



HAL
open science

From branchings to flows: a study of an Edmonds' like property to arc-disjoint branching flows

Cláudio Carvalho, Jonas Costa, Raul Lopes, Ana Karolinnna Maia, Nicolas Nisse, Cláudia Linhares Sales

► To cite this version:

Cláudio Carvalho, Jonas Costa, Raul Lopes, Ana Karolinnna Maia, Nicolas Nisse, et al.. From branchings to flows: a study of an Edmonds' like property to arc-disjoint branching flows. [Research Report] UFC; INRIA; CNRS; Université Côte d'Azur; I3S. 2020. hal-03031759v3

HAL Id: hal-03031759

<https://hal.science/hal-03031759v3>

Submitted on 8 Nov 2022 (v3), last revised 24 Apr 2023 (v4)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

1 From branchings to flows: a study of an Edmonds' 2 like property to arc-disjoint branching flows

3 Cláudio Carvalho ✉

4 Departamento de Computação, Universidade Federal do Ceará, Fortaleza, Brazil

5 Jonas Costa ✉

6 Departamento de Computação, Universidade Federal do Ceará, Fortaleza, Brazil

7 Raul Lopes ✉ 

8 Université Paris-Dauphine, PSL University, CNRS UMR7243, LAMSADE, Paris, France

9 Ana Karolinna Maia ✉

10 Departamento de Computação, Universidade Federal do Ceará, Fortaleza, Brazil

11 Nicolas Nisse ✉

12 Université Côte d'Azur, Inria, CNRS, I3S, Sophia-Antipolis, France

13 Cláudia Linhares Sales ✉

14 Departamento de Computação, Universidade Federal do Ceará, Fortaleza, Brazil

15 — Abstract —

16 An s -branching flow f in a network $\mathcal{N} = (D, u)$, where u is the capacity function, is a flow that
17 reaches every vertex in $V(D)$ from s while loosing exactly one unit of flow in each vertex other than
18 s . Bang-Jensen and Bessy [TCS, 2014] showed that, when every arc has capacity $n - 1$, a network \mathcal{N}
19 admits k arc-disjoint s -branching flows if and only if its associated digraph D contains k arc-disjoint
20 s -branchings. Thus a classical result by Edmonds stating that a digraph contains k arc-disjoint
21 s -branchings if and only if the indegree of every set $X \subseteq V(D) \setminus \{s\}$ is at least k also characterizes
22 the existence of k arc-disjoint s -branching flows in those networks, suggesting that the larger the
23 capacities are, the closer an s -branching flow is from simply being an s -branching. This observation
24 is further implied by results by Bang-Jensen et al. [DAM, 2016] showing that there is a polynomial
25 algorithm to find the flows (if they exist) when every arc has capacity $n - c$, for every fixed $c \geq 1$,
26 and that such an algorithm is unlikely to exist for most other choices of the capacities. In this paper,
27 we investigate how a property that is a natural extension of the characterization by Edmonds' relates
28 to the existence of k arc-disjoint s -branching flows in networks. Although this property is always
29 necessary for the existence of the flows, we show that it is not always sufficient and that it is hard
30 to decide if the desired flows exist even if we know beforehand that the network satisfies it. On the
31 positive side, we show that it guarantees the existence of the desired flows in some particular cases
32 depending on the choice of the capacity function or on the structure of the underlying graph of D ,
33 for example. We remark that, in those positive cases, polynomial time algorithms to find the flows
34 can be extracted from the constructive proofs.

35 **2012 ACM Subject Classification** Theory of computation → Network flows

36 **Keywords and phrases** Digraphs, Branchings, Branching flows, Arc-disjoint flows

37 **Digital Object Identifier** 10.4230/LIPIcs.CVIT.2016.23

38 **Funding** Cláudio Carvalho: FUNCAP Pronem 4543945/2016.

39 Jonas Costa: FUNCAP Pronem 4543945/2016 and PhD scholarship granted by CAPES.

40 Raul Lopes: French Agence Nationale de la Recherche under contract ASSK ANR-18-CE40-0025-01.

41 Work partially done at UFC(MDCC), with a PhD scholarship by FUNCAP.

42 Ana Karolinna Maia: FUNCAP 186-155.01.00/21 and 4543945/2016 (Pronem).

43 Nicolas Nisse: French Agence Nationale de la Recherche under contract Digraphs ANR-19-CE48-
44 0013-01.

45 Cláudia Linhares Sales: FUNCAP Pronem 4543945/2016, and CNPq project 304831/2017-4.



© Cláudio Carvalho, Jonas Costa, Cláudia Linhares Sales, Raul Lopes, Ana Karolinna Maia, and
Nicolas Nisse;

licensed under Creative Commons License CC-BY 4.0

42nd Conference on Very Important Topics (CVIT 2016).

Editors: John Q. Open and Joan R. Access; Article No. 23; pp. 23:1–23:15

Leibniz International Proceedings in Informatics



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

46 **1** Introduction

47 Let $D = (V, A)$ be a digraph. If e is an arc of D from a vertex v to a vertex w , we may refer
 48 to e as vw . The *in-degree* of a set $X \subseteq V(D)$, denoted by $d_D^-(X)$, is the number of arcs
 49 with tail outside of X and head inside of X . If $X = \{v\}$, we simply write $d_D^-(v)$ instead of
 50 $d_D^-(\{v\})$ (omitting braces). For an integer n , we denote by $[n]$ the set $\{1, \dots, n\}$.

51 A *network* $\mathcal{N} = (D, u)$ is formed by a digraph $D = (V, A)$ with a *capacity function*
 52 $u : A(D) \rightarrow \mathbb{Z}_+$. If all arcs in D have capacity λ , we simply write $u \equiv \lambda$. A *flow* in \mathcal{N} is
 53 a function $f : A(D) \rightarrow \mathbb{Z}_+$ such that $f(vw) \leq u(vw)$, $\forall vw \in A(D)$ (note that contrary to
 54 the classical definition of flows, there is no need to satisfy the conservation constraints). For
 55 a vertex $v \in V(D)$, we define $f^+(v) = \sum_{vw \in A} f(vw)$ and $f^-(v) = \sum_{wv \in A} f(wv)$, that is,
 56 $f^+(v)$ and $f^-(v)$ are the amount of flow *leaving* and *entering* v , respectively. The *balance*
 57 *vector* of a flow f in a network $\mathcal{N} = (D, u)$ is the function $b_f : V(D) \rightarrow \mathbb{Z}$ associating each
 58 vertex $v \in V(D)$ to the value $f^+(v) - f^-(v)$. If $b_f(v) = \ell$, we may also say that v has *balance*
 59 ℓ . A vertex $s \in V(D)$ is a *source* of f if $f^+(s) > 0$ and $f^-(s) = 0$. An (s, t) -*cut* on a network
 60 $\mathcal{N} = (D, c)$ is a bipartition $(S, \bar{S} = V(D) \setminus S)$ of $V(D)$ such that $s \in S$ and $t \in \bar{S}$. The
 61 capacity of an (s, t) -cut is the sum of the capacities of arcs with tail in S and head in \bar{S} and
 62 it is denoted by $u(S, \bar{S})$. For other concepts on graphs and digraphs, we follow standard
 63 terminology as in [3, 6].

64 Flows are widely studied as they allow, with a certain elegance and simplicity, modeling
 65 problems in different areas of study such as transportation, logistics and telecommunications.
 66 In the theoretical field, they are used to solve various problems in graphs and digraphs. A
 67 long list of results related to flows can be found in [1, 10]. In this work, we deal with one
 68 variation of the problem where, roughly speaking, k arc-disjoint flows, having in common the
 69 same source and with a prescribed balance vector, have to be found.

