Finding a subdivision of a digraph

Jørgen Bang-Jensen, Frédéric Havet, Ana Karolinna Maia de Oliviera

Finding a subdivision of a digraph

Jørgen Bang-Jensen[1,2], Frédéric Havet[3], A. Karolinna Maia[3]

Abstract

We consider the following problem for oriented graphs and digraphs: Given a directed graph D, does it contain a subdivision of a prescribed digraph F? We give a number of examples of polynomial instances, several NP-completeness proofs as well as a number of conjectures and open problems.

1. Introduction

Many interesting classes of graphs are defined by forbidding induced subgraphs, see [7] for a survey. This is why the detection of several kinds of induced subgraphs is interesting, see [15] where several such problems are surveyed. In particular, the problem of deciding whether a graph G contains, as an induced subgraph, some graph obtained after possibly subdividing prescribed edges of a prescribed graph H has been studied. This problem can be polynomial-time solvable or NP-complete according to H and to the set of edges that can be subdivided. The aim of the present work is to investigate various similar problems in digraphs, focusing only on the following problem: given a digraph H, is there a polynomial-time algorithm to decide whether an input digraph G contains a subdivision of H?

Of course the answer depends heavily on what we mean by "contain". Let us illustrate this by surveying what happens in the realm of undirected graphs. If the containment relation is the subgraph containment, then for any fixed H, detecting a subdivision of H in an input graph G can be performed in polynomial time by the Robertson and Seymour linkage algorithm [18] (for a short explanation of this see e.g. [3]). But, if we want to detect an *induced* subdivision of H, then the answer depends on H (assuming P$\neq$NP). It is proved in [15] that detecting an induced subdivision of K_5 is NP-complete, and the argument can be reproduced for any H whose minimum degree is at least 4. Polynomial-time solvable instances trivially exist, such as detecting an induced subdivision of H when H is a path, or a graph on at most 3 vertices. But non-trivial polynomial-time solvable instances also exist, such as detecting an induced subdivision of $K_{2,3}$ which can be performed in $O(n^{11})$ time by Chudnovsky and Seymour's three-in-a-tree algorithm, see [8]. Note that for many graphs H, nothing is known about the complexity of detecting an induced subdivision of H: when H is cubic (in particular when $H = K_4$) or when H is a disjoint union of two triangles, and in many other cases.

[1]Department of Mathematics and Computer Science, University of Southern Denmark, Odense DK-5230, Denmark (email: jbj@imada.sdu.dk).

[2]Most of this work was done while J. Bang-Jensen visited Projet Mascotte, I3S (CNRS, UNSA) and INRIA, Sophia Antipolis whose hospitality and financial support is gratefully acknowledged.

[3]Projet Mascotte, I3S (CNRS, UNSA) and INRIA, Sophia Antipolis. Partly supported by ANR Blanc AGAPE and CAPES/Brazil. (email:[frederic.havet, karol.maia]@inria.fr)

Preprint submitted to Theoretical Computer Science *June 26, 2013*

When we move to digraphs, the situation becomes more complicated, even for the subdigraph containment relation. In this paper, by digraph we mean a simple digraph, that is a digraph with no parallel arcs nor loops. Sometimes however, multiple arcs are possible. In such cases, we write multidigraph. We rely on [1] for classical notation and concepts. A few things need to be stated here though. Unless otherwise stated the letters n and m will always denote the number of vertices and arcs (edges) of the input digraph (graph) of the problem in question. By *linear time*, we mean $O(n+m)$ time. If D is a digraph, then we denote by $UG(D)$ the underlying (multi)graph of D, that is, the (multi)graph we obtain by replacing each arc by an edge. A digraph D is *connected* if $UG(D)$ is a connected graph. If xy is an arc from x to y, then we say that x *dominates* y. When H, H' are digraphs we denote by $H+H'$ the disjoint union of H and H' (no arcs between disjoint copies of these).

A *subdivision of a digraph* F, also called an F*-subdivision*, is a digraph obtained from F by replacing each arc ab of F by a directed (a,b)-path.

In this paper, we consider the following problem for a fixed digraph F.

F-SUBDIVISION
Input: A digraph D.
Question: Does D contain a subdivision of F as a subgraph?

In [2] the problem INDUCED-F-SUBDIVISION of finding an induced subdivision of a prescribed digraph F in a given digraph D was studied. It turns out that here there is a big difference in the complexity of the problem depending on whether or not D is an oriented graph or it may contain 2-cycles. In the latter case INDUCED-F-SUBDIVISION is NP-complete for every oriented digraph F which is not the disjoint union of spiders (see definition of these digraphs below) and it was conjectured that INDUCED-F-SUBDIVISION is NP-complete unless F is the disjoint union of spiders and at most one 2-cycle.

Let $x_1,x_2,\dots,x_k,y_1,y_2,\dots,y_k$ be distinct vertices of a digraph D. A *k-linkage* from $(x_1,x_2,\dots,x_k)$ to $(y_1,y_2,\dots,y_k)$ in D is a system of disjoint directed paths $P_1,P_2,\dots,P_k$ such that P_i is an (x_i,y_i)-path in D.

Similarly to the situation for undirected graphs, the D-SUBDIVISION problem is related to the following k-LINKAGE problem.

k-LINKAGE
Input: A digraph D and $2k$ distinct vertices $x_1,x_2,\dots,x_k,y_1,y_2,\dots,y_k$.
Question: Is there a k-linkage from $(x_1,x_2,\dots,x_k)$ to $(y_1,y_2,\dots,y_k)$ in D?

However, contrary to graphs, unless P=NP, k-LINKAGE cannot be solved in polynomial time in general digraphs. Fortune, Hopcroft and Wyllie [10] showed that already 2-LINKAGE is NP-complete. Using this result, we show that for lots of F, the F-SUBDIVISION problem is NP-complete. We also give some digraphs F for which we prove that F-SUBDIVISION is polynomial-time solvable. We believe that there is a dichotomy between NP-complete and polynomial-time solvable instances.

Conjecture 1. For every digraph F, the F-SUBDIVISION problem is polynomial-time solvable or NP-complete.

To prove such a conjecture, a first idea would be to try to establish for any digraph G and subdigraph F, that if F-SUBDIVISION is NP-complete, then G-SUBDIVISION is also NP-complete, and conversely, if G-SUBDIVISION is polynomial-time solvable, then F-SUBDIVISION is polynomial-time solvable. However, these two statements are false as shown by the two digraphs depicted Figure 1. The NP-completeness of A-SUBDIVISION follows Theorem 12. The fact that B-SUBDIVISION is polynomial-time solvable is proved in Theorem 27.

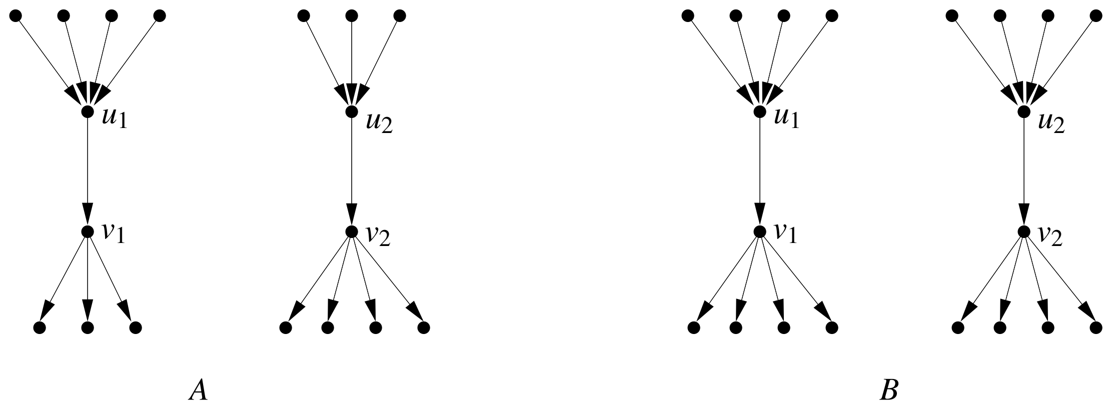

Figure 1: Digraphs A and B such that A is a subdigraph of B, A-SUBDIVISION is NP-complete, and B-SUBDIVISION is polynomial-time solvable.

The paper is organized as follows. We start by giving some general lemmas which allow to extend NP-completeness results of F-SUBDIVISION for some digraphs F to much larger classes of digraphs. Next we give a powerful tool, based on a reduction from the NP-complete 2-linkage problem in digraphs, which can be applied to conclude the NP-completeness of F-SUBDIVISION for the majority of all digraphs F. We then describe different algorithmic tools for proving polynomial-time solvability of certain instances of F-SUBDIVISION. We first give some easy brute force algorithms, then algorithms based on maximum-flow calculations and finally algorithms based on handle decompositions of strongly connected digraphs. After this we give a number of classes of digraphs for which the F-SUBDIVISION is polynomial-time solvable for every F. Then we treat F-SUBDIVISION when F belongs to some special classes of digraphs such as disjoint unions of cycles, wheels, fans, transitive tournaments, oriented paths or cycles or F has at most 3 vertices. Finally, we conclude with some open problems, including an interesting conjecture due to Seymour, which if true would imply some of the polynomial cases treated in this paper.

2. Some general lemmas

Lemma 2. *Let F_1 and F_2 be two digraphs.*

(i) If F_1-SUBDIVISION is NP-complete, then (F_1+F_2)-SUBDIVISION is NP-complete.

(ii) If (F_1+F_2)-SUBDIVISION is polynomial-time solvable, then F_1-SUBDIVISION is polynomial-time solvable.

Proof. Let D be a digraph. We shall prove that D contains an F_1-subdivision if and only if $D+F_2$ contains an (F_1+F_2)-subdivision.

Clearly if D contains an F_1-subdivision S, then $S+F_2$ is an (F_1+F_2)-subdivision in $D+F_2$.

Conversely, assume that $D+F_2$ contains an (F_1+F_2)-subdivision $S=S_1+S_2$ with S_1 an F_1-subdivision and S_2 an F_2-subdivision. Let us consider such an (F_1+F_2)-subdivision that maximizes the number of connected components[4] of F_2 that are mapped (in S) into F_2 again (notice that since there are no arcs between D and F_2 in $D+F_2$, in the subdivision S every component of S_2 will either be entirely inside F_2 or entirely inside D). We claim that $S_2=F_2$. Indeed suppose that some component T of S_2 is in D. Let C be the component of F_2 of which T is the subdivision. Let $U=S\cap C$. Then T contains a subdivision U' of U (because it is a subdivision of all of C). Hence replacing U by U' and T by C in S, we obtain a subdivision with one more component mapped on itself, a contradiction.

Hence $S_2=F_2$, and so D contains S_1 which is an F_1-subdivision. □

Lemma 3. *Let F_1 and F_2 be two digraphs such that F_1 is strongly connected and F_2 contains no F_1-subdivision. Let F be obtained from F_1 and F_2 by adding some arcs with tail in $V(F_1)$ and head in $V(F_2)$.*

*(i) If F_1-*SUBDIVISION *is NP-complete, then F-*SUBDIVISION *is NP-complete.*

*(ii) If F-*SUBDIVISION *is polynomial-time solvable, then F_1-*SUBDIVISION *is polynomial-time solvable.*

Proof. We shall prove that a digraph D contains an F_1-subdivision if and only if $D\mapsto F_2$ contains an F-subdivision, where $D\mapsto F_2$ is obtained from $D+F_2$ by adding all possible arcs from $V(D)$ to $V(F_2)$.

It is easy to see that if D contains an F_1-subdivision S, then $S+F_2$ together with some subset of the arcs from D to F_2 is an F-subdivision in $D\mapsto F_2$. Conversely, if $D\mapsto F_2$ contains an F subdivision S^*, then, since F_1 is strongly connected, the part of S^* forming a subdivision of F_1 has to lie entirely inside D or F_2. Since F_2 contains no F_1-subdivision, the subdivision of F_1 has to be inside D and hence we get that D has an F_1-subdivision. □

It is useful to look at Figure 1 again and notice that the digraphs A,B show that we need the assumption that F_1 is strongly connected in Lemma 3 (and the analogous version where the roles of F_1 and F_2 are interchanged).

A digraph D is *robust* if it is strongly connected and $UG(D)$ is 2-connected.

Lemma 4. *Let F_1 and F_2 be two digraphs such that F_1 is robust and F_2 contains no F_1-subdivision. Let F be obtained from F_1 and F_2 by identifying one vertex of F_1 with one vertex of F_2.*

*(i) If F_1-*SUBDIVISION *is NP-complete, then F-*SUBDIVISION *is NP-complete.*

*(ii) If F-*SUBDIVISION *is polynomial-time solvable, then F_1-*SUBDIVISION *is polynomial-time solvable.*

[4]A connected component of a digraph H is a connected component of $UG(H)$.

Proof. Given a digraph D we form the digraph D^{F_2} by fixing one vertex x in F_2 and adding $|V(D)|$ disjoint copies of F_2 such that the ith copy has its copy of x identified with the ith vertex of D. It is easy to check that D^{F_2} contains an F-subdivision if and only if D contains an F_1-subdivision. This follows from the fact that F_2 contains no F_1-subdivision and $UG(F_1)$ is 2-connected. □

Lemma 5. *Let F be a digraph in which every vertex v satisfies* $\max\{d^+(v), d^-(v)\} \geq 2$*, and let S be a subdivision of F.*

*(i) If F-*SUBDIVISION *is NP-complete, then S-*SUBDIVISION *is NP-complete.*

*(ii) If S-*SUBDIVISION *is polynomial-time solvable, then F-*SUBDIVISION *is polynomial-time solvable.*

Proof. We shall prove a polynomial reduction from F-SUBDIVISION to S-SUBDIVISION.

