Journal of Computer and System Sciences 137 (2023) 66–86

Contents lists available at ScienceDirect

Journal of Computer and System Sciences

journal homepage: www.elsevier.com/locate/jcss

Detours in directed graphs ☆

Fedor V. Fomin [a], Petr A. Golovach [a,*], William Lochet [b], Danil Sagunov [c], Saket Saurabh [a,d], Kirill Simonov [e]

[a] Department of Informatics, University of Bergen, Bergen, Norway
[b] LIRMM, Université Montpellier, CNRS, Montpellier, France
[c] St. Petersburg Department of V.A. Steklov Institute of Mathematics, Russia
[d] Institute of Mathematical Sciences, HBNI, Chennai, India
[e] Hasso Plattner Institute, University of Potsdam, Germany

ARTICLE INFO

Article history:
Received 26 September 2022
Received in revised form 4 May 2023
Accepted 12 May 2023
Available online 22 May 2023

Keywords:
Longest path
Longest detour
Diameter
Directed graphs
Parameterized complexity

ABSTRACT

We study two "above guarantee" versions of the classical Longest Path problem on undirected and directed graphs and obtain the following results. In the first variant of Longest Path that we study, called Longest Detour, the task is to decide whether a graph has an (s,t)-path of length at least $\mathrm{dist}_G(s,t)+k$. Bezáková et al. [7] proved that on undirected graphs the problem is fixed-parameter tractable (FPT). Our first main result establishes a connection between Longest Detour on directed graphs and 3-Disjoint Paths on directed graphs. Using these new insights, we design a $2^{\mathcal{O}(k)} \cdot n^{\mathcal{O}(1)}$ time algorithm for the problem on directed planar graphs. Furthermore, the new approach yields a significantly faster FPT algorithm on undirected graphs. In the second variant of Longest Path, namely Longest Path above Diameter, the task is to decide whether the graph has a path of length at least $\mathrm{diam}(G)+k$. We obtain dichotomy results about Longest Path above Diameter on undirected and directed graphs.

1. Introduction

In the Longest Path problem, we are given an n-vertex graph G and an integer k. (Graph G could be undirected or directed.) The task is to decide whether G contains a path of length at least k. Longest Path is a fundamental algorithmic problem that played one of the central roles in developing parameterized complexity [49,9,1,37,42,11,12,44,54,22,21,45,8]. To further our algorithmic knowledge about the Longest Path problem, Bezáková et al. [7] introduced a novel "above guarantee" parameterization for the problem. For a pair of vertices s,t of an n-vertex graph G, let $\mathrm{dist}_G(s,t)$ be the distance from s to t, that is, the length of a shortest path from s to t. In this variant of Longest Path, the task is to decide whether a graph has an (s,t)-path of length at least $\mathrm{dist}_G(s,t)+k$. The difference with the "classical" parameterization is that instead of parameterizing by the path length, the parameterization is by the offset k.

☆ A preliminary version of this paper appeared in the proceedings of STACS 2022. The research received funding from European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant no. 819416), the Swarnajayanti Fellowship grant DST/SJF/MSA-01/2017-18, the Research Council of Norway via the project BWCA (grant no. 314528), and DFG Research Group ADYN via grant DFG 411362735.

* Corresponding author at: University of Bergen, PB 7803, N-5020, Bergen, Norway.
E-mail addresses: fedor.fomin@uib.no (F.V. Fomin), petr.golovach@uib.no (P.A. Golovach), william.lochet@gmail.com (W. Lochet), danilka.pro@gmail.com (D. Sagunov), saket@imsc.res.in (S. Saurabh), kirillsimonov@gmail.com (K. Simonov).

https://doi.org/10.1016/j.jcss.2023.05.001

Longest Detour **Parameter:** k
Input: A graph G, vertices $s, t \in V(G)$, and an integer k.
Task: Decide whether there is an (s,t)-path in G of length at least $\text{dist}_G(s,t) + k$.

Since the length of a shortest path between s and t can be found in linear time, such a parameterization could provide significantly better solutions than the parameterization by the path length. Bezáková et al. [7] proved that on undirected graphs the problem is fixed-parameter tractable (FPT) by providing an algorithm of running time $2^{\mathcal{O}(k)} \cdot n$. Parameterized complexity of Longest Detour on directed graphs was left as the main open problem in [7]. Our paper makes significant step towards finding a solution to this open problem.

Our results. Our first main result establishes a connection between Longest Detour and another fundamental algorithmic problem p-Disjoint Paths. Recall that the p-Disjoint Paths problem is to decide whether p pairs of *terminal* vertices (s_i, t_i), $i \in \{1, \dots, p\}$, in a (directed) graph G could be connected by pairwise internally vertex disjoint (s_i, t_i)-paths. We prove (the formal statement of our result is given in Theorem 1) that if $\mathcal{C}$ is a class of (directed) graphs such that p-Disjoint Paths admits a polynomial time algorithm on $\mathcal{C}$ for $p = 3$, then Longest Detour is FPT on $\mathcal{C}$. Moreover, the FPT algorithm for Longest Detour on $\mathcal{C}$ is single-exponential in k (running in time $2^{\mathcal{O}(k)} \cdot n^{\mathcal{O}(1)}$).

Unfortunately, our result does not resolve the question about parameterized complexity of Longest Detour on directed graphs. Indeed, Fortune, Hopcroft, and Wyllie [29] proved that p-Disjoint Paths is NP-complete on directed graphs for every fixed $p \geq 2$. However, the new insight helps to establish the tractability of Longest Detour on planar directed graphs, whose complexity was also open. The theorem of Schrijver from [51] states that p-Disjoint Paths could be solved in time $n^{\mathcal{O}(p)}$ when the input is restricted to planar directed graphs. (This result was improved by Cygan et al. [17] who proved that p-Disjoint Paths parameterized by p is FPT on planar directed graphs.) Pipelined with our theorem, this immediately implies that Longest Detour is FPT on planar directed graphs.

Besides establishing parameterized complexity of Longest Detour on planar directed graphs our theorem has several advantages over the previous work even on undirected graphs. By the seminal result of Robertson and Seymour [50], p-Disjoint Paths is solvable in $f(p) \cdot n^3$ time on undirected graphs for some function f of p only. Therefore on undirected graphs p-Disjoint Paths is solvable in polynomial time for every fixed p, and for $p = 3$ in particular. Later the result of Robertson and Seymour was improved by Kawarabayashi, Kobayashi, and Reed [41] who gave an algorithm with quadratic dependence on the input size. Pipelined with our result, this brings us to a Monte Carlo randomized algorithm solving Longest Detour on undirected graphs in time $10.8^k \cdot n^{\mathcal{O}(1)}$. Our algorithm can be derandomized, and the deterministic algorithm runs in time $45.5^k \cdot n^{\mathcal{O}(1)}$. While the algorithm of Bezáková et al. [7] for undirected graphs runs in time $\mathcal{O}(c^k \cdot n)$, that is, is single-exponential in k, the constant c is huge. The reason is that their algorithm exploits the Win/Win approach based on excluding graph minors. More precisely, Bezáková et al. proved that if a 2-connected graph G contains as a minor, a graph obtained from the complete graph K_4 by replacing each edge by a path with k edges, then G has an (s,t)-path of length at least $\text{dist}_G(s,t) + k$. Otherwise, in the absence of such a graph as a minor, the treewidth of G is at most $32k + 46$. Combining this fact with an FPT 2-approximation algorithm [43] for treewidth, running in time $2^{\mathcal{O}(k)} \cdot n$, gives us a tree decomposition of width at most $64k + \mathcal{O}(1)$. Finally, solving Longest Detour on graphs of bounded treewidth by one of the known single-exponential algorithms, see [16,10,23], will result in running time $3^{64k} \cdot n^{\mathcal{O}(1)}$. Thus on undirected graphs, our randomized algorithm reduces the constant c in the base of the exponent from 3^{64} down to 10.8!

Our second set of results addresses the complexity of the problem strongly related to Longest Detour. The maximum length of a shortest path between two vertices in a graph G is the *diameter* of G, denoted $\text{diam}(G)$. Thus every graph G has a path of length at least $\text{diam}(G)$. But does it have a path of length longer than $\text{diam}(G)$? This leads to the following parameterized problem.

Longest Path above Diameter **Parameter:** k
Input: A graph G and an integer k.
Task: Decide whether there is a path in G of length at least $\text{diam}(G) + k$.

As in Longest Detour, the parameterization is by the offset k. When (s,t) is a pair of diametral vertices in G, the length of a shortest (s,t)-path in G is the diameter of G. However, this does not allow us to reduce Longest Path above Diameter to Longest Detour— if there is a path of length $\text{diam}(G) + k$ in G, it is not necessarily an (s,t)-path. Moreover, such a path might connect two vertices with a much smaller distance between them than $\text{diam}(G)$. In fact, our hardness results for Longest Path above Diameter are based precisely on instances where the target path has this property: its length is very close to $\text{diam}(G)$, but much larger than the minimum distance between its endpoints. Thus, the lower bounds we obtain for Longest Path above Diameter are not applicable to Longest Detour.

We obtain the following dichotomy results about Longest Path above Diameter on undirected and directed graphs. For undirected graphs, Longest Path above Diameter is NP-complete even for $k = 1$. However, if the input undirected graph is 2-connected, that is, it remains connected after deleting any of its vertices, then the problem is FPT. For directed graphs, the problem is also NP-complete even for $k = 1$. However, the situation is more complicated and interesting on 2-connected directed graphs (a strongly connected digraph G is 2-connected or strongly 2-connected if for every vertex $v \in V(G)$, graph $G - v$ remains strongly connected). In this case, we show that Longest Path above Diameter is solvable in polynomial time for each $k \in \{1, \dots, 4\}$ and is NP-complete for every $k \geq 5$.

Our approach. A natural way to approach Longest Detour on directed graphs would be to mimic the algorithm for undirected graphs. By the result of Kawarabayashi and Kreutzer [40], every directed graph of sufficiently large directed treewidth contains a sizable directed grid as a "butterfly minor". However, as reported in [6], there are several obstacles to applying the grid theorem of Kawarabayashi and Kreutzer for obtaining a Win/Win algorithm. After several unsuccessful attempts, we switched to another strategy.

We start by checking whether G has an (s,t)-path of length $\mathrm{dist}_G(s,t)+\ell$ for $k \le \ell < 2k$. This can be done in time $2^{\mathcal{O}(k)} \cdot n^{\mathcal{O}(1)}$ by calling the algorithm of Bezáková et al. [7] that finds an (s,t)-path in a directed G of length *exactly* $\mathrm{dist}_G(s,t)+\ell$. If such a path is not found, we conclude that if (G,k) is a yes-instance, then G contains an (s,t)-path of length at least $\mathrm{dist}_G(s,t)+2k$.

Next, we check whether there exist two vertices v and w reachable from s such that $\mathrm{dist}_G(s,w)-\mathrm{dist}_G(s,v) \ge k$ and G has pairwise disjoint (s,w)-, (w,v)-, and (v,t)-paths. If such a pair of vertices exists, we obtain a solution by concatenating disjoint (s,w)-, (w,v)-, and (v,t)-paths. This is the place in our algorithm, where we require a subroutine solving 3-Disjoint Paths.

When none of the above procedures finds a detour, we prove a combinatorial claim that allows reducing the search for a solution to a significantly smaller region of the graph. This combinatorial claim is an essential part of our algorithm. More precisely, we show that there are two vertices u and x, and a specific induced subgraph H of G (depending on u and x) such that G has an (s,t)-path of length at least $\mathrm{dist}_G(s,t)+k$ if and only if H has an (u,x)-path of length at least ℓ for a specific $\ell \le 2k$ (also depending on u and x). Moreover, given u, in polynomial time, we can find a feasible domain for vertex x, and for each choice of x, we can also determine ℓ and construct H in polynomial time. Then we apply the algorithm of Fomin et al. [24] to check whether H has an (u,x)-path in H of length at least ℓ.

Our strategy for Longest Path above Diameter is different. For undirected graphs, the solution turns out to be reasonably simple. It is easy to show that Longest Path above Diameter is NP-complete for $k=1$ by reducing Hamiltonian Path to it. When an undirected graph G is 2-connected, and the diameter is larger than $k+1$, then G always contains a path of length at least $\mathrm{diam}(G)+k$. If the diameter is at most k, it suffices to run a Longest Path algorithm to show that the problem is FPT. For directed graphs, a similar reduction shows that the problem is NP-complete for $k=1$. However, for strongly 2-connected directed graphs, the situation is much more interesting. It is not too difficult to prove that when the diameter of a strongly 2-connected digraph is sufficiently large, it always contains a path of length $\mathrm{diam}(G)+1$. With much more careful arguments, it is possible to push this up to $k=4$. Thus for each $k \le 4$, the problem is solvable in polynomial time because the task boils down to computing the diameter and checking the existence of a path of constant length if the diameter is small. For $k=5$ we can construct a family of strongly 2-connected digraphs of arbitrarily large diameter that do not have a path of length $\mathrm{diam}(G)+5$. These graphs become extremely useful as gadgets that we use to prove that the problem is NP-complete for each $k \ge 5$.

Related work. There is a vast literature in the field of parameterized complexity devoted to Longest Path [49,9,1,37,42,11,12,44,54,22,8]. The surveys [21,45] and the textbook [18, Chapter 10] provide an overview of the advances in the area.

Longest Detour was introduced by Bezáková et al. in [7]. They gave an FPT algorithm for undirected graphs and posed the question about detours in directed graphs. Even the existence of a polynomial time algorithm for Longest Detour with $k=1$, that is, deciding whether a directed graph has a path longer than a shortest (s,t)-path, is open for general graphs. For the special case of planar digraphs and $k=1$, it was shown by Wu and Wang [55] that the problem can be solved in polynomial time. For the related Exact Detour problem, deciding whether there is a detour of length *exactly* $\mathrm{dist}_G(s,t)+k$ is FPT both on directed and undirected graphs [7].

Another problem related to our work is Long (s,t)-Path. Here for vertices s and t of a graph G, and integer parameter k, we have to decide whether there is an (s,t)-path in G of length at least k. A simple trick, see [18, Exercise 5.8], allows us to use color-coding to show that Long (s,t)-Path is FPT on undirected graphs. For directed graphs, the situation is more involved, and the first FPT algorithm for Long (s,t)-Path on directed graphs was obtained only recently [24]. The proof of Theorem 1 uses some of the ideas developed in [24].

Both Longest Detour and Longest Path above Diameter fit into the research subarea of parameterized complexity called "above guarantee" parameterization [47,2,15,31–35,46,48]. Besides the work of Bezáková et al. [6], several papers study the parameterization of longest paths and cycles above different guarantees. Fomin et al. [26] designed parameterized algorithms for computing paths and cycles longer than the girth of a graph. The same set of authors in [25] studied FPT algorithms that find paths and cycles above degeneracy. Fomin et al. [28,27] developed FPT algorithms computing cycles of length $2\delta+k$ and $mad(G)+k$, respectively, where δ is the minimum vertex degree and $mad(G)$ is the maximum average degree of the input graph. Jansen, Kozma, and Nederlof in [39] looked at parameterized complexity of Hamiltonicity below Dirac's conditions. Berger, Seymour, and Spirkl in [5], gave a polynomial time algorithm that, with an input graph G and two vertices s,t of G, decides whether there is an *induced* (s,t)-path that is longer than a shortest (s,t)-path. This result was recently improved by Chiu and Lu in [13]. All these algorithms for computing long paths and cycles above some guarantee are for undirected graphs.

The remaining part of this paper is organized as follows. In Section 2, we give preliminaries. In Section 3, we prove our first main result establishing connections between 3-Disjoint Paths and Longest Detour (Theorem 1). Section 4 is devoted to Longest Path above Diameter. The concluding Section 5 provides open questions for further research.

2. Preliminaries

Parameterized Complexity. We refer to the recent books [18,20] for the detailed introduction to Parameterized Complexity. Here we just remind that the computational complexity of an algorithm solving a parameterized problem is measured as a function of the input size n of a problem and an integer *parameter* k associated with the input. A parameterized problem is said to be *fixed-parameter tractable* (or FPT) if it can be solved in time $f(k) \cdot n^{\mathcal{O}(1)}$ for some function $f(\cdot)$.

