivity of Hamiltonian graphs [4]. A linear-time certifying algorithm was found later by the third author [20]; the algorithm does not use the results of this paper.

2 Preliminaries

The following lemmas are used to prove our claims; their proofs are straightforward and accordingly omitted.

Lemma 2.1 *Let S be a separation triple in a 3-connected graph G, and let C be one of the components of $G \setminus S$. Every vertex in S has a neighbor in C.*

Lemma 2.2 *Let S be a separation triple in a 3-connected graph G, and let C be one of the components of $G \setminus S$. If S' is a separation triple in G with $S' \neq S$ and $S' \subseteq S \cup V(C)$, there is a component of $G \setminus S'$ properly contained in C.*

Lemma 2.3 *Let G be a 3-connected graph, and let $\{x, y, z\}$ and $\{v, y, w\}$ be two separation triples in G intersecting exactly in y. Then v and w are contained in the same component of $G \setminus \{x, y, z\}$ if and only if x and z are contained in the same component of $G \setminus \{v, y, w\}$. Moreover, if v and w belong to distinct components of $G \setminus \{x, y, z\}$, then each of $G \setminus \{x, y, z\}$ and $G \setminus \{v, y, w\}$ has exactly two components.*

According to Veldman [23], two vertex cuts S and S' *interfere* if at least two components of $G \setminus S$ contain vertices of S' and vice versa. We call two separation triples S and S' *crossing* if they interfere and in addition have a common vertex. By Lemma 2.3, each of $G \setminus S$ and $G \setminus S'$ has exactly two components.

Lemma 2.4 *Let G be a 3-connected graph, let $\{x, y, z\}$ and $\{v, y, w\}$ be two crossing separation triples, let D be the component of $G \setminus \{x, y, z\}$ containing v, and let X and Z be the components of $G \setminus \{v, y, w\}$ containing x and z, respectively. Then $\{x, y, v\}$ is a separation triple unless $X \cap D = \emptyset$, and $\{z, y, v\}$ is a separation triple unless $Z \cap D = \emptyset$.*

We also use the following results to prove our claims.

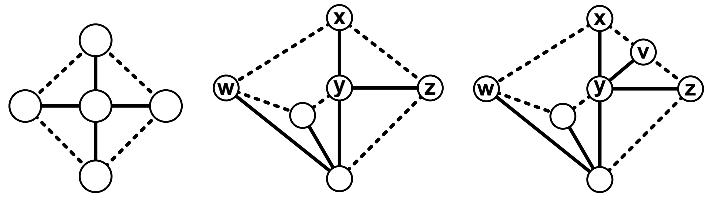
Lemma 2.5 (Dean, Hemminger, Ota [3]) *An edge xy in a 3-connected graph with $n > 4$ is contractible if and only if there is no separation triple containing both x and y.*

Lemma 2.6 (Halin [5]) *In a 3-connected graph with $n > 4$, every degree-3 vertex has an incident contractible edge.*

Lemma 2.7 (Ota [18]) *Let v be a degree-3 vertex in a 3-connected graph G with $n > 4$, and let x, y, and z be its neighbors. If $xy \in E(G)$, then vz is contractible.*

3 Contractible Edges and Spanning Trees

Examples: There are arbitrarily large foxes; the wheel graphs, W_i for $i \geq 5$, with their spokes as spanning trees, form an infinite family; see Figure 1(a). Figure 1(b) shows the base graph of another infinite family of foxes. In this graph, the vertices x, y, and w play a special role. The next larger graph in this family is obtained as follows: Let z be the



(a) W_5 (b) $n = 6$ (c) $n = 7$

Figure 1: The solid edges are non-contractible and form a spanning tree.

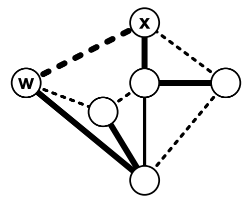


Figure 2: A depth-first-search tree (thick edges) with only one contractible edge wx.

neighbor of x that is neither y nor w in the smaller graph, subdivide xz by one vertex and connect the new vertex with y; see Figure 1(c). More examples are in Figures 5 and 6.