Let D be an instance of F-SUBDIVISION and p be the length of a longest path in S corresponding to an arc in F. Let D_p be the D-subdivision obtained by replacing every arc of D by a directed path of length p. One easily checks that D has an F-subdivision if and only if D_p has an S-subdivision. It follows from the fact that every vertex v corresponding to one of F in S must be mapped onto a vertex corresponding to D in D_p because $\max\{d^+(v), d^-(v)\} \geq 2$. □

We believe that the condition $\max\{d^+(v), d^-(v)\} \geq 2$ for all $v \in V(F)$ is not necessary, although it is in our proof.

Conjecture 6. Let F be a digraph, and let S be a subdivision of F.

(i) If F-SUBDIVISION is NP-complete, then S-SUBDIVISION is NP-complete.

(ii) If S-SUBDIVISION is polynomial-time solvable, then F-SUBDIVISION is polynomial-time solvable.

3. General NP-completeness results

3.1. The tool

The following observations allow us to conclude that F-subdivision is "almost always" NP-complete. We use an easy modification of the 2-linkage problem as the basis for these proofs.

A vertex v is said to be *small* if $d^-(v) \leq 2$, $d^+(v) \leq 2$ and $d(v) \leq 3$. A non-small vertex is called *big*.

Theorem 7. *The* 2-LINKAGE *problem is NP-complete even when restricted to digraphs with no big vertices in which x_1 and x_2 are sources and y_1 and y_2 are sinks.*

Proof. Reduction from 2-LINKAGE in general digraphs.

An *out-arborescence* is the orientation of a tree in which all vertices have in-degree 1 expect one special vertex, called the *root*. A *switching out-arborescence* is an out-arborescence, in which the root has out-degree 1, the leaves have out-degree 0 and all other vertices have out-degree 2. A *switching in-arborescence* is the dual notion to out-arborescence.

Let D be a digraph and x_1, x_2, y_1, y_2 four vertices. Let D^* be the digraph obtained from D by deleting all the arcs entering x_1 and x_2 and all the arcs leaving y_1 and y_2. Let $S(D)$ be the digraph obtained from D^* as follows. For every vertex v, replace all the arcs leaving v by a switching out-arborescence with root v and whose leaves corresponds to the out-neighbours of v in D^*, and replace all the arcs entering v by a switching in-arborescence with root v and whose leaves corresponds to the in-neighbours of v in D^*. It is clear that $S(D)$ has no big vertices and that x_1 and x_2 are sources and y_1 and y_2 are sinks. Furthermore, one checks easily that there is a 2-linkage from (x_1, x_2) to (y_1, y_2) in D if and only if there is a 2-linkage from (x_1, x_2) to (y_1, y_2) in $S(D)$. $\square$

3.2. A general NP-completeness theorem

For a digraph D, we denote by $B(D)$ the set of its big vertices. A *big path* in a digraph is a directed path whose endvertices are big and whose internal vertices all have in- and out-degree one in D (in particular an arc between two big vertices is a big path). Note also that two big paths with the same endvertices are necessarily internally disjoint.

The *big paths digraph* of D, denoted $BP(D)$, is the multidigraph with vertex set $V(D)$ in which there are as many arcs between two vertices u and v as there are big (u,v)-paths in D. By the remark above $BP(D)$ is well-defined and easy to construct in polynomial time given D.

Theorem 8. *Let F be a digraph. If F contains two arcs ab and cd whose endvertices are big vertices and such that $(BP(F) \setminus \{ab, cd\}) \cup \{ad, cb\}$ is not isomorphic to $BP(F)$, then F-*SUBDIVISION *is NP-complete.*

Proof. Reduction from 2-LINKAGE in digraphs with no big vertices in which x_1 and x_2 are sources and y_1 and y_2 are sinks.

Let D, x_1, x_2, y_1, y_2 be an instance of this problem. Let H be the digraph obtained from the disjoint union of $F \setminus \{ab, cd\}$ and D by adding the arcs ax_1, cx_2, y_1b, and y_2d. We claim that H has an F-subdivision if and only if D has a 2-linkage from (x_1, x_2) to (y_1, y_2).

Clearly, if there is a 2-linkage P_1, P_2 in D, then the union of $F \setminus \{ab, cd\}$ and the paths $ax_1P_1y_1b$ and $cx_2P_2y_2d$ is a F-subdivision in H.

Conversely, suppose that H contains an F-subdivision S. Observe that in H, no vertex of D is big. Hence, since S has as many big vertices as F, F and S have the same set of big vertices.

Clearly, S contains as many big paths as F and thus there must be in D two disjoint directed paths between (x_1, x_2) and (y_1, y_2). These two paths cannot be an (x_1, y_2)- and an (x_2, y_1)-path, for otherwise $(BP(F) \setminus \{ab, cd\}) \cup \{ad, cb\} = BP(S)$ would be isomorphic to $BP(F)$ since S is an F-subdivsion. Hence, there is 2-linkage from (x_1, x_2) to (y_1, y_2). $\square$

Remark 9. Observe that if $BP(F)$ has two arcs ab and cd which are consecutive (i.e. $b = c$) or contains an antidirected path (a, b, c, d) of length 3, then $(BP(F) \setminus \{ab, cd\}) \cup \{ad, cb\}$ is not isomorphic to $BP(F)$. Hence, by Theorem 8, F-SUBDIVISION is NP-complete.

Corollary 10. *If F is a digraph with no small vertices, then F-*SUBDIVISION *is NP-complete.*

Proof. If F has no small vertices, then $BP(F) = F$. Moreover if F does not contain two consecutives arcs, then $V(F)$ can be partitionned into two sets A and B such that all arcs in F have tail in A and head in B. In this case, F contains an antidirected path of length 3. So by Remark 9, the F-SUBDIVISION problem is NP-complete. $\square$

For many digraphs F, the condition of Theorem 8 is verified and so F-SUBDIVISION is NP-complete. However, there are graphs F that do not verify this condition but for which F-SUBDIVISION is NP-complete as we shall prove in the following subsection.

3.3. Dumbbells

An *oriented path* is an orientation of an undirected path. Let $P = (x_1, \cdots, x_n)$ be an oriented path. If x_1x_2 is an arc, then P is an *out-path*, otherwise P is an *in-path*. In particular, if P is a directed path then it is an out-path. The *blocks* of P are the maximal directed subpaths of P. We often enumerate them from the origin to the terminus of the path. The number of blocks of P is denoted by $b(P)$.

A *dumbbell* is a digraph D with exactly two big vertices u and v which are connected by an induced oriented (u,v)-path P such that removing the internal vertices of P leaves a digraph with two connected components, one L containing u and one R containing the terminus v. The subdigraph L (resp. R) is the *left* (resp. *right*) *plate* of the dumbbell, vertex u is its *left clip*, vertex v its *right clip* and P its *bar*.

A *dumbbell set* is a disjoint union of dumbbells. In this subsection, we shall give some necessary conditions for F-SUBDIVISION to be NP-complete, F being a dumbbell set. In Subsection 5.3, we give particular cases when F-SUBDIVISION is polynomial-time solvable.

A pair of oriented paths (P,Q) is a *bad pair* if one of the following holds:

- P and Q are both directed paths;
- $\{b(P), b(Q)\} = \{1,2\}$;
- P and Q are both out-paths and $\{b(P), b(Q)\} \in \{\{2,2\};\{2,4\}\}$;
- P and Q are both in-paths and $\{b(P), b(Q)\} \in \{\{2,2\};\{2,4\}\}$.

Lemma 11. *Let P and Q be two oriented paths. If (P,Q) is not a bad pair, then there exists $ab \in A(P)$ and $cd \in A(Q)$ such that the two oriented paths P' and Q' obtained from P and Q by replacing ab and cd by ad and cb verify $\{b(P), b(Q)\} \neq \{b(P'), b(Q')\}$.*

Proof. Let (P,Q) be a non-bad pair of paths. Without loss of generality, we may assume that $b(Q) \geq b(P)$. In particular this implies $b(Q) \geq 3$.

Assume that P is an out-path (resp. in-path) and Q is an in-path (resp. out-path). If $b(P) \geq 2$, then take ab as an arc of the first block of P and cd an arc of the first block of Q. Replacing ab and cd by ad and cb results necessarily in $b(P') = 1$ and $b(Q') = b(P) + b(Q) - 1$. If $b(P) = 1$, take ab as an arc of the first block of P and cd an arc of the second block of Q. Then $\{b(P'), b(Q')\} = \{2, b(Q) - 1\} \neq \{b(P), b(Q)\}$.

So we may assume that P and Q are both out-paths or both in-paths. Observe that this in particular implies that P and Q have an even number of blocks, because the opposite path (same digraph but starting form the terminus and ending at the origin) of an out-path with an odd number of blocks is an in-path with an odd number of blocks.

Take an arc ab of the first block of P and an arc cd of the second block of Q. Then one of P', Q' has two blocks and the other $b(P) + b(Q) - 2$ blocks. So if $\{b(P), b(Q)\} \neq \{2, b(P) + b(Q) - 2\}$, we have the result. Hence we may assume that $\{b(P), b(Q)\} = \{2, b(P) + b(Q) - 2\}$, so $b(P) = 2$ because $b(Q) \geq 3$.

Hence $b(Q) \geq 6$, because (P,Q) is not bad. Take ab be an arc of the first block of P and cd an arc of the third block of Q. Then one of P', Q' has four blocks and the other has $b(P)+b(Q)-4$ blocks, so we have the result. □

If two digraphs D and D' are isomorphic, then we write $D \cong D'$. If they are not, then we write $D \ncong D'$.

Theorem 12. *Let F be a dumbbell set. Let D_1 and D_2 be two dumbbells of F, and for $i = 1,2$, let L_i, R_i, u_i, v_i and P_i be the left plate, right plate, left clip, right clip and bar of D_i. If one of the following holds*

(a) (P_1,P_2) is not a bad pair,

(b) $L_1 \ncong L_2$, $L_1 \ncong R_2$, $R_1 \ncong L_2$ and $R_1 \ncong R_2$,

(c) P_1 and P_2 are both directed paths, $L_1 \ncong L_2$ and $R_1 \ncong R_2$,

(d) P_1 is a directed path and P_2 is an out-path (resp. in-path) with two blocks and $L_1 \ncong L_2$ or $L_1 \ncong R_2$ (resp. $R_1 \ncong L_2$ or $R_1 \ncong R_2$).

*then F-*SUBDIVISION *is NP-complete.*

Proof. By Lemma 2, it is sufficient to prove it when $F = D_1 + D_2$. The proof is very similar to the one of Theorem 8. We give a reduction from 2-LINKAGE in digraphs with no big vertices in which x_1 and x_2 are sources and y_1 and y_2 are sinks.

Let D, x_1, x_2, y_1, y_2 be an instance of this problem. Let ab be an arc of the bar of D_1 and cd be an arc of the bar of D_2. Moreover, if (P_1,P_2) is not a bad pair, we choose ab and cd as decribed in Lemma 11. Let H be the digraph obtained from the disjoint union of $F \setminus \{ab,cd\}$ and D by adding the arcs ax_1, cx_2, y_1b, and y_2d. We can then show that H has an F-subdivision if and only if D has a 2-linkage from (x_1,x_2) to (y_1,y_2).

Clearly, if there is a 2-linkage R_1,R_2 in D, then the union of $F \setminus \{ab,cd\}$ and the paths $ax_1R_1y_1b$ and $cx_2R_2y_2d$ is an F-subdivision in H.

Conversely, suppose that H contains an F-subdivision S. For each vertex x of F, we denote by x^* the vertex corresponding to x in S and for any subdigraph G of F, we denote by G^* the subdigraph of S corresponding to the subdivision of G.

In H, no vertex of D is big, so the sole big vertices of D are the clips of D_1 and D_2. Hence $\{u_1^*, v_1^*, u_2^*, v_2^*\} = \{u_1, v_1, u_2, v_2\}$. Now in S, the paths P_1^* and P_2^* connect big vertices. For connectivity reasons these two paths must use $P_1 \setminus ab$ and $P_2 \setminus cd$. In particular, $(L_1 + L_2 + R_1 + R_2)^*$ is a subdigraph of $L_1 + L_2 + R_1 + R_2$. So $(L_1 + L_2 + R_1 + R_2)^* = L_1 + L_2 + R_1 + R_2$. So for any $G \in \{L_1, L_2, R_1, R_2\}$, the digraph G^* is isomorphic to G and is one of the subdigraphs L_1, L_2, R_1 and R_2.

Moreover $b(P_i^*) = b(P_i)$ for $i = 1,2$. Hence, the subpaths of $P_1^* \cap D$ and $P_2^* \cap D$ must be two disjoint directed paths in D, with origins in $\{x_1,x_2\}$ and terminus in $\{y_1,y_2\}$, for otherwise $b(P_1^*) + b(P_2^*) > b(P_1) + b(P_2)$.

Let P_1' and P_2' be the oriented paths obtained from P_1 and P_2 by replacing ab and cd by ad and cb. By construction, if there is no 2-linkage from (x_1,x_2) to (y_1,y_2) in D, then P_1^* and P_2^* consist in a P_1'-subdivision and a P_2'-subdivision, and so $\{b(P_1'), b(P_2')\} = \{b(P_1^*), b(P_2^*)\}$.

(a) If (P_1,P_2) is not a bad pair, then by our choice of ab and cd, $\{b(P'_1),b(P'_2)\} \neq \{b(P_1),b(P_2)\}$. Since $b(P^*_1) = b(P_1)$ and $b(P^*_2) = b(P_2)$, there is a 2-linkage from (x_1,x_2) to (y_1,y_2) in D.