70 Since a flow in a network can be entirely described by its associated balance vector,
 71 the goal of a flow problem can be seen as finding a flow f whose balance vector respects
 72 a given set of properties. Therefore, the problem that has been considered is: given a
 73 digraph D together with a balanced vector $b : V(D) \rightarrow \mathbb{Z}$, does there exist a flow f in D
 74 such that $b_f = b$? For example, in the decision version of the classical MAXIMUM FLOW
 75 problem we are given a network $\mathcal{N} = (D, u)$ together with a positive integer k and a pair
 76 of vertices $s, t \in V(D)$, and the goal is to decide if \mathcal{N} admits a flow f with $b_f(s) \geq k$ such
 77 that $b_f(v) = 0$ for all $v \in V(D) \setminus \{s, t\}$ and $b_f(s) = -b_f(t)$. Such a flow f is known as an
 78 (s, t) -*flow* and, in this case, we say that f has *value* $b_f(s)$. This problem can be solved in
 79 polynomial time [11] (see [16, Chapter 10] for a survey on algorithms for MAXIMUM FLOW).
 80 Furthermore, by applying a simple reduction to MAXIMUM FLOW (see [3, Lemma 4.2.2]),
 81 we can solve in polynomial time any flow problem in which the aim is to find a flow g with
 82 $\sum_{v \in V(D)} b_g(v) = 0$.

83 The possibility of considering the coexistence of flows respecting some properties in a
 84 network gives more power of modeling for this already useful tool. In [9], a more general
 85 version of the flow problem was investigated. There, the goal was to find a collection of
 86 (s_i, t_i) -flows that sum to a specific value r and such that the sum of all the flows in each
 87 arc respects its capacity. It was shown that this version of the problem is NP-hard since it
 88 generalizes many hard problems in the literature, including the notoriously hard DIRECTED
 89 DISJOINT PATHS problem.

90 Two flows f and g in a network $\mathcal{N} = (D, u)$ are arc-disjoint if $f(vw) \cdot g(vw) = 0$,
 91 $\forall vw \in A(D)$. In [2], the problem of finding arc-disjoint flows was introduced and studied.
 92 The authors considered many constraints showing that some generalize important problems,

93 known to be hard, and that there are cases where polynomial-time algorithms are possible.
 94 Amongst other results, they showed that the following are NP-complete: the problem of
 95 deciding if there are two arc-disjoint flows in a network where all arcs have capacity one, the
 96 problem of deciding if there are two arc-disjoint flows with the same balance vectors in a
 97 network where all arcs have capacity at most 2, and the problem of deciding if there are two
 98 arc-disjoint (s, t) -flows in a network where all arcs have capacity at most two.

99 We say that a digraph D is an s -branching if there is a directed path from s to every
 100 other vertex in D and the underlying¹ graph of D is a tree. We also say that D has root s .
 101 There is an extensive literature concerning the study of branchings (see for example chapter
 102 9 of [3]), given their relevance both from practical and theoretical point of view due to its
 103 numerous applications, some of which we enumerate below. A classical result by Edmonds [8]
 104 characterizes the digraphs containing k arc-disjoint s -branchings and later proofs of the same
 105 theorem done in [14, 18] show polynomial-time algorithms to find such branchings if they
 106 exist.

107 ► **Theorem 1** (Edmonds' branching theorem [8]). *A digraph $D = (V, A)$ has k arc-disjoint*
 108 *s -branchings if and only if $d_D^-(X) \geq k$ for all $X \subseteq V(D) \setminus \{s\}$ with $X \neq \emptyset$.*

109 A stronger version of this result, allowing each branching to have its own set of roots,
 110 can be seen in [16, Chapter 53]. There are many applications for Edmonds' Theorem on
 111 arc-disjoint branchings: it can be used, for example, to prove Menger's Theorem [15], to
 112 characterize arc-connectivity [17], and to characterize branching cover [12]. The problem of
 113 finding disjoint branchings was recently studied in the setting of temporal graphs in [19]. In
 114 this paper, we consider branchings in terms of flow.

115 An s -branching flow in a network $\mathcal{N} = (D, u)$ is a flow f such that $b_f(s) = n - 1$ and
 116 $b_f(v) = -1$ for every $v \in V(D) \setminus \{s\}$. In other words, f reaches all vertices of D and each
 117 vertex other than s retains one unit of flow. Here, we study the problem that, given a
 118 digraph D and a source s and an integer k , aims at deciding whether there exist k arc-disjoint
 119 s -branching flows.

120 **Related work:** Finding one s -branching flow in a given network is easy: since $\sum_{v \in V(D)} b_f(v) =$
 121 0 , we can reduce this problem to the problem of finding one (s, t) -flow, as discussed above.
 122 We can also find k arc-disjoint s -branching flows in polynomial time when $u \equiv n - 1$. In [2]
 123 it was shown that, in this case, \mathcal{N} admits an s -branching flow if and only if D contains an
 124 s -branching. Thus, applying Theorem 1, the authors provided a characterization of networks
 125 admitting k arc-disjoint s -branching flows, which are exactly the networks constructed on
 126 digraphs containing k arc-disjoint s -branchings. They also provided a polynomial time
 127 algorithm that finds such flows if they exist. We now discuss how the tractability of this
 128 problem, in general, depends on the choice of the capacity function.

129 In [2], the authors showed that the problem of deciding if a network has k arc-disjoint
 130 s -branching flows is NP-complete if every arc has capacity at most two, and in [4] this result
 131 was extended to networks with capacity at most ℓ , for every fixed $\ell \geq 2$. For most choices of
 132 larger capacities, the problem remains hard. In [4] it was shown that, unless the *Exponential*
 133 *Time Hypothesis* [13] fails, there is no polynomial time algorithm for the problem of finding
 134 k arc-disjoint s -branching flows in a network $\mathcal{N} = (D, u)$ with $u \equiv \lambda$ for any choice of λ
 135 such that $n/2 \leq \lambda \leq n - (\log n)^{1+\varepsilon}$, $\varepsilon > 0$, even if D is acyclic. In [7] the authors adapted
 136 this last proof to show that the same holds if $(\log n)^{1+\varepsilon} \leq \lambda \leq n/2$. The last two results

¹ The underlying graph of a digraph D is the undirected graph obtained by ignoring the orientation of the arcs of D .

137 have an intersection point, which is when $u \equiv n/2$. The two constructions result in the same
 138 network in this case, and it is worth noticing that it gives a polynomial time reduction. So,
 139 for $u \equiv n/2$, the problem is NP-hard. On the positive side, in [4] it was also shown that
 140 the problem is solvable in polynomial time when $\lambda = n - c$, for fixed k and $c \geq 1$. More
 141 precisely, the authors showed that the problem is XP for fixed k and c when the goal is to
 142 find k arc-disjoint s -branching flows. This was later improved to an FPT algorithm with the
 143 same parameter in [5].

144 In short, the results presented in the previous paragraph roughly say that the problem
 145 of deciding if a given network with $u \equiv n - c$ contains k arc-disjoint s -branching flows
 146 admits a polynomial time algorithm for every fixed $c \geq 1$, and is unlikely to admit such an
 147 algorithm for most other choices of u under reasonable complexity hypothesis. Thus any
 148 *general* characterization of networks admitting such flows must also be hard to verify. In the
 149 following, we extend to s -branching flows the property characterizing networks containing k
 150 arc-disjoint s -branchings, which was originally proposed in [7], .

151 For a vertex $s \in V(D)$ and two non-negative integers k, λ , we say that D is (k, λ, s) -
 152 *sufficient* if, for every $X \subseteq V(D) \setminus \{s\}$ with $X \neq \emptyset$, we have

$$153 \quad d_D^-(X) \geq k \left\lceil \frac{|X|}{\lambda} \right\rceil. \quad (1)$$

154
 155 If $\mathcal{N} = (D, u)$ with $u \equiv \lambda$ is a network constructed on such a digraph, we may also say that
 156 \mathcal{N} is (k, λ, s) -sufficient.

157 It is not hard to visualize the relationship between this property and the one in the
 158 statement of Theorem 1: if $\lambda = n - 1$, then every arc of D has enough capacity to send
 159 as many units of flow as needed to reach any set $X \subseteq V(D)$. Thus, capacities are not an
 160 issue and Inequality 1 states that $d_D^-(X) \geq k$, as in Theorem 1, since $|X| \leq n - 1$. For other
 161 choices of λ , note that every s -branching flow on \mathcal{N} must reach a set of vertices X with at
 162 least $|X|$ units of flow to cover it. Thus each s -branching flow in \mathcal{N} uses at least $\lceil |X|/\lambda \rceil$
 163 arcs to cover X . More formally, the following was proved.