Graphs. Recall that an undirected graph is a pair $G = (V, E)$, where V is a set of vertices and E is a set of unordered pairs $\{u, v\}$ of distinct vertices called *edges*. A directed graph $G = (V, A)$ is a pair, where V is a set of vertices and A is a set of ordered pairs (u, v) of distinct vertices called *arcs*. Note that we do not allow loops and multiple edges or arcs. We use $V(G)$ and $E(G)$ ($A(G)$, respectively) to denote the set of vertices and the set of edges (set of arcs, respectively) of G. We write n and m to denote the number of vertices and edges (arcs, respectively) if this does not create confusion. For a (directed) graph G and a subset $X \subseteq V(G)$ of vertices, we write $G[X]$ to denote the subgraph of G induced by X. For a set of vertices S, $G - S$ denotes the (directed) graph obtained by deleting the vertices of S, that is, $G - S = G[V(G) \setminus S]$. We write $P = v_1 \cdots v_k$ to denote a *path* with the vertices $v_1, \dots, v_k$ and the edges $\{v_1, v_2\}, \dots, \{v_{k-1}, v_k\}$ (arcs $(v_1, v_2), \dots, (v_{k-1}, v_k)$, respectively); v_1 and v_k are the *end-vertices* of P and the vertices $v_2, \dots, v_{k-1}$ are *internal*. We consider only simple paths, that is, the vertices $v_1, \dots, v_k$ are distinct. We say that P is an (v_1, v_k)*-path*. The *length* of P, denoted by length(P), is the number of edges (arcs, respectively) in P. Two paths are *disjoint* if they have no common vertex and they are *internally disjoint* if no internal vertex of one path is a vertex of the other. For a (u, v)-path P_1 and a (v, w)-path P_2 that are internally disjoint, we denote by $P_1 \circ P_2$ the *concatenation* of P_1 and P_2. A vertex v is *reachable* from a vertex u in a (directed) graph G if G has a (u, v)-path. For $u, v \in V(G)$, $\text{dist}_G(u, v)$ denotes the *distance* between u and v in G, that is, the minimum number of edges (arcs, respectively) in an (u, v)-path. An undirected graph G is *connected* if for every two vertices u and v, G has a (u, v)-path. A directed graph G is *strongly connected* if for every two vertices u and v both u is reachable form v and v is reachable from u. For a positive integer k, an undirected (directed, respectively) graph G is *k-connected* (*k-strongly connected*, respectively) if $|V(G)| \geq k$ and $G - S$ is connected (strongly connected, respectively) for every $S \subseteq V(G)$ of size at most $k - 1$. For a directed graph G, by G^{rev} we denote the *reverse* of G, i.e. G^{rev} is a directed graph defined on the same set of vertices and the same set of arcs, but the direction of each arc in G^{rev} is reversed.

We use several known parameterized algorithms for finding long paths. First of all, let us recall the currently fastest deterministic algorithm for Longest Path on directed graphs due to Tsur [53].

Proposition 1 *([53]). There is a deterministic algorithm for* Longest Path *with running time* $2.554^k \cdot n^{\mathcal{O}(1)}$.

We also need the result of Fomin et al. [24] for the Long Directed (s, t)-Path problem. This problem asks, given a directed graph G, two vertices $s, t \in V(G)$, and an integer $k \geq 0$, whether G has an (s, t)-path of length at least k.

Proposition 2 *([24]).* Long Directed (s, t)-Path *can be deterministically solved in time* $4.884^k \cdot n^{\mathcal{O}(1)}$.

Clearly, both results hold for the variant of the problem on undirected graphs.

Finally, we use the result of Bezáková et al. [7] for the variant of Longest Detour whose task is, given a (directed) graph G, two vertices $s, t \in V(G)$, and an integer $k \geq 0$, decide whether G has an (s, t)-path of length *exactly* $\text{dist}_G(s, t) + k$.

Proposition 3 *([7]). There is a bounded-error randomized algorithm that solves* Exact Detour *on undirected graphs in time* $2.746^k \cdot n^{\mathcal{O}(1)}$ *and on directed graphs in time* $4^k \cdot n^{\mathcal{O}(1)}$*. For both undirected and directed graphs, there is a deterministic algorithm that runs in time* $6.745^k \cdot n^{\mathcal{O}(1)}$.

3. An FPT algorithm for finding detours

In this section, we show the first main result of our paper.

Theorem 1. *Let $\mathcal{C}$ be a class of directed graphs such that* 3-Disjoint Paths *can be solved in $f(n)$ time on $\mathcal{C}$. Then* Longest Detour *can be solved in* $45.5^k \cdot n^{\mathcal{O}(1)} + \mathcal{O}(f(n)n^2)$ *time by a deterministic algorithm and in* $23.86^k \cdot n^{\mathcal{O}(1)} + \mathcal{O}(f(n)n^2)$ *time by a bounded-error randomized algorithm when the input is restricted to graphs from $\mathcal{C}$.*

Proof. Let (G, s, t, k) be an instance of Longest Detour with $G \in \mathcal{C}$. For $k = 0$, the problem is trivial and we assume that $k \geq 1$. We also have that (G, s, t, k) is a trivial no-instance if t is not reachable from s. We assume from now that every vertex of G is reachable from s. Otherwise, we set $G := G[R]$, where R is the set of vertices of G reachable from s using the straightforward property that every (s, t)-path in G is a path in $G[R]$. Clearly, R can be constructed in $\mathcal{O}(n + m)$ time by the breadth-first search.

Using Proposition 3, we check in $6.745^{2k} \cdot n^{\mathcal{O}(1)}$ time by a deterministic algorithm (in $4^{2k} \cdot n^{\mathcal{O}(1)}$ time by a randomized algorithm, respectively) whether G has an (s, t)-path of length $\text{dist}_G(s, t) + \ell$ for some $k \leq \ell \leq 2k - 1$ by trying all values of

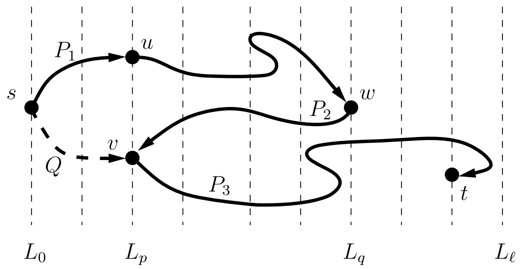

Fig. 1. The choice of the BFS-levels L_p and L_q, vertices u, v, and w, and the paths P_1, P_2, and P_3.

ℓ in this interval. We return a solution and stop if we discover such a path. Assume from now that this is not the case, that is, if (G, s, t) is a yes-instance, then the length of every (s, t)-path of length at least $\mathrm{dist}_G(s, t) + k$ is at least $\mathrm{dist}_G(s, t) + 2k$.

We perform the breadth-first search from s in G. For an integer $i \geq 0$, denote by L_i the set of vertices at distance i from s. Let ℓ be the maximum index such that $L_\ell \neq \emptyset$. Because every vertex of G is reachable from s, $V(G) = \bigcup_{i=0}^{\ell} L_i$. We call $L_0, \dots, L_\ell$ *BFS-levels*.

Our algorithm is based on structural properties of potential solutions. Suppose that (G, s, t, k) is a yes-instance and let a path P be a solution of minimum length, that is, P is an (s, t)-path of length at least $\mathrm{dist}_G(s, t) + k$ and among such paths the length of P is minimum. Denote by $p \in \{1, \dots, \ell\}$ the minimum index such that L_p contains at least two vertices of G. Such an index exists because if $|V(P) \cap L_i| \leq 1$ for all $i \in \{1, \dots, \ell\}$, then P is a shortest (s, t)-path by the definition of $L_0, \dots, L_\ell$ and the length of P is $\mathrm{dist}_G(s, t) < \mathrm{dist}_G(s, t) + k$ as $k \geq 1$. Let u be the first (in the path order) vertex of P in L_p and let $v \neq u$ be the second vertex of P that occurs in L_p. Denote by P_1, P_2, and P_3 the (s, u), (u, v), and (v, t)-subpath of P, respectively. Clearly, $P = P_1 \circ P_2 \circ P_3$. Let $q \in \{p, \dots, \ell\}$ be the maximum index such that P_2 contains a vertex of L_q. Then denote by w the first vertex of P_2 in L_q. See Fig. 1 for the illustration of the described configuration. We use this notation for a (hypothetical) solution throughout the proof of the theorem. The following claim is crucial for us.

Claim 1. *The length of P_2 is at least k.*

Proof of Claim 1. For the sake of contradiction, assume that the length of P_2 is less than k. Let Q be a shortest (s, v)-path in G. By the definition of BFS-levels, $V(Q) \subseteq L_0 \cup \dots \cup L_p$ and v is a unique vertex of Q in L_p. This implies that Q is internally vertex disjoint with P_3. Note that the length of Q is the same as the length of P_1 because P_1 contains exactly one vertex from each of the BFS levels $L_1, \dots, L_p$. Then $P' = Q \circ P_3$ is an (s, t)-path and

$$\begin{aligned} \mathrm{length}(P') &= \mathrm{length}(Q) + \mathrm{length}(P_3) = \mathrm{length}(P_1) + \mathrm{length}(P_3) \\ &= \mathrm{length}(P) - \mathrm{length}(P_2) \leq \mathrm{length}(P) - k. \end{aligned}$$

Recall that the length of every (s, t)-path of length at least $\mathrm{dist}_G(s, t) + k$ is at least $\mathrm{dist}_G(s, t) + 2k$. This means that $\mathrm{length}(P) \geq \mathrm{dist}_G(s, t) + 2k$ and, therefore, the length of P' is at least $\mathrm{dist}_G(s, t) + k$, that is, P' is a solution to the considered instance. However, $\mathrm{length}(P') < \mathrm{length}(P)$ because P_2 contains at least one arc. This contradicts the choice of P as a solution of minimum length. This completes the proof of the claim. $\square$

By Claim 1, solving Longest Detour on (G, s, t, k) boils down to identifying internally disjoint P_1, P_2, and P_3, where the length of P_2 is at least k.

First, we check whether we can find paths for $q - p \geq k - 1$. Notice that if $q - p \geq k - 1$, then for every internally disjoint (s, w)-, (w, v)-, and (v, t)-paths R_1, R_2, and R_3 respectively, their concatenation $R_1 \circ R_2 \circ R_3$ is an (s, t)-path of length at least $\mathrm{dist}_G(s, t) + k$. Recall that $G \in \mathcal{C}$ and p-Disjoint Paths can be solved in polynomial time on this graph class for $p = 3$. For every choice of two vertices $w, v \in V(G)$, we solve p-Disjoint Paths on the instance $(G, (s, w), (w, v), (v, s))$. Then if there are paths R_1, R_2, and R_3 forming a solution to this instance, we check whether $\mathrm{length}(R_1) + \mathrm{length}(R_2) + \mathrm{length}(R_3) \geq \mathrm{dist}_G(s, t) + k$. If this holds, we conclude that the path $R_1 \circ R_2 \circ R_3$ is a solution to the instance (G, s, t, k) of Longest Detour and return it. Assume from now that this is not the case, that is, we failed to find a solution of this type. Then we can complement Claim 1 by the following observation about our hypothetical solution P.

Claim 2. $q - p \leq k - 2$.

This means that we can assume that $k \geq 2$ and have to check whether we can find appropriate P_1, P_2, and P_3, where $V(P_2) \subseteq \bigcup_{i=p}^{p+k-2} L_i$. For this, we go over all possible choices of u. Note that the choice of u determines p, i.e., the index of the BFS-level containing u. We consider the following two cases for each considered choice of u.

Case 1. $t \in L_r$ for some $p \leq r \leq p + k - 2$ (see Fig. 2). Then $\mathrm{dist}_G(s, t) = r$ and (G, s, t, k) is a yes-instance if and only if $G[L_p \cup \dots \cup L_\ell]$ has a (u, t)-path S of length at least $(r - p) + k$ because the (s, u)-subpath of a potential solution should be

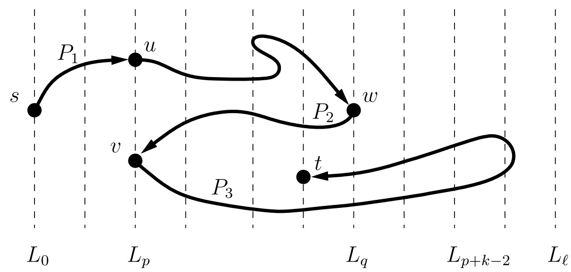

Fig. 2. The structure of paths P_1, P_2, and P_3 in Case 1.

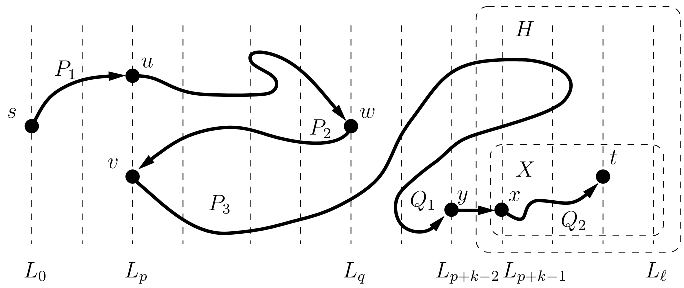

Fig. 3. The structure of paths P_1, P_2, and P_3 in Case 2.

a shortest (s,u)-path. Since $r-p \le k-2$, we have that $(r-p)+k \le 2k-2$ and we can find S in $4.884^{2k} \cdot n^{\mathcal{O}(1)}$ time by Proposition 2 if it exists. If we obtain S, then we consider an arbitrary shortest (s,u)-path S' in G and conclude that $S' \circ S$ is a solution. This completes Case 1.

Case 2. $t \in L_r$ for some $r \ge p+k-1$ (see Fig. 3). We again consider our hypothetical solution $P = P_1 \circ P_2 \circ P_3$. Let $H = G[L_{p+k-1} \cup \dots \cup L_\ell]$. Denote by X the set of vertices $z \in V(H)$ such that t is reachable from z in H. Denote by x the first vertex of P_3 in X. Clearly, such a vertex exists because $t \in X$. Moreover, $x \in L_{p+k-1}$ and its predecessor y in P_3 is in L_{p+k-2}. Otherwise, t would be reachable from $y \in V(H)$ in H contradicting the choice of x. Let Q_1 and Q_2 be the (v,y)- and (x,t)-subpaths of P_3. Then $P_3 = Q_1 \circ yx \circ Q_2$. We show one more claim about the hypothetical solution P.

Claim 3. $V(Q_1) \cap X = \emptyset$.

Proof of Claim 3. The proof is by contradiction. Assume that $z \in V(Q_1) \cap X$. Then t is reachable from z in H. However, x is the first vertex of P_3 with this property by the definition; a contradiction. $\square$

Notice that because $x \in X$, there is an (x,t)-path Q_2' with $V(Q_2') \subseteq X$. By Claim 3, Q_1 and Q_2' are disjoint. Since $X \subseteq L_{p+k-1} \cup \dots \cup L_\ell$, we have that $(V(P_1) \cup V(P_2)) \cap X = \emptyset$. In particular, Q_2' is disjoint with P_1 and P_2 as well. Let $P_3' = Q_1 \circ yx \circ Q_2'$. By Claim 1, $P' = P_1 \circ P_2 \circ P_3'$ is a solution because length$(P_2) \ge k$. This allows us to conclude that (G,s,t,k) has a solution (for the considered choice of u) if and only if there is $y \in L_{p+k-2}$ such that

(i) there is $x \in X$ such that $(y,x) \in A(G)$, and
(ii) the graph $G[L_p \cup \dots \cup L_\ell] - X$ has a (u,y)-path of length at least $2k-2$.

