k-Distinct Branchings Admits a Polynomial Kernel

Jørgen Bang-Jensen ✉
University of Southern Denmark, Odense, Denmark

Kristine Vitting Klinkby ✉
University of Bergen, Bergen, Norway
University of Southern Denmark, Odense, Denmark

Saket Saurabh ✉
University of Bergen, Bergen, Norway
The Institute of Mathematical Sciences, HBNI, Chennai, India

Abstract

Unlike the problem of deciding whether a digraph $D = (V, A)$ has ℓ in-branchings (or ℓ out-branchings) is polynomial time solvable, the problem of deciding whether a digraph $D = (V, A)$ has an in-branching B^- and an out-branching B^+ which are arc-disjoint is NP-complete. Motivated by this, a natural optimization question that has been studied in the realm of Parameterized Complexity is called Rooted k-Distinct Branchings. In this problem, a digraph $D = (V, A)$ with two prescribed vertices s, t are given as input and the question is whether D has an in-branching rooted at t and an out-branching rooted at s such that they differ on at least k arcs. Bang-Jensen et al. [*Algorithmica, 2016*] showed that the problem is fixed parameter tractable (FPT) on strongly connected digraphs. Gutin et al. [*ICALP, 2017; JCSS, 2018*] completely resolved this problem by designing an algorithm with running time $2^{\mathcal{O}(k^2 \log^2 k)} n^{\mathcal{O}(1)}$. Here, n denotes the number of vertices of the input digraph. In this paper, answering an open question of Gutin et al., we design a polynomial kernel for Rooted k-Distinct Branchings. In particular, we obtain the following: Given an instance (D, k, s, t) of Rooted k-Distinct Branchings, in polynomial time we obtain an equivalent instance (D', k', s, t) of Rooted k-Distinct Branchings such that $|V(D')| \leq \mathcal{O}(k^2)$ and the treewidth of the underlying undirected graph is at most $\mathcal{O}(k)$. This result immediately yields an FPT algorithm with running time $2^{\mathcal{O}(k \log k)} + n^{\mathcal{O}(1)}$; improving upon the previous running time of Gutin et al. For our algorithms, we prove a structural result about paths avoiding many arcs in a given in-branching or out-branching. This result might turn out to be useful for getting other results for problems concerning in-and out-branchings.

2012 ACM Subject Classification Theory of computation → Fixed parameter tractability

Keywords and phrases Digraphs, Polynomial Kernel, In-branching, Out-Branching

Digital Object Identifier 10.4230/LIPIcs.ESA.2021.11

1 Introduction

Let $D = (V, A)$ be a digraph and r be a vertex of D. An *out-branching* (respectively, *in-branching)* in D is a connected spanning subdigraph B_r^+ (respectively, B_r^-) of D in which each vertex $v \neq r$ has precisely one entering (respectively, leaving) arc and r has no entering (respectively, leaving) arc. The vertex r is called the *root* of B_r^+ (respectively, B_r^-). The study of finding a spanning tree in an undirected graph or an out-branching in a digraph satisfying specific properties, such as having at least k leaves, or having at least k internal vertices [1, 4, 7, 10, 11, 13, 14, 18, 19, 22] has been at the forefront of research in parameterized algorithms. This paper aims to study a problem of finding an in-branching and an out-branching, in the given digraph, whose arc sets is disjoint on at least k arcs, in the realm of Kernelization Complexity [21] and Parameterized Complexity [12, 15, 17, 24].

A parameterized problem Π is said to admit a *kernel* if there is a polynomial-time algorithm, called a *kernelization algorithm*, that reduces the input instance of Π down to an equivalent instance of Π whose size is bounded by a function $f(k)$ of k. (Here, two instances

29th Annual European Symposium on Algorithms (ESA 2021).
Editors: Petra Mutzel, Rasmus Pagh, and Grzegorz Herman; Article No. 11; pp. 11:1–11:15
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

are equivalent if both of them are either yes-instances or no-instances.) Such an algorithm is called an $f(k)$-*kernel* for Π. If $f(k)$ is a polynomial function of k, we say that the kernel is a *polynomial kernel*. See [12, 15, 17, 21, 24] for more details.

In studying problems around finding edge-disjoint spanning trees, arc-disjoint in-branchings, or arc-disjoint in-branchings, one of the most important results is due to Edmonds. This classical result states that given a digraph D and a positive integer ℓ, we can test whether D contains ℓ arc-disjoint out-branchings in polynomial time [16]. In contrast to this, Thomassen proved that the problem of deciding whether a digraph contains an out-branching B_s^+ and an in-branching B_t^- which are arc-disjoint is NP-complete, even if $s = t$ (the proof is published in [2]). The problem remains NP-complete even for 2-regular digraphs [5] but it is polynomial time solvable on tournaments [2], locally semicomplete digraphs [3] and acyclic digraphs [6, 9]. In fact, even deciding whether a digraph D contains an out-branching which is arc-disjoint from some spanning tree in the underlying undirected graph remains NP-complete [8].

In this paper, we consider the following parameterized version of the arc-disjoint in- and out-branching problem by using a parameter k to measure how distinct a given pair B_s^+, B_t^- are, where the measure is in terms of the number of arcs that belongs to B_s^+ but not to B_t^-. In particular, we study the following problem.

ROOTED k-DISTINCT BRANCHINGS (R-k-DB) **Parameter:** k
Input: A digraph $D = (V, A)$, two fixed vertices $s, t \in V$ and an integer k.
Question: Does there exist an out-branching B_s^+ and an in-branching B_t^- such that $|A(B_s^+) \setminus A(B_t^-)| \geq k$?

Observe that the problem is NP-complete since it contains the arc-disjoint in- and out-branching question as to the particular case when k is the number of vertices minus one.

Context of our Study. The problem R-k-DB has a rich history in the realm of Parameterized Complexity. Bang-Jensen and Yeo [7] asked whether the problem would be FPT when $s = t$. Answering this question in the affirmative, Bang-Jensen et al. [4] showed that the problem is FPT when the input is a strongly connected digraph and asked *whether the problem is* FPT *on general digraphs*. This was confirmed in affirmative by Gutin et al. [22], who showed that the problem is solvable in time $2^{\mathcal{O}(k^2 \log^2 k)} n^{\mathcal{O}(1)}$ time. A natural follow up question that they ask is: *Does R-k-DB admits a polynomial sized kernel?* This open question is the starting point of our work.

Our Results and Methods. We design a polynomial kernel for R-k-DB.

▶ **Theorem 1.1.** *R-k-DB admits a polynomial kernel with $\mathcal{O}(k^2)$ vertices.*

A key ingredient in the work of Gutin et al. [22] is out-branchings with many leaves. A vertex v is a leaf in the out-branching B_s^+ if no arc is leaving v in B_s^+. If the input (D, s, t, k) to the R-k-DB problem is such that D has an out-branching B_s^+ with at least $k + 1$ leaves, then (D, s, t, k) is a "yes"-instance since every in-branching B_t^- will have the property that no arc of B_t^- which leaves a vertex in L will be contained in B_s^+, where L is the set of leaves of B_s^+. It was shown in [13, 23] that the problem (parameterized by p) of deciding the existence of an out-branching with at least p leaves is FPT. Furthermore, if the input is strongly connected and has no out-branching B_s^+ with at least $k + 1$ leaves, then it has pathwidth

$\mathcal{O}(k \log k)$ and now a result from [4] implies that the ROOTED k-DISTINCT BRANCHINGS problem is FPT for strongly connected digraphs. When the input is not strongly connected, the case is considerably more complicated to handle when following the approach used in [22]. Indeed, the result of Gutin et al. [22] could be viewed as an algorithm that obtains an equivalent instance with treewidth of the underlying graph bounded by $k^{\mathcal{O}(1)}$. Our approach is very different, as we bound the size rather than a structural property.