164 ► **Proposition 2.** [7] *Let $\mathcal{N} = (D, u)$ be a network with $u \equiv \lambda$ and $s \in V(D)$. If \mathcal{N} admits
 165 k arc-disjoint s -branching flows, then D is (k, λ, s) -sufficient.*

166 **Our contributions:** In Section 2, we show how to construct a family of a (k, λ, s) -sufficient
 167 networks not containing k arc-disjoint s -branching flows. On the positive side, we show that
 168 some positive cases of sufficiency do occur depending on the structure of D and on the choice
 169 of k and λ .

170 In Section 3, we show that (k, λ, s) -sufficiency guarantees the existence of k arc-disjoint
 171 s -branching flows in a network $\mathcal{N} = (D, u)$ when D is an s -branching where parallel arcs are
 172 allowed, and when D is a collection of internally disjoint multi-paths starting at s and all
 173 ending on the same vertex $t \in V(D)$. In other words, the flows are guaranteed to exist when
 174 the underlying simple graph of D is a tree or a spindle, and thus the considered property
 175 characterizes the existence of the flows in those classes of networks. Additionally, we show
 176 that we also have this guarantee in general digraphs when $k = 1$, when $\lambda = 1$, and when
 177 $\lambda = |V(D)| - 2$. Finally, we show that, for a network $\mathcal{N} = (D, u)$ with $u \equiv \lambda$ and an integer
 178 p satisfying $p\lambda \geq n - 1$, if $d_D^-(X) \geq pk$ for all non-empty $X \subseteq V(D) \setminus \{s\}$, then \mathcal{N} admits k
 179 arc-disjoint s -branching flows.

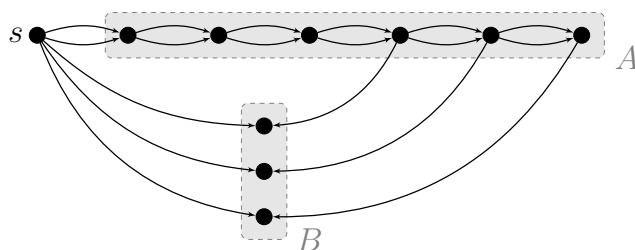
180 In Section 4, we show that the analyzed property is not enough to improve the tractability
 181 of the problem of finding the desired flows. Namely, we show that it is NP-complete to decide

182 if a network \mathcal{N} admits k arc-disjoint s -branching flows even if its underlying simple graph is
 183 constructed by appending a path to the end of a spindle and we know beforehand that \mathcal{N}
 184 is (k, λ, s) -sufficient. Since this property can be easily verified in this class and it is always
 185 necessary for the existence of the flows, it is no surprise that it is not enough to guarantee
 186 the existence of the k arc-disjoint s -branching flows.

187 In Section 5 we close the paper with some open questions. We remark that polynomial
 188 time algorithms to find the flows can be extracted from the provided constructive proofs of
 189 the aforementioned results.

190 **2 Networks on (k, λ, s) -sufficient digraphs not containing the flows**

191 In Figure 1 we give an example of a $(2, n - 3, s)$ -sufficient network that does not contains 2
 192 arc-disjoint s -branching flows, and we show that it can be generalised to $(2, n - c, s)$ -sufficient
 193 networks, for every fixed $c \geq 3$ (Theorem 3). Thus the best that we can expect is that every
 194 $(2, n - 2, s)$ -sufficient network contains the desired s -branching flows. In Section 3 we show
 that this is indeed true for general networks (c.f. Theorem 13).



195 **Figure 1** Example of a network on a $(2, n - 3, s)$ -sufficient digraph that does not contain 2
 196 arc-disjoint s -branching flows. Here every arc has capacity $n - 3$.

195

196 **Theorem 3.** For every $c \geq 3$, there exists a $(2, n - c, s)$ -sufficient network \mathcal{N} which does
 197 not admit 2 arc-disjoint s -branching flows.

198 **Proof.** Let \mathcal{N} be a network as in Figure 1, considering that B has exactly c vertices and
 199 $|A| = n - c - 1$, with each of the last c vertices of A connected to a different vertex of B .
 200 First, we show that \mathcal{N} is $(2, n - c, s)$ -sufficient. For any set $X \subseteq V(D) \setminus \{s\}$, let $a = |X \cap A|$
 201 and $b = |X \cap B|$. Thus, $|X| = a + b$.

202 If $|X| \leq n - c$, then we must show that $d^-(X) \geq 2 \lceil \frac{|X|}{n-c} \rceil = 2$. If $a > 0$, one can see that
 203 there are two arcs entering $X \cap A$, then $d^-(X) \geq 2 + b \geq 2$. Otherwise, we have that $b > 0$
 204 and $d^-(X) = 2b \geq 2$.

205 If $|X| > n - c$, then we must show that $d^-(X) \geq 2 \lceil \frac{|X|}{n-c} \rceil = 4$. Note that in this case
 206 $a > 0$ and $b \geq 2$. So, we have $d^-(X) \geq 2 + b \geq 4$.

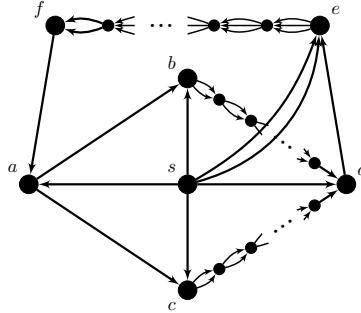
207 Now we show that \mathcal{N} does not admit 2 arc-disjoint s -branching flows. Observe that each
 208 flow can use at most one arc from A to B , by the size of A and the capacity of the arcs
 209 (specially the arcs leaving s to A). So, there is a vertex v in B that must be reached by each
 210 flow through an arc coming from s , but there is only one arc from s to v . ◀

211 Next, we show how to construct a family of (k, λ, s) -sufficient networks that do not contain
 212 k arc-disjoint s -branching flows.

213 **Theorem 4.** For all $\lambda \geq 2$ and for all even $k \geq 2$, there exists a (k, λ, s) -sufficient network
 214 $\mathcal{N} = (D, u)$, with $u \equiv \lambda$, that does not admit k arc-disjoint s -branching flows.

23:6 A study of an Edmonds' like property to arc-disjoint branching flows

215 **Proof.** Given an even $k \geq 2$ and $\lambda \geq 2$, start the construction of \mathcal{N} by adding the vertices
 216 s, a, b, c and d then add an arc from s to every other vertex along with the arcs $ab, ac, bd,$ and
 217 cd . Subdivide the arc bd (cd) $\lambda - 2$ times and let B (C) be the union of $\{b\}$ ($\{c\}$) and the set
 218 of vertices obtained by the subdivisions of bd (cd) (note that if $\lambda = 2$ nothing changes at this
 219 step, which means that $B = \{b\}$ and $C = \{c\}$), and then double the arcs between vertices of
 220 B (C). Let P be a path with λ vertices and denote by e and f , respectively, the first and
 221 last vertex of P . Triple the first $\lambda - 2$ arcs of P and double the last one (the one that enters
 222 f). After that, add P to D along with the arcs de, fa and two parallel arcs se . To complete
 223 the construction, replace each arc by $k/2$ copies of itself and set $u \equiv \lambda$. See Figure 2.