(b) If $L_1 \not\cong L_2$ and $L_1 \not\cong R_2$, then $L^*_1 \in \{L_1,R_1\}$. Similarly, if $R_1 \not\cong L_2$ and $R_1 \not\cong R_2$, then $R^*_1 \in \{L_1,R_1\}$. Hence P^*_1 must go from u_1 to v_1, and so $P^*_1 \cap D$ is a directed (x_1,y_1)-path. Hence there is a 2-linkage from (x_1,x_2) to (y_1,y_2) in D.

(c) If P_1 and P_2 are both directed paths, then $\{u^*_1,u^*_2\} = \{u_1,u_2\}$ as there are the origin of P^*_1 and P^*_2. Now, since $L_1 \not\cong L_2$, we have $L^*_1 = L_1$ and $L^*_2 = L_2$. Similarly, $R^*_1 = R_1$ and $R^*_2 = R_2$. Hence, $P^*_1 \cap D$ and $P^*_2 \cap D$ form a 2-linkage from (x_1,x_2) to (y_1,y_2) in D.

(d) Assume that P_1 is a directed path and that P_2 is an out-path with two blocks. (The proof is analoguous when P_2 is an in-path with two blocks.)

Assume that $L_1 \not\cong L_2$. Then we can choose cd to be an arc of the first block of P_2. Necessarily, $v^*_1 = v_1$ and $R^*_1 = R_1$ since v^*_1 is the only clip with out-degree 0 in $P^*_1 \cup P^*_2$. It follows that $L^*_1 \in \{L_1,L_2\}$, and so $L^*_1 = L_1$ because $L_1 \not\cong L_2$. Thus $P^*_1 \cap D$ is a directed (x_1,y_1)-path and there is a 2-linkage from (x_1,x_2) to (y_1,y_2) in D.

If $L_1 \not\cong R_2$, we get the result similarly by choosing cd to be an arc of the second block of P_2.

□

4. Easy polynomial-time solvable F-subdivision problems

There are digraphs F for which F-SUBDIVISION can be easily proved to be polynomial-time solvable.

A *spider* is a tree obtained from disjoint directed paths by identifying one end of each path into a single vertex. This vertex is called the *body* of the spider.

Proposition 13. *If F is the disjoint union of spiders, then F-*SUBDIVISION *can be solved in $O(n^{|V(F)|})$ time.*

Proof. A digraph D contains an F-subdivision if and only if it contains F as a subdigraph. This can be checked in $O(n^{|V(F)|})$ time. □

A natural question is to ask whether the problem remains polynomial-time solvable when the spider F is no more fixed but specified in the input.

Problem 14. Is the following problem is polynomial-time solvable?
SPIDER-SUBDIVISION
Input: A spider F and a digraph D.
Question: Does D contain a subdivision of F?

Similarly, one could ask if SPIDER-SUBDIVISION can be solved in FPT time whem parameterized by F, that is in $f(|V(F)|) \times n^c$ time, where f is a computable function and c an absolute constant.

Lemma 15. *Let F_1 be a digraph and S a disjoint union of spiders. If F_1-*SUBDIVISION *is polynomial-time solvable, then (F_1+S)-*SUBDIVISION *is also polynomial-time solvable.*

Proof. For each set A of $|S|$ vertices, we check if the digraph $D\langle A\rangle$ induced by A contains S. Then, if yes, we check if $D-A$ has an F-subdivision. □

4.1. Subdivision of directed cycles

We denote by C_k the directed cycle of length k.

Proposition 16. *For every $k \geq 2$, C_k-*SUBDIVISION *can be solved in time $O(n^k \cdot m)$.*

Proof. For any $k \geq 2$, for k-tuple $(x_1,x_2,\dots,x_k)$, we check if $(x_1,x_2,\dots,x_k)$ is a directed path and if yes if there is a directed (x_k,x_1)-path in $D-\{x_2,\dots,x_{k-1}\}$. There are $O(n^k)$ k-tuples, so this can be done in $O(n^k \cdot m)$ time. □

The running time above is certainly not best possible. For example, when $k=2$ or $k=3$, we can find linear-time algorithms.

Proposition 17. *C_2-*SUBDIVISION *can be solved in linear time.*

Proof. A subdivision of the directed 2-cycle is a directed cycle. Hence a digraph has a C_2-subdivision if and only if it is not acyclic. Since one can check in linear time if a digraph is acyclic or not [1, Section 2.1], C_2-SUBDIVISION is linear-time solvable. □

Proposition 18. *C_3-*SUBDIVISION *can be solved in linear time.*

Proof. Let D be a digraph. If D has no directed 2-cycles, then D contains a C_3-subdivision if and only if it is not acyclic, which can be tested in linear time.

Assume now that D has some directed 2-cycles. Let H be the graph with vertex set $V(D)$ and edge-set $\{xy \mid (x,y,x) \text{ is a 2-cycle of } D\}$. The graph H can be constructed in linear time. We first check, in linear time, if H contains a cycle. If H contains a cycle, then it has length at least 3 and any if its two directed orientations is a directed cycle in D, so we return such a cycle, certifying that D is a 'yes'-instance.

If not, then H is a forest. If there is any single arc uv (an arc which is not part of a 2-cycle) in D such that both u and v belong to the same connected component of H, then it is easy to produce a directed cycle of length at least 3 in D (following a path from u to v in H) so we may assume that all single arcs go between different components in H. Now it is easy to see that D contains a cycle of length at least 3 if and only if the digraph obtained by contracting (into a vertex) each connected component of H in D has a directed cycle. In case we find such a cycle, we can easily reproduce a directed cycle of length at least 3 in D. □

If k is not fixed but specified in the input, it is NP-complete to decide if a digraph has a directed cycle of length k because the Hamiltonian directed cycle is a particular case of it. Gabow and Nie proved that it is FPT to decide if a graph has a cycle of length at least k.

Theorem 19 (Gabow and Nie [11, 12])**.** *One can decide in $O(k^{3k} \cdot n \cdot m)$ time whether a digraph contains a directed cycle of length at least k.*

Problem 20. For any fixed k, can we solve C_k-SUBDIVISION in linear time? In other words, does there exists a computable function f such that one can decide in $O(f(k)(n+m))$ time whether a digraph contains a directed cycle of length at least k?

5. Polynomial-time solvable problems via flows

Recall that two paths are *internally disjoint* if they have no internal vertices in common. For any fixed k, there exist algorithms running in linear time that, given a digraph D and two distinct vertices x and y, returns k internally disjoint directed (x,y)-paths in D if some exist, or returns 'no' otherwise. Indeed, in such a particular case, any flow algorithm like Ford–Fulkerson algorithm for example, performs at most k incrementing-path searches, because it increments the flow by 1 each time, and we stop when the flow has value k, or if we find a cut of size less than k, which by Menger's Theorem certifies that there do not exist k internally disjoint directed (x,y)-paths . Moreover each incrementing-path search consists in a search (usually Breadth-First Search) in an auxiliary digraph of the same size, and so is done in linear time. For more details, we refer the reader to the book of Ford and Fulkerson [**?**] or Chapter 7 of [5]. We call such an algorithm a *Menger algorithm*.

5.1. Subdivision of spindles

A $(k_1,\dots,k_p)$-spindle is the union of p pairwise internally disjoint directed (a,b)-paths $P_1,\dots,P_p$ of respective length $k_1,\dots,k_p$. Vertex a is said to be the *tail* of the spindle and b its *head*.

Proposition 21. *If F is a spindle, then F-*SUBDIVISION *can be solved in $O\left(n^{|V(F)|}(n+m)\right)$ time.*

Proof. Let F be a spindle with tail a and head b. Let $a_1,\dots,a_p$ be the out-neighbours of a in F. An F-subdivision may be seen as an F-subdivision in which only the arcs aa_i, $1 \le i \le p$ are subdivided. The following algorithm takes advantage of this property.

Let D be a digraph. For each pair (S,a') where S is a set of $|V(F)|-1$ vertices and a' a vertex of $D-S$, we first enumerate all the possible subdigraphs of $D\langle S\rangle$ isomorphic to $F-a$ with $a'_1,\dots,a'_p$ corresponding to $a_1,\dots,a_p$. We then check if, in $D-(S\setminus\{a'_1,\dots,a'_p\})$, there exist p internally disjoint directed paths P_i, $1\le i\le p$, each P_i starting in a' and ending in a'_i. This can be done using a Menger algorithm. Clearly, this algorithm decides if there is an F-subdivision in D. There are $O(n^{|V(F)|})$ possible pairs (S,a'), and for each of them we run at most $(|V(F)|-1)!$ times a Menger algorithm. Since such an algorithm runs in linear time, the time complexity of the above algorithm is $O\left(n^{|V(F)|}(n+m)\right)$. □

The complexity given in Proposition 21 is certainly not optimal. For example, it can be improved for spindles with paths of small lengths.

Proposition 22. *If F is a $(k_1,\dots,k_p)$-spindle and $k_i\le 2$ for all $1\le i\le p$, then F-*SUBDIVISION *can be solved in $O(n^2(n+m))$ time.*

Proof. If some of the k_i, say k_1, equals 1, then finding an F-subdivision is equivalent to find p internally disjoint directed paths from some vertex a to some other vertex b, which by Menger's theorem is equivalent to check that the connectivity from a and b is at least p. For any pair (a,b), this can be done in linear time by a Menger algorithm.

If $k_i=2$ for all $1\le i\le 2$, then finding an F-subdivision is equivalent to find p internally disjoint directed paths of length at least two from some vertex a to some other vertex b. Such paths exist if and only if in $D\setminus ab$ there are p internally disjoint (a,b)-paths. For any pair (a,b), this can be checked in linear time by a Menger algorithm. □

A natural question is to ask about the complexity of deciding if a digraph contains a subdivision of a spindle, when the spindle is no more fixed but specified in the input.

Proposition 23. *The following problem is NP-complete*
SPINDLE-SUBDIVISION
Input: A spindle F and a digraph D.
Question: Does D contain a subdivision of F?

Proof. Reduction from the (undirected) Hamiltonian cycle problem.

Let G be an undirected graph. Let $D(G)$ be the symmetric digraph associated to G, that is D is the digraph obtained from G by replacing every edge uv by the two arcs uv and vu. Let F be any spindle of the same order as G (and $D(G)$). For order reason, the digraph contains an F-subdivision if and only if it contains F as a subgraph, and thus if and only if G has a Hamiltonian cycle. □

In view of Proposition 23, one could ask whether it is possible to solve SPINDLE-SUBDIVISION in $f(|V(F)|) \times n^c$ time, where f is a computable function and c an absolute constant? This may be formulated in FPT setting as follows.

Problem 24. Is the following problem fixed-paramater tractable?
PARAMETERIZED SPINDLE-SUBDIVISION
Input: A spindle F and a digraph D.
Parameter: $|V(F)|$.
Question: Does D contain a subdivision of F?

There are many other digraphs that can be solved in the same way as spindles using a Menger algorithm. It is in particular the case of any oriented tree T such that there is a vertex r of in-degree 0 such that $T - r$ is the disjoint union of spiders. For such a tree, T-SUBDIVISION can be solved in $O\left(n^{|V(T)|-1}(n+m)\right)$ time. The polynomial-time solvability of F-SUBDIVISION of some other digraphs may also be established by using a Menger algorithm in a slightly different way as we show in the next two subsections.

5.2. Subdivision of windmills

A *cycle windmill* is a digraph obtained from disjoint directed cycles by taking one vertex per cycle and identifying all of these. This vertex will be called the *axis* of the windmill.

Theorem 25. *If W is a cycle windmill, then W-*SUBDIVISION *can be solved in $O(n^{|V(W)|}(n+m))$ time.*

Proof. Suppose W is a windmill with axis o and cycle lengths $a_1, a_2, \dots, a_p$. To check whether a given digraph $D = (V, A)$ contains a subdivision of W with axis at the vertex x we do the following (until success or all subsets have been tried): for all choices of disjoint ordered subsets $X_1, X_2, \dots, X_p$ of V such that $X_i = \{v_{i,1}, \dots, v_{i,a_i-1}\}$, $i = 1, 2, \dots, p$ check whether $Q_i = xv_{i,1}v_{i,2}\dots v_{i,a_i-1}$ is a directed (x, v_{i,a_i-1})-path. If this holds for all i, then delete all the vertices of $X_i - v_{i,a_i-1}$, $i = 1, 2, \dots, p$ and check whether the resulting digraph contains internally disjoint paths $P_1, P_2, \dots, P_p$ where P_i is a path from v_{i,a_i-1} to x using a Menger algorithm. If these paths exist, then return the desired subdivision of W formed by the union of $Q_1, Q_2, \dots, Q_p, P_1, P_2, \dots, P_p$. Otherwise continue to the next choice for $X_1, X_2, \dots, X_p$. Since the size of $X_1 \cup X_2 \cup \ldots \cup X_p$ is $|V(W)|-1$, there are $O(n^{|V(W)|-1})$ choices for it, and there are n choices for x, hence the algorithm runs $O(n^{|V(W)|})$ times a Menger algorithm. Since a Menger algorithm runs in linear time, the overall complexity is $O(n^{|V(W)|}(n+m))$. $\square$

Clearly, given as input a windmill W and a digraph D, deciding if D contains a W-subdivision is NP-complete because the Hamiltonian directed cycle problem is a particular case of it. Theorem 25 tells us that this problem parameterized by $|W|$ is in XP. But is it fixed-parameter tractable?

Problem 26. Is the following problem fixed-paramater tractable?
CYCLE-WINDMILL SUBDIVISION
Input: A cycle windmill W and a digraph D.
Parameter: $|V(W)|$.
Question: Does D contain a subdivision of W?

5.3. Subdivision of palm trees

A *palm tree* is a dumbbell, whose left and right plates are spiders, and whose bar is a directed path of length one. Observe that in a palm tree, the two clips must be the bodies of the spiders. A *palm grove* is a disjoint union of palm trees. For example, the two graphs A and B depicted Figure 1 are palm groves.