Our algorithm proceeds as follows. We construct the set X using the breadth-first search in $\mathcal{O}(n+m)$ time. Then for every $y \in L_{p+k-2}$ we check (i) whether there is $x \in X$ such that $(y,x) \in A(G)$, and (ii) whether $G[L_p \cup \dots \cup L_\ell] - X$ has a (u,y)-path S of length at least $2k-2$. To verify (ii), we apply Proposition 2 allowing to perform the check in $4.884^{2k} \cdot n^{\mathcal{O}(1)}$ time. If we find such a vertex y and path S, then to obtain a solution, we consider an arbitrary shortest (s,u)-path S' and an arbitrary (x,t) path S'' in $G[X]$. Then $P' = S' \circ S \circ yx \circ S''$ is a required solution to (G,s,t,k). This concludes the analysis in Case 2 and the construction of the algorithm.

The correctness of our algorithm has been proved simultaneously with its construction. The remaining task is to evaluate the total running time. Recall that we verify in $6.745^{2k} \cdot n^{\mathcal{O}(1)}$ time whether G has an (s,t)-path of length $\mathrm{dist}_G(s,t)+\ell$ for some $k \le \ell \le 2k-1$ by a deterministic algorithm, and we need $4^{2k} \cdot n^{\mathcal{O}(1)}$ time if we use a randomized algorithm. Then we construct the BFS-levels in linear time. Next, we consider $\mathcal{O}(n^2)$ choices of v and w and apply the algorithm for 3-Disjoint Paths $(G,(s,w),(w,v),(v,s))$ in $f(n)$ time. If we failed to find a solution so far, we proceed with $\mathcal{O}(n)$ possible choices of u and consider either Case 1 or 2 for each choice. In Case 1, we solve the problem in $4.884^{2k} \cdot n^{\mathcal{O}(1)}$ time. In Case 2, we construct X in $\mathcal{O}(n+m)$ time. Then for $\mathcal{O}(n)$ choices of y, we verify conditions (i) and (ii) in $4.884^{2k} \cdot n^{\mathcal{O}(1)}$ time.

Summarizing, we obtain that the total running time is $6.745^{2k} \cdot n^{\mathcal{O}(1)} + \mathcal{O}(f(n)n^2)$. Because $6.745^2 < 45.5$, we have that the deterministic algorithm runs in $45.5^k \cdot n^{\mathcal{O}(1)} + \mathcal{O}(f(n)n^2)$ time. Since $4^2 < 4.884^2 < 23.86$, we conclude that the problem can be solved in $23.86^k \cdot n^{\mathcal{O}(1)} + \mathcal{O}(f(n)n^2)$ time by a bounded-error randomized algorithm. □

In particular, combining Theorem 1 with the results of Cygan et al. [17], we obtain the following corollary.

Corollary 1. Longest Detour *can be solved in* $45.5^k \cdot n^{\mathcal{O}(1)}$ *time by a deterministic algorithm and in* $23.86^k \cdot n^{\mathcal{O}(1)}$ *time by a bounded-error randomized algorithm on planar directed graphs.*

Using the fact that p-Disjoint Paths can be solved in $\mathcal{O}(n^2)$ time by the results of Kawarabayashi, Kobayashi, and Reed [41], we immediately obtain the result for Longest Detour on undirected graphs. However, we can improve the running time of a randomized algorithm by tuning our algorithm for the undirected case.

Corollary 2. Longest Detour *can be solved in* $45.5^k \cdot n^{\mathcal{O}(1)}$ *time by a deterministic algorithm and in* $10.8^k \cdot n^{\mathcal{O}(1)}$ *time by a bounded-error randomized algorithm on undirected graphs.*

Proof. The deterministic algorithm is the same as in the directed case. To obtain a better randomized algorithm, we follow the algorithm from Theorem 1 and use the notation introduced in its proof. Let (G, s, t, k) be an instance of Longest Detour with $G \in \mathcal{C}$. We assume without loss of generality that $k \geq 1$ and G is connected. Using Proposition 3, we check in $2.746^{2k} \cdot n^{\mathcal{O}(1)}$ time by a randomized algorithm whether G has an (s,t)-path of length $\text{dist}_G(s,t) + \ell$ for some $k \leq \ell \leq 2k-1$. If we fail to find a solution this way, we construct the BFS-levels $L_0, \dots, L_\ell$.

Suppose that (G, s, t, k) is a yes-instance with a hypothetical solution P composed by the concatenation of P_1, P_2, and P_3 as in the proof of Theorem 1. Let also L_p and L_q be the corresponding BFS-levels. Observe that if $q - p \geq k/2$, then $\text{length}(P_2) \geq k$ because for every edge $\{x, y\}$ of G, x and y are either in the same BFS-level or in consecutive levels contrary to the directed case where we may have an arc (x, y) where $x \in L_i$ and $y \in L_j$ for arbitrary $j \in \{0, \dots, i\}$. Recall that for every choice of two vertices $w, v \in V(G)$, we solve p-Disjoint Paths on the instance $(G, (s, w), (w, v), (v, s))$ and try to find a solution to (G, s, t, k) by concatenating the solutions for these instances of p-Disjoint Paths. If we fail to find a solution this way, we can conclude now that $q - p \leq k/2 - 1$ improving Claim 2. Further, we pick u and consider two cases.

In Case 1, where $t \in L_r$ for some $p \leq r \leq p + k/2 - 1$, we now find a (u, t)-path S in $G[L_p \cup \dots \cup L_\ell]$ of length at least $(r - p) + k \leq 3k/2$ in $4.884^{3k/2} \cdot n^{\mathcal{O}(1)}$ time. If such a path exists, we obtain a solution.

In Case 2, where $t \in L_r$ for some $r \geq p + k/2$, we consider $H = G[L_{h+1} \cup \dots \cup L_\ell]$ for $h = p + \lceil k/2 \rceil$ and denote by X the set of vertices of the connected component of H containing X. Then for every $y \in L_h$ we check (i) whether there is $x \in X$ such that $\{y, x\} \in E(G)$, and (ii) whether $G[L_p \cup \dots \cup L_\ell] - X$ has a (u, y)-path S of length at least $k + \lceil k/2 \rceil$ in $4.884^{3k/2} \cdot n^{\mathcal{O}(1)}$ time. If such a path exists, we construct a solution containing it in the same way as in the directed case.

The running time analysis is essentially the same as in the proof of Theorem 1. The difference is that now we have that $2.746^2 \leq 4.884^{3/2} < 10.80$. This implies that the algorithm runs in $10.8^k \cdot n^{\mathcal{O}(1)}$ time. □

4. Longest path above diameter

In this section, we investigate the complexity of Longest Path above Diameter. It can be noted that this problem is NP-complete in general even for $k = 1$.

Proposition 4. Longest Path above Diameter *is* NP-*complete for* $k = 1$ *on undirected graphs.*

Proof. We show the claim by reducing the Hamiltonian Path, the classic NP-complete [30] problem. Let G be an undirected graph with $n \geq 2$ vertices. We construct the graph G' as follows (see Fig. 4).

- Construct a copy of G.
- Add a vertex u and make it adjacent to every vertex of the copy of G.
- Add two vertices s and t, and then (s, u) and (u, t) paths P_s and P_t, respectively, of length $n - 1$.

Notice that $\text{diam}(G) = \text{length}(P_s) + \text{length}(P_t) = 2n - 2$. It is easy to verify that G' has a path of length $2n - 1$ if and only if G has a path of length $n - 1$, that is, G is Hamiltonian. This completes the proof. □

Proposition 4 immediately implies that Longest Path above Diameter is NP-complete for $k = 1$ on strongly connected directed graphs. (We reduce the problem on undirected graphs to the directed variant by replacing each edge with the pair of arcs of opposite orientations.) Moreover, the reduction in Proposition 4 strongly relies on the fact that the constructed graph G' has an articulation point u. Hence, it is natural to investigate the problem further imposing connectivity constraints on the input graphs. And indeed, it can be easily seen that Longest Path above Diameter is FPT on 2-connected undirected graphs.

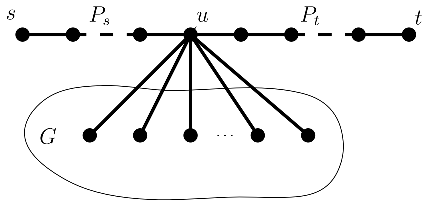

Fig. 4. Construction of G'.

Observation 1. Longest Path above Diameter *can be solved in time* $6.523^k \cdot n^{\mathcal{O}(1)}$ *on undirected 2-connected graphs.*

Proof. Let (G,k) be an instance of Longest Path above Diameter where G is 2-connected. If $d = \text{diam}(G) \le k$, we can solve the problem in time $2.554^{d+k} \cdot n^{\mathcal{O}(1)}$ by using the algorithm of Proposition 1 to check whether G has a path of length $d+k$. Note that $2.554^{d+k} \le 2.554^{2k} \le 6.523^k$. Otherwise, if $d > k$, consider a pair of vertices s and t with $\text{dist}_G(s,t) = d$. Because G is 2-connected, by Menger's theorem (see, e.g., [19]), G has a cycle C containing s and t. Since $\text{dist}_G(s,t) = d$ and $d \ge k+1$, the length of C is at least $d+k+1$. This implies that C contains a path of length $d+k$. $\square$

However, the arguments from the proof of Observation 1 cannot be translated to directed graphs. In particular, if a directed graph G is strongly 2-connected, this does not mean that for every two vertices u and v, G has a cycle containing u and v. We show the following theorem providing a full dichotomy for the complexity of Longest Path above Diameter on strongly 2-connected graphs.

Theorem 2. *On strongly* 2*-connected directed graphs,* Longest Path above Diameter *with* $k \le 4$ *can be solved in polynomial time, while for* $k \ge 5$*, it is* NP*-complete.*

In the remaining part of this section, we prove the theorem. In Subsection 4.1, we prove the algorithmic part, and in Subsection 4.2 we give the hardness proof.

4.1. Algorithm for $k \le 4$

We start with the positive part of Theorem 2. Note that it is sufficient to consider graphs with the diameter greater than a certain constant. This is because in graphs with smaller diameters, the problem can be solved in linear time by making use of the algorithm from [53]. The crucial part of the proof is encapsulated in the following lemma, which states that a path of length $\text{diam}(G)+4$ always exists in a strongly 2-connected graph G of sufficiently large diameter. To construct such a path, we take the diametral pair (s,t) and use the strong 2-connectivity property of the graph to find two disjoint (s,t)-paths and two disjoint (t,s)-paths in the graph. We then show that out of the several possible ways to comprise a path out of the parts of these four paths, at least one always obtains a path of the desired length. The most non-trivial case of this construction involves constructing two paths of length five, one ending in a vertex u that is at the distance three from s and the other starting in a vertex v from which we can reach t using three arcs. We then concatenate these two paths using a specific (u,v)-path in between. Since (s,t) is a diametral pair, the length of any (u,v)-path is at least diameter minus six, so the length of the concatenation is at least diameter plus four. The other cases are analyzed in a similar fashion.

Lemma 1. *Any strongly* 2*-connected directed graph* G *with* $\text{diam}(G) \ge 2^{3^{17}}$ *has a path of length* $\text{diam}(G)+4$.

Proof. Let $d = \text{diam}(G)$ and (s,t) be a pair such that $\text{diam}(G) = \text{dist}_G(s,t)$. Since G is strongly 2-connected, there exist two internally disjoint paths P_1 and P_2 from s to t. Denote the number of internal vertices of P_1 and P_2 by p_1 and p_2 respectively. Denote the internal vertices of P_i by $v_{i,1}, v_{i,2}, \ldots, v_{i,p_i}$ for each $i \in \{1,2\}$. Since $\text{diam}(G) = \text{dist}_G(s,t)$, we know that $p_i \ge d-1$. Therefore, if the length of P_i is at least $d+4$, then P_i is a path of length at least $\text{diam}(G)+4$ and we are done. Hence, from now on we assume that $p_i \le d+2$.

We say that a path between two arbitrary vertices in G is an *outer* path if no internal vertices belong to $V(P_1) \cup V(P_2)$. We now investigate sufficient conditions for G to contain a path of length at least $d+4$.

Claim 4. *If there exists an outer path in G going from $v_{i,j}$ to $v_{3-i,j'}$ with $j' \le j-3$, then there exists a path of length at least $d+4$ in G.*

Proof of Claim 4. Let T be such a path and consider the path $sP_iv_{i,j}Tv_{3-i,j'}P_{3-i}t$. This path is an (s,t)-path of length at least $j+1+(p_{3-i}+1-j') \ge d+(j-j')+1 \ge d+4$. $\square$

Claim 5. *If there exists an outer $(v_{i,j}, s)$-path in G with $i \in \{2\}$ and $j \geq 4$, then G has a path of length at least* $\mathrm{diam}(G) + 4$. *The same holds for an outer $(t, v_{i,j})$-path with $j \leq p_i - 3$.*

Proof of Claim 5. Assume that a $(v_{i,j}, s)$-path with described properties exists. Then concatenate the path $v_{i,1}P_iv_{i,j}$, the $(v_{i,j}, s)$-path and the path P_{3-i}. As all three paths are internally disjoint, we obtain a $(v_{i,1}, t)$-path of length at least $(j-1)+1+d = d+j \geq d+4$ in G as desired.

The case of a $(t, v_{i,j})$-path is symmetrical and we need to concatenate the path P_{3-i} with the $(t, v_{i,j})$-path and with the path $v_{i,j}P_iv_{i,p_i}$. The combined path is of length at least $d+1+(p_i-j) \geq d+4$. □

The following claim shows that we can find either a path of length $d+4$ or many outer paths connecting P_1 and P_2 in G.

Claim 6. *If G has no path of length at least $d+4$, then in any (t, s)-path and for every $i \in \{1, 2\}$ there are at least 8 outer subpaths going from an inner vertex of P_i to an inner vertex of P_{3-i}.*

Proof of Claim 6. Take an (t, s)-path Q. If Q has no inner vertices in $V(P_1) \cup V(P_2)$, then we can concatenate $v_{1,1}P_1t$ with Q and then with sP_2v_{2,p_2} and obtain a path of length at least $p_1 + p_2 + 1 > d + 4$.

Thus, Q should have at least one inner vertex in $V(P_1) \cup V(P_2)$. Denote all inner vertices of Q from $V(P_1) \cup V(P_2)$ by $q_1, q_2, \ldots, q_z$ in the order they appear on Q. Hence, tQq_1, q_zQs and q_kQq_{k+1} for every $k \in \{1, \ldots, z-1\}$ are outer paths in G.

Without loss of generality, we can assume that $q_1 \in V(P_1)$. Let r be the largest number such that $q_k \in V(P_1)$ for each $k \in \{1, \ldots, r\}$. First note that $r \leq 3$, otherwise we can concatenate P_2 with tQq_r and obtain a path of length at least $d + r \geq d + 4$. Suppose now that the length of q_kP_1t is greater than $3(r-1)$ for some $k \in \{1, \ldots, r\}$. The vertices $q_1, q_2, \ldots, q_{k-1}, q_{k+1}, \ldots, q_r$ split this path into $r-1$ parts, and the length of one of these parts is at least four. Hence, for some $a \in \{q_1, q_2, \ldots, q_r\}$ and $b \in \{q_1, q_2, \ldots, q_r, t\}$ the length of aP_1b is at least four and contains no inner vertex among $q_1, q_2, \ldots, q_r, t$. Then concatenate P_2 with $tQaP_1b$ without the vertex b. The obtained path is of length at least $d+1+3 = d+4$. Thus, we have that for each $k \in \{1, \ldots, r\}$ the vertex q_k is at distance at most $3(r-1) \leq 6$ from t on P_1. In particular, $q_r = v_{1,j}$ for $j \geq p_1 - 5$.

If $r = t$, i.e. there is no vertex from P_2 among $q_1, \ldots, q_r$, then we have an outer (q_r, s)-path in G. Since $q_r = v_{1,j}$ for $j \geq p_1 - 5 \geq 4$, by Claim 5 G has a path of length at least $d+4$. We now have that $z > r$ and $q_{r+1} \in V(P_2)$, i.e. Q alternates at least once between P_1 and P_2. We say that k is an alternation point in Q if $q_k \in V(P_i)$ and $q_{k+1} \in V(P_{3-i})$ for some $i \in \{1, 2\}$. As a convenient exception, we also consider $k = z$ as an alternation point in Q. Let $k_1 < k_2 < \ldots < k_{c-1} < k_c$ be the sequence of all such alternation points in Q. We know that $c \geq 1$ and $k_1 = r$ and $k_c = z$.