■ **Figure 2** Counter-example for $k = 2$ and $\lambda \geq 2$

224 Here, we say that a set $X \subseteq V(D)$ is *satisfied* if $d_D^-(X) \geq k \lceil |X|/\lambda \rceil$.
 225 To prove that D is (k, λ, s) -sufficient, we consider the case where $k = 2$ because for a
 226 larger k we multiply the in-degree of each subset of $V(D) - s$ by $k/2$. Observe that, except
 227 for s , every vertex has in-degree at least 2 in D and that for every $X \subseteq V(D) - s$ each vertex
 228 from $\{a, b, c, d\}$ which belongs to X sums at least one unit to its in-degree and the vertex e
 229 sums to two. Consider a set $X \subseteq V(D) - s$. If $D[X]$ has a cycle, then $d_D^-(X) \geq 5$, once X
 230 necessarily contains a, d, P and either B or C (or both), but its also true that $|X| \geq 2\lambda + 1$,
 231 which means that should be at least one extra arc entering in X for it to be satisfied. To
 232 verify the existence of this extra arc assume without loss of generality that $B \subset X$ (the
 233 case where $C \subset X$ is symmetric) and take the longest (z, d) -path in $D[X]$ such that $z \in C$.
 234 If no such path exists, there is at least one arc going from C to d which was not counted
 235 before and if the path exists, then either $z = c$ or the two arcs yz enters X , where y is the
 236 in-neighbor of z which, by the choice of z , cannot be in X .
 237 Now consider that $D[X]$ is acyclic. Note that each source in $D[X]$ contributes with at
 238 least two units in the in-degree of X . Since each weak component of $D[X]$ has at least
 239 one source, then every X with $|X| \leq \lambda$ is satisfied. Similarly, the same also holds when
 240 $\lambda < |X| \leq 2\lambda$ if $D[X]$ has more than one weak component or only one component with many
 241 sources. Otherwise $D[X]$ has only one weak component W with a single source w , and we
 242 have 5 cases:
 243 ■ $w \in B$. Observe that in this case $\{d, e\} \subset X$, once that $|X| \geq \lambda + 1$ and $|B| = \lambda - 1$,
 244 which means that $d_D^-(X) \geq 5$.
 245 ■ $w \in C$. It's analogous to the previous one.
 246 ■ $w = a$. In this case, there are at least two vertices from $\{b, c, d\}$ in X . That is, $d^-(X) \geq 4$.
 247 ■ $w = d$. If d is the source, then $e \in X$, and thus $d_D^-(X) \geq 5$.
 248 ■ $w \in V(P)$. Except for f , every vertex of P has in-degree three and since X must contains
 249 a , then $d_D^-(X) \geq 4$. When $w = f$, besides a , X must also contains at least one vertex
 250 from $\{b, c\}$ which guaranties $d_D^-(X) \geq 4$.

251 When $2\lambda + 1 \leq |X| \leq 3\lambda$, the in-degree of X must be at least 6. We start with the case
 252 that $D[X]$ has just one weak component. If w is a source in $D[X]$, again we have 5 subcases:
 253 ■ $w \in B$. In this case, since $|B \cup \{d\}| = \lambda$, either there is a vertex in C which is also a source
 254 in $D[X]$ or $a \in X$. In both situations, $d, e \in X$ because $|B \cup C| < 2\lambda$. Thus, $d_D^-(X) \geq 6$.
 255 ■ $w \in C$. It's analogous to the previous one.
 256 ■ $w = a$. Once $|\{a, d\} \cup B \cup C| = 2\lambda$, if a is a source, there is a path in $D[X]$ that starts in
 257 a , goes through B or C , d and ends in some vertex of P .
 258 ■ $w = d$. When d is a source, for sure $\{a, b, c, d, e\} \subset X$.
 259 ■ $w \in V(P)$. Here, in order to avoid cycles, we know that if $w = f$, necessarily a, b, c and d
 260 are also included in X and if $w \neq f$, at least a, b and c are included in X but, in this
 261 case, w already contributes with 3 for the in-degree of X . Thus $d_D^-(X) \geq 6$.

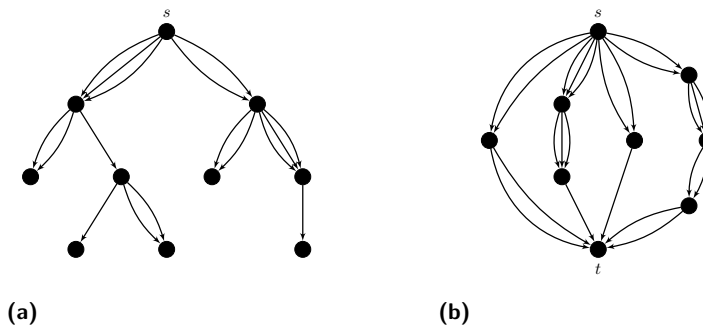
262 If $D[X]$ has more than 2 weak components we already know that $d_D^-(X) \geq 6$. If $D[X]$
 263 has two weak components Y, Z , one of them, say Y , has at least $\lambda + 1$ vertices and hence, as
 264 in the previous cases, $d_D^-(Y) \geq 4$. Thus, $d^-(X) \geq 6$.

265 Notice that if there are k arc-disjoint s -branching flows x_1, x_2, \dots, x_k in \mathcal{N} , as $d_D^-(B) =$
 266 $d_D^-(C) = k$, each one of these k arcs has to be used by a different flow, and carry at least
 267 $\lambda - 1$ units of flow, that is, one unit for each vertex of B and C . Now observe that $k/2$ of the
 268 arcs which enter in B come from a and the same for the ones that enter C , and then a must
 269 receive λ units of flow on each of its k incoming arcs in order to send the proper amount
 270 of flow to B and C . Assume without loss of generality that the $k/2$ copies of sa are going
 271 to be used by $x_1, x_2, \dots, x_{k/2}$ and the $k/2$ copies of fa are left for $x_{k/2+1}, x_{k/2+2}, \dots, x_k$.
 272 Since the in-degree of f is also equal to k , f can receive at most λ units of flow of each
 273 x_1, x_2, \dots, x_k and hence it can send at most $\lambda - 1$ units of any of these flows to a . Therefore,
 274 \mathcal{N} does not admits k arc-disjoint s -branching flows. ◀

275 **3 Networks on (k, λ, s) -sufficient digraphs containing the flows**

276 In this section, we show some positive results regarding (k, λ, s) -sufficiency. We remark
 277 that the constructive proofs of those results can be naturally adapted to polynomial-time
 278 algorithms to find the desired flows in those networks.

279 We say that a digraph $D = (V, A)$ is a *multi-branching* if D is an out-branching when
 280 we ignore its parallel arcs and, for $\ell \geq 1$, we denote by \mathcal{D}_ℓ the class of *multi-spindles*. That
 281 is, the class of digraphs formed by a source vertex s , a sink vertex t , ℓ pairwise internally
 282 vertex-disjoint multi-paths P_1, \dots, P_ℓ from s to t , each with $p_i \geq 1$ internal vertices, $1 \leq i \leq \ell$,
 283 respectively. See Figures 3a and 3b for an example of a multi-branching and an example of a
 284 multi-spindle, respectively.



■ **Figure 3** Examples of multi-branching (a) and multi-spindle (b)

23:8 A study of an Edmonds' like property to arc-disjoint branching flows

285 ► **Theorem 5.** *Let D be a (k, λ, s) -sufficient multi-branching. Then the network $\mathcal{N} = (D, u)$,*
 286 *with $u \equiv \lambda$, admits k arc-disjoint s -branching flows.*

287 The following result is useful for proving Theorem 5.

288 ► **Lemma 6.** *Let D be a (k, λ, s) -sufficient digraph and let W be a nonempty subset of*
 289 *$V(D) \setminus \{s\}$ such that $N_D^+(V(D) \setminus W) = \{w\}$, $w \in W$. Then $D[W]$ is (k, λ, w) -sufficient.*

290 **Proof.** Since w is the unique vertex of W which has in-neighbours in $V(D) \setminus W$, then we
 291 have $d_{D[W]}^-(v) = d_D^-(v)$ for every $v \in W \setminus \{w\}$. Thus, for every $X \subseteq W \setminus \{w\}$ we have
 292 $d_{D[W]}^-(X) = d_D^-(X)$. ◀

293 **Proof of Theorem 5.** We are going to use induction on the height h of the multi-branching
 294 D with root s to construct k arc-disjoint s -branching flows f_1, f_2, \dots, f_k in \mathcal{N} .

295 The base case is $h = 1$. In this case D is a star and, since it is (k, λ, s) -sufficient, there
 296 are at least k arcs from s to every other vertex and we can use one arc for each flow f_j , for
 297 $j \in [k]$.

298 Induction step: Suppose that the statement holds for multi-branchings of height $h < q$.
 299 Assume that D has height q and let r_1, r_2, \dots, r_p be the out-neighbours of s in D .