We use a fundamentally different approach, based on a structural analysis of paths avoiding many arcs of a fixed out-branching, to prove an $\mathcal{O}(k^2)$-vertex kernel for the R-k-DB problem. In the whole analysis, we work with a fixed out-branching and see how an in-branching can be built that avoids as many arcs of the given out-branching. In a step-by-step procedure, we either obtain reduction rules to reduce our input or at the end, we do get an in-branching and an out-branching that avoids k-arcs of each other. Finally, we argue that the instance on which none of the reduction rules can be applied has at most $\mathcal{O}(k^2)$ vertices.

We further look into our kernel and try to bound the treewidth of the underlying undirected graph. In particular, given an instance (D, k, s, t) of R-k-DB, in polynomial time we obtain an equivalent instance (D', k', s, t) of R-k-DB such that $|V(D')| \leq \mathcal{O}(k^2)$ and the treewidth of the underlying undirected graph is at most $\mathcal{O}(k)$. This result yields the FPT algorithm by a standard dynamic programming approach over graphs of bounded treewidth.

▶ **Theorem 1.2** ($\star$). [1] *R-k-DB admits an algorithm with running time $2^{\mathcal{O}(k \log k)} + n^{\mathcal{O}(1)}$.*

The running time obtained in Theorem 1.2 improves upon the running time of Gutin et al. [22]. We conclude by saying that the structural result on so-called *substitute paths of an out-branching* might be of independent interest and could find further applications.

2 Notation and Preliminaries

Given a digraph $D = (V, A)$, we also use $V(D)$ and $A(D)$ to denote the vertex set and the arc set of D, respectively. If it is clear from the context we simply use V and A, respectively. Given a digraph $D = (V, A)$ and an arc $a = (u, v) \in A$ we call u the *tail* of a and v the *head* of a. The vertices u and v are the *endpoints* of a. For a set of arcs $A' \subseteq A$ the *set of heads* denoted $\mathcal{H}(A')$ is the set of heads of the arcs in A'. That is, $\mathcal{H}(A') = \{h | (t, h) \in A' \text{ for some } h, t \in V\}$. Similarly, we have the *set of tails* denoted $\mathcal{T}(A')$, which is defined as follows: $\mathcal{T}(A') = \{t | (t, h) \in A' \text{ for some } h, t \in V\}$. In a digraph D we will denote a path from u to v as $P_{u,v}$. By $P_{u,v}[x, y]$ we will denote the subpath of $P_{u,v}$ which goes from x to y. We will denote the path $P_{u,v}[x, y] - \{y\}$ as $P_{u,v}[x, y[$ and the path $P_{u,v}[x, y] - \{x\}$ as $P_{u,v}]x, y]$. Given a path $P_{u,v}$ and $P_{v,z}$ we will describe the concatenation of the paths $P_{u,v}$ and $P_{v,z}$ as $P_{u,z} = P_{u,v}P_{v,z}$. For a vertex $u \in V(D)$ and an arc $(u, v) \in A(D)$ we will often use the notation $u \in D$ and $(u, v) \in D$ respectively. For a vertex $v \in V(D)$ we denote the out-neighbors of v as $N^+(v)$ and the in-neighbors as $N^-(v)$. An out-branching with a root $r \in V$ is denoted B_r^+. For a vertex $u \in V(B_r^+)$ we say that v is *parent* of u if $(v, u) \in A(B_r^+)$. To indicate that v is parent of u we use the notation $v = \mathcal{P}_{B_r^+}(u)$. We denote the directed path from u to v in B_r^+ as $B_r^+[u, v]$ and we say that u is *ancestor* to v in B_r^+ if the path $B_r^+[u, v]$ exists. An in-branching with a root $r \in V$ is denoted B_r^-. For a vertex $u \in V(B_r^-)$ we say that v is *parent* of u if $(u, v) \in A(B_r^-)$. To indicate this we use the notation $v = \mathcal{P}_{B_r^-}(u)$. We denote the directed path from u to v in B_r^- as $B_r^-[u, v]$ and we say that u is *ancestor* to v in B_r^- if the path $B_r^-[u, v]$ exists.

[1] Proofs labeled with $\star$ is not included in this version.

3 Backward Arcs, Crossing Arcs and Substitute-paths

In this section we initiate our structural analysis of substitute paths. Given an out-branching B_r^+ in a digraph $D = (V, A)$, we will work with the arcs in $A(D) \setminus A(B_r^+)$. We therefore divide these arcs into backward and crossing arcs.

▶ **Definition 1.** *Given an out-branching B_r^+ in a digraph $D = (V, A)$ an arc $(u, v) \in A(D) \setminus A(B_r^+)$ is **backward**, if the path $B_r^+[v, u]$ exists, and **crossing**, if it is not backward.*

In an out-branching B_r^+ the arc $(u, v) \in A$ is backward if and only if v is an ancestor of u. We say that a backward arc (u, v) for B_r^+ ***goes over*** a vertex p if $p \in B_r^+]v, u]$. It means that a backward arc $(u, v) \in B_r^+$ always goes over u. Figure 1 shows an out-branching B_r^+ together with crossing and backward arcs in B_r^+.

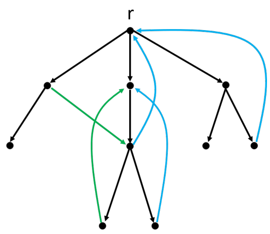

Figure 1 The figure shows $D = (V, A)$. An out-branching B_r^+ is shown in black, the crossing arcs for B_r^+ are shown in green, and the backwards arcs are shown in blue.

▶ **Definition 2.** *For an out-branching B_r^+ in a digraph D a backward arc (u, v) is **irrelevant** with respect to B_r^+ if and only if every path from r to u in D contains v. A backward arc which is not irrelevant is **relevant**.*

Definition 2 implies the following.

▶ **Observation 3.** *An arc a is irrelevant wrt [2] some out-branching B_r^+ if and only if a is not contained in any out-branching B_r^{*+}, that is, a is irrelevant in any out-branching B_r^{*+}.*

In the rest of this section we consider a fixed out-branching B_r^+, unless stated otherwise.

▶ **Lemma 3.1.** *It is possible to find the set of all relevant arcs and the set of all irrelevant arcs for B_r^+ in polynomial time.*

Proof. For any arc $(u, v) \in A(D) \setminus A(B_r^+)$ it can clearly be determined in linear time if the path $B_r^+[v, u]$ exists and, therefore, whether the arc is backward. For every backward arc $(u, v) \in A(D)$ we can also determine in polynomial time if there exists a path $P_{r,u}$ in D which does not contain v, e.g., by determine if a (r, u)-path exists in $D - \{v\}$. ◀

Given the set of relevant arcs wrt B_r^+ in D we can now define the following relation between the relevant arcs and a vertex $u \in V$.

▶ **Definition 4.** *Let $R(u)$ be the set of those relevant arcs wrt B_r^+ which go over u. The **joint relevant arc set** for u is (recursively) defined as the arc set $J(u) = R(u) \cup \left(\bigcup_{h \in \mathcal{H}(R(u))} J(h)\right)$.*

[2] We use wrt for an abbreviation of "with respect to".

Hence for every vertex $u \in B_r^+$ the joint relevant arc set for u is defined recursively as the union of $R(u)$ and the joint relevant arc sets of all the heads of $R(u)$. Figure 2 shows a joint relevant arc set of a vertex v in a out-branching B_r^+. For notational purpose we let $H(u)$ denote the set $\mathcal{H}(J(u))$.