We shall show that every depth-first-search tree of a 3-connected graph on more than 4 vertices contains a contractible edge. The graph on 6 vertices in Figure 2 indicates that this bound is tight. However, we are not aware of any graph on more than 6 vertices that admits a depth-first-search tree containing exactly one contractible edge. $\square$

Consider a fox G, and let T be a spanning tree of G that contains no contractible edge. According to Lemma 2.5, there is a vertex $z \in V(G)$ for every edge $xy \in E(T)$ such that $\{x, y, z\}$ is a separation triple. We call each such $\{x, y, z\}$ a T-*separation* triple. The components that result from the removal of a T-separation triple are called T-*components*. A T-component is *minimal* if there is no T-component properly contained in it. The notions of T-components and minimal T-components are special cases of the more general $\mathfrak{S}$-*fragments* and $\mathfrak{S}$-*ends* definitions, which are studied in [13].

Lemma 3.1 *Let G be a fox, and let T be a spanning tree of G that contains no contractible edge. Then every minimal T-component consists of exactly one vertex, say v. This vertex has degree 3 and is the end-vertex of exactly one edge of T. More precisely, if the neighbors of v in G are x, y, and z with $xy \in E(T)$, then $vz \notin E(T)$, and either $vx \in E(T)$ or $vy \in E(T)$.*

Proof. Let D be a minimal T-component, and let $\{x, y, z\}$ with $xy \in E(T)$ be the associated T-separation triple. Since T is a spanning tree, there exists a vertex $v \in V(D)$ that is a neighbor of x, y, or z in T. We show that D has only one vertex, namely v.

If $vz \in E(T)$, then vz is non-contractible, and hence a separation triple $\{v, z, w\}$ exists. Since $xy \in E(G)$, either $w \in \{x, y\}$ or both x and y are in the same component of $G \setminus \{v, z, w\}$. Consequently, there exists a component C of $G \setminus \{v, z, w\}$ such that $x, y \notin V(S)$. By Lemma 2.1, v has a neighbor, say u, in C. Since $u \notin \{x, y, z\}$, u is in the

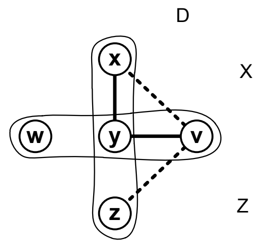


Figure 3: The minimal T-component D.

same component of $G \setminus \{x, y, z\}$ as v, i.e. $u \in V(D)$. It follows that every vertex in C is also in D. Since $v \notin V(C)$, C is properly contained in D, contradicting D being minimal. It follows that $vz \notin E(T)$. Accordingly, either $vx \in E(T)$ or $vy \in E(T)$.

Assume, without loss of generality, that $vy \in E(T)$; see Figure 3. Therefore, vy is non-contractible and a separation triple $\{v, y, w\}$ exists. If there is a component of $G \setminus \{v, y, w\}$ containing neither x nor z, the arguments of the preceding paragraph indicate that the T-component D is not minimal. It follows that $\{v, y, w\}$ splits G into exactly two components, one containing x and one containing z. Call the former component X and the latter Z. We show next that both $X \cap D$ and $Z \cap D$ must be empty.

If $X \cap D \neq \emptyset$, Lemma 2.4 implies that $\{x, y, v\}$ separates $X \cap D$ from the rest of G, contradicting D being minimal. This implies that $X \cap D = \emptyset$. Analogously, $Z \cap D = \emptyset$.

We have thus shown that, assuming $v \in V(G)$ is in a minimal T-component, there exists a separation triple $\{x, y, z\}$ with $xy \in E(T)$, such that $vx, vy, vz \in E(G)$, $deg(v) = 3$, $vz \notin E(T)$ and $vy \in E(T)$. $\square$

We use the next lemma in the proof of Lemma 3.3.

Lemma 3.2 *Let G be a 3-connected graph, let S be a separation triple that splits G in two components, and let X be a component of $G \setminus S$. If $G' = G \setminus V(X)$ is not 2-connected and w' is a separation vertex of G', then $w' \notin S$ and one of the vertices in S has w' as its only neighbor in G' (and hence is a component of $G' \setminus w'$). Conversely, if every vertex in S has at least two neighbors in G', then G' is 2-connected.*

Proof. Assume that G' is not 2-connected. Accordingly, there is a separation vertex w' in G'. If one of the components of $G' \setminus \{w'\}$ does not contain a vertex from S, then w' is a separation vertex in G, contradicting G being 3-connected. It follows that every component of $G' \setminus w'$ contains at least one vertex from S.

Let $S = \{v, y, w\}$. If $w' \in S$, say $w' = y$, then $G' \setminus \{w'\}$ has exactly two components one containing v and one containing w. Since S is a separation triple in G, there are vertices in G' other than those in S. It follows that one of the components of $G' \setminus \{w'\}$ must have at least two vertices, say the component containing w. Then $\{w, w'\}$ splits G, contradicting G being 3-connected. We conclude that $w' \notin S$.