We show that for each $j \in \{1, \ldots, c\}$, for every $k \in \{1, \ldots, k_j\}$ the distance between q_k and t on P_1 or P_2 is at most 2^{3^j}.

We prove this by induction, where the case $j = 1$ has already been proved. Take $j > 1$ and assume that the induction hypothesis holds for $j-1$. Then we have an outer $(q_{k_{j-1}}, q_{k_{j-1}+1})$-path in G and for each $k \in \{1, k_{j-1}+1, k_j\}$ $q_k \in V(P_i)$. We know that $q_{k_{j-1}} = v_{3-i,j'}$ and $j' \geq p_{3-i} - 2^{3^{j-1}} + 1$ by induction. If $k_j - k_{j-1} > 2^{3^{j-1}} + 3$, then $sP_{3-i}q_{k_{j-1}}Qq_{k_j}$ is a path of length at least $p_{3-i} + 5 \geq d + 4$. Hence, we have that $k_j - k_{j-1} \leq 2^{3^{j-1}} + 3$. On the other hand, $q_{k_{j-1}+1} = v_{i,j''}$ and by Claim 4 we have that $j'' \geq j - 2 \geq p_{3-i} - 2^{3^{j-1}} - 1 \geq p_i - 3^{2^{j-1}} - 4$. If $k_j = k_{j-1} + 1$, then everything is proved since $v_{i,j''}$ is on the distance $p_i - j'' + 1 \leq 2^{3^{j-1}} + 5 \leq 2^{3^j}$ from t on P_i. From now we assume that $k_j - k_{j-1} > 1$.

Let $\ell, u \in \{1, \ldots, k_{j-1}+1, k_j\}$ be such that q_ℓ is the farthest from t on P_i and q_u is the closest to t on P_i. Then $q_\ell P_i q_u$ contains q_k for each $k \in \{1, \ldots, k_{j-1}+1, k_j\}$. The vertices $q_{k_{j-1}+1}, \ldots, q_{k_j}$ split this path into $k_j - (k_{j-1}+1)$ parts. If the length of $q_\ell P_i q_u$ is more than $(2^{3^{j-1}}+2)(k_j - k_{j-1} - 1)$, then one of these parts has the length at least $2^{3^{j-1}} + 3$. We denote the endpoints of this part by q_a and q_b, i.e. this part is $q_aP_iq_b$. Let us consider the path $sP_{3-i}q_{k_{j-1}}Qq_aP_iq_b$ without the vertex q_b. This path is of length at least $(p_{3-i} - 2^{3^{j-1}} + 1) + 1 + (2^{3^{j-1}} + 3) \geq p_{3-i} + 5 \geq d + 4$. Thus, we now assume that the length of $q_\ell P_i q_u$ is at most $(2^{3^{j-1}}+2)(k_j - k_{j-1} - 1) \leq (2^{3^{j-1}}+2)^2$.

Then for each $k \in \{1, \ldots, k_{j-1}+1, k_j\}$ the vertex q_k is on a distance at most $(2^{3^{j-1}}+2)^2$ from $q_{k_{j-1}+1}$ on P_i. Since $q_{k_{j-1}+1}$ is on a distance at most $2^{3^{j-1}} + 5$ from t, we have that for each such q_k the distance between q_k and t on P_i is at most

$$2^{3^{j-1}} + 5 + (2^{3^{j-1}}+2)^2 \leq 2^{2\cdot 3^{j-1}} + 5 \cdot 2^{3^{j-1}} + 9 \leq 3 \cdot 2^{2\cdot 3^{j-1}} \leq 2^{3^j},$$

as claimed.

Because for each $j \in \{1, \ldots, c\}$, for every $k \in \{1, \ldots, k_j\}$ the distance between q_k and t on P_1 or P_2 is at most 2^{3^j}, we obtain that $q_z = q_{k_c} = v_{i,j}$, where $i \in \{1, 2\}$ and $j \geq p_i - 2^{3^c} + 1$. Q also yields an outer (q_z, s)-path, and by Claim 5, we have that $j \leq 3$ or G has a path of length $d+4$. Then $p_i \leq 2^{3^c} + 2$. Since $p_i \geq d - 1$, we obtain that $2^{3^c} + 3 \geq d$ so $c \geq 17$. It follows that for each $i \in \{1, 2\}$ there are at least 8 outer $(v_{i,j}, v_{3-i,j'})$-paths in G, and all of them are subpaths of Q. □

The last claim in the proof shows that we can find two disjoint paths, one near s and one near t in G. We shall then combine them in a single path of length $d+4$ in G using paths from Claim 6.

Claim 7. *In G, there is either:*

1. *A path of length at least $d+4$, or*
2. *A path of length 5 ending in $v_{1,3}$ or in $v_{2,3}$ that avoids all vertices of form $v_{i,j}$ for $j>3$, and a path of length 5 starting in v_{1,p_1-2} or in v_{2,p_2-2} that avoids all vertices of form $v_{i,j}$ for $j<p_2-2$. These two paths do not share any common vertex.*

Proof of Claim 7. Since G is strongly 2-connected, there are two internally disjoint (s,t)-paths Q_1 and Q_2. By Lemma 6, either G contains a path of length $d+4$ or for each $i\in\{1,2\}$ Q_i contains at least four vertices in $V(P_1)\cup V(P_2)\setminus\{s,t\}$. Hence, for each $k\in\{1,2\}$, we have four outer paths tQ_ka_k, $a_kQ_kb_k$, d_kQ_ks, $c_kQ_kd_k$ in G. Note that all these eight paths are internally disjoint and the eight vertices a_k,b_k,c_k,d_k are pairwise distinct.

We first show how to construct a path of length 5 ending in $v_{i,3}$ for some $i\in\{1,2\}$, using the paths d_kQ_ks and $c_kQ_kd_k$. If $d_k=v_{i,j}$ for $k,i\in\{1,2\}$ and $j\geq 2$, then take the path $v_{i,1}P_iv_{i,j}Q_ksP_{3-i}v_{3-i,3}$. This is a path of length at least $(j-1)+1+3\geq j+3\geq 5$ ending in $v_{3-i,3}$ avoiding all vertices of form $v_{x,y}$ with $y>3$ as required. Hence, it is left to consider the case when for each $k\in\{1,2\}$ $d_k=v_{i,j}$ where $j\leq 2$. Without loss of generality, we assume that for each $k\in\{1,2\}$ $d_k=v_{k,1}$.

We now require the path $c_1Q_1d_1$ for the construction. Let $c_1=v_{i',j'}$ for some $i'\in\{1,2\}$ and $j'\in\{1,\dots,p_{i'}\}$. Suppose first that $i'=1$. If $j'\geq 4$ then take the path $v_{1,2}P_1v_{1,j'}Q_1v_{1,1}Q_1sP_2t$. This is a path of length at least $(j'-2)+2+d\geq d+j'\geq d+4$ in G. If $j'\leq 3$, then consider the path $v_{1,j'}Q_1v_{1,1}Q_1sP_2v_{2,3}$. This is a path of length at least five ending in $v_{2,3}$ that avoids all vertices $v_{x,y}$ with $y>3$.

It is left to consider $c_1=v_{2,j'}$. If $j'\geq 4$, then $c_1Q_1d_1$ is an outer $(v_{2,j'},v_{1,j})$-path with $j\leq j'-3$ and we are done by Claim 4. Hence, $j'\in\{2,3\}$. Then consider $sP_2v_{2,j'}Q_1v_{1,1}P_1v_{1,3}$. This path has the length at least $j'+1+2\geq 5$, ends in $v_{1,3}$, and avoids all vertices required. The part of the proof for paths ending in $v_{i,3}$ is complete. Note that all constructed paths can contain only vertices $v_{i,j}$ with $j\leq 3$, the vertex s, and inner vertices of the paths c_kQ_ks.

The proof for paths starting in v_{i,p_i-2} is symmetrical. The symmetry lies in that when we take the reversal of G, the roles of s and t exchange, and paths starting in v_{i,p_i-2} become paths ending in $v_{i,3}$. The role of the paths c_kQ_ks is taken by the reversals of the paths tQ_kb_k. Hence, we can consider the graph G^{rev}, exchange s and t and reenumerate each vertex $v_{i,j}:=v'_{i,p_i-j+1}$ for $i\in\{1,2\}$ and $j\in\{1,\dots,p_i\}$. By applying the previous proof to G^{rev}, we obtain either a path of length $d+4$ in G^{rev} or a path of length at least 5 ending in $v'_{i,3}$. Thus, in G, we obtain either a path of length at least $d+4$ or a path of length at least 5 ending in v_{i,p_i-2} and avoiding all required vertices. Note that the constructed paths in this part can contain only vertices $v_{i,j}$ with $j\geq p_i-2$, the vertex t and inner vertices of the paths tQ_kb_k. That is, the two paths of length five from the different parts of the proof do not share any common vertex. $\square$

To conclude the proof of Lemma 1, we combine results of Claim 6 and Claim 7 together. By Claim 7, if G does not contain a path of length $d+4$, there exists a path of length five ending in $v_{i,3}$ avoiding all $v_{x,y}$ with $y\geq 4$ and a path of length five starting in $v_{i',p_{i'}-2}$ avoiding all $v_{x,y}$ with $y\leq p_{i'}-3$. Denote these paths by R and R' respectively. If $i=i'$, then we take the concatenation $R\circ v_{i,3}P_iv_{i,p_i-2}\circ R'$. This is indeed a path without self-intersections as R and R' avoid all vertices of P_i between $v_{i,3}$ and v_{i,p_i-2} and are disjoint. The obtained path is of length $5+(p_i-2-3)+5\geq p_i+5\geq d+4$.

If $i'=3-i$, then we require an outer $(v_{i,y},v_{i',y'})$-path T with $y\geq 3$ and $y'\leq p_{i'}-2$ for the concatenation $R\circ v_{i,3}P_iv_{i,y}Tv_{i',y'}P_{i'}v_{i',p_{i'}-2}\circ R'$. This path can only share vertices $v_{i,3}$ and $v_{i',p_{i'}-2}$ with R and R'. Thus, we want T to avoid vertices $v_{i,1},v_{i,2},v_{i',p_{i'}},v_{i',p_{i'}-1}$ and all vertices in $V(R)\cup V(R')\setminus\{v_{i,3},v_{i',p_{i'}-2}\}$. These sum up to a total of 14 vertices that should be avoided. Since there are two internally disjoint (t,s)-paths in G, by Lemma 6 there are at least 16 outer paths in G going from an inner vertex of P_i to an inner vertex of $P_{i'}$. As each vertex to avoid lies on at most one path among these 16, at least two paths are suitable candidates for T. Take any of these candidates and denote it by T.

To estimate the length of T, consider the path $sP_iv_{i,y}Tv_{i',y'}P_{i'}t$. The length of this concatenation equals $y+\ell+(p_{i'}-y'+1)=(p_{i'}+1)-(y'-y)+\ell$, where ℓ is the length of T. Since the concatenation is an (s,t)-path we have $(p_{i'}+1)-(y'-y)+\ell\geq d$, so the length of T is at least $(y'-y)-(p_{i'}+1-d)$. The length of the path $v_{i,3}P_iv_{i,y}Tv_{i',y'}P_{i'}v_{i',p_{i'}-2}$ is at least $(y-3)+(y'-y)-(p_{i'}+1-d)+(p_{i'}-2-y')=d-6$. It follows that $R\circ v_{i,3}P_iv_{i,y}Tv_{i',y'}P_{i'}v_{i',p_{i'}-2}\circ R'$ is of length at least $d+4$ as required. $\square$

We note that the proof of Lemma 1 is constructive and can be turned into a polynomial-time algorithm finding a path of length $\mathrm{diam}(G)+4$ in a graph with the diameter at least $2^{3^{17}}$. For turning the proof into an algorithm, we require a procedure to find two internally disjoint (s,t)-paths or two internally disjoint (t,s)-paths in G. This can be done in polynomial time using any polynomial-time maximum flow algorithm. For $\mathrm{diam}(G)<2^{3^{17}}$, we use the color coding algorithm for Longest Path to find a path of constant length $\mathrm{diam}(G)+4$. The running time of this algorithm is linear in n. We obtain that Longest Path above Diameter with $k\leq 4$ can be solved in polynomial time on strongly 2-connected digraphs.

4.2. NP-hardness

We proceed to the second and negative result of Theorem 2. The general idea of the proof is similar to that of Proposition 4. We aim to take a path-like gadget graph, then take a sufficiently large Hamiltonian Path instance and connect it to the middle of the gadget. However, while in the general case, it suffices to simply take a path graph (Proposition 4), the strongly 2-connected case is much more technically involved. First, we need a family of gadget graphs that are strongly 2-connected and have arbitrarily large diameters, but each graph in the family does not have a path longer than the diameter plus four. This, in fact, is exactly a counterexample to the positive part of Theorem 2, as the existence of such a family of graphs proves that there cannot always be a path of length diameter plus four in a sufficiently large 2-connected directed graph. Additionally, for the reduction we need that graphs in this family behave like paths. By that, we mean that the length of the longest path that ends in the “middle” of the gadget is roughly half of the diameter. Constructing this graph family is the main technical challenge of the proof. After constructing the gadget graph family the proof is reasonably simple, as we take a 2-connected Hamiltonian Path instance, and connect it to the “middle” of a sufficiently large gadget graph. The connection is done by a simple 4-vertex connector gadget that ensures that the resulting graph is strongly 2-connected but only allows paths that alternate at most once between the gadget graph and the starting instance. The whole reduction is visualized in Fig. 6.

We start the proof with a construction of a family of directed graphs $G_1, G_2, \dots, G_\ell, \dots$ that are strongly 2-connected, while the longest path in G_ℓ has the length $\operatorname{diam}(G_\ell) + 4$ starting from some ℓ. We shall afterward use this graph for a many-to-one reduction from Hamiltonian Path to Longest Path above Diameter.

Construction of the graph G_ℓ. We construct G_ℓ for arbitrary $\ell \geq 1$. We require three types of gadgets for the construction that are shown in Fig. 5. The first two are the source and the sink gadgets, presented in Fig. 5a and Fig. 5c. They contain vertices s and t respectively. Note that the sink gadget is isomorphic to the reversal of the source gadget, and isomorphism is clear from the enumeration of the vertices of both gadgets. The third type of gadget, namely the hat gadget, is presented in Fig. 5b, and consists of ten vertices. To construct the graph G_ℓ for $\ell \geq 1$, we take one source gadget, $2\ell - 1$ hat gadgets, and one sink gadget. Then we identify the vertices s_8 and s_{14} with the vertices h_1 and h_4 of the first hat gadget respectively. Further, for each $i \in \{1, \dots, 2\ell - 2\}$, we identify the vertices h_3 and h_{10} of the i^{th} hat gadget respectively with the vertices h_4 and h_1 of the $(i+1)^{\text{th}}$ hat gadget. Finally, we identify the vertices h_3 and h_{10} of the last, $(2\ell - 1)^{\text{th}}$ gadget, with the vertices t_8 and t_{14} of the sink gadget. Thus, all gadgets are arranged into a chain and form a weakly connected graph. This is the graph G_ℓ, and later in this section, we prove that it is strongly 2-connected.

Paths P_1, P_2 and Q_1, Q_2. The four paths we describe are shown in Fig. 5d, where the graph G_2 is presented. By construction, there are two internally disjoint (s, t)-paths in G_ℓ. The path P_1 starts in s, then goes through vertices s_1, s_2 up to s_8 of the source gadget. Then the vertices h_2 and h_3 of the first hat gadget follow, then the vertices h_5, h_6 up to h_{10} of the second hat gadget follow on P_1, and so on. The path P_1 ends with vertices t_8, t_7 down to t_1 and, finally, the vertex t. The path P_2 starts in s, follows s_9 through s_{14}, and ends with t_{14} through t_9 in t. The paths Q_1 and Q_2 are two (t, s)-paths in G_ℓ. Their construction is shown in Fig. 5, where the arcs of each of them receive a specific color.