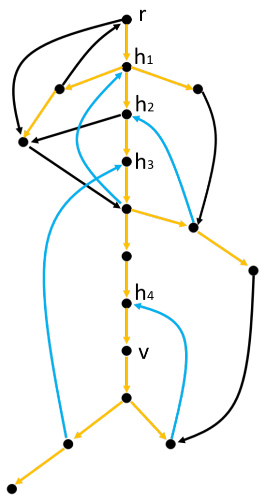

Figure 2 The Figure shows D. $A(B_r^+)$ is shown in yellow and $J(v)$ is shown in blue.

▶ **Lemma 3.2.** *For every vertex $u \in V$ such that $\mathcal{H}(u) \neq \emptyset$, it holds that for the last vertex x on $B_r^+[r,u]$ which is also in $\mathcal{H}(u)$ we have $\mathcal{H}(x) = \mathcal{H}(u) - \{x\}$.*

In Figure 2 we have $\mathcal{H}(v) = \{h_1, h_2, h_3, h_4\}$. Lemma 3.2 means that $\mathcal{H}(h_4) = \{h_1, h_2, h_3\}$. The following is the proof for Lemma 3.2.

Proof. Let $u \in V$ be a vertex such that $\mathcal{H}(u) \neq \emptyset$ and let $l = |\mathcal{H}(u)|$. Observe that for every $h \in \mathcal{H}(u)$ it holds that $h \in B_r^+[r,u]$. Now name the vertices in $\mathcal{H}(u)$ as $h_1, h_2, \cdots, h_l$ such that h_j comes before h_i on $B_r^+[r,u]$ if $j < i$. Let i be an integer in the range $[1, l-1]$. Since $h_i \in \mathcal{H}(u)$ it holds that there exists a relevant arc (t, h_i) which goes over either u or a head $h_p \in \mathcal{H}(u)$ where $i < p \leq l$. Observe that $h_{i+1} \in B_r^+[h_i, u]$ and $h_{i+1} \in B_r^+[h_i, h_j]$ for $j > i$. Hence (t, h_i) goes over h_{i+1}. Therefore, $h_i \in \mathcal{H}(h_{i+1})$. By induction it means that for $i > 0$ we have $\{h_1, h_2, \cdots, h_{i-1}\} \subseteq \mathcal{H}(h_i)$. Now observe that no backward arc with the head h_j for $j \geq i$ can go over h_i as $h_j \in B_r^+[h_i, u]$. Hence $\{h_i, h_{i+1}, \cdots, h_l\} \cap \mathcal{H}(h_i) = \emptyset$. As $h_i \in \mathcal{H}(u)$ we have that $J(h_i) \subseteq J(u)$. Hence $\mathcal{H}(h_i) \subseteq \mathcal{H}(u)$. As a result $\mathcal{H}(h_i) = \{h_1, h_2, \cdots, h_{i-1}\}$. For $i = l$ we therefore have that $\mathcal{H}(h_l) = \{h_1, h_2, \cdots, h_{l-1}\} = \mathcal{H}(u) - \{h_l\}$. ◀

▶ **Lemma 3.3.** *Let C be the set of crossing arcs wrt. to B_r^+. Then for every $v \in V$ we have $H(v) \subseteq H(u)$ for some arc $(w,u) \in C$, where it is possible that $v = u$.*

Proof. Assume for contradiction that there exists a vertex $x \in V$ such that $H(x) \not\subseteq H(u)$ for every arc $(v,u) \in \mathcal{C}$ where we set $H(u) = \emptyset$ if $C = \emptyset$. Among all such vertices choose x such that $|H(x)|$ is maximized. Recall that for every $h \in H(x)$ we have $h \in B_r^+[r,x]$. Now let h be the last vertex on $B_r^+[r,x]$ which is also in $H(x)$. Observe $h \in H(x)$ and there are no vertices in $B_r^+]h,x] \cap H(x)$. Therefore, there must exist an arc $(t,h) \in R(x)$ which goes over x, and since it is relevant there is a path $P_{r,t}$ in D which does not contain h. As a consequence, there is an arc $(p,q) \in P_{r,t}$ which is either a backward arc or a crossing arc such that it's head is in $P_{r,t} \cap B_r^+]h,t]$. Figure 3 shows two possible situations where the arc (p,q) is contained in $P_{r,t}$.

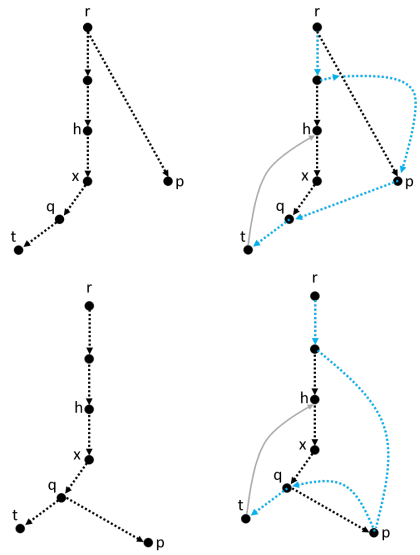

Figure 3 In the top left of the Figure a possible section of B_r^+ is shown in black. In the top right, the path $P_{r,t}$ is shown in blue and the arc (t, h) is shown in gray. The bottom left figure shows an other possible section of B_r^+ and the bottom right figure shows the path $P_{r,t}$in blue and the arc (t, h) in gray.

Observe that since $q \in B_r^+]h, t]$ the arc (t, h) goes over q, implying that $h \in H(q)$ and therefore $\{h\} \cup H(h) \subseteq H(q)$. From Lemma 3.2 we have that $H(h) = H(x) - \{h\}$. Hence $H(x) \subseteq H(q)$. As we have assumed for contradiction that there is no crossing arc (v, u) such that $H(x) \nsubseteq H(u)$ it follows that the arc (p, q) can not be crossing. Hence the arc (p, q) must be backward. Since $(p, q) \in P_{r,t}$ the path $P_{r,t}[r, p]$ does not contain q and, therefore, (p, q) is relevant. Thus $(p, q) \in R(p)$ and $q \in H(p)$. We therefore have that $\{q\} \cup H(q) \subseteq H(p)$. Hence $\{q\} \cup H(x) \subseteq H(p)$. But it means that $|H(x)| < |H(p)|$ which contradicts that x was chosen such that $|H(x)| \geq |H(p)|$. ◀

3.1 Existence of Substitute-path for the Out-branching B_r^+

Now we define a substitute-path for the out-branching B_r^+ in the digraph $D = (V, A)$.

▶ **Definition 5.** *Let u be a vertex in B_r^+, then a* ***substitute-path*** *is a path $S_{r,u}$ from r to u in D such that*

$$|J(u) \cap A(S_{r,u})| + |H(u) \setminus V(S_{r,u})| \geq \left\lceil \frac{|H(u)|}{2} \right\rceil. \tag{1}$$

Thus if there exists a substitute-path $S_{r,u}$ to a vertex u in B_r^+ then $S_{r,u}$ either contain at most half of the vertices in $H(u)$ or for each vertex from $H(u)$ it contains more than the half of the vertices in $H(u)$ it also contain an arc from $J(u)$. Let q be the number of vertices from $H(u)$ which are on a substitute-path $S_{r,u}$. It then follows that the number of arcs from $J(u)$ on $S_{r,u}$ is at least $q - \lceil \frac{|H(u)|}{2} \rceil$. Figure 4 shows two substitute-paths in an out-branching.

The following result may be of independent interests. Theorem 3.1 means that every vertex $v \in V$ is reachable from r by a substitute-path.

