

Routing with Congestion in Acyclic Digraphs

Saeed Akhoondian Amiri¹, Stephan Kreutzer^{*2}, Dániel Marx^{†3},
and Roman Rabinovich⁴

1 Technical University Berlin, Berlin, Germany
saeed.amiri@tu-berlin.de

2 Technical University Berlin, Berlin, Germany
stephan.kreutzer@tu-berlin.de

3 Institute for Computer Science and Control, Hungarian Academy of Sciences
(MTA SZTAKI), Budapest, Hungary
dmarx@cs.bme.hu

4 Technical University Berlin, Berlin, Germany
roman.rabinovich@tu-berlin.de

Abstract

We study the version of the k -disjoint paths problem where k demand pairs $(s_1, t_1), \dots, (s_k, t_k)$ are specified in the input and the paths in the solution are allowed to intersect, but such that no vertex is on more than c paths. We show that on directed acyclic graphs the problem is solvable in time $n^{O(d)}$ if we allow congestion $k - d$ for k paths. Furthermore, we show that, under a suitable complexity theoretic assumption, the problem cannot be solved in time $f(k)n^{o(d/\log d)}$ for any computable function f .

1998 ACM Subject Classification G.2.2 Graph Theory, F.2.2 Nonnumerical Algorithms and Problems

Keywords and phrases Algorithms, Disjoint Paths, Congestion, Acyclic Digraphs, XP, W[1]-hard

Digital Object Identifier 10.4230/LIPIcs.MFCS.2016.7

1 Introduction

The k -disjoint paths problem and related routing problems are among the central problems in combinatorial optimisation. In the most basic variant of the k -disjoint paths problem, a graph G is given with k pairs $(s_1, t_1), \dots, (s_k, t_k)$ of vertices and the task is to find k pairwise vertex-disjoint paths linking each s_i to its corresponding target t_i .

The problem is well known to be NP-complete [14]. On undirected graphs with a fixed number k of source/terminal pairs, Robertson and Seymour proved in their monumental graph minor series [21] that the problem is polynomial-time solvable. In fact, they showed that it is fixed-parameter tractable with parameter k : it can be solved in cubic time for every fixed value of k .

For directed graphs, the problem is computationally much harder. Fortune et al. [15] proved that it is already NP-complete for only $k = 2$ source/terminal pairs. In particular,

* The research of Saeed Amiri, Stephan Kreutzer and Roman Rabinovich has been supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (ERC consolidator grant DISTRUCT, agreement No 648527).

† The research of Dániel Marx was supported by ERC Starting Grant PARAMTIGHT (No. 280152) and OTKA grant NK105645.



© Saeed Akhoondian Amiri, Stephan Kreutzer, Dániel Marx, and Roman Rabinovich;
licensed under Creative Commons License CC-BY

41st International Symposium on Mathematical Foundations of Computer Science (MFCS 2016).

Editors: Piotr Faliszewski, Anca Muscholl, and Rolf Niedermeier; Article No. 7; pp. 7:1–7:11

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

this also implies that it is not fixed-parameter tractable on directed graphs. Following this result a lot of work has gone into establishing more efficient algorithms on restricted classes of digraphs.

Fortune et al. [15] showed that the problem can be solved in time $n^{O(k)}$ on acyclic digraphs, that is, it is polynomial-time for every fixed k . However, as proved by Slivkins [22], the problem is $W[1]$ -hard on acyclic digraphs, and therefore unlikely to be fixed-parameter tractable. On the other hand, Cygan et al. [11] proved that the problem is fixed-parameter tractable with parameter k when restricted to planar digraphs. Related to this, Amiri et al. [1] proved that the problem remains NP-complete even in upward planar graphs, but admits a single exponential fixed-parameter algorithm.

Disjoint paths problems have also been studied intensively in the area of approximation algorithms, both on directed and undirected graphs (see, e.g., [9, 18, 2, 5, 8, 4, 6, 10, 7]). The goal is, given an input graph G and demands $(s_1, t_1), \dots, (s_k, t_k)$ to route as many pairs as possible in polynomial time. There are many variations what it means for a pair to be routable. In particular, a problem studied intensively in the approximation literature is a relaxed version of disjoint paths where the paths are no longer required to be fully disjoint. Instead, they may intersect but every vertex of the graph is allowed to be contained in at most c paths, for some fixed constant c . This is called *congestion c routing*. In particular, the well-linked decomposition framework developed in [10] for undirected graphs and later generalised to digraphs in [7] has proved to be very valuable for obtaining good approximation algorithms for disjoint paths problems on planar graphs and digraphs.

In this paper, we are interested in exact solutions for high congestion routing on acyclic digraphs. More precisely, we study the following problem.

- **Definition 1. 1.** Let G be a digraph and let $I := \{(s_1, t_1), \dots, (s_k, t_k)\}$ be a set of pairs of vertices. Let $c \geq 1$. A c -routing of I is a set $\{P_1, \dots, P_k\}$ of paths such that, for all $1 \leq i \leq k$, path P_i links s_i to t_i and no vertex $v \in V(G)$ appears in more than c paths from $\{P_1, \dots, P_k\}$.
- 2. Let $k, c \geq 1$. In the (k, c) -CONGESTION ROUTING problem, a digraph G is given in the input together with a set $I := \{(s_1, t_1), \dots, (s_k, t_k)\}$ of k pairs of vertices (the demands); the task is to decide whether there is a c -routing of I in G .