The vertices of S cannot all lie in one component of $G' \setminus \{w'\}$. Hence, at least one of these components contains exactly one vertex from S, say w. If w has a neighbor in G' other than w', then $\{w, w'\}$ splits G, contradicting G being 3-connected. We conclude that one of the vertices in S has w' as its only neighbor in G'. $\square$

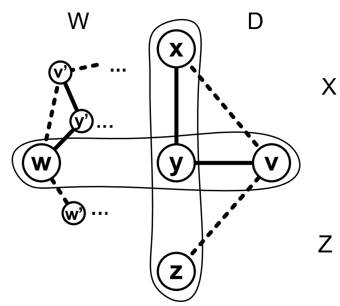


Figure 4: A case contradicting the minimality of X.

Assume that a 3-connected graph G with $n > 4$ is a fox, and let T be a spanning tree of G that contains no contractible edge. Let v be a minimal T-component in G, and let vy be the only tree edge incident with v. We call every T-separation triple $\{v, y, w\}$ a *special T-separation triple*. The components that result from the removal of a special T-separation triple are called *special T-components*. A special T-component is *minimal* if there is no special T-component properly contained in it.

Lemma 3.3 *Let G be a fox, and let T be a spanning tree of G that contains no contractible edge. Then every minimal special T-component consists of exactly one vertex and has a neighbor that is also a minimal special T-component. Let v and v' be any such pair of minimal special T-components with $vv' \in E(G)$. There exists a vertex y such that $vy, v'y \in E(T)$.*

Proof. Let X be a minimal special T-component; it is split off by a special T-separation triple $S = \{v, y, w\}$ with v being a minimal T-component and $vy \in E(T)$. By Lemma 2.2, no other special T-separation triple has its three vertices in $V(X) \cup S$. Since S is a T-separation triple, there exists a minimal T-component $v' \in V(X)$; v' belongs to a special T-separation triple $S' = \{v', y', w'\}$ with $v'y' \in E(T)$, where $y' \in V(X) \cup S$ and $w' \notin V(X) \cup S$ (otherwise, X would not be minimal).

Assume first that $y' \in V(X)$; see Figure 4. Then w' must split $G \setminus X$, and Lemma 3.2 implies that one of the vertices in S has w' as its only neighbor in $G \setminus X$. Since $vy \in E(G)$, such vertex must be w, i.e. $ww' \in E(G)$. We next show that all neighbors of w are contained in S', and hence w has degree 3. Assume to the contrary that w has a neighbor $u' \notin S'$. Then u' and w belong to the same component of $G \setminus S'$. Every path from u' to any vertex in a different component of $G \setminus S'$ must pass through either v', y' or w. Hence, $\{v', y', w\}$ is a special T-separation triple contained in $V(X) \cup S$. But such possibility is ruled out in the previous paragraph because of the minimality of X. It follows that w has degree 3, its neighbors are precisely the vertices in S', and w is a minimal T-component. By Lemma 2.7, ww' is contractible, and accordingly does not belong to T. Also, $wv' \notin E(T)$ since $v'y' \in E(T)$ and v' has only one incident tree edge. Hence, $wy' \in E(T)$. Let z' be the third neighbor of v' besides y' and w. Then $\{w, y', z'\}$ is a special T-separation triple that separates v' from the rest of G. This again contradicts our choice of X being minimal. We conclude that $y' \notin V(X)$, and hence $y' \in S$.

Since v' and w' are in different components of $G \setminus S$, using Lemma 2.3, the triples S and S' cross. Hence, the vertices of $S \setminus \{y'\}$ must belong to different components of $G \setminus S'$. Since $vy \in E(G)$, this excludes the possibility that $y' = w$. Also, $y' \neq v$, since otherwise v would be an end-vertex of two tree edges, namely vy and $v'y'$. It must then be the case that $y = y'$ (and $v' = x$ in Figure 4). If $|V(X)| > 1$, Lemma 2.4 implies that either $\{v', y, v\}$ or $\{v', y, w\}$ is a special T-separation triple. Such a triple has a component properly contained in X, contradicting the minimality of X. It follows that v' is the only vertex in X. Let z be the third neighbor of v besides v' and y. Then $\{v', y, z\}$ is a special T-separation triple that separates v from the rest of G. We conclude that v and v' are both minimal special T-components, $vv' \in E(G)$ and $vy, v'y \in E(T)$. □

Here comes our main theorem implying our best characterization for foxes.