Note that we used $2\ell + 1$ gadgets to construct G_ℓ. Each of these gadgets is an induced subgraph of G_ℓ. Two gadgets can share either zero or two vertices, however, they cannot share any arc of G_ℓ. Moreover, each arc of G_ℓ belongs to exactly one gadget. From now on, by *gadgets* we refer to these $2\ell + 1$ induced subgraphs of G_ℓ.

Separating and containing gadgets. We say that a gadget *separates* two distinct vertices u and v in G_ℓ if there is no (u, v)-path and (v, u)-path in $G_\ell - X$, where X is the arc set of the gadget. We say that a gadget *strictly contains* vertex $v \in V(G_\ell)$ if v belongs to the vertex set of this gadget and does not belong to the vertex set of any other gadget in G_ℓ. Thus, there exist vertices that are not strictly contained in any gadget. We observe some trivial facts about gadgets.

Observation 2. *The following holds*

1. *For every vertex $v \in V(G_\ell)$, if a gadget strictly contains v, then this gadget separates v and every $u \in V(G_\ell) \setminus \{v\}$.*
2. *For every distinct $u, v \in V(G_\ell)$, if u and v are not separated by any gadget in G_ℓ, then $\{u, v\}$ is the intersection of the vertex sets of two gadgets in G_ℓ.*
3. *For every distinct $u, v \in V(G_\ell)$, if two gadgets both separate u and v in G_ℓ and share two vertices u' and v', then there is no path between u and v in $G_\ell - \{u', v'\}$.*

Lemma 2. *If a gadget separates u and v in G_ℓ and strictly contains neither u nor v, then the arcs of any (u, v)-path in G_ℓ induce a single path inside this gadget.*

Proof. First, since the gadget is a separating gadget, it should contain at least one arc of any (u, v)-path in G_ℓ. Hence, the arcs of the (u, v)-path induce one or several paths inside the gadget. The gadget is a hat gadget since the source and the sink gadget can be separating only for the vertices they strictly contain. Any path induced in the gadget by the arcs of the (u, v)-path has endpoints in $\{h_1, h_3, h_4, h_{10}\}$ since both u and v are not strictly contained in the gadget. Then one of the paths induced in the gadget should be a path between $\{h_1, h_4\}$ and $\{h_3, h_{10}\}$, otherwise the gadget is not separating.

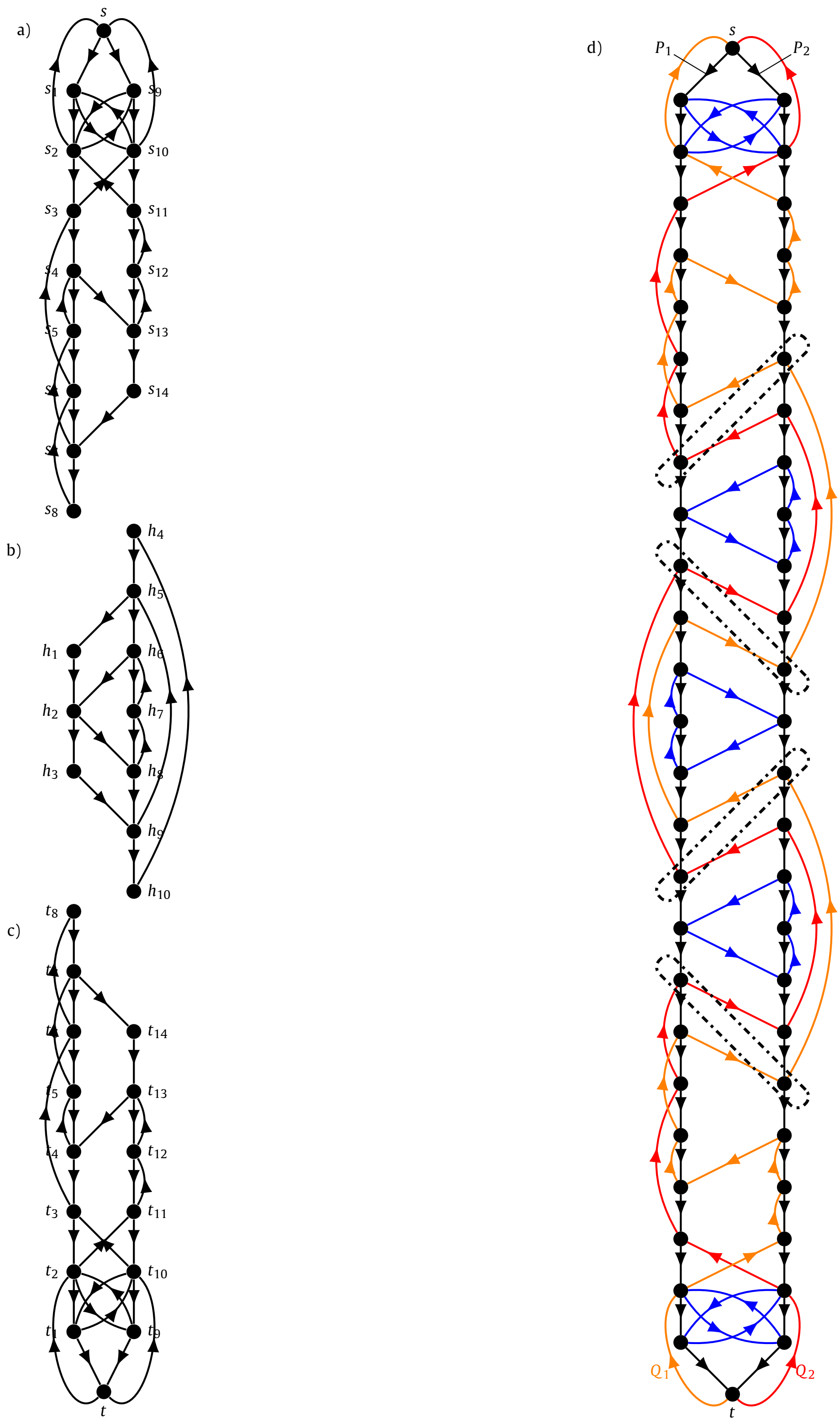

Fig. 5. Three types of gadgets used for the construction of G_ℓ, and the graph G_2. The orange arcs and the red arcs in G_2 are the arcs of Q_1 and Q_2 respectively. P_1 and P_2 are the (s,t)-paths on the left and the right respectively. Blue arcs are the arcs that belong to neither of P_1, P_2 and Q_1, Q_2. Dashed rounded rectangles show the boundaries of the gadgets used in the construction.

Hence, if there is more than one path induced, then there are exactly two paths induced, and the second path is also a path between $\{h_1, h_4\}$ and $\{h_3, h_{10}\}$. Since the gadget is separating and $\{h_1, h_4\}$ with $\{h_3, h_{10}\}$ are boundaries of the separation, we should have an odd number of paths going between them. Thus, there is only one induced path. □

Lemma 3. *G_ℓ is strongly 2-connected.*

Proof. First note that G_ℓ is strongly connected since every vertex of G_ℓ is reachable from s and t is reachable from every vertex of G_ℓ and there is a (t, s)-path in G_ℓ. We now need to show that $G_\ell - v$ is strongly connected for every vertex $v \in V(G_\ell)$.

Since G_ℓ is strongly connected and $G_\ell[N^-_{G_\ell}(s) \cup N^+_{G_\ell}(s)]$ is strongly connected, we have that $G_\ell - s$ is strongly connected. The same argument works for t, so $G_\ell - t$ is strongly connected as well. We shall now prove that $G_\ell - v$ is strongly connected for an arbitrary vertex $v \in V(G_\ell) \setminus \{s, t\}$. Since there are two disjoint (s, t)-paths in G_ℓ, t is reachable from s in (G_ℓ, v). Analogously, s is reachable from t in (G_ℓ, v). Hence, to prove that $G_\ell - v$ is strongly connected it is enough to show for every vertex $u \in V(G_\ell - v) \setminus \{s, t\}$ that there is an (s, u)-path or a (t, u)-path in G_ℓ and there is an (u, s)-path or a (u, t)-path in $G_\ell - v$.

Take a vertex $u \in V(G_\ell) \setminus \{v, s, t\}$. First, note that u lies on an (s, t)-path in G_ℓ. If v does not belong to this path, then this path remains in $G_\ell - v$ yielding an (s, u)-path and an (u, t)-path in G_ℓ as required. We now assume that v belongs to every (s, t)-path in G_ℓ containing u. Then exactly one of (s, u)-path and (u, t)-path exists in $G_\ell - v$.

Claim 8. *If $u \in V(G_\ell) \setminus \{s, t\}$ lies on Q_1 or Q_2 in G_ℓ, then there exists*

- *an (s, u)-path and a (t, u)-path that are internally disjoint;*
- *an (u, t)-path and an (u, s)-path that are internally disjoint.*

Proof of Claim 8. Let i be such that u belongs to P_i and j be such that u belongs to Q_i. Note that the vertices in $V(Q_j) \cap V(P_i)$ appear on Q_j in the order reverse to the order on P_i. Then the (t, u)-subpath of Q_j only uses vertices of P_i that appear after u on P_i. Thus, the (s, u)-subpath of P_i does not share any internal vertex with a (t, u)-subpath of Q_j. These two paths form the first pair of the claim.

To proceed with the second pair of paths, take the (u, s)-subpath of Q_j and the (u, t)-subpath of P_i. Since the (u, s)-subpath uses only those vertices of P_i that appear before u on P_i, these paths are also internally disjoint. □

Suppose that u lies on a (t, s)-path in G_ℓ. From the claim follows that u is reachable from s or t and s or t is reachable from u in $G_\ell - v$ and we are done. Hence, we can assume that u lies neither on Q_1 nor Q_2. Then u is a vertex in $N^+_{G_\ell}(s) \cup N^-_{G_\ell}(t)$ or a vertex in the inner cycle of a hat gadget, i.e. one of the vertices among the vertices h_2, h_6, h_7, h_8 of some hat gadget.

If u is a vertex in $N^+_{G_\ell}(s)$, then there are two internally disjoint paths from s to u and two internally disjoint paths from u to s in G_ℓ, so both (u, s)-path and (s, u)-path exist in $G_\ell - v$ and we are done. Analogously, if u is a vertex in $N^-_{G_\ell}(t)$, we have both (u, t)-path and (t, u)-path in $G_\ell - v$.

It is left to consider the case when u is a vertex of the inner cycle of a hat gadget. Note that there exist two internally disjoint paths starting in distinct vertices of $V(Q_i)$ and ending in u, for each $i \in \{1, 2\}$ in G_ℓ. Symmetrically, we have two internally disjoint paths starting in u and ending in distinct vertices of $V(Q_i)$, for each $i \in \{1, 2\}$ in G_ℓ. Since at least one of Q_1 and Q_2 exists in $G_\ell - v$, we have a (t, u)-path and an (u, s)-path in G_ℓ. The proof is complete. □

Lemma 4. $\mathrm{diam}(G_\ell) = \mathrm{dist}_{G_\ell}(s, t) = 8\ell + 10$.

Proof. Let d be the length of P_1 and P_2 in G_ℓ. First, note that $\mathrm{dist}_{G_\ell}(s, t)$ equals d. Indeed, $V(G_\ell) = V(P_1) \cup V(P_2)$, so an (s, t)-path in G_ℓ consists only of vertices of P_1 and P_2. Note that for every choice of $x, y \in \{1, 2\}$ and $i, j \in \{1, \dots, d+1\}$ with $i + 1 < j$, there is no arc in G_ℓ going from an i^{th} vertex on P_x to a j^{th} vertex on P_y. Hence, to reach t from s one should use at least d arcs in G_ℓ. Our goal is to prove that $\mathrm{dist}_{G_\ell}(u, v) \le d$ for each $u, v \in V(G_\ell)$.

Now denote the internal vertices in P_1 by $a_1, a_2, \dots, a_{d-1}$ in the order corresponding to P_1. Thus, $\mathrm{dist}_{G_\ell}(s, v) \le i$ and $\mathrm{dist}_{G_\ell}(v, t) \le d - i$ for each $i \in \{1, \dots, d-1\}$. Note that it holds that $\mathrm{dist}_{G_\ell}(s, v) = i$ and $\mathrm{dist}_{G_\ell}(v, t) = d - i$ since, otherwise, $\mathrm{dist}_{G_\ell}(s, t) \le \mathrm{dist}_{G_\ell}(s, v) + \mathrm{dist}_{G_\ell}(v, t) < i + (d - i) = d$. Analogously denote the internal vertices in P_2 by $b_1, b_2, \dots, b_{d-1}$. It holds that $\mathrm{dist}_{G_\ell}(s, b_i) = i$ and $\mathrm{dist}_{G_\ell}(b_i, t) = d - i$ for each $i \in \{1, \dots, d-1\}$.

Now consider the distance $\mathrm{dist}_{G_\ell}(t, s)$. We know that $\mathrm{dist}_{G_\ell}(t, s)$ is at most the length of Q_1 or Q_2. Observe that Q_1 uses four arcs in the sink gadget, three arcs in ℓ hat gadgets, one arc in $\ell - 1$ hat gadgets, and four arcs in the source gadget. Hence, the length of Q_1 is $4 + 3\ell + (\ell - 1) + 4 = 4\ell + 7$. As for Q_2, it uses six arcs in each of the sink and the source gadgets, one arc in ℓ hat gadgets, and three arcs in $\ell - 1$ hat gadgets. The length of Q_2 is $12 + \ell + 3(\ell - 1) = 4\ell + 9$. Hence, $\mathrm{dist}_{G_\ell}(t, s) \le 4\ell + 7 < d$.

Now take a vertex $v \in \{a_1, b_1, a_2, b_2, \dots, a_{d-1}, b_{d-1}\}$ and consider the distance $\mathrm{dist}_{G_\ell}(t, v)$. If v belongs to Q_1 or Q_2, then $\mathrm{dist}_{G_\ell}(t, v) < 4\ell + 9 < d$. If v belongs neither to Q_1 nor Q_2, then $v \in N^+_{G_\ell}(s)$ or $v \in N^-_{G_\ell}(t)$ or v is a vertex of the inner cycle of a hat gadget. In the first case, $\mathrm{dist}_{G_\ell}(t,v) \le \mathrm{dist}_{G_\ell}(t,s)+1 \le 4\ell+8 < d$. In the case $v \in N^-_{G_\ell}(t)$, $\mathrm{dist}_{G_\ell}(t,v)=2$. For v being a vertex of some inner cycle, note that v is reachable from a vertex of Q_1 or a vertex of Q_2 using a single arc. Hence, $\mathrm{dist}_{G_\ell}(t,v) < 4\ell+9+1 < d$.

To handle the distance $\mathrm{dist}_{G_\ell}(v,s)$, we note that the graph G_ℓ is isomorphic to the graph ${G_\ell}^{rev}$ with isomorphism $f: V(G_\ell) \to V(G_\ell)$, such that $f(s)=t$, $f(t)=s$, $f(a_i)=a_{d-i}$ and $f(b_i)=b_{d-i}$ for each $i \in \{1,\dots,d-1\}$. Thus, $\mathrm{dist}_{G_\ell}(v,s) = \mathrm{dist}_{{G_\ell}^{rev}}(f(v),t) = \mathrm{dist}_{G_\ell}(t,f(v))$. Since $f(v)$ is a_i or b_i for some $i \in \{1,\dots,d-1\}$, we have that $\mathrm{dist}_{G_\ell}(v,s) \le 4\ell+9 < d$.

It is left to prove that $\mathrm{dist}_{G_\ell}(u,v) \le d$ for every choice of $u,v \in \{a_1,b_1,a_2,b_2,\dots,a_{d-1},b_{d-1}\}$. It is easy to see that for every $1 \le i \le j < d$ $\mathrm{dist}_{G_\ell}(v,a_j)=j-i$ and $\mathrm{dist}_{G_\ell}(b_i,b_j)=j-i$, otherwise $\mathrm{dist}_{G_\ell}(s,t)<d$.