We consider (k, c) -CONGESTION ROUTING on acyclic digraphs. First, it is not very difficult to show that, for every fixed $c \geq 1$, we can generalise the $n^{O(1)}$ time algorithm of Fortune et al. [15] to (k, c) -CONGESTION ROUTING. By revisiting the $W[1]$ -hardness proof of Slivkins [22] and making appropriate modifications, we can establish that the problem remains $W[1]$ -hard for every fixed congestion $c \geq 1$. Moreover, by doing the proof in a more modern way (reducing from general subgraph isomorphism instead of maximum clique and invoking a lower bound of Marx [20]), we can show that the $n^{O(k)}$ time algorithm is essentially best possible with respect to the exponent of n . This lower bound is under the Exponential-Time Hypothesis (ETH), which can be informally stated as n -variable 3SAT cannot be solved in time $2^{o(n)}$ (see [16, 19, 12] for more background).

► **Theorem 2.** For any fixed integer $c \geq 1$, (k, c) -CONGESTION ROUTING is $W[1]$ -hard parameterised by k and, assuming ETH, cannot be solved in time $f(k)n^{o(k/\log k)}$ for any computable function f .

Intuitively, one can expect the problem to become simpler if c is almost as large as k : after all, the problem is trivial if $c \geq k$. Therefore, we study the complexity of the problem in settings close to this extreme case. The main algorithmic result of this paper is to show that

for any fixed value of $d \geq 1$, the $(k, k - d)$ -CONGESTION ROUTING problem can be solved in time $n^{O(d)}$. That is, the exponent of the polynomial bounding the running time of the algorithm only depends on d but not on the number k .

► **Theorem 3.** *For every fixed $d \geq 1$ and for all $k \geq 1$ the $(k, k - d)$ -CONGESTION ROUTING problem on acyclic digraphs can be solved in time $n^{O(d)}$.*

A simple corollary of Theorem 2 shows that $(k, k - d)$ -CONGESTION ROUTING is unlikely to be fixed-parameter tractable and the running time of Theorem 3 essentially cannot be improved (assuming ETH). Observe that if we set $d := k - 1$, then $(k, k - d)$ -CONGESTION ROUTING is simply the standard k -disjoint path problem, thus any algorithmic result for $(k, k - d)$ -CONGESTION ROUTING parameterised by d would imply the essentially same algorithmic result for the fully disjoint version parameterised by k .

► **Corollary 4.** *$(k, k - d)$ -CONGESTION ROUTING is $W[1]$ -hard parameterised by d (if k is part of the input) and, assuming ETH, cannot be solved in time $f(k)n^{o(d/\log d)}$ for any computable function f .*

Organisation. The paper is organised as follows. In Section 3 we fix some notation and prove our main algorithmic result. The corresponding lower bound is then proved in Section 4.

2 Preliminaries

We review basic notation and concepts of graph theory needed in the paper. We refer to [13, 3] for background.

Let G be a digraph. We write $V(G)$ and $E(G)$ for its set of vertices and edges, respectively. We assume that there is no edge with the same head and tail, i.e. there are no loops in the digraphs we consider in this paper. If $(u, v) \in E(G)$ is an edge, then u is its *tail* and v is its *head*. G is *simple* if there are no two distinct edges which have the same tail and the same head. Otherwise we call G a *multi digraph*.

A *path* P in a digraph G is determined by a sequence (v_1, \dots, v_ℓ) of vertices such that $v_i \neq v_j$ for all $1 \leq i < j \leq \ell$ and $(v_i, v_{i+1}) \in E(G)$ for all $1 \leq i < \ell$. We write $E(P)$ for the set $\{(v_i, v_{i+1}) : 1 \leq i \leq \ell - 1\}$ of edges appearing in P and $V(P)$ for the set $\{v_1, \dots, v_\ell\}$ of vertices. We say that P *links* v_1 to v_ℓ .

Two paths P_1 and P_2 are *edge disjoint* if $E(P_1) \cap E(P_2) = \emptyset$.

3 A polynomial-time algorithm on acyclic digraphs

In this section we prove the first main result of this paper, Theorem 3, which we repeat here for convenience.

Theorem 3. *For every fixed $d \geq 1$, the $(k, k - d)$ -CONGESTION ROUTING problem on acyclic digraphs can be solved in time $n^{O(d)}$.*

We first need some additional notation and prove some auxiliary lemmas.

► **Definition 5.** Let G be a digraph and let \mathcal{L} be a set of paths in G . For every $v \in V(G)$ we define the *congestion* of v with respect to \mathcal{L} as the number of paths in \mathcal{L} containing v .

The following lemma provides a simple extension of the algorithm from [15] for disjoint paths in acyclic digraphs.

► **Lemma 6.** *On acyclic digraphs G the (k, c) -CONGESTION ROUTING problem can be solved in time $n^{O(k)}$, where $n := |G|$.*

Proof. In [15], Fortune et al. proved that the k -disjoint paths problem can be solved in time $n^{O(k)}$ on any n -vertex acyclic digraph G .

Let $G, (s_1, t_1), \dots, (s_k, t_k)$ and c be given. We construct a new digraph H with $V(H) := V(G) \times \{1, \dots, c\}$ and $E(H) := \{(u, i), (v, j) : (u, v) \in E(G), 1 \leq i, j \leq c\}$.

Then H contains k pairwise vertex disjoint paths P_1, \dots, P_k such that P_i links $(s_i, 1)$ to $(t_i, 1)$ if, and only if, there is a positive solution to the (k, c) -Congestion Routing Problem on G . By the algorithm in [15] we can decide whether the paths P_1, \dots, P_k exist in H in time $|V(H)|^{O(k)}$ and hence in time $(c \cdot n)^{O(k)} = n^{O(k)}$ as $c \leq n$. ◀

We will use this lemma in the form given in the next corollary.

► **Corollary 7.** *For $c, k \geq 0$ such that $k \in O(c)$, the (k, c) -CONGESTION ROUTING problem can be solved on any acyclic n -vertex digraph G in time $n^{O(c)}$.*

The next lemma provides the main reduction argument for proving Theorem 3.