By Theorem 12(c), if F is a palm grove having two palm trees whose left spiders are not isomorphic and whose right spiders are not isomorphic, then F-SUBDIVISION is NP-complete. We shall now prove that it is indeed the only hard case. Observe that if a digraph contains a subdivision of a palm tree, then it contains a subdivision of this palm tree such that the only subdivided arc is the bar.

Theorem 27. *Let F be a palm grove. Then F-SUBDIVISION is polynomial-time solvable if and only if all its left spiders are isomorphic or all its right spiders are isomorphic.*

Proof. If there are two left spiders that are not isomorphic and there are two right spiders that are not isomorphic, then there exist two palm trees such that their left spiders are not isomorphic and their right spiders are not isomorphic. Then, by Theorem 12-(c), F-SUBDIVISION is NP-complete.

Assume now that all the right spiders are isomorphic to a spider R. Let $L_1, \ldots, L_p$ be the left spiders (possibly some of them are isomorphic). We shall decribe an algorithm to solve F-SUBDIVISION.

Let D be a digraph. By the above remark, if D contains an F-subdivision, then it contains an F-subdivision such that only the bars of the palm trees are subdivided. Hence we look for such a subdivision. Observe that such a subdivision is the disjoint union of copies of each of the L_i, $1 \le i \le p$ and p copies of R together with p disjoint directed paths from the bodies of the copies of the L_i to the bodies of the p copies of R. Hence to decide if D contains an F-subdivision, we try all possibilities for the disjoint union of spiders L_i, $1 \le i \le p$, and p spiders R and for each possibility we check via a Menger algorithm if there are disjoints directed paths from the bodies of the L_i to the bodies of the copies of R.

Formally, the algorithm is the following. For each set of distinct vertices $\{u_1, \ldots u_p, v_1, \ldots, v_p\}$ of D and family of disjoints subsets $\{U_1, \ldots, U_p, V_1, \ldots, V_p\}$ of D such that for $1 \le i \le p$, $u_i \in U_i$ and $v_i \in V_i$, we check if for all i, $D\langle U_i \rangle$ (resp. V_i) contains a spider isomorphic to L_i (resp. R) with body u_i (resp. v_i). If not we proceed to the next case. If yes, we check if there are p disjoint directed paths from $\{u_1,\dots,u_p\}$ to $\{v_1,\dots,v_p\}$ in the digraph $D\setminus(\bigcup_{i=1}^{p}(U_i\cup V_i)\setminus\{u_i,v_i\})$ via a Menger algorithm. If there are such paths, the union of them with the spiders is an F-subdivision and we return it. If such paths do not exist, we proceed to the next case.

The number of possible cases is $O(n^{|V(F)|})$ and each run of the Menger algorithm can be done in linear time. Hence the complexity of the algorithm is $O(n^{|V(F)|}(n+m))$. □

6. The Fork Problem and bispindles

A *fork* with *bottom vertex a*, top vertices b and c and centre t is a digraph in which

- a, b and c are distinct, and t is distinct from b and c (but possibly equal to a),
- every vertex except a has in-degree 1 and a has in-degree 0, and
- all vertices except b, c and t have out-degree 1 and b and c have out-degree 0 and t has out-degree 2.

The following problem is very useful, as it can be efficiently solved.

FORK
Input: A digraph D and three distinct vertices a, b and c.
Question: Does D contain a fork with bottom vertex a and top vertices b and c?

Lemma 28. FORK *can be solved in linear time.*

Proof. Assume that a digraph D contains a fork with bottom vertex a and top vertices b and c. Then, clearly, there are a directed (a,b)-path in $D-c$ and a directed (a,c)-path in $D-b$.

We claim that this necessary condition is also sufficient. Indeed, assume that there is a directed (a,b)-path P in $D-c$ and a directed (a,c)-path Q in $D-b$. Let t be the last vertex on P which also belongs to Q. Such a vertex exists because a is in P and Q. Then the union of P and $Q[t,c]$ is the desired fork.

Since one can decide in linear time if there is a directed (u,v)-path in a digraph, FORK can be solved in linear time. □

The $(k_1,\dots,k_p;l_1,\dots,l_q)$*-bispindle*, denoted $B(k_1,\dots,k_p;l_1,\dots,l_q)$, is the digraph obtained from the disjoint union of a $(k_1,\dots,k_p)$-spindle with tail a_1 and head b_1 and a $(l_1,\dots,l_q)$-spindle with tail a_2 and head b_2 by identifying a_1 with b_2 into a vertex a, and a_2 with b_1 into a vertex b. The vertices a and b are called, respectively, the *left node* and the *right node* of the bispindle. The directed (a,b)-paths are called the *forward paths*, while the directed (b,a)-paths are called the *backward paths*.

We say that $(P_1,\dots,P_p;Q_1,\dots,Q_q)$ is a $(k_1,\dots,k_p;l_1,\dots,l_q)$-bispindle if, for each $1\le i\le p$, P_i is a directed (c,d)-path of length k_i, for each $1\le j\le q$, Q_j is a directed (d,c)-path of length l_j and the union of the P_i and Q_j is $B(k_1,\dots,k_p;l_1,\dots,l_q)$.

Let F be a bispindle with p forward paths and q backward paths. Consider the big paths multidigraph $BP(F)$. By Remark 9, we get the following.

Proposition 29. *Let F be a bispindle with p forward paths and q backward paths. If $p \geq 1$, $q \geq 1$, and $p+q \geq 4$, then F-*SUBDIVISION *is NP-complete.*

On the other hand, if F has no backward paths or exactly one backward path and one forward path, then it is a spindle or a directed cycle, respectively. In both cases, F-SUBDIVISION can be solved in polynomial time as shown in Subsections 5.1 and 4.1, respectively.

We now show using Lemma 28 that, in the remaining cases, that is when F is a bispindle with two forward paths and one backward path, F-SUBDIVISION is polynomial-time solvable.

Theorem 30. *If F is a bispindle with two forward paths and one backward path, then F-*SUBDIVISION *can be solved in $O(n^{|F|+1}(n+m))$ time.*

Proof. Let a be the left node of F and let b and c be its two out-neighbours in F.

For every subset S of $|F|$ vertices, we check if $D\langle S\rangle$ contains a copy of $F \setminus \{ab, ac\}$ with a', b', c' corresponding to a, b, c, respectively. Then we check in $D - (S \setminus \{a', b', c'\})$ if there is a fork with bottom vertex a' and top vertices b' and c'.

Since there are $O(n^{|F|})$ possible set S and FORK can be solved in linear time by Lemma 28, our algorithm runs in $O(n^{|F|+1}(n+m))$ time. □

The complexity given in Theorem 30 is certainly not best possible. Similarly to Proposition 23, one shows that given a digraph D and a bispindle F (with two forward paths and one backward path), deciding if D contains an F-subdivision is NP-complete. It is again natural to ask if it is FPT when parameterized by $|F|$.

Problem 31. Is the following problem fixed-paramater tractable?
PARAMETERIZED BISPINDLE-SUBDIVISION
Input: A bispindle F with two forward paths and one backward path and a digraph D.
Parameter: $|V(F)|$.
Question: Does D contain a subdivision of F?

In the next section, we give faster algorithms to solve $B(1,2;1)$- , $B(1,2;2)$- and $B(1,3;1)$-SUBDIVISION.

7. Polynomial-time solvable problems via handle decomposition

Let D be a strongly connected digraph. A *handle* h of D is a directed path $(s, v_1, \dots, v_\ell, t)$ from s to t (where s and t may be identical) such that:

- for all $1 \leq i \leq \ell$, $d^-(v_i) = d^+(v_i) = 1$, and
- the digraph $D \setminus h$ obtained from D by *suppressing* h, that is removing the arcs and the internal vertices of h, is strongly connected.

The vertices s and t are the *endvertices* of h while the vertices v_i are its *internal vertices*. The vertex s is the *origin* of h and t its *terminus*. The *length* of a handle is the number of its arcs, here $\ell+1$. A handle of length one is said to be *trivial*.

Given a strongly connected digraph D, a *handle decomposition* of D starting at $v \in V(D)$ is a triple $(v, (h_i)_{1\leq i\leq p}, (D_i)_{0\leq i\leq p})$, where $(D_i)_{0\leq i\leq p}$ is a sequence of strongly connected digraphs and $(h_i)_{1\leq i\leq p}$ is a sequence of handles such that:

- $V(D_0) = \{v\}$,
- for $1 \leq i \leq p$, h_i is a handle of D_i and D_i is the (arc-disjoint) union of D_{i-1} and h_i, and
- $D = D_p$.

A handle decomposition is uniquely determined by v and either $(h_i)_{1\leq i\leq p}$, or $(D_i)_{0\leq i\leq p}$. The number of handles p in any handle decomposition of D is exactly $|A(D)| - |V(D)| + 1$. The value p is also called the *cyclomatic number* of D. Observe that $p = 0$ when D is a singleton and $p = 1$ when D is a directed cycle.

A handle decomposition $(v, (h_i)_{1\leq i\leq p}, (D_i)_{0\leq i\leq p})$ is *nice* if all handles except the first on h_1 have distinct endvertices. The following proposition is well-known (see [5] Theorem 5.13).

Proposition 32. *Every robust digraph admits a nice handle decomposition.*

7.1. Subdivision of the lollipop

The *lollipop* is the digraph L with vertex set $\{x, y, z\}$ and arc set $\{xy, yz, zy\}$.

Proposition 33. *L-SUBDIVISION can be solved in linear time.*

Proof. If D contains a strong component of cyclomatic number greater than 1, then it contains a lollipop. Indeed, the smallest directed cycle C in the component is induced and is not the whole strong component. Hence there must be a vertex v dominating a vertex of C thus forming a lollipop-subdivision.

If not, then all the strong components are cycles. Thus D contains a lollipop if and only if one of its component is a directed cycle and is not an initial strong component (i.e some arc is entering it).

All this can be checked in linear time. □

7.2. Faster algorithm for subdivision of bispindles

In this subsection, using handle decomposition, we show algorithms to solve $B(1,2;1)$- , $B(1,2;2)$- and $B(1,3;1)$-SUBDIVISION, whose running time is smaller than the complexity of Theorem 30.

Recall that a digraph D is *robust* if it is strongly connected and $UG(D)$ is 2-connected. The *robust components* of a digraph are its robust subdigraphs which are maximal by inclusion.

Because bispindles are robust, a subdivision S of a bispindle is also robust, and if a digraph D contains S, then S must be in a robust component of D. Finding the robust components of a digraph can be done in linear time, by finding the strong components and the 2-connected components of the underlying graphs of these. Therefore one can restrict our attention to subdivision of bispindles in robust digraphs.

7.2.1. Subdivision of the $(1,2;1)$*-bispindle*

Observe that a subdivision of the $(1,2;1)$-bispindle has cyclomatic number two. Conversely, one can easily check that every robust digraph of cyclomatic number 2 is a subdivision of the $(1,2;1)$-bispindle. Hence, we have the following.

Proposition 34. *A digraph contains a subdivision of the* $(1,2;1)$*-bispindle if and only if one of its robust components has cyclomatic number at least two.*

Corollary 35. $B(1,2;1)$-SUBDIVISION *can be solved in linear time.*

Proof. Finding the robust components can be done in linear time and computing the cyclomatic number of all of them in linear time as well. □

7.2.2. Subdivision of the $(1,2;2)$*-bispindle*

In this subsection, we show that $B(1,2;2)$-SUBDIVISION is polynomial-time solvable. In order to prove it, we characterize the robust digraphs that contain no $B(1,2;2)$-subdivision. Let us now describe the family $\mathcal{F}_{1,2;2}$. A *double ring* is a digraph obtained from an undirected cycle by replacing every edge by two arcs, one in each direction. See Figure 2. A digraph G is in $\mathcal{F}_{1,2;2}$ if it is a double ring or it can be obtained from a $(k_1,\dots,k_p)$-spindle S, $p \geq 1$, with tail x and head y as follows. Add the arc yx and possibly some *back arcs*, that are, arcs vu such that $uv \in A(S)$, so that the unique directed (y,x)-path is the arc yx. See Figure 3.

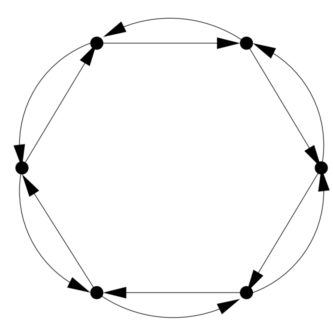

Figure 2: The double ring of order 6.

Theorem 36. *A robust digraph D contains a* $B(1,2;2)$*-subdivision if and only if* $D \notin \mathcal{F}_{1,2;2}$.

Proof. Let us first prove that if $D \in \mathcal{F}_{1,2;2}$, then it contains no $B(1,2;2)$-subdivision. Suppose for a contradiction, that there is such a subdivision S. Let a and b be the left and right nodes of a subdivision of S. Then the connectivity between a and b is at least 2 in one direction. So, by construction, either $(a,b) = (x,y)$, or (a,b) is such that ab is a back arc. But, in both cases, the unique directed (b,a)-path is (b,a) which has length less than 2, this is a contradiction.

Suppose now that $D \notin \mathcal{F}_{1,2;2}$. Let us prove that it contains a $B(1,2;2)$-subdivision. Let $(v,(h_i)_{1\leq i\leq p},\ (D_i)_{0\leq i\leq p})$ be a nice handle decomposition of D, and let i be the smallest positive integer such that $D_i \notin \mathcal{F}_{1,2;2}$. Clearly $i \geq 2$ because every directed cycle is in $\mathcal{F}_{1,2;2}$. Then D_{i-1} is in $\mathcal{F}_{1,2;2}$.

We shall prove that D_i contains a $B(1,2;2)$-subdivision, and thus so does D.