300 Let B_{r_i} be the subgraph of B_s that is a multi-branching with root r_i , for each $i \in [p]$.
 301 Observe that, by Lemma 6, B_{r_i} is (k, λ, r_i) -sufficient and has height $h_i \leq q - 1$, thus, by
 302 induction hypothesis, B_{r_i} has k arc-disjoint r_i -branching flows $f_1^i, f_2^i, \dots, f_k^i$. We know that
 303 $d^-(r_i) \geq k \lceil \frac{|B_{r_i}|}{\lambda} \rceil$ and then we can use $\lceil \frac{|B_{r_i}|}{\lambda} \rceil$ of these arcs to send the proper amount of
 304 flow from s to r_i on each x_j to complete the k s -branching flows. ◀

305 Since a multi-path is also a multi-branching, Theorem 5 generalizes the result of [7] for
 306 multi-paths.

307 ► **Theorem 7.** *If D is a (k, λ, s) -sufficient digraph in \mathcal{D}_ℓ , then the network $\mathcal{N} = (D, u)$ with*
 308 *$u \equiv \lambda$ admits k arc-disjoint s -branching flows.*

309 **Proof.** Let D be a digraph in \mathcal{D}_ℓ that is (k, λ, s) -sufficient. By definition, D is composed
 310 by the multi-paths P_1, P_2, \dots, P_ℓ and $V(P_i) \cap V(P_j) = \{s, t\}$ for every $1 \leq i < j \leq \ell$. We
 311 denote the p_i internal vertices of P_i by $v_{p_i}^i, v_{p_i-1}^i, \dots, v_2^i, v_1^i$ in this order, for every $i \in [\ell]$. For
 312 every $j \in [p_i]$, we define $r_j^i \geq 0$ so that $d_D^-(v_j^i) = k \lceil j/\lambda \rceil + r_j^i$. Remark that $d_D^-(v_j^i) \geq k \lceil j/\lambda \rceil$,
 313 otherwise the set $\{v_j^i, v_{j-1}^i, \dots, v_1^i\}$ would contradict the fact that D is (k, λ, s) -sufficient,
 314 since the only arcs entering in it are those arriving at v_j^i . Considering only the vertices with
 315 an index multiple of λ , that is, the vertices $v_{j\lambda}^i$ for every $1 \leq j \leq \lfloor p_i/\lambda \rfloor$, let

$$316 \quad r(P_i) = \begin{cases} k, & \text{if } p_i < \lambda, \\ \min\{r_{j\lambda}^i \mid 1 \leq j \leq \lfloor p_i/\lambda \rfloor\}, & \text{otherwise.} \end{cases}$$

317 For all $i \in [\ell]$, we define e_i as the number of arcs from v_1^i to t and $m_i = \min\{r(P_i), e_i\}$.
 318 Let $k' = \sum_{i=1}^{\ell} m_i$.

319 Assume without loss of generality that P_1, P_2, \dots, P_q are the paths on which $r(P_i) \leq e_i$, for
 320 $i \in [q]$. Then $k' = \sum_{i=1}^q r(P_i) + \sum_{i=q+1}^{\ell} e_i$. Now, for every $i \in [q]$, we choose a $j_i \in [\lfloor p_i/\lambda \rfloor]$
 321 which satisfies $d_D^-(v_{j_i\lambda}^i) = k j_i + r(P_i)$ and we define $X_i = \{v_{j_i\lambda}^i, v_{(j_i\lambda)-1}^i, \dots, v_1^i\}$. Consider
 322 the set $X = \bigcup_{i=1}^q X_i \cup \{t\}$. We have that

$$323 \quad d_D^-(X) = \sum_{i=1}^q d_D^-(X_i) + \sum_{i=q+1}^{\ell} e_i = k \sum_{i=1}^q j_i + \sum_{i=1}^q r(P_i) + \sum_{i=q+1}^{\ell} e_i = k \sum_{i=1}^q j_i + k'. \quad (2)$$

324 Since D is (k, λ, s) -sufficient and $|X_i| = j_i \lambda$

$$325 \quad d_D^-(X) \geq k \left\lceil \frac{|X|}{\lambda} \right\rceil = k \left\lceil \frac{1}{\lambda} + \frac{\lambda \sum_{i=1}^q j_i}{\lambda} \right\rceil = k \sum_{i=1}^q j_i + k. \quad (3)$$

326 Combining (2) and (3) we conclude that $k' \geq k$. To finally construct the k arc disjoint
327 branching flows, we need the following claim.

328 \triangleright **Claim 8.** There are k arc-disjoint flows x_1^i, \dots, x_k^i on the network $\mathcal{N}_i = (P_i, u)$, with
329 $u \equiv \lambda$, such that $x_1^i, \dots, x_{m_i}^i$ are branching flows in \mathcal{N}_i and $x_{m_i+1}^i, \dots, x_k^i$ are branching
330 flows in $\mathcal{N}_i - t$.

331 *Proof.* We show how to construct the flows $x_1^i, x_2^i, \dots, x_k^i$. For every $j \in [p_i]$, the vertex v_j^i
332 must receive $j + 1$ units of flow on the first m_i flows and j units on remaining $k - m_i$ flows.
333 Since $d_D^-(v_j^i) = d_{P_i}^-(v_j^i) \geq k \lceil j/\lambda \rceil$, we can use a distinct group of $\lceil j/\lambda \rceil$ arcs to send j units
334 of flow on each one of the k flows and we only have to argue how to send the extra unit
335 for the flows $x_1^i, \dots, x_{m_i}^i$. If j is a multiple of λ , then there are at least another $r(P_i)$ extra
336 arcs entering v_j^i and since $m_i \leq r(P_i)$ they can be used to send the extra units (note that, if
337 $r(P_i) = 0$ then $m_i = 0$ and no extra unit is needed). Otherwise, j is not multiple of λ and
338 when we send j units of flow through $\lceil j/\lambda \rceil$ arcs, there is an arc that will not be used in
339 the maximum capacity and so we can use it to send the extra unit. Applying this method
340 iteratively from $v_{p_i}^i$ to v_1^i , we arrive at v_1^i with 2 units of flow on the flows $x_1^i, \dots, x_{m_i}^i$ and we
341 can send the extra unit of each flow to t because $m_i \leq e_i$. \triangleleft

342 For each $i \in [\ell]$, we compute the flows $x_1^i, x_2^i, \dots, x_k^i$ as in the Claim 8. Observe that k' is
343 the number of flows that reach t . If $k' = k$, then we can rename these k flows in such a way
344 that each $x_j = \bigcup_{i=1}^{\ell} x_j^i$ is an s -branching flow on \mathcal{N} . Finally, if $k' > k$, we take a flow that
345 reaches t and we modify it so that it doesn't reach t anymore, and we repeat this process
346 until there are only k flows that reach t . \blacktriangleleft

347 It is worth to notice that the above proofs for multi-branchings and multi-spindles together
348 with Proposition 2, besides giving a complete characterization of the digraphs in these classes
349 having k arc-disjoint branching flows, they lead to polynomial-time algorithms to find such
350 flows, once testing the (k, λ, s) -sufficiency for multi-branchings and multi-spindles can be
351 done in polynomial time. The algorithms work for every value of λ , even those for which the
352 problem of finding k arc-disjoint branching flows is known to be hard in general.

353 We now consider the cases of k or λ with value 1 and for that we need the three following
354 results.

355 \blacktriangleright **Theorem 9** (Max-flow min-cut). *In any network $\mathcal{N} = (D, u)$ with source s and sink t , the
356 value of a maximum flow is equal to the capacity of a minimum cut.*

357 \blacktriangleright **Theorem 10.** [2] *Let k be an integer and $\mathcal{N} = (D, u)$ be a network with $u \equiv 1$ and a
358 prescribed balance vector b such that $b \not\equiv 0$. There exist k arc-disjoint flows in \mathcal{N} , all with
359 balance vector b , if and only if \mathcal{N} has a flow f with balance vector $b_f \equiv kb$. Hence one can
360 decide the existence of these flows in polynomial time.*

361 \blacktriangleright **Lemma 11** (Adapted from [3]). *Given a network $\mathcal{N} = (D, u)$ and a prescribed balance
362 vector b . Let $M = \sum_{v: b(v) > 0} b(v)$ and let $\mathcal{N}' = (D', u')$ be a network defined as follows:*

- 363 \blacksquare $V(D') = V(D) \cup \{s', t'\};$
- 364 \blacksquare $A(D') = A(D) \cup \{s'u : u \in V(D), b(u) > 0\} \cup \{vt' : v \in V(D), b(v) < 0\};$

23:10 A study of an Edmonds' like property to arc-disjoint branching flows

365 ■ $u'(a) = u(a), \forall a \in A(D), u'(s'u) = b(u)$ if $b(u) > 0$ and $u'(vt') = -b(v)$ if $b(v) < 0$.