► **Lemma 8.** *Let G be an acyclic directed graph and let $d \geq 1$ and $k > 3d$. Let $I := \{(s_1, t_1), \dots, (s_k, t_k)\} \subseteq V(G) \times V(G)$ be a set of source/terminal pairs. There exists a $(k-d)$ -routing of I if, and only if, for every pair $(s, t) \in I$ there is a path in G from s to t and there is a subset $I' \subsetneq I$ of order $|I'| = k - 1$ such that there is a $(k-d-1)$ -routing of I' .*

Proof. The if direction is easy to see. Let $\mathcal{S}' := \{P_1, \dots, P_{k-1}\}$ be a $(k-d-1)$ -routing of a set $I' \subseteq I$ of order $k-1$. Let s, t be such that $I = I' \cup \{(s, t)\}$. By assumption there is a simple path P from s to t in G . Then $\mathcal{S} := \mathcal{S}' \cup \{P\}$ is a $(k-d)$ -routing of I .

For the reverse direction let $I := \{(s_1, t_1), \dots, (s_k, t_k)\}$ and let $\hat{\mathcal{S}} := \{\hat{P}_1, \dots, \hat{P}_k\}$ be a $(k-d)$ -routing of I such that \hat{P}_i links s_i to t_i , for all $1 \leq i \leq k$. We define a multi digraph G' on the same vertex set $V(G)$ as G as follows. For every pair $u, v \in V(G')$ such that $e = (u, v) \in E(G)$ and every $1 \leq i \leq k$, if e occurs on the path $\hat{P}_i \in \hat{\mathcal{S}}$, then we add a new edge $e^i = (u, v)$ to G' . Hence, if any edge $e \in E(G)$ is used by ℓ different paths in $\hat{\mathcal{S}}$, then G' contains ℓ parallel edges between the endpoints of e . In the rest of the proof we will work on the multi digraph G' . We can now take a set $\mathcal{S} := \{P_1, \dots, P_k\}$ of pairwise edge disjoint paths, where P_i is the path from s_i to t_i induced by the edge set $\{e^i : e \in E(\hat{P}_i)\}$. That is, by using the parallel edges, we can turn the routing $\hat{\mathcal{S}}$ into a $(k-d)$ -routing \mathcal{S} of I where the paths are mutually edge disjoint.

In the remainder of the proof we will construct a subset $I' \subsetneq I$ of order $k-1$ and a $(k-d-1)$ -routing of I' in G' which is pairwise edge disjoint. This naturally induces a $(k-d-1)$ -routing of I' in G . Note that in G' , if \mathcal{L} is any set of pairwise edge disjoint paths, then the congestion of any vertex with respect to \mathcal{L} is at most the congestion of the vertex with respect to \mathcal{S} (and thus $\hat{\mathcal{S}}$) in G' (and G , respectively). Indeed, every edge in \mathcal{L} has a corresponding path in \mathcal{S} , so no vertex can be contained in more paths from \mathcal{L} than in \mathcal{S} .

Let \sqsubseteq be a topological ordering of G' and let $A := \{a_1, \dots, a_\ell\}$ be the set of vertices of congestion $k-d$ with respect to \mathcal{S} such that $a_i \sqsubseteq a_j$ whenever $i < j$. As $k > 3d$, for all $1 \leq i < \ell$ there is a path in G from a_i to a_{i+1} .

For $1 \leq i \leq k$, an *atomic subpath* of P_i (with respect to \mathcal{S}) is a subpath of P_i that starts and ends in a vertex of $A \cup \{s_i, t_i\}$ and is internally vertex disjoint from A . Hence, every path $P_i \in \mathcal{S}$ consists of the concatenation $P_i^1 \cdot \dots \cdot P_i^{\ell_i}$ of its atomic subpaths where we identify the last vertex of P_i^j with the first vertex of P_i^{j+1} for all $1 \leq j < \ell_i$. Note that any two atomic subpaths of paths P_i, P_j in \mathcal{S} are pairwise edge disjoint.

Let $I' \subset I$ be a subset of order $k-1$. A routing $\mathcal{S}' := \{P'_1, \dots, P'_{k-1}\}$ of I' is *conservative with respect to \mathcal{S}* if it consists of pairwise edge disjoint paths and every path in \mathcal{S}' consists of a concatenation of atomic subpaths of paths in \mathcal{S} . In the sequel, whenever we speak of a conservative I' -routing we implicitly mean that it is conservative with respect to \mathcal{S} .

If \mathcal{S}' is a conservative I' -routing with respect to \mathcal{S} , then it consists of pairwise edge disjoint paths and hence for every $v \in V(G)$ the congestion of v with respect to \mathcal{S}' is at most the congestion of v with respect to \mathcal{S} .

Let $1 \leq i_1 < i_2 \leq \ell$ and let $1 \leq j \leq k$. Let \mathcal{S}' be a conservative I' -routing. An (i_1, i_2) -jump of colour j is a subpath P' of P_j from a_{i_1} to a_{i_2} such that for all i with $i_1 < i < i_2$ the vertex a_i is not on P_j . Note that any jump is an atomic subpath. We call the jump P' *free with respect to \mathcal{S}'* if P' is not used by any path in \mathcal{S}' .

We are now ready to complete the proof of the lemma. Note first that, as $k > 3d$, for any three vertices $b_1, b_2, b_3 \in A$ there is a path $P \in \mathcal{S}$ that contains b_1, b_2, b_3 . Hence, we can choose an $h \in \{1, \dots, k\}$ such that $a_1, a_\ell \in V(P_h)$ and there is a vertex a_r with $1 < r < \ell$ such that $a_r \in V(P_h)$. Let $I' := I \setminus \{(s_h, t_h)\}$. If $A \subseteq V(P_h)$, then $\mathcal{S} \setminus \{P_h\}$ is a $(k-d-1)$ -routing of I' and we are done. Otherwise, for every vertex $a_r \in A$ which has congestion $k-d$ with respect to $\mathcal{S} \setminus \{P_h\}$ there are i, j with $i < r < j$ and an (i, j) -jump P of colour h . This follows as $a_1, a_\ell \in V(P_h)$. Note also that a_1 and a_ℓ have congestion $k-d-1$ in $\mathcal{S} \setminus \{P_h\}$. Note that this jump P is free with respect to $\mathcal{S} \setminus \{P_h\}$.