▶ **Theorem 3.1.** *For the out-branching B_r^+ there exists a substitute-path to every $v \in V$.*

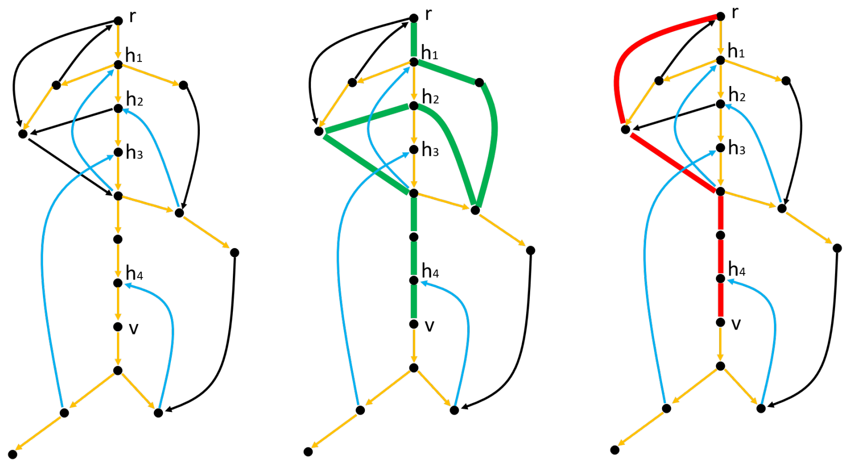

Figure 4 To the left, the digraph $D = (V, A)$ is shown. The arcs of the out-branching B_r^+ are shown in yellow, the arcs in $J(v)$ are blue, and the remaining arcs are black. In the middle and to the right, two paths from r to v are shown. One of the paths is green, the other is shown in red. Both paths are substitute-paths to v.

Proof. In this proof we will be modifying paths and, therefore, we will first consider two claims which deals with these modifications.

▷ Claim 6. For an arbitrary path $P_{r,v}$ in D the following inequality holds for every vertex $x \in V(P_{r,v})$

$$|J(v) \cap A(P_{r,v}[r,x])| + |H(v) \setminus V(P_{r,v}[r,x])| \geq |J(v) \cap A(P_{r,v})| + |H(v) \setminus V(P_{r,v})|. \quad (2)$$

Proof. As $P_{r,v}[r,x] \subseteq P_{r,v}$ we clearly have that $|J(v) \cap A(P_{r,v}[r,x])| \leq |J(v) \cap A(P_{r,v})|$. Let $j = |J(v) \cap A(P_{r,v})| - |J(v) \cap A(P_{r,v}[r,x])|$. For every arc $(t,h) \in J(v)$ which is also on the path $P_{r,v}$ it follows that $h \in H(v)$ and $h \in V(P_{r,v})$. Thus, the set $H(v) \setminus V(P_{r,v}[r,x])$ contains at least j vertices more than $H(v) \setminus V(P_{r,v})$. That is, $|H(v) \setminus V(P_{r,v}[r,x])| - j \geq |H(v) \setminus V(P_{r,v})|$. From this (2) follows. ◁

▷ Claim 7. Given a vertex $v \in B_r^+$, and two paths $P_{r,x}$ and $P_{x,u}$ in D which are disjoint on the vertex set $H(v) - \{x\}$ the following inequality holds for the walk $W = P_{r,x}P_{x,u}$ and for every (r,u)-path $P = P_{r,x}P_{x,u}$ obtained from W by deleting cycles.

$$|J(v) \cap A(W)| + |H(v) \setminus V(W)| \leq |J(v) \cap A(P)| + |H(v) \setminus V(P)|. \quad (3)$$

Proof. If $|J(v) \cap A(W)| > |J(v) \cap A(P)|$ then there must exist one or more cycles in W which contain arcs from $J(v)$. For each such cycle there must exist at least one vertex $z \in (V(P_{r,x}) \cap V(P_{x,u})) - \{x\}$. Since $P_{r,x}$ and $P_{x,u}$ are disjoint on the set $H(v) - \{x\}$ we have that $z \notin H(v)$. It means that for every arc $(t,h) \in J(v)$ contained in W but not in P the head h is not in P either. Therefore, for $j = |J(v) \cap A(W)| - |J(v) \cap A(P)|$ we have $|H(v) \setminus V(P)| - j \geq |H(v) \setminus V(W)|$ and Claim 7 follows. ◁

We are now ready to prove that there exists a substitute-path to every $v \in V$. We will prove this by induction over the cardinality of $H(v) = \mathcal{H}(J(v))$. Clearly, $|H(r)| = 0$ since no backwards arc goes over r. Hence there exists at least one vertex $v \in V$ for which $|H(v)| = 0$.

For every such vertex the path $B_r^+[r, v]$ is clearly a substitute-path by (1). Thus there exists a substitute-path to every $v \in V$ for which $|H(v)| = 0$. Now assume for some positive integer $i \le n$ that for every vertex $v \in V$ where $|H(v)| < i$ it holds that there exists a substitute-path to v.

Let $u \in V$ be a vertex for which $|H(u)| = i$. If no such vertex exists in B_r^+ then clearly there exists a substitute-path to every vertex $u \in V$ where $|H(u)| = i$. Hence assume that u exists. Recall that $H(u) \subseteq V(B_r^+[r, u])$. Denote the vertices in $H(u)$ by $h_0, h_1, \ldots, h_{i-1}$ such that $B_r^+[r, u] = B_r^+[r, h_0]B_r^+[h_0, h_1] \cdots B_r^+[h_{i-2}, h_{i-1}]B_r^+[h_{i-1}, u]$.

▷ Claim 8. $H(h_j) = \{h_0, h_1, \cdots, h_{j-1}\}$ for every $j \le i - 1$.

Proof. From Lemma 3.2 it follows that $H(h_{i-1}) = \{h_0, h_1, \cdots, h_{i-2}\}$. Now the claim follows from Lemma 3.2 by induction over $j \le i - 1$. ◁

Since $h_j \in B_r^+[r, h_{i-1}]$ for $j \le i - 1$ no backwards arc with head in h_{i-1} can go over h_j. As $h_{i-1} \in H(u)$ this means that there exists a relevant arc which goes over u. Fix such an arc (t, h_{i-1}).

▷ Claim 9. If there exists a path $P_{r,t}$ in D such that

$$|J(u) \cap A(P_{r,t})| + |H(u) \setminus V(P_{r,t})| \ge \lceil \frac{i}{2} \rceil. \tag{4}$$

then there is a substitute-path to u

Proof. Assume that there exists a path $P_{r,t}$ such that $|J(u) \cap A(P_{r,t})| + |H(u) \setminus V(P_{r,t})| \ge \lceil \frac{i}{2} \rceil$. Let x be the first vertex on $P_{r,t}$ which is also on the path $B_r^+[h_{i-1}, t]$. If $x = u$ then the path $P_{r,t}[r, u]$ will be a substitute-path to u as $|J(u) \cap A(P_{r,t})| + |H(u) \setminus V(P_{r,t})| \le |J(u) \cap A(P_{r,t}[r, u])| + |H(u) \setminus V(P_{r,t}[r, u])|$ by Claim 6. Therefore, we may assume that $x \ne u$. We will consider the two cases: $x \in B_r^+[h_{i-1}, u]$ or $x \in B_r^+[u, t]$. By claim 6 it holds that

$$\begin{aligned} &|J(u) \cap A(P_{r,t}[r, x])| + |H(u) \setminus V(P_{r,t}[r, x])| \\ &\qquad \ge |J(u) \cap A(P_{r,t})| + |H(u) \setminus V(P_{r,t})| \ge \lceil \frac{i}{2} \rceil. \end{aligned} \tag{5}$$