366 Then, \mathcal{N} admits a flow f with balance b if and only if \mathcal{N}' admits a (s', t') -flow f' with
367 value M .

368 ► **Theorem 12.** Let D be a (k, λ, s) -sufficient digraph. Then, if $\lambda = 1$ or $k = 1$, the network
369 $\mathcal{N} = (D, u)$ with $u \equiv \lambda$ admits k arc-disjoint s -branching flows.

370 **Proof.** We start by showing a result that is common to both cases. Let b_f be the following
371 balance vector: $b_f(s) = k(n - 1)$ and $b_f(v) = -k$, for all $v \in V(D) - s$, where $n = |V(D)|$.
372 Let \mathcal{N}' be the network obtained from \mathcal{N} and b_f , as described in Lemma 11, and (S, \bar{S}) be
373 a minimum (s', t') -cut in \mathcal{N}' . We show that $u(S, \bar{S}) = k(n - 1)$ which, by the Theorem 9,
374 implies that there is an (s', t') -flow with value $k(n - 1)$ in \mathcal{N}' . Remark that, by construction
375 of \mathcal{N}' , $u'(s's) = k(n - 1)$, $u'(vt') = k$ for every $v \in D \setminus s$ and $u'(a) = \lambda$ for every $a \in A(D)$.
376 It is easy to see that $u(S, \bar{S}) \leq k(n - 1)$ because $u(s', V(D) - s') = k(n - 1)$. Then, we
377 might assume that $s \in S$ and we define $X = \bar{S} \setminus \{t'\}$. It follows that

$$378 \quad u(S, \bar{S}) = u(S, X) + u(S, \{t'\}) = \lambda d_D^-(X) + k(n - |X| - 1). \quad (4)$$

379 Case $\lambda = 1$. Since D is $(k, 1, s)$ -sufficient, $d_D^-(X) \geq k|X|$ and resuming (4), we have
380 $u(S, \bar{S}) \geq k|X| + k(n - |X| - 1) = k(n - 1)$. Thus, $u(S, \bar{S}) = k(n - 1)$ and, by the Lemma 11,
381 \mathcal{N} admits a flow with balance b_f . The result follows from Theorem 10 which states that \mathcal{N}
382 admits k arc-disjoint s -branching flows if and only if it admits a single flow f with balance
383 b_f .

384 Case $k = 1$. Here, D is $(1, \lambda, s)$ -sufficient, which means that $d_D^-(X) \geq (|X|/\lambda)$, and so
385 $\lambda d_D^-(X) \geq |X|$. Replacing this in (4), we obtain $u(S, \bar{S}) \geq |X| + n - |X| - 1 = n - 1$. Thus,
386 $u(S, \bar{S}) = k(n - 1)$. Similarly to the previous case, we conclude, by the Lemma 11, that \mathcal{N}
387 admits a flow f with balance b_f . As $k = 1$, we have that f is a branching flow and the result
388 follows. ◀

389 We also show that networks on $(2, n - 2, s)$ -sufficient digraphs contain 2 arc-disjoint
390 s -branching flows. This is a slightly improvement on the result by [2] when $k = 2$, which
391 states that Edmonds' characterizaion of digraphs containing 2 arc-disjoint s -branchings also
392 characterizes the existence of 2 arc-disjoint s -branching flows.

393 ► **Theorem 13.** Let $\mathcal{N} = (D, u)$ be a $(2, n - 2, s)$ -sufficient network with $u \equiv n - 2$. Then
394 \mathcal{N} admits 2 arc-disjoint s -branching flows.

395 **Proof.** Since $\mathcal{N} = (D, u)$ is a $(2, n - 2, s)$ -sufficient network, then for every $X \subset V(D) \setminus \{s\}$
396 we have $d_D^-(X) \geq 2$, and for $X = V(D) \setminus \{s\}$ we have $d_D^-(X) \geq 4$. By Edmonds' Theorem,
397 D has two arc-disjoint s -branchings. Let us call these s -branchings B^b and B^g and let us
398 colour blue the arcs used by B^b , green the arcs used by B^g , and black the arcs of D not used
399 by any of those branchings (observe that every vertex in $V(D) \setminus \{s\}$ is the head of exactly
400 one blue and exactly one green arc). The idea is to construct the desired s -branching flows
401 using the arcs of B^b and B^g . This will be possible unless we are forced to send $n - 1$ units
402 of flow through an arc, which can only happen from an arc that leaves s . In this particular
403 case, we show how to modify the branchings to avoid the problem.

404 Let r_1^b, \dots, r_ℓ^b be the out-neighbours of s in B^b and let B_1^b, \dots, B_ℓ^b be the maximal sub-
405 branchings of $B^b \setminus \{s\}$ with roots r_1^b, \dots, r_ℓ^b , respectively. If $\ell > 1$, then we can construct a
406 branching flow f_b from B^b by sending $|B_i^b|$ units of flow through sr_i^b , for $1 \leq i \leq \ell$, which
407 is possible because $u \equiv n - 2$. The same applies to B^g and we will exchange b to g in the
408 above notation when it concerns B^g in the rest of the proof.

409 If s has only one out-neighbour r^b in B^b , but there is a black arc sv , we can construct
 410 f_b from $B^b \cup sv$. We have to consider two cases: if $v \neq r^b$, let $uv \in A(B^b)$ and let B_v^b be
 411 the maximal connected subgraph of $B^b \setminus \{u\}$ with root v . Then we send $|B_v^b|$ units of flow
 412 through sv and $|B^b \setminus \{s\}| - |B_v^b|$ through sr^b . Otherwise, $v = r^b$ and we have two copies of
 413 the arc sv . So we can send 1 unit of flow through one and $n - 2$ through the other. In both
 414 cases, for the remaining blue arcs, we just need to adjust the proper amount of flow. The
 415 same applies to B^g .

416 Thus we can assume w.l.o.g. that every out-neighbour of s in D is either blue or green
 417 and, from such arcs, exactly one is blue (sr^b) and the remaining are green. Notice that, since
 418 $d_D^-(V(D) \setminus \{s\}) \geq 4$, there must be at least three green arcs from s to $D \setminus \{s\}$.

419 Since B^g is a s -branching, r^b appears in only one B_i^g , and let us consider a r_j^g for which
 420 $B_j^g \neq B_i^g$. Except for sr_j^g , there are no green arcs entering or leaving B_j^g . Then, let E be the
 421 set of blue arcs entering $V(B_j^g)$ together with all blue arcs between vertices in $V(B_j^g)$. We
 422 now recolour blue the arcs in $A(B_j^g) \cup sr_j^g$ and yellow the arcs in E . After this recolouring,
 423 a vertex $w \neq s$ is the head of exactly one blue arc and one yellow arc if it is in $V(B_j^g)$,
 424 otherwise it is still the head of exactly one blue and one green arc. This means that the
 425 vertices of $V(D)$ along with the blue arcs will form a s -branching $B^{b'}$. In particular, there
 426 are two blue arcs leaving s in $B^{b'}$ and we can construct the branching flow $f_{b'}$ as before.

427 We now recolour the arcs of E green. Again, we have the guarantee that every vertex in
 428 B_j^g has exactly one green arc arriving at it (the former yellow arcs), and, since the green arcs
 429 outside $A(B_j^g) \cup sr_j^g$ remain unaltered, the new set of all green arcs also form a s -branching
 430 $B^{g'}$. Once more, we have at least two arcs leaving s in $B^{g'}$, and we can construct $f_{g'}$ as
 431 before. ◀

432 We remark that, by a result of [4], deciding if a network with $u \equiv n - 2$ admits arc-disjoint
 433 flows can be done in polynomial time. Since Theorem 13 completes the proof that a network
 434 is $(2, n - 2, s)$ -sufficient if and only if it admits 2 arc-disjoint s -branching flows (together
 435 with Proposition 2), we get that it can be decided in polynomial time whether a digraph is
 436 $(2, n - 2, s)$ -sufficient.

437 We end this section showing that, for particular choices of $d_D^-(X)$ and λ , a simple stronger
 438 condition than the one presented in the statement of Proposition 2 guarantees the existence
 439 of k arc-disjoint s -branching flows in a given network.