Thus, it is easily seen that $\mathcal{S} \setminus \{P_h\}$ satisfies the following two properties:

1. For every vertex a_r of congestion $k-d$ with respect to $\mathcal{S} \setminus \{P_h\}$ there are indices $i < r < j$ such that there is a free (i, j) -jump P with respect to $\mathcal{S} \setminus \{P_h\}$.
2. For any three vertices b_1, b_2, b_3 of congestion $k-d$ with respect to $\mathcal{S} \setminus \{P_h\}$ there is a path $Q \in \mathcal{S} \setminus \{P_h\}$ with $\{b_1, b_2, b_3\} \subseteq V(Q)$.

Now let \mathcal{S}' be a routing of I' which satisfies Condition 1 and 2 (with respect to \mathcal{S}' instead of $\mathcal{S} \setminus \{P_h\}$) and, subject to this, the number of vertices of congestion $k-d$ with respect to \mathcal{S}' is minimal.

We claim that \mathcal{S}' is a $(k-d-1)$ -routing of I' . Let $\mathcal{S}' := \{Q_1, \dots, Q_{k-1}\}$. Towards a contradiction, suppose there is a vertex a_r of congestion $k-d$ with respect to \mathcal{S}' . As \mathcal{S}' is conservative, we have $a_r \in A$. Hence, by assumption, there are $i < r < j$ and a free (i, j) -jump P with respect to \mathcal{S}' .

Let Q_h be a path in \mathcal{S}' that contains a_i, a_r and a_j , which exists by Condition 2. Let $Q_h := Q_h^1 \cup Q_h^2 \cup Q_h^3$ where

- Q_h^1 is the initial subpath of Q_h from its first vertex to a_i ,
- Q_h^2 is the subpath starting at a_i and ending in a_j and
- Q_h^3 is the subpath starting in a_j and ending at the end of Q_h .

We define $Q'_h := Q_h^1 \cup P \cup Q_h^3$, i.e. Q'_h is the path obtained from Q_h by replacing the inner subpath Q_h^2 by the (i, j) -jump P . Let $\mathcal{L} := (\mathcal{S}' \setminus \{Q_h\}) \cup \{Q'_h\}$. Then \mathcal{L} is a routing of I' . It is also conservative as we have only rerouted a single path along a free jump.

We need to show that for all b_1, b_2, b_3 of congestion $k-d$ with respect to \mathcal{L} there is a path $Q \in \mathcal{L}$ containing b_1, b_2, b_3 . By assumption, such a path Q' exists in \mathcal{S}' . If $Q' \neq Q_h$, then we are done. So suppose $Q_h = Q'$. But then this implies that $b_s \notin \{a_{i+1}, \dots, a_{j-1}\}$ for all $1 \leq s \leq 3$ as otherwise the congestion of b_s would have dropped to $k-d-1$ in \mathcal{L} . But then $b_1, b_2, b_3 \in V(Q'_h)$.

It remains to show that for every vertex a_s of congestion $k-d$ with respect to \mathcal{L} there is a free (i, j) -jump for some $i < s < j$. As before, by assumption, there are $s_1 < s < s_2$ and a free (s_1, s_2) -jump with respect to \mathcal{S}' . If this jump is not P , then it still exists with respect to

\mathcal{L} and we are done. So suppose this jump is P , which implies that $i < s < j$. Furthermore, $a_s \notin Q_h$ as otherwise the congestion of a_s in \mathcal{L} would be $k - d - 1$. But then, there must be indices i_1, i_2 with $i \leq i_1 < s < i_2 \leq j$ such that $a_{i_1}, a_{i_2} \in V(Q_h)$ and $a_{s'} \notin V(Q_h)$ for all $i_1 < s' < i_2$. Hence, the atomic subpath Q'' of Q_h from a_{i_1} to a_{i_2} is an (i_1, i_2) -jump as required. As $Q'' \subseteq Q_h^2$, this jump is now free.

Finally, the vertex a_r now has congestion $k - d - 1$ with respect to \mathcal{L} as a_r is not contained in Q'_h . Hence, \mathcal{L} has fewer vertices of congestion $k - d$ than \mathcal{S}' , contradicting the choice of \mathcal{S}' . Thus, \mathcal{S}' must have been a $(k - d - 1)$ -routing of I' as required. This completes the proof of the lemma. \blacktriangleleft

By repeatedly applying Lemma 8 we obtain the following corollary, which essentially implies Theorem 3.

► Corollary 9. *Let G be an acyclic digraph, $d \geq 0$, $k \geq 3d$ and let $I := \{(s_1, t_1), \dots, (s_k, t_k)\}$ be a set of pairs of vertices such that for all $1 \leq i \leq k$ there is a path in G linking s_i to t_i . Then G contains a $(k - d)$ -routing of I if, and only if, there is a subset $I' \subseteq I$ with $|I'| = 3d$ such that G contains a $2d$ -routing of I' .*

We are now ready to prove Theorem 3.

Proof of Theorem 3. Let G, k, d and $I := \{(s_1, t_1), \dots, (s_k, t_k)\}$ be given. Let $n := |G|$. If for some $1 \leq i \leq k$ there is no path in G from s_i to t_i , then the answer is no and we are done. If $k \leq 3d$, then we can apply Corollary 7 to compute the answer in time $n^{O(d)}$ as required.