Note that for each $i \in \{1,\dots,d-1\}$ with i giving $1,2$ or 3 modulo 8, we have an arc from a_i to b_j in G_ℓ, where j is such that $|j-i|=1$. Then, for each $i \in \{1,\dots,d-1\}$ with i giving $6,7$ or 0 modulo 8 and $i \in \{1,2,3,d-1,d-2,d-3\}$, we have an arc from b_i to a_j in G_ℓ, where j is such that $|j-i|=1$. It follows that for each $i \in \{1,\dots,d-1\}$, we have an arc from a_j to b_t, where $i \le j \le i+5$ and $|t-j|=1$. Hence, $\mathrm{dist}_{G_\ell}(a_i,b_{j+1}) \le \mathrm{dist}_{G_\ell}(a_i,a_j)+1+\max\{\mathrm{dist}_{G_\ell}(b_{j-1},b_{j+1}),\mathrm{dist}_{G_\ell}(b_{j+1},b_{j+1})\} \le j-i+1+2 = j-i+3$. Then $\mathrm{dist}_{G_\ell}(a_i,b_{i+6}) \le \mathrm{dist}_{G_\ell}(a_i,b_{j+1})+\mathrm{dist}_{G_\ell}(b_{j+1},b_{i+6}) \le (j-i+3)+(i-j+5)=8$. Analogously, we have that $\mathrm{dist}_{G_\ell}(b_i,a_{i+6}) \le 8$. We conclude that $\mathrm{dist}_{G_\ell}(a_i,b_j) \le j-i+2 \le d$ and $\mathrm{dist}_{G_\ell}(b_i,a_j) \le j-i+2 \le d$ for each $i,j \in \{1,\dots,d-1\}$ such that $j-i \ge 6$.

It remains to consider distances of form $\mathrm{dist}_{G_\ell}(u,v)$, where $u \in \{a_i,b_i\}$ and $v \in \{a_j,b_j\}$, such that $j-i \le 5$. Then we have that both u and v are vertices of the same gadget or two adjacent gadgets. The diameter of any gadget is at most eight, so clearly the distance between u and v is at most 16, which is less than $8\ell+10$ for $\ell \ge 1$. □

Lemma 5. *Let $u,v \in V(G_\ell)$ be a pair of vertices in G_ℓ and let T be an (u,v)-path in G_ℓ. If T contains an arc of some gadget (either source, sink, or hat gadget) of G_ℓ, then*

- *this gadget contains u or v;*
- *this gadget separates u and v in G_ℓ;*
- *T contains exactly two arcs of this gadget. There are at most two such gadgets for T overall.*

Proof. We first consider the hat gadgets. Targeting towards a contradiction, suppose that there exists an (u,v)-path T and a hat gadget that does not satisfy the lemma statement. Denote the vertices of the hat gadget in G_ℓ by $h_1,\dots,h_{10}$, respectively to the definition of a hat gadget. The hat gadget does not separate u and v in G_ℓ and $u,v \notin \{h_1,h_2,\dots,h_{10}\}$. Finally, T uses more than two arcs of this gadget.

Note that since the gadget does not separate u and v in G_ℓ, the arcs of T in the gadget either form a path between h_1 and h_4, or a path between h_3 and h_{10}, or two disjoint paths, one going from $\{h_1,h_4\}$ to $\{h_3,h_{10}\}$, and one going in the opposite direction from $\{h_3,h_{10}\}$ to $\{h_1,h_4\}$. Other cases are not possible since $\{h_1,h_4\}$ and $\{h_3,h_{10}\}$ both are cuts of G_ℓ.

Consider first that this is a path between h_1 and h_4. First note that the only (h_4,h_1)-path inside the gadget uses exactly two arcs, (h_4,h_5) and (h_5,h_1), but T uses more than two arcs. If this is a (h_1,h_4)-path, then T contains a vertex x outside the gadget such that $(x,h_1) \in A(G_\ell)$. Symmetrically, T contains a vertex $y \notin \{h_1,\dots,h_{10}\}$ such that $(h_4,y) \in A(G_\ell)$, and x and y are distinct. However, the indegree and outdegree of both h_1 and h_4 are equal to two, so there is only one option for x and only one option for y in G_ℓ. However, from the construction of G_ℓ it is clear that both x and y should be a predecessor of h_1 on either P_1 or P_2 in G_ℓ. Hence, the case of a path between h_1 and h_4 is not possible.

The case of a path between h_3 and h_{10} is symmetrical. The only (h_3,h_{10})-path consists of only two arcs, and the (h_{10},h_3)-path is not possible since T cannot contain a (h_{10},h_3)-subpath, as the only outside ingoing neighbor of h_{10} is the only outside outgoing neighbor of h_3.

The only case left for the intersection of T and the arcs of the gadget is two disjoint paths. One path should start either in h_1 or h_4 and end either in h_3 or h_{10}, and the other should go in the opposite direction. Suppose that one of the paths starts in h_3. Then it should use the only outgoing arc (h_3,h_9). However, $\{h_3,h_{10}\}$ separates all paths going from $\{h_1,h_4\}$ to h_{10}, so a disjoint path in the other direction is not possible in this case. Hence, one of the paths should start in h_{10}. The only arc outgoing of h_{10} is (h_{10},h_4), so one of the paths is a (h_{10},h_4)-path and the other is an (h_1,h_3)-path inside the gadget. But then T should contain an (h_1,h_4)-subpath in G_ℓ. We already know that this is not possible.

It is left to consider the case when the source gadget or the sink gadget does not contain u and v, but T has at least three arcs of this gadget. We consider the case of the source gadget as the case of the sink gadget is symmetric. Since T neither starts nor ends in the source gadget, the arcs of T should induce a path between s_8 and s_{14} inside it. The only (s_{14},s_8)-path consists of just two arcs, so it should be an (s_8,s_{14})-path inside the gadget. Then T should contain an arc (x,s_8) and an arc (s_{14},y), where x,y are outside the gadget. The only choice for both x and y is the vertex h_5 of the first hat gadget in G_ℓ. This is a contradiction since x and y should be distinct.

For the very last sentence in the lemma statement, suppose that there are three gadgets containing an arc of T but not containing u and v and not separating u and v. Then one of the three gadgets separates the other two from each other, and this gadget is necessarily a hat gadget. Then the arcs of T should form at least one path between $\{h_1,h_4\}$ and $\{h_3,h_{10}\}$ inside this separating gadget, but this is not the case. □

Lemma 6. *For $\ell \geq 17$, the maximum path length in G_ℓ is $\mathrm{dist}_{G_\ell}(s,t)+4 = 8\ell+14$. Additionally, for a vertex $v \in G_\ell$ that is either h_6 or h_8 in the middle hat gadget (i.e. the ℓ-th hat gadget out of $2\ell-1$), the longest path in G_ℓ that ends in v has the length at most $4\ell+15$, and exactly $4\ell+15$ for h_8.*

Proof. Consider two vertices u and v and an (u,v)-path T. We show that the length of T is at most $d+4$, where $d = \mathrm{dist}_{G_\ell}(s,t) = 8\ell+10$. For any hat gadget not containing u and v but separating u and v, we know that the arcs of T form a path between $\{h_1, h_4\}$ and $\{h_3, h_{10}\}$ inside this gadget.

We first consider the case when $\mathrm{dist}_{G_\ell}(s,u) > \mathrm{dist}_{G_\ell}(s,v)$. Then in any separating hat gadget that does not contain u or v strictly, the arcs of T form a path from $\{h_3, h_{10}\}$ to $\{h_1, h_4\}$. Denote all separating gadgets by $H_1, H_2, \dots, H_p$ in the order in which T traverses them. Denote by x_i the number of arcs of T inside H_i for each $i \in \{1,\dots,p\}$. Note that each $x_i \in \{1,\dots,5\}$, as the longest path from $\{h_3, h_{10}\}$ to $\{h_1, h_4\}$ in H_i is an (h_3,h_1)-path of length five. However, if the path induced by T in H_i ends in h_1, then the path induced by T in H_{i+1} starts in h_{10}. A longest path starting in h_{10} is of length three if it ends in h_1, and is of length one if it ends in h_4. For each $i \in \{1,\dots,p\}$, let $c_i \in \{0,1\}$ be equal to 1 if the path induced by T in H_i ends in h_1, and to 0 otherwise. Additionally, let $c_0 = 1$ if the path induced by T in H_1 starts in h_3, otherwise $c_0 = 0$.

For each $i \in \{0,\dots,p-1\}$, consider the values of c_i and c_{i+1}. If $c_i = 0$, then the path induced by T in H_{i+1} starts in h_3, otherwise it starts in h_{10}. If $c_{i+1} = 0$, then the path induced by T in H_{i+1} ends in h_4, otherwise it ends in h_1. If $(c_i, c_{i+1}) = (0,0)$, then the path inside H_{i+1} is an (h_3,h_4)-path and has the length three. If $(c_i,c_{i+1}) = (0,1)$, then this path goes from h_3 to h_1 and has the length five. If $(c_i,c_{i+1}) = (1,0)$, then the path is an (h_{10},h_4)-path and has the length one. Finally, if the pair equals $(1,1)$, then inside H_{i+1} we have an (h_{10},h_1)-path of length three. Thus, the formula for x_{i+1} is $3-2c_i+2c_{i-1}$. We obtain that $\sum_{i=1}^{p} x_i = 3p+2c_0-2c_p \leq 3p+2$.

Now observe that apart from arcs inside separating gadgets, T can contain arcs inside gadgets containing u and v but not separating them plus four arcs in two additional gadgets. Since each gadget consists of at most 14 vertices, T can have at most 13 arcs inside gadgets containing u and v. The vertex u is either strictly contained in a gadget or is non-strictly contained in two adjacent gadgets. In the second case, one of the two gadgets is necessarily a separating hat gadget. Since the same holds v, there are at most two gadgets in G_ℓ that contain u or v but not separate them. Hence, the length of T is at most $(3p+2)+2\cdot 13+4 \leq 3p+32 \leq 3(2\ell-1)+32 \leq 6\ell+29$. For $\ell \geq 17$, we have that $6\ell+29 < 8\ell+14$, which is the desired bound on the length of T. For the second part of the statement, if v is strictly contained the middle hat gadget, then p is at most $\ell-1$, and the length of T is at most $3\ell+32 \leq 4\ell+15$.

We move on to the case $\mathrm{dist}_{G_\ell}(s,u) \leq \mathrm{dist}_{G_\ell}(s,v)$. If u and v are in the same gadget or are in two adjacent gadgets, then the length of T is at most $4\cdot 14+4 \leq 60 < d+4$. Hence, there is at least one hat gadget separating u and v but not containing u or v. Consider the subpath of T that is formed by arcs of the separating gadgets that do not strictly contain u or v. Let the starting point of this path be u' and the ending point be v'. Note that in any separating hat gadget, T induces a path from $h_i \in \{h_1,h_4\}$ to $h_j \in \{h_3,h_{10}\}$. Note that the length of such path always equals $\mathrm{dist}_{G_\ell}(s,h_j) - \mathrm{dist}_{G_\ell}(s,h_i)$. Hence, the length of the (u',v')-subpath of T is exactly $\mathrm{dist}_{G_\ell}(s,v') - \mathrm{dist}_{G_\ell}(s,u')$. It remains to estimate the length of the (u,u')-subpath and the (v',v)-subpath of T, consider the following expression for the length of the (u,v)-path:

$$\mathrm{length}(T) = \mathrm{length}\left(T_{(u,u')}\right) + \mathrm{dist}_{G_\ell}(s,v') - \mathrm{dist}_{G_\ell}(s,u') + \mathrm{length}\left(T_{(v',v)}\right) =$$
$$d + \left(\mathrm{length}\left(T_{(u,u')}\right) - \mathrm{dist}_{G_\ell}(s,u')\right) + \left(\mathrm{length}\left(T_{(v',v)}\right) - \mathrm{dist}_{G_\ell}(v',t)\right),$$

where we denote the (u,u')-subpath of T by $T_{(u,u')}$, and the (v',v)-subpath of T by $T_{(v',v)}$.

It suffices to show that the length of the (u,u')-subpath is at most $\mathrm{dist}_{G_\ell}(s,u')+2$, then, by symmetry, each of the last two terms above is at most two. There are two cases, either u belongs to the source gadget, or u belongs only to hat gadgets. We start with the first case, u' is then either s_8 or s_{14} in the source gadget. The following claim completes this case.

Claim 9. *In the source gadget, for $w \in \{s_8, s_{14}\}$, the longest path that ends in w has the length at most* $\mathrm{dist}(s,w)+2$.

Proof of Claim 9. Let us call the arcs that increase (resp. decrease) the distance to s forward (resp. backward) arcs. Observe that if a path that ends in w does not take backward arcs, its length is at most $\mathrm{dist}(s,w)$. We now consider the choice of the first backward arc of the potential path. To recall the vertex numeration in the source gadget, see Fig. 5a.

$\boxed{s_2 s}$ After s, the path has to proceed to s_{10} since $\{s_2, s_{10}\}$ is a cut that separates s from $\{s_8, s_{14}\}$. The longest choice for such a subpath starts in s_1 or s_9 and collects all vertices on one side of the cut, e.g. $s_1 s_2 s s_9 s_{10}$. Then the path has the only option to proceed to s_{14}, if $w = s_{14}$ this case is settled as the length of the path is at most $8 = \mathrm{dist}(s,s_{14})+2$. If $w = s_8$, then the path has to take the arc $s_{14}s_7$, and then the arc $s_7 s_8$; the arc $s_7 s_5$ cannot be taken since s_7 separates s_8 from s_5. Again, the length of the path is at most $10 = \mathrm{dist}(s,s_8)+2$.

$\boxed{s_{10} s}$ Identically to the previous case, the first part of the path ends in s_2 and takes at most 4 arcs inside the vertex set $\{s, s_1, s_2, s_9, s_{10}\}$. Then the path has to proceed along P_1 until it either takes the arc $s_4 s_{13}$, and then the argument is exactly the same as in the previous case, or proceeds straight to s_8, thus in the case $w = s_8$ yielding a path of length at most 10.

$\boxed{s_2 s_9}$ Exactly as in the case $\boxed{s_2 s}$, the path has to reach s_{10}, and then the analysis is the same.

$\boxed{s_{10} s_1}$ Symmetrically to the previous case, the analysis repeats the case $\boxed{s_{10} s}$.

$\boxed{s_3 s_{10}}$ After s_{10}, the path has to proceed to s_{11} as $\{s_3, s_{10}\}$ is a separator, the analysis regarding the subpath after s_{10} is identical to the case $\boxed{s_2 s}$. Observe that the subpath leading to s_3 cannot take both vertices s_1 and s_9 since s_2 together with s_{10} separates s_1 and s_9 from s_3. Therefore, prior to reaching s_{10}, the path takes at most 4 arcs, leading again to the total length of at most $\text{dist}(s, w) + 2$.

$\boxed{s_{11} s_2}$ Symmetrically to the previous case, the analysis reduces to the case $\boxed{s_{10} s}$.

$\boxed{s_6 s_3}$ Before s_6, only the vertices s_4 and s_5 can be taken since $s_6 s_3$ is the first backward arc. After s_3, the path either proceeds to s_{10}, or s_4 and then s_{13}. In the first case, the path has to go directly from s_{10} to w via forward arcs, yielding a length of at most 8 or 10 for $w = s_{14}$ and $w = s_8$ respectively. In the second case, no other backward arc can be taken as well, and the resulting path can only be shorter.

$\boxed{s_5 s_4}$ Since $s_5 s_4$ is the first backward arc on the path, the path has to start at s_5. Afterward, the arc $s_4 s_{13}$ has to be taken, and then only forward arcs to s_{14}, resulting in the path of length 3, or s_8, resulting in the length 5.