440 ▶ **Theorem 14.** *Let $\mathcal{N} = (D, u)$ be a network with $u \equiv \lambda$ and $\lambda \geq 2$, and p be an integer
 441 such that $p\lambda \geq n - 1$. If $d_D^-(X) \geq pk$ for all $X \subseteq V(D) \setminus \{s\}$ with $X \neq \emptyset$, then \mathcal{N} admits k
 442 arc-disjoint s -branching flows.*

443 **Proof.** By Theorem 1, there are pk arc-disjoint s -branchings B_1, \dots, B_{pk} in D . For $i \in [k]$,
 444 let D_i be the digraph with $V(D_i) = V(D)$, with $A(D_i) = A(B_{(i-1)p+1}) \cup \dots \cup A(B_{ip})$, and
 445 $\mathcal{N}_i = (D_i, u)$.

446 Now, for every non-empty $X \subseteq V(D) \setminus \{s\}$, we conclude that $d_{D_i}^-(X) \geq p$ since each B_i
 447 with $i \in [pk]$ must reach X at least once. Moreover, by our choice of p , we have that every
 448 \mathcal{N}_i is $(1, \lambda, s)$ -sufficient. Thus, by Theorem 12, we have that each \mathcal{N}_i admits an s -branching
 449 flow x_i . Finally, as every arc of D appears in at most one digraph D_i , we conclude that the
 450 s -branching flows x_1, \dots, x_k are pairwise arc-disjoint. The result follows since each of these
 451 flows can be promptly used to construct an s -branching flow in \mathcal{N} , by copying arcs and flow
 452 functions used by each flow x_j with $j \in [k]$. ◀

453 **4 Hardness of finding the flows in (k, λ, s) -sufficient networks**

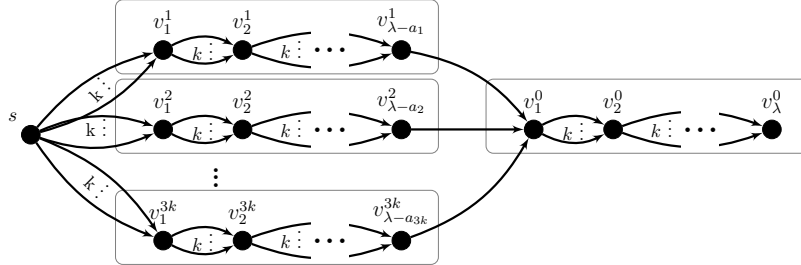
454 We showed in Sections 2 and 3 that there are networks $\mathcal{N} = (D, u)$, with D being (k, λ, s) -
 455 sufficient, which have k arc-disjoint s -branching flows and others that do not have such flows.
 456 We believe it is not always computationally easy to check the (k, λ, s) -sufficiency of a digraph,
 457 but, even if we know that D has such property, it is hard to decide if the desired flows exist,
 458 even if the network is constructed on a digraph formed by appending a path to the sink
 459 vertex of a multi-spindle.

460 **► Theorem 15.** *Given two integers k and λ it is NP-complete to decide whether a (k, λ, s) -*
 461 *sufficient network $N = (D, u)$ with $u \equiv \lambda$ admits k arc-disjoint s -branching flows.*

462 **Proof.** We are going to reduce the 3-PARTITION problem to our problem. Given a set of
 463 natural numbers $S = \{a_1, a_2, \dots, a_{3k}\}$ and a positive integer λ such that $\lambda/4 < a_i < \lambda/2$, for
 464 each $a_i \in S$, and such that $\sum_{i=1}^{3k} a_i = k\lambda$, the 3-PARTITION problem consists in deciding the
 465 following: can S be partitioned in k subsets S_1, S_2, \dots, S_k so that $\sum_{a_j \in S_i} a_j = \lambda$, $1 \leq i \leq k$?
 466 Observe that each S_i should be composed by exactly 3 elements of S .

467 From S , we construct a network $\mathcal{N} = (D, u)$ such that D is a (k, λ, s) -sufficient network
 468 that admits k arc-disjoint branching flows x_1, x_2, \dots, x_k if and only if S is a *yes* instance of
 469 the 3-PARTITION problem.

470 We begin the construction by adding a vertex s , a multi-path $P_0 = (v_1^0, v_2^0, \dots, v_\lambda^0)$ and,
 471 for every $i \in [3k]$, multi-paths $P_i = (v_1^i, v_2^i, \dots, v_{\lambda-a_i}^i)$, each one with k parallel arcs between
 472 consecutive vertices. Also, for every $i \in [3k]$, we add one arc $v_{\lambda-a_i}^i v_1^0$ and k parallel arcs sv_1^i .
 473 See Figure 4. Finally, we set $u \equiv \lambda$ to conclude the construction.



474 **■ Figure 4** The network $\mathcal{N} = (D, c)$ constructed from an instance of the 3-PARTITION problem

474 First, let us check that D is (k, λ, s) -sufficient. By contradiction, assume that there
 475 exists a set $X \subseteq V(D) - s$ such that $d_D^-(X) < k \lceil |X|/\lambda \rceil$. Note that $X \neq V(D) - s$ since
 476 $|V - s| = (2k + 1)\lambda$ and $d_D^-(V - s) = k(3k)$. Recall that the in-degree of every vertex of P_i
 477 is k and thus the in-degree of X is at least k . Therefore X cannot be entirely contained
 478 in a unique P_i , $i \in [3k]$, once, in this case, $|X| \leq \lambda$ and then $\lceil |X|/\lambda \rceil = 1$. Similarly,
 479 X cannot be entirely contained in P_0 , since there are $3k$ arcs arriving in v_1^0 . Now, let
 480 $I = \{i : 1 \leq i \leq 3k, V(P_i) \cap X \neq \emptyset\}$. Observe that $d_D^-(X) \geq k|I|$. We are going now to
 481 consider the following cases:

- 482 (i) $V(P_0) \cap X = \emptyset$. Then $|X| \leq \lambda|I|$ and $\lceil |X|/\lambda \rceil \leq |I|$, so $d_D^-(X) \geq k|I| \geq k \lceil |X|/\lambda \rceil$, a
 483 contradiction to the choice of X .
- 484 (ii) $X \cap V(P_0) \neq \emptyset$. Then $|X| \leq \lambda|I| + \lambda = \lambda(|I| + 1)$. Take the smallest j such that $v_j^0 \in X$.
 485 We analyze two possibilities:

- 486 (ii.1) $j > 1$. Then there are k arcs $v_{j-1}^0 v_j^0$ that enter X , that is, $d^-(X) \geq k|I| + k =$
 487 $k(|I| + 1) \geq k \lceil |X|/\lambda \rceil$ since $|I|$ is a positive integer and $|X| \leq \lambda(|I| + 1)$, a
 488 contradiction.
- 489 (ii.2) $j = 1$. Therefore $v_1^0 \in X$ and $d^-(X) \geq k|I|$. Again, we consider some cases:
- 490 (ii.2.1) If there are at least k different values of i for which $v_{\lambda-a_i}^i \notin X$, then
 491 such vertices contribute with k to the in-degree of X , and again we have
 492 $d^-(X) \geq k|I| + k$.
- 493 (ii.2.2) If there is only $1 \leq l < k$ different values of i for which $v_{\lambda-a_i}^i \notin X$,
 494 then, in order to complete the argument, we need to estimate more
 495 precisely the size of X . Recall that $\lambda/4 < a_i < \lambda/2$. So, $|X| \leq \lambda|I| -$
 496 $\frac{\lambda|I|}{4} + \lambda = \lambda(\frac{3|I|}{4} + 1)$. Observe that the amount of arcs entering X is
 497 $k|I| + l = \frac{3k|I|}{4} + \frac{k|I|}{4} + l$. Since, in this case, $|I| > 2k$, we have that
 498 $\frac{k|I|}{4} \geq \frac{k^2}{2} > k$ if $k > 2$, which we can consider true by the definition of
 499 the 3-PARTITION problem ($k < 3$ would be directly a negative instance).
 500 Then, $d^-(X) \geq \frac{3k|I|}{4} + k = k(\lambda(\frac{3|I|}{4} + 1)/\lambda)$, a contradiction.
- 501 (ii.2.3) If all the last vertices are in X , then $d^-(X) \geq d^-(V - s)$, and we have
 502 another contradiction, since $|X| < |V(D) - s|$.