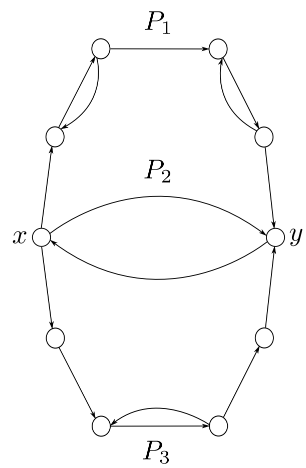

Figure 3: A digraph in $\mathcal{F}_{1,2;2}$, which is not a double ring

Suppose first that D_{i-1} is the double ring associated to a cycle $x_1x_2\dots x_nx_1$. Without loss of generality, we may assume that the origin of h_i is x_1 and its terminus x_j for $2 \le j \le n$. Then $(h_i, x_1\dots x_j; x_j\dots x_nx_1)$ is a $B(1,2;2)$-subdivision. (Observe that if $j=2$, then h_i must have length at least 2, since there are no multiple arcs.)

Suppose now that D_{i-1} is not a double ring. Let x and y be the two vertices of D_{i-1} as in the definition of $\mathcal{F}_{1,2;2}$. In other words, D_{i-1} is obtained from a spindle $(P_1, P_2, \dots, P_k)$ with tail x and head y by adding yx and some back arcs. We distinguish several cases according to the possible locations of the tail u and head v of h_i. Observe that $(u,v) \neq (x,y)$ for otherwise D_i would be in $\mathcal{F}_{1,2;2}$.

(i) $u=y$ and $v=x$. Since yx is an arc of D_{i-1} and there are no multiple arcs, the handle h_i has length at least 2. Hence $(yx, h_i; P_1)$ is a $B(1,2;2)$-subdivision.

(ii) $u=x$ and v is an internal vertex of some P_j. Since there are no multiple arcs, one of the two (x,v)-paths h_i and $P_j[x,v]$ has length at least 2. Hence $(h_i, P_j[x,v]; P_j[v,y]x)$ is a $B(1,2;2)$-subdivision.

(iii) $v=y$ and u is an internal vertex of some P_j. This case is similar to the previous one by directional symmetry.

(iv) $u=y$ and v is an internal vertex of some P_j. Then $(h_i, yP_j[x,u]; P_j[u,y])$ is a $B(1,2;2)$-subdivision. Note that, since $D_i \notin \mathcal{F}_{1,2;2}$, at least one of h_i and $P_j[u,y]$ has length more than one.

(v) $v=x$ and u is an internal vertex of some P_j. This case is similar to the previous one by directional symmetry.

(vi) u and v are internal vertices of the same P_j and u precedes v on P_j. Since there are no multiple arcs, one of the two (u,v)-paths h_i and uP_jv has length at least 2. Hence $(h_i, P_j[u,v]; P_j[v,y]xP_j[x,v])$ is a $B(1,2;2)$-subdivision.

(vii) u and v are internal vertices of the same P_j and v precedes u on P_j. If h_i is of length one, then in D_i all the back arcs associated to arcs of P_j exist, for otherwise D_i would be in $\mathcal{F}_{1,2;2}$. These arcs induce a directed (y,x)-path R_j of length at least 2. Moreover, $k \ge 2$, for otherwise D_i would be in $\mathcal{F}_{1,2;2}$ with y as left node and x as right node. If $k=2$ and the path of $\{P_1,P_2\} \setminus \{P_j\}$ was of length one, then D_i would be a double ring. Hence, there is $j' \neq j$ such that $P_{j'}$ has length at least two, and we have the $B(1,2;2)$-subdivision $(yx, R_j; P'_j)$

(viii) u is an internal vertex of P_j, v is an internal vertex of $P_{j'}$ and $j \neq j'$. Then $(P_j[u,y], h_i P_{j'}[v,y]; yP_j[x,u])$ is a $B(1,2;2)$-subdivision.

□

Corollary 37. $B(1,2;2)$-SUBDIVISION *can be solved in linear time.*

7.2.3. Subdivision of the $(1,3;1)$*-bispindle*

Observe that there is a C_4 in a $(1,3;1)$-bispindle. So, a digraph D that has no directed cycle of length greater than 3 contains no $B(1,3;1)$-subdivision.

Let D be a robust digraph and $C = (v_1, \dots, v_\ell, v_1)$ a directed cycle in D. A handle decomposition $(v, (h_i)_{1 \leq i \leq p}, (D_i)_{0 \leq i \leq p})$ is said to be *C-bad* if

(i) $D_1 = C$;

(ii) for all $i \geq 2$, h_i has length 1 or 2, its endvertices are on C and the distance between the origin and the terminus of h_i around C is 2.

(iii) If h_i is a $(v_k, v_k + 2)$-path and h_j is a $(v_{k-1}, v_k + 1)$-path (indices are taken modulo ℓ), then these two handles have length 1.

(iv) If $\ell \geq 5$, there no k such that (v_{k-2}, v_k), (v_{k-1}, v_{k+1}) and (v_k, v_{k+2}) are handles.

The notion of C-bad handle decomposition plays a crucial role for finding $B(1,3;1)$-subdivision as shown by the next two lemmas.

Lemma 38. *Let D be a digraph and C a directed cycle in D of length at least* 4*. Then one of the following holds:*

- *D contains a* $B(1,3;1)$*-subdivision,*
- *C is not a longest circuit in D, or*
- *D has a C-bad handle decomposition.*

Proof. Set $C = (v_1, \dots, v_\ell, v_1)$. Let $\mathcal{H} = (v, (h_i)_{1 \leq i \leq p}, (D_i)_{0 \leq i \leq p})$ be a nice handle decomposition of D such that $D_1 = C$.

If $\mathcal{H}$ is not C-bad, then let k be the largest integer such that $\mathcal{H}_k = (v, (h_i)_{1 \leq i \leq k}, (D_i)_{0 \leq i \leq k})$ is a C-bad handle decomposition. One of the following occurs:

(i) the origin s_{k+1} of h_{k+1} is the internal vertex of some h_i, $i \geq 2$. Since $\mathcal{H}_k$ is C-bad, then necessarily $h_i = (s_i, s_{k+1}, t_i)$, and there is a directed path (s_i, v_i, t_i) of length 2 in C. Let t_{k+1} be the terminus of h_{k+1}. If t_{k+1} is on C, we set $h^* = h_{k+1}$ and $t^* = t_{k+1}$. If not, then t_{k+1} has an out-neighbour t^* on C and we let h^* be the concatenation of h_{k+1} and (t_{k+1}, t^*). In both cases, h^* is a directed (s_{k+1}, t^*)-path with no internal vertices in C. If $t^* = v_i$, then $h^* \cup (C \setminus \{s_i v_i\}) \cup (s_i, s_{k+1})$ is a directed cycle longer than C. If $t^* = s_i$, then $(C \cup h^* \cup (s_i, s_{k+1})) - v_i$ is a $B(1,3;1)$-subdivision with right node s_i and left node s_{k+1}. If $t^* = t_i$, then $C[t_i, s_i] \cup h^*$ is a directed cycle longer than C because in that case h^* has length at least 2. If $t^* \notin \{s_i, t_i, v_i\}$, then $C \cup h^* \cup (s_i, s_{k+1})$ is a $B(1,3;1)$-subdivision with left node s_i and right node t^*.

(ii) the terminus of h_{k+1} is the internal vertex of some h_i, $i \geq 2$. We get the result in a similar way to the preceding case.

(iii) h_{k+1} has length greater than 2 and its two endvertices are on C. Then the union of C and h_{k+1} is a $B(1,3;1)$-subdivision.

(iv) $h_{k+1} = (s,t)$ with s,t and $C[s,t]$ has length at least 3. Then $C \cup (s,t)$ is a $B(1,3;1)$-subdivision with right node s and left node t.

(v) h_{k+1} is one of the two handles h and h', where h is a (v_{k-1}, v_{k+1})-handle and h' is a (v_k, v_{k+2}) for some k, and one of h and h' has length two. If h has length two, say (v_{k-1}, x_1, v_{k+1}), then the union of $(v_{k-1}, v_k) \cup h'$, $(v_{k-1}, x_1, v_{k+1}, v_{k+2})$ and $C[v_{k+2}, v_{k-1}]$ form a $B(1,3;1)$-subdivision. If h' has length two, say $h' = (v_k, x_2, v_{k+2})$, then the union of $h \cup (v_{k+1}, v_{k+2})$, $(v_{k-1}, v_k, x_2, v_{k+2})$ and $C[v_{k+2}, v_{k-1}]$ form a $B(1,3;1)$-subdivision.

(vi) h_{k+1} is one of the three handles (v_{k-2}, v_k), (v_{k-1}, v_{k+1}), (v_k, v_{k+2}) for some k and $p \geq 5$. In this case, the union of $(v_{k-2}, v_{k-1}, v_{k+1}, v_{k+2})$, (v_{k-2}, v_k, v_{k+2}) and $C[v_{k+2}, v_{k-2}]$ form a $B(1,3;1)$-subdivision.

□

Lemma 39. *Let D be a robust digraph and C a directed cycle in D of length at least 4. If D has a C-bad handle decomposition, then it does not contain any $B(1,3;1)$-subdivision.*

Proof. By induction on the number p of handles of the handle decomposition, the result holding trivially if $p = 1$.

Set $C = (v_1, \dots, v_\ell, v_1)$ and let $\mathcal{H} = (v, (h_i)_{1 \leq i \leq p}, (D_i)_{0 \leq i \leq p})$ be a C-bad handle decomposition of D.

By the induction hypothesis D_{p-1} does not have any $B(1,3;1)$-subdivision.

Suppose, by way of contradiction, that D_p contains a $B(1,3;1)$-subdivision S. Necessarily, h_p is a subdigraph of S. Free to rename, we may assume that v_1 and v_3 are the origin and the terminus, respectively, of h_p. If v_2 is not in S, then replacing h_p with (v_1, v_2, v_3) in S, we obtain a $B(1,3;1)$-subdivision contained in D_{p-1}, a contradiction. Hence $v_2 \in V(S)$. By the conditions (iii) and (iv) of a C-bad handle decomposition, there cannot be both a handle ending at v_2 and a handle starting at v_2. By directional symmetry, we may assume that v_2 has in-degree one, and so $v_1 v_2 \in A(S)$, and v_1 is the left node of S. Now, $v_2 v_3$ is not an arc of S, for otherwise v_3 will be the right node of S, and the two directed (v_1, v_3)-paths in S have length at most 2, a contradiction. But, in S, there is an arc leaving v_2, it must be in a handle, and so by (iv) and (ii) of the definition of C-bad, this arc must be $v_2 v_4$. Again by (iii) of the definition of C-bad, there is no arc leaving v_3 except $v_3 v_4$. Hence $v_3 v_4 \in A(S)$. Then v_4 is the right node of S, and the two directed (v_1, v_4)-paths in S have length 2, a contradiction. □

Theorem 40. $B(1,3;1)$-SUBDIVISION *can be solved in $O(n \cdot m)$ time.*

Proof. Given a digraph D, we compute the robust components of D and solve the problem separately on each of them.

For each robust component, we first search for a directed cycle C_0 of length at least 4. This can be done in $O(n \cdot m)$ time by Theorem 19. If there is no such cycle, then we return 'no'. If not, then we build a handle decomposition starting from $C := C_0$. Each time, we add a new handle, one can mimick the proof of Lemma 38, we either find a $B(1,3;1)$-subdivision which we return, or a C-bad handle decomposition, or a directed cycle C' longer than the current C. Observe that in this case, it is easy to derive a C'-bad handle decomposition containing the vertices added so far from the C-bad one. This can be done in $O(n \cdot m)$ time because an arc has to be considered only when it is added in a handle, and we just need to keep a set of at most m handles.

At the end of this process, if no $B(1,3;1)$-subdivision has been returned, we end up with a C-bad decomposition of D. So, by Lemma 39, D has no $B(1,3;1)$-subdivision, and we can proceed to the next robust component, or return 'no' if there is none. □

8. Classes of digraphs for which F-SUBDIVISION is polynomial-time solvable for all F

Lemma 41. *Let $\mathcal{D}$ be a class of digraphs which is closed under the operation which takes as input a digraph $D \in \mathcal{D}$, a bounded set of vertices $x_1, x_2, \dots, x_r \in V(D)$ and integers $i_1, i_2, \dots, i_r, o_1, o_2, \dots, o_r$, all between 0 and r and outputs the digraph D' that is obtained as follows: For $j = 1, 2, \dots, r$ replace x_j and all arcs incident to it by two sets of vertices $I_j = \{v_{j,1}, \dots, v_{j,i_j}\}, O_j = \{w_{j,1}, \dots, w_{j,o_j}\}$ (if $i_j = 0$ or $o_j = 0$ the corresponding set is empty), all possible arcs from $N^-_D(x_j)$ to I_j and from O_j to $N^+_D(x_j)$. If k-LINKAGE is polynomial-time solvable for all fixed k for digraphs in $\mathcal{D}$, then, for each digraph F, F-SUBDIVISION is polynomial-time solvable on digraphs in $\mathcal{D}$.*

Proof. Let F be a digraph with vertex set $\{1, 2, \dots, r\}$ and let D belong to $\mathcal{D}$. It is sufficient to show that we can decide in polynomial time whether a fixed one-to-one mapping of $V(F)$ to $V(D)$ extends to a subdivision of F in D. So we assume below that a one-to-one mapping of $V(F)$ to $V(D)$ is given.