If $x \in B_r^+[h_{i-1}, u]$ then for the walk $W'_{r,u} = P_{r,t}[r, x]B_r^+[x, u]$ it clearly holds that $|J(u) \cap A(P_{r,t}[r, x])| + |H(u) \setminus V(P_{r,t}[r, x])| = |J(u) \cap A(W'_{r,u})| + |H(u) \setminus V(W'_{r,u})|$. For the path $P'_{r,u} = P_{r,t}[r, x]B_r^+[x, u]$ we therefore have by Claim 7 that $|J(u) \cap A(P'_{r,u})| + |H(u) \setminus V(P'_{r,u})| \ge |J(u) \cap A(P_{r,t}[r, x])| + |H(v) \setminus V(P_{r,t}[r, x])|$. From (5) it therefore follows that $|J(u) \cap A(P'_{r,u})| + |H(u) \setminus V(P'_{r,u})| \ge \lceil \frac{i}{2} \rceil$. Hence $P'_{r,u}$ is a substitute-path to u. Now consider the second case where $x \in B_r^+[u, t]$. Consider the (x, u)-walk $P''_{x,u} = B_r^+[x, t](t, h_{i-1})B_r^+[h_{i-1}, u]$ and observe that $J(u) \cap A(P''_{x,u}) = (t, h_{i-1})$ and $H(u) \cap V(P''_{x,u}) = h_{i-1}$. It means that $P''_{x,u}$ is disjoint from $H(u) - \{h_{i-1}\}$. As x is the first vertex on $P_{r,t}$ which is on $B_r^+[h_{i-1}, t]$ and $x \in B_r^+[u, t]$ it holds that $h_{i-1} \notin P_{r,t}[r, x]$ and $P_{r,t}[r, x]$ and $P''_{x,u}$ are disjoint except in x. Now consider the $P^*_{r,u} = P_{r,t}[r, x]P''_{x,u}$ and observe that

$$\begin{aligned} &|J(u) \cap A(P^*_{r,u})| + |H(u) \setminus V(P^*_{r,u})| \\ &\qquad = |J(v) \cap A(P_{r,t}[r, x])| + 1 + |H(v) \setminus V(P_{r,t}[r, x])| - 1 \\ &\qquad = |J(v) \cap A(P_{r,t}[r, x])| + |H(v) \setminus V(P_{r,t}[r, x])| \end{aligned}$$

From (5) it therefore follows that $|J(u) \cap A(P^*_{r,u})| + |H(u) \setminus V(P^*_{r,u})| \ge \lceil \frac{i}{2} \rceil$. Thus $P^*_{r,u}$ is a substitute-path to u. ◁

Now we prove that there always exists a path such that (4) holds. Recall by the definition of a relevant arc, that the relevant arc (t, h_{i-1}) goes over t and there exists a path $P_{r,t}$ in D such that $h_{i-1} \notin P_{r,t}$. If $|H(u) \setminus V(P_{r,t})| = i$ then (4) clearly holds. Hence we may assume that there exists a vertex $h_j \in H(u) \cap V(P_{r,t})$. Now let h_l be the last such vertex on the path $P_{r,t}$, that is, no vertex $h_j \in H(u) - \{h_l\}$ is on the path $P_{r,t}[h_l, t]$. Observe that since $P_{r,t}$ does not contain h_{i-1} we have

$$l \leq i - 2 \tag{6}$$

By Claim 8, $|H(h_j)| = j$ for every $j \leq i - 1$. By the induction assumption it therefore follows that there is a substitute-path S_{r,h_j} to every $h_j \in H(u)$.

▷ Claim 10. For every substitute-path to h_l, S_{r,h_l}, either there exists a vertex $h_j \in S_{r,h_l}$ such that $j > l$ or (4) holds.

Proof. Assume that there exists a substitute-path S_{r,h_l} to h_l such that there is no vertex $h_j \in S_{r,h_l}$ for $j > l$. Observe that since S_{r,h_l} is a substitute-path it holds that $|J(h_l) \cap A(S_{r,h_l})| + |H(h_l) \setminus V(S_{r,h_l})| \geq \lceil \frac{l}{2} \rceil$. Now consider the walk $W'_{r,t} = S_{r,h_l} P_{r,t}[h_l, t]$. The path S_{r,h_l} does not contain any heads $h_j \in H(u)$ for $j > l$ and $P_{r,t}[h_l, t]$ does not contain any heads $h_j \in H(u)$ for $j \neq l$. Hence we can deduce that:

$$\begin{aligned}
&|J(u) \cap A(W'_{r,t})| + |H(u) \setminus V(W'_{r,t})| \\
&\qquad \geq |J(h_l) \cap A(S_{r,h_l})| + |H(h_l) \setminus V(S_{r,h_l})| + |\{h_j | l < j < i\}| \\
&\qquad = |J(h_l) \cap A(S_{r,h_l})| + |H(h_l) \setminus V(S_{r,h_l})| + i - l - 1 \\
&\qquad \geq \lceil \frac{l}{2} \rceil + i - l - 1
\end{aligned}$$

From this and Claim 7 we have the following inequality for the path $P'_{r,t} = S_{r,h_l} P_{r,t}[h_l, t]$.

$$|J(u) \cap A(P'_{r,t})| + |H(u) \setminus V(P'_{r,t})| \geq \lceil \frac{l}{2} \rceil + i - l - 1 \tag{7}$$

From (6) it follows that $l \leq i - 2$ and since $l \geq 0$ it also holds that $i \geq 2$. These observations together with (7) give us the following inequality.

$$|J(u) \cap A(P'_{r,t})| + |H(u) \setminus V(P'_{r,t})| \geq \lceil \frac{l}{2} \rceil + i - l - 1 \geq \lceil \frac{i}{2} \rceil$$

It means that $P'_{r,t}$ is a path making (4) true. Hence if (4) does not hold we may assume that for every substitute-path S_{r,h_l} to h_l there exists a $j > l$ such that $h_j \in S_{r,h_l}$. ◁

Let S_{r,h_j} be a substitute-path to $h_j \in H(u)$. If there exists a vertex $h_p \in S_{r,h_j}$ such that $p > j$ we will call the first vertex h_p with $p > j$ on the path S_{r,h_j} the **first exceeding head**. Recall that we are proving that a path exists such that (4) holds. Assume for contradiction that no such path exists. Thus by Claim 10 it follows that for every substitute-path S_{r,h_l} to h_l there is a vertex $h_j \in S_{r,h_l}$ for $j > l$, that is, every substitute-path S_{r,h_l} will contain a first exceeding head h_j such that $j > l$. Now let $k \leq l$ be the smallest integer such that for every substitute-path S_{r,h_k} the first exceeding head h_j has $j > l$. Observe that l fulfills this property and therefore k exists. Let S'_{r,h_k} be a substitute-path to h_k and h_j the first exceeding head on this path. Observe that $S'_{r,h_k}[r, h_j]$ is disjoint from the heads $\{h_k, h_{k+1}, \cdots, h_{j-1}\}$. Therefore we have:

$$\begin{aligned}
&|J(h_j) \cap A(S'_{r,h_k}[r, h_j])| + |H(h_j) \setminus V(S'_{r,h_k}[r, h_j])| \\
&\qquad \geq |J(h_k) \cap A(S'_{r,h_k})| + |H(h_k) \setminus V(S'_{r,h_k})| + |\{h_p | k \leq p < j\}| \\
&\qquad \geq |J(h_k) \cap A(S'_{r,h_k})| + |H(h_k) \setminus V(S'_{r,h_k})| + j - k
\end{aligned} \tag{8}$$

As S'_{r,h_k} is a substitute-path to h_k and $|H(h_k)| = k$ we have that $|J(h_k) \cap A(S'_{r,h_k})| + |H(h_k) \setminus V(S'_{r,h_k})| \geq \lceil \frac{k}{2} \rceil$. When we combine this with (8) we obtain the following inequality:

$$|J(h_j) \cap A(S'_{r,h_k}[r,h_j])| + |H(h_j) \setminus V(S'_{r,h_k}[r,h_j])| \geq \lceil \frac{k}{2} \rceil + j - k \tag{9}$$

Now consider the (r,t)-path $P^* = S'_{r,h_k}[r,h_j]B_r^+[h_j,t]$. Recall that the path $B_r^+[h_j,t]$ is disjoint from every head h_p for $p < j$ and $S'_{r,h_k}[r,h_j]$. It therefore follow from (9) that.

$$\begin{aligned}
&|J(u) \cap A(P^*)| + |H(u) \setminus V(P^*)| \\
&\qquad \geq |J(h_j) \cap A(S'_{r,h_k}[r,h_j])| + |H(h_j) \setminus V(S'_{r,h_k}[r,h_j])| \\
&\qquad \geq \lceil \frac{k}{2} \rceil + j - k
\end{aligned} \tag{10}$$

From (10) and the fact that $l + 1 \leq j$ we obtain the following inequality.

$$|J(u) \cap A(P^*)| + |H(u) \setminus V(P^*)| \geq l + 1 - \lfloor \frac{k}{2} \rfloor \tag{11}$$

From the assumption that (4) does not hold it follows that $|J(u) \cap A(P^*[r,t])| + |H(u) \setminus V(P^*[r,t])| < \lceil \frac{i}{2} \rceil$. Therefore, we have the following observation from (11).