503 As we got a contradiction on each case, it follows that D is (k, λ, s) -sufficient.

504 Now we assume that there exists k subsets S_1, S_2, \dots, S_k so that $\sum_{a_j \in S_i} a_j = \lambda$, $1 \leq i \leq k$.
 505 In order to construct the arc-disjoint s -branching flows x_1, x_2, \dots, x_k in \mathcal{N} , we establish that
 506 each of the k copies of a multiple arc belongs to a different flow.

507 For every $1 \leq i \leq k$, we construct x_i in the following way: If $a_j \notin S_i$, the arc sv_1^j of x_i
 508 will carry $\lambda - a_j$ units of flow. If $a_j \in S_i$, the arc sv_1^j of x_i will carry λ units of flow. Since
 509 $\sum_{a_j \in S_i} a_j = \lambda$, the three arcs $v_{\lambda-a_j}^j$ for $a_j \in S_i$ will send the amount of flow needed to
 510 complete the branching flow in P_0 .

511 Finally, we assume that there exist k arc-disjoint s -branching flows x_1, x_2, \dots, x_k in
 512 \mathcal{N} . We first claim that $x_r(v_{\lambda-a_i}^i v_1^0) \leq a_i$, for all $r \in [k], i \in [q]$. Since x_1, x_2, \dots, x_k are
 513 arc-disjoint, there is only one flow x_t which is positive in $v_{\lambda-a_i}^i$. If $x_t(v_{\lambda-a_i}^i v_1^0) > a_i$, then, as
 514 $|V(P_i)| = \lambda - a_i$, v_1^i must receive more than λ units of flow in x_t , that is, x_t uses at least
 515 two copies of the arc sv_1^i . This is a contradiction because there are only k copies of sv_1^i and
 516 each flow must use exactly one copy. Thus, the claim follows. Each x_r must reach v_1^0 with λ
 517 units of flow. Since $\lambda/4 < a_i < \lambda/2$, each x_r should use three arcs of the type $v_{\lambda-a_j}^j v_1^0$, and
 518 the correspondent a_j forms a S_r . Then S_1, S_2, \dots, S_k is an *yes* instance of the 3-PARTITION
 519 problem. ◀

520 Observe that, by the proof of Theorem 15, the networks constructed from negative
 521 instances of the 3-PARTITION problem are also examples that (k, λ, s) -sufficiency does not
 522 guarantee the existence of k arc-disjoint s -branching flows.

523 5 Concluding remarks

524 In this work, we studied the characterization of networks admitting k arc-disjoint s -branching
 525 flows. We showed that, in some cases (Theorems 5 and 7), an “Edmonds like” condition,
 526 which we call (k, λ, s) -sufficiency, is enough to guarantee the existence of the k arc-disjoint
 527 s -branching flows but, in other cases, a stronger statement is needed (Section 2). This
 528 proves that (k, λ, s) -sufficiency in general is not enough to guarantee the existence of the
 529 desired s -branching flows in networks, although the cases for which it does are still interesting

530 because, among other reasons, they have been resulting in polynomial-time algorithms to find
 531 the flows. Some of the positive results we prove consider restrictions on the structure of the
 532 digraphs on which the networks are built: we show that the analyzed property guarantees the
 533 existence of the flows on multi-spindles and on multi-branchings. In other words, we prove
 534 that the flows always exist in (k, λ, s) -sufficient networks built on digraphs whose underlying
 535 simple graphs are trees or spindles.

536 Another direction of this research would be to work on the characterization through a
 537 stronger condition, as cited before. Some interesting complexity questions are still open,
 538 such as a possible dichotomy in DAG's between the easy and hard cases. In other words, it
 539 would be very interesting to know if there a class of digraphs \mathcal{H} , all of which are DAGs, such
 540 that every (k, λ, s) -sufficient network constructed on a digraph $D \in \mathcal{H}$ contains k arc-disjoint
 541 s -branching flows, but that the same is not true for every $D \notin \mathcal{H}$.

542 — References —

- 543 1 R. Ahuja. *Network flows : theory, algorithms, and applications*. Prentice Hall, Englewood
 544 Cliffs, N.J, 1993.
- 545 2 J. Bang-Jensen and S. Bessy. (Arc)-disjoint flows in networks. *Theoretical Computer Science*,
 546 526:28–40, 2014. doi:<https://doi.org/10.1016/j.tcs.2014.01.011>.
- 547 3 J. Bang-Jensen and G. Z. Gutin. *Digraphs: theory, algorithms and applications*. Springer
 548 Science & Business Media, Second edition, 2008.
- 549 4 J. Bang-Jensen, F. Havet, and A. Yeo. The complexity of finding arc-disjoint branching flows.
 550 *Discrete Applied Mathematics*, 209:16–26, 2016. doi:[https://doi.org/10.1016/j.dam.2015.](https://doi.org/10.1016/j.dam.2015.10.012)
 551 10.012.
- 552 5 S. Bessy, F. Hörsch, A.K. Maia, D. Rautenbach, and I. Sau. FPT algorithms for packing
 553 k -safe spanning rooted sub(di)graphs, 2021. arXiv:2105.01582.
- 554 6 J. A. Bondy and U. S. R. Murty. *Graph theory*, volume 244 of *Graduate Texts in Mathematics*.
 555 Springer, New York, 2008. URL: <http://dx.doi.org/10.1007/978-1-84628-970-5>, doi:10.1007/
 556 978-1-84628-970-5.
- 557 7 J. Costa, C. Linhares Sales, R. Lopes, and A.K. Maia. Um estudo de redes com fluxos
 558 ramificados arco-disjuntos. *Matemática Contemporânea*, 46:230–238, 2019.
- 559 8 J. Edmonds. Edge-disjoint branchings. *Combinatorial Algorithms*, 1973.
- 560 9 S. Even, A. Itai, and A. Shamir. On the complexity of timetable and multicommodity flow
 561 problems. *SIAM Journal on Computing*, 5(4):691–703, 1976. doi:10.1137/0205048.
- 562 10 D. R. Ford and D. R. Fulkerson. *Flows in Networks*. Princeton University Press, Princeton,
 563 NJ, USA, 1962.
- 564 11 L. R. Ford and D. R. Fulkerson. Maximal flow through a network. *Canadian Journal of*
 565 *Mathematics*, 8:399–404, 1956. doi:10.4153/CJM-1956-045-5.
- 566 12 A. Frank. Covering branchings. *Acta Scientiarum Mathematicarum (Szeged)*, 41:77–81, 1979.
- 567 13 R. Impagliazzo, R. Paturi, and F. Zane. Which problems have strongly exponential complexity?
 568 *Journal of Computer and System Sciences*, 63(4):512–530, 2001. doi:[https://doi.org/10.](https://doi.org/10.1006/jcss.2001.1774)
 569 1006/jcss.2001.1774.
- 570 14 L. Lovász. On two minimax theorems in graph. *Journal of Combinatorial Theory,*
 571 *Series B*, 21(2):96 – 103, 1976. URL: [http://www.sciencedirect.com/science/article/pii/](http://www.sciencedirect.com/science/article/pii/S0095895676900496)
 572 [0095895676900496](http://www.sciencedirect.com/science/article/pii/S0095895676900496), doi:[https://doi.org/10.1016/0095-8956\(76\)90049-6](https://doi.org/10.1016/0095-8956(76)90049-6).
- 573 15 K. Menger. Zur allgemeinen kurventheorie. *Fundamenta Mathematicae*, 10(1):96–115, 1927.
 574 URL: <http://eudml.org/doc/211191>.
- 575 16 A. Schrijver. *Combinatorial optimization: polyhedra and efficiency*, volume 24. Springer
 576 Science & Business Media, 2003.
- 577 17 Y. Shiloach. Edge-disjoint branching in directed multigraphs. *Inf. Process. Lett.*, 8(1):24–27,
 578 1979. doi:10.1016/0020-0190(79)90086-3.

- 579 **18** R. Tarjan. A good algorithm for edge-disjoint branching. *Information Processing Letters*,
580 3(2):51–53, 1974. doi:10.1016/0020-0190(74)90024-6.
- 581 **19** A. Marino V. Campos, R. Lopes and A. Silva. Edge-disjoint branchings in temporal graphs.
582 In *Proc. of the 31st International Workshop on Combinatorial Algorithms (IWOCA)*, volume
583 12126 of *LNCS*, pages 112–115, 2020. doi:10.1007/978-3-030-48966-3_9.