$\boxed{s_7 s_5}$ The target endpoint w cannot be s_8, as s_7 separates s_8 from s_5. If $w = s_{14}$, the path has to either continue through $s_5 s_4$ and $s_4 s_{13}$, or through $s_5 s_6$ and $s_6 s_3$. In the first case, the path has to finish immediately by taking the arc $s_{13} s_{14}$, and before s_7, only the vertex s_6 can be taken, resulting in the length of at most 5. In the second case, the path has to start at s_7 as s_6 is taken and s_{14} is the end of the path, and then similarly to the case $\boxed{s_6 s_3}$ the only option is to proceed from s_3 to s_{10} and then to s_{14} along forward arcs, resulting in the length 8, or from s_3 directly to s_{14} via forward arcs, resulting in the length 6.

$\boxed{s_8 s_6}$ The endpoint w must be s_{14}. After s_6, the path either goes through $s_6 s_3$, or through $s_6 s_7 s_5 s_4 s_{13} s_{14}$. In the latter case, s_8 has to be the starting vertex of the path and the length is 6. In the former, the starting vertex is either s_7 or s_8, and then the analysis is identical to the case where $s_6 s_3$ is the first backward arc.

$\boxed{s_{12} s_{11}}$ The vertex s_{12} has to be the starting point of the path. Afterward, the arc $s_{11} s_2$ has to be taken, and then only forward arcs. The length of the path is then exactly $\text{dist}(s, w)$.

$\boxed{s_{13} s_{12}}$ The path has to start at s_{13}, as $\{s_4 s_{13}\}$ separates s_{12} from w. The only possible continuation is $s_{12} s_{11} s_2 s_3 s_4 s_5 s_6 s_7 s_8$, resulting in the path of length 9 ending at s_8.

As the above cases cover all backward arcs in the source gadget, the proof of the claim is concluded. □

In the other case, u does not belong to the source gadget, so the first hat gadget does not separate u and v. Consider the topmost hat gadget that contains u. The vertex u' has to be either h_3 or h_{10} in this hat gadget since the next hat gadget necessarily separates u and v. Thus, $\text{dist}_{G_\ell}(s, u') \geq 10$. Now it suffices to show that the length of the (u, u')-subpath is at most twelve. From Lemma 5, it follows that this subpath lies completely inside the gadget, except for the two arcs of the gadget above that form a path from h_4 to h_1 through a vertex outside of the gadget. Therefore, the subpath visits at most eleven vertices, the number of vertices in a hat gadget plus one extra vertex, and thus cannot exceed the length of twelve. To complete the proof, we show that there is indeed a path of length $8\ell + 14$ in G_ℓ. The path proceeds as follows: start at the vertex s_9 of the source gadget, then go to s_{10} and s, then proceed along the path P_1 of length d until t is reached, and finally go to t_{10} and t_9.

For the second part of the statement, consider $v \in \{h_6, h_8\}$ in the middle hat gadget. Then v' is either h_4 or h_1 in the same gadget. From the above, the length of the (u, v')-subpath of T is at most $\text{dist}_{G_\ell}(s, v') + 2$. Going from v' to v, T may have arcs inside the middle hat gadget, and possibly two arcs in the next gadget that form a two-path between h_3 and h_{10}. Thus, the length of T is at most $\text{dist}_{G_\ell}(s, v') + 12$ since there are at most 10 available vertices to T after v'. If v' is h_4 in the gadget, then the length of T is at most $4\ell + 14$ since $\text{dist}_{G_\ell}(s, v')$ is $4\ell + 2$ in this case. It only remains to consider the case where v' is h_1 in the middle hat gadget, $\text{dist}_{G_\ell}(s, v') = 4\ell + 4$. It suffices to show that T cannot take simultaneously all the vertices in the hat gadget, and the two arcs of the next gadget. Assume that happens, then T has to go from h_1 to h_3 and then to h_{10} through the two additional arcs. This leaves only the option to proceed to h_4 and then along the remaining path. Thus, h_9 cannot be reached before h_6 or h_8, if the path is to collect all vertices. Finally, to see that there is a path of length $4\ell + 15$ ending at h_8, consider the following path. Start at s_1, go to s_2, then s, then proceed along P_2 until the vertex h_1 of the middle hat gadget is reached. From there, complete the path with the following sequence:

$$h_1 h_2 h_3 w h_{10} h_4 h_5 h_6 h_7 h_8,$$

where all the vertices are labeled corresponding to the hat gadget, and w is the available vertex of the next gadget. This completes the lemma. □

Now we are ready to prove the hardness result of Theorem 2.

Lemma 7. Longest Path above Diameter *on strongly* 2*-connected directed graphs is* NP*-complete for* $k \geq 5$*.*

Proof. For each $k \geq 5$, we present a reduction from Hamiltonian Path on undirected 2-connected graphs, for an intuitive illustration see Fig. 6. Take a Hamiltonian path instance H where $|V(H)| = n'$, we treat H as a directed graph where every undirected edge is replaced by two directed arcs going in opposite directions. Assume that n' has form $4\ell + (k-5)$ for an integer $\ell \geq k/4 + 17$. For each vertex $w \in V(H)$, we construct the following instance of Longest Path above Diameter. Consider a graph C that we call a *connector gadget*. The graph C has four vertices c_1, c_2, c_3, c_4, and four arcs c_3c_1, c_4c_2, c_1c_4, c_2c_3. The resulting instance of Longest Path above Diameter is the graph G constructed by taking disjoint instances of the graphs H, C, G_ℓ, and then associating the vertex c_2 of C with w in H, c_1 of C with an arbitrary other vertex of H, c_3 of C with the h_6 vertex of the middle hat gadget of G_ℓ, and c_4 with the h_8 vertex of the same gadget. This finishes the construction, and now we show the correctness of the reduction, that is, there is a Hamiltonian path in H if and only if at least one of the constructed graphs G has a path of length $\mathrm{diam}(G) + k$. In the following claims, we show that G shares most of the properties proved for G_ℓ above.

Claim 10. *G is strongly* 2*-connected.*

Proof of Claim 10. Clearly, G is strongly connected. Now, assume we remove a vertex v from G, v either belongs to an induced copy of H or an induced copy of G_ℓ. In the first case, any other vertex of H is still reachable from c_1 or c_2 (whichever is not removed) and vice versa since H is 2-connected. There is also a path from any vertex of G_ℓ to $\{c_1, c_2\} \setminus \{v\}$ and back since the graph $G[V(G_\ell) \cup \{c_1, c_2\} \setminus \{v\}]$ is unchanged. The second case is identical. □

Claim 11. $\mathrm{diam}(G) = \mathrm{diam}(G_\ell)$.

Proof of Claim 11. Clearly, $\mathrm{diam}(G) \geq \mathrm{diam}(G_\ell)$. To show the opposite direction, first, observe that $\mathrm{diam}(H) \leq \frac{n'}{2} = 2\ell + \frac{k-5}{2}$. That holds since, for any two vertices u and v in H, $\mathrm{dist}_H(u, v)$ is at most $\frac{|H|}{2}$, as there are two disjoint paths going from u to v, and it cannot be that they both are longer than $\frac{|H|}{2}$. For any two vertices u and v inside G_ℓ, the distance from u to v is unchanged from G_ℓ since it is impossible to go from u to $H - V(C)$ and then back to v in G_ℓ, and the vertices of C do not change the distances in G_ℓ. Finally, for a vertex $u \in V(G_\ell)$ and a vertex $v \in V(H)$, observe that a path from u to v necessarily goes through the arc wt, where either $w = c_3$, $t = c_1$, or $w = c_4$, $t = c_2$. The subpath (u, w) is a shortest such path inside G_ℓ, and by Lemma 6 its length is at most $4\ell + 15$. The subpath (t, v) is a shortest path inside H, and its length is at most $\mathrm{diam}(H) \leq 2\ell + \frac{k-5}{2}$. Thus, $\mathrm{dist}_G(u, v) \leq 6\ell + \frac{k-5}{2} + 16 \leq 8\ell + 10 = \mathrm{diam}(G_\ell)$ since by construction $2\ell \geq k/2 + 34$. The case $u \in V(H)$, $v \in V(G_\ell)$ is symmetrical. □

Claim 12. *The longest path in G has the length at least* $\mathrm{diam}(G) + k$ *if and only if there is a Hamiltonian path in H starting in w.*

Proof of Claim 12. By Lemma 6, no path inside G_ℓ has the length more than $\mathrm{diam}(G) + 4$. Since $|V(H)| = 4\ell + (k-5)$, there cannot be a path of length more than $4\ell + (k-6) < 8\ell + 10 + k$ inside H. Thus, if there is a path T longer than $\mathrm{diam}(G) + k$ in G, it must use vertices in both G_ℓ and H. By the structure of C any such path either lies completely inside H or G_ℓ while taking only one extra vertex of C, or crosses from G_ℓ to H through C only once (or, from H to G_ℓ, but this case is symmetrical). In the first case, if the path starts and ends in H, its length is at most $4\ell + (k-5)$. Now consider the case where T starts and ends in G_ℓ. If the vertex h_7 of the middle hat gadget does not lie on T, T can be transformed into a path T' of the same length that lies completely inside G_ℓ, by replacing the outer vertex of C (c_1 or c_2) by h_7. Otherwise, if T contains the vertex h_7, it has to start or end in this vertex, as the only two neighbors h_6 and h_8 of h_7 lie also on T separated by the outer vertex of C. Then by Lemma 6 the length of T is at most $4\ell + 18 < 8\ell + 10 + k$, as T is a path inside G_ℓ that starts or ends in h_6 or h_8 of the middle hat gadget, plus three extra arcs.

Therefore, the only option when T can have the length at least $\mathrm{diam}(G) + k$ is when it has the following structure: it starts at a vertex $u \in V(G_\ell)$, then continues inside G_ℓ until it takes the arc wt in C, and then takes a final subpath (t, v) inside H. Here either $w = c_3$, $t = c_1$, or $w = c_4$, $t = c_2$, and we drop the completely symmetrical case where the path goes from H to G_ℓ through C. By Lemma 6, the length of the (u, w)-subpath is at most $4\ell + 15$. Now if the length of T is at least $\mathrm{diam}(G) + k = 8\ell + 10 + k$, the subpath (t, v) must be a Hamiltonian path in H since $|V(H)| = 4\ell + (k-5)$, and the length of T is exactly $8\ell + 10 + k$.

In the other direction, if there is a Hamiltonian path P in H, consider its starting vertex w, and the instance of Longest Path above Diameter constructed with this choice of w. By Lemma 6, there is a path of length $4\ell + 15$ inside G_ℓ that ends in c_4, which is also the vertex h_8 of the middle hat gadget. Continuing this path through the arc c_4c_2 and then along the Hamiltonian path (recall that c_2 in C is w in H), we obtain a path of length $8\ell + 10 + k$. □

Clearly, the lemma follows from the three claims above. □

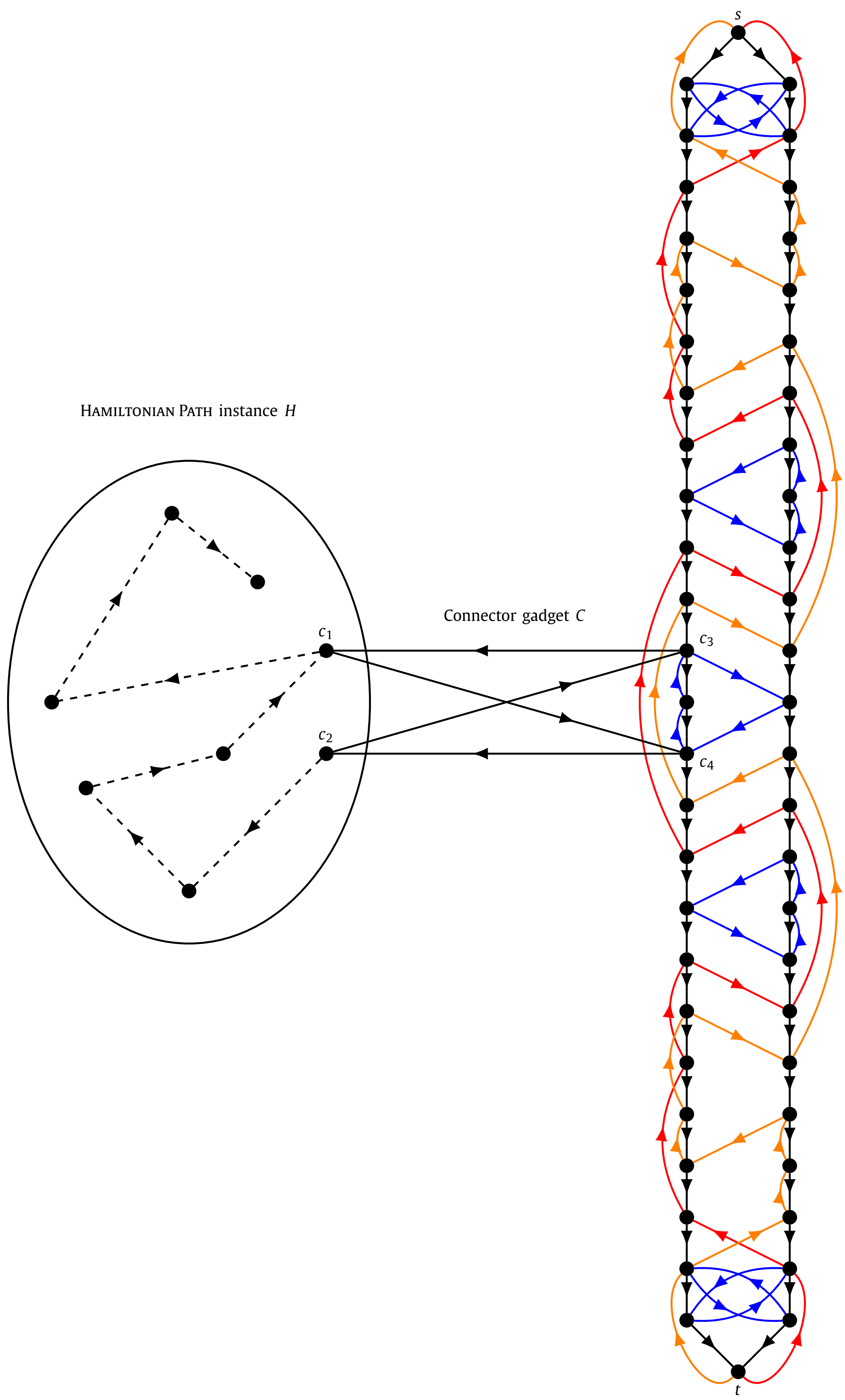

Fig. 6. The illustration of the hardness reduction of Lemma 7. While, in fact, the graph G_ℓ used in the reduction must have $\ell \geq 17$, here we use the graph G_2 for clarity.

5. Conclusion

We proved that if $\mathcal{C}$ is a class of directed graphs such that p-Disjoint Paths is in P on $\mathcal{C}$ for $p = 3$, then Longest Detour is FPT on $\mathcal{C}$. Very recently, this result was improved by Jacob, Włodarczyk, and Zehavi [38], who showed that Longest Detour is FPT on $\mathcal{C}$ whenever p-Disjoint Paths is polynomial on the class for $p = 2$. However p-Disjoint Paths is NP-complete on directed graphs for every fixed $p \geq 2$ [29]. This leaves open the question of Bezáková et al. [7] about parameterized complexity of Longest Detour on general directed graphs. Even the complexity (P versus NP) of deciding whether a directed graph contains an (s,t)-path longer than $\text{dist}_G(s,t)$ (the case of $k = 1$) remains open. Notice that Longest Detour is not equivalent to p-Disjoint Paths for $p = 2$ and, therefore, the hardness of p-Disjoint Paths does not imply the hardness of Longest Detour.