▶ **Observation 11.** *$l + 1 - \lfloor \frac{k}{2} \rfloor < \lceil \frac{i}{2} \rceil$.*

Recall that the path $B_r^+[r,h_0]$ is a substitute-path to h_0 and as every substitute-path to h_k has a first exceeding head h_p such that $p > l$ it must hold that $k > 0$. Furthermore, recall that k was chosen as the smallest integer such that the first exceeding head h_p on every substitute-path to h_k had $p > l$. Hence there exist a substitute-path $S^*_{r,h_{k-1}}$ such that either there is no exceeding head or for the first exceeding head h_p it holds that $k - 1 < p \leq l$. If there is no exceeding head let $p = k - 1$ and otherwise let h_p be the first exceeding head. Consider $S^*_{r,h_{k-1}}[r,h_p]$. Observe by Claim 6 and the fact that $S^*_{r,h_{k-1}}$ is a substitute-path that

$$|J(h_{k-1}) \cap A(S^*_{r,h_{k-1}}[r,h_p])| + |H(h_{k-1}) \setminus V(S^*_{r,h_{k-1}}[r,h_p])| \geq \lceil \frac{k-1}{2} \rceil$$

Now consider the (r,t)-walk $W'_{r,t} = S^*_{r,h_{k-1}}[r,h_p]B_r^+[h_p,h_l]P_{r,t}[h_l,t]$. Note that $S^*_{r,h_{k-1}}[r,h_p]$ is disjoint from every head h_q for $q > p$, $B_r^+[h_p,h_l]$ is disjoint from every head h_q for $q < p$ and $q > l$, and furthermore $P_{r,t}[h_l,t]$ is disjoint from $H(u) - \{h_l\}$. Hence:

$$\begin{aligned}
&|J(u) \cap A(W')| + |H(u) \setminus V(W')| \\
&\quad = |J(h_{k-1}) \cap A(S^*_{r,h_{k-1}}[r,h_p])| + |H(h_{k-1}) \setminus V([S^*_{r,h_{k-1}}[r,h_p])| + |\{h_q | l < q < i\}| \\
&\quad = |J(h_{k-1}) \cap A(S^*_{r,h_{k-1}}[r,h_p])| + |H(h_{k-1}) \setminus V(S^*_{r,h_{k-1}}[r,h_p])| + i - 1 - l
\end{aligned} \tag{12}$$

Consider the (r,t)-path $P' = S^*_{r,h_{k-1}}[r,h_p]B_r^+[h_p,h_l]P_{r,t}[h_l,t]$. Then from (12), claim 7 and the fact that $S^*_{r,h_{k-1}}$ is a substitute-path to h_{k-1} we obtain the following:

$$\begin{aligned}
&|J(u) \cap A(P')| + |H(u) \setminus V(P')| \\
&\qquad \geq |J(u) \cap A(W')| + |H(u) \setminus V(W')| \\
&\qquad \geq \lceil \frac{k-1}{2} \rceil + i - 1 - l
\end{aligned} \tag{13}$$

By the assumption that (4) does not hold we have $|J(u) \cap A(P')| + |H(u) \setminus V(P')| < \lceil \frac{i}{2} \rceil$. This combined with (13) gives us:

$$\lceil \frac{i}{2} \rceil > \lceil \frac{k-1}{2} \rceil + i - 1 - l \quad \Leftrightarrow \quad l + 1 - \lceil \frac{k-1}{2} \rceil > \lfloor \frac{i}{2} \rfloor \tag{14}$$

Combining Observation 11 with (14) and the fact that $\lceil \frac{k-1}{2} \rceil = \lfloor \frac{k}{2} \rfloor$ we obtain the following inequality:

$$\lfloor \frac{i}{2} \rfloor < l + 1 - \lfloor \frac{k}{2} \rfloor < \lceil \frac{i}{2} \rceil \tag{15}$$

Now observe that $l + 1 - \lfloor \frac{k}{2} \rfloor$ is an integer and from (15) we therefore have a contradiction. Hence either the path P^* or the path P' makes (4) true. It means that there exists a path making (4) true. Thus by Claim 9 there exists a substitute-path to u.

Recall that u was an arbitrary fixed vertex $x \in V$ for which $|H(x)| = i$. Thus we can conclude that for every vertex $x \in V$ where $|H(x)| = i$ there exists a substitute path to x. This concludes the induction step over for every vertex $x \in V$ where $|H(x)| = i$.Thus we can conclude that for every vertex $x \in V$ where $|H(x)| = i$ for $i \in [0, |V|]$ there exists a substitute-path. As $|H(x)| \leq n$ for every $x \in V$, the proof is complete. ◀

3.2 Finding Desired Paths in Polynomial Time

We now obtain the following lemma about finding a substitute-path to a vertex $v \in V$.

▶ **Lemma 3.4** (⋆). *There exists a polynomial time algorithm which given an out-branching B_r^+ in $D = (V, A)$ and a vertex $v \in V$ finds a substitute-path to v.*

▶ **Lemma 3.5** (⋆). *For every vertex $u \in V(B_r^+)$ there exists a path $P_{r,u}$ from r to u such that*

$$|R(u) \cap A(P_{r,u})| + |\mathcal{H}(R(u)) \setminus V(P_{r,u})| \geq \left\lceil \frac{|\mathcal{H}(R(u))|}{2} \right\rceil \tag{16}$$

and it can be found in polynomial time

▶ **Lemma 3.6** (⋆). *For every vertex $v \in V(B_r^+)$ let $R_v \in \{J(v), R(v)\}$. If there exists a path $P_{r,v}$ such that*

$$|R_v \cap A(P_{r,v})| + |\mathcal{H}(R_v) \setminus V(P_{r,v})| \geq \left\lceil \frac{|\mathcal{H}(R_v)|}{2} \right\rceil \tag{17}$$

then there exists an out-branching $\widehat{B}_r^+$ such that $|R_v \cap A(\widehat{B}_r^+)| \geq \left\lceil \frac{|\mathcal{H}(R_v)|}{2} \right\rceil$, and $\widehat{B}_r^+$ can be found in polynomial time.

4 Backward arcs, Crossing Arcs and Substitute-paths with respect to In-branchings

For in-branchings we have similar definitions and results as we have for out-branchings. Given a digraph $D = (V, A)$ which contains a fixed in-branching B_r^- we can create a corresponding digraph $D' = (V, A')$ and an out-branching B_r^+ by creating an arc (v, u) in A' for each arc $(u, v) \in A$. That is, the direction of the arcs in A are flipped. Observe that the in-branching B_r^- in D will correspond to a fixed out-branching B_r^+ in D'. Due to this one-to-one correspondence all the results given in Section 3 regarding a fixed out-branching will be turned into equivalent statements about a fixed in-branching.