For each vertex $\alpha \in V(F)$, fix an ordering of the arcs entering α and an ordering of the arcs leaving α: We label the $d^-_F(\alpha)$ in-neighbours of α by $i_{\alpha,1}, i_{\alpha,2}, \dots, i_{\alpha,d^-_F(\alpha)}$ and we label the $d^+_F(\alpha)$ out-neighbours of α by $o_{\alpha,1}, o_{\alpha,2}, \dots, o_{\alpha,d^+_F(\alpha)}$. For a given arc $e = \alpha\beta \in A(F)$ this gives two labels $l^+_{\alpha\beta}$ and $l^-_{\alpha\beta}$ (the number it has in α's out-labelling and in β's in-labelling). Given the one-to-one mapping $f : V(F) \to V(D)$ we make a new digraph D_F from D by replacing each vertex $f(\alpha)$, $\alpha \in V(F)$ by two sets $I_{f(\alpha)} = \{i_{\alpha,1}, i_{\alpha,2}, \dots, i_{\alpha,d^-_F(\alpha)}\}$ and $O_{f(\alpha)} = \{o_{\alpha,1}, o_{\alpha,2}, \dots, o_{\alpha,d^+_F(\alpha)}\}$ and joining every in-neighbour x of $f(\alpha)$ in D to every vertex y in $I_{f(\alpha)}$ by an arc $x \to y$ and every vertex p of $O_{f(\alpha)}$ to every out-neighbour q of $f(\alpha)$ in D (it is possible that one of the sets $I_{f(\alpha)}, O_{f(\alpha)}$ is empty in which case we add no arcs corresponding to that set).

Now it is easy to check that f can be extended to a subdivision of F in D if and only if D_F contains vertex-disjoint paths $\{P_{\alpha\beta} \mid \alpha\beta \in A(F)\}$ where $P_{\alpha\beta}$ starts in $o_{\alpha,l^+_{\alpha\beta}}$ and ends in $i_{\beta,l^-_{\alpha\beta}}$. Since D_F is in $\mathcal{D}$ we can check the existence of the desired paths in polynomial time. Doing this for (at most) all possible one-to-one mappings of $V(F)$ to $V(D)$ we can decide in polynomial time (since $|V(F)|$ is constant) whether D contains an F-subdivision. □

Theorem 42 (Fortune, Hopcroft and Wyllie [10])**.** *For every fixed k the k-LINKAGE problem is polynomial-time solvable for acyclic digraphs.*

Clearly the class of acyclic digraphs is closed under the operation given in Lemma 41 and hence we have the following.

Corollary 43 (Fortune, Hopcroft and Wyllie [10])**.** *For every digraph F, F-SUBDIVISION is polynomial-time solvable for acyclic digraphs.*

The algorithm given by Fortune, Hopcroft and Wyllie to solve k-LINKAGE problem has a runinng time in $O(k!n^{k+2})$. Hence a natural question is to ask if it can be solved in $O(f(k)n^c)$ time for some absolute constant c and computable function f. In the FPT setting, it can be phrased as follows.

Problem 44. Is the following parameterized problem FPT?
PARAMETERIZED ACYCLIC k-LINKAGE
Input: An acyclic digraph D and $2k$ distinct vertices $x_1, x_2, \ldots, x_k, y_1, y_2, \ldots, y_k$.
Parameter: k.
Question: Is there a k-linkage from $(x_1, x_2, \ldots, x_k)$ to $(y_1, y_2, \ldots, y_k)$ in D?

Theorem 45 (Johnson et al. [14])**.** *For every fixed k, k-LINKAGE is polynomial-time solvable on digraphs of bounded directed tree-width.*

We will not give the definition of directed tree-width here as it is rather technical, but it suffices to say that the class of digraphs with bounded directed tree-width is closed on the operation of Lemma 41 so we have.

Theorem 46 (Johnson et al. [14])**.** *For every digraph F, F-SUBDIVISION is polynomial-time solvable on digraphs of bounded directed tree-width.*

Theorem 47 (Chudnovsky et al. [6])**.** *For any digraph F, F-SUBDIVISION is polynomial-time solvable when restricted to the class of tournaments.*

Let $D = (V, A)$ be a digraph. We say that $W \subseteq V$ *guards* $V' \subseteq V$ in D if $N^+(V') \subseteq W$, that is, all out-neighbours of V' are in W. A *DAG-decomposition* of a digraph D is a pair (H, χ) where H is an acyclic digraph and $\chi = \{W_h : h \in V(H)\}$ is a family of subsets of $V(D)$ satisfying the following three properties:

(i) $V(D) = \bigcup_{h \in V(H)} W_h$,

(ii) for all $h, h', h'' \in V(H)$, if h' lies on a directed (h, h'')-path, then $W_h \cap W_{h''} \subseteq W_{h'}$, and

(iii) if $(h, h') \in A(H)$, then $W_h \cap W_{h'}$ guards $W_{\geq h'} \setminus W_h$, where $W_{\geq h'}$ is the union of all $W_{h''}$ for which there exists a directed (h', h'')-path in H.

The *width* of a DAG-decomposition (H, χ) is $\max_{h \in V(H)} |W_h|$. The *DAG-width* of a digraph D ($\mathrm{dagw}(D)$) is the minimum width over all possible DAG-decompositions of D. It is easy to see that a digraph D is acyclic if and only if it has DAG-width 1 (and then we can use D itself as H).

Theorem 48 (Berwanger et al. [4], Johnson et al. [14])**.** *For every fixed k, k-LINKAGE is polynomial-time solvable on digraphs of bounded DAG-width.*

Digraphs of bounded DAG-width are closed under the operation in Lemma 41 so we have.

Corollary 49. *For any digraph F, F-SUBDIVISION is polynomial-time solvable when restricted to the class of digraphs of bounded DAG-width.*

More generally, the property of having an F-subdivision can be defined in MSO1 monadic second order logic with vertex-set quantifications) and so can be solved in polynomial time on the class of digraphs with bounded directed clique-width. If F is not fixed, but specified in the input, it can also be solved in FPT-time when parameterized by $V(F)|$. See Theorem 1.24 of [9].

A *feedback vertex set* or *cycle transversal* in a digraph D is a set of vertices S such that $D-S$ is acyclic. The minimum number of vertices in a cycle transversal of D is the *cycle-transversal number* and is denoted by $\tau(D)$.

Corollary 50. *For any digraph F, F-SUBDIVISION is polynomial-time solvable when restricted to the class of digraphs with bounded cycle-transversal number.*

Proof. Let X be a cycle-transversal of D. Then $D' = D-X$ is acyclic and it is easy to see that D has DAG-width at most X, since we can take $H = D'$ and $W_h = \{h\} \cup X$ for all $h \in V(D')$ to obtain a DAG-decomposition of D whose width is $|X|$. Now the result follows from Corollary 49. □

The maximum number of disjoint directed cycles in a digraph D is called the *cycle-packing number* and is denoted by $\nu(D)$. Clearly, $\nu(D) \leq \tau(D)$. Conversely, proving the so-called Gallai-Younger Conjecture, Reed et al. [17] proved that $\tau(D)$ is bounded above by a function of $\nu(D)$.

Theorem 51 (Reed et al. [17])**.** *For every k, there is an integer $f(k)$ such that every digraph has either k disjoint directed cycles or a feedback vertex set of size at most $f(k)$.*

The function f constructed by Reed at al. [17] grows very quickly. It is a multiply iterated exponential, where the number of iterations is also a multiply iterated exponential. The correct value of $f(2)$ is 3 as shown by McCuaig [16] who also gave a polynomial-time algorithm for finding two disjoint directed cycles in a digraph or showing that it has $\tau(D) \leq 3$.

Corollary 52. *For any digraph F, F-SUBDIVISION is polynomial-time solvable when restricted to the class of digraphs with bounded cycle-packing number.*

9. F-SUBDIVISION for some special classes of digraphs

In this section the focus is on the structure of F rather than the method for solving F-SUBDIVISION or proving it NP-complete. For several of the classes we can provide (almost) complete characterizations in terms of complexity of F-SUBDIVISION .

9.1. Disjoint union of directed cycles

Since C_k-SUBDIVISION can be solved in polynomial time for any fixed k, a natural question is to ask for the complexity of F-SUBDIVISION when F is the disjoint union of directed cycles. This is not a simple problem as can be seen from the observation that a digraph D contains k disjoint directed cycles if and only if it contains an F-subdivision where F is the disjoint union of k directed 2-cycles.

Hence, if F is the disjoint union of k directed 2-cycles, F-SUBDIVISION is equivalent to deciding if $\nu(D) \geq k$ for a given digraph D. Using Theorem 51, Reed et al. [17] proved that this can be done in polynomial time.

Theorem 53 (Reed et al. [17])**.** *For any fixed k, deciding if a digraph D has k disjoint directed cycles is polynomial-time solvable. Equivalently, if F is the disjoint union of directed 2-cycles, then F-SUBDIVISION is polynomial-time solvable.*

Remark 54. Determining $\nu(D)$ is NP-hard. Indeed, given a digraph D and an integer k, deciding whether D has at least k disjoint cycles is NP-complete. See Theorem 13.3.2 and Exercise 13.25 of [1]. As observed in [13], the problem parameterized with k is hard for the complexity class W[1] (this follows easily from the results of [19]). This means that, unless $FPT = W[1]$, there is no algorithm solving the problem with a $f(k) \cdot n^{O(1)}$ running time.

Problem 55. Let F be the disjoint union of p directed cycles of lengths $k_1, k_2, \dots k_p$, respectively. Is F-SUBDIVISION polynomial-time solvable?

Theorem 56. $(C_2 + C_3)$-SUBDIVISION *is polynomial-time solvable.*

Proof. Let D be a digraph. If D has no 2-cycles, then D has a $C_2 + C_3$-subdivision if and only if it contains two disjoint cycles. This can be checked in polynomial time by Theorem 51.

Assume now that D contains 2-cycles. For each 2-cycle (x,y,x), we check if $D - \{x,y\}$ has a directed cycle of length at least 3. This can be done in linear time according to Theorem 18. If the answer is ‘yes’ for one of them, then we return ‘yes’.

Suppose now that the answer is ‘no’ for all 2-cycles. Let D' be the digraph obtained from D by deleting the arcs of all the 2-cycles.

Claim 56.1. *D contains a $(C_2 + C_3)$-subdivision if and only if D' contains two disjoint directed cycles.*

Subproof. Suppose that D contains a $(C_2 + C_3)$-subdivision S. No cycle of S can contain two vertices x and y in a 2-cycle because $D - \{x,y\}$ contains no directed cycle of length at least 3. In particular, all the arcs of S are in D'.

Conversely, if D' contains two disjoint directed cycles, they form a $(C_2 + C_3)$-subdivision since D' has no 2-cycles. $\Diamond$

Hence we check if D' has two disjoint directed cycles, which can be done in polynomial time according to Theorem 51. $\square$

9.2. Subdivisions of wheels and fans

The *fan* F_k is the graph obtained from the directed path P_k by adding a vertex, called the *centre*, dominated by every vertex of P_k. The *wheel* W_k is the graph obtained from the directed cycle C_k by adding a vertex, called the *centre*, dominated by every vertex of C_k. The path P_k (resp. cycle C_k) is called the *rim* of F_k (resp. W_k) and the arcs incident to the centre are called the *spokes*. Similarly, if D' is a subdivision of a wheel or a fan D, the *centre* of D' is the vertex corresponding to the centre of D, the *rim* of D' is the directed path or cycle corresponding to the rim of D, and the *spokes* of D' are the directed paths corresponding to the spokes of D.

Proposition 57. *A digraph D contains a W_2-subdivision if and only if it contains some vertex z such that $D - z$ has a strong component S and two directed (S,z)-paths having only z in common.*

Proof. Suppose D contains a subdivision of W_2 with centre z and cycle C. Then the strong component of $D-z$ which contains C satisfies the required property.

Conversely, assume z is a vertex and S is a strong component of $D-z$ such that there are two directed (S,z)-paths P and Q having only z in common. Let x and y be the origins of P and Q respectively.

Let R be a directed (x,y)-path in S and R' a directed (y,x)-path in S. (Such paths exists since S is a strong component.) If R and R' form a cycle we are done, with this cycle as rim and P,Q as spokes. Otherwise let q be the last vertex in $R' \setminus \{x,y\}$ which is also on R. Then we have a W_2-subdivision with rim $R[x,q]R'[q,x]$ and spokes P and $R[q,y]Q$. □

Corollary 58. *W_2-*SUBDIVISION *is solvable in $O(n\cdot(n+m))$ time.*

Theorem 59. *For all $k\geq 4$, W_k-*SUBDIVISION *is NP-complete.*

Proof. We give the proof for $k=4$ (the case for larger k is very similar). We show a reduction from 2-LINKAGE in digraphs with no big vertices in which x_1 and x_2 are sources and y_1 and y_2 are sinks.

Let D,x_1,x_2,y_1,y_2 be an instance of this problem. Let D' be the graph obtained by adding five new vertices z, a, b, c, d and the arcs az, bz, cz, dz, ab, cd, y_2a, bx_1, y_1c, and dx_2.

Let us prove that D' has a W_4-subdivision if and only if D has a 2-linkage from (x_1,x_2) to (y_1,y_2).

If P_1,P_2 form the desired 2-linkage in D, then we take $P_1y_1cdP_2abx_1$ as the rim and the four arcs az, bz, cz, dz as the spokes.

Conversely, suppose W is a subdivision of W_4 in D' and let C be its rim. The centre of W must be z as this is the only vertex of in-degree 4 in D'. Thus the four paths ending in z will end in the arcs az, bz, cz, dz, respectively. Now observe that a (and similarly c) must belong to C since otherwise the path containing az cannot be disjoint from the path containing bz (they will meet in a). Thus a is on C and then b is on C since it is the only out-neighbour of a different from z. Similarly d is on C. Hence C contains the arcs ab and cd and this implies that C contains disjoint paths from x_1 to y_1 and x_2 to y_2 respectively. □

Remark 60. It is not difficult to modify the proof above to a proof that F-SUBDIVISION is NP-complete whenever F is any digraph obtained from a W_k with $k\geq 4$ by reorienting one or more of the spokes. E.g. if the arc dz is reversed, then we replace the arcs ab and cd by arcs ax_1,y_1b,cx_2,y_2d. We leave the details to the interested reader.