Theorem 3.4 *Let G be a fox, and let T be a spanning tree of G that contains no contractible edge. There exist two disjoint edges in G such that their end-vertices are minimal special T-components.*

Proof. Let v and v' be adjacent minimal special T-components as in Lemma 3.3; v is split off by $S' = \{v', y, w'\}$ and v' is split off by $S = \{v, y, w\}$.

Assume first that there is a minimal special T-component in $V(G) \setminus \{v, v', y, w, w'\}$. Call it z, and let z' be the adjacent minimal special T-component. Then $z' \notin \{v, v'\}$, and hence (v, v') and (z, z') are the desired pairs.

Otherwise, any minimal special T-component of G is contained in $\{v, v', y, w, w'\}$. Let W' be the component of $G \setminus S$ containing w', and let W be the component of $G \setminus S'$ containing w. Both W' and W are special T-components, and hence contain minimal special T-components. These components must be w for W and w' for W'. Then (v, w') and (v', w) are the desired pairs. □

The previous theorem implies that every fox has at least four degree-3 vertices. Remarkably, there are arbitrarily large foxes with exactly four degree-3 vertices; see Figure 5.

Next, we use Theorem 3.4 to prove the following algorithmically-utilizable result.

Corollary 3.5 *Consider a 3-connected graph G with $n > 4$. Every depth-first-search tree of G contains a contractible edge.*

Proof. Let T be a depth-first-search tree of G, and assume that T contains no contractible edge. By Theorem 3.4, there exist two disjoint pairs of degree-3 vertices, each vertex is a minimal T-component, such that the vertices of each pair are adjacent in G. By Lemma 3.1, every minimal T-component is a degree-3 vertex that is either the root or a leaf in T. Accordingly, there exists a pair of vertices that are leaves in T while being adjacent in G, contradicting the fact that T is a depth-first-search tree. □

It is interesting to note that although foxes must have some degree-3 vertices as indicated earlier, not all vertices of a fox can be of degree 3.

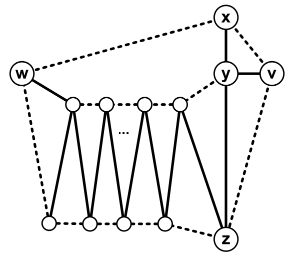


Figure 5: An arbitrarily large fox with exactly four degree-3 vertices.

Lemma 3.6 *If G is a 3-connected 3-regular graph with $n > 4$, then G is not a fox.*

Proof. Assume that G has a spanning tree T containing no contractible edge. According to Lemma 3.1, there are vertices $v, x, y, z \in V(G)$, such that $vx, vy, vz \in E(G)$, $xy, vy \in E(T)$ but $vz \notin E(T)$. Because G is 3-regular, $deg(x) = deg(y) = 3$. As T is a spanning tree of G, either the third edge incident with x, say xr, or the third edge incident with y, say ys, is a tree edge. Since $vy \in E(G)$, xr is contractible by Lemma 2.7. Since $xy, vy \in E(T)$, both edges are non-contractible by assumption. Accordingly, ys is contractible by Lemma 2.6. This contradicts the assumption that T contains no contractible edge. $\square$

As a last remark, we give another property for which a 3-connected graph is not a fox.

Lemma 3.7 *Let G be a 3-connected graph with $n > 4$, and let F be an edge cut of G. If every edge e in F has an end-vertex x, where $\deg(x) = 3$ and x has two neighbors in $G \setminus F$ adjacent to each other, then G is not a fox.*

4 Conclusions

Our main objective was to show that every depth-first-search tree of a 3-connected graph with $n > 4$ contains a contractible edge. However, there are 3-connected graphs that have spanning trees with no contractible edges. We called those graphs foxes. Foxes seem to be rare though; we supported this statement by giving restricted properties for foxes.

An interesting fact is that all wheel graphs, as well as the family of foxes depicted in Figure 1, satisfy $m = 2n-2$. This raises the question about the existence of an infinite class of foxes where $|m - 2n|$ grows with n. So far, we have only found foxes with $|m - 2n| \leq 3$; see Figure 6 for an extremal example. Another open question is whether there exists an inductive characterization of foxes. Such a characterization may provide more insight into the distribution of contractible edges in 3-connected graphs.

Acknowledgements

We thank the anonymous referees for their constructive comments.