5 A kernel for Rooted k-Distinct Branchings

In this section we prove the existence of an $\mathcal{O}(k^2)$-vertex kernel for the problem Rooted k-Distinct Branchings. The proof of the next lemma will give the desired Theorem 1.1.

▶ **Lemma 5.1.** *Given an instance $(D = (V, A), k, s, t)$ of R-k-DB; In polynomial time we can either find a kernel (D', k, s', t') such that $|V(D')| < 16k^2 + 5k = \mathcal{O}(k^2)$ or determine the answer to the instance.*

Proof. In linear time it can be determined if an out-branching B_s^+ and an in-branching B_t^- exist. If one of these do not exist then clearly (D, k, s, t) is a no-instance. Hence assume that both exist. At any time during the proof let $\mathcal{C}^+$, $\mathcal{R}^+$, and $\mathcal{I}^+$, respectively, denote the crossing, relevant, and irrelevant arcs wrt.. B_s^+. Similarly, let $\mathcal{C}^-$, $\mathcal{R}^-$, and $\mathcal{I}^-$, respectively, denote the crossing, relevant, and irrelevant arcs wrt.. B_t^-. Moreover, at all times during the proof let $F = A(D) \setminus \big(A(B_s^+) \cup A(B_t^-)\big)$ be defined as the **free arcs** and at all times let $E^+ = A(B_s^+) \setminus A(B_t^-)$, $E^- = A(B_t^-) \setminus A(B_s^+)$ denote the **exclusive set**, respectively, for B_s^+ and B_t^-. Note that $|E^+| = |E^-|$. Now we execute the following procedure.

▶ **Procedure 5.1.** *Change B_s^+ and B_t^- as follows until no longer possible or $|E^+| \geq k$.*

1. *If there exists an arc $(u, v) \in F \cap \mathcal{C}^+$ such that $(\mathcal{P}_{B_s^+}(v), v) \notin E^+$, then remove $(\mathcal{P}_{B_s^+}(v), v)$ from B_s^+ and insert the arc (u, v) into B_s^+.*
2. *If there exists an arc $(u, v) \in F \cap \mathcal{C}^-$ such that $(u, \mathcal{P}_{B_t^-}(u)) \notin E^-$, then remove $(u, \mathcal{P}_{B_t^-}(u))$ from B_t^- and insert the arc (u, v) in to B_t^-.*

▶ **Lemma 5.2** ($\star$)**.** *Procedure 5.1 can be executed in polynomial time.*

If we have not found a solution when Procedure 5.1 terminates, then we have

$$|E^+| = |E^-| < k. \tag{18}$$

Since $|E^+| < k$ and we can not change B_s^+ further in Procedure 5.1 we have that for every crossing arc $(u, v) \in \mathcal{C}^+$ either $(u, v) \in A(B_t^-)$ or $(p_{B_s^+}(v), v) \in E^+$. As C^+ is disjoint from B_s^+ it follows that every arc $(u, v) \in A(B_t^-) \cap \mathcal{C}^+$ is contained in E^-. Hence there are less than k such arcs. For the arcs $(u, v) \in \mathcal{C}^+$ where $(p_{B_s^+}(v), v) \in E^+$ there is less than k different heads as $|E^+| < k$. Therefore, there must be less than $2k$ different heads for the crossing arcs of B_s^+, that is, $|\mathcal{H}(C^+)| < 2k$. Similarly, we have that there are less than $2k$ different tails of C^-, that is, $|\mathcal{T}(C^-)| < 2k$.

▶ **Observation 12.** $|\mathcal{H}(C^+)| < 2k$ *and* $|\mathcal{T}(C^-)| < 2k$

A vertex $v \in V$ is a **Type 1** vertex if it is the tail of an arc $(v, u) \in \mathcal{R}^- \cup \mathcal{C}^-$ and a **Type 2** vertex if it is the head of an arc $(u, v) \in \mathcal{R}^+ \cup \mathcal{C}^+$ Note that a vertex $v \in V$ can be both a Type 1 and a Type 2 vertex. Now we have the following reduction rule.

▶ **Reduction Rule 5.1.** *As long as there exists an arc $(u, v) \in A(B_s^+) \cap A(B_t^-)$ such that u is not a Type 1 vertex and v is not a Type 2 vertex contract (u, v) into one vertex.*

▶ **Lemma 5.3** ($\star$)**.** *Reduction Rule 5.1 is safe and can be applied in polynomial time.*

Let D_R, B_s^+ and B_t^- be the digraph and branchings that we have obtained after performing Reduction 5.1. We now have the following lemma.

▶ **Lemma 5.4.** *If $|V(D_R)| \geq 16k^2 + 5k$ then a solution exists.*

Proof. Assume $|V(D_R)| \geq 16k^2 + 5k$. As B_s^+ is an out-branching we have $|A(B_s^+)| = |V(D_R)| - 1$. For every arc $(u, v) \in A(B_s^+)$ it holds that either $(u, v) \notin A(B_t^-)$, u is a Type 1 vertex or v is a Type 2 vertex. By (18) we see that $|A(B_s^+) \setminus A(B_t^-)| < k$. It means that $|A(B_s^+) \cap A(B_t^-)| \geq |V(D_R)| - k \geq 2 \cdot (8k^2 + 2k)$. For every arc $(u, v) \in A(B_s^+) \cap A(B_t^-)$ we have that u is a Type 1 vertex or v is a Type 2 vertex. Hence we can conclude that either there is $8k^2 + 2k$ vertices of Type 1 or $8k^2 + 2k$ vertices of Type 2 (or both). In the following we will only explicitly give the proof that a solution exists if there are at least $8k^2 + 2k$ vertices of Type 2 as the proof for a solution exists if there are at least $8k^2 + 2k$ vertices of Type 1 will follow from the symmetry between B_s^+, $\mathcal{C}^+$, $\mathcal{R}^+$, $\mathcal{I}^+$ and B_t^-, $\mathcal{C}^-$, $\mathcal{R}^-$, $\mathcal{I}^-$. Assume therefore that there are at least $8k^2 + 2k$ vertices of Type 2. Recall that it means, there are at least $8k^2 + 2k$ different vertices which are heads of the arcs $\mathcal{R}^+ \cup \mathcal{C}^+$. By Observation 12 there are less than $2k$ vertices which are heads of $\mathcal{C}^+$. Consequently, the number of vertices which are heads of $\mathcal{R}^+$ must be larger than $8k^2$. Clearly, $|\mathcal{H}(\mathcal{R}^+)| = |\cup_{v \in V} \mathcal{H}(J(v))|$ and we therefore have $8k^2 \leq |\mathcal{H}(\mathcal{R}^+)| = |\cup_{v \in V} \mathcal{H}(J(v))|$. By Lemma 3.3 we have that for every $v \in V$ it holds that $\mathcal{H}(J(v)) \subseteq \mathcal{H}(J(u))$ for some arc $(v, u) \in \mathcal{C}^+$. We therefore have that: $|\bigcup_{(v,u) \in \mathcal{C}^+} \mathcal{H}(J(u))| = |\bigcup_{v \in V} \mathcal{H}(J(v))| \geq 8k^2$. By Observation 12 there are at most $2k$ heads of the arcs in $\mathcal{C}^+$ so there must exist at least one arc $(u, v) \in \mathcal{C}^+$ such that $|\mathcal{H}(J(v))| \geq 4k$. Fix (u, v) to be such an arc. By Theorem 3.1 there exist a substitute-path $S[s, v]$ in D such that $|J(v) \cap A(S[s, v])| + |\mathcal{H}(J(v)) \setminus A(S[s, v])| \geq \left\lceil \frac{|\mathcal{H}(J(v))|}{2} \right\rceil$ and therefore by from Lemma 3.6 there exists an out-branching $\widehat{B}_s^+$ such that $|J(v) \cap A(\widehat{B}_s^+)| \geq \left\lceil \frac{|\mathcal{H}(J(v))|}{2} \right\rceil \geq 2k$. Observe that B_s^+ is disjoint from $J(v)$ and therefore if $|J(v) \cap A(B_t^-)| \geq k$ then B_s^+ and B_t^- would have been a solution. If $|J(v) \cap A(B_t^-)| < k$ then $|A(\widehat{B}_s^+) \setminus A(B_t^-)| \geq k$ and $\widehat{B}_s^+$ and B_t^- is a solution. Hence we conclude that if the number of different heads of R^+ is larger than $4k^2$ a solution exists. ◀