Otherwise, by Corollary 9, there is a $(k - d)$ -routing for I in G if, and only if, there is a subset $I' \subsetneq I$ of order $3d$ such that I' has a $2d$ -routing. There are $\binom{k}{3d} \leq k^{3d} \leq n^{3d}$ subsets I' of order $3d$. By Corollary 7, we can decide for any such I' of order $3d$ in time $n^{O(d)}$ whether a $2d$ -routing of I' exists. Hence, by iterating through all possible subsets I' , we can decide in time $n^{O(d)}$ whether there is a $(k - d)$ -routing of I in G . \blacktriangleleft

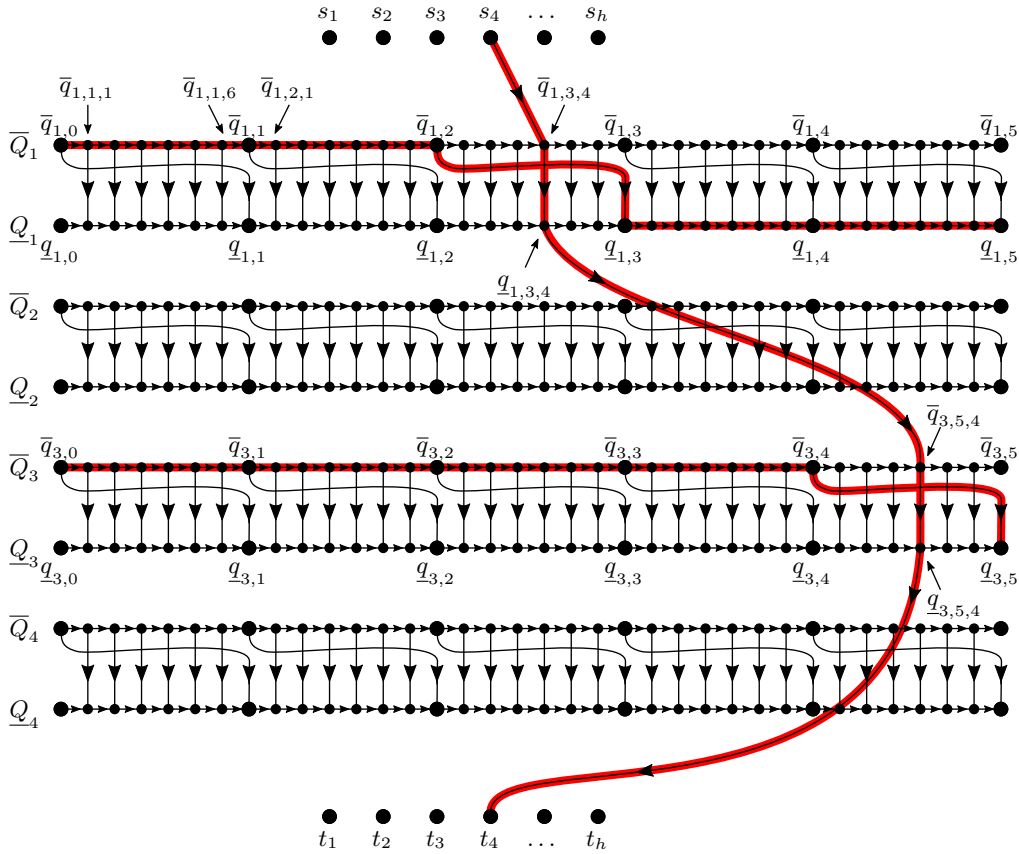
4 Lower Bounds

In this section, we prove Theorem 2 by a reduction from PARTITIONED SUBGRAPH ISOMORPHISM. The input of the PARTITIONED SUBGRAPH ISOMORPHISM problem consists of a graph H with vertex set $\{u_1, \dots, u_k\}$ and a graph G whose vertex set is partitioned into k classes V_1, \dots, V_k . The task is to find a mapping $\mu : V(H) \rightarrow V(G)$ such that $\mu(u_i) \in V_i$ for every $1 \leq i \leq k$ and μ is a subgraph embedding, that is, if u_i and u_j are adjacent in H , then $\mu(u_i)$ and $\mu(u_j)$ are adjacent in G .

► Theorem 10 ([20]). *Assuming ETH, PARTITIONED SUBGRAPH ISOMORPHISM cannot be solved in time $f(k)n^{o(k/\log k)}$ (where $k = |V(H)|$) for any computable function f , even when H is assumed to be 3-regular and bipartite.*

To prove Theorem 2, we need a reduction from PARTITIONED SUBGRAPH ISOMORPHISM (for 3-regular bipartite graphs) to (k, c) -CONGESTION ROUTING, where the number k of demands is linear in the number of vertices of H .

Proof (of Theorem 2). We prove the theorem by a reduction from PARTITIONED SUBGRAPH ISOMORPHISM. Let H and G be two graphs, let $V(H) = \{u_1, \dots, u_k\}$, and let (V_1, \dots, V_k) be a partition of $V(G)$. By copying vertices if necessary, we may assume that every V_i has the same size n ; let us denote by $\{v_{i,1}, \dots, v_{i,n}\}$ the vertices in V_i . By Theorem 10, we may assume that H is 3-regular and bipartite. This means that H has exactly $h = 3k/2$ edges



■ **Figure 1** Part of the directed graph D constructed in the proof of Theorem 2 with $k = 4$, $h = 6$, and $n = 5$. For clarity, we consider only one edge e_4 of H , which connects u_1 and u_3 , and assume that the only edge between V_1 and V_3 is between $v_{1,3}$ and $v_{3,5}$. The highlighted red paths show the paths P_1^v , P_3^v , and P_4^e of the solution.

and both partite classes contain $k/2$ vertices. Without loss of generality, we can assume that $U_1 = \{u_1, \dots, u_{k/2}\}$ and $U_2 = \{u_{k/2+1}, \dots, u_k\}$ are the two partite classes. Let us fix an arbitrary ordering e_1, \dots, e_h of the edges of H .