From this remark and Lemmas 2, 3 and 4 we get the following corollary. Notice that the resulting digraphs may still have only one big vertex so the conclusion does not follow from Theorem 8.

Corollary 61. *Let W'_k, $k\geq 4$ be a strongly connected digraph obtained from W_k by reversing between one and $k-1$ spokes and let G be any digraph not containing a subdivision of W'_k. Then F-*SUBDIVISION *and F'-*SUBDIVISION *are NP-complete, where F is obtained from W'_k and G by adding zero or more arcs from $V(W'_k)$ to $V(G)$ and F' is obtained from W'_k and G by identifying the big vertex of W'_k with an arbitrary vertex of G.*

Corollary 58 and Theorem 59 determine the complexity of W_k-SUBDIVISION for all k except 3. So we are left with the following problem.

Problem 62. What is the complexity of W_3-SUBDIVISION ?

We now turn to fans. Notice that F_k is W_k where one arc of the rim is deleted. Observe that F_2 is the $(1,2)$-spindle. Thus F_2-SUBDIVISION can be solved in $O(n^2(n+m))$ time by Proposition 22. The next result shows that F_3-SUBDIVISION is polynomial.

Let z be a vertex in a digraph D. A triple (x_1,x_2,x_3) is *F_3-nice* with respect to z in D if the following holds:

- x_1, x_2, x_3 are distinct vertices of $D-z$;
- x_3z is an arc;
- in $D-x_3$, there exist a directed (x_1,z)-path P_1 and a directed (x_2,z)-path P_2 which intersect only in z;
- in $D-\{x_3,z\}$, there is a directed (x_1,x_2)-path Q_1, and in $D-\{x_1,z\}$, there is a directed (x_2,x_3)-path Q_2.

Theorem 63. *A digraph contains an F_3-subdivision with centre z if and only if there is an F_3-nice triple with respect to z. In particular F_3-SUBDIVISION is polynomial-time solvable.*

Proof. Trivially, if D contains an F_3-subdivision with centre z, then it contains an F_3-nice triple (x_1,x_2,x_3) with respect to z.

Conversely, assume that D contains an F_3-nice triple (x_1,x_2,x_3) with respect to z. Let P_1, P_2, Q_1 and Q_2 be the directed paths as defined in the definition of F_3-nice triple. We may assume that (x_1,x_2,x_3) is an F_3-nice triple (x_1,x_2,x_3) with respect to z that minimizes $\ell=\ell(P_1)+\ell(P_2)+\ell(Q_1)+\ell(Q_2)$, that is the sum of the lengths of these paths.

We shall prove that P_1, P_2, Q_1 and Q_2 are internally disjoint, implying that these paths and the arc x_3z form an F_3-subdivision with centre z.

a) Let us prove that Q_2 and P_1 are internally disjoint. Suppose not. Then let x_2' be the last vertex on Q_2 which also belongs to P_1. Then (x_2,x_2',x_3) is F_3-nice by the choice of paths $P_1'=P_2$, $P_2'=P_1[x_2',z]$, $Q_1'=Q_2[x_2,x_2']$ and $Q_2'=Q_2[x_2',x_3]$. Indeed, P_1' and P_2' are internally disjoint because P_1 and P_2 were, Q_1' does not go through x_3 nor z, because Q_2 is a directed (x_2,x_3)-path in $D-z$, and Q_2' does not go through x_2 nor z,for the same reason. This contradicts the minimality of ℓ.

b) Let us prove that Q_2 and P_2 are internally disjoint. Suppose not. Then let x_2' be the last vertex on Q_2 which also belongs to P_2. One easily verifies that (x_1,x_2',x_3) is F_3-nice by the choice of paths $P_1'=P_1$, $P_2'=P_2[x_2',z]$, Q_1' a directed (x_1,x_2')-path included in $Q_1[x_1,x_2]Q_2[x_2,x_2']$ (which can be a walk), and $Q_2'=Q_2[x_2',x_3]$. This contradicts the minimality of ℓ.

c) Let us prove that Q_1 and P_1 are internally disjoint. Suppose not. Then let x_1' be the last vertex on Q_1 which also belongs to P_1. The path Q_2 does not go through x_1' because Q_2 and P_1 are internally disjoint. Thus (x_1',x_2,x_3) is F_3-nice with associated paths $P_1'=P_1[x_1',z]$, $P_2'=P_2$, $Q_1'=Q_1[x_1',x_2]$, and $Q_2'=Q_2$. This contradicts the minimality of ℓ.

d) Let us prove that Q_1 and P_2 are internally disjoint. Suppose not. Then let x'_2 be the last internal vertex on Q_1 which also belongs to P_2. Then (x_1, x'_2, x_3) is F_3-nice with associated paths $P'_1 = P_1$, $P'_2 = P_2[x'_2, z]$, $Q'_1 = Q_1[x_1, x'_2]$, and Q'_2 a directed (x_1, x'_2)-path included in $Q_1[x'_2, x_2]Q_2$ (which can be a walk). This contradicts the minimality of ℓ.

e) Let us prove that Q_1 and Q_2 are internally disjoint. Suppose not. Then let x'_2 be the last internal vertex on Q_2 which also belongs to Q_1. Then (x_1, x'_2, x_3) is a good triple with associated paths $P'_1 = P_1$, $P'_2 = Q_1[x'_2, x_2]P_2$, $Q'_1 = Q_1[x_1, x'_2]$, and $Q'_2 = Q_2[x'_2, x_3]$. Indeed, since P_2 and Q_1 are internally disjoint, P'_2 is a path, and since P_1 and Q_1 are internally disjoint, the paths P'_1 and P'_2 are also internally disjoint.

□

Theorem 64. *For all $k \geq 5$, F_k-*SUBDIVISION *is NP-complete.*

Proof. Reduction from 2-LINKAGE in digraphs with no big vertices in which x_1 and x_2 are sources and y_1 and y_2 are sinks.

Let D, x_1, x_2, y_1 and y_2 be an instance of this problem. Let us denote by z the centre of F_k and by $(v_1, v_2, \ldots, v_k)$ the directed path $F_k - z$. Let D_k be the digraph obtained from the disjoint union of D and F_k by removing the arcs v_1v_2 and v_3v_4 and adding the arcs v_1x_1, y_1v_2, v_3x_2 and y_2v_4.

We claim that D_k has an F_k-subdivision if and only if D has a linkage from (x_1, x_2) to (y_1, y_2).

Clearly, if there is a linkage (P_1, P_2) from (x_1, x_2) to (y_1, y_2) in D, then D_k contains an F_k-subdivision, obtained from F_k by replacing the arc v_1v_2 and v_3v_4 by the directed paths $(v_1, x_1) \cup P_1 \cup (y_1, v_2)$ and $(v_3, x_2) \cup P_2 \cup (y_2, v_4)$, respectively.

Suppose now that D_k contains an F_k-subdivision S in D_k. Since z is the unique vertex with in-degree k, the centre of S' is necessarily z. For $1 \leq i \leq k$, let v'_i be the vertex corresponding to v_i in S, and P_i be the directed (v'_i, z)-path in S.

Since z has in-degree exactly k in D_k, the v_i's are the penultimate vertices of the P_j's, each v_i on a different P_j. Since v_1 is a source in D_k, then $v_1 = v'_1$. Moreover, for $i = 3$ and $i \geq 5$, the path P'_j containing v_i must start at v_i because the unique in-neighbour of v_i is v_{i-1}. Hence $v_i = v'_j$. Furthermore, necessarily $v_{i-1} = v'_{j-1}$. Now, because v_k is a sink in $D_k - z$, then necessarily $v'_k = v_k$ and so for all $1 \leq i \leq k$, we have $v'_i = v_i$.

Let Q_1 and Q_2 be the directed (v_1, v_2)- and (v_3, v_4)-paths, respectively. Necessarily, the second vertex of Q_1 (resp. Q_2) is x_1, (resp. x_2) and its penultimate vertex is y_1 (resp. y_2). Hence $(Q_1[x_1, y_1], Q_2[x_2, y_2])$ is a linkage from (x_1, x_2) to (y_1, y_2) in D. □

Proposition 22 and Theorems 63 and 64 determine the complexity of F_k-SUBDIVISION for all k except 4. So we are left with the following problem.

Problem 65. What is the complexity of F_4-SUBDIVISION ?

9.3. Subdivisions of transitive tournaments

Denote by TT_k the transitive tournament on k vertices. For $k \leq 3$, TT_k-SUBDIVISION is polynomial-time solvable because TT_1 and TT_2 are spiders and TT_3 is the $(1,2)$-spindle. On the other hand, for all $k \geq 5$, TT_k-SUBDIVISION is NP-complete by Corollary 10. We shall now prove that TT_4-SUBDIVISION is polynomial-time solvable.

In fact we will prove it for some classes of graphs contructed from TT_4. For any non-negative integer p, let $TT_4(p)$ be the digraph obtained from TT_4 with source u and sink v by adding p new vertices dominated by u and dominating v. In particular, $TT_4(0) = TT_4$. We denote by $TT_4^*(p)$, the digraph obtained from $TT_4(p)$ by deleting the arc from its source u to its sink v. For simplicity, we abbreviate $TT_4^*(0)$ in TT_4^*.

We need the following definitions. Let X be a set of vertices in a digraph D. The *out-section* generated by X in D is the set of vertices y to which there exists a directed path (possibly restricted to a single vertex) from $x \in X$; we denote this set by $S_D^+(X)$. For simplicity, we write $S_D^+(x)$ instead of $S_D^+(\{x\})$. The dual notion, the *in-section*, is denoted by $S_D^-(X)$. Note that the out-section and the in-section of a set may be found in linear time by any tree-search algorithm.

Theorem 66. *For every non-negative integer p, one can solve $TT_4(p)$-*SUBDIVISION *in $O(n^3(n+m))$-time.*

Proof. Let D be a digraph and let u and v be two distinct vertices of D. We shall describe a $O(n(n+m))$-time algorithm for finding a $TT_4(p)$-subdivision in D with source u and sink v, if one exists.

Observe that all vertices in such a subdivision are in $S_D^+(u) \cap S_D^-(v)$, hence we first restrict to the graph D' the digraph induced by this set.

Then, using a maximum flow algorithm, we can find in D' a set of internally disjoint directed (u,v)-paths of maximum size in $O(n(n+m))$-time. Let $(P_1,\dots,P_k)$ denote this set. If $k < p+3$, then return 'no', because in any $TT_4(p)$-subdivision with source u and sink v, there are $p+3$ internally disjoint directed (u,v)-paths Hence, we now assume that $k \geq 3$.

For $1 \leq i \leq k$, set $Q_i = P_i - \{u,v\}$, and set $H = D' - \{u,v\}$. For every vertex x in $V(H)$, we compute $S(x) = S_H^-(x) \cup S_H^+(x)$, and deduce $I(x) = \{i \mid V(Q_i) \cap S(x) \neq \emptyset\}$. If there exists x, such that $|I(x)| \geq 2$, then return 'yes'. Otherwise return 'no'.

The validity of this algorithm is proved by Claim 66.2.

Claim 66.1. *For all $x \in V(H)$, $I(x) \neq \emptyset$.*

Subproof. In D', there are directed (u,x)- and (x,v)-paths, whose concatenation contains a directed (u,v)-path R. Since $(P_1,\dots,P_k)$ is a set of internally disjoint directed (u,v)-paths of maximum size, $R - \{u,v\}$ must intersect one of the Q_i's, say Q_{i_0}. By definition, $V(R) \setminus \{u,v\} \subseteq S(x)$, so $i_0 \in I(x)$. $\Diamond$

Claim 66.2. *D' contains a $TT_4(p)$-subdivision with source u and sink v if and only if there exists $x \in V(H)$ such that $|I(x)| \geq 2$.*

Subproof. Assume that $|I(x)| \geq 2$. Without loss of generality, $\{1,2\} \subset I(x)$. We shall prove that D' contains a $TT_4(p)$-subdivision with source u and sink v.

- Suppose first that $S_H^-(x) \cap Q_1 \neq \emptyset$ and $S_H^+(x) \cap Q_2 \neq \emptyset$. Then there is a directed (Q_1,x)-path and a directed $(x,Q_2)-path$ whose concatenation contains a directed (Q_1,Q_2)-path R. Let y be the first vertex on R in $\bigcup_{i=2}^k Q_i$. Free to swap the names of Q_2 and the path Q_l containing y and taking the subpath of R from its origin to y instead of R, we may assume that y is the last vertex of R. Now the union of $P_1,\dots,P_{p+3}$, and R form a $TT_4(p)$-subdivision.

- If $S^-_H(x) \cap Q_2 \neq \emptyset$ and $S^+_H(x) \cap Q_1 \neq \emptyset$, the proof is similar to the previous case.

- Suppose now that $S^+_H(x) \cap Q_1 \neq \emptyset$ and $S^+_H(x) \cap Q_2 \neq \emptyset$. We may assume that $S^-_H(x) \cap \bigcup_{i=1}^k Q_i = \emptyset$, otherwise we are in one of the previous case, and we get the result. Let R be a shortest (u,x)-path in D'. Then every vertex in $R - u$ is a vertex of $H - \bigcup_{i=1}^k Q_i$.

  Let S_1 be a shortest directed (x, Q_1)-path and S_2 be a shortest directed (x, Q_2)-path. For $i = 1,2$, let z_i be the terminus of S_i. We may assume that all the internal vertices of S_1 and S_2 are in $H - \bigcup_{i=1}^k Q_i$ for otherwise one vertex z among z_1 and z_2 satisfies the condition of one of the previous cases (up to a permutation of the labels). Then the union of paths $P_2, \dots, P_{p+3}$, R, S_1, S_2 and $P_1[z_1, v]$ form a $TT_4(p)$-subdivision.