From Lemma 5.4 we can conclude that after applying Reduction Rule 5.1 it either holds that we have a solution or $|V(D_R)| < 16k^2 + 5k$. In the first case we have a solution in the latter we have a kernel of size $\mathcal{O}(k^2)$. It therefore only remains to argue that the solution or the kernel can be found in polynomial time. For an instance $(D = (V, A), k, s, t)$ of R-k-DB it is possible in polynomial time to decide if an out-branching B_s^+ and B_t^- in D exists and find them. By Claim 5.2, and Claim 5.3 we can execute the Procedure 5.1 and afterwards apply Reduction Rule 5.1 in polynomial time. Furthermore, it is polynomial to decide if the resulting digraph has at least $16k^2 + 5k$ vertices and applying Lemma 5.4. Hence in polynomial time we can either find a kernel with a vertex set of size less than $16k^2 + 5k$ or determine the answer to the instance. ◀

6 Conclusion

In this paper, we studied the problem of deciding if a digraph $D = (V, A)$, contains an in- and out-branching rooted at specific vertices s and t, such that the in- and out-branching are distinct on at least k arcs. Before this paper, it was not known if the problem admitted a polynomial kernel, and the best known complexity for solving the problem was $2^{\mathcal{O}(k^2 \log^2 k)} n^{O(1)}$. We designed a polynomial kernel for the problem with $\mathcal{O}(k^2)$ vertices and found an algorithm with the complexity $2^{\mathcal{O}(k \log k)} + n^{O(1)}$. To obtain these results, we defined the concept of substitute-paths in out- and in-branchings. This graph-theoretical concept might be useful for obtaining other results on problems regarding in-and out-branchings. It is still open whether there exists a kernel with $\mathcal{O}(k)$ vertices. We believe that using representative set approach applied for obtaining exact exponential time algorithm for finding a strongly connected subgraph of a given digraph with minimum number of arcs [20], it seem possible to get an algorithm for R-k-DB running in time $2^{\mathcal{O}(k)} + n^{\mathcal{O}(1)}$. Making this work seems an interesting direction to pursue.

References

1. Noga Alon, Fedor V. Fomin, Gregory Gutin, Michael Krivelevich, and Saket Saurabh. Spanning directed trees with many leaves. *SIAM J. Discrete Math.*, 23(1):466–476, 2009. doi:10.1137/070710494.
2. J. Bang-Jensen. Edge-disjoint in- and out-branchings in tournaments and related path problems. *J. Combin. Theory Ser. B*, 51(1):1–23, 1991.
3. J. Bang-Jensen and J. Huang. Decomposing locally semicomplete digraphs into strong spanning subdigraphs. *J. Combin. Theory Ser. B*, 102:701–714, 2010.
4. J. Bang-Jensen, S. Saurabh, and S. Simonsen. Parameterized algorithms for non-separating trees and branchings in digraphs. *Algorithmica*, 76(1):279–296, 2016.
5. J. Bang-Jensen and S. Simonsen. Arc-disjoint paths and trees in 2-regular digraphs. *Discrete Applied Mathematics*, 161(16-17):2724–2730, 2013.
6. J. Bang-Jensen, S. Thomassé, and A. Yeo. Small degree out-branchings. *J. Graph Theory*, 42(4):297–307, 2003.
7. J. Bang-Jensen and A. Yeo. The minimum spanning strong subdigraph problem is fixed parameter tractable. *Discrete Appl. Math.*, 156:2924–2929, 2008.
8. J. Bang-Jensen and A. Yeo. Arc-disjoint spanning sub(di)graphs in digraphs. *Theor. Comput. Sci.*, 438:48–54, 2012.
9. K. Bérczi, S. Fujishige, and N. Kamiyama. A linear-time algorithm to find a pair of arc-disjoint spanning in-arborescence and out-arborescence in a directed acyclic graph. *Inform. Process. Lett.*, 109(23-24):1227–1231, 2009.
10. Daniel Binkele-Raible, Henning Fernau, Fedor V. Fomin, Daniel Lokshtanov, Saket Saurabh, and Yngve Villanger. Kernel(s) for problems with no kernel: On out-trees with many leaves. *ACM Transactions on Algorithms*, 8(4):38, 2012. doi:10.1145/2344422.2344428.
11. Nathann Cohen, Fedor V. Fomin, Gregory Gutin, Eun Jung Kim, Saket Saurabh, and Anders Yeo. Algorithm for finding k-vertex out-trees and its application to k-internal out-branching problem. *J. Comput. Syst. Sci.*, 76(7):650–662, 2010. doi:10.1016/j.jcss.2010.01.001.
12. M. Cygan, F.V. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, and S. Saurabh. *Parameterized Algorithms*. Springer, 2015.
13. J. Daligault, G. Gutin, E.J. Kim, and A. Yeo. FPT algorithms and kernels for the directed k-leaf problem. *J. Comput. Syst. Sci.*, 76:144–152, 2010.
14. Frederic Dorn, Fedor V. Fomin, Daniel Lokshtanov, Venkatesh Raman, and Saket Saurabh. Beyond bidimensionality: Parameterized subexponential algorithms on directed graphs. In *STACS*, volume 5, pages 251–262, 2010. doi:10.4230/LIPIcs.STACS.2010.2459.
15. Rodney G. Downey and Michael R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013.
16. J. Edmonds. Edge-disjoint branchings. In *Combinatorial Algorithms*, pages 91–96. Academic Press, 1973.
17. J. Flum and M. Grohe. *Parameterized Complexity Theory*. Springer-Verlag, 2006.
18. Fedor V. Fomin, Serge Gaspers, Saket Saurabh, and Stéphan Thomassé. A linear vertex kernel for maximum internal spanning tree. *J. Comput. Syst. Sci.*, 79(1):1–6, 2013. doi:10.1016/j.jcss.2012.03.004.
19. Fedor V. Fomin, Fabrizio Grandoni, Daniel Lokshtanov, and Saket Saurabh. Sharp separation and applications to exact and parameterized algorithms. *Algorithmica*, 63(3):692–706, 2012. doi:10.1007/s00453-011-9555-9.

20 Fedor V. Fomin, Daniel Lokshtanov, Fahad Panolan, and Saket Saurabh. Efficient computation of representative families with applications in parameterized and exact algorithms. *J. ACM*, 63(4):29:1–29:60, 2016.

21 Fedor V Fomin, Daniel Lokshtanov, Saket Saurabh, and Meirav Zehavi. *Kernelization: theory of parameterized preprocessing.* Cambridge University Press, 2019.

22 Gregory Z. Gutin, Felix Reidl, and Magnus Wahlström. k-distinct in- and out-branchings in digraphs. *J. Comput. Syst. Sci.*, 95:86–97, 2018.

23 J. Kneis, A. Langer, and P. Rossmanith. A new algorithm for finding trees with many leaves. *Algorithmica*, 61(4):882–897, 2011.

24 R. Niedermeier. *Invitation to Fixed Parameter Algorithms.* Oxford University Press, Oxford, 2006.