Construction. We construct an instance of (k, c) -Congestion Routing in the following way. We construct a directed graph D that contains, for every $1 \leq i \leq k$, two directed paths \bar{Q}_i and Q_i (see Figure 1). Path \bar{Q}_i has $n(h + 1) + 1$ vertices: it contains the vertices $\bar{q}_{i,0}, \dots, \bar{q}_{i,n}$ in this order and additionally, for every $1 \leq j \leq n$, the vertices $\bar{q}_{i,j,1}, \dots, \bar{q}_{i,j,h}$ are inserted between $\bar{q}_{i,j-1}$ and $\bar{q}_{i,j}$. The path Q_i is defined the same way, with vertices q instead of \bar{q} . For every $1 \leq \ell \leq h$, we introduce two vertices s_ℓ and t_ℓ . Then we complete the construction of the graph D by introducing further edges as follows.

- For every $1 \leq i \leq k$ and $1 \leq j \leq n$, we introduce the edge $(\bar{q}_{i,j-1}, \underline{q}_{i,j})$ (the curved bypass edges in Figure 1).
- For every $1 \leq i \leq k$, $1 \leq j \leq n$, and $1 \leq s \leq h$, we introduce the edge $(\bar{q}_{i,j,s}, \underline{q}_{i,j,s})$ (the vertical edges in Figure 1).
- For every $1 \leq \ell \leq h$, we do the following. Suppose that edge e_ℓ of H connects u_{i_a} and u_{i_b} for some $1 \leq i_a \leq k/2$ and $k/2 + 1 \leq i_b \leq k$. Then for every pair of vertices $v_{i_a,j_a} \in V_{i_a}$ and $v_{i_b,j_b} \in V_{i_b}$ that are adjacent in G , we introduce the following three edges into D :

$(s_\ell, \bar{q}_{i_a, j_a, \ell})$, $(\underline{q}_{i_a, j_a, \ell}, \bar{q}_{i_b, j_b, \ell})$, and $(\underline{q}_{i_b, j_b, \ell}, t_\ell)$.

To complete the construction of the (k, c) -Congestion Routing instance, we define the following set of $k + 2k(c - 1) + h$ demands:

- For every $1 \leq i \leq k$, we introduce the demand $(\bar{q}_{i,0}, \underline{q}_{i,n})$ (vertex demands).
- For every $1 \leq i \leq k$, we introduce $c - 1$ copies of the demand $(\bar{q}_{i,0}, \bar{q}_{i,n})$ (blocking demands).
- For every $1 \leq i \leq k$, we introduce $c - 1$ copies of the demand $(\underline{q}_{i,0}, \underline{q}_{i,n})$ (blocking demands).
- For every $1 \leq \ell \leq h$, we introduce the demand (s_ℓ, t_ℓ) (edge demands).

Note that, for every fixed $c \geq 1$, the number of demands is $O(k)$. In the rest of the proof, we show that a routing with congestion c exists if and only if the PARTITIONED SUBGRAPH ISOMORPHISM instance has a solution. Then the W[1]-hardness and lower bound stated in Theorem 10 implies the same hardness results for the routing problem.

Subgraph embedding \Rightarrow routing. Suppose first that vertices $v_{1,z_1} \in V_1, \dots, v_{k,z_k} \in V_k$ form a solution to the PARTITIONED SUBGRAPH ISOMORPHISM instance. We construct a routing that contains the following paths, satisfying the demands defined above:

- For every $1 \leq i \leq k$, the vertex demand $(\bar{q}_{i,0}, \underline{q}_{i,n})$ is satisfied by a path P_i^v that goes from $\bar{q}_{i,0}$ to \bar{q}_{i,z_i-1} on \bar{Q}_i , uses the edge $(\bar{q}_{i,z_i-1}, \underline{q}_{i,z_i})$, and then goes from \underline{q}_{i,z_i} to $\underline{q}_{i,n}$ on \underline{Q}_i .
- For every $1 \leq i \leq k$, each of the $c - 1$ copies of the blocking demand $(\bar{q}_{i,0}, \bar{q}_{i,n})$ is satisfied by a path going on \bar{Q}_i .
- For every $1 \leq i \leq k$, each of the $c - 1$ copies of the blocking demand $(\underline{q}_{i,0}, \underline{q}_{i,n})$ is satisfied by a path going on \underline{Q}_i .
- For every $1 \leq \ell \leq h$, the edge demand (s_ℓ, t_ℓ) is satisfied by a 5-edge path $P_\ell^e = (s_\ell, \bar{q}_{i_a, z_{i_a}, \ell}, \underline{q}_{i_a, z_{i_b}, \ell}, \bar{q}_{i_b, z_{i_b}, \ell}, \underline{q}_{i_b, z_{i_b}, \ell}, t_\ell)$.

It is easy to verify that these are indeed paths: all the required edges exist. We claim that each vertex of D is used by at most c of these paths. It is easy to see that two paths $P_{i'}^v$ and $P_{i''}^v$ with $i' \neq i''$ satisfying vertex demands do not intersect, and this is also true for any two paths $P_{\ell'}^e$ and $P_{\ell''}^e$ with $\ell' \neq \ell''$ satisfying edge demands (note that each vertex of the path P_ℓ^e has ℓ in its index). The crucial observation is that the path P_i^v does not intersect the path P_ℓ^e for any ℓ . The only way this could possibly happen is if edge e_ℓ of H connects u_{i_a} with u_{i_b} , and i is equal to i_a or i_b . But the path P_ℓ^e uses only vertex $\bar{q}_{i_a, z_{i_a}, \ell}$ from \bar{Q}_{i_a} and vertex $\underline{q}_{i_a, z_{i_b}, \ell}$ from \underline{Q}_{i_b} , while the path P_i^v does not use these vertices, as it jumps from \bar{q}_{i, z_i-1} to \underline{q}_{i, z_i} . Thus each vertex is used by at most $c - 1$ paths satisfying a blocking demand and at most one additional path satisfying a vertex or edge demand. We can conclude that each vertex is used by at most c of the paths, what we had to show.