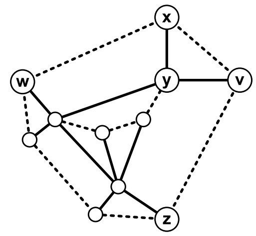


Figure 6: A fox with $m = 2n - 3$.

References

[1] K. Ando, H. Enomoto, and A. Saito. Contractible edges in 3-connected graphs. *J. Comb. Theory Ser. B*, 42(1):87–93, 1987.

[2] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*. The MIT Press, Cambridge, 3nd edition, 2009.

[3] N. Dean, R. Hemminger, and K. Ota. Longest cycles in 3-connected graphs contain three contractible edges. *J. Graph Theory*, 12(1):17–21, 1989.

[4] A. Elmasry, K. Mehlhorn, and J. M. Schmidt. An $O(n+m)$ certifying triconnnectivity algorithm for Hamiltonian graphs. *Algorithmica*, 2011. To appear.

[5] R. Halin. Zur Theorie der n-fach zusammenhängenden Graphen. *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg*, 33(3):133–164, 1969.

[6] D. A. Holton, B. Jackson, A. Saito, and N. C. Wormald. Removable edges in 3-connected graphs. *J. Graph Theory*, 14(4):465–473, 1990.

[7] J. E. Hopcroft and R. E. Tarjan. Dividing a graph into triconnected components. *SIAM J. Comput.*, 2(3):135–158, 1973.

[8] H. Kang, J. Wu, and G. Li. Removable edges of a spanning tree in 3-connected 3-regular graphs. In *Frontiers in Algorithmics, LNCS 4613*, pages 337–345, 2007.

[9] D. Kratsch, R. M. McConnell, K. Mehlhorn, and J. P. Spinrad. Certifying algorithms for recognizing interval graphs and permutation graphs. *SIAM J. Comput.*, 36(2):326–353, 2006.

[10] M. Kriesell. A survey on contractible edges in graphs of a prescribed vertex connectivity. *Graphs and Combinatorics*, 18(1):1–30, 2002.

[11] M. Kriesell. On the number of contractible triples in 3-connected graphs. *J. Comb. Theory Ser. B*, 98(1):136–145, 2008.

[12] É. Lucas. *Récréations Mathématiques. Part 1.* Gauthier-Villars et Fils, Paris. second edition, 1891 (first edition in 1881).

[13] W. Mader. Generalizations of critical connectivity of graphs. *Discrete Math.*, 72:267–283, 1988.

[14] R.M. McConnell, K. Mehlhorn, S. Nher, and P. Schweitzer. Certifying algorithms. *Computer Science Review*, 5(2):119–161, 2011.

[15] K. Mehlhorn, S. Näher, and C. Uhrig. *The LEDA Platform of Combinatorial and Geometric Computing*. Cambridge University Press, 1999.

[16] K. Mehlhorn and P. Schweitzer. Progress on certifying algorithms. In *Proceedings of the 4th International Workshop on Frontiers in Algorithmics (FAW'10)*, pages 1–5, 2010.

[17] G. L. Miller and V. Ramachandran. A new graph triconnectivity algorithm and its parallelization. *Combinatorica*, 12(1):53–76, 1992.

[18] K. Ota. The number of contractible edges in 3-connected graphs. *Graphs and Combinatorics*, 4(1):333–354, 1988.

[19] J. M. Schmidt. Construction sequences and certifying 3-connectedness. In *27th International Symposium on Theoretical Aspects of Computer Science (STACS'10), Nancy, France*, pages 633–644, 2010.

[20] J. M. Schmidt. *Structure and Constructions of 3-Connected Graphs*. PhD thesis, Freie Universität Berlin, Germany, 2011.

[21] R. E. Tarjan. Depth-first search and linear graph algorithms. *SIAM Journal on Computing*, 1(2):146–160, 1972.

[22] W. T. Tutte. A theory of 3-connected graphs. *Indag. Math.*, 23:441–455, 1961.

[23] H. Veldman. Non k-critical vertices in graphs. *Discrete Math.*, 44:105–110, 1983.

[24] K.-P. Vo. Finding triconnected components of graphs. *Linear and Multilinear Algebra*, 13:143–165, 1983.

[25] K.-P. Vo. Segment graphs, depth-first cycle bases, 3-connectivity, and planarity of graphs. *Linear and Multilinear Algebra*, 13:119–141, 1983.

[26] H. Wu. Contractible elements in graphs and matroids. *Combinatorics, Probability and Computing*, 12(4):457–465, 2003.