Our result implies, in particular, that Longest Detour is FPT on planar directed graphs. The result for planar was recently improved by Hatzel, Majewski, Pilipczuk, and Sokolowski [36] who followed the same ideas but gave a simpler and faster algorithm that relies on planarity. There are other classes of directed graphs on which p-Disjoint Paths is tractable for fixed p (see, e.g., the book of Bang-Jensen and Gutin [3]). For example, by Chudnovsky, Scott, and Seymour [14], p-Disjoint Paths can be solved in polynomial time for every fixed p on semi-complete directed graphs. Together with Theorem 1, this implies that Longest Detour is FPT on semi-complete directed graphs and tournaments. However, from what we know, these results could be too weak in the following sense. Using the structural results of Thomassen [52], Bang-Jensen, Manoussakis, and Thomassen in [4] gave a polynomial-time algorithm to decide whether a semi-complete directed graph has a Hamiltonian (s,t)-path for two given vertices s and t. Thus the real question is whether Longest Detour is in P on semi-complete directed graphs or tournaments.

The second part of our results is devoted to Longest Path above Diameter. We proved that this problem is NP-complete for general graphs for $k = 1$ and showed that it is in FPT when the input graph is undirected and 2-connected. We established the complexity dichotomy for Longest Path above Diameter for the case of strongly 2-connected directed graphs by showing that the problem can be solved in polynomial time for $k \leq 4$ and is NP-complete for $k \geq 5$. This naturally leaves an open question for larger values of strong connectivity. The computational complexity of Longest Path above Diameter on t-strongly connected graphs for $t \geq 3$ is open. For a very concrete question, is there a polynomial algorithm for Longest Path above Diameter with $k = 5$ on graphs of strong connectivity 3?

CRediT authorship contribution statement

Fedor V. Fomin: Conceptualization, Methodology, Investigation, Writing. Petr A. Golovach: Conceptualization, Methodology, Investigation, Writing. William Lochet: Conceptualization, Methodology, Investigation, Writing. Danil Sagunov: Conceptualization, Methodology, Investigation, Writing. Saket Saurabh: Conceptualization, Methodology, Investigation, Writing. Kirill Simonov: Conceptualization, Methodology, Investigation, Writing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

References

[1] Noga Alon, Raphael Yuster, Uri Zwick, Color-coding, J. ACM 42 (4) (1995) 844–856, https://doi.org/10.1145/210332.210337.
[2] Noga Alon, Gregory Gutin, Eun Jung Kim, Stefan Szeider, Anders Yeo, Solving MAX-r-SAT above a tight lower bound, in: Proceedings of the 21st Annual ACM-SIAM Symposium on Discrete Algorithms, SODA, SIAM, 2010, pp. 511–517.
[3] Jørgen Bang-Jensen, Gregory Z. Gutin, Digraphs – Theory, Algorithms and Applications, second edition, Springer Monographs in Mathematics, Springer, ISBN 978-1-84800-997-4, 2009.
[4] Jørgen Bang-Jensen, Yannis Manoussakis, Carsten Thomassen, A polynomial algorithm for hamiltonian-connectedness in semicomplete digraphs, J. Algorithms 13 (1) (1992) 114–127, https://doi.org/10.1016/0196-6774(92)90008-Z.
[5] Eli Berger, Paul Seymour, Sophie Spirkl, Finding an induced path that is not a shortest path, Discrete Math. 344 (7) (2021) 112398.
[6] Ivona Bezáková, Radu Curticapean, Holger Dell, Fedor V. Fomin, Finding detours is fixed-parameter tractable, in: Proceedings of the 44th International Colloquium on Automata, Languages, and Programming, ICALP, in: LIPIcs, vol. 80, Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, 2017, pp. 54:1–54:14.
[7] Ivona Bezáková, Radu Curticapean, Holger Dell, Fedor V. Fomin, Finding detours is fixed-parameter tractable, SIAM J. Discrete Math. 33 (4) (2019) 2326–2345, https://doi.org/10.1137/17M1148566.
[8] Andreas Björklund, Thore Husfeldt, Petteri Kaski, Mikko Koivisto, Narrow sieves for parameterized paths and packings, J. Comput. Syst. Sci. (ISSN 0022-0000) 87 (2017) 119–139, https://doi.org/10.1016/j.jcss.2017.03.003.
[9] Hans L. Bodlaender, On linear time minor tests with depth-first search, J. Algorithms 14 (1) (1993) 1–23, https://doi.org/10.1006/jagm.1993.1001.
[10] Hans L. Bodlaender, Marek Cygan, Stefan Kratsch, Jesper Nederlof, Deterministic single exponential time algorithms for connectivity problems parameterized by treewidth, Inf. Comput. 243 (2015) 86–111, https://doi.org/10.1016/j.ic.2014.12.008.

[11] Jianer Chen, Songjian Lu, Sing-Hoi Sze, Fenghui Zhang, Improved algorithms for path, matching, and packing problems, in: Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms, SODA, SIAM, 2007, pp. 298–307.

[12] Jianer Chen, Joachim Kneis, Songjian Lu, Daniel Mölle, Stefan Richter, Peter Rossmanith, Sing-Hoi Sze, Fenghui Zhang, Randomized divide-and-conquer: improved path, matching, and packing algorithms, SIAM J. Comput. (ISSN 0097-5397) 38 (6) (2009) 2526–2547, https://doi.org/10.1137/080716475.

[13] Yung-Chung Chiu, Hsueh-I. Lu, Blazing a trail via matrix multiplications: a faster algorithm for non-shortest induced paths, in: Petra Berenbrink, Benjamin Monmege (Eds.), 39th International Symposium on Theoretical Aspects of Computer Science, STACS 2022, March 15–18, 2022, Marseille, France (Virtual Conference), in: LIPIcs, vol. 219, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022, pp. 23:1–23:16.

[14] Maria Chudnovsky, Alex Scott, Paul D. Seymour, Excluding pairs of graphs, J. Comb. Theory, Ser. B 106 (2014) 15–29, https://doi.org/10.1016/j.jctb.2014.01.001.

[15] Robert Crowston, Mark Jones, Gabriele Muciaccia, Geevarghese Philip, Ashutosh Rai, Saket Saurabh, Polynomial kernels for lambda-extendible properties parameterized above the Poljak-Turzik bound, in: IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS, in: LIPIcs, vol. 24, Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, 2013, pp. 43–54.

[16] Marek Cygan, Jesper Nederlof, Marcin Pilipczuk, Michał Pilipczuk, Johan M.M. van Rooij, Jakub Onufry Wojtaszczyk, Solving connectivity problems parameterized by treewidth in single exponential time, in: Proceedings of the 52nd Annual Symposium on Foundations of Computer Science, FOCS, IEEE, 2011, pp. 150–159.

[17] Marek Cygan, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, The planar directed k-vertex-disjoint paths problem is fixed-parameter tractable, in: 54th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013, 26–29 October 2013, Berkeley, CA, USA, IEEE Computer Society, 2013, pp. 197–206.

[18] Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, Saket Saurabh, Parameterized Algorithms, Springer, ISBN 978-3-319-21274-6, 2015.

[19] Reinhard Diestel, Graph Theory, 4th edition, Graduate Texts in Mathematics, vol. 173, Springer, ISBN 978-3-642-14278-9, 2012.

[20] Rodney G. Downey, Michael R. Fellows, Fundamentals of Parameterized Complexity, Texts in Computer Science, Springer, ISBN 978-1-4471-5558-4, 2013.

[21] Fedor V. Fomin, Petteri Kaski, Exact exponential algorithms, Commun. ACM 56 (3) (2013) 80–88, https://doi.org/10.1145/2428556.2428575.

[22] Fedor V. Fomin, Daniel Lokshtanov, Fahad Panolan, Saket Saurabh, Efficient computation of representative families with applications in parameterized and exact algorithms, J. ACM 63 (4) (2016) 29:1–29:60, https://doi.org/10.1145/2886094.

[23] Fedor V. Fomin, Daniel Lokshtanov, Fahad Panolan, Saket Saurabh, Representative families of product families, ACM Trans. Algorithms 13 (3) (2017) 36:1–36:29, https://doi.org/10.1145/3039243.

[24] Fedor V. Fomin, Daniel Lokshtanov, Fahad Panolan, Saket Saurabh, Meirav Zehavi, Long directed (s,t)-path: FPT algorithm, Inf. Process. Lett. 140 (2018) 8–12, https://doi.org/10.1016/j.ipl.2018.04.018.

[25] Fedor V. Fomin, Petr A. Golovach, Daniel Lokshtanov, Fahad Panolan, Saket Saurabh, Meirav Zehavi, Going far from degeneracy, SIAM J. Discrete Math. 34 (3) (2020) 1587–1601, https://doi.org/10.1137/19M1290577.

[26] Fedor V. Fomin, Petr A. Golovach, Daniel Lokshtanov, Fahad Panolan, Saket Saurabh, Meirav Zehavi, Multiplicative parameterization above a guarantee, ACM Trans. Comput. Theory 13 (3) (2021) 18:1–18:16, https://doi.org/10.1145/3460956.

[27] Fedor V. Fomin, Petr A. Golovach, Danil Sagunov, Kirill Simonov, Longest cycle above Erdős-Gallai bound, in: Shiri Chechik, Gonzalo Navarro, Eva Rotenberg, Grzegorz Herman (Eds.), 30th Annual European Symposium on Algorithms, ESA 2022, September 5–9, 2022, Berlin/Potsdam, Germany, in: LIPIcs, vol. 244, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022, pp. 55:1–55:15.

[28] Fedor V. Fomin, Petr A. Golovach, Danil Sagunov, Kirill Simonov, Algorithmic extensions of Dirac's theorem, in: Joseph (Seffi) Naor, Niv Buchbinder (Eds.), Proceedings of the 2022 ACM-SIAM Symposium on Discrete Algorithms, SODA 2022, Virtual Conference, Alexandria, VA, USA, January 9–12, 2022, SIAM, 2022, pp. 406–416.

[29] Steven Fortune, John E. Hopcroft, James Wyllie, The directed subgraph homeomorphism problem, Theor. Comput. Sci. 10 (1980) 111–121, https://doi.org/10.1016/0304-3975(80)90009-2.

[30] M.R. Garey, David S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, W. H. Freeman, ISBN 0-7167-1044-7, 1979.

[31] Shivam Garg, Geevarghese Philip, Raising the bar for vertex cover: fixed-parameter tractability above a higher guarantee, in: Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA, SIAM, 2016, pp. 1152–1166.

[32] Gregory Gutin, Eun Jung Kim, Michael Lampis, Valia Mitsou, Vertex cover problem parameterized above and below tight bounds, Theory Comput. Syst. 48 (2) (2011) 402–410, https://doi.org/10.1007/s00224-010-9262-y.

[33] Gregory Gutin, Leo van Iersel, Matthias Mnich, Anders Yeo, Every ternary permutation constraint satisfaction problem parameterized above average has a kernel with a quadratic number of variables, J. Comput. Syst. Sci. 78 (1) (2012) 151–163, https://doi.org/10.1016/j.jcss.2011.01.004.

[34] Gregory Z. Gutin, Viresh Patel, Parameterized traveling salesman problem: beating the average, SIAM J. Discrete Math. 30 (1) (2016) 220–238.

[35] Gregory Z. Gutin, Arash Rafiey, Stefan Szeider, Anders Yeo, The linear arrangement problem parameterized above guaranteed value, Theory Comput. Syst. 41 (3) (2007) 521–538, https://doi.org/10.1007/s00224-007-1330-6.

[36] Meike Hatzel, Konrad Majewski, Michal Pilipczuk, Marek Sokolowski, Simpler and faster algorithms for detours in planar digraphs, in: Kavitha Telikepalli, Kurt Mehlhorn (Eds.), 2023 Symposium on Simplicity in Algorithms, SOSA 2023, Florence, Italy, January 23–25, 2023, SIAM, 2023, pp. 156–165.

[37] Falk Hüffner, Sebastian Wernicke, Thomas Zichner, Algorithm engineering for color-coding with applications to signaling pathway detection, Algorithmica 52 (2) (2008) 114–132, https://doi.org/10.1007/s00453-007-9008-7.

[38] Ashwin Jacob, Michal Wlodarczyk, Meirav Zehavi, Long directed detours: reduction to 2-disjoint paths, CoRR, arXiv:2301.06105 [abs], 2023, https://doi.org/10.48550/arXiv.2301.06105.

[39] Bart M.P. Jansen, László Kozma, Jesper Nederlof, Hamiltonicity below Dirac's condition, in: Proceedings of the 45th International Workshop on Graph-Theoretic Concepts in Computer Science, WG, in: Lecture Notes in Computer Science, vol. 11789, Springer, 2019, pp. 27–39.

[40] Ken-ichi Kawarabayashi, Stephan Kreutzer, The directed grid theorem, in: Proceedings of the 47th Annual ACM Symposium on Theory of Computing, STOC, ACM, 2015, pp. 655–664.

[41] Ken-ichi Kawarabayashi, Yusuke Kobayashi, Bruce A. Reed, The disjoint paths problem in quadratic time, J. Comb. Theory, Ser. B 102 (2) (2012) 424–435, https://doi.org/10.1016/j.jctb.2011.07.004.

[42] Joachim Kneis, Daniel Mölle, Stefan Richter, Peter Rossmanith, Divide-and-color, in: Proceedings of the 32nd International Workshop on Graph-Theoretic Concepts in Computer Science, WG, in: Lecture Notes in Computer Science, vol. 4271, Springer, 2006, pp. 58–67.

[43] Tuukka Korhonen, A single-exponential time 2-approximation algorithm for treewidth, in: 62nd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2021, Denver, CO, USA, February 7–10, 2022, IEEE, 2021, pp. 184–192.

[44] Ioannis Koutis, Faster algebraic algorithms for path and packing problems, in: Proceedings of the 35th International Colloquium on Automata, Languages and Programming, ICALP, in: Lecture Notes in Computer Science, vol. 5125, Springer, 2008, pp. 575–586.

[45] Ioannis Koutis, Ryan Williams, Algebraic fingerprints for faster algorithms, Commun. ACM 59 (1) (2016) 98–105, https://doi.org/10.1145/2742544.

[46] Daniel Lokshtanov, N.S. Narayanaswamy, Venkatesh Raman, M.S. Ramanujan, Saket Saurabh, Faster parameterized algorithms using linear programming, ACM Trans. Algorithms 11 (2) (2014) 15:1–15:31, https://doi.org/10.1145/2566616.

[47] Meena Mahajan, Venkatesh Raman, Parameterizing above guaranteed values: MaxSat and MaxCut, J. Algorithms 31 (2) (1999) 335–354, https://doi.org/10.1006/jagm.1998.0996.

[48] Meena Mahajan, Venkatesh Raman, Somnath Sikdar, Parameterizing above or below guaranteed values, J. Comput. Syst. Sci. 75 (2) (2009) 137–153, https://doi.org/10.1016/j.jcss.2008.08.004.

[49] Burkhard Monien, How to find long paths efficiently, in: Analysis and Design of Algorithms for Combinatorial Problems, in: North-Holland Math. Stud., vol. 109, North-Holland, Amsterdam, 1985, pp. 239–254.

[50] Neil Robertson, Paul D. Seymour, Graph minors. XIII. The disjoint paths problem, J. Comb. Theory, Ser. B 63 (1) (1995) 65–110, https://doi.org/10.1006/jctb.1995.1006.

[51] Alexander Schrijver, Finding k disjoint paths in a directed planar graph, SIAM J. Comput. 23 (4) (1994) 780–788, https://doi.org/10.1137/S0097539792224061.

[52] Carsten Thomassen, Hamiltonian-connected tournaments, J. Comb. Theory, Ser. B 28 (2) (1980) 142–163, https://doi.org/10.1016/0095-8956(80)90061-1.

[53] Dekel Tsur, Faster deterministic parameterized algorithm for k-path, Theor. Comput. Sci. 790 (2019) 96–104, https://doi.org/10.1016/j.tcs.2019.04.024.

[54] Ryan Williams, Finding paths of length k in $O^*(2^k)$ time, Inf. Process. Lett. 109 (6) (2009) 315–318, https://doi.org/10.1016/j.ipl.2008.11.004.

[55] Bang Ye Wu, Hung-Lung Wang, The next-to-shortest path problem on directed graphs with positive edge weights, Networks 65 (3) (2015) 205–211, https://doi.org/10.1002/net.21598.