Routing \Rightarrow subgraph embedding. Next we show that given a routing with congestion c , it is possible to construct the required subgraph embedding from H to G . It is clear that the path satisfying the blocking demand $(\bar{q}_{i,0}, \bar{q}_{i,n})$ is exactly \bar{Q}_i : after leaving \bar{Q}_i , there is no way to return back to it. Similarly, the solution must use path \underline{Q}_i to satisfying the blocking demand $(\underline{q}_{i,0}, \underline{q}_{i,n})$. It is also clear that the path P_i^v satisfying the vertex demand $(\bar{q}_{i,0}, \underline{q}_{i,n})$ has to be contained in the union of \bar{Q}_i and \underline{Q}_i . Let $1 \leq z_i \leq n$ be the smallest value such that \underline{q}_{i, z_i} is on path P_i^v (note that this value is positive, as vertex $\underline{q}_{i,0}$ cannot be reached from $\bar{q}_{i,0}$). Observe that path P_i^v uses every vertex of \underline{Q}_i from \underline{q}_{i, z_i} to $\underline{q}_{i,n}$ (as it cannot leave from $\bar{q}_{i,0}$). Moreover, since P_i^v does not use the part of \underline{Q}_i from $\underline{q}_{i,0}$ to \underline{q}_{i, z_i-1} by definition, it has to use the part of \bar{Q}_i from $\bar{q}_{i,0}$ to \bar{q}_{i, z_i-1} .

We claim that mapping vertex u_i of H to vertex v_{i,z_i} of G is a correct subgraph embedding of H into G . To show this, suppose that edge e_i of H connects u_{i_a} and u_{i_b} with $1 \leq i_a \leq k/2$ and $k/2 + 1 \leq i_b \leq k$; we need to show that $v_{i_a,z_{i_a}} \in V_{i_a}$ and $v_{i_b,z_{i_b}} \in V_{i_b}$ are adjacent. Consider the path P_ℓ^e satisfying edge demand (s_ℓ, t_ℓ) . By construction, the vertex of P_ℓ^e after s_ℓ has to be on the path \bar{Q}_{i_a} and the vertex of P_ℓ^e before t_ℓ has to be on Q_{i_b} . The only way to go from \bar{Q}_{i_a} to Q_{i_b} is to use an edge of the form $(q_{i_a,j_a,\ell}, \bar{q}_{i_b,j_b,\ell})$: the only way we can leave the union of \bar{Q}_{i_a} and Q_{i_a} is to enter some \bar{Q}_i with $k/2 + 1 \leq i \leq k$, and there is no edge connecting \bar{Q}_{i_b} or Q_{i_b} with any \bar{Q}_i with $k/2 + 1 \leq i \leq k$ and $i \neq i_b$ (this is the part of the proof where we use that H is bipartite). We claim that $j_a = z_{i_a}$. If $j > z_{i_a}$, then $q_{i_a,j_a,\ell}$ is also used by the $c - 1$ paths satisfying the blocking demand $(q_{i_a,0}, q_{i_a,n})$ and (as we have seen) the path $P_{i_a}^v$, contradicting the assumption that the routing has congestion c . If $j < z_{i_a}$, then there is no way for the path P_ℓ^e to reach $q_{i_a,j_a,\ell}$ from s_ℓ : each vertex of the path \bar{Q}_{i_a} from $\bar{q}_{i_a,0}$ to \bar{q}_{i_a,j_a} is used by $c - 1$ paths satisfying the blocking demand $(q_{i_a,0}, q_{i_a,n})$ and (as shown above) by the path $P_{i_a}^v$. This shows $j_z = z_{i_a}$ and a similar argument shows $j_b = z_{i_b}$. Now the existence of the edge $(q_{i_a,z_a,\ell}, \bar{q}_{i_b,z_b,\ell})$ means, by construction, that G contains an edge between $v_{i_a,z_a} \in V_{i_a}$ and $v_{i_b,z_b} \in V_{i_b}$, what we had to show. ◀

5 Conclusion

In this paper we have studied the (k, c) -CONGESTION ROUTING problem on acyclic digraphs. It is easy to see that the $n^{O(k)}$ algorithm in [15] for solving the disjoint paths problem on acyclic digraphs can be extended to an $n^{O(k)}$ algorithm for (k, c) -CONGESTION ROUTING. As we proved in Theorem 2, the $n^{O(k)}$ time algorithm is essentially best possible with respect to the exponent of n , under the Exponential-Time Hypothesis (ETH). We therefore studied the extreme cases of relatively high congestion $k - d$ for some fixed value of d . In Theorem 3 we showed that in this case we can obtain an $n^{O(d)}$ algorithm on acyclic digraphs, i.e. the algorithm only depends on the offset d in $(k, k - d)$ -CONGESTION ROUTING but not on the number k of demand pairs. The proof relied on a reduction argument that shows that as long as k is big enough compared to d , then a demand pair can be eliminated without changing the answer.

It will be interesting to see whether our result can be extended to larger classes of digraphs. In particular classes of digraphs of bounded directed tree width would be a natural target. On such classes, the k -disjoint paths problem can be solved in time $n^{O(k+w)}$, where w is the directed tree width of the input digraph (see [17]). It is conceivable that our results extend to bounded directed tree width classes and we leave this for future research.