- If $S^-_H(x) \cap Q_1 \neq \emptyset$ and $S^-_H(x) \cap Q_2 \neq \emptyset$, the proof is similar to the previous case by directional symmetry.

Assume now that $|I(x)| < 2$ for all $x \in V(H)$. Then, by Claim 66.1, $|I(x)| = 1$ for all $x \in V(H)$. For $1 \leq i \leq k$, let $V_i = \{x \mid I(x) = \{i\}\}$. Then $(V_1, \dots V_k)$ is a partition of $V(H)$. Moreover, by definition, there is no arc between two distinct parts of this partition. In addition, in $D'\langle X_i \cup \{u,v\}\rangle$, there cannot be two internally disjoint directed (u,v)-paths, for otherwise it would contradicts the maximality of $(P_1, \dots, P_k)$. Hence, D' contains no TT_4^*-subdivision, and so no $TT_4(p)$-subdivision. $\Diamond$

This finishes the proof of Theorem 66. $\square$

Corollary 67. *For all non-negative integer p, the $TT_4^*(p)$-SUBDIVISION problem can be solved in $O(n^3(n+m))$.*

Proof. Observe that a graph D contains a $TT_4^*(p)$-subdivision with source u and sink v, if and only if the graph $D \cup \{uv\}$ contains a $TT_4(p)$-subdivision. Hence by just adding the arc uv to D if it does not exists in the above algorithm, we obtain a polynomial-time algorithm for $TT_4^*(p)$-SUBDIVISION. $\square$

9.4. Subdivisions of digraphs with three vertices

Let us denote by $\vec{K}_n$ the complete digraph on n vertices, in which there is an arc uv for any two distinct vertices u and v. Let D_3 be the digraph obtained from $\vec{K}_3$ by removing an arc.

Theorem 68. *Let F be a digraph on three vertices. Then F-SUBDIVISION is polynomial-time solvable unless $F = \vec{K}_3$ in which case it is NP-complete.*

Proof. If F is neither D_3 nor $\vec{K}_3$, then it is either a disjoint union of spiders, or a spindle, or a bispindle, or the lollipop (or its converse), or a windmill, and so F-SUBDIVISION can be solved in polynomial time by virtue of the results of the previous sections. If $F = \vec{K}_3$, then F-SUBDIVISION is NP-complete by Corollary 10.

It remains to prove that D_3-SUBDIVISION is polynomial-time solvable.

The *bulky vertex* of a D_3-subdivision S is the unique vertex of S with degree 4. We now give a procedure that given a vertex v, two of its out-neihbours s_1, s_2 and two of its in-neighbours t_1, t_2 check if there is a D_3-subdivision S in which v is the bulky vertex and $\{vs_1, vs_2, t_1v, t_2v\} \in A(S)$. Such a subdivision will be called *suitable*.

Applying a Menger algorithm, check if in $D-v$ there are two disjoint directed paths P_1 and P_2 from $\{s_1,s_2\}$ to $\{t_1,t_2\}$. If not, then D certainly does not contain any suitable D_3-subdivision. If yes, then check if there is a directed path Q from P_1 to P_2 or from P_2 to P_1. If such a Q exists, then P_1, P_2, Q together with v and the arcs vs_1, vs_2, t_1v, t_2v form a suitable D_3-subdivision. If not, then no suitable D_3-subdivision using the chosen arcs exists, because there is no vertex $s \in \{s_1,s_2\}$ such that there exists in $D-v$ both a directed (s,t_1)-path and a directed (s,t_2)-path.

A D_3-subdivision is clearly suitable with respect to its bulky vertex and its neighbours in this subdivision. Hence checking if there is a suitable D_3-subdivision for every 5-tuple (v,s_1,s_2,t_1,t_2) such that s_1, s_2 are out-neighbours of v and t_1, t_2 are out-neighbours yields a polynomial-time algorithm to decide if there is a D_3-subdivision in a digraph. □

9.5. Subdivision of oriented paths and cycles

Conjecture 69. If F is an oriented path or cycle, then F-SUBDIVISION is polynomial-time solvable.

Proposition 70. *If P is an oriented path with at most four blocks, then P-*SUBDIVISION *is polynomial-time solvable.*

An *antidirected path* is an oriented path in which every vertex has either in-degree 0 or out-degree 0.

Theorem 71. *If P is an antidirected path, then P-*SUBDIVISION *is polynomial-time solvable.*

Proof. Let $P=(a_1,\dots,a_p)$ be an antidirected path. By directional symmetry, we may assume that a_i has indegree 0 in P if and only if i is odd.

Let D be a digraph. For a p-tuple of vertices $(v_1,\dots,v_p)$, we shall describe a procedure that either returns a P-subdivision, or returns that there exists no P-subdivision in which each v_i is the image of a_i. Then applying this procedure for all p-tuples of vertices, we obtain the desired algorithm to finding a P-subdivision.

The procedure is as follows: For all odd (resp. even) i, we remove all the arcs entering v_i (resp. leaving v_i) in D. Let D' be the resulting digraph. Clearly, D contains a P-subdivision in which each v_i is the image of a_i if and only if D' does. In $UG(D')$, we check if there is a path $\tilde{Q}$ going through $v_1,\dots,v_p$ in this order. This can be done by checking for a linkage from $(v_1,v_2,\dots,v_{p-1})$ to $(v_2,v_3,\dots,v_p)$ and thus in polynomial time by Robertson and Seymour algorithm [18].

If no such $\tilde{Q}$ is found, then D' (and thus D) contains certainly no P-subdivision in which each v_i is the image of a_i.

If such a $\tilde{Q}$ is found, let Q be the oriented path corresponding to Q in D'. Since v_i is a source in D' when i is odd, and a sink in D' when i is even, the path Q has at least $p-1$ blocks, and so contains a subdivision of P. □

Remark 72. Using the same technique, one can show that if P is an oriented path, all blocks of which have length one except possibly two consecutive blocks, then P-SUBDIVISION is polynomial-time solvable.

10. Concluding remarks

The following conjecture, due to Seymour (private communication, 2011) would imply a number of the results on polynomial instances in the previous sections.

Conjecture 73 (Seymour)**.** F-SUBDIVISION is polynomial-time solvable when F is a planar digraph with no big vertices.

This conjecture would indeed be implied by the following conjecture. An arc uv in a digraph is *contractible* if $\min\{d^+(u), d^-(v)\} = 1$. A *minor* of a digraph D is any subdigraph $\tilde{D}$ of D which can be obtained from a subdigraph H of D by contracting zero or more contractible arcs of H. For $k = 1, 2, \ldots, k$ the digraph J_k is obtained from the union of k directed cycles (each of length $2k$) $C_1, C_2, \ldots, C_k$, where $C_i = u_{i,1}v_{i,1}u_{i,2}v_{i,2}\ldots u_{i,k}v_{i,k}u_{i,1}$, for $i = 1, 2\ldots, k$ and paths P_i, Q_i, $i = 1, 2\ldots, k$, where $P_i = u_{1,i}u_{2,i}\ldots u_{k,i}$ and $Q_i = v_{k,i}v_{k,i-1}\ldots v_{k,1}$ for $i = 1, 2\ldots, k$.

Conjecture 74 (Johnson et al. [14])**.** For every positive integer k there exists $N(k)$ such that the following holds: If a digraph D has directed treewidth more than $N(k)$, then D contains a minor isomorphic to J_k.

If the directed tree-width of D is bounded, then, by Theorem 46, F-SUBDIVISION can be solved in polynomial time. If, on the other hand, the directed tree-width of D is unbounded, then (if the algorithmic version of the conjecture also holds) we can find a minor isomorphic to J_k for a sufficiently large k and presumably use this to realize the desired subdivision using the fact the F is planar and has no big vertices.

Conjecture 75. F-SUBDIVISION is NP-complete for every non-planar digraph F.

For any positive integer p, let us denote by $\mathcal{C}_p$, the class of digraphs in which all directed cycles have length at most p. Then $\mathcal{C}_1$ may be seen as the class of acyclic digraphs.

Problem 76. Is k-LINKAGE polynomial-time solvable on $\mathcal{C}_p$?

Thomassen proved [20] that for every natural number p there exists a p-strongly connected digraph D_p which is not 2-linked, that is, there exists no linkage from (s_1, s_2) to (t_1, t_2) for some choice of distinct vertices s_1, s_2, t_1, t_2 of D_p.

Problem 77. Let F be a fixed digraph. Does there exists k_F such that every k_F-strongly connected digraph contains an F-subdivision or at least such that F-SUBDIVISION is polynomial-time solvable when restricted to k_F-strongly connected digraphs?

Note that if F_1-SUBDIVISION and F_2-SUBDIVISION are both polynomial-time solvable, then (F_1+F_2)-SUBDIVISION is sometimes polynomial-time solvable and sometimes NP-complete. For example, if F_1 is the disjoint union of spiders and F_2-SUBDIVISION is polynomial-time solvable, then (F_1+F_2)-SUBDIVISION is polynomial time solvable. On the other hand, assume that F_1 and F_2 are $(1,2,2)$-spindles. Then by Proposition 22, F_1-SUBDIVISION and F_2-SUBDIVISION are both polynomial-time solvable, but according to Theorem 8, (F_1+F_2)-SUBDIVISION is NP-complete.

Hence for every two digraphs F_1 and F_2 such that F_1-SUBDIVISION and F_2-SUBDIVISION have been proved to be polynomial-time solvable, it is natural to ask for the complexity of (F_1+F_2)-SUBDIVISION. In particular, the following problem is one of the first to study.

Problem 78. Let F_1 and F_2 be two $(1,2)$-spindles, i.e. transitive tournaments of order 3. What is the complexity of (F_1+F_2)-SUBDIVISION?

Acknowledgement

The authors would like to thank Nicolas Trotignon for stimulating discussion and suggesting them the FORK problem.

[1] J. Bang-Jensen and G. Gutin. *Digraphs: Theory, Algorithms and Applications*. Springer Monographs in Mathematics. Springer Verlag, London, 2008.

[2] J. Bang-Jensen, F. Havet and N. Trotignon. Finding an induced subdivision of a digraph. Theoretical Computer Science, 2012 to appear.

[3] J. Bang-Jensen and M. Kriesell. Disjoint directed and undirected paths and cycles in digraphs *Theoretical Computer Science* 410, 5138–5144, 2009.

[4] D. Berwanger, A. Dawar, P. Hunter, and S. Kreutzer. DAG-width and parity games. In STACS 2006, Proc. 23rd Symposium on Theoretical Aspects of Computer Science, Volume 3884 of Lect. Notes Comput. Sci., pages 524-536. Springer-Verlag, 2006.

[5] J. A. Bondy and U. S. R. Murty. *Graph theory, vol. 244 of Graduate Texts in Mathematics.* Springer, New York, 2008.

[6] M. Chudnovsky, A. Scott and P.D. Seymour. Disjoint paths in tournaments. Manuscript 2012.

[7] M. Chudnovsky and P.D. Seymour. Excluding induced subgraphs. In *Surveys in Combinatorics*, volume 346, pages 99–119. London Mathematical Society Lecture Notes Series, 2007.

[8] M. Chudnovsky and P.D. Seymour. The three-in-a-tree problem. *Combinatorica*, Volume 30, pages 387–417, 2010.

[9] B. Courcelle and J. Engelfriet. *Graph structure and monadic second-order logic. A language-theoretic approach. Encyclopedia of Mathematics and its Applications*, 138. Cambridge University Press, Cambridge, 2012.

[10] S. Fortune, J.E. Hopcroft, and J. Wyllie. The directed subgraph homeomorphism problem. *Theoretical Computer Science*, 10:111–121, 1980.

[11] H. N. Gabow and S. Nie. Finding a long directed cycle. In *Proceedings of the fifteenth annual ACM-SIAM symposium on Discrete algorithms (SODA '04)*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 49–58, 2004.

[12] H. N. Gabow and S. Nie. Finding a long directed cycle. *ACM Trans. Algorithms*, 4(1):Article 7, 21 pages, 2008.

[13] M. Grohe and M. Grüber Parameterized approximability of the disjoint cycle problem. In *Proceedings of the 34th International Colloquium on Automata, Languages and Programming (ICALP'07)*, volume 4596 of *LNCS*, pages 363–374, 2007.

[14] T. Johnson, N. Robertson, P.D. Seymour and R. Thomas Directed tree-width. *J. Combinatorial Theory, Ser. B* **82** (2001)

[15] B. Lévêque, D. Lin, F. Maffray, and N. Trotignon. Detecting induced subgraphs. *Discrete Applied Mathematics*, **157** (2009) 3540–3551.

[16] W. McCuaig, "Intercyclic digraphs", Graph structure theory (Seattle, WA, 1991), Contemp. Math. 147, Amer. Math. Soc., Providence, RI (1993), 203–245.

[17] B. Reed, N. Robertson, P.D. Seymour, and R. Thomas. Packing directed circuits. *Combinatorica*, 16(4):535–554, 1996.

[18] N. Robertson and P.D. Seymour. Graph minors. XIII. The disjoint paths problem. *Journal of Combinatorial Theory, Series B*, 63:65–110, 1995.

[19] A. Slivkins. Parameterized tractability of edge-disjoint paths on directed acyclic graphs. In G. D. Battista and U. Zwick, editors, *Proceedings of the 11th Annual European Symposium on Algorithms, ESA 03*, volume 2832 of *LNCS*, pages 482–493, 2003.

[20] C. Thomassen. Highly connected non-2-linked digraphs. Combinatorica, 11(4):393-395, 1991.