References

- 1 Saeed Akhoondian Amiri, Ali Golshani, Stephan Kreutzer, and Sebastian Siebertz. Vertex disjoint paths in upward planar graphs. In Edward A. Hirsch, Sergei O. Kuznetsov, Jean-Éric Pin, and Nikolay K. Vereshchagin, editors, *Computer Science - Theory and Applications: 9th International Computer Science Symposium in Russia, CSR 2014, Moscow, Russia, June 7-11, 2014. Proceedings*, pages 52–64. Springer International Publishing, Cham, 2014. doi:10.1007/978-3-319-06686-8_5.
- 2 Matthew Andrews, Julia Chuzhoy, Venkatesan Guruswami, Sanjeev Khanna, Kunal Talwar, and Lisa Zhang. Inapproximability of edge-disjoint paths and low congestion routing on undirected graphs. *Combinatorica*, 30(5):485–520, 2011. doi:10.1007/s00493-010-2455-9.

- 3 Jorgen Bang-Jensen and Gregory Z. Gutin. *Digraphs - Theory, Algorithms and Applications*. Springer, 2nd edition, 2010.
- 4 Parinya Chalermsook, Julia Chuzhoy, Alina Ene, and Shi Li. Approximation algorithms and hardness of integral concurrent flow. In *Proceedings of the Forty-fourth Annual ACM Symposium on Theory of Computing*, STOC '12, pages 689–708, New York, NY, USA, 2012. ACM. doi:10.1145/2213977.2214040.
- 5 C. Chekuri, S. Khanna, and F. B. Shepherd. Edge-disjoint paths in planar graphs. In *Foundations of Computer Science, 2004. Proceedings. 45th Annual IEEE Symposium on*, pages 71–80, Oct 2004. doi:10.1109/FOCS.2004.27.
- 6 Chandra Chekuri and Alina Ene. Poly-logarithmic approximation for maximum node disjoint paths with constant congestion. In Sanjeev Khanna, editor, *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2013, New Orleans, Louisiana, USA, January 6-8, 2013*, pages 326–341. SIAM, 2013. doi:10.1137/1.9781611973105.24.
- 7 Chandra Chekuri and Alina Ene. The all-or-nothing flow problem in directed graphs with symmetric demand pairs. In Jon Lee and Jens Vygen, editors, *Integer Programming and Combinatorial Optimization - 17th International Conference, IPCO 2014, Bonn, Germany, June 23-25, 2014. Proceedings*, volume 8494 of *Lecture Notes in Computer Science*, pages 222–233. Springer, 2014. doi:10.1007/978-3-319-07557-0_19.
- 8 Chandra Chekuri, Sanjeev Khanna, and F. Bruce Shepherd. Multicommodity flow, well-linked terminals, and routing problems. In *Proceedings of the Thirty-seventh Annual ACM Symposium on Theory of Computing*, STOC '05, pages 183–192, New York, NY, USA, 2005. ACM. doi:10.1145/1060590.1060618.
- 9 Chandra Chekuri, Sanjeev Khanna, and F. Bruce Shepherd. An $o(\sqrt{n})$ approximation and integrality gap for disjoint paths and unsplittable flow. *Theory of Computing*, 2(7):137–146, 2006. doi:10.4086/toc.2006.v002a007.
- 10 J. Chuzhoy and S. Li. A polylogarithmic approximation algorithm for edge-disjoint paths with congestion 2. In *Foundations of Computer Science (FOCS), 2012 IEEE 53rd Annual Symposium on*, pages 233–242, Oct 2012. doi:10.1109/FOCS.2012.54.
- 11 M. Cygan, D. Marx, M. Pilipczuk, and M. Pilipczuk. The planar directed k-vertex-disjoint paths problem is fixed-parameter tractable. In *Foundations of Computer Science (FOCS), 2013 IEEE 54th Annual Symposium on*, pages 197–206, Oct 2013. doi:10.1109/FOCS.2013.29.
- 12 Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015. doi:10.1007/978-3-319-21275-3.
- 13 Reinhard Diestel. *Graph Theory*. Springer-Verlag, 4th edition, 2010.
- 14 S. Even, A. Itai, and A. Shamir. On the complexity of timetable and multicommodity flow problems. *SIAM J. Comput.*, 5(4):691–703, 1976.
- 15 Steven Fortune, John Hopcroft, and James Wyllie. The directed subgraph homeomorphism problem. *Theoretical Computer Science*, 10(2):111 – 121, 1980. URL: <http://www.sciencedirect.com/science/article/pii/0304397580900092>, doi:http://dx.doi.org/10.1016/0304-3975(80)90009-2.
- 16 Russell Impagliazzo, Ramamohan Paturi, and Francis Zane. Which problems have strongly exponential complexity? *J. Comput. System Sci.*, 63(4):512–530, 2001.
- 17 Thor Johnson, Neil Robertson, Paul D. Seymour, and Robin Thomas. Directed tree-width. *J. Comb. Theory, Ser. B*, 82(1):138–154, 2001.
- 18 G. Stavros Kolliopoulos and Clifford Stein. Approximating disjoint-path problems using packing integer programs. *Mathematical Programming*, 99(1):63–87, 2003. doi:10.1007/s10107-002-0370-6.

- 19 Daniel Lokshtanov, Dániel Marx, and Saket Saurabh. Lower bounds based on the exponential time hypothesis. *Bulletin of the EATCS*, 105:41–72, 2011. URL: <http://albcom.lsi.upc.edu/ojs/index.php/beatcs/article/view/96>.
- 20 Dániel Marx. Can you beat treewidth? *Theory of Computing*, 6(1):85–112, 2010. arXiv: [toc:v006/a005](https://arxiv.org/abs/1006.0005), doi:10.4086/toc.2010.v006a005.
- 21 N. Robertson and P. D. Seymour. Graph minors I – XXIII, 1982 – 2010. *Appearing in Journal of Combinatorial Theory, Series B.*, from 1982-2010.
- 22 Aleksandrs Slivkins. Parameterized tractability of edge-disjoint paths on directed acyclic graphs. *SIAM J. Discrete Math.*, 24(1):146–157, 2010. doi:10.1137/070697781.